THE UNIVERSITY *of* EDINBURGH

Robust Loss Functions for Machine Learning in the Presence of Noisy Labels

William Toner

Doctor of Philosophy

School of Informatics

University of Edinburgh

2025

Abstract

Over the last decade, there has been a significant improvement in machine learning methods for classification, particularly in computer vision. This progress has increased the demand for large labelled datasets. However, obtaining clean, accurately labelled datasets on the required scale can be prohibitively expensive. Consequently, practitioners often resort to methods that yield larger datasets but with substantial label noise, such as web querying or crowd-sourcing. Even conventional data collection methods can introduce errors due to human fallibility, particularly in complex domains like medical imaging. While broadly beneficial, the high expressibility of neural network classifiers renders them prone to overfitting on noisy labels. This issue can severely impact model performance and generalisation, leading to significant setbacks in practical applications. The challenge of noisy labels has sparked substantial interest in developing methodologies robust to such conditions. Among various strategies for handling noisy labels, 'robust loss functions' have emerged as a favoured method due to their simplicity and effectiveness. Despite extensive research, important theoretical gaps in understanding robust loss functions persist. Moreover, a disconnect between the theory and practice of these functions remains. As a result, there is an ongoing lack of principled yet simple and computationally economical loss-based approaches for learning in the presence of noisy labels. This thesis addresses these challenges. We show how overfitting can be curbed by lower-bounding the training loss, motivating this loss-bounding policy theoretically. We derive the relevant lower bound, showing how it can be estimated via the label noise rate. We also develop a straightforward early-stopping policy that operates without knowledge of the noise rate or access to a cleanly-labelled validation set. We validate the effectiveness of our approaches through extensive experiments across various noisily-labelled benchmark datasets. We build on and generalise the existing theory of noise-tolerant loss functions, demonstrating that such loss functions are rare and do not exist for most noise models. This work provides important theoretical insights and establishes conditions under which a loss function can be considered noise-tolerant. Our final chapter describes how GANs can be conceptualised in terms of noisy labels, deriving a novel loss function and providing both theoretical insights and experimental results.

Acknowledgements

I want to thank my parents and my wife, without whose support I could not have completed this PhD.

I would also like to thank Professor Amos Storkey for his supervision and all the interesting discussions we've had throughout my doctoral studies.

Contents

Notation Table

Table 1: Glossary of Symbols used in the analysis. Each symbol is essential for understanding the mathematical framework and assumptions involved in our discussion of classification accuracy under various noise models.

Symbol	Description
c	Number of classes/labels.
$\mathcal{X}$	The data domain, a subset of $\mathbb{R}^d$.
$\mathcal{Y}$	The label space, defined as $\{1,2,3,\ldots,c\}$.
Δ	Probability simplex: The set of vectors $(p_1,p_2,\ldots,p_c)$ where each $p_i \geq 0$ and $\sum p_i = 1$.
$\boldsymbol{q}$	A probability vector representing a forecast.
$\boldsymbol{p}$	A probability vector representing ground-truth probabilities.
$\boldsymbol{q} : \mathcal{X} \to \Delta$	A probability estimator model producing a forecast at each point in $\mathcal{X}$.
$\boldsymbol{p}(y \mid x)$	The vector representing the class posterior probabilities at x, expressed as $\boldsymbol{p}(y \mid x) = (p(y=1 \mid x), p(y=2 \mid x), \ldots, p(y=c \mid x))$.
f	A classifier function mapping each point in $\mathcal{X}$ to a label in $\mathcal{Y}$.
L	The loss function used to evaluate the accuracy of predictions against actual labels.
$\boldsymbol{L}$	The vector-valued function of the loss function L, where $\boldsymbol{L}(\boldsymbol{q}) = (L(\boldsymbol{q},1),\ldots,L(\boldsymbol{q},c))$.
$R_L(\boldsymbol{q})$	The L-risk of an estimator $\boldsymbol{q}$.
$R_L(\boldsymbol{q})(x)$	The *pointwise* L-risk of an estimator $\boldsymbol{q}$ at x.
$R_L^{\eta}(\boldsymbol{q})$	The noisy L-risk of an estimator $\boldsymbol{q}$.
$\mathcal{H}$ or $\mathcal{H}_L$	The entropy function corresponding to the loss function L.
$H_L(\boldsymbol{p},\boldsymbol{q})$	The expected L-loss for a forecast $\boldsymbol{q}$ given the true label distribution $\boldsymbol{p}$.
η	The noise rate of the label noise model.
$y, \widetilde{y}$	The actual label and the noisy label, respectively.
$p(x,y)$	The joint distribution of data and labels.
$\widetilde{p}(x,\widetilde{y})$	The joint distribution of data and labels after corruption by label noise.
$p(\widetilde{y} \mid y,x)$	The noise model generating noisy labels $\widetilde{y}$ from clean labels y given x.
T	The label noise transition matrix describing the probabilities of transforming a true label into a noisy label.
$\boldsymbol{e}_k$	The standard basis vector in $\mathbb{R}^c$ where only the k^{th} element is 1, and all others are 0.

Symbol	Description
argkmax	Returns the indices of the k^{th} largest elements in vector, sequentially excluding higher ranked elements.

Chapter 1

Introduction

1.1 Introduction

Deep learning has emerged as a transformative force across various domains, heralding a new era of technological advancements. Its impact is particularly profound in the realm of supervised learning, with image classification being a standout application. This methodology has revolutionised numerous fields, including medical diagnostics, autonomous driving, retail management, astronomy, surveillance, wildlife conservation, and education. The success of supervised learning hinges on the availability of large, accurately labelled datasets. However, procuring such datasets is often costly, leading researchers to opt for more economical yet imperfect data collection methods that introduce noisy labels.

The presence of incorrect labels within training sets can severely undermine the learning process, yielding models that perform inadequately on new, unseen data. This issue is critical, as accuracy and reliability are paramount in many applications. Moreover, the high costs associated with data cleaning and the challenges of acquiring expansive datasets restrict the benefits of deep learning to larger organisations, leaving smaller entities at a disadvantage.

Amidst extensive research on machine learning algorithms capable of operating in the presence of label noise, robust loss functions have emerged as a promising yet underexplored solution. These functions aim to modify the traditional objectives used in training to reduce the likelihood of overfitting. While these methods have demonstrated

effectiveness, their operational principles and theoretical underpinnings remain inadequately understood. Moreover, existing theoretically motivated loss functions often present practical barriers to implementation due to their complexity or computational demands.

Objectives In this work, we explore robust loss functions for learning in environments characterised by noisy labels. Our goals are to elucidate why certain loss functions exhibit greater robustness, to identify the conditions under which these functions maintain their robustness, and to develop practical, theoretically-backed strategies for their use. This investigation seeks to broaden the theoretical understanding of robust loss functions and offer actionable insights that can democratise the advantages of machine learning.

1.1.1 Chapter Outline

Section 1.2 provides an overview of supervised learning with a specific focus on image classification. We discuss empirical risk minimisation framework and data collection methods, highlighting how these can introduce noisy labels. In Section 1.3, we address the challenges posed by label noise, briefly review existing strategies for managing it, and discuss the gaps in current research. In Section 1.4, we define the research objectives of this thesis, delineate their scope and limitations, and present a structured outline of the thesis.

1.2 Supervised Learning

Supervised learning is a type of learning in which one attempts to infer a mapping between an input and output domain through observing a dataset of labelled instances. The objective is to uncover the underlying function that maps inputs to outputs, enabling the model to accurately and reliably predict labels for new, unlabelled instances that come from the same distribution as the training data. The scope of supervised learning is broad, as it can be applied to a wide array of practical problems where input-output relationships need to be learned. The two primary types of supervised learning are regression, where the output is continuous, and classification, where the output is categorical and involves discrete labels.

1.2.1 Empirical Risk Minimisation

Among the techniques employed in supervised learning, **Empirical Risk Minimisation (ERM)** is particularly notable. ERM consists of four main components:

1. A **hypothesis class**, which is a collection of candidate functions from which a proposed solution is selected.
2. A **dataset of labelled instances**, used to infer the true underlying mapping between the input and output domains.
3. A **loss function** that evaluates how accurately a function from the hypothesis class predicts the outputs in the dataset.
4. An **optimisation method** that selects the function from the hypothesis class which minimises the loss on the dataset, a process termed minimising the 'empirical risk'.

1.2.2 Neural Networks

Neural networks, a prominent parametric method, offer a versatile way to define a hypothesis class for Empirical Risk Minimisation (ERM). They consist of sequentially layered parametric functions—typically linear maps combined with component-wise non-linear operations—creating a network structure. Each layer's parameter settings modify the network's overall function, making neural networks capable of representing a vast array of functions. Neural networks are 'universal approximators' (Hornik, Stinchcombe, & White, 1989), capable of closely approximating any function as their size increases. The high expressibility of neural networks means they excel in complex scenarios where the input-output relationships are intricate and the available data is abundant.

1.2.2.1 Labelled Dataset

A labelled dataset is crucial for supervised learning, serving as the foundation for models to learn the mapping from inputs to outputs. For neural networks, a large, well-labelled dataset is particularly vital due to their high expressibility and ability to model complex relationships (Goodfellow, 2016). Insufficient data can lead neural networks to **overfit**, where they learn noise and irrelevant details of the specific training dataset rather than the underlying patterns needed for generalisation. Increasing the dataset size is a simple, effective strategy to combat overfitting (Banko & Brill, 2001).

1.2.2.2 Loss Functions

The choice of a **loss function** is a critical component of empirical risk minimisation in supervised learning. A loss function measures the discrepancy between the model's predictions and the actual dataset labels, providing a quantitative basis for model training. Given a labelled dataset and a hypothesis class of candidate models, the loss function assesses the 'goodness of fit' and, hence, the suitability of each model. The effectiveness of a neural network's learning process heavily depends on the loss function's appropriateness to the specific data characteristics and the task at hand. Minimising the dataset loss over the hypothesis class should align with the desired objective; for example, in classification contexts, minimising the loss should correlate with achieving high accuracy. In the case of gradient-based optimisation, loss functions should also be selected to avoid issues like vanishing gradients.

1.2.2.3 Optimisation - SGD

Stochastic Gradient Descent (SGD) is the primary optimisation method used in empirical risk minimisation with neural networks. The process begins by selecting a differentiable loss function, which ensures that the overall loss from a batch of data can be differentiated with respect to the model parameters. SGD operates by iteratively updating model parameters to reduce the loss on randomly sampled 'minibatches' of data drawn from the training dataset. Each iteration aims to decrease the loss over these minibatches, cumulatively minimising the empirical risk or the average loss over the entire dataset. This process is facilitated by the backpropagation algorithm, which efficiently computes the gradient of the loss with respect to the model parameters (Rumelhart, Hinton, & Williams, 1986). The stochastic nature of SGD not only facilitates efficient handling of large datasets but also acts as a regulariser improving its ability to generalise to new data.

1.2.3 Image Classification

This study focuses on image classification within the spectrum of supervised learning, deliberately narrowing its scope to manage research better and maintain thematic consistency due to the pivotal role of image-based data in technology.

Significance Image classification serves as a foundational task in the field of computer vision, with significant implications across various domains such as healthcare, autonomous driving, and security systems (Algan & Ulusoy, 2021). The ability to accurately and reliably classify images is crucial because these systems often operate in environments where decisions must be both precise and consistent. For example, in medical imaging, accurate classification can mean the difference between detecting a disease early or missing it entirely (Suganyadevi, Seethalakshmi, & Balasamy, 2022). The reliability of these systems is equally important, as failures can lead to unexpected or harmful outcomes. A notable example occurred in 2016 when an autonomous driving system misclassified a white truck against a brightly lit sky, leading to a fatal collision (Board, 2016). This incident illustrates the need for robust image classification algorithms that perform well across a range of conditions and minimise the risk of such errors.

History and Successes The field of neural-network-based deep learning for image classification has significantly advanced since the introduction of the LeNet architecture in the 1990s (Lecun, Bottou, Bengio, & Haffner, 1998), demonstrating the efficacy of convolutional neural networks (CNNs) for handwritten digit recognition. The introduction of AlexNet in 2012 (Krizhevsky, Sutskever, & Hinton, 2012) marked a transformative moment, significantly boosting accuracy on the ImageNet challenge (J. Deng et al., 2009). Subsequent developments like VGG (Simonyan & Zisserman, 2014), ResNet (He, Zhang, Ren, & Sun, 2016), and Inception (Szegedy et al., 2015) introduced increased depth and residual connections, enhancing learning from large datasets. The advent of Vision Transformers (ViTs) in 2020 (Dosovitskiy et al., 2020) adapted self-attention mechanisms from natural language processing for image classification, treating images as sequences of patches. These advancements have set new performance benchmarks and impacted practical applications profoundly. For instance, VGG and Inception have improved diagnostic accuracy in healthcare (Authors, 2021; Bodapati, Shaik, & Naralasetti, 2021), while ResNet has been crucial in real-time tasks like autonomous vehicle navigation and industrial quality control, requiring rapid and accurate image classification (Huang & Chen, 2020).

Remaining Challenges Despite the successes, image classification still faces several unresolved challenges, particularly concerning data efficiency and performance in the presence of noise (Drenkow, Sani, Shpitser, & Unberath, 2021). Current state-of-the-art models require vast amounts of clean, labelled data, with models often performing

poorly when trained on smaller or noisier datasets (Song, Kim, Park, Shin, & Lee, 2023). This necessity skews the field's benefits towards large organisations that can afford extensive data collection and cleaning efforts. This imbalance limits smaller entities with fewer resources from leveraging the full potential of machine learning. Addressing issues such as algorithmic bias and the environmental impact of training complex models also remains a critical challenge that needs more focused attention in future research.

1.2.4 Big Data and Data Collection Techniques

Deep learning models in image classification reach peak performance when trained with large, diverse datasets. Increasing the depth of neural networks often enhances their effectiveness but also necessitates extensive data volumes (Algan & Ulusoy, 2020). This section explores various data collection methods, highlighting how some techniques generate substantial datasets that may, however, include noisy labels.

1.2.4.1 Data Collection Techniques

Labelled image data can be collected through various methods, each offering unique challenges and benefits. Common techniques include **crowdsourcing**, where platforms like Amazon Mechanical Turk and CrowdFlower (now Figure Eight) are used to gather large volumes of annotations quickly (Whitla, 2009). However, this presupposes that one already has a dataset of images in need of annotation. Crowdsourcing can also be used to gather the images themselves; a prime example is the iNaturalist platform (Aristeidou et al., 2021). This citizen science project enables individuals around the world to contribute by uploading photos of flora and fauna, which are then collaboratively identified and verified by the online community. **Web scraping**, another common technique, automatically extracts images and their associated labels from various websites. For example, a web scraper might target specific online galleries, e-commerce sites, or search engine results, pulling images and any accompanying text that often serves as labels, such as captions, product descriptions, or tags. This process can quickly accumulate large datasets; however, it often requires significant post-processing to verify and clean the data, ensuring its usability for training models. Notable examples of datasets created through web scraping include ImageNet (J. Deng et al., 2009), assembled from millions of web images that were subsequently annotated with the help of hired annotators to ensure diversity and accuracy, and the Open Images dataset (Kuznetsova et al., 2020), which incorporates both automated and manual

annotations to refine the scraped data. Professional data labelling companies such as Scale AI and Samasource provide high-quality annotations for specialised tasks, often used in autonomous vehicle training and medical imaging datasets. A different approach involves **original data collection**, such as the Cohn-Kanade dataset (Lucey et al., 2010), where researchers captured images of facial expressions in controlled environments to study emotions. In medical settings, datasets like the Cancer Imaging Archive **consolidate** CT scans from multiple institutions, offering a rich source for training diagnostic models (Clark et al., 2013). Additionally, researchers sometimes leverage **existing datasets**, expanding them or refining annotations to suit specific tasks. However, this strategy is limited by the scope and scale of the original data (L. Deng, 2012; Krizhevsky, Nair, & Hinton, 2009).

1.2.4.2 Label Noise

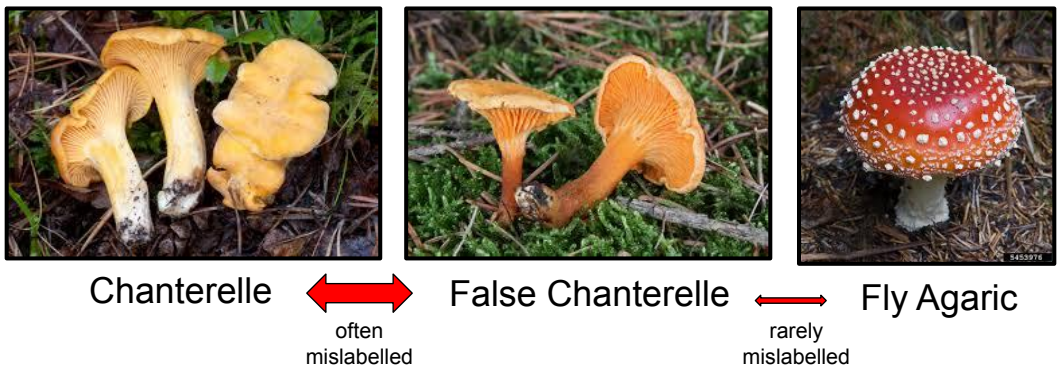


Figure 1.1: This image displays three different types of mushrooms: a Chanterelle, a False Chanterelle, and a Fly Agaric mushroom. Due to their similar appearance, the Chanterelle and False Chanterelle are often more likely to be confused and mislabelled as one another than either is to be confused with the distinctly different Fly Agaric. This example exemplifies asymmetric label noise, where the probability of mislabelling is not uniform across classes. The probability of mislabelling likely also depends on the appearance of the false chanterelle, meaning that label noise would be non-uniform.

Label noise refers to errors in the labels of training datasets. Label noise can arise from inaccuracies during the data collection or annotation process. Such noise can significantly affect the training and performance of machine learning models by misleading the learning algorithm, potentially leading to less accurate models (Z. Zhang & Sabuncu, 2018). For instance, web scraping is prone to 'open-set' noise, where the dataset may include images that do not fit any of the predefined classes,

leading to incorrect or irrelevant labels (C. Feng, Tzimiropoulos, & Patras, 2024). Another common source of noise comes from data collected via online multiple-choice questionnaires, which can introduce 'symmetric' label noise. This occurs when a proportion of respondents select answers uniformly at random to quickly access content, leading to inaccurately labelled data. For example, consumer surveys that list brands in a choice format may suffer from this type of noise if participants select answers without genuine recognition of the brand. Asymmetric and non-uniform label noise often arises when annotations are crowdsourced from non-experts. For instance, when gathering annotations for mushroom species through crowdsourcing, certain pairs of mushrooms are more likely to be confused with each other than others due to their similar appearances. This likelihood of mislabelling is not evenly distributed across all species, leading to asymmetric label noise. An example of this is illustrated in Figure 1.1, where the Chanterelle and False Chanterelle are often misidentified as one another, unlike the distinctly different Fly Agaric.

1.3 Label Noise and Robustness

1.3.1 The Problem of Label Noise

Label noise can significantly **undermine the performance** of overparameterised machine learning models such as deep neural networks (Song et al., 2023). While adept at capturing subtle patterns within large datasets, these models are also susceptible to fitting spurious patterns introduced by noisy labels, compromising their ability to generalise to new data (C. Zhang, Bengio, Hardt, Recht, & Vinyals, 2021). For instance, in medical imaging, if label noise results from misdiagnosed conditions, the model might learn incorrect associations, leading to potentially harmful predictions. An actual case occurred with Google's AI system for diabetic retinopathy, where discrepancies in diagnosis between different sets of doctors led to inconsistencies in training data, affecting the model's performance (Beede et al., 2020). The issue of deep models learning erroneous patterns from noisily labelled data is becoming increasingly significant. This is particularly problematic as models trained on such data are often used to label more online content (Williams, 2023). This newly labelled data, in turn, is used to train further models, potentially reinforcing and **amplifying the initial errors** in an iterative cycle (Alemohammad et al., 2023). This exact phenomenon has been observed by Veselovsky, Ribeiro, and West (2023), who demonstrated that for NLP tasks, 33–46% of crowd workers used LLMs when completing a particular task. A

hypothetical example of this in an image context would be the common mislabelling of Pelargonium species as Geraniums by laypeople contributing to a crowdsourced plant database. If such a dataset were used to train an identification model, it could perpetuate this confusion, leading to further misinformation. This scenario becomes more significant when incorrect information could have serious consequences, such as with the misidentification of edible and poisonous mushrooms as illustrated in our example in Figure 1.1 or in scenarios where label noise embeds racial **biases**. The temptation to gather more data as a common mitigation strategy for label noise Algan and Ulusoy (2020) comes with substantial environmental costs. Training larger models on expanded datasets significantly increases the **carbon footprint**, with Strubell, Ganesh, and McCallum (2019) estimating that training a single AI model can emit as much carbon as two adult Americans do each year.

1.3.2 Robust Loss Functions

This thesis primarily explores how modifying the objective function can enhance robustness in neural network classifiers trained by SGD on noisy data. Historically, cross-entropy has been the predominant choice due to its effectiveness and relationship with maximum likelihood estimation. However, this theoretical underpinning relies on the assumption that labels are reliable, which, as discussed, is often not the case. When data contains erroneous labels, using conventional loss functions can lead the training process astray, causing models to overfit to noise and degrading generalisation performance. This issue prompts a reevaluation of how loss functions are designed to better manage unreliable data and how such modifications affect training dynamics to promote more robust behaviour. The goal is to study, understand and develop loss functions that are inherently less sensitive to label noise.

As outlined in Section 1.2.1, the fundamental components of neural network classification include a labelled dataset, a hypothesis class, a loss function, and an optimisation method. Enhancing any of these components can improve model robustness when dealing with a dataset contaminated with label noise. However, in the context of ERM via SGD, the loss function's role is particularly critical as it serves two purposes: assessing the 'goodness of fit' for each model within the hypothesis class against the given labelled dataset and facilitating the optimisation of this objective. In principle, the optimisation

method and the objective could be independent. However, when using SGD, the loss function uniquely addresses both aspects, highlighting its critical importance in the training process. This dual functionality makes the choice of loss function especially important when addressing learning in the presence of label noise.

A comprehensive overview of the literature on loss-based approaches to handling label noise is given in the literature review in Chapter 3.

1.3.2.1 Robust Loss Function: Problem Statement

The primary objective of robust loss functions is to enable neural network classifiers to effectively train on datasets with noisy labels while still generalising well to the underlying true distribution. Practitioners and researchers in this area focus on two key questions:

1. What characteristics make certain loss functions more robust against label noise?

2. How can loss functions be designed or modified to improve their resilience to label noise?

These inquiries focus on designing loss functions that maintain accuracy despite erroneous labels, thereby enabling models to perform reliably in real-world applications where data noise is common.

1.3.3 Limitations of Current Approaches

Current approaches to handling noisy data in machine learning are fraught with challenges that limit their practicality and effectiveness. As elaborated on in Chapter 3, data-based approaches, such as gathering more data or cleaning existing datasets, are costly and, in the case of the former, also increase environmental costs and training times. Among algorithmic strategies, while there is a plethora of robust machine learning algorithms, many are computationally intensive or overly elaborate, limiting their broader application (Han et al., 2018; Jiang, Zhou, Leung, Li, & Fei-Fei, 2018; J. Li, Socher, & Hoi, 2020; Malach & Shalev-Shwartz, 2017; Sachdeva, Cordeiro, Belagiannis, Reid, & Carneiro, 2021). One of the primary aspirations of label noise robust algorithms is to democratise machine learning by enhancing the utility of noisy datasets. However, these benefits are negated if the proposed approaches are computationally intensive or excessively complex.

Robust loss functions have been widely studied, often avoid the pitfalls of other approaches described and demonstrate promising results on the standard noisy benchmarks (S. Liu, Niles-Weed, Razavian, & Fernandez-Granda, 2020). Nevertheless, significant gaps remain that hinder their application.

1.3.3.1 Theoretical Limitations of Empirically Motivated Robust Loss Functions

Many of the proposed loss functions, such as Taylor Cross-Entropy (L. Feng et al., 2021) (TCE), Symmetric Cross Entropy (Y. Wang et al., 2019) (SCE), and Generalised Cross Entropy (GCE) (Z. Zhang & Sabuncu, 2018) among others, are effective but lack robust theoretical underpinnings. These loss functions are primarily motivated by empirical observations. This is a critical shortcoming because understanding the theoretical basis of these loss functions is essential for predicting and managing their performance in practical settings. Additionally, understanding when and why certain loss functions are effective will allow practitioners to tailor new loss functions for their particular use case.

1.3.3.2 Practical Limitations of Theoretically Motivated Robust Loss Functions

Many loss functions, such as correction-based loss functions (Patrini, Rozza, Krishna Menon, Nock, & Qu, 2017), are theoretically well-founded but come with their own set of limitations. A significant drawback of correction-based losses is that they rely heavily on accurate noise model approximations (Xia et al., 2019). This requires either prior knowledge or an additional inference step, making these approaches less accessible for general use. Correction-based loss functions also perform poorly when dealing with a large number of classes and cannot be utilised when label noise is not class-conditional. Practitioners currently face a choice between empirically motivated loss functions, which are simpler but poorly understood, and theoretically robust loss functions, which, while more sound, are often less practical to implement.

1.3.3.3 Current Theory on Robust Losses Is Underdeveloped

The theory behind robust loss functions, where it does exist, is often underdeveloped. A notable example is 'Noise-Tolerant' loss functions (Ghosh & Kumar, 2017). Noise-Tolerant loss functions satisfy the same robustness criteria as loss correction approaches *without* needing to apply a correction to the loss. In addition to having a strong theoretical basis, these approaches have proven to be effective empirically (Ma, Huang, Wang, Erfani, & Bailey, 2020). Nevertheless, there remains a gap in the existing theory

outlining under which noise models these loss functions can exist and how they may be constructed. Additionally, the theoretical justifications for loss-correction strategies are currently inadequate to fully explain or predict the practical effectiveness of these approaches.

In response to these challenges, our research aims to develop loss functions that are not only easy to implement but also rigorously grounded in theory, ensuring they are both effective and broadly applicable across different noise conditions and dataset complexities.

1.4 Overview of the Study

1.4.1 Research Objectives

The main research aims of this work mirror the limitations listed in Section 1.3.3. Our work is formed of three main objectives: 1) Understand causes of loss function robustness (and lack of robustness), 2) Further develop the theory of loss robustness, and 3) Construct simple but principled noise-robust methodologies.

Improve Understanding of Existing Robust Loss Functions Our first objective is to improve understanding of what makes a loss function robust to label noise. As discussed, many effective loss functions are empirically motivated. We would like to better understand why these loss functions are so effective and under what noise conditions they remain effective. Related to these enquiries, we would like to understand the reasons behind a loss functions robustness (or lack of robustness). For example, is there a single identifiable condition that, when satisfied by a loss function, ensures the robustness of that loss function, or are there multiple such conditions? A related enquiry is to explore the maximum extent of robustness a loss function can achieve. Is it possible for a loss function to be robust against all distributions and noise models? If not, can we meaningfully define a 'maximal' level of feasible robustness?

Further Develop Theory of Robust Losses One second key objective is to make theoretical contributions to the field of label noise robust loss functions. Existing theory about the robustness of loss functions to label noise is often limited in its scope, demanding generalisation. One of our key objectives is to contribute to this process of generalisation, particularly for Noise-Tolerant loss functions. In addition to building upon existing theory, this work aims to leverage observational insights to provide novel

theoretical contributions regarding loss function robustness in deep-learning settings. The existing theory primarily focuses on the infinite data regime and is inadequate to explain the observed differences in the robustness of different loss functions. A core objective is to reduce the gap between the theory and practice of loss robustness.

Provide Theoretically-Motivated but Practical Approaches Utilising insights and theory established in the pursuit of our first two objectives, our third key objective is to develop theoretically grounded but practical and simple approaches for learning in the presence of noisy labels. This is the primary objective of this study. As discussed in Section 1.3.3 users must currently pick between practical loss functions, which are poorly understood, and theoretically-grounded robust loss functions, which are less practical to implement [1]. Our ambition is to bridge this gap by providing approaches with the benefits of both.

1.4.2 Significance of the Research

Enhancing Theoretical Understanding This research contributes meaningfully to the theoretical framework of label noise robustness within deep learning contexts. By enhancing our understanding of how label noise impacts deep neural network training, this work adds to the comprehensive body of knowledge in deep learning. Such theoretical advancements are crucial as they not only foster further research and technological breakthroughs but also streamline the research direction by clarifying productive avenues and identifying less promising ones. This efficiency in research focus potentially saves substantial time and computational resources, accelerating the pace of innovation in machine learning.

Empowering Smaller Organisations The development of label noise-robust methods from this study particularly benefits smaller organisations by making advanced machine learning tools more accessible. By providing effective yet computationally inexpensive solutions, these organisations can leverage smaller and noisier datasets to compete with larger entities. This democratisation of technology is vital, as it allows a broader range of users to implement machine learning solutions without the prerequisite of large-scale data resources, thus levelling the playing field in various industries.

[1] This dichotomy is an oversimplification as e.g. for symmetric label noise, Mean-Absolute Error is a simple, effective and theoretically grounded robust loss function.

Improving Data Efficiency and Environmental Impact Introducing simple and cost-effective approaches to handling label noise also enhances data efficiency. These methods reduce the necessity for vast amounts of data typically required to dilute the effects of noisy labels, thereby lowering the computational load and associated energy consumption. Consequently, this speeds up the model training process and contributes positively to environmental sustainability. The reduced need for excessive data collection and processing ensures a lower carbon footprint of machine learning operations, aligning with global efforts towards greener computing practices.

1.4.3 Scope and Limitations

While the contributions of this research are valuable and advance the field of machine learning in handling label noise, it is essential to acknowledge that they represent **only incremental steps** within a much larger landscape of ongoing research. Significant challenges and unexplored territories remain that continue to limit the full applicability and potential impact of these findings. In the following section, we will discuss the specific restrictions of this project's scope, delineating the boundaries of its applicability and significance.

As we have mentioned, the scope of this work is deliberately narrowed in several key areas to maintain focus and depth, which we reiterate here.

Tasks: We limit our scope to classification. By itself, the scope and applicability of classification is enormous; we apply this restriction to provide clear boundaries for this research while ensuring the depth and manageability of the study.

Models: We consider only deep neural network-based classifier models; the focus of this thesis is *specifically* on robustness for deep, neural network-based models. This focus stems from the current dominance of these models in practical applications and the unique challenges they present in terms of robustness.

Modalities: We restrict our focus to image data, reflecting the widespread significance of image classification tasks. This approach excludes other frequently studied modalities, such as text or video, despite their relevance in neural network applications. This limitation enables us to maintain consistent baselines, benchmarks, and models throughout the chapters.

Approaches: This thesis restricts its focus to simple, primarily loss-based approaches that are theoretically sound and straightforward to implement. This decision is intended to ensure that the proposed solutions are accessible and practical for widespread deployment.

1.4.4 Structure of the Thesis

This thesis comprises nine chapters, including this introduction. Chapter 2 and Chapter 3 set the stage with Background and Literature Review, respectively. The core content spans several topics: exploring Noise-Tolerant loss functions (Chapter 7), strategies for Loss bounding to mitigate overfitting (Chapter 5) and methods of Early stopping with noisy labels (Chapter 6). Additionally, Chapter 4 discusses 'class-preserving label noise,' a concept introduced in the Background chapter. The thesis concludes with Chapter 8, summarising our findings. Detailed overviews of each content chapter are below, summarising the narrative for this thesis.

Class-Preserving Label Noise. *Minor Content:* We discuss the limitations of loss functions in the context of diverse noise models and distributions. We propose that without specific information about the noise model, the optimal strategy is to aim for robustness to 'class-preserving label noise'. This chapter identifies conditions under which label noise is class-preserving, demonstrating that most commonly studied label noise falls into this category.

Risk-Bounding. *Major Content:* Leading on from the discussion in Chapter 2, this chapter introduces a methodology to enhance a loss function's robustness to label noise by mitigating overfitting to empirical risks. We achieve this through techniques that effectively lower bound the training loss.

Early-Stopping. *Major Content:* Chapter 5 proposes an effective approach to mitigate overfitting to label noise. In this chapter, we explore the implementation of early stopping in environments where a cleanly-labelled validation set is not available. We propose monitoring performance on a validation set drawn from the same noisy distribution as the training dataset to guide the stopping decision. We demonstrate that this method is effective across various noisy settings.

Noise Tolerance. *Major Content:* In this chapter, we investigate Noise-Tolerant loss functions. We explore which noise models support the existence of such loss functions and detail methods for their construction.

1.4.5 Summary of Thesis Narrative

Correction-based loss functions hypothesise that poor robustness results from label noise distorting the learning objective. By applying a correction, these approaches ensure consistency of the learning algorithm. However, when the label noise is 'class-preserving', correcting the risk isn't necessary, and any standard loss is sufficient to

ensure consistency. Notably, as demonstrated in Chapter 4, almost all label noise that is commonly studied is class-preserving. This leads us to consider alternative reasons why certain loss functions are non-robust. We argue that poor robustness is caused by certain loss functions' propensity to induce overfitting to finite noisy data. When training highly-expressive neural network models which can fit to a training set, overfitting cannot be resolved by applying a correction. Instead, we propose in Chapter 5 bounding the loss below to prevent overfitting, demonstrating that this strategy has a solid theoretical grounding. We show in Chapter 6 that overfitting can also be mitigated by early-stopping, even when cleanly labelled data is not available. Chapter 7 contributes to the theory of Noise-Tolerant loss functions.

Chapter 2

Robustness of Loss Functions to Label Noise

2.1 Background

Notation In the context of classification, $\mathcal{X} \subset \mathbb{R}^d$ represents the data space, and $\mathcal{Y} \coloneqq \{1, 2, 3, \ldots, c\}$ denotes the label space, where c is the number of classes. The *probability simplex*, Δ, is the set of non-negative c-dimensional vectors whose elements sum to 1, facilitating the modelling of class distribution predictions. For clarity in representation, vector quantities are denoted in **bold**. Table 1 provides a comprehensive reference for the notation used throughout this thesis.

Probability Estimators We define a *probability estimator* as a model $\boldsymbol{q} : \mathcal{X} \to \Delta$ that predicts a class distribution for each data point in $\mathcal{X}$. We consider the case where $\boldsymbol{q}$ is parameterised by a neural network. We use the notation Q to denote the domain of $\boldsymbol{q}$, i.e. the set of probability estimators under consideration.

A *classifier* is a model which assigns a label to each location in dataspace $f : \mathcal{X} \to \mathcal{Y}$. Each probability estimator induces a natural classifier through

$$f(x) = \arg\max_{i \in \mathcal{Y}} \boldsymbol{q}(x)_i$$

this is known as the *plug-in* classifier for $\boldsymbol{q}$. In places where it aids exposition, the distinction may be abused, describing $\boldsymbol{q}$ as a 'classifier'. In this case, we mean the plug-in classifier for $\boldsymbol{q}$.

Standard Classification Setting The standard setting for classification involves a (latent) distribution over the data-label space, denoted as $p(x,y)$. We operate with a dataset of N independent and identically distributed (i.i.d) samples drawn from $p(x,y)$, represented as $\mathcal{D} \coloneqq \{(x_i,y_i)\}_{i=1}^{N}$. The goal is to identify an optimal model $\boldsymbol{q}^*$ within our model family $\mathcal{Q}$, such that the plug-in classifier it induces minimises the expected misclassification rate over the distribution $p(x,y)$.

2.1.1 Loss Functions and Risk

Loss Functions and Expected Loss A loss function $L : \Delta \times \mathcal{Y} \to \mathbb{R}$ evaluates the discrepancy between predicted and actual labels, producing a loss value. For a given prediction $\boldsymbol{q}$, we represent the loss across all classes as

$$\boldsymbol{L}(\boldsymbol{q}) = (L(\boldsymbol{q},1), L(\boldsymbol{q},2), \ldots, L(\boldsymbol{q},c)).$$

Expected Loss The *expected loss* of a forecast $\boldsymbol{q}$ with respect to the true class distribution $\boldsymbol{p}$ is the mean loss for labels sampled from $\boldsymbol{p}$. We denote this by $H_L(\boldsymbol{p},\boldsymbol{q})$. This can be succinctly represented through a vector operation:

$$H_L(\boldsymbol{p},\boldsymbol{q}) = \boldsymbol{p}^T \boldsymbol{L}(\boldsymbol{q}),$$

Pointwise L-Risk Given a distribution $p(x,y)$, a point $x \sim p(x)$, and an estimator $\boldsymbol{q}(x)$, we define the pointwise L-risk of $\boldsymbol{q}$ at x to be the expected loss incurred by the estimator at x. Specifically;

$$R_L(\boldsymbol{q})(x) \coloneqq H_L(\boldsymbol{p}(y \mid x), \boldsymbol{q}(x)).$$

L-Risk We define the (generalised) L-risk as the expected loss of the estimator over the *entire* data distribution $p(x,y)$, capturing the overall performance of $\boldsymbol{q}(x)$:

$$R_L(\boldsymbol{q}) \coloneqq \mathbb{E}_{x\sim p(x)}[H_L(\boldsymbol{p}(y \mid x), \boldsymbol{q}(x))].$$

In the context of classification, an important loss function is the 0-1 loss, which outputs 1 when a model misclassifies and 0 otherwise.

0-1 Loss Defined as

$$L(\boldsymbol{q},k) = \begin{cases} 1 & \text{if } \arg\max_i \boldsymbol{q}_i \neq k, \\ 0 & \text{otherwise.} \end{cases}$$

Formal Objective The learning objective aims to minimise the 0-1 risk, effectively seeking a model $\boldsymbol{q}^*$ that achieves the lowest possible error rate with respect to the true distribution $p(x,y)$. Formally, we seek $\boldsymbol{q}^*$ such that:

$$\boldsymbol{q}^* = \arg\min_{\boldsymbol{q}\in\mathcal{Q}} R_{0\text{-}1}(\boldsymbol{q}),$$

where $R_{0\text{-}1}(\boldsymbol{q})$ denotes the 0-1 risk of $\boldsymbol{q}$ over the distribution $p(x,y)$.

Empirical Risk Minimisation and Stochastic Gradient Descent The standard procedure for learning a neural network classifier given a dataset of samples is known as *Empirical Risk Minimisation* (ERM). The *empirical risk* is an approximation of the generalised risk, computed over the training dataset rather than the entire distribution $p(x,y)$ The empirical risk $\hat{R}$ for a model $\boldsymbol{q}$, given a loss function L and a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$, is defined as:

$$\widehat{R}_L(\boldsymbol{q}) = \frac{1}{N}\sum_{i=1}^{N} L(\boldsymbol{q}(x_i), y_i).$$

When using neural networks, we typically minimise empirical risk using Stochastic Gradient Descent (SGD). SGD updates model parameters by computing gradients of the loss function on small, random batches of the dataset and moving the parameters in the direction which decreases this loss.

As the 0-1 loss has zero gradient almost everywhere, this complicates its direct use in gradient-based optimisation. Instead, so-called *surrogate loss functions* are used, with derivatives properties more suited to SGD optimisation. The ERM objective under a surrogate loss L_{surr} becomes:

$$\hat{\boldsymbol{q}}^* = \arg\min_{\boldsymbol{q}\in\mathcal{Q}} \hat{R}_{L_{\text{surr}}}(\boldsymbol{q}),$$

where $\hat{R}_{L_{\text{surr}}}(\boldsymbol{q})$ denotes the empirical risk calculated with the surrogate loss.

Examples of Loss Functions Below are three commonly used surrogate loss functions. Each loss function $L(\boldsymbol{q},k)$ is described with its respective formula, where k denotes the true class

1. **Cross-Entropy (CE) Loss**: Defined as

$$L(\boldsymbol{q},k) := -\log(q_k),$$

where q_k is the predicted probability of the correct class k.

2. **Mean-Absolute-Error (MAE) Loss**: Expressed as

$$L(\boldsymbol{q},k) \coloneqq \|\boldsymbol{q}-\boldsymbol{e_k}\|_1 = 2(1-q_k),$$

where $\boldsymbol{e}_k$ is the kth coordinate vector, having a 1 at the kth position and 0s elsewhere, indicating the true class k.

3. **Mean-Squared Error (MSE) Loss**: Given by

$$L(\boldsymbol{q},k) \coloneqq \|\boldsymbol{q}-\boldsymbol{e_k}\|_2^2,$$

similar to MAE, $\boldsymbol{e}_k$ represents the true class indicator vector.

Cross-Entropy The cross-entropy loss function stands as the predominant choice for optimising neural network classifiers. This selection stems from its foundation in Maximum Likelihood Estimation (MLE), where the goal is to maximise the probability of observing the given data under the model. Specifically, cross-entropy minimisation is equivalent to maximising the likelihood of the observed data when the model outputs are interpreted as probabilities. This loss function also possesses the characteristic of being a strictly proper loss function (Definition 2.1.2), an attribute that ensures the model's predicted probabilities are aligned with the true underlying probabilities. This is crucial for applications requiring not just high accuracy but also the correct assessment of predictive confidence, known as calibration.

2.1.1.1 Fisher Consistency

For a loss function L to be an effective surrogate, optimising the $L-$risk must guide the model towards decisions that align with those of the Bayes-optimal classifier - the minimiser of the 0-1-risk. This idea is encapsulated by the notion of *Fisher consistency*. A loss L is called Fisher consistent if the minimiser of L-risk achieves optimal classification accuracy according to the true data distribution. Here, one supposes that we are able to take a minima over the space of all possible probability estimators. Most commonly used loss functions satisfy this property, including CE, MSE and MAE losses.

Definition 2.1.1. *(Fisher consistent) A loss function L is called* Fisher consistent *if, for any probability distribution $\boldsymbol{p} \in \Delta$, a minimiser of the expected loss under L also incurs minimal expected misclassification error:*

$$\arg\min_{\boldsymbol{q}\in\Delta} H_L(\boldsymbol{p},\boldsymbol{q}) \subseteq \arg\min_{\boldsymbol{q}\in\Delta} H_{0\text{-}1}(\boldsymbol{p},\boldsymbol{q}).$$

A strict subset of Fisher consistent losses, known as *proper losses*, guarantee the stronger property that the L-risk is minimised by true conditional distribution.

Definition 2.1.2 (Proper Loss)**.** *A loss L is called (strictly) **proper** if, for any $\boldsymbol{p} \in \Delta$, the expected loss is (uniquely) minimised by setting $\boldsymbol{q} = \boldsymbol{p}$, explicitly;*

$$\boldsymbol{p} \in \arg\min_{\boldsymbol{q} \in \Delta} H_L(\boldsymbol{p}, \boldsymbol{q})$$

Proper Loss Examples Mean Squared Error (MSE) Loss: $L_{MSE}(\boldsymbol{q}, k) = ||\boldsymbol{q} - \boldsymbol{e}_k||_2^2$, Cross-Entropy Loss: $L_{CE}(\boldsymbol{q}, k) = -\log(q_k)$, and Spherical Loss: $L_{Spherical}(\boldsymbol{q}, k) = 1 - \frac{q_k}{\sqrt{\sum_{i=1}^{c} q_i^2}}$ where q_k is the predicted probability of the correct class k and c is the number of classes.

Savage's Theorem Given the notion of a proper loss function, it is natural to ask how many proper loss functions exist and whether the set of proper losses admits parameterisation. Leonard Jimmie Savage addressed this question in (Savage, 1971), offering a characterisation through associated entropy functions.

Entropy Function Given a proper loss function L, define its *entropy function* (Ovcharov, 2018) as $\mathcal{H} : \Delta \subset \mathbb{R}^c \to \mathbb{R}$ by:

$$\mathcal{H}(\boldsymbol{p}) \coloneqq H_L(\boldsymbol{p}, \boldsymbol{p}) = \boldsymbol{p}^T \boldsymbol{L}(\boldsymbol{p}),$$

where $\boldsymbol{p}$ is a probability distribution over the classes and H_L is the expected loss defined previously.

Theorem 2.1.3 (Savage's Theorem (Gneiting & Raftery, 2007))**.** *A differentiable loss function L is (strictly) proper if and only if there exists a (strictly) concave function $\mathcal{J} : \mathbb{R}^c \to \mathbb{R}$ such that for each $\boldsymbol{q} \in \Delta$ and $k \in \mathcal{Y}$,*

$$L(\boldsymbol{q}, k) = \nabla \mathcal{J}(\boldsymbol{q})(\boldsymbol{e}_k - \boldsymbol{q}) + \mathcal{J}(\boldsymbol{q}).$$

Moreover, $\mathcal{J}$ is precisely the entropy function for L. Thus, in particular, a loss is (strictly) proper if and only if its associated entropy function is (strictly) concave.

Remark We address differentiable loss functions primarily, but more general forms using subgradients for non-differentiable functions are discussed in (Ovcharov, 2018).

Example: Shannon Entropy The entropy function associated with the cross-entropy loss function is called the Shannon Entropy defined $\mathcal{H}(\boldsymbol{p}) = -\sum_{i=1}^{c} p_i \log(p_i)$.

Example: Quadratic Entropy The entropy function associated with the MSE loss is $\mathcal{H}(\boldsymbol{p}) := 1 - ||\boldsymbol{p}||_2^2$, a special instance of Rao's Quadratic Entropy called the Gini-Simpson Index (Botta-Dukát, 2005; Jost, 2006).

2.2 Label Noise

Label Noise Label noise involves any random alterations of labels from their original distribution. It is categorised into two primary types: closed-set and open-set. We specifically address *closed-set label noise*, wherein the original and noisy label sets are identical. This contrasts open-set noise, where the true label may not be included in the established label set (H. Wei, Tao, Xie, & An, 2021). For instance, in a web-scraped dataset of animal images, a photograph of 'Tiger Woods' might be erroneously labelled as 'Tiger', even though the correct label, 'golfer', is absent from the set of labels.

Transition Matrices We use $p(\widetilde{y} \mid y, x)$ to denote the noise model that generates noisy labels $\widetilde{y}$ conditioned on the true label y and the position in dataspace $x \in \mathcal{X}$. For closed-set label noise, $p(\widetilde{y} \mid y, x)$ may be modelled by a square transition matrix $T(x)$ at each point $x \in \mathcal{X}$. This matrix is column-stochastic, meaning that the sum of each column in T equals 1. The matrix converts the true label distribution at x; $\boldsymbol{p}(y \mid x) \in \Delta$ into a noisy distribution $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x) = T(x)\boldsymbol{p}(y \mid x)$, with the tilde signifying noise-affected quantities.

2.2.1 Label Noise Taxonomy

When proving results about label-noise robust algorithms, one typically needs to make some assumptions about the properties of the label noise. This necessitates a label noise taxonomy. Below we provide a summary of some common ways in which this is done.

Uniform Label Noise Label noise is classified as *uniform* or *class-conditional* when $T(x)$ is constant across $\mathcal{X}$, denoted as T. Equivalently; $p(\widetilde{y} \mid y, x) = p(\widetilde{y} \mid y)$.

Symmetric/Asymmetric Label Noise Label noise is called *symmetric* if all off-diagonal elements of T are equal, indicating uniform mislabelling across classes. The transition matrix for symmetric label noise in the case of three classes ($c = 3$) is shown in Equation 2.1. Conversely, *asymmetric* noise occurs when mislabelling probabilities vary among classes.

$$T = \underbrace{\begin{pmatrix} 1-\eta & \frac{\eta}{2} & \frac{\eta}{2} \\ \frac{\eta}{2} & 1-\eta & \frac{\eta}{2} \\ \frac{\eta}{2} & \frac{\eta}{2} & 1-\eta \end{pmatrix}}_{\text{Symmetric Label Noise}} \tag{2.1}$$

Pairwise Label Noise An important type of asymmetric label noise is pairwise label noise. Given a classification task with c classes and a transition matrix T, if for any class i, there exists at most one class $j \neq i$ with $T_{ij} = \eta$ and at most one class $k \neq i$ with $T_{ki} = \eta$, the noise is *pairwise*. This mislabelling occurs between specific class pairs.

Circular Label Noise A special case of pairwise noise occurs when, with probability $1-\eta$, labels remain uncorrupted and with probability η a label transitions to the next class $j \mapsto j+1$, with the final class wrapping back around $c \mapsto 1$. This leads to transition matrix T with a structure so that $T_{ii} = 1-\eta$ and $T_{i,i-1} = \eta$ and $T_{1,c} = \eta$ (See Equation 2.2). We say that label noise of this type is *circular*.

$$T = \underbrace{\begin{pmatrix} 1-\eta & 0 & \cdots & 0 & \eta \\ \eta & 1-\eta & \cdots & 0 & 0 \\ 0 & \eta & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & 1-\eta & 0 \\ 0 & 0 & \cdots & \eta & 1-\eta \end{pmatrix}}_{\text{Circular Label Noise}} \tag{2.2}$$

Diagonally Dominant We say that T is *diagonally dominant* (DD) if for each i, the diagonal entry is greater than any other entry in its column $T_{ii} > \max_{j \neq i} T_{ji}$[1]. Diagonally dominant label noise is depicted in Figure 2.1 in the case of $c = 3$ classes.

Remark Our definition of diagonal dominance (DD) may initially seem to deviate from those presented in X. Li et al. (2021) and Xu, Cao, Kong, and Wang (2019). However, these references employ the convention where T_{ij} represents the probability of transitioning from $y = i$ to $\widetilde{y} = j$, which contrasts with our usage where T_{ij} indicates the probability of transitioning from $y = j$ to $\widetilde{y} = i$. This alternate convention simplifies the linear algebra involved. When this convention difference is considered, our definition aligns with those found in the literature.

[1] As mentioned in (X. Li, Liu, Han, Niu, & Sugiyama, 2021) this definition, commonly used in the context of noisy labels (Nguyen et al., 2019), differs from the definition of DD from linear algebra.

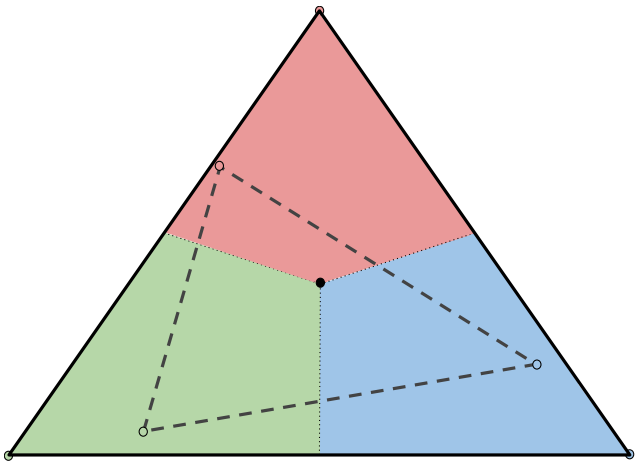


Figure 2.1: The image of the probability simplex under diagonally dominant label noise for three classes. The decision boundaries are shown for each class so that, for example, the red region consists of all vectors $\boldsymbol{p} \in \Delta$ where $p_1 \geq p_2, p_3$. The image of the simplex under the transition matrix T is overlain (given by a dotted line). The image of each corner of the simplex under T lies within its own coloured region. This illustrates that T is diagonally dominant.

2.2.2 Class-Preserving Label Noise

This thesis's primary focus is identifying and developing loss functions that enable training in the presence of label noise whilst maintaining generalisation in the resulting classifier. However, unless the noise model is explicitly known and integrated into the loss function (as seen with correction-based loss functions, for example), it is infeasible to create a loss function that is robust against all types of label noise (This issue is elaborated upon in Chapter 4). Consequently, the most realistic goal is to develop a loss function that is robust to a broad family of label noise, encompassing many common types. To this end, we define such a family of label noise in this section, which we term 'dominant-class-preserving' (or 'class-preserving' for short). This noise type consists of all label noise models that maintain which class is the most probable at every x, generalising the concept of Diagonal Dominance. This represents a significant subset of label noise; Chapter 4 demonstrates that almost all frequently studied noise types adhere to this definition. Crucially, however, this noise type is sufficiently restrictive to enable the construction of robust loss functions.

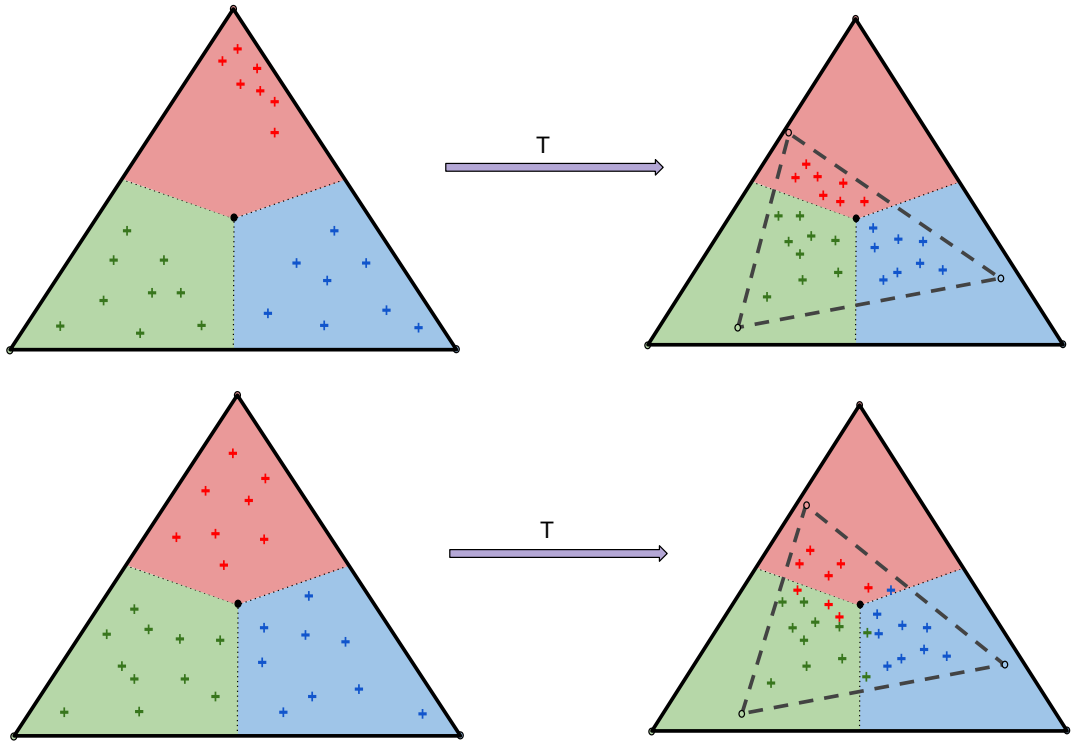


Figure 2.2: Class-preserving (top) and non-class-preserving (bottom) label noise in the case of three classes. Some $x_i \sim p(x)$ are sampled and their respective class distributions $\boldsymbol{p}(y \mid x) \in \Delta$ are plotted on the left simplex and coloured according to which region they lie in - i.e. the value of $\arg\max_i p(y = i \mid x)$. On the right-hand simplex we plot the noisy class distributions $\widetilde{\boldsymbol{p}}(y \mid x) \in \Delta$ at these same locations. In the top figure all of the $\widetilde{\boldsymbol{p}}$ lie in the same region as their respective (clean) $\boldsymbol{p}$ indicating that the label noise is class-preserving for the given distribution. Conversely, the bottom figure shows non-class-preserving noise in that some of the $\boldsymbol{p}$ are mapped into different regions.

Definition 2.2.1 (Dominant Class)**.** *For a given data-label distribution $p(x,y)$ over a set $\mathcal{X}$, the dominant class at a data point $x \in \mathcal{X}$ is defined as the class with the highest conditional probability at x. This is denoted by $k_{max}(x)$ and is mathematically represented as:*

$$k_{max}(x) := \arg\max_{i \in \mathcal{Y}} p(y = i \mid x).$$

The dominant class represents the most probable class label given the data point x. We call $p_{max}(x) := \max_{i \in \mathcal{Y}} p(y = i \mid x)$ the dominant class probability.

Definition 2.2.2 (Dominant-Class-Preserving Noise)**.** *Given a data-label distribution $p(x,y)$ and its noisy version $\widetilde{p}(x,\widetilde{y})$, resulting from label noise, the noise is considered* ***dominant-class-preserving*** *if, for every $x \in \mathcal{X}$, the dominant class remains unchanged after noise application. Formally, this is expressed as:*

$$\underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \widetilde{\boldsymbol{p}}(\widetilde{y}=i \mid x) = \underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \boldsymbol{p}(y=i \mid x),$$

where c is the number of classes. We will often abbreviate to calling this noise type ***class-preserving*** *for purposes of brevity.*

As the name suggests, class-preserving noise *preserves* whichever class has the highest probability. For example, let $c = 3$ and suppose the clean class distribution is $\boldsymbol{p} = (0.1, 0.6, 0.3)$. If a transition matrix T is class-preserving, this ensures that then the $(T\boldsymbol{p})_2 \geq (T\boldsymbol{p})_1, (T\boldsymbol{p})_3$. Figure 2.2 illustrates class-preserving and non-class-preserving noise for the case of three classes.

Discussion This concept of class-preserving noise deviates from traditional taxonomic terms such as symmetric, uniform, or diagonally dominant label noise because it inherently relies on the underlying data distribution. This dependency means a label noise model might be class-preserving in one context but not in another. We make extensive use of this definition in subsequent chapters when looking at methods which do not directly utilise the noise model.

When is Noise Class-Preserving? Chapter 4 is dedicated exclusively to looking at class-preserving label noise. The chapter provides a more detailed discussion about the class-preserving assumption and conditions under which noise is class-preserving, demonstrating that *the vast majority of label noise investigated in the relevant literature is class-preserving*.

2.2.3 Ergodicity

In certain sections it proves convenient to assume that the (class-conditional) label noise has a unique stationary distribution. By this, we mean that there exists some $\boldsymbol{\pi} \in \Delta$ for which $T\boldsymbol{\pi} = \boldsymbol{\pi}$ and that no other $\boldsymbol{p} \in \Delta$ satisfies this condition. A sufficient condition for a unique stationary distribution is ergodicity which we define below.

Definition 2.2.3. *Let T be the transition matrix representing class-conditional label noise. The matrix T is said to be* ergodic *if it satisfies the following two conditions:*

1. Irreducibility*: For every pair of class labels i and j, there exists some integer* $n > 0$ *such that* $(T^n)_{ij} > 0$*, indicating that it is possible to transition from any true class label to any other observed class label through some sequence of noisy observations.*

2. Aperiodicity*: For every class label i, the greatest common divisor of all integers* $n > 0$ *for which* $(T^n)_{ii} > 0$ *is 1, meaning there are no fixed cycles that constrain how often a true class label can be observed as itself through noise.*

The following standard result, which can be referenced in Norris (1998), confirms that ergodicity is sufficient for the existence of a unique stationary distribution:

Theorem 2.2.4. *Let* T *be an ergodic transition matrix. Then,* T *has a unique stationary distribution* $\boldsymbol{\pi} \in \Delta$*, satisfying* $T\boldsymbol{\pi} = \boldsymbol{\pi}$*.*

Ergodicity of Label Noise Ergodicity comprises two conditions: aperiodicity and irreducibility. Aperiodicity will hold for any label noise transition matrix which isn't completely degenerate. Specifically, for any reasonable label noise, we should expect there to be some probability that the noising process does not alter the label, $T_{ii} > 0$. This condition by itself is sufficient to give aperiodicity. The validity of assuming irreducibility holds for label noise is much less clear. In many natural, non-synthetic settings, there will be some probability (albeit sometimes very small) of any class being mislabelled as any other class T_{ij}. For example, while it is more likely that a cat may mislabelled as a dog it could also be mislabelled as a toaster with some small probability. In these settings the label noise is irreducible and therefore ergodic. Similarly, symmetric noise is irreducible, as is circular noise (Defined Equation 2.2). However, one can construct other synthetic noise which is non-irreducible. For example, pairwise noise, where the permutation can be expressed as a product of two or more disjoint permutations.

2.2.3.1 Noise Rate

We define the *noise rate* at x, denoted by $\eta(x)$, of a noise model as being the probability a label is altered, $p(\widetilde{y} \neq y)$. When the noise model is class-conditional, the noise rate is constant as a function of x; $\eta(x) = \eta$. For class-conditional label noise with balanced classes the noise rate is given by $1 - \frac{1}{c}Tr(T)$, where Tr denotes the trace of the transition matrix. For example, the matrix shown in Equation 2.1 has a noise rate η.

2.2.4 Noisy Risk

The efficacy of Empirical Risk Minimisation (ERM) hinges on the assumption that minimising empirical risk over a large dataset approximates minimising the generalised risk. This approach presupposes that our dataset comprises independent and identically distributed (i.i.d) samples from $p(x,y)$. However, the presence of label noise disrupts this assumption. Instead of drawing i.i.d samples from the clean distribution $p(x,y)$, our dataset effectively contains i.i.d samples from a noisy distribution $\widetilde{p}(x,\widetilde{y})$. This discrepancy introduces the concept of *noisy risk*, denoted as $R_L^{\eta}(\boldsymbol{q})$, which is the risk of an estimator $\boldsymbol{q}$ evaluated against the noisy distribution rather than the clean one.

Noisy *L*-Risk We define the *noisy L-risk*, denoted $R_L^{\eta}(\boldsymbol{q})$, as the expected loss when our model confronts noisy labels, formalised as:

$$R_L^{\eta}(\boldsymbol{q}) = \mathbb{E}_{(x,\widetilde{y})\sim\widetilde{p}(x,\widetilde{y})}[L(\boldsymbol{q}(x),\widetilde{y})].$$

Similarly, we define the *noisy empirical L-risk*, $\widehat{R}_L^{\eta}(\boldsymbol{q})$, as the risk calculated over a dataset corrupted by label noise.

$$\widehat{R}_L^{\eta}(\boldsymbol{q}) \coloneqq \sum_{n=1}^{N} L(\boldsymbol{q}(x^{(n)}),y^{(n)}).$$

2.3 Robust Loss Functions

In this section, we introduce the notion of a robust loss function. A loss function is deemed 'robust' if training a classifier using the loss on noisy data results in a model that generalises effectively to the clean data distribution. First, we must briefly establish the foundational fact that the choice of loss function significantly influences generalisation to clean data; in other words, some loss functions are inherently more robust than others. Subsequently, we will discuss various types of robust loss functions, starting with correction-based loss functions and moving on to more heuristic approaches.

2.3.1 Loss Function Impacts Robustness - Experiments

Cross-Entropy is not Robust The cross-entropy loss function is the preferred method for optimising neural network classifiers in environments without label noise. Yet, this loss function is notably susceptible to label noise. Even modest levels of noise can significantly impair a model's ability to generalise.

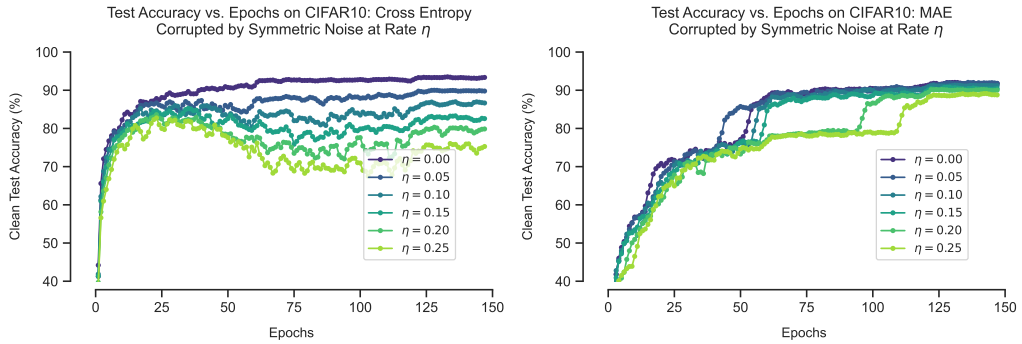


Figure 2.3: Comparison of Clean Test Accuracy over 150 Epochs with Different Loss Functions on CIFAR10: The figure presents results from training with Cross-Entropy (left) and MAE (right) loss functions under varying rates of symmetric label noise $(0.0, 0.05, ..., 0.25)$. As noise levels increase, performance degradation is observed; however, the impact is more pronounced with Cross-Entropy, where the decline in clean test accuracy reaches 20%, compared to a modest 3% decrease with MAE, indicating a relative robustness of MAE to label noise.

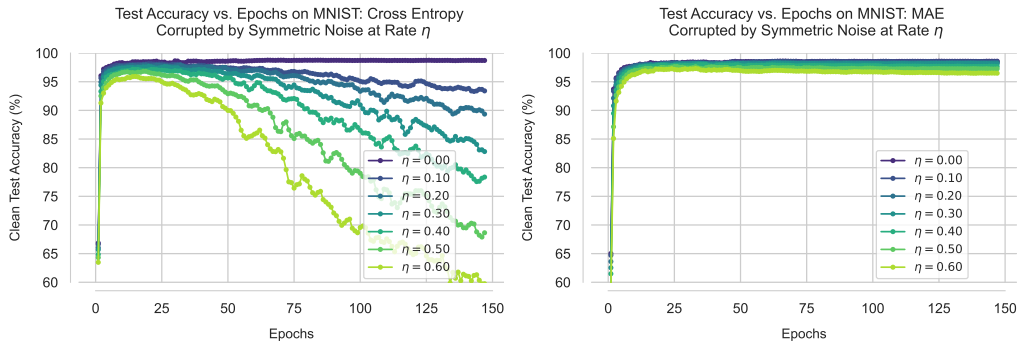


Figure 2.4: Comparison of Clean Test Accuracy over 150 Epochs with Different Loss Functions on MNIST: The figure presents results from training with Cross-Entropy (left) and MAE (right) loss functions under varying rates of symmetric label noise $(0.0, 0.1, ..., 0.6)$. As noise levels increase, huge performance degradation is observed for CE. In contrast, the decline in clean test accuracy reaches with MAE is minute, indicating a much greater robustness of MAE to label noise in this setting.

Cross-Entropy Experiment Figure 2.3 (left) shows the true (clean) test accuracy of a neural network at each epoch when trained on the CIFAR10 dataset using a cross-entropy loss function at different noise rates: ranging from no noise ($\eta = 0$) to 25% symmetric label noise ($\eta = 0.25$). This figure demonstrates how label noise undermines generalisation when using a cross-entropy (CE) loss function. While initial stages of training on a noisy dataset might show a temporary improvement in test accuracy, the model eventually overfits to the noise, evidenced by a downturn in generalisation performance. It is crucial to note that the test set remains clean, ensuring that the observed test accuracy reflects the model's performance on *uncorrupted* data. Figure 2.4 (left) repeats this for the MNIST dataset showing similar behaviour.

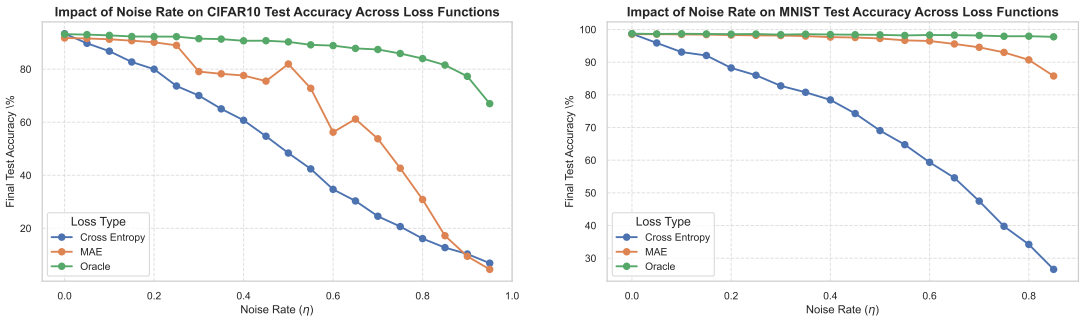


Figure 2.5: Comparative Analysis of Final Test Accuracy Under Symmetric Label Noise Across CIFAR10 and MNIST Datasets: The plots illustrate the impact of different noise rates on the clean test accuracy for cross-entropy (CE), MAE, and Oracle loss functions. For CIFAR10, CE loss shows a significant degradation with increasing noise rates, while MAE demonstrates greater resilience. On MNIST, the difference is more pronounced, with CE loss deteriorating rapidly, whereas MAE maintains higher accuracy levels. The Oracle loss function, representing an ideal scenario in which cross-entropy updates only on un-noised samples, is included as an idealised benchmark to contextualise the other losses.

Comparison With MAE Conversely, when employing a Mean Absolute Error (MAE) loss function as shown in Figure 2.3 (right) and Figure 2.4 (right), the model displays remarkable resistance to label noise. Across settings, the final test accuracy remains approximately constant, demonstrating the superior robustness of MAE under these settings and underlining the importance of loss function selection on model robustness against label noise.

Figure 2.5 shows the final clean test accuracy of a classifier trained on noisy CIFAR10 (left) and MNIST (right) as a function of noise rate for the CE (blue) and MAE (orange) loss functions (label noise is symmetric). For both datasets and all noise rates, MAE outperforms CE, reaffirming that the MAE loss function is more robust under these settings. The graph also shows the performance of an 'oracle' (green), which can identify and ignore all the corrupted labels, taking CE gradient updates only on the clean data. This is included to contextualise the other loss functions by providing an idealised baseline.

2.3.1.1 Choice Of Loss Function Matters

This distinct contrast in performance between Cross-Entropy (CE) and MAE loss functions underlines a critical insight: the choice of loss function significantly influences a model's robustness to label noise. This observation prompts several pertinent questions:

1. What characteristics define a robust loss function?
2. Why do some loss functions offer greater robustness than others?

Robust Loss Functions In the context of neural network classification, a *Robust Loss Function* refers to a loss function that allows a classifier to maintain effective generalisation to the clean underlying data distribution, even when trained on data corrupted by label noise.

Problem Formulation Uncover the characteristics of a robust loss function. Specifically, identify which properties of L enable neural network classifiers, trained via gradient descent on noisy datasets $\widetilde{\mathcal{D}}$, to generalise successfully to the clean data distribution $p(x,y)$. Essentially, we're asking:

> Which features of L help a classifier $\boldsymbol{q}$ maintain high accuracy on clean data, despite being trained with the noise in $\widetilde{\mathcal{D}}$?

2.3.2 Robust Loss Functions: Loss Corrections

In this section we outline a variety of robust loss functions which we collectively call 'loss correction approaches'. This includes within it the 'forward' and 'backward' correction (Patrini et al., 2017), Noise-Tolerant loss functions (Manwani & Sastry, 2013) and loss-reweighting approaches (T. Liu & Tao, 2015). These loss functions are unified by their aim to correct the risk to account for the distortion caused by the presence of label noise.

2.3.2.1 Risk Distortion

As outlined in Section 2.1.1, when training a classifier in the noise-free setting, our goal is to select loss function L so that by optimising the empirical surrogate L-risk within $\mathcal{Q}$, we approximate minimising the true 0-1 risk;

$$\arg\min_{\boldsymbol{q}\in\mathcal{Q}} \widehat{R}_L(\boldsymbol{q}) \approx \arg\min_{q} R_{0\text{-}1}(\boldsymbol{q}). \tag{2.3}$$

Conventionally, this involves selecting a Fisher consistent (Definition 2.1.1) surrogate loss L and optimising the L-risk on a large dataset of i.i.d samples via SGD. This process hinges on two pivotal assumptions leading to our main objective:

1) Fisher Consistency This principle asserts that minimising the surrogate L-risk is equivalent to minimising the 0-1 risk, formulated as:

$$\arg\min_{\boldsymbol{q}} R_L(\boldsymbol{q}) = \arg\min_{\boldsymbol{q}} R_{0\text{-}1}(\boldsymbol{q}).$$

2) Generalisation: This assumption posits that optimising the empirical L-risk over the model space $\mathcal{Q}$ closely approximates the minimisation of the L-risk:

$$\arg\min_{\boldsymbol{q}\in\mathcal{Q}} \widehat{R}_L(\boldsymbol{q}) \approx \arg\min_{\boldsymbol{q}} R_L(\boldsymbol{q}).$$

These assumptions together imply our goal (Equation 2.3):

$$\textbf{1, 2} \implies \arg\min_{\boldsymbol{q}\in\mathcal{Q}} \widehat{R}_L(\boldsymbol{q}) \approx \arg\min_{\boldsymbol{q}} R_{0\text{-}1}(\boldsymbol{q}). \tag{2.4}$$

Label Noise and Risk Distortion The introduction of label noise disrupts the approximation between the (noisy) empirical L-risk and the generalised clean L-risk. Label noise distorts the risk landscape, meaning:

$$\arg\min_{\boldsymbol{q}\in\mathcal{Q}} \widehat{R}^{\eta}_L(\boldsymbol{q}) \not\approx \arg\min_{\boldsymbol{q}} R_L(\boldsymbol{q}),$$

This distortion breaks the validity of the Generalisation assumption and, by extension, interrupts the logical flow (Equation 2.4) that guarantees an approximately Bayes-optimal classifier under a Fisher consistent L and a large dataset. Consequently, we cannot rely on Fisher consistency and Generalisation alone to ensure our objective:

$$\arg\min_{q\in\mathcal{Q}} \widehat{R}^{\eta}_L(q) \approx \arg\min_{q} R_{0\text{-}1}(q).$$

Risk Correction To counter label noise's distorting effects, strategies often modify the loss function to 'correct' the risk. The aim is to adjust L to a new loss $L^{'}$, so minimising the noisy $L^{'}$-risk aligns with minimising the clean L-risk:

$$\arg\min_{\boldsymbol{q}} R^{\eta}_{L'}(\boldsymbol{q}) = \arg\min_{\boldsymbol{q}} R_L(\boldsymbol{q}). \tag{2.5}$$

Assuming L is chosen to be Fisher consistent, this correction allows us to make a similar argument to the noise-free case, this time consisting of *three* assumptions:

1) Fisher: $\arg\min_{\boldsymbol{q}} R_L(\boldsymbol{q}) = \arg\min_{\boldsymbol{q}} R_{0\text{-}1}(\boldsymbol{q})$.

2) Correction: $\arg\min_{\boldsymbol{q}} R^{\eta}_{L'}(\boldsymbol{q}) = \arg\min_{\boldsymbol{q}} R_L(\boldsymbol{q})$.

3) Generalisation: $\arg\min_{\boldsymbol{q}} \widehat{R}^{\eta}_{L'}(\boldsymbol{q}) \approx \arg\min_{\boldsymbol{q}} R^{\eta}_{L'}(\boldsymbol{q})$.

Together, these assumptions imply our goal:

$$\textbf{1, 2, 3} \implies \arg\min_{\boldsymbol{q}\in Q} \widehat{R}^{\eta}_{L}(\boldsymbol{q}) \approx \arg\min_{\boldsymbol{q}} R_{0\text{-}1}(\boldsymbol{q}). \tag{2.6}$$

2.3.2.2 Approaches

Forward Correction The most popular example of the scheme described above involves a loss alteration referred to as the *forward correction* (Patrini et al., 2017). Given a loss function L and class-conditional label noise with (estimated) noise transition matrix $\hat{T}$, one defines the forward-correction via $L_F(\boldsymbol{q},k) \coloneqq L(\hat{T}\boldsymbol{q},k)$. When $\hat{T} = T$, this correction ensures that risk relation in Equation 2.5 holds between a loss and its forward correction. The forward-correction is typically applied to a Fisher consistent loss such as cross-entropy, thus ensuring that the entailment summarised in Equation 2.6 applies. Since the matrix T is typically unknown, it is usually estimated from the data. In our subsequent analysis, however, we will typically assume the idealised scenario in which T is known $\hat{T} = T$ unless explicitly stated otherwise.

Backward Correction The *backward correction* is an alternative approach where, instead of noising the predictions of the model using the (estimated) noise transition matrix $\hat{T}$ (as in the forward correction), we denoise the loss: $\boldsymbol{L}_B(\boldsymbol{q}) \coloneqq \hat{T}^{-T}\boldsymbol{L}(\boldsymbol{q})$[2] (Natarajan, Dhillon, Ravikumar, & Tewari, 2013; Patrini et al., 2017). When $\hat{T} = T$, this correction ensures that risk relation in property in Equation 2.5 holds. As with the forward-correction we will assume in our analysis that T is known unless explicitly stated otherwise.

[2]Where A^{-T} denotes the transpose of the inverse of A.

Noise Tolerance An alternative approach involves finding conditions under which Equation 2.5 holds without needing to explicitly apply a correction. i.e $L = L'$. Ghosh and Kumar (2017) give conditions which a loss must satisfy in order to be intrinsically Noise Tolerant for symmetric label noise. They showed that MAE, a loss already known empirically to have good robustness properties, was Noise Tolerant in this sense.[3].

Importance Reweighting Importance Reweighting (T. Liu & Tao, 2015) is similar to the backward correction. Here one chooses weights $w(x,y)$ to that

$$\mathbb{E}_{x,y\sim p(x,y)}[L(\boldsymbol{q},y)] = \mathbb{E}_{x,\widetilde{y}\sim\widetilde{p}(x,\widetilde{y})}[w(x,\widetilde{y})L(\boldsymbol{q},\widetilde{y})]$$

The weights work out to be the ratio of the noisy and clean class probabilities at x. One may envision $w(x,y)L(\boldsymbol{q},y)$ as an augmented/corrected loss function with a x-dependence satisfying Equation 2.5.

Loss Name	**Definition** $L(\boldsymbol{q},k)$
Cross-Entropy	$-\log(q_k)$
MSE	$\|\boldsymbol{q}-\boldsymbol{e}_k\|_2^2 := \left((q_k-1)^2+\sum_{i\neq k} q_i^2\right)$
SCE	$A(1-q_k+\sum_{i\neq k} q_i) - \log(q_k)$
MAE	$\|\boldsymbol{q}-\boldsymbol{e}_k\|_1 := 1-q_k+\sum_{i\neq k} q_i$
GCE	$(1-q_k^a)/a$
NCE-MAE	$\frac{-\log(q_k)}{\sum_i -\log(q_i)} + 1 - q_k + \sum_{i\neq k} q_i$
Mix-Up	$-\alpha\log(q_{k_1}) - (1-\alpha)\log(q_{k_2})$
Spherical	$-q_k\|\boldsymbol{q}\|^{-2}$
Bootstrap Loss	$-(1-a)\log(q_k) - a\log(q_{\text{pred.}})$
Label Smoothing	$-(1-\varepsilon)\log(q_k) - \frac{\varepsilon}{c-1}\sum_{i\neq k}\log(q_i)$
ELR	$-\log(q_k) + \lambda\log(1-\boldsymbol{t}\cdot\boldsymbol{q})$
TCE	$(1-q_k) + \frac{(1-q_k)^2}{2}$

Table 2.1: Definitions for each loss. We assume the given label is k, and the vector q denotes the prediction probabilities. q_{pred} denotes the model's current prediction. For Mix-Up, k_1, k_2 are the targets for the two samples used in constructing the mixed sample. A denotes the coefficient for the RCE loss in the SCE loss (e.g. $A = 4$). ε in the label smoothing loss and a in Bootstrap denote some small numbers ≈ 0.1. The $\boldsymbol{t}$ in the ELR loss function denotes a moving average of its predictions during training.

[3]MAE satisfies a condition stronger than Equation 2.5 which is discussed in more detail in Section 7.1

2.3.3 Robust Loss Functions: Heuristic Approaches

In contrast to correction-based methods are heuristic robust loss functions. These loss functions are typically empirically motivated and do not directly handle the problem of risk distortion described in Section 2.3.2. In contrast to forward/backward and reweighting approaches, these loss functions do not rely on knowledge of the underlying noise model. Instead, this family of loss functions offer a straightforward and adaptable solution for managing label noise in classification tasks.

Examples Many popular loss functions are based upon the observation that MAE is robust but tends to underfit. Taylor-CE (L. Feng et al., 2021), Symmetric-CE (Y. Wang et al., 2019) and Generalised-CE (Z. Zhang & Sabuncu, 2018) attempt to resolve this by altering the MAE loss by adding on a CE term or by interpolating between CE and MAE to obtain the best of both loss functions. Other loss functions such as label-smoothing (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016), bootstrap losses Reed et al. (2014) and mix-up H. Zhang, Cisse, Dauphin, and Lopez-Paz (2017) regularise by smoothing the predictions and/or the noisy labels. ELR S. Liu et al. (2020) regularises by encouraging the model predictions to remain somewhat constant during training. A non-exhaustive summary of the formulae of some heuristic robust loss functions can be found in Table 2.1. In each case $\boldsymbol{q}_k$ denotes the probability the forecast $\boldsymbol{q}$ assigns to the revealed label. Other notation is explained in the table caption.

2.4 Reasons for Non-Robustness of Loss Functions

In order to develop robust loss functions it is crucial to understand what makes a loss function robust/non-robust to label noise. Roughly speaking, two competing hypotheses aim to answer this question. These two hypotheses broadly correspond to the two families of robust loss functions described in Section 2.3.2 and Section 2.3.3. Correction-based loss functions contend that a loss function (such as cross-entropy) is non-robust because of the distorting impact that label noise has on the risk. Consequently, these methods aim to repair the risk by correcting the loss using the noise model. Conversely, the more heuristic approaches (e.g. GCE) contend that a robustness failure is caused by the propensity of certain loss functions to induce overfitting when handling noisy data (S. Liu et al., 2020). Consequently, these methods are generally designed to mitigate overfitting.

There are undoubtedly regimes in which each of these explanations is superior. Nevertheless, we argue the latter is a better explanation in the typical settings we see studied in the literature. That is to say, poor robustness is caused by overfitting to noisy data - not due to risk distortion. This argument has two parts: 1) The backward correction has better theoretical guarantees but performs worse than the forward correction - which is inconsistent with the risk hypothesis. 2) Most label noise studied in the literature is class-preserving; in such settings, loss-corrections are not necessary given sufficient data.

2.4.1 The Risk Hypothesis

Correction methods are built on the implicit assumption that the distortion of the risk caused by label noise means that

$$\arg\min_{\boldsymbol{q}} R_L^{\eta}(\boldsymbol{q}) \neq \arg\min_{\boldsymbol{q}} R_{0\text{-}1}(\boldsymbol{q}). \tag{2.7}$$

By correcting the loss $L \mapsto L'$ these methods repair this equality

$$\arg\min_{\boldsymbol{q}} R_{L'}^{\eta}(\boldsymbol{q}) = \arg\min_{\boldsymbol{q}} R_{0\text{-}1}(\boldsymbol{q}), \tag{2.8}$$

ensuring that, when the dataset is large, we may expect generalisation by optimising the empirical noisy risk.

There are three hidden assumptions here. Firstly, that the presence of label noise results in the equality violation in Eqn. 2.7. Secondly, that this violation explains the observed lack of robustness exhibited by certain loss functions e.g. cross-entropy. Thirdly, that by repairing this equality (Eqn. 2.8) by applying a correction one may improve robustness. We call this collection of assumptions the *Risk Hypothesis*.

The Risk Hypothesis suggests that a major factor in explaining why loss corrections improve robustness to label noise is that they guarantee Equation 2.8 is satisfied.

2.4.1.1 Robustness of Fisher Consistent Loss Functions

A key motivation for loss correction is that by correcting the loss, we guarantee that a minimiser of the noisy risk is Bayes-optimal (Equation 2.8). However, in many cases, this condition already holds *before* the correction applied. I.e. without doing any correction at all, it is already often the case that

$$\arg\min_{\boldsymbol{q}} R_L^{\eta}(\boldsymbol{q}) = \arg\min_{\boldsymbol{q}} R_{0\text{-}1}(\boldsymbol{q}), \tag{2.9}$$

a condition that we were supposedly correcting the risk to obtain.

Indeed, if our loss function L is Fisher consistent and the label noise is class-preserving then this condition is satisfied without applying a loss correction. This is stated formally in the following Lemma.

Lemma 2.4.1. *Let $\widetilde{p}(x,\widetilde{y})$ be a (noised) data-label distribution obtained by applying class-preserving label noise to $p(x,y)$. Let L be a loss function. If L is Fisher consistent, then any noisy risk minimiser $q^\star := \arg\min_q R_L^\eta(q)$ is Bayes-optimal for the clean distribution.*

Proof. The Fisher consistency of the loss implies that the noisy risk $R_L^\eta(q)$ is minimised by $\boldsymbol{q}$ where for every $x \in \mathcal{X}$, $\arg\max_i \boldsymbol{q}(x)_i = \arg\max_i \widetilde{\boldsymbol{p}}(y = i \mid x)$. Since the label noise is class-preserving then $\forall x \in \mathcal{X}$ it holds that $\arg\max_{i\in\{1,2,\ldots,c\}} \boldsymbol{p}(y = i \mid x) = \arg\max_{i\in\{1,2,\ldots,c\}} \widetilde{\boldsymbol{p}}(\widetilde{y} = i \mid x)$. Thus $\arg\max_i \boldsymbol{q}(x)_i = \arg\max_{i\in\{1,2,\ldots,c\}} \boldsymbol{p}(y = i \mid x)$ meaning that $\boldsymbol{q}$ is Bayes-optimal for the clean distribution - it has minimal misclassifcation rate. □

Corollary 2.4.2. *Let L denote the cross-entropy loss, then when the label noise is class-preserving*

$$\arg\min_{\boldsymbol{q}} R_L^\eta(\boldsymbol{q}) \subseteq \arg\min_{\boldsymbol{q}} R_{0\text{-}1}(\boldsymbol{q})$$

Implications Corollary 2.4.2 states, when label-noise is class-preserving, that *given a large enough noisy dataset*, a suitably expressible neural network and an effective optimiser, optimising a cross-entropy loss will yield a Bayes-optimal classifier for the *clean* data distribution. (Likewise for any other standard, Fisher consistent loss function.) This is to say there is nothing intrinsically problematic about a cross-entropy objective which should prevent us from generalising well when label noise is class-preserving and data is plentiful.

Most Noise is Class-Preserving The significance of this observation derives from the fact that most label noise studied in the relevant literature *is* class-preserving. (Chapter 4 is dedicated exclusively to demonstrating this fact.) This suggests that the '*risk hypothesis*' is inadequate to explain the poor robustness observed in certain loss functions.

Forward vs Backward The risk hypothesis is further undermined by the fact that the backward correction satisfies much stronger theoretical properties than the forward correction but performs worse. While the backward correction satisfies powerful theoretical properties, the theoretical properties satisfied by the forward correction are pretty weak. If the risk hypothesis was true, we might, therefore, expect the backward correction to outperform the forward. In fact, the opposite is generally true (Ma et al., 2018; Patrini et al., 2017). Instead, the backward correction typically results in larger gradients causing rapid overfitting to the noisy data, damaging generalisation. (See Appendix A.3 for experimental comparison).

2.4.1.2 Summary

Our analysis suggests that the risk hypothesis is inadequate for explaining why some loss functions are robust while others are not. While applying a loss correction guarantees the consistency of the learning algorithm, these guarantees are generally already satisfied before applying a correction.

When training with highly expressible, overparameterised neural network models, there is a significant probability of overfitting to noise when data is insufficient. In contrast to the risk hypothesis, we propose that robustness might be improved simply by designing losses to mitigate this overfitting rather than devising mechanisms which ensure consistency. In subsequent chapters, we investigate approaches to mitigating overfitting.

2.5 Conclusions

2.5.1 Chapter Overview

In Section 2.2, we introduced the concept of label noise and provided a taxonomy. In Section 2.3, we discussed robust loss functions, highlighting how some loss functions are more vulnerable to label noise than others. In Sections 2.3.2 and 2.3.3, we clearly distinguished between two sets of robust loss functions: correction-based methods, which aim to rectify the risk distortion caused by label noise, and regularisation-based methods, which seek to mitigate overfitting. We explored in Section 2.4 why some loss functions are more robust than others, explaining how the two loss families are founded on competing hypotheses. Our comparison of backward and forward losses

and our observation that most studied label noise is class-preserving (expanded on in Chapter 4) leads us to conclude that overfitting to noisy datasets is the primary cause of poor robustness. This discussion is crucial as it sets the stage for Chapter 6 and Chapter 5, where we explore principled ways to mitigate overfitting to noisy datasets.

2.5.2 Content Chapters

Our discussions in Section 2.4 suggest that correction-based loss functions, although theoretically elegant, do not necessarily guarantee robustness in practice. We propose that robustness may be imbued by providing theoretically-motivated mechanisms which prevent loss functions overfitting to finite noisy datasets. In Chapters 6 and 5, we suggest some approaches. Specifically, Chapter 5 describes how to use an estimate of the noise rate to approximate the entropy, which then serves as a loss lower-bound to prevent overfitting. Conversely, Chapter 6 employs a simpler strategy of monitoring *noisy* validation accuracy during training to halt training before overfitting begins. Chapter 7 looks at Noise-Tolerant loss functions initially introduced in Section 2.3.2.2. We demonstrate that such loss functions are rare and attempt to categorise all possible Noise-Tolerant loss functions.

Chapter 3

Literature Review

3.1 Label Noise

Supervised learning relies on accurately labelled datasets, as algorithms use these to infer the labelling function. Unfortunately, many datasets suffer from incorrect labels due to annotation errors and various data collection issues. The study of label noise in supervised learning, a field that has been active for several decades (Angluin & Laird, 1988), continues to be highly relevant due to its significant impact across multiple fields. These fields include the medical domain, financial analytics, and autonomous driving, where the accuracy of labelled data is crucial for effective decision-making (Frenay & Verleysen, 2014). The advent of big data and advancements in deep learning, particularly for computer vision tasks, have underscored the necessity for algorithms that can robustly handle noisy labels (Song et al., 2023). This section provides an overview of the pertinent literature on label noise, offers a taxonomy of the types of label noise, and discusses both classical and modern deep-learning approaches for learning in the presence of label noise.

Adversarial and Feature Noise This thesis primarily explores label noise, in contrast to feature noise. Label noise entails corruption within the label space, transforming $p(y \mid x)$ to a corrupted version $\widetilde{p}(\widetilde{y} \mid x)$, whereas feature noise corrupts the data space, altering $p(x)$ to $\widetilde{p}(x)$ (Shanthini, Vinodhini, Chandrasekaran, & Supraja, 2019). Feature noise may include missing components, Gaussian or other forms of additive noise, and various distortions that affect the feature space.

Under the Mutually-Contaminated label noise framework (A. Menon, Rooyen, Ong, & Williamson, 2015), there is recognised overlap between feature and label noise models, since label noise is assumed to be generated by contamination between class distributions. However, label noise and feature noise are typically distinct; it is possible to encounter one without the other.

A sub-category of both label and feature noises are adversarial noises. In cases of adversarial noise, one assumes that, rather than having been generated by some agnostic random process, the noise is produced by an antagonistic agent, typically for the purpose of maximally disrupting learning (Frenay & Verleysen, 2014). The problems of feature noise and adversarial label noise are rich and widely studied. Nevertheless, the emphasis of this work remains strictly on the varieties and implications of non-adversarial label noise, excluding feature noise and adversarial scenarios from the scope of investigation. This focus allows for a more in-depth exploration of label noise, its sources, its impact on learning models, and the development of strategies to mitigate its effects without the additional complexity introduced by other noise types.

3.1.1 Taxonomy

During the development of the label noise robust learning algorithms various taxonomies have emerged for categorising different varieties of label noise. Establishing a taxonomy is convenient as it can expedite theoretical and experimental analyses. This section will review some of these taxonomies.

Scott, Blanchard, and Handy (2013) and A. Menon et al. (2015) introduce the 'Mutually Contaminated' (MC) framework for studying label noise wherein corrupted distributions are modelled as mixtures of the clean distribution. For example, letting P, Q denote the densities of two classes,

$$P_{corr} := (1-\alpha)P + \alpha Q,$$
$$Q_{corr} := \beta P + (1-\beta)Q,$$

are the noised variants. An alternative framework is the 'Positive and Unlabelled' (PU) framework for label noise for binary labels (A. Menon et al., 2015). In this framework one has a dataset of positive samples and a dataset of unlabelled samples (including unlabelled positive instances). Each of the samples in unlabelled dataset is labelled as

negative meaning that some of the positive samples are mislabelled. This framework sis alternatively referred to as 'asymmetric semi-supervised learning' Stempfel and Ralaivola (2009).

Frenay and Verleysen (2014) provides an alternate label noise taxonomy, dividing label noise into three main types - 'noised completely at random' (NCAR), 'noised at random' (NAR) and 'noisy not at random' (NNAR). NCAR occurs when the label in the dataset is flipped to another label in a manner completely independent of both the original label and the location in dataspace. Whereas NAR allows conditioning on the label. Using the taxonomy we adopt (Section 2.2.1), the terms NCAR and NAR correspond approximately to symmetric and asymmetric label noise (in the binary case this is an exact correspondence). The final term, NNAR, allows dependence on location and would be referred to as non-uniform asymmetric label noise using the nomenclature we adopt in this work.

Most modern deep-learning literature studying noisy labels adopts and builds upon the class-conditional noise (CCN) framework described in Angluin and Laird (1988). The class-conditional noise framework *'[assumes that the sampling oracle is able to draw elements from the relevant distribution D without error, but that the process of determining and reporting whether the example is positive or negative is subject to independent random mistakes with some unknown probability* $\eta < 1/2$*]'* (Angluin & Laird, 1988). A. Menon et al. (2015) gives a detailed breakdown showing how the CCN framework relates to the PU and MC frameworks described above. Interestingly, unlike the CCN framework, the MC framework described in A. Menon et al. (2015) permits non-linear models - i.e., the noisy and clean conditional class distributions are related via a non-linear rather than linear transformation.

Building upon the CCN framework, a standard modern taxonomy has developed which categorises the different sub-varieties of CCN. Class-conditional label noise where the transition probabilities between all distinct classes are identical is called 'symmetric' noise (Van Rooyen, Menon, & Williamson, 2015; Z. Zhang & Sabuncu, 2018), e.g.,

$$(1,0,0) \mapsto \left(1-\eta, \frac{\eta}{2}, \frac{\eta}{2}\right).$$

However, some work (e.g., (Ghosh & Kumar, 2017)) refers to this as 'simple' label noise. Conversely, 'asymmetric' corresponds to any non-symmetric class-conditional noise model (Scott et al., 2013). Expanding upon class-conditional noise, non-uniform' or 'feature-dependent' label noise extends to the case where the transition probabilities depend on the datapoint (Patrini et al., 2017; Y. Zhang, Zheng, Wu, Goswami, & Chen, 2021). Each noise type may be further subdivided, for example, pairwise and cyclical label noises are subsets of asymmetric noise (Song et al., 2023). An overview of this taxonomy, which we adopt throughout this study, is detailed in Section 2.2.1.

Open and Closed-Set Label Noise Label noise may be divided into open and closed set noise (C. Feng et al., 2024). Open set noise is where the true labels of dataset instances do not belong to classes present during training (Lee et al., 2019). C. Feng et al. (2024) provide an illustrative example of this noise type, imagining a web-scraped dataset of tigers containing a picture of the golfer Tiger Woods. Conversely, closed-set label noise involves random mislabelling *within* the existing label categories (Xia et al., 2023). Standard image datasets are known to contain both noise types (Sachdeva et al., 2021) at differing rates; Song et al. (2023) states that noise rates range from 8% to 38.5%. For example, the Clothing1M dataset is believed to contain closed-set label noise at a rate of around 38% (Y. Wang et al., 2019). ImageNet is also believed to contain substantial label noise - mostly open-set (Yun et al., 2021). Typically, the evaluation of noise-robust algorithms includes tests with synthetic label noise, where the labels in the training set are deliberately altered according to a specified noise model, for example, (Englesson & Azizpour, 2021b; L. Feng et al., 2021; Z. Zhang & Sabuncu, 2018) and others. The use of synthetic noise ensures the availability of the clean ground-truth labels, allowing for easier evaluation of noise-robust algorithms.

Among studies proposing noise-robust learning algorithms, some target closed-set label noise (Hendrycks, Mazeika, Wilson, & Gimpel, 2018; Patrini et al., 2017; Stempfel & Ralaivola, 2009), while others focus more on open-set noise (Y. Wang et al., 2018; Yu & Aizawa, 2020). However, a number of studies explore methods applicable to both noise types (Lee et al., 2019; H. Wei et al., 2021; Xia et al., 2023; Z. Zhang & Sabuncu, 2018). This work will examine both types, with a closer focus on closed-set noise.

Notably, Xia et al. (2023); Yu and Aizawa (2020) demonstrate that by introducing a 'meta' class, which includes all classes not in the label set, open-set label noise can be managed as a form of unbalanced closed-set classification. This indicates that the distinction between these noise types may not be entirely clear-cut.

3.1.1.1 Restrictions and Identifiability

Work studying label noise robust learning often involves imposing some restrictions on the label noise model to allow learning (Cannings, Fan, & Samworth, 2020). Consequently, these restrictions form an important aspect of label noise taxonomy in this setting. The core issue stems from the fact that, from mathematical perspective, *noise* in the label distribution is indistinguishable from *intrinsic uncertainty* without additional assumptions. By this, we mean that, given a noisy conditional distribution over labels, this distribution cannot be uniquely decomposed into a clean conditional class distribution and a noise model—the noise model is a priori '*non-identifiable*' (Fu, Huang, & Sidiropoulos, 2018; Y. Liu, Cheng, & Zhang, 2023)—. This section briefly reviews some assumptions which are made to resolve this identifiability issue.

In their seminal work Angluin and Laird (1988) limit the noise rate to $\eta < 1/2$ stating *'Why do we restrict* η *to be less than* $1/2$*? Clearly, when* $\eta = 1/2$*, the errors in the reporting process destroy all possible information about membership in the unknown set and no identification procedure could be expected to work'*. A similar assumption is made in works such as Cohen (1997); Ghosh and Kumar (2017); Stempfel and Ralaivola (2009). Ghosh and Kumar (2017); Manwani and Sastry (2013) assume data distribution separability to ensure identifiability and facilitate learning despite noisy labels. Alternatively, X. Li et al. (2021); Patrini et al. (2017); Xia et al. (2019); Xu et al. (2019) rely on assuming diagonal dominance of the transition matrix in the class-conditional label noise setting. X. Zhou, Liu, Jiang, Gao, and Ji (2021) define a condition they call 'clean-labels-domination' in which it is assumed that the most likely label of each element of the dataset is the true class. This maps closely onto the ideas we introduce in Chapter 4. The importance of making assumptions on the noise model to ensure identifiability is expanded upon within Section 4.2.1.

3.2 Label Noise Robust Classification Methods

3.2.1 Classic Approaches

Dating back to at least the work of Lachenbruch (1966), practitioners have been studying the impact of label noise on classification algorithms. Numerous studies have investigated, both theoretically and experimentally, the robustness of various learning algorithms to label noise and developed approaches for improving noise-tolerance. Before the prevalence of deep-learning for classification, research initially focused on classic algorithms including Support Vector Machines (SVMs) (Cortes & Vapnik, 1995), k-Nearest Neighbour (kNN) approaches, tree-based methods, Naive-Bayes classifiers, Linear Discriminant Analysis (LDA) and logistic regression among others (Angluin & Laird, 1988; Lachenbruch, 1966). This section reviews some of the literature on label noise robust classification, for non-deep-learning algorithms.

Empirical Studies In their 2010 study, Nettleton, Orriols-Puig, and Fornells (2010) provided an empirical comparison of SVM, Naive Bayes, kNN and decision trees to feature and label noise. This study demonstrated a superior robustness of Naive Bayes, in particular relative to SVM classifiers, at high label noise levels. At low noise levels performance was similar across methods. Pechenizkiy, Tsymbal, Puuronen, and Pechenizkiy (2006) studied how feature extraction affects label noise impact on kNN, Naive-Bayes, and SVM classifiers in the medical domain. Their analysis indicated that the robustness of each method has a high dependence on the dataset, with each method being superior to the others in certain settings. Cannings et al. (2020) explored the impact of noisy training labels on classification methods, finding that kNN and SVM classifiers can achieve statistical consistency under specific conditions of label noise, whereas LDA typically fails to maintain consistency unless class prior probabilities are equal. H. Zhang, Cheng, Zhang, and Li (2021) investigated the robustness of Naive-Bayes, SVM, logistic regression and decision trees (and other approaches) to adversarial label flipping. Their study reiterated the findings of Nettleton et al. (2010), demonstrating the superior robustness of Naive Bayes over the other approaches, albeit in the context of *adversarial* label noise.

Theoretical Studies Beigman and Klebanov (2009) and Cohen (1997) investigated the robustness of the Voted Perceptron and Perceptron algorithms (Freund & Schapire, 1999) to label noise. Beigman and Klebanov (2009) concluded that the Voted Perceptron algorithm is vulnerable to 'annotation noise', meaning random label noise that affects certain samples in the dataset (non-uniform label noise). Building on earlier work by Bylander (1994), Cohen (1997) examines the learning of a linear classifier (Perceptron) in polynomial time on linearly separable binary-labelled distributions corrupted by symmetric label noise. Lawrence and Schölkopf (2001) looked at extending kernel Fisher discriminant analysis (KDA) to class-conditional label noise using an Expectation-Maximisation (EM) optimisation strategy, demonstrating good performance on some simple datasets.

Other work takes a loss function oriented perspective. For example, Stempfel and Ralaivola (2009) studied label noise in the context of learning an SVM classifiers as an approach to handling 'asymmetric semi-supervised learning'. This work introduces 'SloppySVM,' which adjusts the hinge loss so that the expected noisy empirical risk with the adjusted loss is an unbiased estimator of the clean empirical risk. They provide an analysis quantifying the deviation between these quantities in terms of dataset size, data support. Manwani and Sastry (2013) also take a loss function oriented perspective, demonstrating that linear classifiers for binary labelled data are tolerant to uniform symmetric label noise when using a mean-squared error loss function. This work also demonstrates that Fisher Linear discriminant is robust to symmetric label noise.

3.2.1.1 Regression

A widely-studied problem is that of regression in the presence of noisy labels (Adomaityte, Defilippis, Loureiro, & Sicuro, 2024; Y. Guo, Wang, & Wang, 2023; Huber, 1973). While the scope of this thesis is *classification* in the presence of noisy labels, insights from the field of noisy regression may be leveraged for classification. For example, Huber (1973) famously introduced the Huber loss function, which handles outliers by modifying the MSE loss to become linear beyond a certain threshold— a method now referred to as 'Huberising.'

$$L_{\text{Huber}}(f(x),y) = \begin{cases} \frac{1}{2}(y-f(x))^2 & \text{if } |y-f(x)| \leq \delta, \\ \delta\left(|y-f(x)| - \frac{1}{2}\delta\right) & \text{otherwise.} \end{cases}$$

This concept parallels ‘gradient clipping’ in classification, as used by A. K. Menon, Rawat, Reddi, and Kumar (2020), to prevent overfitting due to label noise. Similarly, L. Feng et al. (2021) apply a Huberisation-style approach through truncating the Taylor expansion of the log loss to improve robustness. For linear regression, the MSE loss is robust if zero-mean noise is added to each of the regression targets. In Englesson and Azizpour (2024), this principle is adapted for classification by mapping noisy labels to regression targets and using an MSE loss, effectively treating the classification problem as a regression to enhance robustness against label noise.

3.2.2 Deep Learning Approaches

The focus of this thesis is deep learning. Specifically, we are interested in studying and improving the robustness of neural network classifiers to noisy labels. This is a large area of research. In this section we give an overview of some of the methods which have been developed to improve the robustness of deep learning to noisy labels.

Unlike some simpler pre-deep-learning classifiers, neural networks are highly vulnerable to noisy labels. H. Zhang et al. (2021) compared the robustness of classic and deep methods (AlexNet and LeNet) under adversarial label noise showing that the deep methods were the most vulnerable to label noise of all methods they analysed. Even when it is non-adversarial neural networks incur a large drop in generalisation when label noise is present in the training set (Khanal & Kanan, 2021). This is caused by the ability of neural networks to fit to completely random labels, as shown by Arpit et al. (2017). This high expressivity means that, when trained on noisy data for sufficient epochs, the network will fit to the corrupted labels degrading generalisation (Cheng et al., 2024).

In contrast to these aforementioned studies, Rolnick, Veit, Belongie, and Shavit (2017) have argued that deep learning is fairly robust to label noise even at high levels. They demonstrate that neural network classifiers achieve satisfactory generalisation performance even when noisy samples outnumber clean samples. Nevertheless, the experiments in Rolnick et al. (2017) reaffirm that label noise is detrimental, in particular

for smaller datasets where the decline in generalisation can be substantial. A key contribution of this study is showing that the detrimental impact of label noise may be offset by substantially increasing dataset size. However, for small and medium-sized datasets, the problem of label noise remains an important problem.

3.2.2.1 Regularisation

The tendency of neural network classifiers to overfit to noisy data has prompted numerous regularisation strategies aimed at mitigating this issue. Arpit et al. (2017) explores the effects of dropout (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014), noting that it mitigates but does not entirely prevent overfitting. Other approaches involve augmenting existing training samples; it has been observed that introducing feature noise during training can enhance generalisation in the presence of noisy labels. For instance, H. Zhang et al. (2017) describes 'MixUp', which linearly combines targets and data points; $x_1, x_2 \mapsto \alpha x_1 + (1-\alpha)x_2$ and $\boldsymbol{y}_1, \boldsymbol{y}_2 \mapsto \alpha \boldsymbol{y}_1 + (1-\alpha)\boldsymbol{y}_2$. Strategies such as 'AutoAugment' (Cubuk, Zoph, Mane, Vasudevan, & Le, 2019) and 'RandAugment' (Cubuk, Zoph, Shlens, & Le, 2020) construct image augmentations by randomly selecting from a variety of transformations. Nishi, Ding, Rich, and Hollerer (2021) introduces 'Augmented Descent' (AUGDESC), a method that integrates data augmentation into dual-network architecture approaches to handling label noise (J. Li et al., 2020). Despite many augmentation methods not being initially developed with label noise in mind, Pereira, Carneiro, and Cordeiro (2022) notes their effectiveness in improving generalisation in such contexts. H. Wei et al. (2021) takes a targeted approach by intentionally incorporating open-set noisy examples into the training dataset, which further enhances robustness against existing intrinsic label noise. Alternative approaches focus on regularising model weights, such as those described by Harutyunyan, Reing, Steeg, and Galstyan (2020), including techniques like weight decay (Song et al., 2023) and gradient clipping (A. K. Menon et al., 2020). These methods leverage theoretical insights, such as those discussed in Gouk, Frank, Pfahringer, and Cree (2021), which demonstrate that imposing Lipschitz constraints can limit model complexity and potentially improve generalisation.

3.2.2.2 Architecture

Noise Adaption Layers A group of methods handles label noise by making architectural alterations to improve robustness. Many of these approaches consist of appending a 'noise adaption' layer to the end of the architecture. Sukhbaatar and Fergus (2014) introduces two approaches; a 'bottom-up' approach which changes model outputs in accordance with the (estimated) noise model and 'top-down' approaches which adjust the labels. The bottom-up method requires the estimation of the noise transition matrix, which is then attached as a final layer following the softmax computation,

$$\boldsymbol{q} \mapsto \hat{T}\boldsymbol{q}.$$

Goldberger and Ben-Reuven (2017) applies a similar bottom-up approach attaching a noise-adaption layer after the softmax. The network is initially trained without the noise adaption layer; thereafter, both the network and the noise adaption layer are trained concurrently, with the latter initialised to the confusion matrix. This work applies this approach to both class-conditional label noise and instance-dependent (non-uniform) label noise. Sukhbaatar, Bruna, Paluri, Bourdev, and Fergus (2015) also utilise a noise adaption layer at the end of the network, this is subtly different in that they directly parameterise a transition matrix instead of utilising a second softmax layer as in Goldberger and Ben-Reuven (2017) to map back to probability space. Sukhbaatar et al. (2015) utilises trace regularisation to prevent the learned matrix from becoming too close to the identity. In an earlier work Mnih and Hinton (2012) take a similar approach albeit in the binary label setting. In this case their equivalent of the noise adaption layer is learned by a variant of the EM-algorithm (Dempster, Laird, & Rubin, 1977). Many of these noise-adaption layer methods are closely related to forward corrections, which we discuss in Section 3.2.2.4.

Chen and Gupta (2015) introduces 'Webly' learning, a bottom-up approach that starts by training a neural network classifier on 'easy' images sourced from search engines. A confusion matrix derived from this training step is then sparsified to form a graph that reflects similarities among categories. This graph is incorporated into the loss function to guide the fine-tuning process when the network is subsequently exposed to more complex and noisier images. During fine-tuning, the graph ensures that the network incurs a reduced penalty for mis-classifications among related categories, aiding in robust learning from noisy data.

3.2.2.3 Early Stopping

Early stopping serves as a straightforward regularisation method that modulates model complexity by curtailing the number of training epochs, with performance assessed against a separate validation set (Prechelt, 2002). Applying early stopping in noisy label settings can be problematic due to the unavailability of a cleanly-labelled validation set to gauge generalisation. Nevertheless a number of approaches devise methods for applying this approach to prevent overfitting to label noise. The concept of 'Pre-stopping,' introduced by Song, Kim, Park, and Lee (2019), incorporates early stopping into a robust pipeline against label noise. This technique halts training when signs of deteriorating generalisation appear, identified using a small, accurately labelled dataset. M. Li, Soltanolkotabi, and Oymak (2020) underlines the utility of early stopping in contexts of symmetric label noise and offers theoretical insights showing its effectiveness in networks with a single hidden layer, given specific data distribution conditions. Unlike typical approaches that use a validation dataset to gauge generalisation, their method determines the early-stopping point through the clustering characteristics of the data distribution. In the domain of adversarial robustness, Rice, Wong, and Kolter (2020) demonstrates that early stopping can perform on par with other advanced robust learning strategies when facing adversarial conditions, assuming access to a cleanly labelled dataset for evaluating generalisation. Xia et al. (2021) proposes 'robust-early-learning,' a strategy that manages label noise by dividing network parameters into critical and non-critical groups, each subject to different updating protocols, and then implementing early stopping based on the misclassification rates on a noisy validation set. Furthermore, Bai et al. (2021) develops Progressive Early Stopping (PES), a method that initially trains the deeper layers of a network—those responsible for capturing general features—before applying early stopping to the more superficial layers that are more susceptible to overfitting. Notably, PES eschews the use of a validation set and instead ceases training after a predetermined number of epochs.

Multi-Network Approaches Many deep-learning methods for handling label noise are elaborate and require pipelines involving multiple networks and multiple stages (Han et al., 2018; Jiang et al., 2018; J. Li et al., 2020; Malach & Shalev-Shwartz, 2017; Sachdeva et al., 2021). Sachdeva et al. (2021) introduces 'EvidenceMix'. This approach trains two networks for a few epochs on a dataset corrupted by both closed and open set label noise. A GMM on the loss values is used to identify and remove open-set samples. The networks are then trained using techniques from semi-supervised learning (SSL)

on the remaining data. Other two-network models work on similar lines. 'Co-teaching' (Han et al., 2018) works by training one network on the outputs of the other with lowest loss values. 'Decoupling' (Malach & Shalev-Shwartz, 2017) works by having the two networks update on the basis of disagreement with each other. 'Mentor-Net' (Jiang et al., 2018) harnesses a teacher network for training a student network by re-weighting probably correct samples. J. Li et al. (2020) introduce 'DivideMix' in which a Gaussian mixture model selects clean samples based on their loss values. These samples are then taken and used to train the other network.

3.2.2.4 Label Correction

A major category of methods consists of identifying and correcting corrupted labels

$$\widetilde{y} \mapsto y_{\text{corrected}}.$$

There is substantial overlap with these approaches and the reweighting and sample selection approaches which we outline in Section 3.2.2.5 and Section 3.2.2.6. Many methods in these three categories rely on the heuristic that noisy samples incur higher losses, especially earlier in training. This is based upon the well-known observation that complex models generally learn to classify easier data points before over-fitting on noise (Arpit et al., 2017). Song, Kim, and Lee (2019) use the entropy of the historical prediction distribution to identify refurbishable samples. Arazo, Ortego, Albert, O'Connor, and McGuinness (2019) deploy a beta mixture model in the loss space and use the posterior probabilities that a sample is corrupted in the parameters of a bootstrapping loss. T. Zhou, Wang, and Bilmes (2020) define a loss which ignores samples that incur a higher loss value. Other approaches include the already mentioned two-network model (J. Li et al., 2020) which implements a Gaussian mixture model as part of its two-network process of identifying clean samples based on their loss values. Yu and Aizawa (2020) provides a label cleaning approach for open-set label noise by adding a c+1$^{\text{th}}$ 'unknown' label representing that a sample's true label is outside the label set. Pseudo-labels are initialised for each sample and iteratively updated through joint optimisation alongside the network parameters.

Many label correction approaches use ideas from semi-supervised learning. Early in training a classifier model often predicts the correct, un-noised label for many of the noisy samples (H. Kim, Chang, Cho, Lee, & Han, 2024). Leveraging this property, methods may correct the labels so that they are a mixture of the noisy labels and the model's predictions early in training. Mixing the label in this way can help prevent overfitting to noise. For example, Cheng et al. (2024) employs a dynamic weighting strategy where the influence of model predictions on noisy labels decreases throughout training. Similarly, Englesson and Azizpour (2024) use a moving average of old predictions as part of their noise robust algorithm. Yun et al. (2021) presents a method 'ReLabel' for handling noisy labels by using a label correction technique that relies on an auxiliary model. This model is trained on a portion of the data believed to have correct labels, which is used to adjust the labels in the noisy training set. G. Zheng, Awadallah, and Dumais (2019) is a meta-learning label correction approach in which a label correction network corrects labels and feeds them to a classifier to train. The label correction network is meta-learned to optimise the performance of the classifier on a held-out validation set. Similarly, in Vyas, Saxena, and Voice (2020) soft labels are treated as learnable parameters and learned to maximise the performance on the meta-set. Szegedy et al. (2016) takes a simpler approach which proves to be effective; smoothing the noisy labels by mixing with the uniform distributions of classes.

3.2.2.5 Sample Re-weighting

Sample reweighting approaches associate weights to the training samples in the dataset. The goal is to attach higher weights to samples which have a higher probability of being correctly labelled and lower weights to corrupted samples, minimising their corrosive impact on training. Specifically, we associate a weight w_i to each data-label pair (x_i, y_i) in the training set and alter the loss objective for the pair via

$$L(f(x_i), y_i) \mapsto w_i L(f(x_i), y_i).$$

The weight w_i may be determined by numerous different strategies. Chang, Learned-Miller, and McCallum (2017) introduces 'Active-Bias' which reweights samples according to the variance of the model predictions during training showing that in noisy settings an improvement in generalisation can be obtained by emphasising 'easy' (more certain) samples. Other reweighting approaches adopt ideas from 'self-paced learning'. 'Self-paced learning' in machine learning is a training strategy where models initially

focus on simpler examples to improve robustness and efficiency, examples include Pi et al. (2016) and Meng, Zhao, and Jiang (2015). Bar, Koren, and Giryes (2021) typically assigns lower weights to samples that incur higher loss values during training, using a multiplicative update mechanism. The weights are bounded to prevent them from decreasing below a certain threshold. Majidi, Amid, Talebi, and Warmuth (2021) also assigns lower weights to samples that incur higher loss values. They present an approach where the training weights are treated as learnable parameters and updated multiplicatively using the exponentiated gradient update. Sukhbaatar and Fergus (2014) trains using a mixture of clean and noisy data; the data is mixed together at training time, with a lower weighting given to the noisy versus the clean data. The weight is learned by cross-validation. Ren, Zeng, Yang, and Urtasun (2018) learns sample weights using a meta-learning approach. A learnable weight is initialised for each sample in the training-dataset; these weights are meta-learned to maximise generalisation of the classifier being learned to a held-out cleanly-labelled meta-set. 'Meta-Weight-Net' introduced by Shu et al. (2019) adopts a similar strategy.

3.2.2.6 Sample Selection

Instead of reweighting noisy samples or attempting to refurbish corrupted labels, some methods target either the complete removal of noisy samples or decreasing the probability with which they are sampled during training. Multiple sample selection methods expect that noisily labelled data lie heavily out of class distribution under an appropriate metric. In FINE (T. Kim, Ko, Choi, & Yun, 2021), noisy samples are detected and removed using an eigendecomposition in the latent space. Alternatively, applying kNN C. Feng, Tzimiropoulos, and Patras (2021) in the latent space can identify and select samples based on their coherence to their neighbours' classes. The 'NoiseBox' algorithm (C. Feng et al., 2024) is a clean sample selection method where label consistency among nearest neighbours in the feature space is used to remove samples with lower consistency. Active-Bias (Chang et al., 2017) described in Section 3.2.2.5 experiments with a 'sample by variance' strategy. This method increases the sampling frequency of data samples with lower model prediction variance, showing an improvement in generalisation by emphasising easier (lower variance) samples. T. Zhou et al. (2020) and Shen and Sanghavi (2019) adopt a simpler approach, ignoring samples that incur a higher loss value.

3.2.2.7 Combination Approaches to Label Noise

Numerous methodologies exist for addressing label noise in machine learning, extending beyond those detailed in this document. Many of these approaches utilise pipelines that combine multiple previously mentioned methods. For instance, the 'NoiseBox' algorithm (C. Feng et al., 2024), discussed earlier, augments its clean sample selection process with additional steps including dataset expansion, balancing, and a training phase that integrates consistency regularisation. Similarly, H. Kim et al. (2024) employs a dual-network strategy, with one network tasked with identifying clean labels and the other focused on refurbishing corrupted labels by estimating pseudo-labels. Another notable method by S. Zheng et al. (2020) involves a comprehensive pipeline that includes a warm-up stage, a label refurbishment stage, and a training stage designed to use a pair of loss functions that enhance temporal prediction consistency.

3.3 Robust Loss Functions

A principal set of deep-learning methods used to handle noisy labels involves 'robust loss functions' (Song et al., 2023). The cross-entropy loss function, commonly employed to train neural network classifiers, is known to be susceptible to label noise

$$L_{\mathrm{CE}}(\boldsymbol{q},k) = -\log(q_k).$$

Even modest levels of noise can lead to significant drops in generalisation, particularly in smaller datasets (Rolnick et al., 2017; Xiao, Xia, Yang, Huang, & Wang, 2015; Z. Zhang & Sabuncu, 2018). This decline in performance with the cross-entropy function raises the question of whether alternative, more robust loss functions could offer better resilience. While many approaches to handling label noise are complex, requiring multiple networks or elaborate noise detection pipelines (C. Feng et al., 2024; Han et al., 2018; J. Li et al., 2020; Malach & Shalev-Shwartz, 2017; Sachdeva et al., 2021), robust loss functions stand out due to their simplicity and minimal computational demands. This makes robust loss functions especially attractive, as they can be applied universally. In this section we review some key literature on robust loss functions, focusing particularly on their application in deep learning.

3.3.1 Lp-losses

Janocha and Czarnecki (2016) observed that Lp-losses, typically used for regression, show good robustness in a classification setting. By this, we mean loss functions of the form

$$L(\boldsymbol{q},k) = \|\boldsymbol{q} - \boldsymbol{e}_k\|_p,$$

where $\boldsymbol{e}_k$ denotes the k^{th} coordinate vector in the setting where k is the target label. The findings of Janocha and Czarnecki (2016) are consistent with those of Manwani and Sastry (2013), which showed that the MSE loss (equivalent to the L2 loss) is noise-tolerant for linear classifiers and more robust than cross-entropy in more general settings. Among the Lp losses the L1 loss (which is equivalent to Mean-Absolute Error, MAE) is known to be especially robust (Ghosh & Kumar, 2017; Y. Wang et al., 2019), albeit with a tendency to under-fit and train slowly (Ma et al., 2020).

A number of methods have tried to leverage the superior robustness of Lp losses to improve the cross-entropy loss while avoiding underfitting. Generalised Cross-Entropy (Z. Zhang & Sabuncu, 2018) constructs a family of losses which interpolate between CE and MAE in order to get the best of both loss functions. Y. Wang et al. (2019) propose a solution to cross-entropy's propensity to overfit by adding a second 'reverse cross-entropy' (RCE) term, however this RCE actually just works out to be an MAE term meaning that this approach is similar in spirit to GCE. X.-C. Li et al. (2023) builds on the work of Z. Zhang and Sabuncu (2018), providing a loss which interpolates between CE and MAE where the interpolation parameter is increased cosinusoidally through training. The work of Y. Wang et al. (2019) is extended by Ma et al. (2020) which applies normalisation to the loss functions so that they satisfy theoretical guarantees Ghosh and Kumar (2017) adhered to by MAE.

$$L_{\text{GCE}}(\boldsymbol{q},k) := \frac{1-q_k^a}{a}$$

3.3.2 Loss Correction Approaches

An important and widely studied subset of robust loss methods are loss correction approaches. In contrast to more heuristically motivated approaches such as generalised and symmetrised cross-entropies discussed in Section 3.3.1, these methods are more theoretically grounded. The goal is to use an estimate $\hat{T}$ of the noise model to alter the

loss function to negate the distorting impact of noise (Larsen, Nonboe, Hintz-Madsen, & Hansen, 1998).

$$\boldsymbol{L}(\boldsymbol{q}) \mapsto \boldsymbol{L}_{\text{corr},\widehat{T}}(\boldsymbol{q})$$

Loss correction methods come in two forms: forward corrections, where the outputs of the network are noised before being evaluated on noisy data. And backward corrections, where the noisy data is de-noised before being used to evaluate model predictions. (The difference between the forward and backward correction is illustrated in Figure 3.1). The terms 'forward' and 'backward' were introduced by Patrini et al. (2017). As alluded to in a previous section, an alternative naming convention for these correction approaches is adopted by Stempfel and Ralaivola (2009) which calls these 'bottom-up' and 'top-down' approaches respectively but we opt for using forward and backward in this thesis.

$$\begin{aligned} \text{Forward Correction:} \quad & \boldsymbol{L}(\boldsymbol{q}) \mapsto \boldsymbol{L}(\widehat{T}\boldsymbol{q}) \\ \text{Backward Correction:} \quad & \boldsymbol{L}(\boldsymbol{q}) \mapsto \widehat{T}^{-1}\boldsymbol{L}(\boldsymbol{q}) \end{aligned}$$

Forward The forward-correction methods have substantial overlap with some of the architecture-based approaches described in Section 3.2.2.2. Specifically, a noise adaption layer (Goldberger & Ben-Reuven, 2017) can be viewed as an augmentation of the network made by introducing an additional layer. However, it can equivalently be conceptualised as a property of the loss function. Numerous work employ the forward correction as an approach to label noise. Some earlier work includes Larsen et al. (1998); Mnih and Hinton (2012) although these look exclusively at binary labels.Bootkrajang and Kabán (2012) derived the forward correction for multi-class logistic regression to class-conditional label noise and Goldberger and Ben-Reuven (2017); Hendrycks et al. (2018); Patrini et al. (2017); Sukhbaatar et al. (2015); Sukhbaatar and Fergus (2014) look at class-conditional label noise in the multi-class setting for neural network classifiers.

Backward The origin of the backward correction remains unclear. In the context of deep-learning an early work is Stempfel and Ralaivola (2009) which derived the backward correction for learning in the presence of noisy binary labels using an SVM classifier. Similarly, Natarajan et al. (2013) derived the backward correction for class-conditional label noise for binary labels, demonstrating that under noisy data it is an

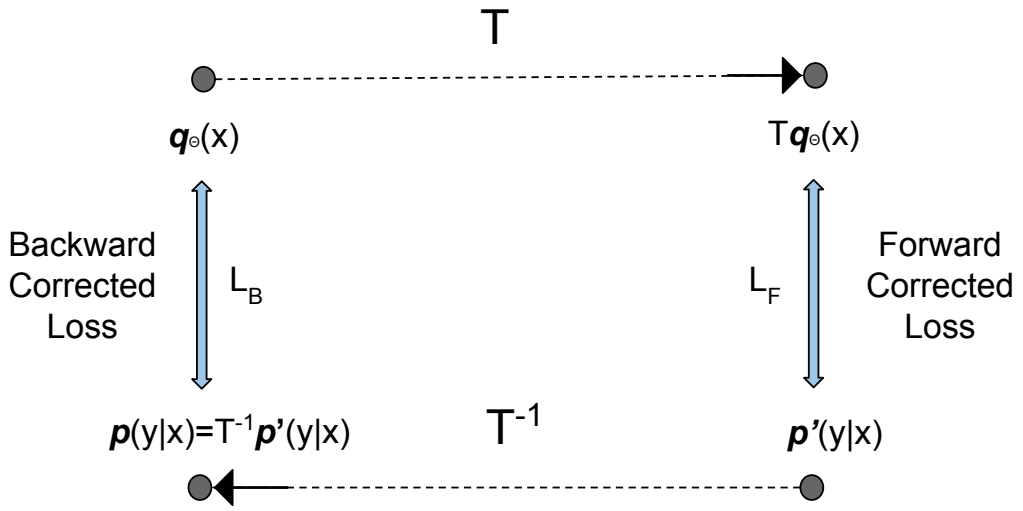


Figure 3.1: An illustration of the difference between the forward and backward correction approaches. The backward-correction (left) compares the model prediction $\boldsymbol{q}_\theta(x)$ against the de-noised noisy distribution $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x) \mapsto T^{-1}\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x)$ whereas the forward-correction (right) noises the model predictions $\boldsymbol{q}_\theta(x) \mapsto T\boldsymbol{q}_\theta(x)$ before evaluating them against the noisy distribution $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x)$.

unbiased estimator of the original loss under clean data. Van Rooyen et al. (2015) discussed the backward correction for binary labels as part of their work on noise-tolerance. Goldberger and Ben-Reuven (2017) also discusses the backward correction, arguing that while it satisfies certain desirable properties, the denoising matrix $S = T^{-1}$ cannot be learned effectively. Consequently they use a surrogate for S consisting of a symmetric label noise transition matrix at some rate learned by hyperparameter selection. Patrini et al. (2017) give a multiclass version of the backward-correction providing empirical comparisons with the forward-correction.

Loss correction methods are an effective family of approaches for handling noise labels. Nevertheless, both forward and backward corrections require an estimate of the noise transition matrix which can be difficult for a large number of classes (Goldberger & Ben-Reuven, 2017). Moreover, they typically assume that this matrix has minimal dependence on the datapoints, which impacts generality since it limits the scope to class-conditional, which is to say uniform, label noise (Goldberger & Ben-Reuven, 2017; Sukhbaatar et al., 2015).

3.3.2.1 Transition Matrix Estimation

A crucial aspect of correction-based loss functions is having an estimate of the noise transition matrix $\hat{T} \approx T$. While some methods utilising loss corrections, such as Stempfel and Ralaivola (2009), simply assume knowledge of the noise model or do not directly address the problem of matrix estimation, others provide algorithms for estimating T. We review the literature on methods for transition matrix estimation in this section.

This procedure can involve using either noisy (Patrini et al., 2017) or clean data (Hendrycks et al., 2018) to infer the noise transition matrix. A naive approach is simply to train on noisy data and then compute the confusion matrix on some clean data, using this as the transition estimate (a description of this baseline approach is given in Hendrycks et al. (2018)). Sukhbaatar and Fergus (2014) improves on this approach by estimating the noise transition matrix through computing the confusion matrices of a trained model on *both* clean and noisy data. This allows one to distinguish the impact of class confusion caused by model biases and confusions between clean classes from class confusion caused by label noise. Hendrycks et al. (2018) provides a further improvement assuming the presence of trusted clean data. Unlike the standard computation of the confusion matrix, which depends only on the predicted and true classes, their approach instead utilises the probabilities the trained model ascribes to each class: Each column of the estimated transition matrix is given as the mean of the model's predicted distribution for that class.

Xia et al. (2019) introduced the 'T-Revision' method, where an initial transition matrix is estimated from noisy data and then revised during training alongside the classifier. Yao et al. (2020) introduced the 'dual-T' estimator, which reduces estimation errors by splitting the transition matrix estimation into two simpler stages. Initially, a model is trained on noisy data. Then, an intermediate class is defined to facilitate the estimation of two transition matrices: from clean to intermediate ($T^{\clubsuit}$) and from intermediate to noisy ($T^{\spadesuit}$). These matrices are then multiplied to derive the overall transition matrix T. X. Li et al. (2021) provides a method for computing the transition matrix *during* training by adding a regularisation term to the corrected loss composed of the determinant of the estimated matrix (see Equation (3.1)). This relies on the observation that if the noisy conditional class distribution are 'sufficiently scattered' on the probability simplex, then the transition matrix is identifiable. Xia et al. (2023)

applies some of these ideas of transition matrix estimation beyond the closed-set label noise setting. They add an extra class to incorporate all open-set label noise and estimate the expanded transition matrix using the anchor points method, as described in Patrini et al. (2017). This method involves estimating the confusion matrix using specific instances called 'anchor points,' which can be unambiguously associated with a given label.

$$\boldsymbol{L}(\boldsymbol{q}) \mapsto \boldsymbol{L}(\widehat{T}\boldsymbol{q}) + \lambda \log(\det(\widehat{T})) \tag{3.1}$$

3.3.3 Noise-Tolerant Loss Functions

Within the context of this thesis, 'Noise-Tolerant loss functions' is the name given to loss functions that are innately robust, requiring no correction to achieve good generalisation despite label noise in the training set (Ghosh & Kumar, 2017). This term originates from Manwani and Sastry (2013), which defines a loss as 'Noise Tolerant' when the minimiser of the noisy and clean risks over the relevant model space Q yields identical classifiers (see Equation 3.2). However, the term 'noise-tolerant' is used with a less precise meaning in other studies, for example, Fürnkranz (1997); Nettleton et al. (2010); Pechenizkiy et al. (2006). Manwani and Sastry (2013) show that the 0-1 loss is noise-tolerant under symmetric label noise in the binary label setting. A. Menon et al. (2015) demonstrates the noise-tolerance of Balanced Error (BER) and Area Under the Curve (AUC) to label noise within the MC label noise framework. Charoenphakdee, Lee, and Sugiyama (2019) build on this work by utilising the MC noise framework to provide further theoretical insights on noise-tolerant loss functions.

$$\arg\min_{q \in Q} R_L^{\eta}(\boldsymbol{q}) = \arg\min_{q \in Q} R_L(\boldsymbol{q}) \tag{3.2}$$

The findings of Manwani and Sastry (2013) are extended in a follow-up paper (Ghosh, Manwani, & Sastry, 2015), which provides a condition under which a loss function is Noise Tolerant to simple (i.e. symmetric) label noise. Van Rooyen et al. (2015) builds on these insights, re-deriving the sufficient condition for a loss to be noise-tolerant to symmetric label noise. They define the 'unhinged loss', showing that it is the only possible convex loss satisfying the derived noise-tolerance condition. Ghosh et al. (2015) extends the results of earlier works with binary labels to the multiclass setting, offering further theoretical insights and suggesting that robustness may be improved by choosing losses that are bounded or which satisfy a '*symmetry*' property,

wherein

$$\sum_{k=1}^{c} L(\boldsymbol{q},k) = const. \tag{3.3}$$

These ideas are extended by X. Zhou et al. (2021) who derive an '*asymmetry*' condition for other class-conditional noise models. Motivated by the findings of Ghosh et al. (2015), Ma et al. (2020) show that one can take unbounded loss functions and renormalise them to achieve this objective:

$$L(\boldsymbol{q},k) \mapsto \frac{L(\boldsymbol{q},k)}{\sum_{i=1}^{c} L(\boldsymbol{q},i)}.$$

The authors note however that normalisation by itself often results in underfitting.

3.3.4 Loss Reweighting

Loss reweighting approaches employ weighting strategies to modify the loss function. These methods significantly overlap with the sample selection and reweighting strategies, discussed in Section 3.2.2. Often, whether a method is categorised as 'sample-based' or 'loss-based' reweighting is merely a matter of perspective. For clarity, we refer to a method as 'sample-based' reweighting if each sample in the training set carries a weight (which may change during training). In contrast, a method is deemed as 'loss-based' reweighting if weights are calculated at the time of loss computation. This distinction implies that loss-based methods do not require storing a large array of weights for each instance in the training set, thus reducing memory requirements.

Loss reweighting is commonly used in machine learning to address class imbalances between training and test distributions (D. Guo, Li, Zhao, Zhou, & Zha, 2022), or to adjust for discrepancies between training and test data distributions, known as covariate shift. Within the context of label noise, Stempfel and Ralaivola (2009) describe 'top-down' approaches that adjust the loss based on the noise rate at a particular location. Similarly, 'Importance Reweighting' (T. Liu & Tao, 2015) adjusts each sample's loss so that the noisy risk, when reweighted, aligns with the clean risk using the original loss function. Shen and Sanghavi (2019) introduce the 'Trimmed Loss', which excludes high-loss samples within a batch from the network parameter update, effectively assigning them zero weight for that iteration. The 'Curriculum Loss' (T. Zhou et al., 2020) adopts a similar strategy. Kumar and Amid (2021) calculate weights for each element in a minibatch by solving a constrained optimisation problem that down-weights higher-loss samples while maintaining a degree of uniformity. The approach in Englesson

and Azizpour (2024) involves a learnable variance function that reweights the loss, decreasing the weights assigned to more uncertain samples. X. Wang, Hua, Kodirov, Clifton, and Robertson (2023) presents the 'Improved MAE,' a modification of the MAE loss that reweights samples to more heavily emphasise those for which the model exhibits uncertainty.

3.3.5 Regularisation-Based Loss Functions

A major set of robust loss approaches to label noise are loss regularisation based approaches. These methods mitigate overfitting by regularising the loss function during training. This may come in various forms. For example, adding terms to the loss function so that the network optimises multiple objectives can prevent overfitting to the primary task. Augmenting the model predictions or targets within the loss is another approach. This section reviews the various loss regularisation methods for handling noisy labels.

Consistency Regularisation Consistency-regularisation approaches add loss terms

$$L \mapsto L + L_{\text{reg}},$$

to achieve consistency between different views of a data sample or consistency of model predictions for a sample during training (Englesson & Azizpour, 2021a). For example, 'MixUp' (H. Zhang et al., 2017) computes convex combinations of data-label pairs before applying the loss function. Similar augmentation-based strategies described in Section 3.2.2.1 can be viewed as loss-based regularisation approaches (Cubuk et al., 2019, 2020). (Englesson & Azizpour, 2021b) introduces Generalised Jensen-Shannon divergence (GJS),' which measures the divergence between model predictions on different augmentations of a data sample. 'Early-Learning Regularisation (ELR)' (S. Liu et al., 2020) adds a regularisation term to the cross-entropy loss which encourages the model to produce consistent predictions during training for the same sample rather than different viewpoints of the same sample. Iscen, Valmadre, Arnab, and Schmid (2022) introduces 'Neighbourhood Consistency Regularisation' which encourages the model to give similar predictions for examples with similar feature representations. Sun, Zhang, and Ma (2024) incorporates a regularisation term to ensure consistency in predictions among samples within the same cluster. These clusters are determined using a contrastive learning approach that relies on data augmentations and

does not use label information. Y. Wang et al. (2018) also uses a contrastive loss to push same-class, correctly-labelled instances together in feature space, where 'correctly labelled' is determined by an outlier detection approach using the features of each instance.

Other Regularisation-Based Approaches 'Label Smoothing' (Szegedy et al., 2016) can be conceptualised as a form of loss-based regularisation. It operates by mixing the potentially noisy target labels with a uniform distribution

$$\boldsymbol{y} \mapsto (1-\alpha)\boldsymbol{y} + \alpha\left(\frac{1}{c}, \frac{1}{c}, \ldots, \frac{1}{c}\right)$$

before evaluating the model's predictions. This technique has been shown to improve both calibration and performance across a variety of tasks (Ferianc, Bohdal, Hospedales, & Rodrigues, 2024) and is particularly effective in environments with noisy labels (J. Wei et al., 2021). Reed et al. (2014) introduces bootstrapping for learning with noisy labels. This approach mixes the noisy label with the model's prediction to create a new target to train the model.

$$\boldsymbol{y} \mapsto (1-\alpha)\boldsymbol{y} + \alpha\boldsymbol{q}(x)$$

In 'soft' bootstrapping, this target is the model's predicted distribution over labels, whereas in 'hard' bootstrapping, it is formed using only the model's predicted class. Ishida, Yamane, Sakai, Niu, and Sugiyama (2020) bound the training loss below to prevent overfitting. Fatras et al. (2019) introduce Wasserstein Adversarial Regularisation, adding a regularisation term based on the Wasserstein distance to the loss function. This term penalises prediction changes for small perturbations in the input space.

3.3.6 Miscellaneous Loss-Based Approaches

Amid, Warmuth, Anil, and Koren (2019) introduces the 'Bi-Tempered Logit loss' to handle noisy labels. This approach generalises the log and softmax functions commonly used to compute the loss, switching to a heavily-tailed 'tempered' softmax function, paired with a lower-temperature tempered log function. These loss functions are typically bounded and show good robustness properties. Xu et al. (2019) introduces the 'Determinant based Mutual Information' loss (DMI). This is an information-theory-based approach motivated by the observation that increasing the mutual information (MI) between a model and noisy labels does not guarantee an increase in MI between

the model and clean labels. However, DMI satisfies this desirable property and can be used to formulate a robust loss. L. Feng et al. (2021) introduce 'Taylor Cross-Entropy' which consists of taking the first k terms in the Taylor expansion of the log loss.

$$L(\boldsymbol{q},k) = -log(q_k) \approx (1-q_k) + \frac{(1-q_k)^2}{2}$$

Y. Liu and Guo (2020) defines 'peer loss' functions as a combination of two terms: the first term is a standard loss function that evaluates the model on noisy data, and the second term evaluates the model on a randomly selected pair of datapoints and their noisy labels, offsetting the noise impact. D.-B. Wang, Wen, Pan, and Zhang (2021) proposes 'complementary loss functions' that train deep neural networks by applying Cross-Entropy (CE) to pseudo-labelled 'easy' samples likely to be correct, and a robust loss, such as MAE, to 'hard' samples likely to have noisy labels.

3.3.7 Chapter Summary

This section has reviewed pertinent literature on label noise robust algorithms. Section 3.1 explored the historical context of label noise robust classification, discussing label noise taxonomies and addressing the issue of identifiability. Section 3.2 summarised approaches to noise robustness both within a deep-learning context and for pre-deep-learning classification algorithms. Section 3.3 examined robust loss functions, with a particular focus on loss-correction methods, noise-tolerant loss functions, loss-reweighting techniques, and regularisation-based loss strategies.

This review is not exhaustive; in each subsequent chapter, we supplement this context with a more detailed review of literature pertinent to the specific topics discussed. For a more comprehensive overview of learning in the presence of noisy labels, we recommend consulting several key survey papers: Frenay and Verleysen (2014), which has a broader focus on non-deep methods; (Song et al., 2023), which concentrates on deep learning methods; Han et al. (2020), known for its comprehensive sections on objective function and transition matrix-based approaches; Algan and Ulusoy (2021), which focuses on noisy image classification; and Johnson and Khoshgoftaar (2022), which deals with big data challenges. Additional insights can be found in (Nigam, Dutta, & Gupta, 2020) and other related works.

Chapter 4

Class-Preserving Label Noise

4.1 Introduction

Before we dive into the specific contributions to improving label noise robustness in Chapters 5 onwards, in this chapter, we elaborate on the class-preserving assumption introduced in Section 2.2.2. This minor content chapter is dedicated to showing two things:

1. No fixed[1] loss can be robust to all possible label noise models. Consequently, there is a maximal subset of noise models over which a loss function can be robust. These are precisely the class-preserving label noise models.

2. Most label noise previously studied in the literature on label noise robust loss functions is class-preserving.

Establishing this second fact is crucial as it shows that the class-preserving assumption we are forced to adopt is broad, containing most frequently studied noise models. The findings of this chapter also support the assertion made in Section 2.4 that a lack of robustness is (typically) caused by a loss function's propensity to induce overfitting rather than a failure to satisfy theoretical guarantees. This motivates Chapter 6 and Chapter 5, where we explore ways to mitigate overfitting.

[1] By 'fixed' we mean that the loss function is not defined in terms of the noise model as is the case for say the backward correction.

4.1.1 Related Work

Identifiability Without assumptions about the label noise model or data distribution, learning in the presence of label noise can suffer from the problem of 'non-identifiability,' where, say, the clean data distribution cannot be uniquely determined from the noisy data. This issue occurs in methods that estimate the noise transition matrix: Given the noisy data and no prior knowledge about the data distribution, there are typically multiple ways to decompose the observed data into a noise model and a clean data distribution (Fu et al., 2018; X. Li et al., 2021). To address the non-identifiability problem, practitioners often make some standard assumptions about the data distribution and the label noise model. Notable common assumptions include the separability of the data distribution (Ghosh & Kumar, 2017; Manwani & Sastry, 2013) and assuming diagonal dominance of the transition matrix for class-conditional label noise (X. Li et al., 2021; Patrini et al., 2017; Xia et al., 2019; Xu et al., 2019). In other studies they adopt conditions which work out to be special cases of the class-preserving condition. Each theorem in Ghosh and Kumar (2017) uses a class-preserving assumption. Cannings et al. (2020) use a definition equivalent to class-preserving label noise for binary labels. Rolnick et al. (2017) use an implicit class-preserving assumption in their experiments. Earlier works with binary labels (Angluin & Laird, 1988; Cohen, 1997; Stempfel & Ralaivola, 2009) make restrictions on the noise model which amount to a class-preserving assumption. Most relevant is the 'clean-labels-domination' assumption defined in X. Zhou et al. (2021) which, while defined with respect to a dataset rather than a distribution, maps very closely onto the definition of class-preserving.

Class-Preserving In this work, our focus is on recovering the clean Bayes classifier from noisy data rather than the noise transition matrix, which represents a weaker form of identifiability since the Bayes classifier can be uniquely recovered even when the noise model is non-identifiable. As we see in Section 4.2.1, a sufficient condition for this is that the noise model is 'class-preserving' for the clean data-label distribution. To our knowledge, this is the first time this type of noise has been explicitly named and categorised. However, many existing works implicitly rely on class-preserving noise or use assumptions corresponding to special cases. For example, Theorem 1 in Ghosh and Kumar (2017) addresses symmetric label noise with $\eta < \frac{c-1}{c}$, which Section 4.3 shows is class-preserving. Theorem 2 in the same study extends this threshold to non-uniform label noise, while Theorem 3 deals with diagonally dominant class-conditional noise under separability. In each case, these correspond to class-

preserving assumptions (Corollary 4.3.3 and Lemma 4.3.7 respectively). Similarly, Manwani and Sastry (2013) examines binary noise, imposing a $\eta < 0.5$ threshold for uniform and non-uniform symmetric noise, which we show is also class-preserving. In essence, the class-preserving assumption consolidates and formalises several common assumptions already present in the literature.

4.1.2 Outline and Preliminaries

Chapter Outline Section 4.2 restates the definition of class-preserving label noise and presents a sketch 'No-Free-Lunch' Theorem explaining how no loss can be robust to all possible label noise models, motivating the class-preserving assumption. Section 4.3 provides conditions under which various standard noise models are class-preserving. Section 4.4 demonstrates that most standard datasets used by practitioners exhibit sharp conditional class distributions, meaning that each data sample is strongly associated with one particular label. Combining the results of the previous sections, Section 4.5 shows that almost all experiments on label noise in the relevant literature focus on class-preserving noise.

4.2 Class-Preserving Label Noise

Dominant class-preserving label noise (usually abbreviated to 'class-preserving label noise') describes any label noise model which do not alter the most likely (dominant) class at each $x \in \mathcal{X}$. A formal restatement of class-preserving label noise is given below.

Definition 4.2.1 (Dominant Class)**.** *For a given data-label distribution $p(x,y)$ over a set $\mathcal{X}$, the dominant class at a data point $x \in \mathcal{X}$ is defined as the class with the highest conditional probability at x. This is denoted by $k_{max}(x)$ and is mathematically represented as:*

$$k_{max}(x) := \arg\max_{i \in \mathcal{Y}} p(y = i \mid x).$$

The dominant class represents the most probable class label given the data point x. We call $p_{max}(x) := \max_{i \in \mathcal{Y}} p(y = i \mid x)$ the dominant class probability.

Definition 4.2.2 (Class-Preserving Noise)**.** *Given a data-label distribution $p(x,y)$ and its noisy version $\widetilde{p}(x,\widetilde{y})$, resulting from label noise, the noise is considered class-preserving if, for every $x \in \mathcal{X}$, the dominant class remains unchanged after noise application. Formally, this is expressed as:*

$$\underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \widetilde{\boldsymbol{p}}(\widetilde{y}=i|x) = \underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \boldsymbol{p}(y=i|x),$$

where c is the number of classes.

As the name suggests, class-preserving noise *preserves* whichever class has the highest probability. For example, suppose we have a labelled dataset of images of animals. We may expect some of the wolves to be mislabelled as dogs. If wolves are *more likely* to be mislabelled as dogs than correctly labelled as wolves, then this noise is *not* class-preserving.

4.2.1 A No-Free Lunch Theorem For Label Noise

In Section 4.1.1 we discussed 'identifiability', explaining how it is essential to make limiting assumptions about the label noise model since no learning algorithm can be robust to all combinations of distribution and label noise. In this section we support this assertion, giving a sketch no-free-lunch (NFL) theorem (Wolpert & Macready, 1997) for learning in the presence of label noise. This argument is not rigorous; we do not take care to formally define the domains of the relevant distributions or write a formal theorem statement. The goal is to intuitively demonstrate the impossibility of having a general-purpose algorithm which can be robust to a sufficiently diverse set of distributions and noise models. We show that the class-preserving assumption emerges as a natural resolution to the problem of non-identifiability.

NFL Setup Let $\mathcal{A}$ be a (possibly random) learning algorithm which takes a noisily-labelled dataset $\widetilde{\mathcal{D}} \coloneqq \{(x_i,\widetilde{y}_i)\}_{i=1}^N$ and outputs a classifier $f : \mathcal{X} \to \mathcal{Y}$. We assume that the dataset samples are drawn i.i.d from a latent data-label distribution $p(x,y)$, upon which a latent noise model (unknown to the algorithm) $r(\widetilde{y} \mid x,y)$ noises the labels. We evaluate the clean 0-1 risk of the classifier output by the algorithm $\mathcal{A}$ and take an average across all possible datasets (and algorithm seeds where relevant). Thus an algorithm $\mathcal{A}$ induces a map, which we denote $\mathcal{A}_R$, taking a distribution $p(x,y)$ and a noise model $r(\widetilde{y} \mid x,y)$ and outputting a misclassification rate;

$$\mathcal{A}_R(p(x,y), r(\widetilde{y} \mid x,y) \coloneqq \mathbb{E}_{\widetilde{\mathcal{D}}\sim\widetilde{p}(x,\widetilde{y})}\left[R_{0\text{-}1}\left(\mathcal{A}(\widetilde{\mathcal{D}})\right)\right] \in [0,1].$$

Given a distribution $p(x,y)$ and a noise model $r(\widetilde{y} \mid x,y)$, $\mathcal{A}_R$ measures how well (on average) the classifier learned by the algorithm $\mathcal{A}$ generalises to $p(x,y)$.

NFL Sketch The key insight is identifying that $\mathcal{A}_R$ depends only on the resulting noisy distribution $\widetilde{p}(x,\widetilde{y})$, not on the clean distribution and noise model ($p(x,y), r(\widetilde{y} \mid x,y)$ respectively) which generate it. Therefore, given data-distributions/noise-model pairs; $(p_1(x,y), r_1(\widetilde{y} \mid x,y)$ and $(p_2(x,y), r_2(\widetilde{y} \mid x,y)$, which generate the same noisy distribution $\widetilde{p}(x,\widetilde{y})$,

$$p(x,y) = \sum_y p_1(x,y) r_1(\widetilde{y} \mid x,y) = \sum_y p_2(x,y) r_2(\widetilde{y} \mid x,y),$$

an algorithm $\mathcal{A}$ will make the same predictions for both p_1 and p_2. This is *despite* the fact that $p_1(x,y)$ and $p_2(x,y)$ potentially have dramatically different Bayes-optimal classifiers. It follows that unless we restrict the problem space, i.e. the domain of admissible data-distribution/noise-model pairs, then every algorithm must have a region in the problem space in which it performs poorly. The following example with binary labels illustrates this idea.

Example Let $r_1(\widetilde{y} \mid y), r_2(\widetilde{y} \mid y)$ be two class-conditional noise models expressible by the transition matrices

$$T_1 \coloneqq \begin{bmatrix} 0.4 & 0.6 \\ 0.6 & 0.4 \end{bmatrix}, \quad T_2 \coloneqq \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$$

respectively. Given some data-label distribution $p(x,y)$, let $p_1(x,y)$ denote the distribution obtained by noising $p(x,y)$ using T_1 and then let $p_2(x,y)$ denote the distribution obtained by noising $p_1(x,y)$ using T_2. This is represented in Figure 4.1.

Given the distribution $p_2(x,y)$, there are two ways[2] to decompose this distribution into a clean-distribution and noise-model;

1. Clean distribution $p_1(x,y)$ corrupted by noise model T_1.
2. Clean distribution $p(x,y)$ corrupted by noise model T_2T_1,

where

$$T_2T_1 = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix} \begin{bmatrix} 0.4 & 0.6 \\ 0.6 & 0.4 \end{bmatrix} = \begin{bmatrix} 0.42 & 0.58 \\ 0.58 & 0.42 \end{bmatrix}.$$

[2]There are an infinite number of ways to decompose p_2 into a clean distribution and a noise model but we consider just two.

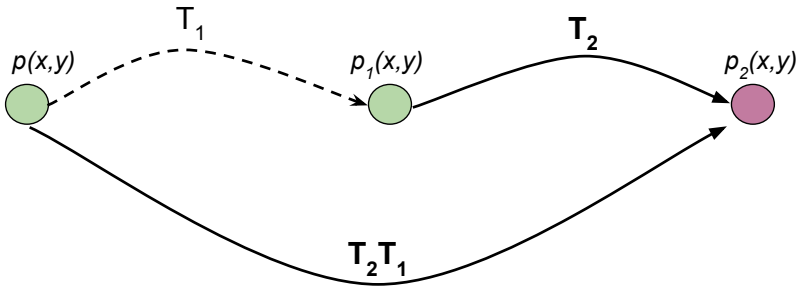


Figure 4.1: $p(x,y)$ is noised by T_1 to create $p_1(x,y)$ which is noised by T_2 to create $p_2(x,y)$. Thus, only observing $p_2(x,y)$, we do not know whether the noise model is T_2 (applied to $p_1(x,y)$), or T_2T_1 (applied to $p(x,y)$).

Given $p_2(x,y)$, and without prior knowledge about the noise model (or clean data-distribution), there is no way to know which of these decompositions is 'correct'. The significance of this is that $p(x,y)$ and $p_1(x,y)$ have dramatically different Bayes-optimal classifiers: Given a point $x_0 \sim p(x)$ where $\boldsymbol{p}(y \mid x_0) = (1,0)$ we have $\boldsymbol{p}_1(y \mid x_0) = (0.4,0.6)$. Thus while for $p(x,y)$ the Bayes-optimal classifier predicts $f(x_0) = 1$, the Bayes-optimal classifier for $p_1(x,y)$ predicts $f(x_0) = 2$. In particular, this must mean that any algorithm which generalises well for $(p_1(x,y), T_2)$ must generalise poorly for $(p(x,y), T_2T_1)$.

Restricting the Problem Domain Given two decompositions $(p_1(x,y), r_1(\widetilde{y} \mid x,y))$ and $(p_2(x,y), r_2(\widetilde{y} \mid x,y))$ of some noisy distribution $\widetilde{p}(x,\widetilde{y})$, we say that these decompositions are *equivalent*, written $(p_1(x,y), r_1(\widetilde{y} \mid x,y)) \sim (p_2(x,y), r_2(\widetilde{y} \mid x,y))$, if $p_1(x,y)$ and $p_2(x,y)$ have the same Bayes classifiers. It is simple to demonstrate, given noisy distribution $\widetilde{p}(x,\widetilde{y})$, that $\sim$ defines an equivalence relation. To avoid the decomposition problem outlined above, we must limit our problem domain so that whenever $(p_1(x,y), r_1(\widetilde{y} \mid x,y))$ and $(p_2(x,y), r_2(\widetilde{y} \mid x,y))$ induce the same noisy distribution then they must lie in the same equivalence class. For each $\widetilde{p}(x,\widetilde{y})$ we opt for the equivalence containing $(\widetilde{p}(x,\widetilde{y}), id)$ where *id* denotes the trivial noise model which does not alter the labels. We call this restriction of the set of admissible noise-model/distribution pairs the '*class-preserving* assumption'. We call the set of noise models r for which $(p(\widetilde{y} \mid x,y), p(x,y)) \sim (\widetilde{p}(x,\widetilde{y}), id)$ the '*class-preserving noise models for* $p(x,y)$'.

4.2.2 Conclusion

The no-free lunch sketch illustrates that, without restrictions on either the data distribution or noise model, we cannot expect to recover the clean Bayes classifier from noisy data in all cases. In particular, one cannot define a loss function such that performing empirical risk minimisation on noisy data with this loss will always result in good generalisation to the clean data distribution. Put simply; there is no fixed, universally robust loss function. We went on to show that under a class-preserving assumption it *is* possible to recover the Bayes-classifier from noisy data. Consequently, by making a class-preserving noise assumption we avoid identifiability issues.

4.2.3 When Is Label Noise Class-Preserving?

We have shown that it is necessary to make some limiting assumption about the label noise model in order to ensure identifiability. Specifically we assume that the label noise is class-preserving. A natural question is how limiting this assumption is. Throughout the remainder of this chapter we explore this question, demonstrating that class-preserving noise is a broad set of label noise. For example, we show in Section 4.3 that symmetric noise at a rate of less than $\frac{c-1}{c}$ where c is the number of classes is class-preserving; for instance, in the case of $c = 100$ classes, symmetric noise of any rate up to 99%.

More generally, we show that *the vast majority of label noise used in the experiments sections of the relevant literature is class-preserving.* This finding is summarised in Tables 4.1 and 4.2, which detail the symmetric and asymmetric noise experiments, respectively, from a review of 11 papers on label noise robust loss functions. We give the dataset/noise-rate combinations used in every experiment from these papers. When the label noise is class-preserving, the noise rate is highlighted in green; otherwise, it is highlighted in red. Each paper is cited according to the specific loss function it introduces. The loss functions analysed include Generalised Cross Entropy (**GCE**) (Z. Zhang & Sabuncu, 2018), Normalised Cross Entropy (**NCE**) (Ma et al., 2020), Taylor Cross Entropy (**TaylorCE**) (L. Feng et al., 2021), Symmetric Cross Entropy (**SCE**) (Y. Wang et al., 2019), Improved MAE (**IMAE**) (X. Wang et al., 2023), Bootstrap Loss (**Bootstrap**) (Reed et al., 2014), Forward Loss Correction (**FCorrection**) (Patrini

et al., 2017), Generalised Jensen-Shannon (**GJS**) (Englesson & Azizpour, 2021b), and Early-Learning Regularisation (**ELR**) (S. Liu et al., 2020). Only three of the 211 experiments reviewed are not class-preserving, all from the same paper (Patrini et al., 2017).

Demonstrating the claims made by Tables 4.1 and 4.2 requires groundwork. In Section 4.3, we derive some sufficient conditions for label noise to be class-preserving. Each of these conditions is similar, giving a required bound on the noise rate and a condition that the conditional class distributions are sufficiently 'peaked', meaning that there is a clear label for any given datapoint $x \in \mathcal{X}$. In Section 4.4, we demonstrate that each of the datasets in Table 4.2 have highly peaked conditional class distributions. In Section 4.5, we explain the structure of the label noise for all the experiments in Table 4.2. Using the results of Sections 4.3,4.4, we can show that this noise is class-preserving in most cases.

Table 4.1: Experiments implementing class-preserving (green) and non-class-preserving (red) **symmetric** label noise from a summary of 11 papers studying label noise robust loss functions. Each row corresponds to a distinct paper, while columns represent various datasets. Within the cells, we list the percentage noise rates applied by each paper to the respective dataset. Noise rates are highlighted in green when they denote class-preserving noise would be in red when they do not. For example, the CIFAR10 dataset in the GCE paper is analysed with noise rates of 20%, 40%, 60%, and 80%, which are all class-preserving. All settings using used in all papers studied are class-preserving.

Paper	CIFAR10	CIFAR100	MNIST	FashionMNIST	IMDB	Kuzushiji	TinyImageNet
GCE	20, 40, 60, 80	20, 40, 60, 80		20, 40, 60, 80			
NCE	20, 40, 60, 80	20, 40, 60, 80	20, 40, 60, 80				
TaylorCE	20, 40, 60, 80	20, 40, 60, 80	20, 40, 60, 80	20, 40, 60, 80		20, 40, 60, 80	
SCE	20, 40, 60, 80	20, 40, 60, 80	20, 40, 60, 80				
IMAE	20, 40, 60	20, 40, 60					
Bootstrap Loss	18, 45, 63, 72, 77, 81	20, 50, 69, 79, 84, 89					20, 50, 80
FCorrection	20	20	20		10		
GJS	20, 40, 60, 80	20, 40, 60, 80					
VolMinNet	20, 50	20, 50	20, 50				
CL	10, 20, 30, 40, 50	10, 20, 30, 40, 50	10, 20, 30, 40, 50				20, 50
ELR	20, 40, 60, 80	20, 40, 60, 80					

Paper	CIFAR10	CIFAR100	MNIST	FashionMNIST	IMDB	Kuzushiji	Clothing1M
GCE	10, 20, 30, 40	10, 20, 30, 40		10, 20, 30, 40			
NCE	10, 20, 30, 40	10, 20, 30, 40	10, 20, 30, 40				
TaylorCE	10, 20, 30, 40	10, 20, 30, 40	10, 20, 30, 40	10, 20, 30, 40		10, 20, 30, 40	
SCE	20, 30, 40	20, 30, 40	20, 30, 40				38
IMAE		20, 30, 40					38
Bootstrap Loss							38
FCorrection	20, 60	20, 60	20, 60		10, 40		38
GJS	20, 40	20, 40					
VolMinNet	20, 45	20, 45	20, 45				38
CL	35	35	35				
ELR	10, 20, 30, 40	10, 20, 30, 40					38

Table 4.2: Experiments implementing class-preserving (green) and non-class-preserving (red) **asymmetric** label noise from a summary of 11 papers studying label noise robust loss functions. Each row corresponds to a distinct paper, while columns represent various datasets. Within the cells, we list the percentage noise rates applied by each paper to the respective dataset. Noise rates are highlighted in green when they denote class-preserving noise and in red when they do not. All settings used in all papers studied are class-preserving with one exception: 60% noise on the MNIST dataset for (Patrini et al., 2017).

4.3 Sufficient Conditions for Noise to be Class-Preserving

Lemma 4.3.1. *Symmetric noise (uniform) is class-preserving for any data-label distribution $p(x,y)$ whenever the noise rate η satisfies $\eta < \frac{c-1}{c}$.*

Lemma 4.3.1 states that (uniform) symmetric noise is class-preserving so long as it does not exceed the threshold noise rate of $\frac{c-1}{c}$. This bound is respected in all experiments in Table 4.1, meaning that these experiments are class-preserving. We should note that in (Reed et al., 2014), they claim to use a noise rate of 90% for CIFAR10, which would be non-class-preserving. However, they use a different noise definition in this paper, leaving the original label in the set from which they select the noisy label. Hence, their effective noise rate is 81% and is class-preserving.

4.3.0.1 Universally Class-Preserving: Symmetric Label Noise

The following Lemma establishes the stronger property that symmetric label noise at rate $\eta < \frac{c-1}{c}$ is the *only* class-dependent (uniform) noise model, which is class-preserving for *all* $p(x,y)$. We call this *universally class-preserving* since it is class-preserving for all distributions.

Clarification To clarify, most noise models are class-preserving for some data-label distributions $p(x,y)$ and are not class-preserving for other data-label distributions. Symmetric label noise is class-preserving for *any* $p(x,y)$; this is a property shared by no other class-conditional label noise models. This is an important property of symmetric label noise, which we establish to expedite some of the proofs in Chapter 7.

Intuition Given *asymmetric* label noise, one may always construct a probability vector $\boldsymbol{p}$ with two very close largest entries, where the asymmetry will tip more probability mass one way than the other and hence, in one direction, change the dominant class.

Lemma 4.3.2. *Let T be a transition matrix for class-conditional label noise. Then, T represents symmetric label noise at some rate $\eta < \frac{c-1}{c}$ if and only if, for all $\boldsymbol{p} \in \Delta$,*

$$\arg\max_i (T\boldsymbol{p})_i = \arg\max_i \boldsymbol{p}_i. \tag{4.1}$$

Non-Uniform Symmetric Label Noise If we relax the condition that the label noise is uniform, one obtains the following corollary characterising all universally class-preserving label-noise models.

Corollary 4.3.3. *A label noise model is class-preserving for all distributions $p(x,y)$ if and only if it describes (possibly non-uniform) symmetric label noise where, for all $x \in \text{supp}(p(x))$, the noise rate satisfies $\eta(x) < \frac{c-1}{c}$.*

4.3.0.2 Pairwise and Circular Label Noise

Pairwise Label Noise The following gives conditions under which pairwise label noise (Defined in Section 2.2.1) is class-preserving.

Lemma 4.3.4. *Let $p(x,y)$ be a data-label distribution. Pairwise label noise with mislabelling probability $\eta < \frac{1}{2}$ is class-preserving if for every x, the dominant class probability $p_{max}(x) := \max_i p(y=i \mid x)$ and the next highest class probability $p_{res}(x) := \max_{i \neq k} p(y=i \mid x)$, satisfy $p_{max}(x) \geq \frac{1-\eta}{1-2\eta} p_{res}(x)$.*

Circular Label Noise As circular label noise (Defined in Section 2.2.1) is a type of pairwise label noise Lemma 4.3.4 establishes the following corollaries.

Corollary 4.3.5. *Let $p(x,y)$ be a data-label distribution. Circular label noise is class-preserving for $p(x,y)$ if, the dominant class probability $p_{max}(x) := \max_i p(y=i \mid x)$ and the next highest class probability $p_{res} := \max_{i \neq k} p(y=i \mid x)$, satisfy $p_{max}(x) \geq \frac{1-\eta}{1-2\eta} p_{res}(x)$ with $\eta < \frac{1}{2}$.*

Corollary 4.3.6. *For a system with classes partitioned into m equal-sized subgroups and circular label noise applied within each subgroup, the noise is class-preserving if, for each subgroup, for each x, the dominant class probability $p_{max}(x) := \max_i p(y=i \mid x)$ and the next highest class probability $p_{res}(x) := \max_{i \neq k} p(y=i \mid x)$, satisfy $p_{max}(x) \geq \frac{1-\eta}{1-2\eta} p_{res}(x)$, with p_{res} being the next highest probability within the subgroup, and $\eta < \frac{1}{2}$.*

4.3.0.3 Class-Preserving vs Diagonally Dominant

Class-preserving label noise is closely related to diagonally dominant (DD) noise (refer to Section 2.2.1 for a definition of DD), but it is a more general definition. Specifically, in scenarios where the data-label distribution is *separable*—i.e., the class-conditional distributions $p(x \mid y=k)$ have non-overlapping supports—the class-preserving transition matrices coincide exactly with diagonally dominant transition matrices. However, for arbitrary distributions $p(x,y)$, DD transition matrices may not be class-preserving.

DD is Class-Preserving for Separable Distributions

Lemma 4.3.7. *Suppose that a data-label distribution is separable. Suppose $p(x,y)$ is corrupted by class-conditional, diagonally dominant label noise. This label noise model is class-preserving for $p(x,y)$.*

4.4 The Standard Curated Image Datasets have Sharp Conditional Class Distributions

4.4.1 Clean Conditional Class Distributions

The conditional class distribution, denoted $p(y \mid x)$, defines the probability of an instance x belonging to each class y. In complex domains, such as medical diagnosis, $p(y \mid x)$ may be spread across multiple classes, reflecting uncertainty or symptom overlap between conditions. In contrast, curated image datasets like MNIST, FashionMNIST, and CIFAR-10 tend to exhibit sharply peaked distributions, where $p(y \mid x)$ assigns high probability to a single class, confidently indicating the correct label.

The extreme case is when all probability mass is on one label, known as an *anchor point* (X. Li et al., 2021). A separable distribution consists solely of anchor points. In the next section, we argue that the class distributions in standard image datasets—particularly those in Table 4.2—are (typically) highly peaked, approximating anchor points. Using this property and results from Section 4.3, we substantiate that most label noise in these datasets is class-preserving.

4.4.1.1 Image Dataset Conditional Class Distributions

Given an image dataset it is difficult to precisely quantify what the conditional class distributions look like since they are not directly observable. However, we can get some idea of the peakedness of the true class distributions for a given dataset by looking at the **test** accuracy obtained by a model trained on this dataset.

Table 4.3: Test Accuracies for Various Datasets. The test accuracy values for CIFAR are taken from (Dosovitskiy et al., 2020), FashionMNIST taken from (Tanveer et al., 2021), IMDB from (Yang et al., 2019) and MNIST from (Byerly et al., 2021)

Dataset	**Test Accuracy (%)**
CIFAR-10	99.50
CIFAR-100	94.55
MNIST	99.87
Fashion-MNIST	96.91
IMDB	96.21

Test Accuracy Consider a data-label distribution $p(x,y)$ over a set $\mathcal{X}$. For each data point $x \in \mathcal{X}$, let $p_{max}(x)$ denote the *dominant class probability* $p_{max}(x) := \max_i p(y = i \mid x)$. The maximum achievable accuracy by any classifier on this distribution, often referred to as the accuracy of a Bayes-optimal classifier, is given by the expected value of $p_{max}(x)$ over the distribution of x, which is $\mathbb{E}_{x\sim p(x)}[p_{max}(x)]$. To express this as a percentage, we multiply by 100.

This value serves as an upper limit on the accuracy attainable by any classifier. Therefore, a probability estimator $q : \mathcal{X} \to \Delta$, when evaluated on a sufficiently large, independent and identically distributed (i.i.d) test dataset drawn from $p(x,y)$, will almost certainly achieve an accuracy that does not exceed this limit.

In practice, the test accuracy obtained by a model on a large held-out test set, therefore, provides an empirical lower bound for $\mathbb{E}_{x\sim p(x)}[p_{max}(x)]$. For instance, a Vision Transformer model (ViT) cited in (Dosovitskiy et al., 2020) achieves a test accuracy of 99.50% on the CIFAR-10 dataset. This high accuracy suggests that the average p_{max} is at least 0.995, signifying that CIFAR-10 is near-separable with very sharply peaked class distributions. The accuracies obtained by SoTA models on the CIFAR100, FashionMNIST, MNIST and IMDB datasets are given in Table 4.3. In each case, the exceptionally high test accuracies (All $> 94.55\%$) indicate that these datasets have highly peaked class distributions, with a clear dominant class at most $x \in \mathcal{X}$.

Caveat In practice, however, it is common to reuse the same test set across various model iterations and years of research, which can lead to potential overlaps in information between the model and the test data. This repeated use makes it difficult to ensure the test data remains unseen by practitioners, potentially causing models to overfit to these test sets. Consequently, this can inflate perceived model performance and the estimated accuracy of the Bayes Classifier.

4.5 Asymmetric Noise Experiments are Class-Preserving

In this section, we go into the details of how the noise is constructed for each of the experiments given in Table 4.2, demonstrating that the noise is indeed class-preserving in all but three cases as we have claimed.

CIFAR10 The label noise for CIFAR10 is constructed by mapping pairs of classes into each other, specifically, with probability η one transitions $Truck \to Autombile$, $Bird \to Plane$, $Deer \to Horse$, $Cat \leftrightarrow Dog$ (Patrini et al., 2017). This label noise is used in all papers which use this dataset with asymmetric label noise except (X. Li et al., 2021). In (X. Li et al., 2021), they use circular label noise, flipping each class to the following class with probability η.

MNIST Most studies utilising MNIST with asymmetric noise adopt a procedure similar to that used for CIFAR10, wherein pairs of classes are switched with a probability η. Examples include transitions from $2 \to 7$, $3 \to 8$, $5 \to 6$, and $7 \to 2$. An exception is noted in the work by X. Li et al. (2021), who employ circular label noise similarly to their approach with CIFAR10.

FashionMNIST The synthetic asymmetric label noise for FashionMNIST is constructed in a similar fashion to CIFAR10 and MNIST, via pairwise transitions; $Boot \to Sneaker$, $Sneaker \to Boot$, $Sneaker \to Sandals$, $Pullover \to Shirt$, $Coat \leftrightarrow Dress$.

Clothing1M Clothing1M differs from the other datasets because the label noise is not synthetic but results from genuine labelling errors. The dataset is believed to contain around 38% corrupted labels (Y. Wang et al., 2019). This noise is primarily pairwise, with *sweater* and *knitwear* being examples of confused categories.

IMBD Unlike the other datasets mentioned, IMDB is not an image dataset; instead, it consists of reviews labelled either as positive or negative. Asymmetric noise is constructed by transitioning negative labels to positive at rate 5% and positive to negative at a rate x where x is given in Table 4.2. Some snippets from the IMDB dataset may be found in Table 4.4.

The asymmetric label noise experiments on CIFAR10, MNIST, FashionMNIST Clothing1M and IMDB use pairwise label noise. By Lemma 4.3.4, this noise is therefore class-preserving so long there is a heavily dominant class for each x, specifically $p_{max} \geq \frac{1-\eta}{1-2\eta} p_{res}$ and $\eta < 1/2$. At a noise rate of $\eta = 0.4$, this means we require $p_{max} \geq 3p_{res}$ where p_{max} denotes the probability of the dominant class and p_{res} the second most dominant. Each curated image dataset consists of clear images of a single subject. This implies that the class distributions for these datasets are highly peaked, meaning that $p_{max} >> p_{res}$. This intuition is corroborated by our test accuracies obtained by SoTA models showing that, on average, $\boldsymbol{p_{max} > 0.94}$ for every dataset.

Table 4.4: Random sample text snippets from the IMDB dataset, two labelled positive and two negative. Most samples from the dataset are fairly clearly positive or negative.

Sentiment	**Review Excerpt**
Negative	Spend your time any other way, even housework is better than this movie... AVOID THIS MOVIE. It isn't funny, isn't cute...
Negative	When a comedy movie boasts its marvelous soundtrack on the back cover you know you're not dealing with a top notch movie... Don't waste your time even renting this one.
Positive	Hello, I was alanrickmaniac. I'm a Still Crazy-holic.... Then I wanted the DVD, because the tape showed first signs of decay after a few weeks... It contains some of the best actors possible.
Positive	Director Sidney Lumet has made some masterpieces, like Network, Dog Day Afternoon or Serpico... We need more movies like this.

(a) Fashion-MNIST

(b) CIFAR-10

(c) MNIST

(d) Clothing1M

Figure 4.2: Sample images from MNIST, Fashion-MNIST, CIFAR-10, and Clothing1M datasets. The Clothing1M dataset is taken from (J. Li et al., 2020).

This strongly implies that our inequality holds. We reason, therefore, that, except Patrini et al. (2017) at 60% noise level, all the experiments on these datasets are highly likely to be class-preserving. For reference, examples of images from these datasets can be found in Figure 4.2.

CIFAR100 In (Patrini et al., 2017), the authors introduce asymmetric noise to the CIFAR100 dataset by implementing circular noise within each of the 20 'superclasses.' The CIFAR100 dataset is organised into groups of 5 classes, termed as superclasses. For instance, the 'Aquatic Mammals' superclass comprises Beaver, Dolphin, Otter, Seal, and Whale (Patrini et al., 2017). Within each superclass, labels are cyclically permuted (e.g., Beaver $\mapsto$ Otter $\mapsto$ Dolphin, and so forth) with a probability of η. This label noise is also used in the following papers Z. Zhang and Sabuncu (2018), Ma et al. (2020), L. Feng et al. (2021), Y. Wang et al. (2019), Reed et al. (2014), Patrini et al. (2017), Englesson and Azizpour (2021b), T. Zhou et al. (2020), S. Liu et al. (2020). Since the noise is applied to each of these superclasses independently, the label noise is class-preserving only if the relation from Lemma 4.3.5 holds. The highest noise rate used in most papers is 40%, meaning we require $p_{max} \geq 3p_{res}$ to ensure this is class-preserving.

A different type of label noise for CIFAR100 is used in (X. Li et al., 2021). They use circular noise but on all the classes rather than independently within the superclasses. This was introduced first in (Han et al., 2018). The highest noise rate used is 45%, meaning that this label noise is class-preserving if $p_{max} \geq 5.5p_{res}$.

As previously discussed, CIFAR100 is a curated image dataset consisting primarily of clear images of a single subject (See Figure 4.2). As a result, the conditional class distributions are highly peaked at the dominant class. SoTA model test accuracies (Table 4.3) show that the average class distribution satisfies $p_{max} \geq 0.94$. Consequently, we argue that this CIFAR100 asymmetric label noise is well approximated as class-preserving. In conclusion, all experiments in Table 4.1 (symmetric noise) and all but three experiments in Table 4.2 (asymmetric noise) satisfy the conditions to be class-preserving which we derived in Section 4.3.

4.6 Conclusion

Summary In this chapter, we aimed to demonstrate that the majority of research on label noise focuses exclusively on class-preserving noise. We provided theoretical results indicating that common noise types remain class-preserving when the class distribution is sufficiently peaked, and the noise rate stays below a certain threshold value. We then showed that standard datasets typically exhibit heavily peaked class

distributions, leading us to conclude that standard noise models do not alter the dominant class in these settings—they are class-preserving. The discovery that most noise models studied in the robust loss literature are class-preserving is significant for two main reasons:

1) Class-Preserving Noise is a Broad Noise Type Demonstrating that most noise models studied in the robust loss literature are class-preserving reveals that class-preserving label noise constitutes a large family of label noise. Although class-preserving label noise may initially appear to be a restrictive assumption, the findings in this chapter suggest otherwise. This validates our use of the class-preserving assumption and bolsters the relevance of any theoretical result proven about this noise type in future studies.

2) Sheds Light on Reasons for Differences in Loss Robustness Conventional wisdom would suggest that a lack of robustness of a loss function such as cross-entropy is caused by the distortion of the risk caused by the introduction of label noise. This motivates and explains the efficacy of correction-based loss functions, which correct the (noisy) risk to account for this distortion. However, when label noise is class-preserving, a Bayes-optimal classifier for the noisy distribution will be Bayes-optimal for the clean distribution. This suggests that, in these settings, one should be able to employ any Fisher consistent loss function during training and ensure, *given enough data*, generalisation to the clean data distribution without employing a loss correction. This chapter has established that most studied noise *is* class-preserving. This suggests that the issue of poor robustness arises not from a lack of theoretical guarantees but insufficient data. This idea is further elaborated on in Section 2.4 and motivates our research in Chapters 5, 6 where we explore principled ways to avoid overfitting.

Chapter 5

Risk Bounding

5.1 Introduction

As discussed in Chapter 2, training neural network classifiers on datasets corrupted by label noise poses a risk of overfitting to the noisy labels. To address this issue, researchers have explored alternative loss functions that aim to be more robust. However, many of these alternatives, including correction-based losses, are still susceptible to overfitting or underfitting. Therefore, merely correcting the loss is insufficient to ensure robustness. While overfitting can be mitigated by ad hoc regularisation techniques, a principled, theoretically motivated approach is lacking. This chapter explores this question, providing such an approach.

$$\underbrace{\frac{1}{N}\sum_{i=1}^{N} L(\boldsymbol{q}(x_i), y_i)}_{\text{CE}} \xrightarrow{\text{forward-correct}} \underbrace{\frac{1}{N}\sum_{i=1}^{N} L(\widehat{T}(\boldsymbol{q}(x_i)), y_i)}_{\text{FCE}} \xrightarrow{\text{noise-bound}} \underbrace{\left\| B(\eta, c) - \frac{1}{N}\sum_{i=1}^{N} L(\widehat{T}\boldsymbol{q}(x_i), y_i) \right\|_1}_{\text{FCE+B}} \tag{5.1}$$

Figure 5.1: CE, FCE and, our proposed FCE+B loss functions. The forward-correction ensures consistency while the application of a bound (FCE+B) mitigates overfitting.

Contributions This work comprises two main contributions: The first, a more minor contribution, is to **generalise forward corrections** to include non-linear models, demonstrating that some popular heuristic robust loss functions (such as GCE and SCE) are, in fact, correction loss functions in disguise. The second contribution, which constitutes the majority of this chapter, shows how overfitting can be avoided by ensuring the training loss remains above a certain threshold. We refer to augmenting a loss function in

this manner as a **'bounded-loss'**. The crucial insight of this work is that when labels are noisy, no model can achieve a loss below the average entropy of the noisy conditional class distribution. Under a separability assumption, the noisy entropy depends only on the noise model and can thus be crudely estimated when only the average noise rate is known. This estimate, called the 'noise-bound,' is chosen as our threshold for the bounded loss. We demonstrate empirically that this significantly enhances robustness across various settings.

5.2 Generalised Forward-Corrections

5.2.1 Robust Loss Functions

Label noise robust loss functions can be categorised into two broad sets: regularisation-based robust losses and correction-based losses.

Regularisation-Based Losses A popular approach to tackling label noise by selecting losses less prone to fit the entire training set than the standard cross-entropy (CE). An archetypal example of such a loss is a mean absolute error (MAE) ($L_{MAE}(\boldsymbol{q}, y = k) = 1 - q_k$). MAE will typically ignore the harder-to-fit samples; on noisy datasets, this often corresponds to those with corrupted labels. The downside is that MAE dramatically underfits on datasets with many classes (Ma et al., 2020). Alternative losses mitigate this underfitting by interpolating between CE and MAE to avoid both of their pitfalls. Two well-known examples are the Generalised Cross-Entropy (GCE) and Symmetric Cross-Entropy (SCE) defined $L_{GCE}(\boldsymbol{q}, y = k) := \frac{1-q_k^a}{a}$ and $L_{SCE}(\boldsymbol{q}, y = k) = -log(q_k) + A(1 - q_k)$ respectively (Y. Wang et al., 2019; Z. Zhang & Sabuncu, 2018). By varying the parameters a, A, we can alter the losses' behaviour from being more like CE to MAE.

Correction-based Losses Correction-based loss functions arise as an alternative, motivated by the observation that the empirical risk ceases to be an effective proxy for the generalised clean risk (Stempfel & Ralaivola, 2009) under label noise. By altering the loss through incorporating the noise model, one may ensure the corrective property that

$$\arg\min_{\boldsymbol{q}} R^{\eta}_{L_F}(\boldsymbol{q}) = \arg\min_{\boldsymbol{q}} R_L(\boldsymbol{q}).$$

Thus ensuring that minimising the noisy generalised risk aligns with minimising the generalised clean risk. A popular and effective method is the *forward-correction* (Patrini et al., 2017). Given a *base loss* L, the forward-correction of L is defined

$$L_F(\boldsymbol{q}, k) \coloneqq L(\widehat{T}\boldsymbol{q}, k), \tag{5.2}$$

where $\widehat{T}$ is a column stochastic matrix approximating the true transition matrix $T \coloneqq p(\widetilde{y} \mid y)$ for class-conditional label noise. Conversely, a loss function L is a *forward-corrected loss* if it is the forward-correction of another loss function.

5.2.2 Non-Linear Noise Models

In the class-conditional label noise framework (Angluin & Laird, 1988), a label noise model is defined by a column stochastic matrix T, which represents the transition probabilities $p(\widetilde{y} \mid y)$. This traditional formulation assumes that the noisy label depends on an unobserved clean label, which might not reflect real-world scenarios where the noisy annotator never sees the clean label.

An equivalent conceptualisation of class-conditional label noise considers the transition matrix merely as a tool to link noisy and clean class distributions. For example, if the true conditional class distribution at x is $\boldsymbol{p}(y \mid x)$, then according to a class-conditional label noise model, the conditional class distribution of the noisy labeller is given by $T\boldsymbol{p}(y \mid x)$. This approach does not presume dependence on a specific true label, instead describing how noisy and clean label distributions are related.

However, this perspective raises questions about the necessity of assuming a linear relationship between these distributions. It is plausible that a labeller might make fewer errors on instances clearly representative of their class and many more errors when the class distribution $\boldsymbol{p}(y \mid x)$ is more evenly distributed. This scenario suggests a non-linear noise model:

$$\widetilde{\boldsymbol{p}}(y \mid x) \coloneqq f(\boldsymbol{p}(y \mid x)),$$

where $f : \Delta \to \Delta$ is some (possibly non-linear) transformation on this simplex - which, for simplicity, we will limit to being injective.

It is straightforward to generalise the forward-correction to allow for $\widehat{T}$ in Equation 5.2 being a non-linear transformation f.

Definition 5.2.1 (Generalised Forward-Correction)**.** *Let L_f be a loss function and $f : \Delta \to \Delta$ be an injective function. We say L_f is a 'generalised forward-corrected loss' if there exists a loss function L such that for all $\boldsymbol{q} \in \Delta$, $k \in \{1, 2, \ldots, c\}$*

$$L_f(\boldsymbol{q}, k) = L(f(\boldsymbol{q}), k).$$

*We refer to L as the **base loss**. f can be thought of as a label noise model.*

The forward-correction is trivially an example of a generalised forward-correction loss obtained by setting $f := \widehat{T}$. We now demonstrate that the GCE and SCE losses previously discussed are generalised forward-correction losses, deriving expressions for the underlying (non-linear) noise models f. This derivation relies on the assumption that the base losses which generate these loss functions are proper - i.e. we decompose both SCE and GCE into a proper loss corrected by a non-linear noise model. This decomposition is unique.

Lemma 5.2.2. *The GCE, SCE and forward-corrected CE (denoted FCE) loss functions can be formulated as generalised forward-corrected losses with a proper base loss. The noise models $f_{GCE}, f_{SCE}, f_{FCE}$ satisfy*

$$\begin{aligned}
(f_{GCE}^{-1}(\boldsymbol{p}))_i &= \frac{p_i^{\frac{1}{1-a}}}{\sum_{i=1}^{c} p_i^{\frac{1}{1-a}}}, \\
(f_{SCE}^{-1}(\boldsymbol{p}))_i &= \frac{p_i}{\lambda - A p_i}, \\
f_{FCE}(\boldsymbol{p}) &= \widehat{T}\boldsymbol{p},
\end{aligned}$$

where $\widehat{T}$ is the invertible stochastic matrix used to define FCE, and λ is a constant selected to ensure the correct normalisation.

Plots of the noise models; f_{GCE}, f_{SCE} are given in Figure 5.2. Lemma 5.2.2 demonstrates that GCE and SCE can be conceptualised as non-linear forward-corrected losses; the noise model is represented by the function f. We stress that these are by no means the only robust losses which adhere to Definition 5.2.1. However, they provide valuable examples when empirically demonstrating the results of Section 5.4.

The generalisation established by Definition 5.2.1 offers three advantages. i) It enhances our understanding of losses like GCE, demonstrating that they implicitly incorporate a noise model. ii) Partially unifies correction losses with other robust loss functions. iii) Ensures that theoretical results derived for generalised forward-correction losses are widely applicable, encompassing traditional, linear correction losses and many other robust loss functions.

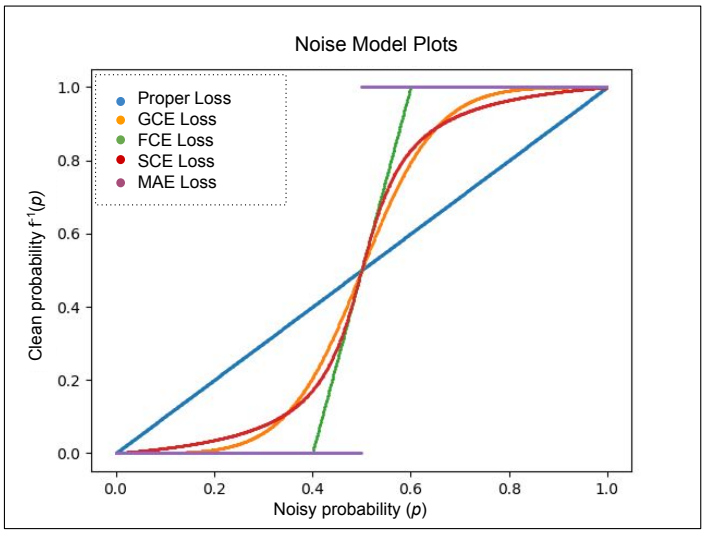


Figure 5.2: Plot of $f^{-1}(p)$ for SCE ($A = 8$), GCE ($a = 0.7$), FCE ($\eta = 0.4$), CE and MAE in the binary case. We have the true probability p on the x-axis and the choice of q, which minimises the expected loss on the y-axis.

5.3 Loss Bounding

Our analysis in this section reveals that merely correcting for the noise model (as in Definition 5.2.1) is inadequate for achieving robustness. We must also adjust our loss function to incorporate a lower bound to account for the randomness introduced by label noise.

Key Observation When a data distribution contains label noise, there is a lower bound on the optimal noisy risk a model can achieve. An analogy to this is that no forecaster can predict the outcome of a biased coin flip 100% of the time - e.g. if a coin comes up heads 60% of the time we cannot expect a forecaster to predict more accurately than 60% over a large number of flips. Similarly, even an optimal model, which minimises the noisy risk, will *still* incur a non-zero loss on a randomly sampled noisy dataset.

5.3.1 Overfitting to Label Noise

If one trains a classifier on a noisy dataset using the cross-entropy loss function, the classifier learns to fit all labels in the training set - including those corrupted by label noise (Arpit et al., 2017) - damaging generality. Ideally, when training a model on a noisy dataset, we wish to fit the clean labels without overfitting the noisy ones. A significant obstacle to achieving this desire is the difficulty in determining whether a specific label is clean or corrupted. However, while it is difficult to determine whether a model has overfit to a *specific* label, it is often apparent when a model has overfit to a dataset. For example, if a dataset is known to contain label noise, obtaining a training loss of zero heavily implies that overfitting has occurred.

Forward-Corrections Consequently, when learning a classifier on noisy labels, targeting a zero training loss is inappropriate. Utilising the forward-correction partly addresses this issue. The forward-correction works by noising our model predictions $\boldsymbol{q} \mapsto \widehat{T}\boldsymbol{q}$ before applying the loss. This guarantees that a zero loss is no longer possible since our noised model never predicts any label with complete confidence. Despite possessing this desirable property, the forward-correction does not go far enough in that the lower bound it imposes is still too high to prevent overfitting. An illustrative example for a dataset polluted by 40% symmetric label noise is presented in Table 5.1. The table gives the lowest attainable training loss for a model trained on this noisy dataset versus the 'optimal training loss', i.e., the loss that would be obtained by an optimal model, possessing complete knowledge of how the dataset was generated, but which has not observed the dataset labels. Obtaining a training loss lower than this optimal value would suggest overfitting has occurred. While FCE imposes a bound (unlike CE), this bound is still too low.

5.3.2 Bounded Loss

We propose, therefore, that the principled way to handle label noise is to limit the minimum allowable risk on the training set. Specifically, we define a lower bound 'B' and train - preventing the training loss from going below this value. Explicitly, we augment our loss as follows:

Loss Function	Lowest Attainable Training Loss	Optimal Training Loss
CE	0	0.673
FCE	0.511	0.673
FCE+B	0.673	0.673

Table 5.1: Bounds imposed on the attainable training loss by different cross-entropy variants versus the *optimal* training loss: This table compares the minimum training loss achievable by a model on a large noisy dataset using different loss functions, distinguishing between scenarios where the model is allowed or not allowed to observe the dataset labels. We assume a separable binary-label dataset corrupted by 40% symmetric label noise. When the dataset labels are not observable, the loss on the dataset is minimised (in expectation) by an optimal model - a minimiser of the (noisy) generalised risk. This model will obtain a loss of 0.673 on this dataset with high probability. In contrast, using CE or FCE loss functions can result in training losses below this optimal figure, making overfitting likely. While FCE introduces an inherent loss bound, improving robustness, it may still permit overfitting. Our bounded variant, FCE+B, is designed to better mitigate overfitting.

Definition 5.3.1 (Bounded Loss)**.** *Let L be a loss function. Let $\mathcal{D}$ be a batch of N data-label pairs (x_i, y_i). Given a lower bound, $B \in \mathbb{R}$, we define the B-**bounded** loss $L_{\underline{B}}$ obtained from L as follows:*

$$\boxed{L_{\underline{B}}(\boldsymbol{q}(x), \mathcal{D}) := \left|\left| B - \tfrac{1}{N}\sum_{i=1}^{N} L(\boldsymbol{q}(x_i), y_i) \right|\right|_1} \tag{5.3}$$

When the average loss on a batch of samples is above our bound B, training proceeds as usual; however, if the training loss on a batch dips below B, the learning rate effectively becomes negative, resulting in 'untraining' which proceeds until the average loss is back above B. Bounding the loss in this way has been previously explored by Ishida et al. (2020). We extend this work by grounding loss bounding within the context of loss corrections and providing a theoretically justified method for selecting the loss bound.

5.4 Risk Bounds

In the last section, we remarked that despite theoretical motivation, correction losses are still prone to overfitting. We proposed training against a lower bound to prevent this, noting that the minimal achievable generalised noisy risk is non-zero. This section explicitly derives lower bounds on the generalised noisy L-risk for generalised forward-corrected losses. We conclude by presenting a formula for choosing a bound B to use in Definition 5.3.1. We call this the *noise-bound*.

Assumptions Throughout this section, we suppose that all loss functions are generalised forward-corrected losses, which we denote L_f. Furthermore, we suppose that their base loss functions, L, are proper. We also assume that the loss function has no inherent bias toward any particular class, i.e. the loss is unaffected by a random permutation of the label set. Examples of such losses include CE, MSE, FCE, GCE, SCE and many others.

5.4.1 Entropy As Lower Bound

In Lemma 5.4.2 we establish a general lower bound on noisy risk in terms of the average entropy of the noisy label distribution. Precisely stating Lemma 5.4.2 requires us to define the 'entropy function' of a proper loss.

Definition 5.4.1 (Entropy Function)**.** *Given a proper loss function L, define its entropy function (Ovcharov, 2018) as the expected loss incurred when the forecast equals the true distribution over classes: $\mathcal{H} : \Delta \to \mathbb{R}$ by:*

$$\mathcal{H}(\boldsymbol{p}) \coloneqq H_L(\boldsymbol{p}, \boldsymbol{p}) = \boldsymbol{p}^T \boldsymbol{L}(\boldsymbol{p}),$$

where $\boldsymbol{p}$ is a probability distribution over the classes and H_L denotes the expected loss.

The entropy function for a (strictly) proper loss function is (strictly) concave and, by the definition of properness, satisfies $\mathcal{H}(\boldsymbol{p}) \leq H(\boldsymbol{p}, \boldsymbol{q})$ for all $\boldsymbol{p}, \boldsymbol{q} \in \Delta$. This leads immediately to the following lemma.

Lemma 5.4.2. *Let L_f be a generalised forward-corrected loss whose 'base-loss' L is strictly proper (Recall the definition of 'base-loss' from Definition 5.2.1). The noisy risk of any probability estimator $\boldsymbol{q}$ is lower bounded:*

$$R^{\eta}_{L_f}(\boldsymbol{q}) \geq \mathbb{E}_{x \sim p(x)}[\mathcal{H}(\widetilde{\boldsymbol{p}}(\tilde{y} \mid x))], \tag{5.4}$$

where $\mathcal{H}$ is the entropy function of the base-loss. This bound is tight when f equals the true noise model. Equality is attained by setting $\boldsymbol{q}(x) = f^{-1}(\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x))$.[1]

Lemma 5.4.2 establishes that when using a generalised forward-corrected loss (with proper base loss), the average entropy of the noisy distribution provides a lower bound on the noisy risk. Ideally, one would calculate the entropy in Equation 5.4 to use as the lower bound 'B' in Equation 5.3. However, due to limited knowledge of the data distribution and noise model, it is often impractical to calculate this precisely, necessitating simplifying assumptions for approximation.

Separability The key simplifying assumption we employ is assuming that the clean label distribution is (approximately) separable. While an idealised assumption, it is a reasonable approximation for many real-world image classification tasks where clear images of a single subject dominate the dataset. This assumption is less suitable in domains with high inherent randomness, such as medical diagnostics. In the following, we leverage this assumption to construct a few bounds.

5.4.2 Estimating The Entropy

Given a datapoint x, the entropy of the noisy class distribution $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x)$ will depend both on the noise rate and the *type* of label noise. For example, symmetric label noise (where noise occurs uniformly between classes) will result in a different entropy than pairwise label noise (where noise occurs between pairs of classes), even given the same noise rate. The following Lemma gives a range on the possible entropies of $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x)$ given that the noise rate at x is equal to $\eta(x)$. For brevity, we introduce the following notation:

$$\boldsymbol{u}_{\text{pair}}(\eta, c) \coloneqq (1-\eta, \eta, 0, \ldots, 0) \tag{5.5}$$

$$\boldsymbol{u}_{\text{sym}}(\eta, c) \coloneqq \left(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\right) \tag{5.6}$$

Lemma 5.4.3. *Let L_f be a generalised forward-corrected loss function whose base-loss L has entropy function $\mathcal{H}$. Suppose that label noise is applied to a separable data-label distribution and let $x \sim p(x)$. Given that the noise rate at x is $\eta(x)$, the entropy of the noisy label distribution at x, $\mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x))$ must lie in the following interval:*

$$\left[\mathcal{H}(\boldsymbol{u}_{pair}(\eta(x), c)), \mathcal{H}(\boldsymbol{u}_{sym}(\eta(x), c))\right].$$

[1] Note that if f is the true noise model then $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x) \in f(\Delta)$ and the inverse is unique by injectivity.

In particular, for a fixed noise rate $\eta(x)$, the highest entropy occurs under symmetric label noise at x, while the lowest entropy is observed with pairwise label noise.

Corollary 5.4.4. *Given an average noise rate $\eta := \mathbb{E}_{x\sim p(x)}[\eta(x)]$, the greatest possible value of $\mathbb{E}_{x\sim p(x)}[\mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y}\mid x)]$ occurs when $\eta(x)$ is constant:*

$$\sup_{p(\widetilde{y}|x,y)} \left(\mathbb{E}_{x\sim p(x)}[\mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y}\mid x))]\right) = \mathcal{H}\left(\boldsymbol{u}_{sym}(\eta, c)\right),$$

where the supremum is taken over all noise models such that $\mathbb{E}_{x\sim p(x)}[\eta(x)] = \eta$.

Worst-Case Entropy Corollary 5.4.4 establishes a worst-case entropy given a specified average noise rate η. Simply put, the Corollary tells us *'given that the average noise rate does not exceed η, the noise model with the highest entropy is uniform symmetric label noise'*.

5.4.3 Main Proposal: Noise-Bounded Loss

Discussion The goal of this section is to derive lower bounds on the noisy risk, which can be used as 'B' in our B-bounded loss (Definition 5.3.1). Ideally, with precise knowledge of the noise model, the lower bound would be set equal to the average entropy of the noisy label distribution. However, the label noise model is typically unknown, and we might only have access to an approximate noise rate.

What bound do we choose when we only know the noise rate? Corollary 5.4.4 establishes that given a known noise rate η, a 'worst-case' entropy occurs when the label noise is symmetric and uniform. This means that if we set our bound 'B' under the assumption of symmetric-uniform label noise at rate η, B can never be lower than the true noisy entropy - making overfitting unlikely. We call this the 'noise-bound'.

Definition 5.4.5 (Noise-Bound)**.** *Let L_f be a generalised forward-corrected loss whose base loss L has entropy function $\mathcal{H}$. Using the notation $\boldsymbol{u}_{sym}(\eta, c)$ from Equation 5.6, we define the **noise-bound** as:*

$$\boxed{B(\eta, c) := \mathcal{H}(\boldsymbol{u}_{sym}(\eta, c)) = \boldsymbol{u}_{sym}(\eta, c) \cdot \boldsymbol{L}(\boldsymbol{u}_{sym}(\eta, c)).} \tag{5.7}$$

Examples For CE and FCE the noise-bound corresponds to the Shannon Entropy of the distribution $\boldsymbol{u}_{\text{sym}}(\eta, c) = (1-\eta, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1})$. For SCE and GCE we remark that

$$\begin{aligned} B(\eta, c) &= \boldsymbol{u}_{\text{sym}}(\eta, c) \cdot \boldsymbol{L}(\boldsymbol{u}_{\text{sym}}(\eta, c)) \\ &= \boldsymbol{u}_{\text{sym}}(\eta, c) \cdot \boldsymbol{L}_f(f^{-1}(\boldsymbol{u}_{\text{sym}}(\eta, c))) \end{aligned}$$

allowing us to compute $B(\eta, c)$ by substituting expressions for f^{-1} derived in Lemma 5.2.2. The bound for GCE is

$$\begin{aligned} B_{GCE}(\eta, c) := \frac{(1-\eta)}{a}\left(1 - \left(\frac{(1-\eta)^{\frac{1}{1-a}}}{(1-\eta)^{\frac{1}{1-a}} + (c-1)(\frac{\eta}{c-1})^{\frac{1}{1-a}}}\right)^a\right) + \\ \frac{\eta}{a}\left(1 - \left(\frac{\frac{\eta}{c-1}^{\frac{1}{1-a}}}{(1-\eta)^{\frac{1}{1-a}} + (c-1)(\frac{\eta}{c-1})^{\frac{1}{1-a}}}\right)^a\right). \end{aligned}$$

The noise-bound for SCE is

$$\begin{aligned} B_{SCE}(\eta, c) := (1-\eta)\left(-log\left(\frac{1-\eta}{\lambda - A(1-\eta)}\right) + A\left(1 - \frac{1-\eta}{\lambda - A(1-\eta)}\right)\right) \\ +\eta\left(-log\left(\frac{\eta}{\lambda(c-1) - A\eta}\right) + A\left(1 - \frac{\eta}{\lambda(c-1) - A\eta}\right)\right). \end{aligned}$$

Recollect that λ is chosen so that the resulting distribution normalises: $\frac{1-\eta}{\lambda - A(1-\eta)} + \frac{\eta(c-1)}{\lambda(c-1) - A\eta} = 1$ and may be computed numerically or by solving the resulting quadratic.

This leads us to the main proposal of this chapter. When our dataset has label noise, we propose using the bounded loss (Equation 5.3) with B set to $B(\eta, c)$, the 'noise-bound' from Equation 5.7. We call this the **noise-bounded loss**:

Definition 5.4.6 (Noise-Bounded Loss)**.** *Let L be a loss function. Let $\mathcal{D}$ be a batch of N data-label pairs (x_i, y_i). Given a noise rate η, we define the **noise-bounded** loss $L_{B(\eta,c)}$ obtained from L as follows:*

$$\boxed{L_{B(\eta,c)}(\boldsymbol{q}(x), \mathcal{D}) := \left|\left|B(\eta, c) - \frac{1}{N}\sum_{i=1}^{N} L(\boldsymbol{q}(x_i), y_i)\right|\right|_1} \tag{5.8}$$

where $B(\eta, c)$ is as given in Equation 5.7.

Our proposed noise-bounding method is summarised algorithmically in Algorithm 1.

Example FCE: The noise-bounded variant of FCE (which we denote FCE+B) is given in Equation 5.1.

Bound Optimality By construction, the noise-bound is only equal to the average entropy of the noisy label distribution if the label noise is symmetric and uniform. In all other cases, the noise-bound will be higher than strictly necessary. Ideally, we would like the gap between the noise-bound and the true noisy entropy to be small in a typical setting: If the gap is large, the noise-bounded loss will cease training long before overfitting occurs and possibly when there is still signal to be learned. A small gap occurs when all distributions of the form $(1-\eta, \eta_2, \ldots, \eta_c)$ (where $\eta \coloneqq \sum_i \eta_i$) have roughly the same entropy - i.e. the entropy is 'insensitive' to the *structure* of the noise model, depending mainly on the noise *rate*. The level of insensitivity depends on the entropy function itself. Shannon entropy is relatively sensitive to the structure of the noise model. In contrast, losses like GCE and SCE induce insensitive entropy functions. This topic is discussed further in Appendix C.2.1.

Empirical and Generalised Risk Our analysis in Section 5.4.1 demonstrates that an estimator cannot achieve a noisy risk below the mean noisy label entropy. However, in a small finite dataset of i.i.d. samples, it is possible for an estimator to achieve a noisy *empirical* risk that is lower than the noisy entropy, and this can occur with non-zero probability, even without access to the actual dataset labels. However, as the size of the dataset increases, the likelihood of an estimator achieving a loss significantly lower than the noisy entropy rapidly diminishes. According to the Central Limit Theorem, the probability of obtaining a loss more than δ below the noisy label entropy diminishes at the rate of $O\left(\frac{1}{\sqrt{N}}\right)$, where N is the dataset size. Practically, this means obtaining a training loss below the noise-bound is *almost impossible* unless one has overfit to the noisy labels.

5.5 Experiments

5.5.1 Loss Functions

In this section, we empirically investigate the effectiveness of the noise-bounded loss (Equation 5.8) for improving robustness to label noise. We consider several loss functions: CE, SCE, forward-corrected CE (FCE), and GCE. Additionally, we explore a variant of CE that includes a prior on the model probabilities (CEP). Our experiments all follow a similar structure. We use a dataset containing intrinsic or synthetic label noise in the training set. We train neural network models using each loss on this noisy

Algorithm 1 Training with Noise-Bounded Loss

Input: Noisy dataset $\mathcal{D} = \{(x_i, \widetilde{y}_i)\}_{i=1}^{N}$, estimated noise rate η, number of classes c, epochs T
Output: Trained model parameters Θ
function COMPUTENOISEBOUND(η, c)
    $\boldsymbol{u}_{\text{sym}}(\eta, c) \leftarrow \left(1-\eta, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\right)$
    return $\boldsymbol{u}_{\text{sym}}(\eta, c) \cdot L(\boldsymbol{u}_{\text{sym}}(\eta, c))$
end function
procedure TRAINMODEL($\mathcal{D}$, η, c, T)
    $B(\eta, c) \leftarrow$ COMPUTENOISEBOUND(η, c)
    for $epoch = 1$ to T **do**
        for each (x_i, y_i) in $\mathcal{D}$ **do**
            $\boldsymbol{q}(x_i) \leftarrow \text{ModelPrediction}(x_i; \Theta)$
            $loss \leftarrow \left| B(\eta, c) - \frac{1}{N} \sum_{j=1}^{N} L(\boldsymbol{q}(x_j), y_j) \right|$
            $\Theta \leftarrow \text{UpdateModel}(\Theta, loss)$
        end for
    end for
    return Θ
end procedure

training set and evaluate their performance on a clean test set. We compare results from models trained without noise-bounds to those trained with noise-bounds, denoted by a '+B' suffix in the loss name (e.g., CE+B indicates the noise-bounded cross-entropy loss).

Baseline Loss Functions Our results are benchmarked against other standard robust loss functions, including mean squared error (MSE) (Janocha & Czarnecki, 2016), mean absolute error (MAE), NCE-MAE (Ma et al., 2020), ELR (S. Liu et al., 2020), Curriculum loss (CL) (T. Zhou et al., 2020), Bootstrapping loss (Boot.) (Reed et al., 2014), Spherical loss (Spher.), Mix-up (H. Zhang et al., 2017), and a version of GCE that incorporates the additional tricks outlined by Z. Zhang and Sabuncu (2018). To differentiate this version of GCE from our simplified GCE, we refer to it as 'Truncated loss' (Trunc.) due to its use of truncation.

5.5.2 Datasets

We evaluate each loss on various datasets with different label noise types. We consider versions of EMNIST, FashionMNIST, CIFAR10, CIFAR100 corrupted by symmetric label noise at rates of 0.2 and 0.4 and MNIST with rates of 0.4 and 0.6. Additionally, we explore more sophisticated noise types. In the case of 'Asym-CIFAR100,' we introduce asymmetric noise by randomly transitioning labels within the 20 superclasses of CIFAR100. For example, within the superclass 'fish' (comprised of aquarium-fish, flatfish, ray, shark, trout), we change training labels to other members of the set with a probability of $\eta \in \{0.2, 0.4\}$ (e.g., flatfish $\rightarrow$ trout). For 'Non-Uniform EMNIST,' we investigate the impact of using non-uniform noise. We train a linear classifier on EMNIST and, with a probability of 0.6, modify the label of a data point in our training set to match the output of this classifier. Since the classifier's performance varies across dataspace, this creates label noise with an x-dependence. Further experiments, including on the TinyImageNet and Animals-10N datasets, which contain real, intrinsic open-set noise, and precise experimental details are given in Appendix C.3.

Hyperparameters For the Animals and TinyImageNet experiments, we use a ResNet-34 to parameterise our model. For the other datasets, we use a ResNet-18. For each experiment, the number of epochs is kept consistent across losses. The bounds we employ in each experiment are obtained by substituting the relevant number of classes c and the noise rate η into Equation 5.7. An exception is the case of Non-uniform EMNIST, where we use a class number of $c = 2$ to reflect that the label is a mixture of the clean and classifier labels.

5.5.3 Results

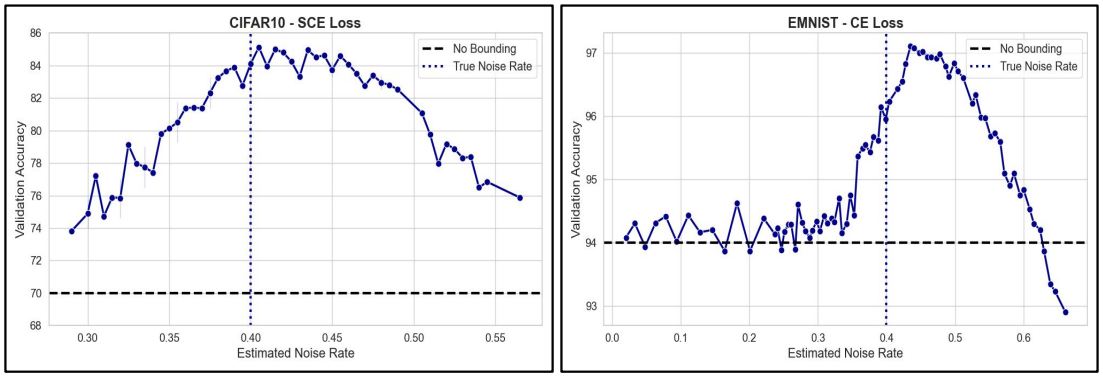


Figure 5.3: **Performance as a function of the estimated noise rate used to compute noise-bound:** We plot the final (clean) validation accuracy of a model against the estimated noise rate used to compute the noise-bound (Eqn. 5.7) on the noisy CIFAR10/EMNIST datasets using SCE/CE losses respectively. **The noise-bound, as computed with the *true* noise rate is highlighted by the green dotted line**; both graphs show a bump with a peak near this line demonstrating that underestimating the noise rate causes overfitting while overestimating causes underfitting. Most crucially, the prominent 'bump' reinforces that robustness can be greatly improved by training using a well-selected bound.

The results of our experiments are presented in two tables. Table 5.2 includes the simpler datasets of MNIST, FashionMNIST, EMNIST, and CIFAR10, while Table 5.3 displays CIFAR100, Asym-CIFAR100, and Non-uniform-EMNIST. Each table follows a similar structure, with losses listed in rows and datasets in columns. The baselines are grouped together at the top. Our main losses are organised into pairs, such as CE, CE+B. The rows that use the noise-bound (e.g., GCE+B) are highlighted in blue to

enhance readability. If using our noise-bound leads to higher mean accuracy compared to training without the bound, this is indicated by a $\boxed{box}$. The best overall model for each dataset is highlighted in **bold**. In **82**% of cases, utilising the noise-bound improves performance relative to the unbounded loss variant.

Exceptions With few exceptions, our bound leads to improved performance compared to the standard, unbounded version of each loss. For the Asym-CIFAR100 and Non-Uniform-EMNIST datasets, our CE+B loss performs worse than regular CE. This outcome was expected since our derived bounds are optimal for symmetric noise and may be suboptimal for non-symmetric noise - this discrepancy is especially pronounced for losses based on Shannon-Entropy like CE. In contrast, the other generalised forward-corrected losses, as we had anticipated, exhibit greater resilience to the precise noise structure and consistently outperform the baseline across different types of noise.

Impact of Estimated Noise Rate Figure 5.3 shows how clean test accuracy varies with the estimated noise rate, $\widehat{\eta}$, for CIFAR10 (SCE loss) and EMNIST (CE loss) datasets, both corrupted by 40% symmetric noise. Models were trained using noise-bounds based on $\widehat{\eta}$ ($B(\widehat{\eta}, c = 10)$ in Equation 5.7), and performance was plotted against $\widehat{\eta}$. As $\widehat{\eta}$ increases, the bound restricts overfitting, enhancing test accuracy. Optimal performance occurs near the true noise rate at $\widehat{\eta} \approx 0.4$, marked by a vertical green dotted line. Beyond this point, performance declines as the model underfits. The prominent peak in performance near this green line empirically validates our theoretical approach. Slight overestimations of the noise rate marginally improve performance, likely originating from our simplifying assumption, which modelled the underlying distributions as separable.

5.6 Conclusion, Limitations and Further Work

In this work, we have looked at mitigating the impact of label noise in forward-corrected losses by training subject to a bound, motivated by our observation that label noise implies a minimum achievable risk.

Losses	MNIST		FashionMNIST		EMNIST				CIFAR10	
					0.2		0.4			
	0.4	0.6	0.2	0.4	Top 1	Top 5	Top 1	Top 5	0.2	0.4
MSE	93.3±0.47	85.8±0.95	84.8±0.22	80.6±0.84	82.9±0.29	98.1±0.04	80.2±0.19	97.1±0.07	78.7±1.51	56.4±0.11
MAE	97.9±0.08	96.4±0.08	83.2±0.10	82.2±0.37	49.8±2.83	52.2±0.10	50.4±1.14	51.4±0.96	88.6±1.34	78.9±5.95
NCE	97.8±0.06	96.0±0.25	87.7±0.26	86.3±0.14	84.5±0.25	97.9±0.05	82.6±0.81	96.7±0.03	**89.3**±0.40	**86.0**±0.81
MixUp	95.8±1.24	86.8±0.85	86.9±0.10	82.3±0.54	84.3±0.08	98.1±0.04	81.6±0.48	97.1±0.08	86.0±0.46	77.9±0.49
Spher.	95.0±0.41	88.1±0.82	87.2±0.04	84.1±0.75	84.6±0.12	98.3±0.05	83.2±0.29	98.1±0.58	86.6±0.01	72.1±0.80
Boot.	86.6±0.56	71.2±1.17	82.0±0.61	73.4±1.06	80.5±0.24	96.7±0.06	77.3±0.98	95.0±0.25	77.0±1.57	58.2±2.99
Trunc.	97.1±0.12	94.2±0.39	87.8±0.29	85.3±0.77	84.1±0.53	97.4±1.03	83.1±0.55	97.2±1.00	88.3±0.56	84.2±0.69
CL	82.7±0.57	67.5±1.83	81.2±0.34	73.1±0.66	79.6±0.17	96.4±0.05	75.1±0.67	94.2±0.24	76.0±2.16	59.4±4.20
ELR	**98.1**±0.04	**97.8**±0.07	85.3±0.23	83.4±0.02	81.8±0.26	97.5±0.21	76.6±0.10	96.5±0.11	88.1±0.82	85.7±0.06
FCE.	95.4±0.25	92.3±0.13	83.6±0.11	79.9±0.78	83.1±0.12	98.4±0.20	80.6±0.12	98.0±0.03	84.7±0.40	75.1±0.04
FCE+B	95.7±0.18	92.7±0.74	84.8±0.26	81.7±0.27	83.4±0.09	**98.5**±0.03	81.6±0.51	**98.1**±0.15	86.7±0.21	82.2±0.06
GCE	94.4±0.36	83.8±1.14	86.4±0.24	81.6±0.37	84.3±0.13	98.4±0.08	82.7±0.07	97.9±0.02	81.1±0.72	60.0±1.31
GCE+B	96.6±0.22	94.0±0.13	86.5±0.56	85.5±0.13	84.1±0.29	98.4±0.04	82.8±0.28	98.0±0.06	86.1±0.22	79.0±1.17
SCE	89.5±5.29	70.2±0.69	82.7±0.64	74.4±0.37	82.1±0.33	96.8±0.10	79.6±0.61	95.4±0.15	78.2±0.42	59.0±4.43
SCE+B	97.0±0.16	93.4±0.29	87.5±0.22	85.2±0.98	83.5±0.29	97.3±0.14	81.8±0.52	96.4±0.20	88.9±0.44	84.7±0.37
CE	80.8±2.31	67.3±0.80	80.9±1.11	72.1±2.16	79.9±0.28	96.4±0.08	75.6±0.20	94.2±0.24	76.9±1.22	59.9±2.15
CE+B	96.2±0.32	93.0±0.09	87.9±0.10	84.7±0.37	80.8±0.08	97.0±0.04	78.9±0.12	96.1±0.26	84.5±0.73	76.0±1.13
CEP	97.5±0.08	92.1±0.44	87.8±0.12	84.8±0.23	85.5±0.10	98.1±0.07	84.3±0.22	97.6±0.14	84.2±0.51	58.2±2.94
CEP+B	95.6±0.32	85.5±0.77	**88.1**±0.31	84.2±0.33	**85.8**±0.12	98.3±0.02	**84.8**±0.10	98.0±0.04	88.5±0.32	85.1±0.20

Table 5.2: Test accuracies obtained by using different losses on the noisy MNIST/ FashionMNIST/EMNIST/CIFAR10 datasets. Losses implementing the noise-bound shaded in blue. When using this bound provides benefit, the corresponding value is $\boxed{boxed}$. Overall top values in **bold**.

Summary We began by defining a family of loss functions we called 'generalised forward-corrected losses' since they contain correction losses as a strict subset. We showed how some popular existing robust losses can be formulated as generalised forward-corrected loss functions. We explained how label noise implies the existence of a lower bound on the achievable risk. We proposed training a model and preventing the training loss going below a given threshold - we called this a 'bounded loss'. We derived this lower bound for generalised forward-corrected losses, showing it is the average entropy of the noisy label distribution (with respect to the entropy function of the base loss). We showed that uniform symmetric label noise is a 'worst-case' noise, meaning that it has the highest entropy for a given noise rate η. When the label noise rate is known, but the noise model is otherwise unknown, we proposed using this worst-case entropy as a bound for our bounded loss. Finally, we empirically showed that training using the 'noise-bound' improves performance for different loss functions across various noisy settings.

Losses	CIFAR100				ASYM-CIFAR100				Non-Uniform-EMNIST	
	0.2		0.4		0.2		0.4		0.6	
	Top1	Top5	Top1	Top5	Top1	Top5	Top1	Top5	Top 1	Top 5
MSE	57.2$_{\pm 0.93}$	78.6$_{\pm 0.25}$	40.6$_{\pm 0.38}$	63.0$_{\pm 0.24}$	56.3$_{\pm 0.11}$	82.6$_{\pm 0.22}$	40.7$_{\pm 0.12}$	74.4$_{\pm 0.25}$	44.7$_{\pm 2.66}$	86.7$_{\pm 3.10}$
MAE	10.0$_{\pm 0.11}$	13.8$_{\pm 0.28}$	7.6$_{\pm 1.89}$	11.6$_{\pm 1.25}$	7.1$_{\pm 6.02}$	11.1$_{\pm 6.6}$	11.1$_{\pm 5.43}$	25.1$_{\pm 5.76}$	9.8$_{\pm 1.74}$	23.1$_{\pm 1.80}$
NCE	38.7$_{\pm 3.13}$	51.8$_{\pm 3.77}$	19.1$_{\pm 0.20}$	28.8$_{\pm 0.15}$	16.3$_{\pm 1.24}$	25.4$_{\pm 1.80}$	21.8$_{\pm 1.24}$	37.2$_{\pm 1.80}$	18.0$_{\pm 1.17}$	38.8$_{\pm 1.93}$
MixUp	59.6$_{\pm 0.31}$	81.5$_{\pm 0.39}$	51.3$_{\pm 8.63}$	75.8$_{\pm 8.09}$	61.2$_{\pm 0.88}$	86.0$_{\pm 1.12}$	47.2$_{\pm 0.60}$	81.3$_{\pm 0.23}$	**52.4**$_{\pm 0.80}$	**95.5**$_{\pm 0.08}$
Spher.	57.7$_{\pm 0.18}$	82.9$_{\pm 0.54}$	48.8$_{\pm 0.51}$	74.3$_{\pm 0.73}$	54.2$_{\pm 0.32}$	81.2$_{\pm 0.29}$	39.2$_{\pm 0.31}$	72.1$_{\pm 0.15}$	41.9$_{\pm 0.10}$	94.4$_{\pm 0.04}$
Boot.	54.0$_{\pm 0.37}$	76.4$_{\pm 0.39}$	37.7$_{\pm 0.89}$	60.9$_{\pm 1.52}$	56.0$_{\pm 0.34}$	83.8$_{\pm 0.03}$	43.2$_{\pm 0.35}$	78.3$_{\pm 0.20}$	49.1$_{\pm 0.29}$	95.3$_{\pm 0.42}$
Trunc.	58.1$_{\pm 0.36}$	82.7$_{\pm 0.37}$	50.9$_{\pm 1.17}$	77.2$_{\pm 0.59}$	56.3$_{\pm 0.62}$	82.3$_{\pm 0.61}$	45.2$_{\pm 0.81}$	75.6$_{\pm 0.29}$	23.7$_{\pm 0.98}$	40.1$_{\pm 1.24}$
CL	53.0$_{\pm 0.21}$	76.3$_{\pm 0.19}$	36.3$_{\pm 0.77}$	60.1$_{\pm 0.66}$	55.3$_{\pm 0.48}$	83.5$_{\pm 0.28}$	42.4$_{\pm 0.45}$	78.1$_{\pm 0.14}$	48.2$_{\pm 0.45}$	95.0$_{\pm 0.04}$
ELR	10.4$_{\pm 0.24}$	31.7$_{\pm 0.44}$	10.0$_{\pm 0.64}$	30.1$_{\pm 0.88}$	10.8$_{\pm 0.21}$	32.7$_{\pm 0.53}$	10.3$_{\pm 0.39}$	30.8$_{\pm 0.35}$	40.3$_{\pm 0.39}$	93.0$_{\pm 0.24}$
FCE	56.9$_{\pm 0.58}$	79.2$_{\pm 0.14}$	43.7$_{\pm 0.15}$	66.2$_{\pm 0.19}$	55.3$_{\pm 0.54}$	83.5$_{\pm 0.24}$	41.4$_{\pm 0.55}$	77.3$_{\pm 0.75}$	39.0$_{\pm 0.05}$	67.8$_{\pm 0.47}$
FCE+B	56.1$_{\pm 2.22}$	81.8$_{\pm 1.37}$	50.2$_{\pm 0.02}$	77.2$_{\pm 0.19}$	54.2$_{\pm 0.44}$	83.3$_{\pm 0.43}$	43.8$_{\pm 0.02}$	77.5$_{\pm 0.13}$	40.0$_{\pm 0.35}$	73.2$_{\pm 0.08}$
GCE	60.0$_{\pm 0.13}$	82.6$_{\pm 0.63}$	44.9$_{\pm 0.07}$	67.2$_{\pm 0.34}$	53.8$_{\pm 0.55}$	81.6$_{\pm 0.14}$	39.4$_{\pm 0.44}$	74.0$_{\pm 0.36}$	44.8$_{\pm 0.62}$	91.2$_{\pm 0.70}$
GCE+B	59.4$_{\pm 0.02}$	83.5$_{\pm 0.24}$	50.3$_{\pm 0.11}$	75.3$_{\pm 0.64}$	55.4$_{\pm 0.55}$	83.0$_{\pm 0.35}$	46.5$_{\pm 1.44}$	77.7$_{\pm 0.35}$	47.1$_{\pm 0.20}$	93.5$_{\pm 0.43}$
SCE	55.9$_{\pm 0.53}$	76.5$_{\pm 0.15}$	38.7$_{\pm 0.60}$	60.9$_{\pm 0.41}$	57.5$_{\pm 0.19}$	83.7$_{\pm 0.17}$	43.3$_{\pm 0.87}$	77.5$_{\pm 0.75}$	47.2$_{\pm 0.33}$	92.5$_{\pm 0.01}$
SCE+B	55.5$_{\pm 0.90}$	77.4$_{\pm 0.84}$	47.1$_{\pm 1.32}$	69.2$_{\pm 1.18}$	57.9$_{\pm 0.83}$	83.7$_{\pm 0.41}$	50.0$_{\pm 1.62}$	80.4$_{\pm 0.65}$	47.9$_{\pm 0.80}$	93.8$_{\pm 0.05}$
CE	52.3$_{\pm 1.35}$	75.6$_{\pm 0.93}$	35.3$_{\pm 1.14}$	59.3$_{\pm 0.81}$	54.9$_{\pm 0.12}$	83.3$_{\pm 0.25}$	42.4$_{\pm 0.16}$	78.9$_{\pm 0.56}$	48.6$_{\pm 0.11}$	95.3$_{\pm 0.10}$
CE+B	50.9$_{\pm 1.01}$	76.5$_{\pm 0.86}$	39.9$_{\pm 1.02}$	65.8$_{\pm 1.19}$	52.9$_{\pm 1.86}$	83.2$_{\pm 0.88}$	34.7$_{\pm 2.51}$	73.4$_{\pm 1.50}$	45.5$_{\pm 5.11}$	93.0$_{\pm 0.16}$
CEP	58.8$_{\pm 0.87}$	78.6$_{\pm 0.38}$	43.5$_{\pm 0.24}$	65.1$_{\pm 1.27}$	59.4$_{\pm 0.08}$	82.2$_{\pm 0.03}$	46.5$_{\pm 0.17}$	76.4$_{\pm 0.25}$	48.2$_{\pm 0.05}$	95.4$_{\pm 0.07}$
CEP+B	**62.3**$_{\pm 0.87}$	**85.1**$_{\pm 0.46}$	**54.3**$_{\pm 0.86}$	**79.2**$_{\pm 0.93}$	**63.0**$_{\pm 0.92}$	**87.5**$_{\pm 0.32}$	**53.0**$_{\pm 0.28}$	**82.8**$_{\pm 0.13}$	45.0$_{\pm 0.48}$	95.0$_{\pm 0.08}$

Table 5.3: Test accuracies for different losses on the noisy CIFAR100/Asym-CIFAR100/Non-Uniform EMNIST datasets. Losses implementing the noise-bound shaded in blue. When using this bound provides benefit, the corresponding value is $\boxed{boxed}$. Overall top values in **bold**.

5.6.1 Limitations and Future Work

While effective in specific settings, our method has limitations due to its reliance on a data separability assumption. This can restrict its effectiveness on datasets with inherent randomness. Future research could extend these methods to non-separable datasets. Also, while our approach improves on methods requiring detailed noise models, it can be applied only in settings where the noise rate is approximately known.

Although our proposed method generally offers benefits, there are observable differences in performance between different loss functions. Understanding these differences is a crucial direction for future research. Another promising area of future work involves extending these ideas to backward-corrections (Patrini et al., 2017), which are more prone to overfitting than forward-corrected losses.

Chapter 6

Early Stopping For Noisy Labels

6.1 Introduction

6.1.1 Chapter Summary

In Chapter 5, we saw how overfitting on nosily labelled datasets could be mitigated by preventing training loss from going below the entropy of the noisy distribution. However, computing the entropy requires an estimate of the noise rate - something which may not always be known. Moreover, the results of this section assume (approximate) separability of the clean data distribution. In this chapter, we investigate an alternative approach to preventing overfitting, which avoids the limitations of risk-bounding: Early Stopping. Under ideal circumstances, Early Stopping (ES) utilises a validation set uncorrupted by label noise to monitor generalisation during training effectively. However, this can be costly and challenging to obtain. This study establishes that, in many typical learning environments, a clean validation set is unnecessary for effective Early Stopping. Instead, near-optimal results can be achieved by monitoring accuracy on a **noisy** dataset throughout the training process. Referred to as 'Noisy Early Stopping' (NES), this method simplifies and reduces the cost of implementing Early Stopping. We provide theoretical insights into the conditions under which this method is effective and empirically demonstrate its robust performance across standard benchmarks using common loss functions.

6.1.2 Context and Problem Statement

Robust loss functions have been developed to counteract the tendency of the widely used cross-entropy loss to overfit when faced with noisy data. However, these approaches still result in overfitting or underfitting in various settings. It is unrealistic to expect that a single loss function could avoid both overfitting and underfitting across all datasets, noise models, and hyperparameter configurations. Therefore, it becomes crucial to understand how and when to implement Early Stopping, particularly when cleanly labelled validation datasets are not available to monitor generalisation during training. A related problem is how to compare and rank label-noise-robust machine learning algorithms in the absence of cleanly labelled test datasets. Practitioners currently resort to comparing methods via the accuracy they obtain on *noisy* test set, e.g. for the Clothing1M dataset, which lacks a clean subset. This chapter aims to explore, theoretically and empirically, under which conditions optimising noisy accuracy aligns with optimising clean accuracy.

Figure 6.1: Illustration of the difference between noisy and clean accuracy and the non-trivial relationship between them: A web-scraped dataset of chihuahuas contains label noise as it has accidentally scraped images of muffins (Cortinhas, 2022). A model, which correctly identifies all of the chihuahuas (red), will obtain a **noisy accuracy** of $\frac{4}{8} \approx 50\%$ despite a **clean accuracy** of 100%.

Contributions The primary purpose of this chapter is to demonstrate that noisy test accuracy - i.e. accuracy on a held-out dataset, drawn from the same distribution as the noisy training set - can often be used to reliably evaluate generalisation to the *clean* (un-noised) data distribution. Consequently, noisy accuracy can be used to define an effective policy for Early Stopping - specifically stopping training when the noisy validation accuracy starts to drop - we call this 'Noisy Early Stopping'. These results hold for the standard image datasets and standard noise models, including non-uniform and asymmetric label noise. This finding is useful since it 1) Gives ML practitioners a simple and reliable way to early stop in the presence of label noise 2) Partially validates the existing approach of using noisy test accuracy to evaluate and compare label-noise robust algorithms when no cleanly labelled dataset exists.

6.1.3 Chapter Outline

Section 6.2 introduces the background and terminology. We define 'Noisy Early Stopping' (NES) as the strategy of Early Stopping by monitoring performance on a validation set polluted by label noise. Section 6.3 discusses related work. In Section 6.4, we derive relationships between the clean and noisy 0-1 risks of a model in different label noise environments.

The theory derived in Section 6.4 suggests that NES should be effective for symmetric label noise and not other noise types. In Section 6.5, we empirically evaluate NES. Remarkably, we demonstrate the effectiveness of NES across various datasets, noise models, and six popular robust loss functions.

In Section 6.6, we build a partial explanation for the effectiveness of NES. Under a separability assumption, we show that NES should be effective when overfitting occurs simultaneously across classes. We show experimentally that this condition is satisfied when training neural network classifiers. Finally, in Section 6.7, we discuss our findings, exploring how they can be integrated with other methods, the limitations of our study, and potential avenues for future work.

6.2 Background

6.2.1 Terminology

Throughout this section we use the term *'risk'* (defined in Section 2.1.1) to refer to 0-1 risk unless specified otherwise. When computed with respect to the noisy distribution $\widetilde{p}(x,\widetilde{y})$ it is called the *noisy risk*, denoted $R^{\eta}(\boldsymbol{q})$, distinguished from the clean risk through an η superscript.

'Noisy' and 'Clean' More broadly, we use '*noisy*' and '*clean*' to distinguish between quantities evaluated using the noised or un-noised data distributions, respectively. For example, 'clean accuracy' refers to accuracy computed on samples drawn from the un-noised distribution, whereas 'noisy accuracy' refers to accuracy computed in samples drawn from the noisy distribution $\widetilde{p}(x,\widetilde{y})$ (See Figure 6.1 for an illustration of the difference between noisy and clean accuracy). Similarly, 'clean Early Stopping' will refer to Early Stopping, where generalisation is evaluated using a cleanly labelled validation set; this is to be contrasted with 'Noisy Early Stopping', where generalisation is evaluated on a validation set whose labels are corrupted with label noise.

6.2.2 Assumptions and Problem Statement

Throughout the chapter we assume that we have a finite set of probability estimators $\mathcal{Q} := \{\boldsymbol{q}_i\}_{i=1}^{N}$ and our objective is to select the model with the lowest clean 0-1 risk. We assume we have a sufficiently large noisy validation set from which we can estimate the noisy 0-1 risk of these models with arbitrarily low variance. Our policy is to select the estimator within $\mathcal{Q}$ with the minimal noisy 0-1 risk. This section aims to determine when this policy yields an optimum of the clean 0-1 risk within $\mathcal{Q}$ or otherwise bound the worst-case cost of this noisy selection policy.

6.2.2.1 Class-Preserving Label Noise

Many of the results of this chapter utilise the class-preserving assumption defined in Section 2.2.2. Recall that we say label noise is *class-preserving* when it preserves the dominant class.

Definition 6.2.1 (Class-Preserving Noise)**.** *Given a data-label distribution $p(x,y)$ and its noisy version $\widetilde{p}(x,\widetilde{y})$, resulting from label noise, the noise is considered class-preserving if, for every $x \in \mathcal{X}$, the dominant class (refer to Definition 4.2.1) remains unchanged after noise application. Formally, this is expressed as:*

$$\underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \widetilde{\boldsymbol{p}}(\widetilde{y}=i|x) = \underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \boldsymbol{p}(y=i|x), \tag{6.1}$$

where c is the number of classes.

Class-preserving noise *preserves* which class has the highest probability.

6.2.3 Early Stopping

Early Stopping (ES) is a regularisation technique in machine learning designed to prevent overfitting by terminating the training process once generalisation begins to worsen (Prechelt, 2002). While numerous criteria can be used to determine when to early-stop, the 'gold-standard' approach (Mahsereci, Balles, Lassner, & Hennig, 2017) monitors performance on a held-out validation set drawn from the distribution of interest. Suppose we are learning a probability estimator, letting $\boldsymbol{q}_n$ denote the model obtained after n training epochs. We have a sequence of models $\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3, \ldots$. Utilising a loss function L and a large validation dataset, we can estimate the risks of these models with low variance: $\widehat{R}_L(\boldsymbol{q}_n) \approx R_L(\boldsymbol{q}_n)$. This allows us to detect when generalisation starts to decline, marking an optimal point to halt training. The cessation occurs if no improvements are noted over a set number of epochs, a period known as the 'patience'.

The Problem Of Noise Early Stopping is generally effective when the validation set is representative of the distribution to which we wish to generalise. However, complications arise when both the training and validation datasets are contaminated by systematic noise. Consider a regression scenario where an agent randomly adds a small positive value $\varepsilon \in [0,1]$ to the targets in both the training and validation sets. In this case, the model may inadvertently fit to this noise, impairing its ability to generalise to the true underlying distribution. Crucially, Early Stopping is unlikely to correct for this issue, as the validation set, being similarly polluted, does not accurately reflect the target distribution we seek to generalise to.[1]

[1] In this study we look exclusively at *classification*, not regression, this example is purely illustrative.

6.2.4 Main Proposal: Noisy Early Stopping

When Early Stopping is applied by monitoring the accuracy on a validation set that is subject to the same label noise as the training set, we call this **Noisy Early Stopping** (NES) (Expressed algorithmically in Algorithm 2). This technique aims to optimise training cessation to maximise the model's generalisation performance on *cleanly-labelled* data. We contrast Noisy Early Stopping with clean Early Stopping (Abbreviated simply as ES), an idealised setting in which we have a cleanly-labelled validation set to monitor clean performance.

6.2.5 Assumptions and Problem Statement

The primary objective of this work is to evaluate the effectiveness of the Noisy Early Stopping algorithm (Algorithm 2) across a broad range of conditions when training with noisy labels. We begin with a theoretical analysis in Section 6.4, based on two idealised assumptions:

1. We assume the availability of an arbitrarily large validation dataset, allowing the noisy risk at each epoch to be estimated with high precision and independently[2].

2. We assume a large patience parameter, allowing Noisy Early Stopping to select from any model obtained during training.

These assumptions allow us to formulate the problem as follows:

Problem Statement Consider a data-label distribution $p(x,y)$ and its noisy-label counterpart $\widetilde{p}(x,\widetilde{y})$. Let Q denote a finite set of probability estimators $\{\boldsymbol{q}_n\}_{n=1}^{N}$. The goal is to identify, through theoretical and empirical analysis, the conditions under which selecting the estimator from Q with the minimal noisy 0-1 risk leads to optimal clean 0-1 risk performance.

6.3 Related Work

A summary of the most relevant literature is given below. A comprehensive related work may be found in Chapter 3.

[2]i.e. The noisy risk estimates are statistically independent random variables; $\widehat{R}^{\eta}(\boldsymbol{q}_i) \perp\!\!\!\perp \widehat{R}^{\eta}(\boldsymbol{q}_j)$

Algorithm 2 Noisy Early-Stopping Algorithm

Require: Noisy dataset D, split into training set D_{train} and validation set D_{val}; Fisher consistent Loss function L (e.g., cross-entropy); Patience parameter P

Ensure: Trained model with Early Stopping based on noisy validation accuracy

initialise neural network f with random weights
initialise patience counter $p \leftarrow 0$
initialise best validation accuracy $\text{acc}_{\text{best}} \leftarrow 0$
while $p < P$ **do**
    for each epoch **do**
        Train f on D_{train} using loss function L
        Calculate noisy accuracy $\text{acc}_{\text{epoch}}$ on D_{val}
        if $\text{acc}_{\text{epoch}} > \text{acc}_{\text{best}}$ **then**
            Update $\text{acc}_{\text{best}} \leftarrow \text{acc}_{\text{epoch}}$
            Save the current model as the best model
            Reset patience counter $p \leftarrow 0$
        else
            Increment patience counter $p \leftarrow p + 1$
        end if
    end for
    if $p \geq P$ **then**
        break
    end if
end while
Load the best saved model
return The trained model f

Overfitting Persists The development of robust loss functions has been fruitful but challenges remain. While robust loss functions are generally less vulnerable to label noise than cross-entropy, they still exhibit overfitting or underfitting problems (Ma et al., 2020; Z. Zhang & Sabuncu, 2018). For instance, losses like Mean Absolute Error (MAE) and certain normalised losses tend to underfit, while others (e.g., backward corrections, Generalised Cross-Entropy (GCE), etc.) tend to overfit, albeit less severely than cross-entropy Patrini et al. (2017). Each loss function behaves differently depending on numerous factors, including factors such as the number of classes, the noise model, hyperparameters (e.g., learning rate and batch size), and the datasets. Thus, a loss function may avoid overfitting in one setting but overfit or underfit in another. Consequently, it is crucial to develop methods to reliably measure model generalisation during training to assess whether overfitting (or underfitting) is occurring and to determine the optimal point for Early Stopping.

ES for Label Noise Limited work addresses Early Stopping (ES) for classification in the presence of label noise. Bai et al. (2021) introduces Progressive Early Stopping (PES), a technique that trains initial neural network layers before implementing ES on later layers, which are more prone to overfitting, without using a validation set and halting training after a preset number of epochs. This approach raises questions about optimally determining duration without clean validation or test sets for hyperparameter tuning. Xia et al. (2021) presents 'robust-early-learning,' which manages label noise by dividing parameters into critical and non-critical sets with different update rules and utilises ES on a noisy validation set. Yuan, Feng, and Liu (2024) proposes an approach to ES that uses statistical properties of gradients during training to decide when to stop, without needing a validation set. This method aims to use all available training data to maximise model performance in noisy limited data environments. M. Li et al. (2020) highlights ES's efficacy for symmetric label noise and theoretical effectiveness in one-hidden layer networks under specific data distribution assumptions. Our analysis differs notably in that their ES criteria are based on clustering properties of the data distribution, not on generalisation from noisy data. In adversarial robustness, (Rice et al., 2020) finds ES comparably effective to other robust methods when a cleanly labelled dataset is available for generalisation assessment. Song, Kim, Park, and Lee (2019) introduces 'Pre-stopping,' using ES in a label noise robust pipeline, assuming a small cleanly labelled dataset for assessment. The backward correction allows ES

without a clean validation set when the true transition matrix is known, as the empirical risk on a noisy set becomes an unbiased estimator of the un-noised risk, though its effectiveness depends on the accuracy of the transition matrix estimate and is limited in non-uniform label noise scenarios.

Without Early Stopping The majority of studies introducing label noise robust loss functions appear not to utilise any sort of Early Stopping at all (X. Wang et al., 2023; Y. Wang et al., 2019; Z. Zhang & Sabuncu, 2018). Instead, these approaches train for a pre-specified number of epochs, relying on the loss functions intrinsic robust properties to mitigate overfitting during the training run. However, multiple works, including (Patrini et al., 2017; Q. Wang et al., 2021), while not implementing Early Stopping, use a noisy validation set for hyperparameter tuning.

Noisy and Clean 0-1-Risk A common Early-Stopping approach consists of monitoring the loss (typically cross-entropy) on the validation set (Mahsereci et al., 2017). In this study, however, we adopt the approach of monitoring the validation *accuracy* (0-1-loss) (Xia et al., 2021). Consequently, understanding the relationship between the noisy and clean 0-1 risks of a classifier is crucial. Prior research, such as (Ghosh et al., 2015; Manwani & Sastry, 2013), has demonstrated the robustness of the 0-1 loss to uniform symmetric noise within binary classification contexts. Building on this, (Ghosh & Kumar, 2017) establishes conditions under which the minimisers of both noisy and clean risks coincide for loss functions that exhibit a 'symmetry' property. Given that the 0-1 loss satisfies this symmetry criterion, the findings of Ghosh and Kumar (2017) apply to our work. While the primary focus of the referenced papers differs from ours, the results presented in Section 6.4 concerning symmetric label noise naturally extend the relationships they derived between noisy and clean risks.

6.4 The Relationship Between Noisy and Clean 0-1-Risk

This section provides conditions under which the minimiser of the noisy risk in a set of estimators $\mathcal{Q}$ coincides with the minimiser of the clean risk in $\mathcal{Q}$. These conditions are summarised into five propositions. Precise statements and proofs are given in Appendix D.1.

6.4.1 Uniform Symmetric Label Noise

The most studied label noise type is uniform symmetric label noise (Song et al., 2023) in which the transition probabilities between every pair of distinct labels are the same.

Proposition 6.4.1 (Symmetric Uniform Noise)**.** *Given a set of estimators* $Q := \{\boldsymbol{q}_n\}_{n=1}^{N}$ *and any data-label distribution. The minimiser of the noisy risk within the set* Q *will also minimise the clean risk if the label noise is simultaneously uniform, symmetric and class-preserving.*[3] *Moreover, the noisy and clean risks are related by an affine linear relationship;*

$$R^{\eta}(\boldsymbol{q}) = R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta. \tag{6.2}$$

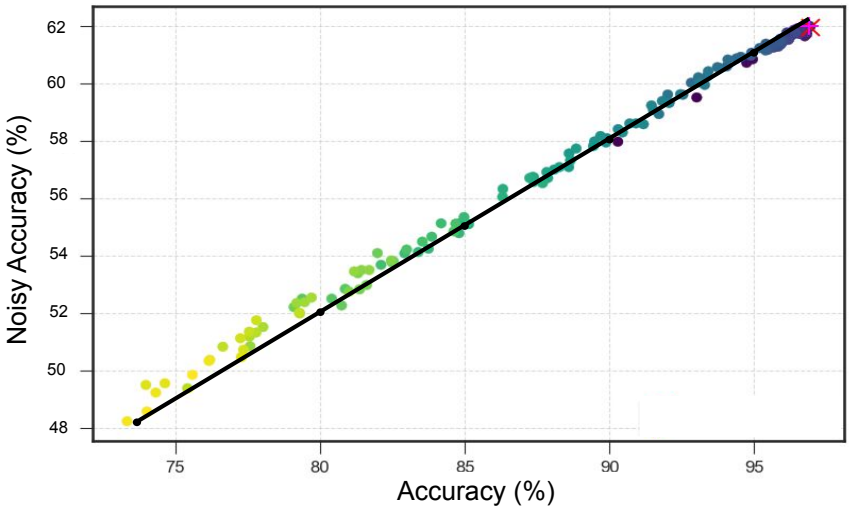


Figure 6.2: Symmetric label noise - noisy vs clean accuracy: A classifier model is trained using cross-entropy loss on the MNIST dataset, corrupted by 36% symmetric label noise. We plot the model's noisy and clean accuracies against each other at the end of each epoch, with early epochs coloured in dark blue and later epochs (around 100) in yellow. As expected, a linear relationship emerges between the noisy and clean accuracies. The black line depicts the theoretical relationship (Equation 6.2), showing near-perfect alignment between the experimental results and theoretical predictions.

Proposition 6.4.1 asserts that, in the case of uniform symmetric label noise, the model that minimises the noisy risk within Q will also minimise the clean risk and that the clean and noisy risks should be related by a linear mapping. Figure 6.2 illustrates the relationship between noisy and clean accuracy for a neural network model at each

[3]Recall that we adopt the convention of using 'risk' as shorthand for 0-1 risk in this study.

training epoch on the symmetrically-noised MNIST dataset (36% noise). The black line represents the expected theoretical relationship, demonstrating strong alignment with the experimental results. We may conclude from Proposition 6.4.1 that a Noisy Early Stopping policy is likely to be effective under uniform symmetric label noise.

6.4.2 Asymmetric and Non-Uniform Label Noise

The theoretical result established by Proposition 6.4.1 for uniform symmetric label noise is strong. Proposition 6.4.2 establishes that uniform symmetric label noise is the *only* noise model for which this strong result holds: For all other label noise models, there is *no inherent reason to assume that performance evaluations on a noisy dataset will reliably reflect performance on the underlying clean distribution*.

Proposition 6.4.2. *Let $p(\widetilde{y} \mid y,x)$ be a label noise model with the property that, for any set of estimators $Q := \{\boldsymbol{q}_n\}_{n=1}^N$ and any data-label distribution, the minimiser of the noisy risk within Q coincides with the minimiser of the clean risk within Q; then $p(\widetilde{y} \mid y,x)$* must *describe uniform, symmetric and class-preserving label noise.*

Example: Decision Tree Classifier Figure 6.3 (Left) plots the noisy versus clean validation accuracy for **decision tree classifiers** of increasing depth trained on a dataset corrupted by pairwise (*asymmetric*) label noise at 42%. Low-depth models are indicated with dark blue and deeper classifiers with yellow. In sharp contrast to Figure 6.2, the relationship between clean and noisy validation accuracy is quite chaotic. In particular, the depth that optimises the noisy accuracy (indicated by the red vertical dotted line) is highly suboptimal for clean accuracy, achieving an accuracy 18% lower than optimal accuracy.

6.4.2.1 Additional Assumptions

Under specific assumptions about the data distribution and the model set, optimising noisy accuracy can align with optimising clean accuracy. The following propositions demonstrate examples of such assumptions:

Proposition 6.4.3 (Bayes-Optimality)**.** *Let $p(\widetilde{y} \mid y,x)$ be a class-preserving label noise model. If the set of estimators $Q := \{\boldsymbol{q}_n\}_{n=1}^N$ contains a Bayes-optimal estimator (a minimiser of the clean risk over* all *possible estimators) then this estimator will be a minimiser of the noisy risk in Q.*

Proposition 6.4.4 (Correlation)**.** *For* non-uniform*, symmetric, class-preserving noise, the minimiser of the noisy risk within $\mathcal{Q}$ will coincide with the minimiser of the clean risk within $\mathcal{Q}$ if each of the estimators in $\mathcal{Q}$ are uncorrelated with the noise model. More generally, the worst-case performance of selecting the estimator with minimal noisy risk can be bounded in terms of the correlation between the noise model and the estimators.*

Proposition 6.4.5 (Bound for Asymmetric Noise)**.** *For uniform,* asymmetric*, class-preserving label noise one can upper-bound the worst-case clean risk difference between the minimiser of the noisy risk ($\boldsymbol{q}^{\eta}_{*}$) and the minimiser of the clean risk ($\boldsymbol{q}_{*}$) in $\mathcal{Q}$, i.e. we upper bound $|R(\boldsymbol{q}^{\eta}_{*}) - R(\boldsymbol{q}_{*})|$. The upper bound is given in terms of the maximum and minimum transition probabilities between classes and the minimal achievable noisy risk within $\mathcal{Q}$.*

6.4.3 Expectations

Based on the five Propositions (6.4.1,6.4.2,6.4.3,6.4.4 and 6.4.5) we anticipate that Noisy Early Stopping (NES) should be effective for uniform symmetric label noise. However, for other noise models, the effectiveness of NES appears less promising. During typical gradient descent training of classifiers, achieving a Bayes-optimal classifier is unlikely; thus, Proposition 6.4.3 does not hold. Furthermore, Proposition 6.4.4 applies to non-uniform symmetric (not *asymmetric*) noise and the bounds outlined in Proposition 6.4.5 for more general noise types are poor[4]. Additionally, it is uncertain whether the classifiers in $\mathcal{Q}$ can remain uncorrelated with the noise model, given that they are trained on data affected by this noise. Moreover, Figure 6.3 (Left) illustrates that NES performs poorly when optimising the depth of decision tree classifiers for asymmetric label noise. For decision trees, model complexity is regulated by tree depth. For neural networks the number of training epochs regulates model complexity. Hence, given the failure of NES in optimising the depth parameter for decision trees it is reasonable to anticipate that NES may also significantly underperform ES for neural network classifier under most label noise conditions.

[4]See Appendix D.1.4.1 for further discussion

6.5 Experiments

6.5.1 Experiment Details

We evaluate Noisy Early Stopping (NES) against clean Early Stopping (ES), which uses a cleanly labelled validation set, and Without Early Stopping (WES), the standard for noisy label training.

6.5.1.1 Experiment Setup

We prepare our experiments using cleanly labelled datasets divided into training, validation, and test segments. For Noisy Early Stopping (NES), we inject synthetic label noise into both the training and validation sets. We then train neural network classifiers on this noisy training data for 100-150 epochs, evaluating them at each epoch using the noisy validation set accuracy. If there is no improvement in validation accuracy for ten epochs, we halt training and revert to the best model observed (patience parameter $P = 10$).

We follow a similar training procedure for clean Early Stopping (ES), but the validation set remains clean and unaffected by synthetic noise. In contrast, the Without Early Stopping (WES) method involves continuous training through the predetermined epoch count without monitoring validation performance. All methods are ultimately evaluated on the clean test set.

Remark In real-world scenarios, a clean validation set may not be available. The comparison with ES serves to contextualise NES's performance relative to an idealised 'gold-standard' baseline (Mahsereci et al., 2017).

6.5.1.2 Datasets and Noise Models

We evaluate the performance of NES/ES/WES across ten noisy datasets: FashionMNIST, MNIST, CIFAR10, and CIFAR100, all corrupted by symmetric label noise, as well as a variety of non-symmetric label noise types. These include non-uniform asymmetrically noised EMNIST (NU-EMNIST), non-uniform asymmetric MNIST (NU-MNIST), and uniform-asymmetric noise variants of FashionMNIST, MNIST, CIFAR10, and CIFAR100. Full details of these label noise models are provided in Appendix D.2.1. Although Clothing1M and Animals-10 are common datasets for evaluating label noise-robust methods, they lack cleanly labelled validation sets and, therefore, are unsuitable for testing this work's core hypothesis.

Loss Functions We evaluate the performance of NES, ES, and WES using six different loss functions: Cross-Entropy (**CE**), and five popular robust loss functions: **MSE** (Janocha & Czarnecki, 2016), **GCE** (Z. Zhang & Sabuncu, 2018), **SCE** (Y. Wang et al., 2019), forward-corrected CE (**FCE**) (Patrini et al., 2017), and backward-corrected CE (**BCE**) (Natarajan et al., 2013). The transition matrix used in the backward and forward-corrected loss functions is set to the true noise model used to construct the label noise, except for non-uniform label noise, where it is set to uniform symmetric at the given noise rate.

6.5.2 NES Results

Table 6.1 presents the results of each method (NES/WES/ES) across the datasets for each of the six loss functions. The table shows the average final clean test accuracy and the standard deviation over three runs. The results of our proposed NES method are shaded in light blue. Where the performance of NES exceeds that of ES, or their uncertainty intervals overlap, the corresponding entry is **bolded**. This occurs in **93**% of settings, indicating that NES performs nearly as well as ES despite using a noisy validation set instead of a clean one. These results are consistent across different loss functions and hold for datasets corrupted by uniform symmetric label noise and those affected by other noise types, despite the absence of strong theoretical guarantees.

Table 6.1 also presents the results of the standard Without Early Stopping (WES) approach for these datasets. The final clean test accuracies for WES are often substantially lower than those for ES and NES, highlighting the importance of Early-Stopping approaches to mitigate overfitting, even when using robust loss functions. NES outperforms WES in **75**% of the experimental settings.

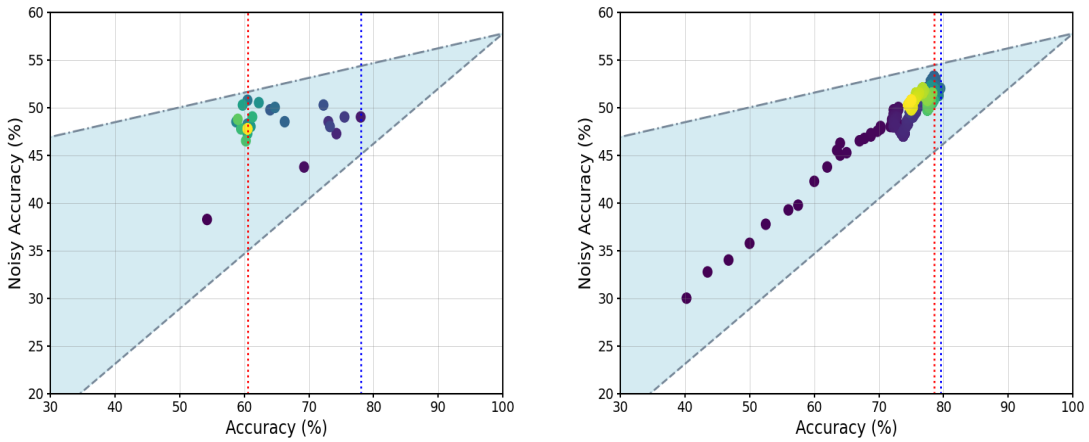
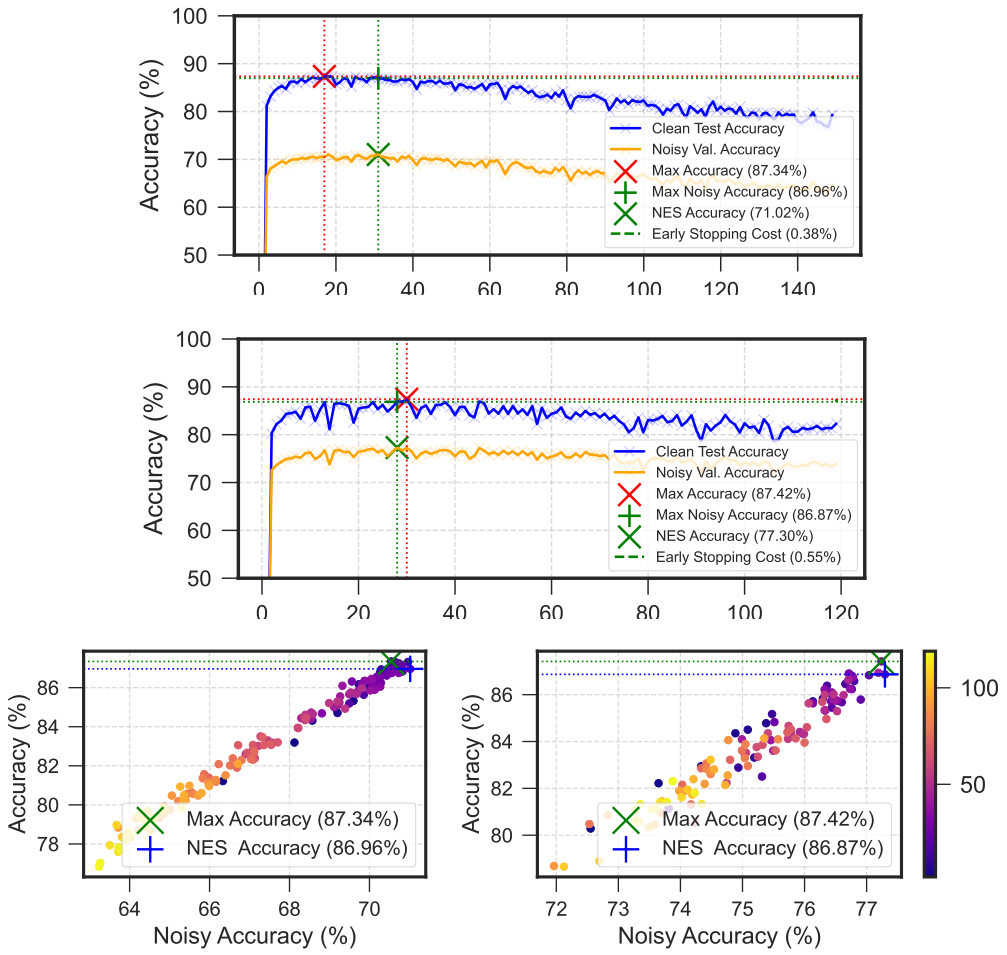


Figure 6.3: Noisy validation accuracy plotted against clean validation accuracy: **Left: Decision Tree Classifier** at increasing depths, fitted to a dataset containing 42% pairwise label noise. Shallow depth models are represented in blue, transitioning to yellow as depth increases. A red dotted vertical line highlights the classifier achieving the highest noisy validation accuracy, while a blue dotted vertical line marks the classifier with the highest clean validation accuracy. The significant horizontal gap between these lines illustrates the limited effectiveness of Noise Early Stopping (NES) for optimising the depth of decision trees under this type of label noise. **Right: Neural Network Classifier** trained on the same noisy dataset. Early epochs are represented in blue, transitioning to yellow. A red dotted vertical line highlights the epoch with the highest noisy validation accuracy, while a blue dotted vertical line marks the epoch with the highest clean validation accuracy. The small horizontal gap between these lines illustrates the effectiveness of Noise Early Stopping (NES) for neural network models under similar noise conditions. For both graphs, the light blue region represents bounds established by Proposition 6.4.5, indicating that no model may achieve an accuracy/noisy-accuracy combination outside this region.



(a) Symmetrically-Noised FashionMNIST (b) Asymmetrically-Noised FashionMNIST

Figure 6.4: The top figure shows the clean test accuracy (blue) and noisy validation accuracy (yellow) during training on symmetrically-noised Fashion dataset ($\eta = 0.2$), highlighting the maximum clean accuracy (red cross $\times$) and the accuracy obtained by NES (green plus $+$) with a minimal difference of 0.38%. The second figure displays similar metrics for *asymmetrically*-noised FashionMNIST, with a comparable accuracy difference of 0.55%. The bottom two figures provide the same information as the top two but are expressed differently (FashionMNIST on the left and MNIST on the right). We plot the clean test accuracy at each epoch is plotted against the noisy validation accuracy. Epochs are coloured so that the first few epochs are blue and the final epochs are yellow, with the hue shifting gradually from blue to yellow through red. Initially, the noisy and clean accuracy both increase and the data points move into the upper right corner of the graph, after which overfitting occurs, and both the noisy and clean accuracies decline. On both datasets, the noisy and clean accuracies are maximised approximately simultaneously.

6.5.3 Plots and Figures

In Section 6.4, we evaluated the effectiveness of Noisy Early Stopping (NES) for determining the optimal depth of a decision tree classifier trained with noisy labels. NES proved ineffective for asymmetric label noise. Figure 6.3 (Right) explores the application of NES to neural network classifiers for this noisy dataset. Contrary to the findings with decision trees, we observe that for the neural network, the noisy and clean validation accuracies peak almost simultaneously, indicating the effectiveness of NES in this context.

Figure 6.4 displays the clean test accuracy (blue) and noisy validation accuracy (yellow) after each epoch for a classifier trained on symmetrically noised FashionMNIST (top) and asymmetrically noised FashionMNIST (2^{nd} from top). On each graph, we indicate the epoch at which the clean test accuracy is maximised with a red cross $\times$ and the maximum noisy validation accuracy with a green plus $+$. We then mark the clean test accuracy obtained by the model which maximises the noisy validation accuracy with a green dotted line, and the maximum clean test accuracy attained during training with a red dotted line. For both datasets, these lines are almost identical; the clean test accuracy attained by the model which maximises the noisy validation accuracy is within a fraction of a percent of the optimal clean test accuracy. This re-emphasises that Early Stopping using a noisy validation set can achieve nearly optimal clean test accuracy.

The bottom of Figure 6.4 presents the same data as the top two figures, albeit expressed differently (symmetric noise on the left and asymmetric on the right). We plot the clean test accuracy at each epoch against the noisy validation accuracy. Each epoch is coloured so that the initial epochs are blue and the final epochs are yellow, with the hue gradually shifting from blue to yellow through red. This visualisation helps trace how the clean and noisy (validation) accuracies evolve during training. Initially, both accuracies increase and the data points move into the upper right corner of the graph, after which overfitting occurs, and both accuracies decline. Crucially, overfitting on the noisy and clean datasets co-occurs for both datasets, meaning the peak clean and noisy accuracies occur simultaneously. The left figure uses uniform symmetric label noise, where we know that the noisy and clean accuracies are related via a linear map (See Lemma D.1.2) - Figure 6.4 confirms this linear relationship. Despite the absence

of similar guarantees for the asymmetrically-noised FashionMNIST dataset, a similar relationship is visible in its graph. We emphasise that the simultaneous peaking of the noisy and clean accuracies for AsymFashionMNIST is non-trivial and somewhat unexpected.

6.5.3.1 The Importance of Being Class-Preserving

In Figure 6.5 we visualise the performance of NES as a function of the noise rate. We plot the final (clean) test accuracy of models trained using NES in blue and the final (clean) test accuracy of ES in yellow against increasing noise rates for two noisy datasets (AsymMNIST left and FashionMNIST right). The class-preserving condition is violated beyond a certain threshold noise level, highlighted by a vertical dashed line. Until this point, noisy and clean ES perform identically, with an immediate divergence once the class-preserving property no longer holds. This finding reaffirms the necessity for label noise to be class-preserving for NES to be an effective substitute for clean ES. When the noise rate is increased to the point where the label noise is no longer class-preserving, the original signal relating the data points to the label is essentially destroyed. As a result, the noisy accuracy of a model no longer aligns with its clean accuracy.

6.5.4 Implications of Findings

The results shown in Table 6.1 and Figure 6.3 (right) illustrate that NES performs well and is about as effective as the clean Early Stopping idealised baseline. While this was expected for uniform symmetric label noise, the fact that NES works for other noise types is remarkable. This means that even when no clean validation set is available, we can early stop and obtain near-optimal clean test performance. The disparity in performance between the Without-Early-Stopping baseline and NES underscores the importance of employing Early Stopping when data is corrupted by label noise. Therefore, discovering that we can do almost optimal Early Stopping without a cleanly labelled validation set is a meaningful finding.

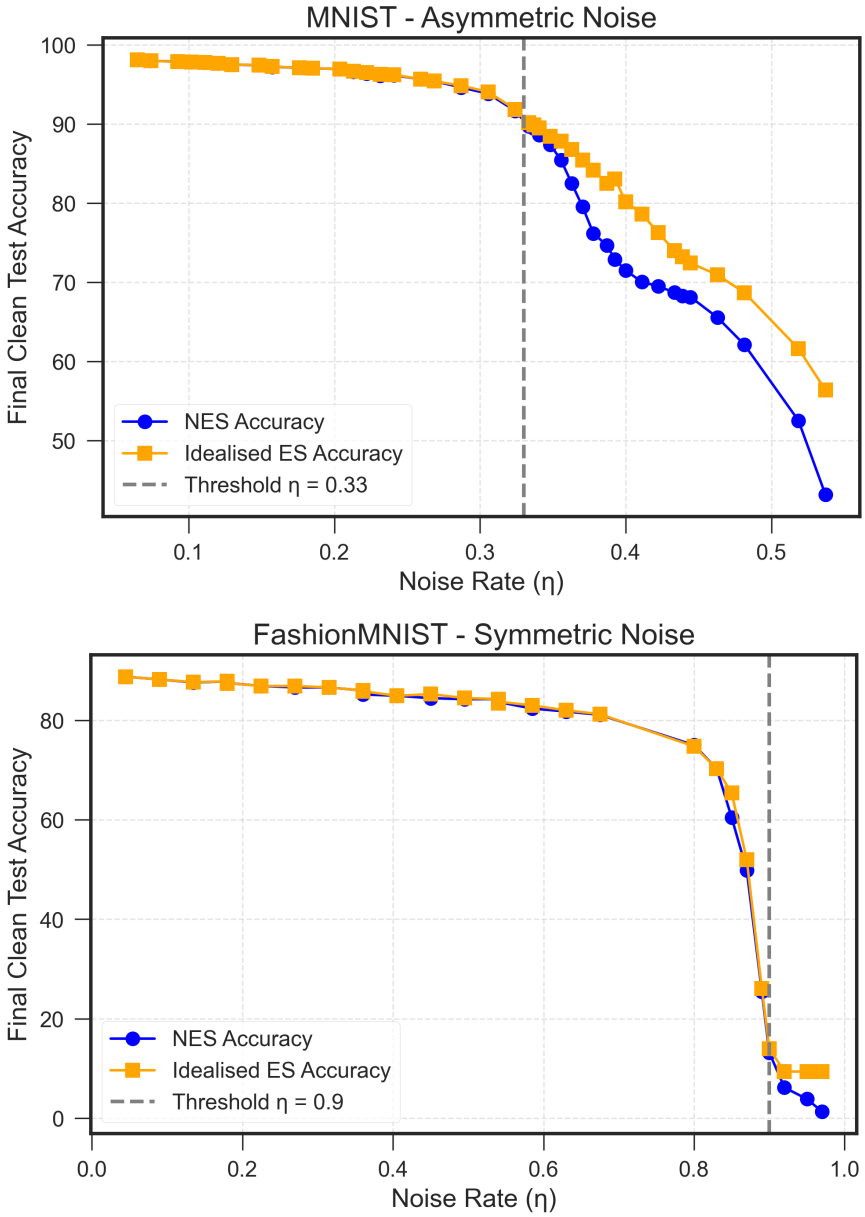


Figure 6.5: Comparison of NES (blue) and idealised Early Stopping (ES) (yellow) (using a clean validation set) on two datasets. We plot the final clean test accuracies against increasing noise rates (η) for the asymmetrically-noised MNIST dataset (top) and the symmetrically-noised Fashion-MNIST dataset (bottom). For MNIST, a vertical line at $\eta = 0.33$ indicates the threshold beyond which noise ceases to be class-preserving; similarly, for Fashion-MNIST, this threshold is at $\eta = 0.9$. Up to these thresholds, the performances of noisy and idealised Early Stopping closely align, demonstrating the robustness of Noisy Early Stopping under varying degrees of label noise.

Table 6.1: Performance Metrics across Different Datasets and Methods. This table presents a comparison of three methods—NES (Noisy Early Stopping), ES (Early Stopping), and WES (Without Early Stopping)—using six commonly used loss functions across various noisy datasets. NES, highlighted in blue, is our proposed method, placed first to highlight its performance. Bold values indicate where NES overlaps or exceeds the performance of ES (93% of cases), an idealised method that uses a clean validation set, providing a context for NES's efficacy when such a set is unavailable. WES, typically used in training with robust loss functions without Early Stopping, is outperformed by NES in 75% of cases, demonstrating the significant benefits of Early-Stopping strategies.

Dataset		Loss Function					
		CE	MSE	GCE	SCE	FCE	BCE
MNIST	NES	**88.43**$_{\pm 0.65}$	**88.78**$_{\pm 0.57}$	**92.37**$_{\pm 0.72}$	**89.91**$_{\pm 0.28}$	**92.45**$_{\pm 0.58}$	**88.50**$_{\pm 1.06}$
	WES	34.18$_{\pm 3.10}$	52.35$_{\pm 1.74}$	66.64$_{\pm 1.11}$	32.20$_{\pm 1.37}$	86.83$_{\pm 0.83}$	77.23$_{\pm 1.41}$
	ES	88.34$_{\pm 0.74}$	88.78$_{\pm 0.57}$	92.63$_{\pm 0.38}$	89.91$_{\pm 0.28}$	92.50$_{\pm 0.50}$	89.94$_{\pm 0.31}$
Fashion	NES	**83.21**$_{\pm 0.37}$	83.48$_{\pm 0.07}$	**84.77**$_{\pm 0.28}$	**84.17**$_{\pm 0.31}$	**84.43**$_{\pm 0.26}$	**80.35**$_{\pm 0.54}$
	WES	49.94$_{\pm 0.28}$	68.17$_{\pm 1.60}$	80.67$_{\pm 0.46}$	51.06$_{\pm 0.42}$	78.74$_{\pm 0.57}$	68.58$_{\pm 0.30}$
	ES	82.96$_{\pm 0.19}$	83.69$_{\pm 0.09}$	84.77$_{\pm 0.28}$	83.48$_{\pm 0.90}$	84.47$_{\pm 0.32}$	79.84$_{\pm 0.16}$
CIFAR10	NES	59.46$_{\pm 2.51}$	**60.63**$_{\pm 0.33}$	**72.67**$_{\pm 2.04}$	**68.82**$_{\pm 1.39}$	**70.63**$_{\pm 1.63}$	**59.47**$_{\pm 4.19}$
	WES	37.23$_{\pm 1.43}$	34.54$_{\pm 0.88}$	64.21$_{\pm 2.32}$	38.69$_{\pm 0.39}$	54.43$_{\pm 0.61}$	42.32$_{\pm 1.37}$
	ES	63.70$_{\pm 0.98}$	61.22$_{\pm 0.88}$	72.65$_{\pm 2.05}$	67.73$_{\pm 1.19}$	70.74$_{\pm 1.78}$	59.47$_{\pm 4.19}$
CIFAR100	NES	**51.86**$_{\pm 1.71}$	**41.80**$_{\pm 0.99}$	**63.74**$_{\pm 1.55}$	**53.82**$_{\pm 2.31}$	**65.98**$_{\pm 0.45}$	**55.58**$_{\pm 0.25}$
	WES	38.96$_{\pm 1.21}$	48.14$_{\pm 0.85}$	66.53$_{\pm 0.81}$	39.59$_{\pm 0.47}$	47.33$_{\pm 1.12}$	44.82$_{\pm 0.56}$
	ES	51.31$_{\pm 2.69}$	43.33$_{\pm 0.98}$	62.91$_{\pm 1.87}$	54.07$_{\pm 2.14}$	66.34$_{\pm 0.66}$	55.58$_{\pm 0.25}$
AsymMNIST	NES	**95.27**$_{\pm 0.93}$	**94.38**$_{\pm 1.25}$	**97.53**$_{\pm 0.18}$	**96.87**$_{\pm 0.42}$	**97.98**$_{\pm 0.12}$	**94.69**$_{\pm 1.35}$
	WES	83.32$_{\pm 3.81}$	84.03$_{\pm 2.38}$	94.83$_{\pm 0.18}$	78.17$_{\pm 0.30}$	97.53$_{\pm 0.17}$	83.56$_{\pm 1.22}$
	ES	95.28$_{\pm 0.94}$	95.12$_{\pm 0.69}$	97.53$_{\pm 0.10}$	96.87$_{\pm 0.42}$	98.20$_{\pm 0.15}$	95.70$_{\pm 0.10}$
AsymFashion	NES	**72.76**$_{\pm 2.59}$	**73.17**$_{\pm 1.99}$	**74.56**$_{\pm 0.82}$	**73.68**$_{\pm 2.99}$	**77.09**$_{\pm 1.07}$	**77.38**$_{\pm 1.95}$
	WES	68.61$_{\pm 1.38}$	68.57$_{\pm 0.70}$	70.74$_{\pm 0.22}$	68.11$_{\pm 0.87}$	77.65$_{\pm 1.05}$	71.35$_{\pm 0.31}$
	ES	74.24$_{\pm 0.97}$	73.46$_{\pm 1.27}$	73.50$_{\pm 1.17}$	75.03$_{\pm 0.65}$	77.92$_{\pm 0.58}$	71.95$_{\pm 8.99}$
AsymCIFAR10	NES	**83.59**$_{\pm 0.51}$	**82.62**$_{\pm 1.52}$	**82.50**$_{\pm 1.16}$	**81.43**$_{\pm 0.67}$	**84.56**$_{\pm 1.31}$	**74.92**$_{\pm 3.64}$
	WES	84.49$_{\pm 2.02}$	82.80$_{\pm 1.21}$	86.34$_{\pm 0.57}$	85.60$_{\pm 0.64}$	89.62$_{\pm 0.49}$	84.78$_{\pm 0.15}$
	ES	83.59$_{\pm 0.51}$	82.62$_{\pm 1.52}$	82.07$_{\pm 1.13}$	81.43$_{\pm 0.67}$	85.35$_{\pm 0.89}$	76.90$_{\pm 5.03}$
AsymCIFAR100	NES	**73.28**$_{\pm 0.69}$	**54.56**$_{\pm 0.47}$	**57.97**$_{\pm 4.82}$	**72.04**$_{\pm 2.01}$	**66.71**$_{\pm 2.21}$	**41.45**$_{\pm 1.09}$
	WES	70.58$_{\pm 0.73}$	57.25$_{\pm 0.75}$	64.77$_{\pm 0.61}$	68.58$_{\pm 0.30}$	64.49$_{\pm 1.18}$	48.86$_{\pm 0.83}$
	ES	73.28$_{\pm 0.69}$	54.94$_{\pm 0.84}$	62.69$_{\pm 2.91}$	72.04$_{\pm 2.01}$	67.41$_{\pm 1.37}$	42.28$_{\pm 0.84}$
NU-MNIST	NES	**97.41**$_{\pm 0.29}$	**97.68**$_{\pm 0.31}$	**98.10**$_{\pm 0.11}$	**98.03**$_{\pm 0.18}$	**98.01**$_{\pm 0.06}$	**97.07**$_{\pm 0.08}$
	WES	92.79$_{\pm 0.25}$	96.16$_{\pm 0.07}$	98.25$_{\pm 0.03}$	92.99$_{\pm 0.73}$	97.70$_{\pm 0.13}$	95.62$_{\pm 0.66}$
	ES	97.45$_{\pm 0.28}$	97.69$_{\pm 0.32}$	98.10$_{\pm 0.11}$	98.20$_{\pm 0.20}$	98.01$_{\pm 0.06}$	97.07$_{\pm 0.08}$
NU-EMNIST	NES	**91.55**$_{\pm 2.55}$	53.93$_{\pm 27.93}$	11.93$_{\pm 0.16}$	**88.51**$_{\pm 0.64}$	**68.25**$_{\pm 4.17}$	**89.94**$_{\pm 0.94}$
	WES	91.50$_{\pm 2.70}$	76.33$_{\pm 1.16}$	15.14$_{\pm 1.58}$	86.60$_{\pm 0.57}$	71.02$_{\pm 3.76}$	89.22$_{\pm 0.46}$
	ES	92.35$_{\pm 2.28}$	75.10$_{\pm 0.91}$	14.38$_{\pm 2.13}$	89.02$_{\pm 0.38}$	70.88$_{\pm 3.69}$	90.81$_{\pm 0.20}$

Example Application Consider a scenario with a large dataset that requires labelling. To annotate the dataset, we assemble a diverse group of human labellers, from novices to experts. This yields a large dataset with noisy labels where we do not know the precise noise model or the noise rate. We assume that the label noise obeys the class-preserving condition. We select a standard loss function and wish to train a neural network model on the noisy dataset. However, we are concerned about overfitting. We randomly split off a portion of the noisy dataset to form a noisy validation set. During training, we monitor the accuracy on this noisy validation set and cease training once noisy validation begins to decline (subject to some patience parameter). Our experimental results suggest that this would result in almost perfect Early Stopping, achieving near-peak clean test accuracy and avoiding overfitting.

6.6 Why Does NES Work?

In Section 6.4, we showed that unless label noise is uniform and symmetric, the minimiser of the noisy risk within a set of estimators $\mathcal{Q}$ may not be a minimiser of the clean risk. Moreover, the bounds which we derived give weak guarantees; by this, we mean that the minimiser of the noisy risk can generalise very poorly to the clean distribution. Therefore, based on our five Propositions alone, we would not expect NES to be as effective as it turns out to be in Section 6.5. The efficacy of NES, therefore, remains largely unexplained. In this section, we study this topic in more detail and provide a partial explanation for this non-trivial phenomenon, concluding that NES is effective due to the way neural network classifiers fit: Roughly speaking, if the off-diagonal elements of the confusion matrix[5] for the classifier being learned are minimised around the same training epoch, then NES will be effective.

Assumptions Throughout this section, we assume that we train a neural network classifier on a dataset corrupted by label noise. We make a minor notational alteration, letting $\boldsymbol{q}^{(n)}$ denote the model attained after training for n epochs and letting $\mathcal{Q} \coloneqq \{\boldsymbol{q}^{(n)}\}_{n=1}^{N}$[6]. We use the notation $\boldsymbol{q}_*$ to denote the minimiser of the clean risk in $\mathcal{Q}$ and

[5] Strictly speaking, it is not precisely the confusion matrix but something very similar.

[6] We used $\boldsymbol{q}_n$ previously to denote the model attained after training for n epochs, however, this notation results in subscript overcrowding if utilised in the following section; hence the change.

$\boldsymbol{q}_*^\eta$ to denote the minimiser of the noisy risk. That is

$$\boldsymbol{q}_*^\eta := \arg\min_{\boldsymbol{q}\in Q} R^\eta(\boldsymbol{q}),$$
$$\boldsymbol{q}_* := \arg\min_{\boldsymbol{q}\in Q} R(\boldsymbol{q}).$$

We continue to assume the label noise being discussed satisfies the class-preserving assumption (Definition 6.2.1). We assume that the data-label distribution is separable and that the label noise is class-conditional asymmetric, where each column is a permutation of every other column. This latter assumption simplifies the mathematics and helps with the exposition. Results extend to general asymmetric label noise.

6.6.1 Section Outline

In this section, we construct an example setting in which a Noisy Early-Stopping policy would fail to select the clean risk minimiser within Q. Constructing this setting allows us to develop an intuition for why examples like this do not occur in practice. The key idea of this section involves measuring the proportions by which a classifier predicts the most likely noisy label, how often it predicts the second most likely noisy label, how often it predicts the third most likely noisy label, etc. We store these proportions, which we denote $g_1, g_2, g_3, \ldots, g_c$, in a vector which we call the '$g-$vector' for the classifier, and record how the components of this vector change during training. For example, if a classifier has a $g-$vector $\boldsymbol{g} = (g_1, g_2, g_3) = (0.6, 0.3, 0.1)$ this means it predicts the most likely noisy label 60% of the time, the second most likely noisy label 30% of the time and the least likely noisy label 10% of the time. We show, theoretically, that if during training, for $i \geq 2$ each of the g_i attain their minima simultaneously, then $\boldsymbol{q}_*^\eta = \boldsymbol{q}_*$; the noisy risk minimiser is also the minimiser of the clean risk. We conclude by experimentally demonstrating that when training a neural network classifier, this condition is satisfied, thus explaining why NES is effective in practice.

6.6.2 g-vector

This section formally defines the $g-$vector associated with a classifier. We assume, for purposes of convenience, for all $i \neq j$, $\widetilde{p}(\widetilde{y} = i \mid x) \neq \widetilde{p}(\widetilde{y} = j \mid x)$, meaning that for all integers k, $\text{argkmax}_i\, \widetilde{p}(\tilde{y} = i \mid x)$ consists of a single element.[7] This choice aids in the subsequent exposition, which holds in the general case.

[7]Where argkmax is a generalisation of arg max to the k^{th} largest elements in a set. See notation Table 1.

$g-$functions Given an estimator $\boldsymbol{q}$ with plug-in classifier f, noisy data-label distribution $\widetilde{p}(x,\widetilde{y})$ and some $x \in \mathcal{X}$, we define the following set of c binary-valued functions $g_1, g_2, \ldots, g_c$ where $g_k(x)$ is defined;

$$g_k(x) \coloneqq \begin{cases} 1, & \text{if } f(x) = \text{argkmax}_i \, \widetilde{p}(\widetilde{y} = i \mid x) \\ 0, & \text{otherwise} \end{cases} \tag{6.3}$$

Note that for any $x \in \mathcal{X}$ *exactly one* of the functions $g_k(x)$ is equal to one and the rest equal zero. We define the following vector-valued function by concatenating the g_k;

$$\boldsymbol{g}(x) \coloneqq (g_1(x), g_2(x), \ldots, g_c(x)). \tag{6.4}$$

If $g_1(x) = 1$, the associated classifier predicts the most likely (noisy) label to appear at x. If $g_2(x) = 1$, the classifier predicts the second most likely to appear given x and so on for the given (noisy) data distribution.

The expectation of this vector over the data distribution is denoted

$$\boldsymbol{g} \coloneqq \int p(x)\boldsymbol{g}(x)dx \in \Delta.$$

We call this the **g-vector** obtained from the classifier f. The vector, $\boldsymbol{g}$, gives us an average of how often our classifier predicts the most likely label, second mostly likely and so on. For example, if the classifier is Bayes-optimal for the noisy distribution, then $\boldsymbol{g} = (1,0,\ldots,0)$ indicating that at every x in the support of $p(x)$ the classifier f predicts the most likely noisy label.

6.6.3 When Would NES Fail?

When is it true that the minimiser of the noisy risk in Q is not a minimiser of the clean risk? The key insight is identifying that two classifiers f_1, f_2 can have the same clean accuracy and dramatically different noisy accuracies if, whenever f_1 predicts incorrectly, it predicts a probable noisy class whereas, whenever f_2 predicts incorrectly, it predicts an *improbable* noisy class. Put simply, when f_2 is wrong, it is *very* wrong, predicting a class which occurs with very low probability.

To make this more concrete, consider the following simplified example where we have a (three class) data-label distribution $p(x,y)$ where for every $x \in \text{supp}(p(x))$ the clean and noisy conditional class distribution are as follows

$$\begin{aligned} \boldsymbol{p}(y \mid x) &= (1,0,0) \\ \text{and} \quad \widetilde{\boldsymbol{p}}(\widetilde{y} \mid x) &= (0.6, 0.35, 0.05). \end{aligned}$$

This is to say that the clean label at every x is $y = 1$ with probability 100%. Likewise, for every x, the probability of the noisy label being $\widetilde{y} = 1$ is 60%, the probability of the noisy label being $\widetilde{y} = 2$ is 35%, and the probability of the noisy label being $\widetilde{y} = 3$ is 5%. Suppose we have two classifiers f_1, f_2, which each correctly predict the most likely noisy label ($\widetilde{y} = 1$) 60% of the time. However, the remainder of the time, $f_1(x) = 2$ and $f_2(x) = 3$. Utilising the language of Section 6.6.2 we say the $g-$vector for f_1 is $\boldsymbol{g}_1 = (0.6, 0.4, 0)$ and is $\boldsymbol{g}_2 = (0.6, 0, 0.4)$ for f_2. In this case, even though both models have the same *clean* accuracy of 60%, f_1 obtains a noisy accuracy of 50% whereas f_2 obtains a noisy accuracy of 38%. (The noisy accuracy is obtained by computing a dot product between $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x)$ and the $g-$vectors.)

We can employ the principle outlined above to construct instances in which $\boldsymbol{q}_*^\eta$ has a lower clean accuracy than $\boldsymbol{q}_*$. To construct such an instance, one must ensure that wherever $\boldsymbol{q}_*$ does not predict the correct clean label, it predicts an unlikely noisy label. Whereas, wherever $\boldsymbol{q}_*^\eta$ does not predict the correct clean label, it predicts a likely noisy label. This is represented visually in Figure 6.6. Figure 6.6 plots bar charts of the $g-$vectors of $\boldsymbol{q}_*^\eta$ and $\boldsymbol{q}_*$ in a ternary label setting in which NES would fail. The figure shows the proportion by which each classifier predicts the true label[8] (blue), the second most likely noisy label (orange) and the least likely noisy label (green), respectively. While the noisy accuracy is higher for $\boldsymbol{q}_*^\eta$, it has a lower clean accuracy than $\boldsymbol{q}_*$.

6.6.4 Overfitting

Since NES works in practice, pathological examples of the type described in Section 6.6.3 and represented in Figure 6.6 must not arise when we train a classifier. To understand why this, suppose that the minimiser of the noisy risk $\boldsymbol{q}_*^\eta$ is *not* the minimiser of the clean risk ($\boldsymbol{q}_*^\eta \neq \boldsymbol{q}_*$). In this case, there are two possibilities: Either the minimiser or the noisy risk $\boldsymbol{q}_*^\eta$ occurs earlier in training that he clean risk minimiser $\boldsymbol{q}_*$, or $\boldsymbol{q}_*^\eta$ occurs later in training than $\boldsymbol{q}_*$ (depicted in Figure 6.7). In either case, there must exist distinct $i, j \geq 2$ for which g_i increases while g_j decreases. If we can demonstrate that this does not occur in practice, then we will have an explanation for why NES succeeds. We formalise this with the following Lemma.

[8] Crucially since we assume the label noise is class-preserving, the true clean label is also the most likely noisy label.

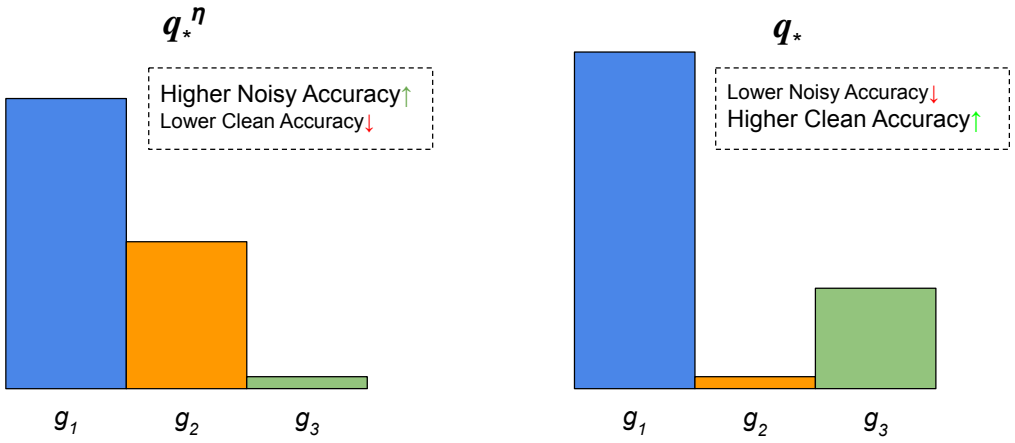


Figure 6.6: Example where NES would fail: Histograms of predictions for $\boldsymbol{q}_*^\eta$ and $\boldsymbol{q}_*$ in the case of three classes. This shows the frequency with which each classifier predicts the true clean label (blue), the second most likely noisy label (orange) and the least likely noisy label (green), respectively. While the noisy accuracy is higher for $\boldsymbol{q}_*^\eta$, it has a lower clean accuracy than $\boldsymbol{q}_*$.

Lemma 6.6.1. *Suppose we train a classifier on a dataset corrupted by (class-preserving) label noise for N epochs. Let $f^{(n)}$ denote the classifier obtained after training for n epochs. Let $\boldsymbol{g}^{(n)} = (g_1^{(n)}, g_2^{(n)}, \ldots, g_c^{(n)})$ denote the $g-$vector of the n^{th} classifier $f^{(n)}$. Suppose there exists an integer $T < N$ so that for all $i \geq 2$, $g_i^{(n)}$ is decreasing for $n \in \{1,2,\ldots,T\}$ and increasing for $n \in \{T, T+1 \ldots, N\}$ then the minimiser of the noisy risk within $\{f^{(n)}\}_{n=1}^N$ also minimises the clean risk.*

6.6.5 Experimental Confirmation

The core hypothesis utilised in Lemma 6.6.1 is that, during training for $i \geq 2$, the g_i start by decreasing. They each reach their minima approximately simultaneously at some epoch T before increasing for the remainder of training. We have shown that under these conditions, the minimiser of the noisy risk in Q will minimise the clean risk, meaning that NES would be effective. It remains to be demonstrated that this assumption holds true in practice.

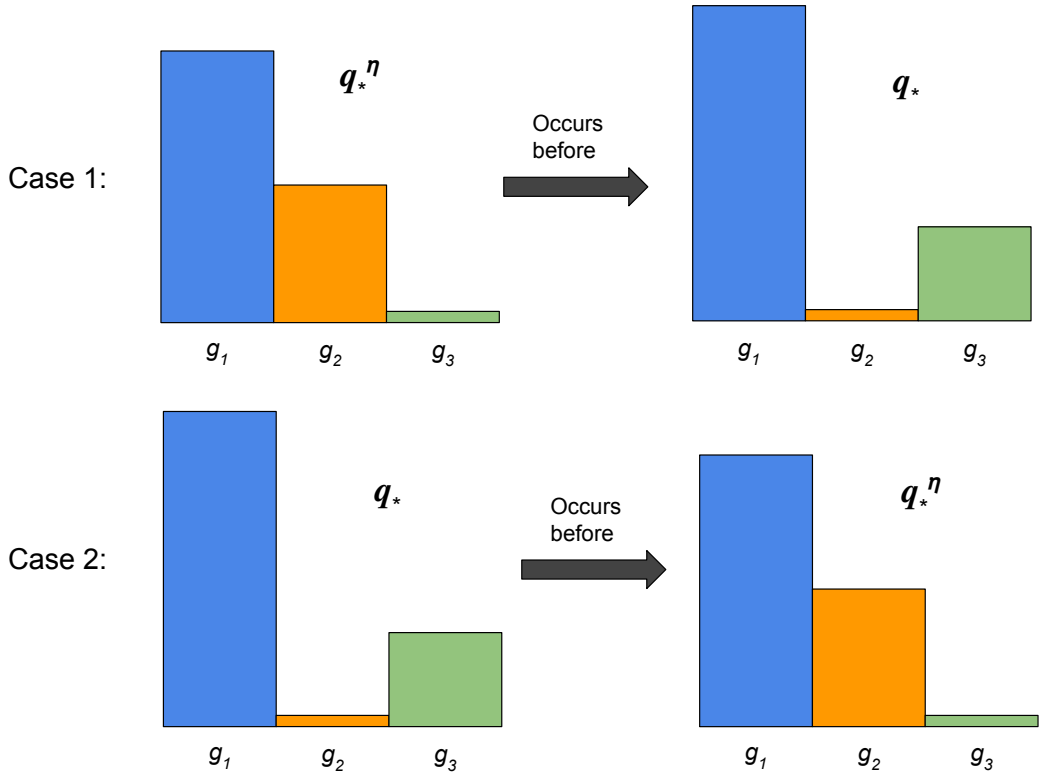


Figure 6.7: The two possibilities assume that the minimiser of the noisy risk does not minimise the clean risk. Case 1: The minimiser of the noisy risk occurs earlier in training (top row). Case 2: The minimiser of the noisy risk occurs later in training (bottom row).

Experiment Details Our experiment uses a noised, ternary version of the MNIST dataset. We remove all classes other than $\{0,1,2\}$, and we then apply synthetic asymmetric label noise to the training set according to the following transition matrix (setting $\eta = 0.3$)

$$\begin{bmatrix} 1-1.5\eta & 0.5\eta & \eta \\ \eta & 1-1.5\eta & 0.5\eta \\ 0.5\eta & \eta & 1-1.5\eta. \end{bmatrix}$$

We train a neural network classifier on this noisy dataset for 100 epochs. After each epoch, we examine the model's predictions on the held-out test set. At each datapoint in the test set, we record whether the model predicts the most likely noisy label, second most likely, etc. We can do this since we have access to the underlying noise model. We record the proportions and display how this evolves during training, shown in Figure 6.8.

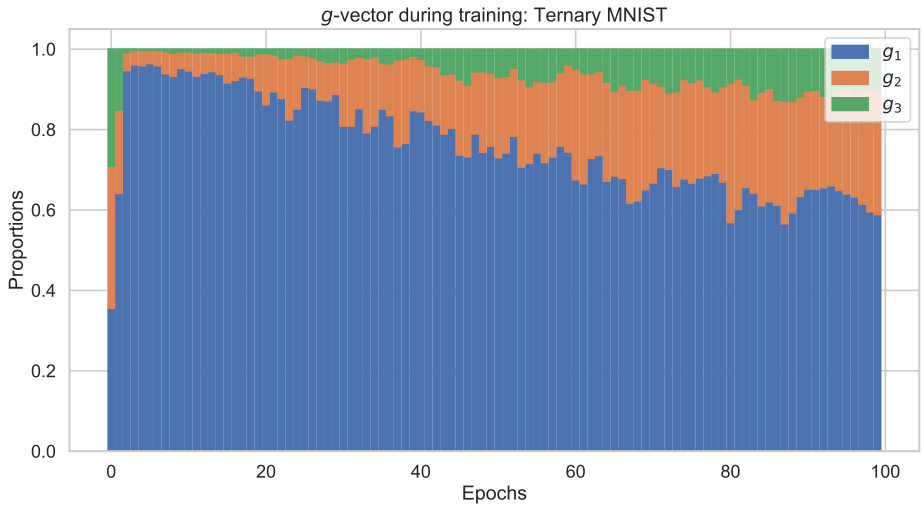


Figure 6.8: We train a neural network classifier on an asymmetrically noised ternary MNIST dataset. At each epoch, we evaluate on a held-out dataset of samples. Since we can access the noisy conditional class distributions, we can compute the $g-$vectors for the classifier at each epoch. We plot these as a stacked bar chart.

Figure 6.8 shows that the proportion by which our model predicts the second or third most likely noisy class decreases for the first six epochs or so. At around epoch 6, the proportion by which our model predicts the second or third most likely noisy class begins to increase. Crucially, both attain their minima almost simultaneously, satisfying the condition in Lemma 6.6.1. This can be seen in more detail in Figure 6.9, which shows g_2 and g_3 only during training. Both attain their minima almost simultaneously as epochs $5, 6$, respectively.

CIFAR The experiment above is repeated for a five-class asymmetrically noised version of the CIFAR dataset. The transition matrix used to construct the label noise is given in Equation 6.5. A plot of the $g-$vectors during training and scatter plots of g_1, g_2, g_3, g_4 can be seen in Figure 6.11 and Figure 6.10 respectively. Once again the g_i attain their minima around the same time.

$$T := \begin{bmatrix} 0.5 & 0.05 & 0.1 & 0.15 & 0.2 \\ 0.2 & 0.5 & 0.05 & 0.1 & 0.15 \\ 0.15 & 0.2 & 0.5 & 0.05 & 0.1 \\ 0.1 & 0.15 & 0.2 & 0.5 & 0.05 \\ 0.05 & 0.1 & 0.15 & 0.2 & 0.5 \end{bmatrix} \tag{6.5}$$

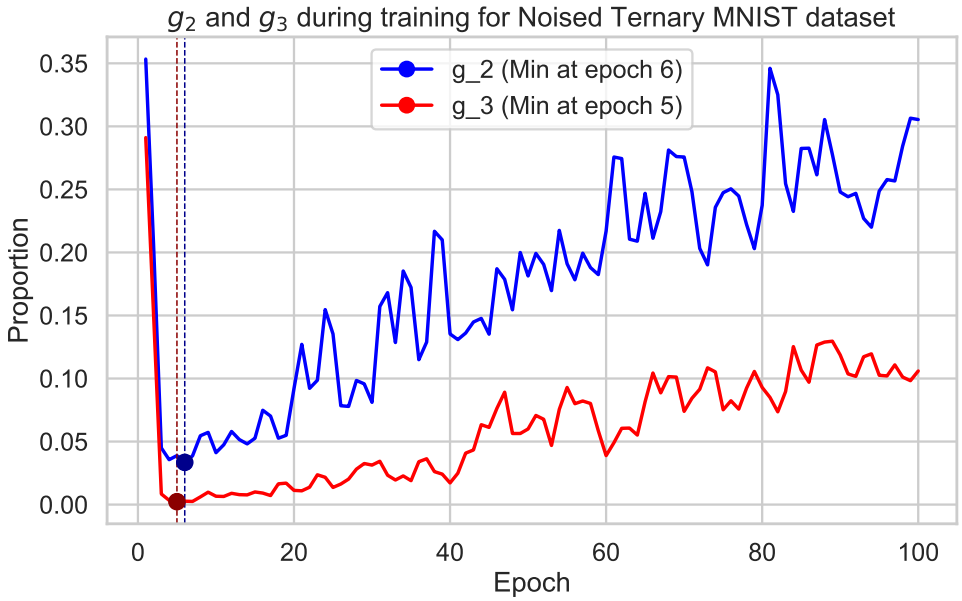


Figure 6.9: We train a neural network classifier on an asymmetrically noised ternary MNIST dataset. At each epoch, we evaluate on a held-out dataset of samples. Since we can access the noisy conditional class distributions, we can compute the $g-$vectors for the classifier at each epoch. We plot the g_2 (blue) and g_3 (red) during training and note when each attains its minima. Each minima occurs almost simultaneously (one epoch difference).

6.7 Conclusion

6.7.1 Chapter Objectives

This chapter examined the relationship between a classifier's noisy and clean risk during training when a dataset is corrupted by label noise. Our primary goal was to investigate, empirically and theoretically, whether noisy accuracy can be used as an effective criterion for Early Stopping.

What We've Shown In Section 6.4 we gave some theoretical insights regarding NES. We showed that NES should be effective when label noise is uniform and symmetric. We proved that these strong guarantees enjoyed by uniform symmetric noise do not apply to all other noise types. Consequently, our theoretical results implied that NES might not be effective unless label noise was uniform and symmetric. Unexpectedly, Section 6.5 showed empirically that using noisy accuracy as an Early-Stopping criterion is highly

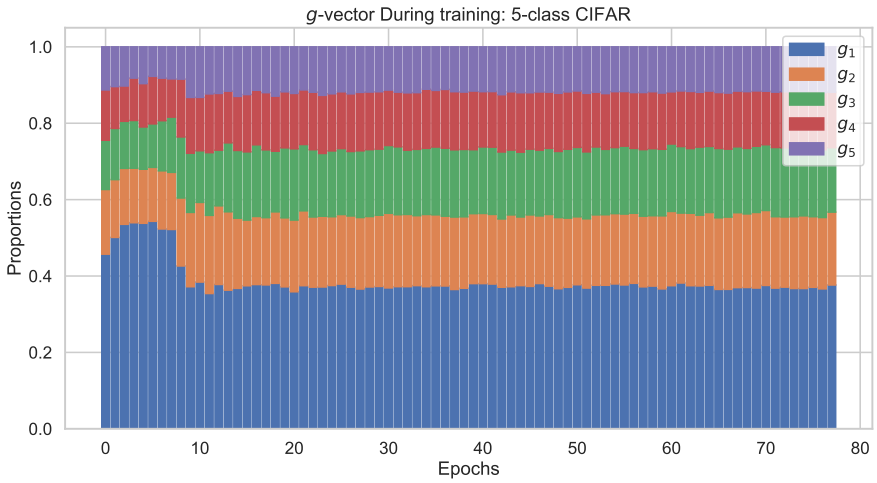


Figure 6.10: We train a neural network classifier on an asymmetrically noised 5-class CIFAR dataset. At each epoch, we evaluate on a held-out dataset of samples. Since we can access the noisy conditional class distributions, we can compute the $g-$vectors for the classifier at each epoch. We plot these as a stacked bar chart.

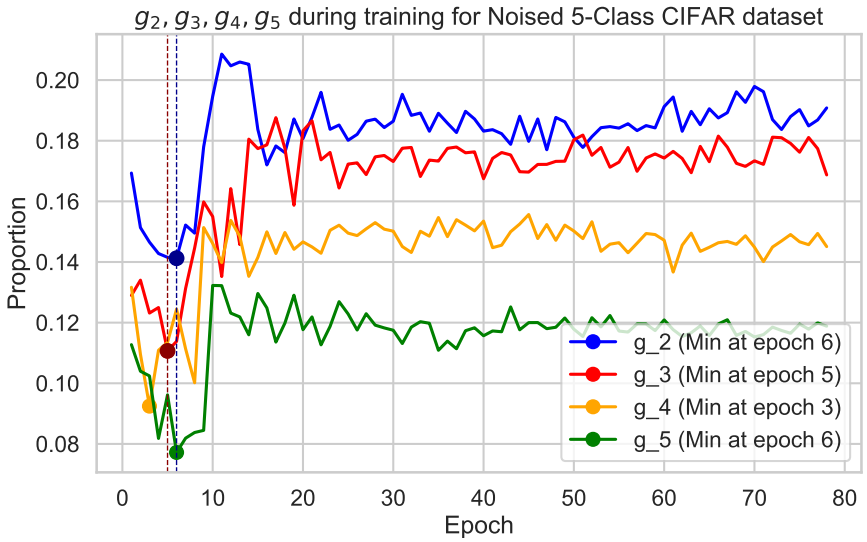


Figure 6.11: As in Figure 6.9, we train a neural network classifier on an asymmetrically noised 5-class CIFAR dataset. At each epoch, we evaluate on a held-out dataset of samples. Since we can access the noisy conditional class distributions, we can compute the $g-$vectors for the classifier at each epoch. We plot g_2 (blue), g_3 (red), g_4 (yellow) and g_5 (green) during training and note when each attains its minima. Each minima occurs almost simultaneously.

effective, performing comparably with clean Early Stopping across datasets and noise types. This finding is interesting and useful as it potentially allows practitioners to apply near-optimal Early Stopping even when clean data is unavailable. Nevertheless, the unexpectedness of these findings prompted us to re-investigate why NES is effective in settings where we initially expected it might fail. We partially explained this in Section 6.6 under the assumption of separability. To oversimplify dramatically, we showed that if overfitting occurs simultaneously between noisy classes, then NES will be effective. We then empirically showed that neural networks do overfit this way. This result applies to noise model beyond uniform symmetric label noise for which we already had good guarantees.

6.7.2 Limitations and Future Directions

While our finding that NES is effective is useful and interesting, there are several limitations we enumerate in this section, along with possible research directions our study opens up.

6.7.2.1 Future Work

The Class-Preserving Assumption Throughout this work, we assume that the label noise we are dealing with is class-preserving. As shown by Figure 6.5, when this condition is violated, NES and ES begin to diverge. As we have tried to emphasise in this chapter and Chapter 4, this noise condition is broad; most label noise studied in the literature is of this variety. Nevertheless, rare instances of non-class-preserving label noise have been studied in prior work (Patrini et al., 2017). Can Noisy Early Stopping be extended to these settings? An appeal of NES is that it can be used even when the exact noise model is unknown. This agnosticism mandates that we make some limiting assumption about the set of admissible noise models (Wolpert & Macready, 1997). Consequently, to extend NES to the non-class-preserving setting it becomes necessary to know something about the structure of the noise model. Although the non-class-preserving setting is outside the scope of this work, we speculate that if one has a Fisher consistent loss function (e.g. CE) and corrects the loss $(L \mapsto L_F)$ in such a way that a global minimiser of the noisy L_F-risk is Bayes-optimal for the clean distribution then NES will be effective when training using L_F. Nevertheless, this is speculation and should be explored in future work.

Closed-Set We also reiterate that these findings apply to closed-set label noise - label noise where the true label of each sample lies within the given label set. Open-set label noise is outside the scope of this work and should be investigated in further studies.

Separability A critical direction for future work is studying whether the results in Section 6.6 can be extended to non-separable data distributions. In Appendix D.1.5, we give an intuitive discussion of why this should be possible, but this requires developing further.

Overfitting It is important additionally to develop a theoretical understanding and justification for why the components g_i discussion in Section 6.6.2 attain their minima (approximately) simultaneously when training a neural network classifier on noisy data. This would allow a more comprehensive explanation of the efficacy of NES.

Revaluation of Robust Loss Functions Finally, we feel that it would be useful to comprehensively compare existing robust loss functions across the standard noisy dataset benchmarks now that we have an effective method of Early Stopping that works across loss functions.

6.7.2.2 Limitations

Data Scarcity In a setting where noisy data is scarce, separating some of this data into a validation set may not be practical as this could leave an impractically small training set. Our results apply in settings with ample noisy data to form a validation set while retaining a sufficiently large training set.

Double Descent Previous research has shown that when labels contain noise, an epochwise double descent phenomenon can manifest Stephenson and Lee (2021)(Nakkiran et al., 2021). While the clean test error rises after a certain number of epochs, it proceeds to dip again if training continues beyond this point. Simultaneously the clean test accuracy undergoes a second increase. When the noise rate is low $\approx 5\%$, this second increase in test accuracy can be larger than the first peak. A consequence is that Early Stopping with a small patience parameter would stop training before this second larger peak is obtained. This is a problem of Early Stopping more broadly rather than Noisy Early Stopping specifically, nevertheless, it is important to be cognisant of this double descent phenomenon in the low noise rate setting and to select a suitably large patience parameter.

Early Stopping In our theory section (Section 6.4), we give conditions under which the minimiser of the noisy risk will minimise the clean risk, arguing that under these conditions, noisy Early Stopping will be effective. This relies on the assumption that we can precisely estimate the noisy risk at each training epoch. In practice, we estimate the noisy risk by computing the misclassification rate on a held-out noisy validation set, which remains constant through training. Since we are using a finite test set there will be some uncertainty in this noisy risk approximation. Moreover, since we use the *same* validation set during training, this introduces covariances between noisy risk estimations for each epoch. If the validation set is sufficiently large, these variances/covariances are small. However, the variances and covariances will be non-trivial when the validation set is insufficiently large. In this regime, selecting the model with the lowest estimated noisy risk may not correspond to selecting the model with the lowest noisy risk within $\mathcal{Q}$.

Peak Test Accuracy During Training Noisy Early Stopping allow us to cease training at or near the point where clean test accuracy peaks. The effectiveness of NES, therefore, depends on the height of this peak; in particular, if the classifier being learned does not generalise well at any epoch during training, then NES cannot alter this fact. Thus, while NES allows us to get the best performance out of a particular method, it is limited by the performance of the given method. As established in the relevant literature, different loss functions attain different peak test accuracies during training in the presence of noisy labels (Janocha & Czarnecki, 2016; Z. Zhang & Sabuncu, 2018). Since our work provides a way to obtain this peak model, it reinforces the importance of continuing to develop and improve robust loss functions. We believe that the findings of this study should redirect future efforts from developing loss functions that prevent overfitting to those that enhance the peak performance attained during training.

Chapter 7

Noise-Tolerant Loss Functions

7.1 Noise-Tolerant Loss Functions

In Section 2.3.2.2, we discussed a set of approaches referred to as *Noise-Tolerant loss functions* (Ghosh & Kumar, 2017). These approaches aim to define a loss function that is intrinsically robust to label noise. By this we mean that label noise does not necessitate any correction to the loss function; training with a Noise-Tolerant loss results in good generalisation despite noisy training labels.

This chapter explores Noise-Tolerant loss functions building on the earlier work of Ghosh and Kumar (2017) and comprises two main contributions. The first contribution is deriving a necessary and sufficient condition for a loss function to be Noise Tolerant - we call this the **Noise-Tolerance Theorem**. This theorem relates the noise transition matrix's eigenspaces with a loss function's codomain. The second contribution is finding all possible label noise models for which a Noise-Tolerant loss function can exist. This contribution comes at the end of the chapter and requires some groundwork to establish. The chapter outline is as follows.

Outline We begin by formally defining Noise Tolerance and stating and proving the Noise-Tolerance theorem linking Noise-Tolerant loss functions to eigenspaces of the noise transition matrix. We look at the implications of this theorem in the case of symmetric label noise, deriving a simple condition, which, when satisfied by loss function, imbues Noise Tolerance. We then explore the extent to which Fisher consistency is compatibility with Noise Tolerance. We show that, in general, a loss function cannot be Noise Tolerant and Fisher consistent. This finding prompts us

to define a weakened notion of Fisher consistency, demonstrating that this relaxed definition *is* compatible with Noise Tolerance, using binary label as a case study. In Section 7.3, we conjecture that for a loss function to be non-degenerate, its codomain must have dimension no less than $c-1$. Using this, we characterise all label noise models for which Noise-Tolerant loss functions can exist. All proof statements, where omitted, mat be found in Appendix D.

7.1.0.1 Noise Tolerance: Related Work

The term 'Noise Tolerance' first appears in the work of Manwani and Sastry (2013), defined as a loss for which the minimisers of the noisy and clean risks over the relevant model family induce identical plug-in classifiers. In this work, the authors focus on the binary label setting, showing that the 0-1 loss is Noise Tolerant under uniform symmetric label noise. They demonstrate that when the model family is restricted to linear classifiers, the mean squared error (MSE) is Noise Tolerant, suggesting that MSE could be robust to this type of noise even when using a linear classifier on a pre-learned featurisation. This finding sheds some light on the later findings of (Janocha & Czarnecki, 2016). Additionally, they show that the Fisher linear discriminant is Noise Tolerant to uniform symmetric label noise. In contrast, the hinge, log, and exponential losses are not Noise Tolerant to this label noise type.

The work of Manwani and Sastry (2013) is extended in a follow-up paper by Ghosh et al. (2015). This paper retains a focus on binary labels but generalises the results of the earlier work. They show that if the sum of the loss across labels is constant and independent of the forecast, then the loss will be Noise Tolerance to symmetric label noise. This condition applies to the probit and ramp loss functions, implying their Noise Tolerance to symmetric label noise. Under what is equivalent to a class-preserving noise assumption, they extend this to non-uniform label noise, assuming separability of the data distribution.

The final paper of this trio on Noise-Tolerant loss functions is Ghosh and Kumar (2017). The main contribution of this work is the generalisation of the results of Ghosh et al. (2015) to the multi-class setting. They demonstrate that loss functions which satisfy a 'symmetry' property are Noise Tolerant to symmetric label noise. This applies to the 0-1 and mean absolute error (MAE) loss functions, thus generalising the results of Manwani and Sastry (2013).Van Rooyen et al. (2015) builds on these insights, defining

the 'unhinged loss' and showing that it is the only possible convex loss that satisfies the symmetric condition. Charoenphakdee et al. (2019) as well as A. Menon et al. (2015) demonstrate the Noise Tolerance of Balanced Error Rate (BER) and Area Under the Curve (AUC) to label noise. The results of Ghosh and Kumar (2017) are partly extended by X. Zhou et al. (2021) who, derive a Noise-Tolerance condition for some asymmetric noise models.

7.1.1 Noise Tolerance and Eigenfunctions

In the label noise literature, the term 'Noise Tolerance' encompasses several related but subtly distinct definitions. For this chapter, we define Noise Tolerance as follows.

Definition 7.1.1. *We say that a loss function L is* Noise Tolerant *to some label noise model if there exists an increasing function*[1] f *such that, for any data-label distribution $p(x,y)$ and estimator $\boldsymbol{q}$ the noisy and clean risks satisfy the following relation;*

$$R_L^{\eta}(\boldsymbol{q}) = f(R_L(\boldsymbol{q})). \tag{7.1}$$

This says that the noisy and clean risks of an estimator are related by an order-preserving transformation. Thus, in particular, for any set of probability estimators $\mathcal{Q}$, the minimiser of the noisy risk within $\mathcal{Q}$ always coincides with the clean risk minimiser in $\mathcal{Q}$;

$$\arg\min_{q\in\mathcal{Q}} R_L^{\eta}(\boldsymbol{q}) = \arg\min_{q\in\mathcal{Q}} R_L(\boldsymbol{q})$$

Noise-Tolerance Theorem for Class-Conditional Noise The following Theorem 7.1.2 gives a functional equation which must be satisfied by all the Noise-Tolerant loss functions for class-conditional label noise, demonstrating the strong relationship between the eigenfunctions of T^T and the Noise-Tolerant loss functions with respect to T.

Theorem 7.1.2 (Noise-Tolerance Theorem)**.** *Let T be a stochastic matrix describing class-conditional label noise. A loss function L is Noise Tolerant to T if and only if there exists $\lambda > 0$ and $\boldsymbol{c} \in \mathbb{R}^c$ such that for all $\boldsymbol{q} \in \Delta$,*

$$T^T \boldsymbol{L}(\boldsymbol{q}) = \lambda \boldsymbol{L}(\boldsymbol{q}) + \boldsymbol{c}. \tag{7.2}$$

[1] defined on $Dom(R(\boldsymbol{q}))$; the domain of the risk function, taken over all estimators and distributions.

Eigenspaces Theorem 7.1.2 gives us a precise condition under which a loss function L is Noise Tolerant (as defined in Definition 7.1.1) with respect to class-conditional label noise with transition matrix T. With reference to Equation 7.2, we can see that this condition strongly resembles the definition of an eigenvector. Indeed, Noise-Tolerant loss functions are in precise correspondence with the eigenspaces of the matrix T^T. We formalise this correspondence in the following Corollary 7.1.3.

Corollary 7.1.3. *Let $E_\lambda(T^T)$ denote the eigenspace of the matrix T^T associated with eigenvalue λ. That is*

$$E_\lambda(T^T) \coloneqq \{\boldsymbol{v} \in \mathbb{R}^c \colon T^T\boldsymbol{v} = \lambda\boldsymbol{v}\}$$

L is Noise Tolerant with respect to T if and only if for some (positive) eigenvalue λ of T^T, there exists vector $\boldsymbol{k} \in \mathbb{R}^c$ so that

$$\boldsymbol{L}(\Delta) \coloneqq \{\boldsymbol{L}(\boldsymbol{q}) \colon \boldsymbol{q} \in \Delta\} \subseteq E_\lambda(T^T) + \boldsymbol{k}.$$

Corollary 7.1.3 states that a necessary and sufficient condition for Noise Tolerance is that the image of the simplex Δ under the loss function, $\boldsymbol{L}(\Delta)$ be wholly contained in a translate of one of the eigenspaces of the matrix T^T. In the following section, we unpack this by looking at the example where matrix T describes symmetric label noise at rate η.

7.1.2 Symmetric Label Noise

Let T describe symmetric label-noise at rate η. That is;

$$T \coloneqq \begin{bmatrix} 1-\eta & \frac{\eta}{c-1} & \cdots & \frac{\eta}{c-1} \\ \frac{\eta}{c-1} & 1-\eta & \cdots & \frac{\eta}{c-1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\eta}{c-1} & \frac{\eta}{c-1} & \cdots & 1-\eta \end{bmatrix} \tag{7.3}$$

Corollary 7.1.3 tells us that a loss is Noise Tolerant when the image of the simplex under our loss is contained in one of the translates of an eigenspace of T^T. One can easily show that T^T has two eigenspaces corresponding eigenvalues, $\lambda = 1$ and $\lambda = 1 - \frac{c\eta}{c-1}$. The former corresponds to an eigenspace of dimension one spanned by the vector $(1,1,\ldots,1)$. (Indeed, since T is a column stochastic matrix, this is always an eigenvector of T^T). The second eigenvalue corresponds to an eigenspace of dimension $c-1$. We now ask what must be true of a loss function L for $\boldsymbol{L}(\Delta)$ to lie in one of the translates of these eigenspaces. We start by considering the $\lambda = 1$ eigenspace.

Case 1: $\lambda = 1$ Suppose that there exists $\boldsymbol{u} \in \mathbb{R}^c$ and some function $\alpha(\boldsymbol{q})$ so that for all $\boldsymbol{q} \in \Delta$,

$$(L(\boldsymbol{q},1),\ldots,L(\boldsymbol{q},c)) = \alpha(\boldsymbol{q})(1,1,\ldots,1) + \boldsymbol{u}.$$

This implies that, for each $i \neq j$ we have $L(\boldsymbol{q},i) = L(\boldsymbol{q},j) + const$. One can show that there are no non-degenerate loss functions consistent with this condition. Specifically, given $\boldsymbol{p} \in \Delta$, the pointwise risk of a forecast $\boldsymbol{q} \in \Delta$ may be written

$$\boldsymbol{p} \cdot \boldsymbol{L}(\boldsymbol{q}) = L(\boldsymbol{q},1) + const.$$

Therefore, the pointwise risk is minimised by the same $\boldsymbol{q}^\star \in \Delta$ for all $\boldsymbol{p} \in \Delta$, meaning that this is useless as loss function.

Case 2: $\lambda = 1 - \frac{c\eta}{c-1}$ Let us consider the second eigenspace. Since T (Equation 7.3) is a symmetric matrix ($T^T = T$), then it follows from a standard result from linear algebra that the eigenvectors are orthogonal. Thus one may characterise the eigenspace for $\lambda = 1 - \frac{c\eta}{c-1}$ via the relation $\boldsymbol{v} \cdot (1,1,1,\ldots,1) = 0$. Thus, in order to satisfy the condition of Corollary 7.1.3 we require that for all $\boldsymbol{q} \in \Delta$,

$$\begin{aligned} &\boldsymbol{L}(\boldsymbol{q}) \cdot (1,1,1,\ldots,1) = const. \\ \iff &\sum_{k=1}^{c} L(\boldsymbol{q},k) = const. \end{aligned} \tag{7.4}$$

Where $\eta < \frac{c-1}{c}$ to ensure that $\lambda > 0$. Unlike case 1, non-trivial loss functions exist that meet this criterion. Most notably, we have MAE

Example: $\sum_{k=1}^{c} L_{MAE}(\boldsymbol{q},k) = \sum_{k=1}^{c} ||\boldsymbol{e}_k - \boldsymbol{q}||_1 = c.$

Normalised Losses This exact condition is derived in (Ghosh & Kumar, 2017). Here, they look for functions which satisfy the weaker condition that the *global* minimiser of the noisy and clean L-risks are identical. They show that, for symmetric label noise, a loss function satisfying Equation 7.4 achieves this objective. In (Ma et al., 2020), they show that loss functions can be 'normalised' via

$$L(\boldsymbol{q},k) \mapsto \frac{L(\boldsymbol{q},k)}{\sum_{i=1}^{c} L(\boldsymbol{q},i)}$$

to achieve this condition.

All Noise Rates A remarkable property is that the condition derived in Equation 7.4 is that it has no dependence on the noise rate η. For all choices of η, we obtain the same eigenspace and, hence, the same condition on our loss function. In order to satisfy the non-negativity conditions of Corollary 7.1.3 we require that $\lambda = 1 - \frac{c\eta}{c-1} > 0$ which occurs precisely when $\eta \in [0, \frac{c-1}{c})$. Beyond this, *any loss function* satisfying Equation 7.4 is Noise Tolerant to *all symmetric noise at rates less than* $\frac{c-1}{c}$ *- without any requirement to know or utilise the noise rate*. (This threshold corresponds precisely to all class-preserving symmetric label noise).

7.2 Fisher Consistency

Theorem 7.1.2 establishes the necessary and sufficient conditions for a loss function to be Noise Tolerant to class-conditional label noise. The primary focus of the remainder of this chapter is to examine under what circumstances these conditions facilitate the creation of meaningful loss functions. Here, the term 'meaningful' denotes that a function must meet specific criteria to be viable as a loss function; merely satisfying the Noise-Tolerance condition is insufficient. For example, the constant function $\boldsymbol{L}(\boldsymbol{q}) = \boldsymbol{c}$ satisfies the necessary condition but is useless as a loss function as it scores all predictions equally. As discussed previously, among the criteria we desire of a loss, Fisher consistency is arguably the most critical, describing loss functions which induce Bayes-optimal decisions when optimised. This section explores the compatibility of Fisher consistency with Noise Tolerance. In Section 7.2.0.1, we show no Fisher consistent loss functions exist for asymmetric label noise. However, in Section 7.2.1, we show that a slight relaxation of Fisher consistency allows for the existence of Noise-Tolerant losses.

7.2.0.1 Fisher Consistency and Noise Tolerance

We begin by providing some intuition as to why Fisher consistency and Noise Tolerance may generally be at odds with one another.

Intuition If a loss function L is Noise Tolerant to a given noise model, then, by definition, minimising the noisy risk yields a minimiser of the clean risk. However, if the label noise model is not class-preserving for a given distribution $p(x,y)$, then for some $x_0 \sim p(x)$, $p(y \mid x_0)$ and $\widetilde{p}(\widetilde{y} \mid x_0)$ have different dominant classes. It follows that there exists some $\boldsymbol{p} \in \Delta$ for which the minimiser of the expected loss $\boldsymbol{q}^* \in \arg\min_{\boldsymbol{q} \in \Delta} H_L(\boldsymbol{p}, \boldsymbol{q})$

does not satisfy $\arg\max_i \boldsymbol{q}_i^* = \arg\max_i \boldsymbol{p}_i$, violating the definition of Fisher consistency. This suggests no loss function can be Noise Tolerant and Fisher consistent unless the noise model is class-preserving for all distributions.

The following lemma formally establishes that if a label noise model is not universally class-preserving (as defined in Section 4.3.0.1) then there are no Fisher consistent loss functions for this noise model.

Lemma 7.2.1. *Suppose that the transition matrix T satisfies the property that $\exists \boldsymbol{p} \in \Delta$ such that $\arg\max_i (T\boldsymbol{p})_i \neq \arg\max_i \boldsymbol{p}_i$. Then, there are no Fisher Consistent loss functions which are Noise Tolerant with respect to T.*

Corollary 7.2.2. *Let T be a transition matrix. Unless T describes symmetric noise, there are no Fisher-consistent Noise-Tolerant loss functions with respect to T.*

Proof. Lemma 4.3.2 established that the only class-conditional label noise model which is class-preserving for all distributions is symmetric label noise where $\eta < \frac{c-1}{c}$. For any other class-conditional noise model there exists $\boldsymbol{p} \in \Delta$ for which $\arg\max_i (T\boldsymbol{p})_i \neq \arg\max_i \boldsymbol{p}_i$. Our Corollary then follows directly from Lemma 7.2.1. □

Discussion Corollary 7.2.2 tell us that, aside from the case of symmetric label noise, we cannot have any Noise-Tolerant losses which are Fisher consistent. If L is Noise Tolerant to some asymmetric label noise, there exists a distribution $p(x,y)$ where the minimiser of the L-risk is not Bayes-optimal. This finding is somewhat disappointing, albeit arguably unsurprising. In Section 7.1.2, we saw that we can obtain Noise Tolerance to symmetric label noise by imposing a simple condition on the loss function. We hoped to find similar conditions for other noise models; however, Corollary 7.2.2 tells us that this cannot be achieved with the same generality.

7.2.1 Partial Consistency

Is Fisher Consistency Necessary? In the absence of knowledge about the data distribution $p(x,y)$ Fisher consistency ensures that, for any $p(x,y)$, minimising the L-risk will induce Bayes-optimal decisions. However, when we have prior information about the data distribution, Fisher consistency is stronger than strictly necessary. For example, suppose that know $p(x,y)$ is separable. Then, for each $x \sim p(x)$, the conditional class distribution $p(y \mid x)$ always lies on one of the vertices of the probability simplex.

Therefore, we only need Fisher consistency at these vertices, rather than the entire simplex. Definition 7.2.3 formally defines a relaxed notion of Fisher consistency. Through an analysis of class-conditional label noise for binary labels, we demonstrate that this weakened Fisher consistency is compatible with the existence of Noise-Tolerant loss functions for asymmetric label noise.

Definition 7.2.3. *Let $A \subset \Delta$ be a subset of the probability simplex. We say a loss function L is A- Fisher consistent if, for all $\boldsymbol{p} \in A$ a minimiser of the pointwise risk*

$$\boldsymbol{q}^* \in \arg\min_{\boldsymbol{q}\in\Delta} H_L(\boldsymbol{p},\boldsymbol{q}) \implies \arg\max_i \boldsymbol{q}^*_i = \arg\max_i \boldsymbol{p}_i.$$

I.e., so long as $\boldsymbol{p} \in A$, the loss function induces a Bayes-optimal decision by minimisation of the pointwise risk over Δ. Notice that when $A = \Delta$ we recover the definition of a Fisher consistent loss function from Definition 2.1.1

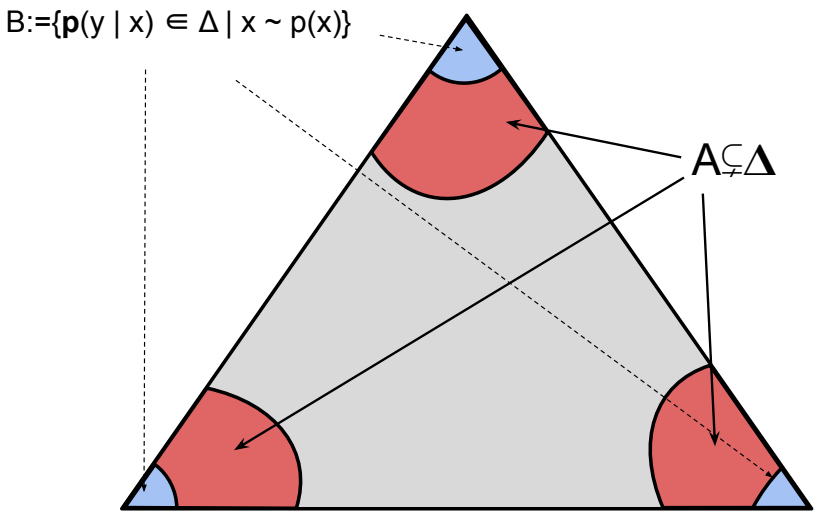


Figure 7.1: An image of the simplex (for three classes) with a region $A \subsetneq \Delta$ shaded in red. For a data distribution $p(x,y)$, the set of all conditional class distributions; $B := \{\boldsymbol{p}(y \mid x) \in \Delta \mid x \in \text{supp}(p(x))\}$ is shaded in light blue. If a loss function L is $A-$Fisher consistent, then it would induce Bayes-optimal decisions when optimised on this distribution since the conditional class distributions (light blue) of our data distribution lie within the red-shaded regions, $B \subset A$.

In settings where we can bound the domain of the true class distribution, we can use this weaker notion of consistency to ensure that the minimiser of the L-risk induces a Bayes-optimal classifier. An example is illustrated in Figure 7.1.

7.2.2 The Binary Case

Consider the case of binary classification $c = 2$. An arbitrary transition matrix may be written as follows where $a, b \in [0,1]$

$$T = \begin{bmatrix} 1-a & b \\ a & 1-b \end{bmatrix}. \tag{7.5}$$

Our objective in this section is to find Noise-Tolerant loss functions for this label noise and derive the sets A upon which we have A-Fisher consistency. With reference to Theorem 7.1.2 we must begin by finding the eigenspaces of T^T. T^T has eigenvectors $\boldsymbol{v}_1 = (1,1)$ and $\boldsymbol{v}_2 = (-a,b)$ (with eigenvalues $\lambda = 1, \lambda = 1-a-b$ respectively) which span the two eigenspaces of T^T. We know that in order for a loss function L to be Noise Tolerant to T, the span of $\boldsymbol{L}(\boldsymbol{q})$ must lie in the translates of one of the eigenspaces of T^T. Equivalently, there must exist a function $\alpha(\boldsymbol{q})$ and constants c_1, c_2 such that either

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{q}) &= \alpha(\boldsymbol{q})(-a,b) + (c_1,c_2), \\ \text{or} \quad \boldsymbol{L}(\boldsymbol{q}) &= \alpha(\boldsymbol{q})(1,1) + (c_1,c_2). \end{aligned} \tag{7.6}$$

As in Section 7.1.2, there are no non-trivial loss functions consistent with Equation 7.6. Hence a loss function is Noise Tolerant to T iff $\boldsymbol{L}(\boldsymbol{q}) = \alpha(\boldsymbol{q})(-a,b) + (c_1,c_2)$, which implies that $bL(\boldsymbol{q},1) + aL(\boldsymbol{q},2) = const$. Equivalently, letting $\alpha := \frac{a}{b}$

$$L(\boldsymbol{q},1) + \alpha L(\boldsymbol{q},2) = const.$$

Conversely, given a loss function L, suppose that there exists constants $\alpha > 0$ and $c \in \mathbb{R}$ such that

$$L(\boldsymbol{q},1) + \alpha L(\boldsymbol{q},2) = c. \tag{7.7}$$

Then L is Noise Tolerant to all class-conditional label noise where $\alpha = \frac{a}{b}$ and $a+b < 1$ (this is necessary so that the eigenvalue is greater than zero).

Remark In the case where $a = b$, T describes symmetric label noise and one recovers the loss condition given in Lemma 7.4.

What About Consistency? A loss function satisfying Equation 7.7 is Noise Tolerant to T. However, L must satisfy additionally some notion of Fisher consistency. For $a \neq b$, we cannot have full Fisher consistency by Corollary 7.2.2, and we must be content with the notion of partial consistency introduced in Definition 7.2.3. The Lemma below derives the maximal set upon which L can be $A-$Fisher consistent.

Lemma 7.2.4. *Let L be a $A-$Fisher consistent loss function which is Noise Tolerant to class-conditional label noise described as by the transition matrix in Equation 7.5 (without loss of generality letting $b \geq a$). Then*

$$A \subseteq \left[0, \frac{1}{2}\right] \cup \left[\frac{1}{\alpha+1}, 1\right], \tag{7.8}$$

where $\alpha := \frac{a}{b}$. Note that when $a = b$, this corresponds to the interval $[0,1]$ since fully Fisher consistent loss Noise-Tolerant loss functions do exist for symmetric label noise.

Lemma 7.2.4 gives the maximal possible set $A \subseteq \Delta$ for which a loss function, Noise-Tolerant to Equation 7.5, can be $A-$Fisher consistent. The following example demonstrates that this maximal consistency may be obtained.

Example Let $b > a$ so that $1-a-b > 0$. Define the following loss function L;

$$L(q,1) = 1-q, \quad L(q,2) = \frac{b}{a}q.$$

L is Noise Tolerant with respect to label noise with transition matrix in Equation 7.5. Moreover, L is $A-$Fisher consistent where A is as defined on the right-hand side of Equation 7.8.

7.2.2.1 Summary For Binary Labels

We have shown that for class-conditional noise described by the matrix in Equation 7.5, one may construct Noise-Tolerant loss functions when $1-a-b > 0$. These are precisely the loss functions where $aL(q,1) + bL(q,2) = const$. We require these losses to induce Bayes-optimal decisions when optimised. We showed that this imposes the restriction that, for each $x \sim p(x)$, the true probability $p := p(y = 1 \mid x)$ lies within the set A defined in Equation 7.8.

Remark The Noise-Tolerance condition in Equation 7.7 and the set defined in Equation 7.8 depend only on the ratio $\alpha := \frac{a}{b}$. Consequently, a loss function satisfying $L(q,1) + \alpha L(q,2) = const$ will be Noise Tolerant to *all* label noise models such that $\frac{a}{b} = \alpha$ (assuming also that $1 > a+b$). This is remarkable as it allows one to build loss functions which are Noise Tolerant to an entire family of noise models. Moreover, to construct a Noise-Tolerant loss function, one only needs to know the ratio $\frac{a}{b}$. In contrast, the forward correction requires specifying the exact noise model. Below, we give a concrete example of how one might use a Noise-Tolerant loss function.

Example Application Consider a setting in which we have a (large) noisily-labelled binary-label dataset. While the exact noise transition matrix is unknown, we anticipate that class 2 is mislabelled as class 1 twice as often as class 1 is mislabelled as class 2. (i.e the label noise transition matrix in Equation 7.5 satisfies $\frac{a}{b} = \frac{1}{2}$). Suppose in this setting that there is very little uncertainty in the class distributions, meaning that each data sample can unambiguously be associated with some true label (i.e. for each x, either $p(y = 1 \mid x) \approx 0$ or $p(y = 1 \mid x) \approx 1$). We define the loss function

$$L(q,1) = 1 - q,$$
$$L(q,2) = 2q,$$

and train a classifier on the noisy data using this loss. Our results tell us that for *any* set of probability estimators Q the minimiser of the noisy risk in Q will also be a minimiser of the clean risk in Q. This holds regardless of the true values of a, b so long as $\frac{a}{b} = \frac{1}{2}$. If Q contains the global minima of the noisy risk, then this classifier will attain the minimal possible misclassification rate on the clean distribution.

7.2.3 Multiclass Settings

This section explores the more complex general case where $c > 2$. We conjecture that a non-degenerate Noise-Tolerant loss function requires the matrix T^T to have an eigenspace of dimension $c - 1$. We offer some rationale for this conjecture and use it to derive properties of Noise-Tolerant loss functions, ultimately defining the broadest form of label noise that permits their existence.

Non-Degenerate Loss Function A loss function L is termed '*non-degenerate*' if for at least one distribution $p(x,y)$, with balanced classes, any minimiser of the L-risk is Bayes-optimal for $p(x,y)$. An example of a degenerate loss would be the constant loss $\boldsymbol{L}(\boldsymbol{q}) = \boldsymbol{c}$.

Conjecture 7.2.5. *Let T be the transition matrix for some class-conditional label noise. For a non-degenerate loss function to exist, which is Noise Tolerant to T, matrix T^T must have an eigenspace of dimension $c - 1$.*

Intuition We are unable to formally prove the conjecture beyond $c = 3$ (See Appendix E.3.0.1); however, we can provide intuition for why it might hold more generally. The probability simplex Δ lies in a $(c-1)$-dimensional subspace. By Theorem 7.1.2, L is Noise Tolerant if and only if $\boldsymbol{L}(\Delta)$ is contained within a translate of one of the eigen-

spaces of T^T. Thus, if the dimension of this eigenspace is strictly less than $c-1$, then $\boldsymbol{L}$ necessarily maps the simplex into a lower-dimensional space. This would make L highly atypical; in practice, we are unaware of a single instance of a loss function for which $dim(\boldsymbol{L}(\Delta)) < c-1$. In particular, this condition would have two main consequences:

Consequence 1 For any forecast $\boldsymbol{q}$, the expected loss $H(\boldsymbol{p},\boldsymbol{q})$ would be the same for multiple $\boldsymbol{p} \in \Delta$. Specifically, there exists a vector $\boldsymbol{v}$ (where $\boldsymbol{v} \cdot \mathbf{1} = 0$) such that for any $\boldsymbol{p},\boldsymbol{q} \in \Delta$, $H(\boldsymbol{p}+\boldsymbol{v},\boldsymbol{q}) = H(\boldsymbol{p},\boldsymbol{q})$. This suggests that the expected loss incurred by a forecast is unaffected by changes in the underlying distribution.

Consequence 2 If L is differentiable, for each forecast $\boldsymbol{q}$, a perturbation $\boldsymbol{q} \mapsto \boldsymbol{q} + \delta\boldsymbol{q}$ can be made without altering the expected loss: $H(\boldsymbol{p},\boldsymbol{q}+\delta\boldsymbol{q}) = H(\boldsymbol{p},\boldsymbol{q})$. This characteristic makes L unsuitable for gradient-based optimisation.

Throughout the remainder of this chapter we assume the truth of Conjecture 7.2.5

7.2.3.1 Properness and Noise Tolerance

When learning a classifier by empirical risk minimisation, one often uses a proper loss function to ensure the model's predicted probabilities align with the true underlying probabilities. This section explores the compatibility of properness and Noise Tolerance. We conclude that Noise Tolerance precludes loss properness.

Improperness Recall that a loss is called (strictly) proper when the true probability $\boldsymbol{p}$ (uniquely) minimises the expected risk. We introduce the concept of 'improperness', describing a loss function where the expected loss is always minimised at the boundary of the simplex.

Definition 7.2.6. *A loss function $L : \Delta \times \mathcal{Y} \to \mathbb{R}$ is termed 'improper' if, except for at most one $\boldsymbol{p} \in \Delta$, the expected loss is uniquely minimised at the boundary of the simplex. Formally:*

$$\boldsymbol{q}^* \in \arg\min_{\boldsymbol{q}\in\Delta} H_L(\boldsymbol{p},\boldsymbol{q}) \implies \boldsymbol{q}^* \in \partial\Delta,$$

where $\partial\Delta$ denotes the boundary of the simplex. A loss is improper for $c = 2$ if the pointwise risk is minimised by setting $q = 0$ or $q = 1$, precluding properness.

Example MAE is improper.

Noise-Tolerant Losses are Usually Improper The Noise-Tolerant losses MAE and Normalised Cross-Entropy (Ma et al., 2020) are improper. This motivates the question whether Noise-Tolerant functions are always improper. While this question remains unresolved in the general case, Lemma 7.2.7 demonstrates its truth whenever $\boldsymbol{L}$ is continuous and injective and has a continuous inverse (also known as a *homeomorphism*), a common property for loss functions used in machine learning optimisation.

Homeomorphism A homeomorphism is a one-to-one continuous map whose inverse is also continuous.

Lemma 7.2.7. *Let $\boldsymbol{L} : \Delta \to \boldsymbol{L}(\Delta)$ be a homemorphism*[2]*. Then L is improper, which means that for all but one choice of $\boldsymbol{p} \in \Delta$, the expected loss $H(\boldsymbol{p}, \boldsymbol{q})$ is uniquely minimised by $\boldsymbol{q}$ in the boundary of the simplex; $\boldsymbol{q} \in \partial\Delta$.*

Although the conditions given in Lemma 7.2.7 hold for almost all frequently studied loss functions, they are probably stronger than necessary. We leave finding a more general version of this lemma for future work.

Noise-Tolerant Losses are not Strictly Proper Lemma 7.2.7 is strong but depends on the condition that $\boldsymbol{L}$ is continuous and injective. The following Lemma 7.2.8 demonstrates (without using these assumptions) the weaker claim that no Noise-Tolerant loss function can be strictly proper. Thus, obtaining Noise Tolerance must necessarily come at the cost of calibration.

Lemma 7.2.8. *There are no strictly proper Noise-Tolerant loss functions.*

7.3 When Can Noise-Tolerant Loss Functions Exist?

We now present our second contribution, in which we characterise all possible label noise models for which a non-degenerate Noise-Tolerant loss function may exist.

Non-Uniform Noise Up to this point, the discussion has focused exclusively on class-dependent, which is to say uniform, label noise. Below, we establish that there are no Noise-Tolerant loss functions for non-uniform label noise. Note that this relies on the truth of Conjecture 7.2.5.

[2] See Definition E.3.1 for formal definition.

Lemma 7.3.1. *If a loss function L is Noise Tolerant to some label noise model, then this label noise must be uniform.*

Theorem 7.3.2. *Assume the truth of Conjecture 7.2.5. Assume that a (non-degenerate) loss function L is Noise Tolerant to some noise model. This noise model must be class-conditional and given by a matrix T, which can be written in the following form:*

$$\begin{bmatrix} 1-\widehat{\eta_1} & \eta_1 & \dots & \eta_1 \\ \eta_2 & 1-\widehat{\eta_2} & \dots & \eta_2 \\ \dots & & & \\ \eta_c & \eta_c & \dots & 1-\widehat{\eta_c} \end{bmatrix} \tag{7.9}$$

for some $\eta_1, \eta_2, \dots, \eta_c$ *such that* $\sum_{i=1} \eta_i > 1$. *Where* $\widehat{\eta_k} := (\sum_{i=1}^{c} \eta_i) - \eta_k$ *- the sum of all the* η_i except η_k. *Note that when* $\eta_i = \frac{\eta}{c-1}$, *this gives a matrix corresponding to symmetric label noise.*

Remark 7.3.3. *How do we interpret the matrix form given in Equation 7.9? This matrix corresponds to label noise where, given a class k, the probability of being mislabelled as class k does not depend on the true class. For example, the probability of cat* $\mapsto$ *dog and* $fish \mapsto dog$ *is the same and equal to some* η_{dog} *(depending only on the target class).*

7.4 Conclusions

Ultimately, Theorem 7.3.2 is restrictive. For non-uniform and most asymmetric label noise models, Noise-Tolerant loss functions do not exist. Nevertheless, the results established in this chapter expand on the previous work by Ghosh and Kumar (2017), Ghosh et al. (2015), Manwani and Sastry (2013) and Van Rooyen et al. (2015) where strong results were limited to symmetric label noise. Moreover, we have extended our understanding of when Noise-Tolerant losses can and cannot exist, providing useful knowledge for practitioners in the field. The Noise-Tolerance property defined in Definition 7.1.1 is extremely strong, as it applies to any distribution and any family of estimators. It is somewhat miraculous that a loss function can satisfy this property under any type of noise.

7.4.0.1 Limitations

The main limitation of this work is that it relies on the truth of Conjecture 7.2.5. We have shown that this conjecture holds for $c = 3$ and provided intuition for a more general proof; nevertheless, a full general proof is required.

A second important limitation which should be considered is the unresolved question of Fisher consistency in the multi-class ($c > 2$) setting. In Section 7.2.2, we showed how to derive Noise-Tolerant loss functions to asymmetric label noise in the binary case, deriving the set $A \subseteq \Delta$ upon which these loss functions were Fisher consistent. For the multi-class setting this analysis is more difficult and remains unresolved. Specifically, what is the maximal set A upon which a loss function can be Fisher consistent given a transition matrix T of the form Equation 7.9? Can we find a Noise-Tolerant loss function that obtains consistency on this set?

We have defined Noise Tolerance in the strongest possible sense, requiring that a minimiser of the noisy risk within the parametric family Q should also minimise the clean risk—this must hold for any Q and for any data distribution. Further work might look at relaxing this definition in case this permits the existence of other non-trivial loss functions. The definition of Noise Tolerance could be relaxed in two ways: Rather than requiring that a minimiser of the noisy risk is also a minimiser of the clean risk, we could consider a loss function Noise Tolerant if the minimisers of the noisy and clean risks over a particular model class induce the same plug-in classifiers. Since our objective is minimising the misclassification rate on the clean distribution, this weakened objective is sufficient. Secondly, we could consider a specific set of models Q rather than requiring Noise Tolerance for all Q. This approach is taken in Manwani and Sastry (2013), where, for example, they examine Noise Tolerance with respect to linear classifiers. Given that our focus is on neural network classifier models, this avenue is more challenging, and since such models are universal approximators, it might be argued that a stronger form of Noise Tolerance is required. Nevertheless, this remains an interesting avenue for future study.

Chapter 8

Conclusions

8.1 Introduction

8.1.1 Chapter Outline

In Section 8.2, we condense the thesis's key findings into a brief list. Section 8.3 elaborates on this, summarising each chapter's main contributions, their integration into our overall narrative, and noting key research limitations. Section 8.4 outlines the primary omissions and flaws of our work. Section 8.5 proposes future research directions that could enhance the thesis's coherence and impact. Finally, Section 8.6 offers concluding remarks, discussing the practical implications of this research and personal professional insights gained during my PhD.

8.2 Key Findings

8.2.1 Key Findings Summary

- No algorithm can perform optimally in all label noise settings (unless it directly utilises information about the data distribution and/or noise model). Consequently, a restriction on the problem domain is necessary. The 'class-preserving' assumption is a suitable choice. Most label noise studied in the related work satisfies this condition.

- Poor robustness is caused primarily by overfitting to finite noisy datasets - correction-based losses cannot prevent this. Existent regularisation-based approaches cannot consistently prevent overfitting.

- Label noise implies a lower bound on the risk, by preventing the training loss from going below this lower bound, overfitting can be avoided.
- Early Stopping by monitoring performance using a noisy validation dataset is effective. This reduces the requirement to develop loss functions which entirely avoid overfitting.
- Noise-Tolerant loss functions can only exist for specific class-preserving, class-conditional noise models.

8.3 Contributions

8.3.1 Reasons for Robustness

In the Background Chapter 2, we began by briefly demonstrating that some loss functions are more robust than others to label noise. We then enquired about the reason for the difference in robustness between different loss functions. We critiqued an existing explanation we named 'the Risk Hypothesis', concluding by proposing that poor robustness primarily comes from a neural network's tendency to overfit to noisy datasets, with loss functions which mitigate this tendency being those which exhibit greater robustness.

8.3.1.1 Risk Hypothesis

In Chapter 2, we discussed 'correction-based' losses. These approaches utilise an estimate of the noise model to 'correct' the loss to ensure consistency of the learning algorithm. We argued that these approaches are based on the implicit assumption that a primary reason why a loss function exhibits poor robustness is that, in the presence of label noise, the empirical risk no longer approximates our desired learning objective. We called this explanation for the observed differentials in loss robustness 'The Risk Hypothesis'.

We argued that the risk hypothesis is inadequate in explaining why certain loss functions are more robust than others. Our argument consisted of three main strands. 1) We argued that uncorrected loss functions like MAE and CE all exhibit very different robustness profiles. Thus, there must be major explanatory factors for why losses exhibit differing level of robustness beyond label noise's distortion of the empirical

risk. 2) We demonstrated the radically stronger corrective properties enjoyed by the backward correction over the forward correction. Despite this apparent weakness, the forward correction is more commonly utilised than the backward correction and typically performs better. This suggests that the improved robustness imbued by applying a forward correction may not be a consequence of it correcting the empirical risk. 3) We showed that when the label noise satisfies a property which we called 'class-preserving' no correction of the risk is needed to ensure consistency of the learning algorithm.

8.3.1.2 Class-Preserving Label Noise

An important contribution of this thesis was the introduction of the class-preserving label noise assumption. When proposing an approach to a problem, it is usually beneficial to begin by defining an explicit problem statement, ensuring this problem statement is well-defined and, in principle, attainable. Unfortunately, this practice is not always adhered to within the field of robust loss functions for neural networks. Researchers propose methods without outlining when these approaches might not be expected to work. In effect, researchers often present their methods as if they apply universally.

Despite generally poor practice, more conscientious label noise studies do introduce some restrictions on the problem statement when developing robust losses. Notable assumptions include the diagonal dominance assumption, clean-labels-dominance assumption, and various bounds imposed on noise rates (Angluin & Laird, 1988; Ghosh & Kumar, 2017; X. Li et al., 2021; X. Zhou et al., 2021). The class-preserving assumption is a formalisation and generalisation of all these assumptions. Chapter 4 demonstrates moreover, that this assumption is not especially restrictive, with most studied label noise being in this category. We hope this contribution will provide future researchers with a clearer understanding of what can be achieved when developing robust losses and will guide further research in this area.

8.3.1.3 Limitations

The primary objective of these chapters was to supplement the argument that the poor robustness of a loss function like cross-entropy mostly stems from it tendency to induce overfitting to noisily labelled datasets. We contrasted this explanation with the competing hypothesis that poor robustness is a consequence of a distorted risk objective. In retrospect, this narrative is probably oversimplified. In particular, these hypotheses are not entirely mutually exclusive: We illustrated how applying a loss correction

ensures consistency, but that when label noise is class-preserving, consistency is already assured. We also pointed out how applying a loss correction cannot prevent overfitting to a noisily-labelled dataset. We concluded from this that loss correction approaches are of minimal utility. However, there may still be a benefit in applying a correction (See e.g. Table A.1). Applying a correction probably alters how a model fits noisy data, potentially assuring that the model fits true labels before it fits noisy labels. In addition, it may help with data efficiency improving generalisation for smaller noisy datasets than with an uncorrected loss. This aligns well with Chapter 5 which shows that loss correction approaches work best when combined with approaches to prevent overfitting.

8.3.2 Loss-Bounding to Prevent Overfitting

8.3.2.1 Risk Bounding Contributions

The main contribution of Chapter 5 was explaining that the presence of label noise implies the existence of a lower bound on the achievable generalised (noisy) risk. We demonstrated that, for separable data distributions, this lower bound could be estimated given an approximation of the noise rate. Crucially, we argued that the existence of this lower bound would allow us to ascertain whether a model had overfit to the noisy training dataset simply by inspecting whether its loss had dipped below this value. This led us to propose lower bounding the training loss, ensuring that it could not go below the computed minimal risk. We showed empirically that this approach could greatly improve the robustness of several standard loss functions. One minor contribution of Chapter 5 was providing a partial explanation for the effectiveness of the forward correction. We saw how forward correcting a loss function has the impact of imposing a lower bound on the training loss. We speculated that this bounding may contribute to its robustness properties given our earlier insights. This might also explain why the forward correction paradoxically performs better than the backward correction despite satisfying weaker theoretical guarantees since the backward correction imposes no such bound.

8.3.2.2 Generalising Corrections

The second contribution of Chapter 5 was to generalise the forward correction to allow for non-linear noise models. This allowed us to demonstrate that certain popular robust loss functions can be re-conceptualised as generalised forward corrections of proper loss functions. One of the benefits of this contribution is that label noise in the wild is probably often more accurately described by non-linear noise models. It would potentially be beneficial to explore non-linear loss functions more, evaluating their effectiveness in more realistic noise scenarios.

8.3.2.3 Limitations

A significant limitation was including loss bounding and our generalisation of forward corrections together into a single contribution. This mostly results in a more convoluted narrative which compromises the clarity of both contributions. The notion of loss bounding is not dependent on one utilising a forward correction - as shown with our experiments using cross-entropy, even a standard loss can benefit from loss bounding. It would be worth exploring how effective this approach is for other robust loss functions like MSE or MAE. It would probably have been more fruitful to employ our loss-bounding ideas to backwards-corrected loss functions which do not already imbue a bound on the training loss like forward-corrections. Likewise, splitting off the loss bounding contribution from generalising loss corrections would have allowed a greater exploration of generalised correction losses. Interestingly backward corrections cannot be generalised to allow for non-linear noise models. A consequence of our desire to combine these two contributions into a single narrative meant that this insight and others were omitted.

8.3.3 Noisy Early Stopping

8.3.3.1 Motivation

The primary motivation of Chapter 6 was to develop a standard approach for determining how many epochs to train for so that this isn't left up to the researcher proposing the loss, which can result in p-hacking issues. We wanted particularly to find an approach which works even when a cleanly labelled validation dataset doesn't exist to monitor performance during training.

8.3.3.2 Contributions

The major contribution of Chapter 6 is an approach to determine when to cease training a neural network classifier on datasets polluted with noisy labels to avoid overfitting. We demonstrated that monitoring accuracy on a noisy validation dataset, drawn from the same distribution as the noisy training data, can reliably indicate when clean accuracy begins to decline. A minor contribution of this work is the provision of some theoretical insights relating the noisy and clean 0-1 risks of a model under different noise environments. In the latter portion of Chapter 6, we explored reasons for the surprisingly good empirical performance of our Noisy Early Stopping (NES) policy. This exploration, while interesting, was not entirely conclusive. The main contribution of this section was our proposed mechanism for analysing overfitting in neural networks trained on label noise. We introduced a stochastic matrix summarising how the model overfits during training, linking the behaviour of this matrix to the performance of NES. We believe this tool is intriguing and warrants further study.

8.3.3.3 Limitations

The main limitation of Chapter 6 is that we do not conclusively explain why NES is effective. We propose a possible explanation for the effectiveness of NES in terms of how neural networks tend to overfit. Specifically, we show that if overfitting satisfies some notion of simultaneity, then NES will be effective. While we demonstrate with some examples that neural networks satisfy this simultaneous overfitting condition, we do not fully understand *why* they do. A rigorous understanding of when and why a neural network should behave in this manner would satisfyingly conclude this line of work.

We initially stated that a major motivation for our research was to understand how to evaluate and compare models trained on noisy data when there is no clean test dataset. Unfortunately, we did not manage to directly address this question. While we showed that noisy accuracy could be used to evaluate a model during training, we did not demonstrate whether noisy accuracy is a reliable metric for comparing between different models.

8.3.4 Noise-Tolerant Loss Functions

In Chapter 7, we explored Noise-Tolerant loss functions, building on the existing literature for this topic. We gave a necessary and sufficient condition for a loss to be Noise Tolerant and when such losses could and could not exist.

8.3.4.1 Motivation

In Chapter 2, we argued that the robustness of a loss function is determined by its propensity to induce overfitting to finite data, suggesting that efforts should focus on finding principled ways to avoid overfitting. However, in Chapter 6, we developed an effective early-stopping approach to prevent overfitting when training with noisily-labelled data. With such a mechanism at our disposal, we contended that it becomes more crucial to construct loss functions that optimise peak performance during training rather than merely avoiding overfitting. Noise-Tolerant losses are known to achieve good peak performance during training; thus, enhancing our understanding of these approaches is valuable. Previous work has developed novel Noise-Tolerant loss functions for practical use (Ma et al., 2020); however, these are constrained by theoretical limits that only allow for the construction of losses tolerant to symmetric label noise. This section aimed to extend this theory to develop novel Noise-Tolerant losses for asymmetric label noise models.

8.3.4.2 Contributions

The main contribution of Chapter 7 was the so-called 'Noise-Tolerance Theorem,' which establishes a necessary and sufficient condition for a loss function to be Noise Tolerant for a specific noise model. This theorem extends the work by (Ghosh & Kumar, 2017), which provided a necessary condition for Noise Tolerance in the case of uniform symmetric label noise. Our theorem prompted us to explore our second major question: 'For which noise models can a Noise-Tolerant loss function exist?' We demonstrated that Noise-Tolerant loss functions are feasible only in settings of class-preserving label noise. In other scenarios, the Noise-Tolerance property compromises the consistency of the learning algorithm. We concluded by affirming that Noise-Tolerant loss functions exist solely for a specific subset of class-conditional label noise models which we described.

8.3.4.3 Limitations

This work has two primary limitations. Firstly, some of our proofs rely on the assumption that the image of the simplex under the loss function should have a dimension of at least $c-1$ to ensure the non-degeneracy of the loss. While this assumption is reasonable, it is regrettable from a completeness perspective that we cannot establish this rigorously. Secondly, there is a lack of a section where we construct Noise-Tolerant loss functions and evaluate their performance on relevant benchmarks. It is crucial to determine whether the Noise-Tolerance property genuinely enhances robustness in practical settings or if it remains more of a mathematical curiosity.

8.4 Limitations

8.4.1 Lack of Learning Theory

A significant limitation of this research is the absence of a formal learning theory analysis regarding the robustness of loss functions. Ideally, such an analysis would clarify how generalisation is influenced by the choice of loss function and the properties of a noisy dataset. However, applying learning theory to neural network classifiers trained by SGD is challenging. A complete theoretical framework that accurately predicts the performance of neural networks trained with SGD remains elusive. Consequently, the generalisation bounds derived for neural networks often differ significantly from the actual observed outcomes, including rates of convergence and generalisation to unseen data. While these theoretical bounds can be illustrative, they often lack practical relevance.

Despite these challenges, in retrospect, the inclusion of some learning theory, perhaps initially focusing on more classical models like SVMs or logistic regression, might have been beneficial. This approach could have provided foundational insights into how different loss functions impact simpler settings, potentially enhancing the overall analysis presented in this work.

8.4.2 Noise Models

A significant limitation of our work, which reflects a broader issue within the field of label noise robust learning, is the need for more diversity in label noise models. We focus almost exclusively on closed-set noise and primarily look at synthetic noises where the dataset labels are manually corrupted. One of the primary reasons for this is

necessity; to evaluate the performance of trained models, one needs to have a cleanly labelled test set. Without a clean dataset, one must resort to evaluating test performance using a noisy test set - this is the practice for the Clothing1M and ImageNet datasets from Chapter 5. However, there is a risk that models evaluated on noisy test sets might be inadvertently rewarded for fitting systematic noise rather than generalising to the underlying clean data-label distribution. This can lead to situations where a model that performs poorly on clean data may appear to perform well on noisy data. By employing synthetic noise models, we sidestep this issue, but at the cost of possibly limiting the practical applicability of our findings. At least half of all experiments in the relevant literature are on uniform symmetric label noise - however, in the context of image classification, this isn't a realistic noise in most real-world scenarios.

8.5 Future Research Directions

8.5.1 Future Research Pathways Motivated by Our Findings

'Peak' Performance of Different Loss Functions At the end of Chapter 6, we argued that future work on developing label noise robust loss functions should concentrate more on optimising peak performance rather than mitigating overfitting during training. Let us clarify what we mean by this. When training with, say, a cross-entropy loss function on noisily labelled data, one typically observes an initial increase in generalisation performance followed by a decrease as overfitting sets in (see, e.g., Figure 2.3). Most research into robust loss functions has focused on preventing or at least mitigating this overfitting stage. However, if one has a clean validation set to evaluate model performance during training, training can be halted early once overfitting begins, thus capturing the peak-performing model. Notably, the findings of Chapter 6 suggest that a *clean* validation set isn't even necessary. Consequently, in our view, what matters is not creating losses that prevent overfitting but rather developing loss functions that achieve a higher 'peak'. This question is intriguing from both practical and theoretical standpoints and merits further investigation.

Backward Corrections In Chapter 2, we demonstrated that backward correction possesses strong properties: specifically, it is the only modification one can make to a loss function that renders the noisy empirical risk an unbiased estimator of the clean risk (Lemma A.1.1). While there is some existing research on backward corrections (Natarajan et al., 2013; Patrini et al., 2017; Stempfel & Ralaivola, 2009), this area

remains under-explored. The backward correction of the cross-entropy loss function is often unstable when used with gradient-based methods. Moreover, backward correction involves inverting a matrix, which can be computationally expensive. Additionally, it generally underperforms compared to forward correction. These factors may explain why it has not attracted as much attention as the forward variant. Despite these challenges, we believe that many of these issues can be addressed—mainly through some of the overfitting avoidance techniques presented in Chapters 5, 6—which could unlock the full potential of backward correction. As discussed in Section 8.4.2, developing methods for evaluating and comparing models without a clean test set is crucial. The theoretical properties of the backward correction make it particularly well-suited for this task.

Batchwise Loss Functions One of the major limitations of this work has been our insistence (in the majority of chapters) that loss functions be 'elementwise', which is to say expressible in the form:

$$L : \Delta \times \mathcal{Y} \to \mathbb{R},$$
$$(\boldsymbol{q}, y) \mapsto L(\boldsymbol{q}, y = k).$$

Such losses evaluate a single forecast over labels, represented as a point in the simplex, against a single revealed label. In the context of classification, the approach we have taken is standard, and the vast majority of commonly used loss functions can be expressed this way. Nevertheless, this form is limiting. In Chapter 5, we introduced the 'noise-bounded loss' (Equation 5.4.6), which is defined over a batch of data, i.e., rather than computing the loss independently for each sample in the training set, we compute a loss over a minibatch of data. If the minibatch $\mathcal{D} \coloneqq \{(x_i, y_i)\}_{i=1}^N$ contains N data-label pairs, a 'batchwise' loss function takes the form:

$$L : \Delta^N \times \mathcal{Y}^N \to \mathbb{R},$$
$$L(\{\boldsymbol{q}(x_i)\}_{i=1}^N, \{y_i\}_{i=1}^N) \mapsto \mathbb{R}.$$

Any elementwise loss function L can be conceptualised as a batchwise loss L_{batch} via

$$L_{batch}(\{\boldsymbol{q}(x_i)\}_{i=1}^N, \{y_i\}_{i=1}^N) \coloneqq \sum_{i=1}^N L(\boldsymbol{q}(x_i), y_i).$$

However, the converse is not true, meaning that batchwise loss functions are necessarily a richer space of loss functions. In generative modelling, such loss functions are commonplace, for instance, the MMD or Wasserstein losses. Other work within the classification literature implements batchwise loss functions such as 'Peer loss functions'

(Y. Liu & Guo, 2020) and L_{DMI} (Xu et al., 2019). Theoretical work exploring topics like Noise Tolerance and loss corrections within this expanded framework would potentially be fruitful. It remains possible that this richer functional family allows the construction of objects which are not feasible in the elementwise case, such as backward corrections for non-linear losses or Noise-Tolerant loss functions for non-uniform label noise models.

Noise Tolerance for Particular Hypothesis Classes When we discuss Noise Tolerance, we refer to the condition that, for any distribution $p(x,y)$, and any set of estimators Q, the minimiser of the noisy and clean risks should be identical. This property of a loss function is extremely strong and, therefore, highly restrictive. As shown in Chapter 7, such loss functions cannot exist for most label noise models. By relaxing this definition, we believe a richer variety of Noise-Tolerant loss functions may be derived. Restrictions can be applied either to the space of distributions or to the hypothesis class. An example of this approach is found in the work of Manwani and Sastry (2013), where they demonstrate that linear classifiers are tolerant to uniform symmetric label noise when using a mean-squared error loss function. The scope of our research was specifically deep models, so restricting the hypothesis class was not a viable option. Nevertheless, we anticipate that fruitful enquiries can probably be made here.

8.5.2 Additional Research Directions

In Section 8.4, we highlighted a limitation of our research, which primarily utilised benchmarks derived from synthetic closed-set noise models. We expressed concerns about how this might limit the practical applicability of our findings. Another significant limitation is our exclusive focus on balanced classification tasks. In real-world scenarios, such as medical diagnostics, class frequencies often vary significantly; for example, the prevalence of a disease is typically much lower than that of non-disease cases. Understanding the effects of label noise in these unbalanced classification settings is crucial for practical applications. Therefore, an essential direction for future research is to explore robust loss functions specifically tailored to the challenges posed by unbalanced data contexts.

Additionally, we have focused exclusively on accuracy when evaluating our approaches. However, in many practical situations, practitioners also care about the calibration of methods. Calibration is important because it measures the reliability of the probability scores assigned to predictions, which is critical in decision-making

scenarios such as medical diagnosis or risk assessment. As noted in Chapter 7, Noise-Tolerant loss functions can never be proper, suggesting that such approaches might suffer from poor calibration. In contrast, correction approaches may retain good calibration when the noise transition matrix is accurately approximated. It would be interesting to compare different robust losses, including backward and forward corrections, with respect to their calibration properties.

8.6 Concluding Remarks

8.6.1 Theoretical and Practical Implications

Our main interest in developing label noise robust approaches was to enhance the utility of small noisy datasets, thereby assisting in the democratisation of machine learning, which is increasingly dominated by large, resource-rich organisations. For this reason, we have focused on providing simple and computationally inexpensive approaches: loss bounding to prevent overfitting, early stopping without a clean dataset and Noise-Tolerant loss functions. Although the proposals in this work are not groundbreaking, we hope they contribute positively to the broader body of research and help nudge us toward this desired direction.

8.6.2 What I Have Learned

One of the most significant lessons from my PhD is understanding the real nature of research. At first, maybe influenced too much by movies, I thought research had to be groundbreaking and contrarian to be worth publishing. This led me to waste a lot of time trying to redesign and rethink approaches which did not need rethinking. A lot of this effort didn't end up contributing to my thesis and came from a misunderstanding of what research actually involves.

I've since learned that effective research is usually more about being thorough than being revolutionary. It's about carefully reviewing what's already known, understanding the main challenges in the field, and making sensible, step-by-step contributions that build on existing knowledge. It's also crucial to present these contributions clearly and honestly. This experience has reshaped my understanding of how to make a meaningful contribution to academic discussions.

8.6.3 Personal Reflections

Conducting a PhD is a strange experience. At times, the freedom is wonderful and exhilarating; however, it can also be incredibly lonely. While the research for this PhD did not always go as well as I had hoped, I have learned a great deal, both about conducting research and about life more broadly. I have developed good work practices, improved my time-management skills, and gained a solid understanding of how to structure and carry out a large project. I feel privileged to have had the opportunity to spend my days exploring ideas and coding up different projects. It has also been an honour and a delight to make friends with so many brilliant and kind individuals.

Appendices

Appendix A

Robustness of Loss Functions to Label Noise

A.1 Comparing Forward and Backward Corrections

In this section, we compare the backward and forward corrections. We reveal that these two correction strategies are fundamentally distinct in that backward and forward corrections never align when the label noise is ergodic. We show that the backward correction exhibits stronger properties than its forward counterpart. We critique the forward correction, indicating that it fails to meet some of the desiderata one might anticipate from a loss correction method.

A.1.1 Forward Correction

Arguably the most popular loss-function-based way to handle label noise robustness is through forward-correction, having been published in different forms on numerous occasions (Patrini et al., 2017; Sukhbaatar et al., 2015) and combined with other methods X. Li et al. (2021). Recall, given class-conditional label noise, a loss L and an estimate of the transition matrix $\hat{T}$, the forward-correction is defined as $L_F(\boldsymbol{q},k) := L(\hat{T}\boldsymbol{q},k)$. When the true transition matrix is known ($\hat{T} = T$), this correction ensures that

$$\underbrace{\arg\min_q R_L(q) = \arg\min_q R^{\eta}_{L_F}(q)\,.}_{\text{The } \textbf{Weak} \text{ Corrective Property}} \tag{A.1}$$

We call this the 'Weak Corrective Property'. The minimum here is taken over the space of *all* probability estimators $\boldsymbol{q} : \mathcal{X} \to \Delta$ for which the corresponding risks are defined.

In practical settings we are not optimising over the space of all probability estimators, rather we are restricted to a parametric set Q which, in our setting, is parameterised by a neural network. Suppose that for any Q, the restricted relation holds wherein the argmin is computed over Q. That is;

$$\underbrace{\forall Q, \quad \arg\min_{q\in Q} R_L(q) = \arg\min_{q\in Q} R^{\eta}_{L_{corr}}(q)\,.}_{\text{The } \textbf{Strong} \text{ Corrective Property}} \tag{A.2}$$

We call this the 'Strong Corrective Property' since it is stronger than Equation A.1. The following example (Example A.1.1) demonstrates that the strong corrective property does *not* always hold for the forward correction. In other words, if a probability estimator model minimises the clean $L-$risk, there is *no* guarantee that this model also minimises the noisy L_F-risk, even when the corrected loss function L_F uses the true noise model in constructing the noise correction.

Example A.1.1. ***Minimising Risk with and without Label Noise.*** *Consider a binary data-label distribution $p(x,y)$ supported at points x_1, x_2 with $p(x_1) = p(x_2) = 0.5$, $p(y=1|x_1) = 0.2$, and $p(y=1|x_2) = 0.8$. Define Q as probability estimator models with $q(x_1) = 1 - q(x_2)$. For cross-entropy loss L, we seek to minimise the risk:*

$$-\log(q(x_1)) - \log(1-q(x_1)).$$

One may show this occurs at $q(x_1) = 0.5$.

With label noise via $T = \begin{bmatrix} 1 & 0.2 \\ 0 & 0.8 \end{bmatrix}$, the noisy probabilities become $\widetilde{p}(\widetilde{y}=1|x_1) = 0.36$ and $\widetilde{p}(\widetilde{y}=1|x_2) = 0.84$. For the forward-corrected loss L_F, the risk can be written:

$$\begin{aligned} -0.36\log(0.8q(x_1)+0.2) - 0.64\log(0.8-0.8q(x_1)) \\ -0.84\log(1-0.8q(x_1)) - 0.16\log(0.8q(x_1)), \end{aligned}$$

which is minimised at $q(x_1) = 0.2$, confirmed by the positive second derivative at this point. This shows that minimisers of the noisy and clean risks differ despite correcting the loss function with the true noise transition matrix.

A.1.2 Backward Correction

Comparison with Backward Correction The failure of the forward correction to satisfy the strong-corrective criterion in Equation A.2 contrasts with the backward correction. The following Lemma and subsequent Corollary establish that the strong corrective property holds for the backward corrected loss function. That is, for any

distribution $p(x,y)$,

$$\forall Q, \quad \arg\min_{q\in Q} R_L(q) = \arg\min_{q\in Q} R^{\eta}_{L_B}(q). \tag{A.3}$$

Lemma A.1.1. *Given any loss function $L : \Delta \times \mathcal{Y} \to \mathbb{R}$ and a non-singular stochastic transition matrix T that models class-conditional label noise, there exists a unique 'Backward-Corrected' loss function L_B ensuring that, for any given data-label distribution $p(x,y)$ and any probability estimator $\boldsymbol{q}(x)$, the noisy-risk calculated with L_B equals the noise-free risk under L, formally:*

$$R^{\eta}_{L_B}(\boldsymbol{q}) = R_L(\boldsymbol{q}). \tag{A.4}$$

The backward-corrected loss L_B is defined by $\boldsymbol{L}_B(\boldsymbol{q}) := T^{-T}\boldsymbol{L}(\boldsymbol{q})$, establishing a one-to-one correspondence between L and L_B for any specified label noise described by T.

Proof. Proof given in Appendix A.2. □

In (Patrini et al., 2017) the uniqueness part of Lemma A.1.1 is not given. While the authors show that the risk relation given in Equation A.4 holds for a backward-corrected loss function, Lemma A.1.1 shows this is the *only* loss which satisfies this condition.

Corollary A.1.2. *The Backward Correction satisfies the strong corrective property given in Equation A.3.*

Proof. Given an arbitrary set of estimators Q and data-label distribution, Lemma A.1.1 ensures that for each $\boldsymbol{q} \in Q$, $R^{\eta}_{L_B}(\boldsymbol{q}) = R_L(\boldsymbol{q})$. Our result follows by taking an argmin on both sides. □

A.1.2.1 The Ubiquity of the Weak Corrective Property

Example A.1.1 demonstrates how one can easily construct a scenario in which the minimiser of clean risk and minimiser of the corrected noisy risk are dramatically different when using the forward correction. This illustrates that the 'weak-corrective property' satisfied by the forward-correction is indeed rather weak. The following lemma builds on this further, demonstrating that many loss functions satisfy the weak-corrective property beside the forward correction.

Lemma A.1.3. *Given the cross-entropy loss function L and an arbitrary data-label distribution $p(x,y)$, consider class-conditional label noise with an invertible transition matrix T. For any arbitrary strictly proper loss function $\widetilde{L}$ (Definition 2.1.2), applying forward correction yields $\widetilde{L}_F(\boldsymbol{q},k) \coloneqq \widetilde{L}(T\boldsymbol{q},k)$. The corrected loss $\widetilde{L}_F$ then satisfies the weak corrective property*

$$\arg\min_q R_L(q) = \arg\min_q R^{\eta}_{\widetilde{L}_F}(q). \tag{A.5}$$

Proof. Proof given in Appendix A.2.0.2. □

Lemma A.1.3 is stated for the cross-entropy loss but holds more generally. Specifically, given some loss function L one can construct any number of loss functions $\widetilde{L}$ with the weak corrective property. For example, given a loss L and $\alpha \in [0,1]$ one may define

$$\boldsymbol{L}_\alpha \coloneqq T^{-\alpha T}\boldsymbol{L}(T^{1-\alpha}\boldsymbol{q}).$$

It is straightforward to show this constructed loss satisfies Equation A.1.

A.1.2.2 Biased Gradients

We optimise neural network models through gradient descent by sampling from the joint distribution, evaluating the model's loss on these samples, and updating the parameters based on the loss's gradient. In this context, for a loss function L, an ideal characteristic of a corrective loss function L_{corr} is that the gradients it generates from noisy data match (in expectation) those generated by L from clean data when applied to a model $\boldsymbol{q}$. This can be succinctly expressed as:

$$\mathbb{E}_{x,y\sim p(x,y)}[\nabla_\theta L(q(x;\theta),y)] = \mathbb{E}_{x,\widetilde{y}\sim\widetilde{p}(x,\widetilde{y})}[\nabla_\theta L_{corr}(q(x;\theta),\widetilde{y})]. \tag{A.6}$$

If Equation A.6 is satisfied, this means the corrective loss function L_{corr} produces gradients from noisy data that are, on average, equivalent to those L produced from clean data when applied to a model $\boldsymbol{q}$. This equivalence would suggest that L_{corr} could counteract the negative impact of data noise. We show in this section that the gradients of the forward correction are usually biased, meaning that they do not satisfy the gradient property given in Equation A.6. We begin by demonstrating that when the noise transition matrix T has a unique stationary distribution, the forward and backward corrections never coincide. This proves critical in following derivations.

Lemma A.1.4. *Let T be the transition matrix associated with some class-conditional label noise, and let Δ denote the probability simplex. Suppose T has a unique stationary distribution, meaning there exists a unique vector $\boldsymbol{\pi} \in \Delta$ such that $T\boldsymbol{\pi} = \boldsymbol{\pi}$, and for any vector $\boldsymbol{v}$, if $T\boldsymbol{v} = \boldsymbol{v}$, then $\boldsymbol{v} = \boldsymbol{\pi}$. Under these conditions, there is no non-trivial loss function L whose forward correction L_F and backward correction L_B are identical across all $\boldsymbol{q} \in \Delta$, formally:*

$$\begin{gathered} \text{if } \boldsymbol{L} : \Delta \to \mathbb{R}^c, \\ \text{satisfies } \boldsymbol{L}_B(\boldsymbol{q}) = \boldsymbol{L}_F(\boldsymbol{q}), \forall \boldsymbol{q} \in \Delta, \\ \text{then } L = \text{const.} \end{gathered}$$

Proof. Proof in Appendix A.2.0.3. □

Remark If a transition matrix T has a unique stationary distribution, then Lemma A.1.4 shows there are no non-trivial loss functions for which the backward and forward corrections are equal. We suspect that Lemma A.1.4 may be generalised to broader classes of label noise, but we are unable to show this.

Lemma A.1.5 (Biased Gradients)**.** *Given a loss function L and its forward correction L_F, tailored for class-conditional label noise with transition matrix T. If T has a unique stationary distribution, then the gradients produced by L_F are generally biased. This means that the ideal condition outlined in Equation A.6 typically does not hold.*

Proof. By switching the order of expectation and gradient, we see that Equation A.6 is equivalent to requiring that $R^{\eta}_{L_F}(\boldsymbol{q}) = R_L(\boldsymbol{q})$. Lemma A.1.1 tells us that the only loss which satisfies this property is the backward correction. However, we know that when T has a unique stationary distribution, there are no non-trivial losses where the backward and forward corrections are equal (Lemma A.1.4). □

Lemma A.1.5 says that the gradients one obtains from the forward corrected loss are biased. Conversely, it also demonstrates that the gradients obtained by the backward correction are unbiased. This condition holds if T has a unique stationary distribution. As discussed in Section 2.2, a sufficient condition for T to have a unique stationary distribution is ergodicity (Definition 2.2.3).

A.1.2.3 Summary

Correcting the loss using the forward-correction is the most common method for handling label noise. However, in this section, we observed that forward-corrected losses fail to satisfy a few conditions one may expect from a loss correction. We noted the following problems:

1. The forward-correction satisfies the weak-corrective property in Equation A.1. However, *many* other loss functions satisfy this property (Lemma A.1.3).

2. The strong-corrective property does not hold. For example, when our probability estimator models $\boldsymbol{q}$ are parameterised by a neural network, then minimising the noisy L_F-risk will not typically yield a minimiser of the clean risk.

3. The forward-correction typically induces biased gradients (Lemma A.1.5).

Conversely, the backward correction doesn't have any of these problems, satisfying a much stronger risk relation than the forward correction (Equation A.2 versus A.1). It is the *only* loss function which satisfies this condition (Lemma A.1.1). Unlike the forward correction, the backward correction satisfies Equation A.3 for any model family and never results in biased gradients when one uses the true noise model to perform the correction.

A.2 Forward vs Backward: Proofs

In Section A.1 we compared the forward and backward correction. In this section we give proofs of the various lemma and theorems from that section.

A.2.0.1 Backward Correction

Lemma A.2.1. *Given any loss function $L : \Delta \times \mathcal{Y} \to \mathbb{R}$ and a non-singular stochastic transition matrix T that models class-conditional label noise, there exists a unique 'Backward-Corrected' loss function L_B ensuring that, for any given data-label distribution $p(x,y)$ and any probability estimator $\boldsymbol{q}(x)$, the noisy-risk calculated with L_B equals the noise-free risk under L, formally:*

$$R^{\eta}_{L_B}(\boldsymbol{q}) = R_L(\boldsymbol{q}). \tag{A.7}$$

The backward-corrected loss L_B is precisely defined by $\boldsymbol{L}_B(\boldsymbol{q}) \coloneqq T^{-T}\boldsymbol{L}(\boldsymbol{q})$, establishing a one-to-one correspondence between L and L_B for any specified label noise described by T.

Proof. We begin by showing that, if one defines $\boldsymbol{L}_B(\boldsymbol{q}) \coloneqq T^{-T}\boldsymbol{L}(\boldsymbol{q})$ then, for any probability estimator $\boldsymbol{q} : \mathcal{X} \to \Delta$ and data-label distribution $p(x,y)$ we have $R^{\eta}_{L_B}(\boldsymbol{q}) = R_L(\boldsymbol{q})$. Fix arbitrary $x \in \mathcal{X}$. Consider the pointwise noisy L_B-risk at x. This may be written in vector form as follows where $\widetilde{\boldsymbol{p}}(x)$ is the vector whose i^{th} entry is $\widetilde{p}(\widetilde{y} = i \mid x)$

$$\begin{aligned}
\tilde{\boldsymbol{p}}(x)^T\boldsymbol{L}_B(\boldsymbol{q}) &= (T\boldsymbol{p}(x))^T \cdot \boldsymbol{L}_B(\boldsymbol{q}) \\
&= \boldsymbol{p}(x)^T T^T \boldsymbol{L}_B(\boldsymbol{q}) \\
&= \boldsymbol{p}(x)^T T^T T^{-T}\boldsymbol{L}(\boldsymbol{q}) \\
&= \boldsymbol{p}(x)^T\boldsymbol{L}(\boldsymbol{q}) \qquad \text{(A.8)}
\end{aligned}$$

The final line, Equation A.8, is the clean pointwise risk at x. Thus, we have shown that the pointwise $L-$risk equals the noisy pointwise L_B-risk at x. The generalised $R_L(\boldsymbol{q})$ and pointwise $R_L(\boldsymbol{q})(x)$ risks are related via $R_L(\boldsymbol{q}) \coloneqq \int p(x)R^{\eta}_L(\boldsymbol{q})(x)dx$. Since x is arbitrary it follows that $R^{\eta}_{L_B}(\boldsymbol{q}) = R_L(\boldsymbol{q})$ as desired.

We now show the converse. Let $p(x,y)$ be some arbitrary data-label distribution and let $\boldsymbol{q} : \mathcal{X} \to \Delta$ be an arbitrary probability estimator model. Equation A.4 can be written as:

$$\begin{aligned}
&R_L(\boldsymbol{q}) = R_{L_B}(\boldsymbol{q}) \\
\iff &\int p(x)\mathbb{E}_{y\sim p(y|x)}[L(\boldsymbol{q}(x),y)] = \int p(x)\mathbb{E}_{\widetilde{y}\sim\widetilde{p}(\widetilde{y}|x)}[L_B(\boldsymbol{q}(x),\widetilde{y})].
\end{aligned}$$

By assumption, this equality holds for all distributions $p(x,y)$. In particular, we may set $p(x) = \delta(x_0)$ - a Dirac delta at some arbitrary point x_0. Using $\boldsymbol{p}$ to denote the vector whose i^{th} entry is $p(y = i \mid x_0)$ then, for all $\boldsymbol{p}, \boldsymbol{q} \in \Delta$ we must have

$$\begin{aligned}
&\boldsymbol{p}^T\boldsymbol{L}(\boldsymbol{q}) = (T\boldsymbol{p})^T\boldsymbol{L}_{\boldsymbol{B}}(\boldsymbol{q}) \\
\iff &\boldsymbol{p}^T(\boldsymbol{L}(\boldsymbol{q}) - T^T\boldsymbol{L}_B(\boldsymbol{q})) = 0
\end{aligned}$$

Setting $\boldsymbol{p} = \boldsymbol{e}_k$[1], it follows that for all $\boldsymbol{q} \in \Delta$, that $\boldsymbol{L}(\boldsymbol{q}) = T^T\boldsymbol{L}_B(\boldsymbol{q})$, thus $\boldsymbol{L}_B(\boldsymbol{q}) = T^{-T}\boldsymbol{L}(\boldsymbol{q})$ as desired.

□

Remark In Section 2.3.2.2 we looked at 'Importance Reweighting' (T. Liu & Tao, 2015). This guaranteed the same relationship between the noisy risk as the backward correction (Equation A.4). At first, Lemma A.1.1 would seem to render this impossible. How do we reconcile these facts? Given a transition matrix T, the backward correction

[1] $\boldsymbol{e}_k$ is the k^{th} coordinate vector where entry k^{th} is 1 and other entries are 0

satisfies the relation Equation A.4 for any probability estimator $\boldsymbol{q}$ and distribution $p(x,y)$. This differs from importance weighting where the weights are a function of the distribution $p(x,y)$. Thus, uniqueness relies on the fact that we don't allow our corrected loss to be a function of the underlying distribution. When this is relaxed there are other ways in which Equation A.4 may be obtained. Practically speaking, allowing the loss to be a function of the underlying latent class distribution is problematic.

A.2.0.2 Forward Correction

Lemma A.2.2. *Given the cross-entropy loss function L and an arbitrary data-label distribution $p(x,y)$, consider class-conditional label noise with an invertible transition matrix T. For any arbitrary strictly proper loss function $\widetilde{L}$ (Definition 2.1.2), applying forward correction yields $\widetilde{L}_F(\boldsymbol{q},k) := \widetilde{L}(T\boldsymbol{q},k)$. The corrected loss $\widetilde{L}_F$ then satisfies the risk relation*

$$\arg\min_q R_L(q) = \arg\min_q R^{\eta}_{\widetilde{L}_F}(q). \tag{A.9}$$

Proof. Since the minimum is taken over the space of all probability estimators, this relation follows almost immediately from the definition of a proper loss. Specifically, given a data-label distribution $p(x,y)$, one minimises the risk by minimising the pointwise risk at all points x in the support of $p(x)$. When L is proper the pointwise risk is minimised by setting $\boldsymbol{q}(x)$ equal to the true class distribution $\boldsymbol{q}(x) = \boldsymbol{p}(y \mid x)$. Thus, the left-hand side of Equation $A.5$ is minimised by setting $\boldsymbol{q}(x) = \boldsymbol{p}(y \mid x)$ for all $x \in \text{supp}(p(x))$. Outside of the support of $p(x)$, q may take any value since this does not impact the risk.

Similarly, one minimises the right-hand side (RHS) by minimising the pointwise risk at each x in the support of $p(x)$. Since $\widetilde{L}$ is strictly proper we minimise the pointwise $\widetilde{L}-$risk by setting $q(x)$ to equal the class distribution at x. It follows therefore than one minimises the pointwise **noisy** $\widetilde{L}_F-$risk by setting $T\boldsymbol{q}(x) = \widetilde{\boldsymbol{p}}(\widetilde{y} \mid x) = T\boldsymbol{p}(y \mid x)$. Since T is assumed to be invertible, then the RHS is minimised by any estimator where for all $x \in \text{supp}(p(x))$, $\boldsymbol{q}(x) = \boldsymbol{p}(y \mid x)$. □

A.2.0.3 Backward and Forward Losses are Distinct

Lemma A.2.3. *Let T be the transition matrix associated with some class-conditional label noise, and let Δ denote the probability simplex. Suppose T has a unique stationary distribution, meaning there exists a unique vector $\boldsymbol{\pi} \in \Delta$ such that $T\boldsymbol{\pi} = \boldsymbol{\pi}$, and for any vector $\boldsymbol{v}$, if $T\boldsymbol{v} = \boldsymbol{v}$, then $\boldsymbol{v} = \boldsymbol{\pi}$. Under these conditions, there is no non-trivial loss*

function L for which the forward correction L_F and the backward correction L_B are identical across all $\boldsymbol{q} \in \Delta$, formally:

$$\nexists L : \forall \boldsymbol{q} \in \Delta, \boldsymbol{L}_B(\boldsymbol{q}) = \boldsymbol{L}_F(\boldsymbol{q})$$

Proof. The premise that forward and backward corrections are identical for some loss L can be expressed as:

$$\boldsymbol{L}_F(\boldsymbol{q}) = \boldsymbol{L}_B(\boldsymbol{q}).$$

This implies $\forall \boldsymbol{q} \in \Delta$:

$$\boldsymbol{L}(T\boldsymbol{q}) = T^{-T}\boldsymbol{L}(\boldsymbol{q}) \quad \text{and} \quad T^T\boldsymbol{L}(T\boldsymbol{q}) = \boldsymbol{L}(\boldsymbol{q})$$

Letting $\boldsymbol{q} \mapsto T\boldsymbol{q}$, we obtain;

$$\boldsymbol{L}(T\boldsymbol{q}) = T^T\boldsymbol{L}(T^2\boldsymbol{q}) \quad \Rightarrow \quad \boldsymbol{L}(\boldsymbol{q}) = (T^2)^T\boldsymbol{L}(T^2\boldsymbol{q})$$

By induction, for all $n \in \mathbb{N}$ and $\boldsymbol{q} \in \Delta$, it holds that:

$$\boldsymbol{L}(\boldsymbol{q}) = (T^n)^T\boldsymbol{L}(T^n\boldsymbol{q})$$

Given T has a unique stationary distribution, in the limit, T^n converges to a matrix T^* where every column is $\boldsymbol{\pi}$. Therefore:

$$\boldsymbol{L}(\boldsymbol{q}) = \lim_{n\to\infty}(T^n)^T\boldsymbol{L}(T^n\boldsymbol{q}) = (T^*)^T\boldsymbol{L}(\boldsymbol{\pi})$$

This final expression has no $q-$dependence, showing that $\boldsymbol{L}(\boldsymbol{q})$ is constant for all $\boldsymbol{q}$, implying that L is trivial. □

A.3 Forward vs Backward: Performance Comparison

We compare the performance of the backward and forward corrections on some standard image datasets whose training datasets have been corrupted by synthetic label noise.

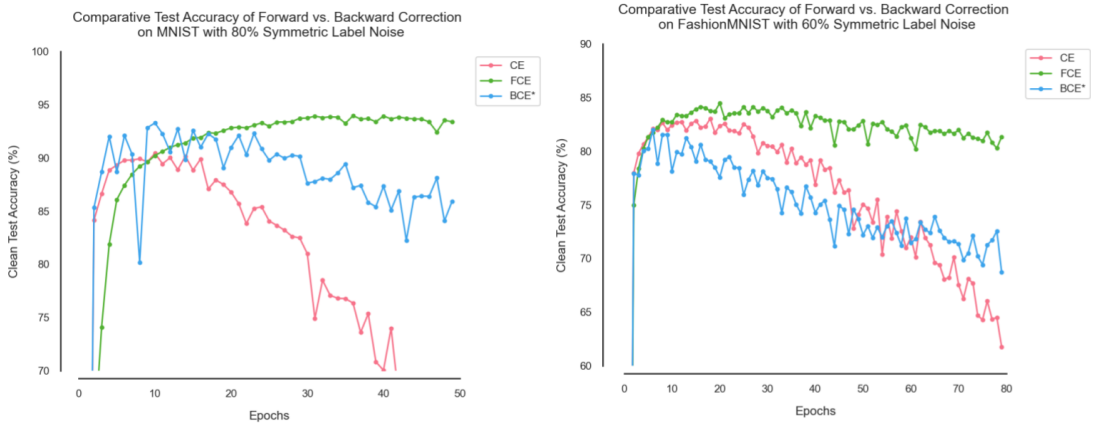
BCE is Unstable The backwards-corrected version of cross-entropy is unstable, and after a short period of training ($\sim$ 8 epochs), the loss starts returning NaNs. The time before this occurs can be prolonged by choosing a lower learning rate, but it cannot be avoided without changing the loss slightly. We alter the loss, preventing $q_k < \varepsilon$ for any $k \in \mathcal{Y}$, denoting this altered backward-correction as BCE*.

$$\boldsymbol{L}_B^*(\boldsymbol{q}) = T^{-T}(-log(q_1+\varepsilon), -log(q_2+\varepsilon), \ldots, -log(q_c+\varepsilon))$$

In our experiments, we set $\varepsilon = 0.0001$

Table A.1: Test accuracies (in %) for MNIST, FashionMNIST, CIFAR10, and CIFAR100 (Top 5 acc.) datasets at different noise rates for the Cross-Entropy (CE) loss and its forward and backward corrected variants (FCE and BCE respectively). Since noise is synthetic, we can use the true transition matrix to make these corrections. FCE is more robust than BCE despite worse theoretical guarantees. The $*$ after BCE indicates that it is altered slightly to make it more stable since BCE has exploding gradients.

Dataset	MNIST		FashionMNIST		CIFAR10		CIFAR100	
Loss/Noise Rate	**0.6**	**0.8**	**0.6**	**0.8**	**0.2**	**0.4**	**0.2**	**0.4**
CE	70.07	61.36	61.70	37.85	90.44	83.47	58.73	38.76
FCE	**93.78**	**93.41**	**81.29**	**78.54**	**92.00**	**87.41**	**68.29**	**58.20**
BCE*	83.89	85.92	68.74	65.10	90.90	85.48	63.12	44.43



(a) MNIST with 80% noise

(b) FashionMNIST with 60% noise

Figure A.1: Comparative test accuracy trajectories for Forward Correction Error (FCE, green), Backward Correction Error (BCE*, blue), and uncorrected Cross-Entropy (CE, red) on the MNIST (a) and FashionMNIST (b) datasets under high symmetric label noise conditions. For both datasets, FCE consistently outperforms BCE* and CE, demonstrating superior stability and accuracy. CE and BCE* exhibit initial accuracy gains, peaking near epoch ten before declining due to overfitting. The asterisk in BCE* signifies a stability-enhancing modification discussed in the text. This analysis highlights FCE's robustness across different datasets and noise levels, whereas BCE*, despite its enhancements, still lags behind FCE but outpaces the baseline CE, which overfits more significantly in the FashionMNIST scenario.

Experiment Setup For our experiments, we introduce label noise into the training datasets by randomly altering labels. Specifically, we apply symmetric label noise to the FashionMNIST and MNIST datasets at rates of $\eta = 0.6$ and 0.8. For CIFAR100, the symmetric label noise rates are $\eta = 0.4$ and 0.6. In contrast, for CIFAR10, we implement *asymmetric* label noise following the class-conditional model outlined in (Patrini et al., 2017) and detailed in Section 4.5. We use a neural network classifier to train on these noisy datasets, employing cross-entropy loss and its backward and forward-corrected variants. The backward correction method is modified as described to ensure stability. The known noising process allows us to utilise the true transition matrix for correction. We measure the classifier's performance by recording the accuracy on the clean test dataset over 80 training epochs (50 for MNIST). Each experiment is conducted with a learning rate of 0.0001 and a batch size of 300.

FCE is more Robust Table A.1 gives the clean test accuracies achieved by each loss function after training for the specified number of epochs. In most cases, the forward correction (FCE) performs better than the backward correction (BCE). This is despite the better theoretical guarantees possessed by the backward correction. Figure A.1 gives the clean test accuracy during training for the MNIST (left) and FashionMNIST datasets.

Appendix B

Class-Preserving Label Noise

B.1 Proofs

B.1.1 DD is Class-Preserving for Separable Distributions

Lemma B.1.1. *Suppose that a data-label distribution is separable. Suppose $p(x,y)$ is corrupted by class-conditional, diagonally dominant label noise. This label noise model is class-preserving for $p(x,y)$.*

Proof. Let T denote the transition matrix for the DD class-conditional label noise. Let $x \sim p(x)$ be some arbitrary data point. Let $\boldsymbol{p}(y \mid x)$ denote the clean conditional class distribution at x and $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x)$ the noisy conditional class distribution. Our claim is that

$$\underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \widetilde{\boldsymbol{p}}(\widetilde{y}=i|x) = \underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \boldsymbol{p}(y=i|x).$$

Since $p(x,y)$ is separable we know that there exists $k \in \mathcal{Y}$ for which $\boldsymbol{p}(y \mid x) = \boldsymbol{e}_k$. Hence

$$\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x) = T\,\boldsymbol{p}(y \mid x) = T\boldsymbol{e}_k = T_{\cdot,k},$$

the k^{th} column of the matrix T. The diagonal dominance assumption means that $T_{kk} > \max_{j\neq k} T_{jk}$ so

$$\underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \widetilde{\boldsymbol{p}}(\widetilde{y}=i \mid x) = k = \underset{i\in\{1,2,\ldots,c\}}{\arg\max}\ \boldsymbol{p}(y=i|x)$$

as required. □

B.1.2 Sufficient Conditions for Noise to be Class-Preserving Proofs

Lemma B.1.2. *Symmetric noise is class-preserving for any data-label distribution* $p(x,y)$ *whenever the noise rate* η *satisfies* $\eta < \frac{c-1}{c}$.

Proof. Let $p(x,y)$ be some arbitrary data-label distribution and let $x \in \mathcal{X}$ be some arbitrary point. Let $\boldsymbol{p}$ denote the class distribution at x, that is $\boldsymbol{p} \coloneqq \boldsymbol{p}(y \mid x)$. Without loss of generality we suppose that class 1 is a dominant class; $p_1 \geq p_i$ for all $i \in \{1,2,\ldots,c\}$. T denotes the symmetric noise transition matrix so that $T_{ii} = 1-\eta$ and $T_{ij} = \frac{\eta}{c-1}$ otherwise. We suppose that $\eta < \frac{c-1}{c}$. Let $k \neq 1$ be such that $p_k < p_1$ then

$$(Tp)_k = (1-\eta)p_k + \sum_{i \neq k} \frac{\eta}{c-1} p_i$$
$$(Tp)_1 = (1-\eta)p_1 + \sum_{i \neq 1} \frac{\eta}{c-1} p_i$$

Thus,

$$\begin{aligned}(Tp)_1 - (Tp)_k &= \left(1 - \eta - \frac{\eta}{c-1}\right)(p_1 - p_k) \\ &= \left(1 - \frac{c\eta}{c-1}\right)(p_1 - p_k) \\ &> 0\end{aligned}$$

Note that the inequality at the end follows by the assumption that $p_1 > p_k$ and $1 - \frac{c\eta}{c-1} > 0 \iff \frac{c-1}{c} - \eta > 0 \iff \eta < \frac{c-1}{c}$.

Thus, we have $\arg\max_i p(y=i \mid x) = \arg\max_i \widetilde{p}(\widetilde{y}=i \mid x)$ as desired. □

Example The following example illustrates the importance of the $\frac{c-1}{c}$ threshold on the noise rate. Consider the case where we have three classes ($c=3$). Since $c=3$, the class-preserving threshold occurs at a noise rate $\eta = \frac{c-1}{c} = \frac{2}{3} \approx 67\%$. Suppose the noise rate exceeds this critical threshold, say $\eta = 80\%$. Let x_0 be a datapoint where the conditional class distribution satisfies $\boldsymbol{p}(y \mid x_0) = (0.7, 0.2, 0.1)$: i.e. the dominant class at x_0 is $y=1$, with $y=2$ being the second most likely class followed by $y=3$. We can compute the noisy conditional class distribution at x_0 by multiplying $\boldsymbol{p}(y \mid x_0)$ by the relevant transition matrix[1], obtaining $\widetilde{\boldsymbol{p}}(y \mid x_0) = (0.26, 0.36, 0.38)$. By inspecting this noisy conditional distribution, we see that the dominant class has not been preserved at this extreme noise level. In fact, *least* likely noisy label to be observed at x_0 is the most-likely clean label ($y=1$). The order has been reversed, where the most likely

[1] Recall Equation 2.1.

noisy label was previously the least likely clean label. This indicates that this label noise beyond this threshold level completely corrupts the signal associating datapoints with their clean labels.

Lemma B.1.3. *Let $p(x,y)$ be a data-label distribution. Pairwise label noise with mislabeling probability $\eta < \frac{1}{2}$ is class-preserving if for every x, the dominant class probability $p_{max}(x) \coloneqq \max_i p(y=i \mid x)$ and the next highest class probability $p_{res}(x) \coloneqq \max_{i \neq k} p(y=i \mid x)$, satisfy $p_{max}(x) \geq \frac{1-\eta}{1-2\eta} p_{res}(x)$.*

Proof. We fix some arbitrary $x \in \mathcal{X}$ and denote the class distribution at x by $\boldsymbol{p} \coloneqq \boldsymbol{p}(y \mid x)$. Let $k \in \{1,2,\ldots,c\}$ be the dominant class; $p_k \geq p_i$ for all i and suppose without loss of generality that $k=1$. Let p_{res} be the maximum of the remaining components of $\boldsymbol{p}$. In order to be class-preserving we must show that $(T\boldsymbol{p})_1 \geq (T\boldsymbol{p})_i$ when the p_1 and p_{res} satisfy the specified relation.

Suppose that $p_1 \geq \frac{1-\eta}{1-2\eta} p_{res}$ and let $i=2$ then

$$\begin{aligned}
(T\boldsymbol{p})_2 &= (1-\eta)p_2 + \eta p_1 \\
&\leq (1-\eta)p_{res} + \eta p_1 \\
&\leq (1-2\eta)p_1 + \eta p_1 \qquad \text{(B.1)} \\
&\leq (1-\eta)p_1 \\
&\leq (1-\eta)p_1 + \eta p_c \\
&= (T\boldsymbol{p})_1
\end{aligned}$$

Equation B.1 follows from our assumption that $p_1 \geq \frac{1-\eta}{1-2\eta} p_{res}$.

Now let $i=c$ then

$$\begin{aligned}
(T\boldsymbol{p})_c &= (1-\eta)p_c + \eta p_{c-1} \\
&\leq \eta p_c + (1-2\eta)p_c + \eta p_{c-1} \\
&\leq \eta p_c + (1-\eta)p_{res} + \eta p_{res} \\
&\leq \eta p_c + (1-2\eta)p_{res} + \eta p_{res} \\
&\leq \eta p_c + (1-\eta)p_{res} \\
&\leq \eta p_c + (1-\eta)p_1 \\
&= (T\boldsymbol{p})_c
\end{aligned}$$

Finally let $i \neq 1, c, 2$. Then

$$\begin{aligned}
(T\boldsymbol{p})_i &= (1-\eta)p_i + \eta p_{i-1} \\
&\leq (1-\eta)p_{res} + \eta p_{res} \\
&\leq p_{res} \\
&\leq \frac{(1-2\eta)}{1-\eta} p_1 \qquad \text{(B.2)} \\
&\leq (1-\eta)p_1 \qquad \text{(B.3)} \\
&\leq \eta p_c + (1-\eta)p_1 \\
&= (T\boldsymbol{p})_c
\end{aligned}$$

Line B.2 follows from the assumption $p_1 \geq \frac{1-\eta}{1-2\eta} p_{res}$. While line B.3 follows from the fact that $\frac{(1-2\eta)}{1-\eta} \leq (1-\eta)$ on the interval $[0, 0.5)$. □

B.1.3 Symmetric Noise Is The Only Universally Class-Preserving Noise Model

Lemma B.1.4. *Let T be a transition matrix for class-conditional label noise. Then, T represents symmetric label noise at some rate $\eta < \frac{c-1}{c}$ if and only if, for all $\boldsymbol{p} \in \Delta$,*

$$\arg\max_i (T\boldsymbol{p})_i = \arg\max_i \boldsymbol{p}_i. \qquad \text{(B.4)}$$

Proof. Assume that T satisfies the required property. Start by setting $\boldsymbol{p} = \boldsymbol{e}_k$ then it follows that for all $i \neq k$, $T_{kk} > T_{ik}$. To help clarify the following, we write out an arbitrary transition matrix

$$\begin{bmatrix} T_{11} & T_{12} & \dots & T_{1c} \\ T_{21} & T_{22} & \dots & T_{2c} \\ \dots & & & \\ T_{c1} & T_{c2} & \dots & T_{cc} \end{bmatrix}.$$

Let $\boldsymbol{p} = \frac{1}{c}(1, 1, \dots, 1)$, i.e. each of the components of $\boldsymbol{p}$ are equal. The condition in Equation 4.1 implies that the sum of each row must be equal; T is row stochastic and column stochastic. Now setting $\boldsymbol{p} = \frac{1}{c-1}(1, 1, \dots, 1, 0)$, Equation 4.1 implies that, there exists some constant a, where for each $k < c$, the sum $\sum_{i=1}^{c-1} T_{ki}$ equals a. Using the knowledge that T is row stochastic then $T_{1c} = T_{2c} = \dots = T_{c-1,c}$. Now, letting $\boldsymbol{p} = \frac{1}{c-1}(1, 1, \dots, 1, 0, 1)$ Equation 4.1 implies $T_{1,c-1} = T_{2,c-1} = \dots = T_{c-2,c-1} = T_{c,c-1}$.

Continuing this process of setting $\boldsymbol{p} = \mathbf{1} - \boldsymbol{e}_i$ for $i \in \{1, 2, \ldots, c\}$, we can conclude that each non-diagonal entry of every column is constant: e.g. for $c = 4$, T takes the following form

$$\begin{bmatrix} T_{11} & a_2 & a_3 & a_4 \\ a_1 & T_{22} & a_3 & a_4 \\ a_1 & a_2 & T_{33} & a_4 \\ a_1 & a_2 & a_3 & T_{44} \end{bmatrix}.$$

Since each column sums to 1 then, we must also have

$$T_{ii} = 1 - (c-1)a_i$$

Let $\boldsymbol{p} = (0.5, 0.5, 0, \ldots, 0)$ then the condition from Equation 4.1 implies that

$$\begin{aligned} & T_{11} + T_{12} = T_{21} + T_{22} \\ \iff & 1 - (c-1)a_1 + a_2 = a_1 + 1 - (c-1)a_2 \\ \implies & a_1 = a_2. \end{aligned}$$

We can similarly conclude $a_1 = a_2 = \ldots = a_c$. Overall T must take the form, e.g.

$$\begin{bmatrix} 1-(c-1)a & a & a & a \\ a & 1-(c-1)a & a & a \\ a & a & 1-(c-1)a & a \\ a & a & a & 1-(c-1)a \end{bmatrix}$$

which corresponds to symmetric label noise at rate $\eta = (c-1)a$. The condition that $T_{ii} > T_{ki}$ (for $k \neq i$) implies $\frac{1}{c} > a \iff \eta < \frac{c-1}{c}$.

($\Longleftarrow$) We now look at the converse - showing that symmetric label noise satisfies the desired class-preserving condition for all $\boldsymbol{p} \in \Delta$. The k^{th} component of $T\boldsymbol{p}$ can be written

$$\begin{aligned} (T\boldsymbol{p})_k &= (1-\eta)p_k + \sum_{i \neq k} \frac{\eta}{c-1} p_i \\ &= (1-\eta)p_k + (1-p_k)\frac{\eta}{c-1} \\ &= p_k \left(1 - \frac{c}{c-1}\eta\right) + \frac{\eta}{c-1} \end{aligned}$$

Hence, when $\eta < \frac{c}{c-1}$ the ordering of the p_k is preserved by the application of T, and so in particular, the argmax of the vector $\boldsymbol{p}$ is conserved. □

Corollary B.1.5. *A label noise model is class-preserving for all distributions $p(x,y)$ if and only if it describes (possibly non-uniform) label noise where, for all $x \in \text{supp}(p(x))$, the noise rate satisfies $\eta(x) < \frac{c-1}{c}$.*

Proof. The proof follows immediately from Lemma 4.3.2. For label noise to be class-preserving at a specified location $x \in \text{supp}(p(x))$, we require that $T(x)$ describes symmetric label noise at a rate $\eta(x) < \frac{c-1}{c}$. □

Appendix C

Risk Bounding

C.1 Proofs

Lemma C.1.1. *The GCE, SCE and FCE losses can be formulated as generalised forward-correction losses with a proper base loss. The noise models $f_{GCE}, f_{SCE}, f_{FCE}$ satisfy*

$$
\begin{aligned}
(f_{GCE}^{-1}(\boldsymbol{p}))_i &= \frac{p_i^{\frac{1}{1-a}}}{\sum_{i=1}^{c} p_i^{\frac{1}{1-a}}}, \\
(f_{SCE}^{-1}(\boldsymbol{p}))_i &= \frac{p_i}{\lambda - A p_i}, \\
f_{FCE}(\boldsymbol{p}) &= T^{-1}\boldsymbol{p},
\end{aligned}
$$

where T is the invertible stochastic matrix used to define the correction, and λ is a constant selected to ensure the correct normalisation.

Proof Idea: Suppose that L is a proper loss, let $f : \Delta \to \Delta$ be injective noise-model, and consider the minimiser of the expected loss defined $L_f(\boldsymbol{q}, k) \coloneqq L(f(\boldsymbol{q}), k)$ at $\boldsymbol{p} \in \Delta$;

$$
\begin{aligned}
\operatorname*{arg\,min}_{\boldsymbol{q}\in\Delta} H(\boldsymbol{p}, \boldsymbol{q}) &= \operatorname*{arg\,min}_{\boldsymbol{q}\in\Delta} \sum_{i=1}^{c} p_i L_f(\boldsymbol{q}, i) \\
&= \operatorname*{arg\,min}_{\boldsymbol{q}\in\Delta} \sum_{i=1}^{c} p_i L(f(\boldsymbol{q}), i).
\end{aligned}
$$

Since L is proper, then we know this is minimised by $\boldsymbol{q}$ such that $f(\boldsymbol{q}) = \boldsymbol{p}$. In other words, *we can uncover the noise model f by finding the minimiser of the expected loss*. This is how we find f for each of the loss functions. The core idea of the following proof is to write out the expected loss for each loss function and, for each $\boldsymbol{p} \in \Delta$, to find $\arg\min_{\boldsymbol{q}\in\Delta} H(\boldsymbol{p},\boldsymbol{q})$. Assuming that this arg min consists of a single point, then this induces a map $f(\boldsymbol{p}) := \arg\min_{\boldsymbol{q}\in\Delta} H(\boldsymbol{p},\boldsymbol{q})$ which, for the reasons given, can be identified with the noise model.

Proof. We begin by introducing the following notation: Let L be an elementwise loss and let $\boldsymbol{p},\boldsymbol{q}$ be two distributions, we denote the expected loss of $\boldsymbol{q}$ with respect to $\boldsymbol{p}$ to be $H_L(\boldsymbol{q},\boldsymbol{p}) := \sum_{i=1}^{c} p_i L(\boldsymbol{q},i)$.

Let us begin by considering GCE. The expected loss may be written $L_{GCE}(\boldsymbol{q},\boldsymbol{p}) := \sum_{i=1}^{c} p_i L_{GCE}(\boldsymbol{q},i) := \sum_{i=1}^{c} p_i \frac{1-q_i^a}{a}$. We find the minima by constructing the Langrangian $A(\boldsymbol{q},\lambda) := \sum_{i=1}^{c} p_i \frac{1-q_i^a}{a} + \lambda(\sum_{i=1}^{c} q_i - 1)$. By taking partials and equating to zero, we obtain $q_i^{1-a} = \frac{ap_i}{\lambda}, \forall i$. Using the fact that $\sum_{i=1}^{c} q_i = 1$ one may find the value of λ. Specifically, $\lambda = a(\sum_{i=1}^{c} p_i^{\frac{1}{1-a}})^{1-a}$. Thus overall one has $q_i^* = (\frac{ap_i}{\lambda})^{\frac{1}{1-a}} = \frac{p_i^{\frac{1}{1-a}}}{\sum_{i=1}^{c} p_i^{\frac{1}{1-a}}}$. Let us repeat this for the SCE loss. The expected loss may be written $L_{SCE}(\boldsymbol{q},\boldsymbol{p}) := \sum_{i=1}^{c} p_i L_{SCE}(\boldsymbol{q},i) := \sum_{i=1}^{c} p_i(A(1-q_i) - log(q_i))$. As before, we construct the relevant Lagrangian and find the stationary points: $B(\boldsymbol{q},\lambda) := \sum_{i=1}^{c} p_i(A(1-q_i) - log(q_i)) + \lambda(\sum_{i=1}^{c} q_i - 1)$. Taking partials and equating to zero we obtain $p_i(A + \frac{1}{q_i}) = \lambda \implies q_i^* = \frac{p_i}{\lambda - Ap_i}$. Here the value of the normalisation constant λ cannot be found in closed form for high values of c and must be computed numerically. Finally, we consider the forward-corrected CE loss. We assume that the loss is corrected by some invertible stochastic matrix T. $L_F(\boldsymbol{q},\boldsymbol{p}) := \sum_{i=1}^{c} p_i L_F(\boldsymbol{q},i) := \sum_{i=1}^{c} -p_i log((T\boldsymbol{q})_i)$. We remark that since CE is proper that this is minimised on the simplex by $\boldsymbol{p} = T\boldsymbol{q}^* \iff \boldsymbol{q}^* = T^{-1}\boldsymbol{p}$. For each loss, the function f obtained is injective as desired. □

C.1.1 Entropy Bounds

Lemma C.1.2. *Let L_f be a generalised-correction-loss whose 'base-loss' L is strictly proper (Recall the definition of 'base-loss' from Definition 5.2.1). The noisy risk of any probability estimator $\boldsymbol{q}$ is lower bounded:*

$$R_{L_f}^{\eta}(\boldsymbol{q}) \geq \mathbb{E}_{x\sim p(x)}[\mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y}|x))], \tag{C.1}$$

where $\mathcal{H}$ is the entropy function of the base-loss. This bound is tight when f equals the true noise model. Equality is attained by setting $\boldsymbol{q}(x) = f^{-1}(\widetilde{\boldsymbol{p}}(\widetilde{y}|x))$.[1]

Proof. Recollect that L_f is a generalised forward-correction loss with (strictly) proper base loss L: $L_f(\boldsymbol{q},k) = L(f(\boldsymbol{q},k))$. Let x be some arbitrary point in the support of $p(x)$ and let $\boldsymbol{q}(x)$ be some probability estimator. The pointwise noisy risk of $\boldsymbol{q}$ at x may be written as

$$\begin{aligned} R^{\eta}_{L_f}(\boldsymbol{q})(x) &:= \sum_{i=1}^{c} \widetilde{p}(\widetilde{y}=i|x) L_f(\boldsymbol{q}(x),i) \\ &= \sum_{i=1}^{c} \widetilde{p}(\widetilde{y}=i|x) L(f(\boldsymbol{q}(x)),i) \\ &\geq \sum_{i=1}^{c} \widetilde{p}(\widetilde{y}=i|x) L(\widetilde{\boldsymbol{p}}(\widetilde{y}|x),i) \\ &=: \mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y}|x)) \end{aligned}$$

The inequality follows from the definition of the properness of L. Inequality 5.4 follows by taking expectation with respect to $p(x)$ on both sides. Equality is attained setting by $f(\boldsymbol{q}(x)) = \widetilde{\boldsymbol{p}}(\widetilde{y}|x))$ for each x, (equivalently $f^{-1}(\boldsymbol{q}(x)) = \widetilde{\boldsymbol{p}}(\widetilde{y}|x))$ which is possible when f is the true noise model as then $\widetilde{\boldsymbol{p}} \in f(\Delta)$. The injectivity of f (as specified in the definition of $f-$proper) means this occurs uniquely at $\boldsymbol{q}(x) = f(\widetilde{\boldsymbol{p}}(\widetilde{y}|x))$ as desired. □

Lemma C.1.3 (Class-Conditional Label Noise)**.** *When the classes are balanced, and label noise is asymmetric and given by transition matrix T, the noisy risk of a probability estimator $\boldsymbol{q}$ may be lower bounded as follows,*

$$R^{\eta}_{L_f}(\boldsymbol{q}) \geq \frac{1}{c} \sum_{i=1}^{c} \mathcal{H}(\boldsymbol{T}_{\cdot,i}), \tag{C.2}$$

where $\boldsymbol{T}_{\cdot,i}$ denotes the i^{th} column of the matrix T.

Proof. The right-hand side of Inequality 5.4 can be written

$$\begin{aligned} \mathbb{E}_{x\sim p(x)}\left[\mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y}|x))\right] &= \mathbb{E}_{x\sim p(x)}\left[\sum_{i=1}^{c} \widetilde{p}(\widetilde{y}=i|x) L(\widetilde{\boldsymbol{p}}(\widetilde{y}|x),i)\right] \\ &= \mathbb{E}_{x\sim p(x)}\left[T\boldsymbol{p}(y|x) \cdot \boldsymbol{L}(T\boldsymbol{p}(y|x))\right] \\ &= \frac{1}{c}\sum_{k=1}^{c} (T\boldsymbol{e}_k) \cdot \boldsymbol{L}(T\boldsymbol{e}_k), \end{aligned}$$

[1] Note that if f is the true noise model then $\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x) \in f(\Delta)$ and the inverse is unique by injectivity.

where the final equality comes from using the fact that classes are balanced and all points are anchor points. This is equal to

$$\frac{1}{c}\sum_{k=1}^{c} T_{\cdot,k}\cdot \boldsymbol{L}(T_{\cdot,k}) = \frac{1}{c}\sum_{k=1}^{c} \mathcal{H}(\boldsymbol{T}_{\cdot,k}),$$

as desired. □

Corollary C.1.4 (Uniform Symmetric Label Noise)**.** *Given uniform, symmetric label noise at rate* η*, the risk associated with any probability estimator can be bounded as follows:*

$$R^{\eta}_{L_f}(\boldsymbol{q}) \geq \mathcal{H}\Big(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\Big). \tag{C.3}$$

This can be written equivalently as

$$R^{\eta}_{L_f}(\boldsymbol{q}) \geq \boldsymbol{u}_{sym}(\eta, c)\cdot \boldsymbol{L}(\boldsymbol{u}_{sym}(\eta, c)).$$

where

$$\boldsymbol{u}_{sym}(\eta, c) := \left(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\right). \tag{C.4}$$

Proof. When label is symmetric every column of the matrix T is a permutation of $\big(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\big)$. The result follows immediately from the symmetry assumption on the entropy. □

Lemma C.1.5 (Non-Uniform Symmetric Label Noise)**.** *Let* $p(x,y)$ *be a separable distribution, and let* $\widetilde{p}(x,\widetilde{y})$ *be a noisy distribution obtained by applying non-uniform symmetric label noise to* $p(x,y)$*. Assume that L is a generalised forward-correction loss and let* $\mathcal{H}$ *denote the (symmetric) entropy function of its base loss. For any probability estimator* $\boldsymbol{q}$*, we have the following lower bound on its noisy risk,*

$$R^{\eta}_{L_f}(\boldsymbol{q}) \geq \mathbb{E}_{x\sim p(x)}\left[\mathcal{H}\Big(1-\eta(x), \frac{\eta(x)}{c-1}, \frac{\eta(x)}{c-1}, \ldots, \frac{\eta(x)}{c-1}\Big)\right],$$

where $\eta(x)$ *denotes the noise rate at x. This inequality is strict and may be obtained by setting* $\boldsymbol{q}(x) = f^{-1}(\widetilde{\boldsymbol{p}}(y \mid x))$*, if* $\widetilde{\boldsymbol{p}}(y \mid x) \in f(\Delta)$*.*

Proof. The right-hand side of Inequality 5.4 can be written

$$\begin{aligned}\mathbb{E}_{x\sim p(x)}\left[\mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y}|x))\right] &= \mathbb{E}_{x\sim p(x)}\left[\sum_{i=1}^{c}\widetilde{p}(\widetilde{y}=i|x)L(\widetilde{\boldsymbol{p}}(\widetilde{y}|x), i)\right] \\ &= \mathbb{E}_{x\sim p(x)}\left[T(x)\boldsymbol{p}(y|x)\cdot \boldsymbol{L}(T(x)\boldsymbol{p}(y|x))\right].\end{aligned}$$

Our separability assumption means that $\boldsymbol{p}(y|x) = \boldsymbol{e}_k$ for some k. For each x it follows that $T(x)\boldsymbol{p}(y \mid x)$ is some rearrangement of the vector $(1-\eta(x), \frac{\eta(x)}{c-1}, \frac{\eta(x)}{c-1}, \ldots, \frac{\eta(x)}{c-1})$. By the assumption that the entropy function is symmetric, we may conclude that

$$\mathbb{E}_{x\sim p(x)}\left[\mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y}|x))\right] = \mathbb{E}_{x\sim p(x)}\left[\mathcal{H}\left(1-\eta(x), \frac{\eta(x)}{c-1}, \frac{\eta(x)}{c-1}, \ldots, \frac{\eta(x)}{c-1}\right)\right].$$

□

C.1.1.1 The General Case

Lemma C.1.6. *Let L_f be a generalised forward-corrected loss function whose base-loss L has entropy function $\mathcal{H}$. Suppose that label noise is applied to a separable data-label distribution and let $x \sim p(x)$. Given that the noise rate at x is $\eta(x)$, the entropy of the noisy label distribution at x, $\mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x))$ must lie in the following interval:*

$$\left[\mathcal{H}(\boldsymbol{u}_{pair}(\eta(x), c)), \mathcal{H}(\boldsymbol{u}_{sym}(\eta(x), c))\right].$$

In particular, for a fixed noise rate $\eta(x)$, the highest entropy occurs under symmetric label noise at x, while the lowest entropy is observed with pairwise label noise.

Proof. Let $\boldsymbol{q}(x)$ be a probability estimator and let x be some point in the support of $p(x)$. We established in the proof of Lemma 5.4.2 that $R_L^{\eta}(\boldsymbol{q})(x) \geq \mathcal{H}(\widetilde{\boldsymbol{p}}(\widetilde{y}|x))$. We have equality (uniquely) when $\boldsymbol{q}(x) = \widetilde{\boldsymbol{p}}(\widetilde{y}|x)$. Let $T(x)$ denote the noising transition matrix at x. By the separability assumption, we have some k such that $p(y=k|x) = 1$ and $p(y=i|x) = 0$ otherwise. Thus $\widetilde{p}(\widetilde{y}|x) = \sum_{y=1}^{c} \widetilde{p}(\widetilde{y}|y,x)p(y|x) = \widetilde{p}(\widetilde{y}|y=k,x) = (T_{1k}(x), T_{2k}(x), \ldots, T_{ck}(x))$. Let $A(\eta(x), c) := \mathcal{H}(T_{1k}(x), T_{2k}(x), \ldots, T_{ck}(x))$ where $\eta(x) := 1 - T_{kk}$ is the noise rate at x. The symmetry of $\mathcal{H}$ means that, without loss of generality, we may let $k = 1$. It remains to show that $A(\eta(x), c) \in \left[\mathcal{H}(1-\eta(x), \eta(x), 0, 0, \ldots, 0), \mathcal{H}\left(1-\eta(x), \frac{\eta(x)}{c-1}, \frac{\eta(x)}{c-1}, \ldots, \frac{\eta(x)}{c-1}\right)\right]$.

Upper Limit: We begin by demonstrating that $A(\eta(x), c)$ is upper bounded by $\mathcal{H}(1-\eta(x), \frac{\eta(x)}{c-1}, \frac{\eta(x)}{c-1}, \ldots, \frac{\eta(x)}{c-1})$. Let $\Delta(\eta(x))$ denote the set of non-negative vectors $(a_1, a_2, \ldots, a_{c-1})$ such that $a_i \leq 1$ and $\sum_{i=1}^{c-1} a_i = \eta(x)$. We wish to show the supremum of $\mathcal{H}(1-\eta(x), a_1, a_2, \ldots, a_{c-1})$ is attained on $\Delta(\eta(x))$ by setting $a_i = \frac{\eta(x)}{c-1}$ for all i. This corresponds to the label noise being symmetric at x. By Theorem 2.1.3 $\mathcal{H}$ is a (strictly) concave function. Moreover, the symmetry assumption implies that $\mathcal{H}$ is a symmetric function of its variables. Define the function $g(a_1, a_2, \ldots, a_{c-1}) := \mathcal{H}(1-\eta(x), a_1, a_2, \ldots, a_{c-1})$. We wish to show that g attains its maximum on $\Delta(\eta(x))$

when $a_i = a_j$ for all i, j. We begin by noting that the (strict) concavity of $\mathcal{H}$ implies the (strict) concavity of g. To see this consider two arbitrary vectors $\boldsymbol{x} = (x_1, x_2, \ldots x_{c-1}), \boldsymbol{y} = (y_1, y_2, \ldots y_{c-1})$. Now $g(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}) = \mathcal{H}(\lambda\boldsymbol{x}' + (1-\lambda)\boldsymbol{y}')$ where $\boldsymbol{x}' := (1-\eta(x), x_1, x_2, \ldots, x_{c-1})$ and $\boldsymbol{y}' := (1-\eta(x), y_1, y_2, \ldots, y_{c-1})$. Thus the concavity of $\mathcal{H}$ implies $g(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}) := \mathcal{H}(\lambda\boldsymbol{x}' + (1-\lambda)\boldsymbol{y}') \geq \lambda\mathcal{H}(\boldsymbol{x}') + (1-\lambda)\mathcal{H}(\boldsymbol{y}') = \lambda g(\boldsymbol{x}) + (1-\lambda)g(\boldsymbol{y})$ as desired. Thus, g is a symmetric (strictly) concave function of its variables.

Let $\boldsymbol{a}^*$ denote a maxima of g on $\Delta(\eta(x))$. Let σ denote the cyclic permutation of the components of $\boldsymbol{a}$. That is $\sigma(a_1, a_2, \ldots, a_{c-1}) := (a_{c-1}, a_1, a_2, \ldots, a_{c-2})$. By the symmetry of g, we know that if $\boldsymbol{a}^*$ is a maxima then so is $\sigma^i(\boldsymbol{a}^*)$ for all i: $g(\boldsymbol{a}^*) = g(\sigma^i(\boldsymbol{a}^*))$ for all $i \in \mathbb{N}$. The defining property of a concave function is that

$$g(\lambda_1\boldsymbol{v}_1 + \lambda_2\boldsymbol{v}_2 + \ldots + \lambda_d, \boldsymbol{v}_d) \geq \sum_{i=1}^{d} \lambda_i g(\boldsymbol{x}_i)$$
$$\text{where } \sum_i \lambda_i = 1.$$

Hence by the (strict) concavity of g, setting $\lambda_i := \frac{1}{c-1}$;

$$\begin{aligned} g\left(\frac{\eta(x)}{c-1}, \frac{\eta(x)}{c-1}, \ldots, \frac{\eta(x)}{c-1}\right) &= g\left(\frac{1}{c-1}(\boldsymbol{a}^* + \sigma(\boldsymbol{a}^*) + \sigma^2(\boldsymbol{a}^*) + \ldots + \sigma^{c-2}(\boldsymbol{a}^*))\right) \\ &\geq \frac{1}{c-1} g(\boldsymbol{a}^*) + \frac{1}{c-1} g(\sigma(\boldsymbol{a}^*)) + \ldots + \frac{1}{c-1} g\left(\sigma^{c-2}(\boldsymbol{a}^*)\right) \\ &= g(\boldsymbol{a}^*) \end{aligned}$$

Hence g is maximised by setting $a_i = \frac{\eta(x)}{c-1}$ for all i as desired. This is the unique maxima when the base loss strictly proper.

Lower Limit: It now remains to show that the lower bound on $A(\eta(x), c)$ holds. The (strict) concavity means that g attains it minima on the vertices of $\Delta(\eta(x))$ (eg $(\eta(x), 0, \ldots, 0)$. To see this let $\boldsymbol{a}^* = (a_1^*, a_2^*, \ldots, a_{c-1}^*)$ denote a minima of g on $\Delta(\eta(x))$. Then we have,

$$\begin{aligned} g(a_1^*, a_2^*, \ldots, a_{c-1}^*) &= g(a_1^*\boldsymbol{e_1} + a_2^*\boldsymbol{e_2} + \ldots + a_{c-1}^*\boldsymbol{e}_{c-1}) \\ &\geq \sum_{i=1}^{c-1} \frac{a_i^*}{\eta(x)} g(\eta(x)\boldsymbol{e_i}) \\ &= g(\eta(x), 0, \ldots, 0) \\ &= \mathcal{H}(1-\eta(x), \eta(x), 0, 0, \ldots, 0) \end{aligned} \tag{C.5}$$

$\boldsymbol{e}_i$ denotes the coordinate vector with 1 in the ith position and zeros elsewhere. Equation C.5 holds by the symmetry of g and since $\sum a_i^* = \eta(x)$. Thus we have shown that g is lower bounded by $\mathcal{H}(1-\eta(x), \eta(x), 0, 0, \ldots, 0)$ as desired. Moreover, this infimum is obtained on the vertices of $\Delta(\eta(x))$. □

Corollary C.1.7. *Given an average noise rate* $\eta := \mathbb{E}_{x\sim p(x)}[\eta(x)]$*, the greatest possible value of* $\mathbb{E}_{x\sim p(x)}[\mathcal{H}(\widetilde{p}(\widetilde{y} \mid x)]$ *occurs when* $\eta(x)$ *is constant:*

$$\sup_{p(\widetilde{y}|x,y)} \left(\mathbb{E}_{x\sim p(x)}[\mathcal{H}(\widetilde{p}(\widetilde{y} \mid x))]\right) = \mathcal{H}\left(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\right),$$

where the supremum is taken over all noise models such that $\mathbb{E}_{x\sim p(x)}[\eta(x)] = \eta$.

Proof. We established in the proof of Lemma 5.4.3 that, given that the noise rate at x is $\eta(x)$,

$$\mathcal{H}(\widetilde{p}(\widetilde{y} \mid x) \in \left[\mathcal{H}(1-\eta(x), \eta(x), 0, 0, \ldots, 0), \mathcal{H}\Big(1-\eta(x), \frac{\eta(x)}{c-1}, \frac{\eta(x)}{c-1}, \ldots, \frac{\eta(x)}{c-1}\Big)\right].$$

Thus, given a fixed average noise rate η, we maximise the expected entropy when the noise model describes symmetric label noise at each point in dataspace. We now wish to demonstrate that

$$\mathbb{E}_{x\sim p(x)}\left[\mathcal{H}\Big(1-\eta(x), \frac{\eta(x)}{c-1}, \frac{\eta(x)}{c-1}, \ldots, \frac{\eta(x)}{c-1}\Big)\right] \leq \mathcal{H}\left(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\right),$$

which is to say that we maximise the entropy of symmetric noise by setting $\eta(x) = const$. $\mathcal{H}$ is concave (strictly concave if the base loss is strictly proper) thus, we can use Jensen's Inequality, which tells us that

$$f(\mathbb{E}[X]) \geq \mathbb{E}[f(X)],$$

if f is concave. Hence, by setting our random variable $X := (1-\eta(x), \eta(x)/(c-1), \ldots, \eta(x)/(c-1))$, and $f = \mathcal{H}$ yields the desired result. □

C.2 Additional Theory and Discussion

C.2.1 Sensitivity of Bounds

The noise-bound is equal to the average entropy of the noisy label distribution when label noise is uniform and symmetric. When we deviate from these noise conditions, this bound is too high in that an optimal probability estimator could achieve a (noisy) risk lower than this value without overfitting. Since we use this bound in all noise

conditions, it is essential to understand the size of the gap between our bound and the minimum achievable risk. Ideally, we want this gap to be small. This section briefly examines this topic, noting that this gap is usually smaller for GCE and SCE than CE. This implies that the noise-bound is more suitably used with SCE and GCE than with CE when noise deviates from idealised assumptions. Given a noise rate η, the following Lemma gives the worst-case gap between the actual average entropy of the noisy distribution and the noise-bound, assuming uniform label noise.

Corollary C.2.1. *Suppose we have some uniform label noise at noise rate* η. *Let* $\mathcal{H}$ *denote the average entropy of the noisy label distribution, that is*

$$\mathcal{H} := \mathbb{E}_{x\sim p(x)}[\mathcal{H}(\widetilde{\boldsymbol{p}}(y \mid x))].$$

Let $B(\eta, c)$ *denote the noise-bound defined in Definition 5.4.5. Then*

$$|B(\eta,c) - \mathcal{H}| \leq \mathcal{H}\Big(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\Big) - \mathcal{H}(1-\eta, \eta, 0, 0, \ldots, 0)$$

Proof. This follows immediately from Lemma 5.4.3 when $\eta(x)$ has no dependence on x ($\eta(x) = \eta$). □

As discussed previously, when noise is uniform but not symmetric, our noise-bound (Definition 5.4.5) of $\mathcal{H}(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1})$ is too high since the true minimum achievable risk is lower than this bound. In other words, a probability estimator exists that attains a risk lower than our bound. This non-optimality is the cost we incur as a result of requiring a simple, easily computable bound depending on only on the noise rate. Importantly, Corollary C.2.1 gives us a rough way to quantify this non-optimality, using the difference between the upper and lower entropy limits

$$\left[\mathcal{H}\Big(1-\eta, \frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1}\Big), \mathcal{H}(1-\eta, \eta, 0, 0, \ldots, 0)\right] \tag{C.6}$$

When this difference is large, one can construct two types of label noise with the same rate η, such that the difference in the minimum achievable risks between these noise types is significant. Conversely, when this gap is small, the minimum achievable risk for any type of label noise at a fixed rate η is similar. This is a desirable property and suggests that simply setting our bound to our noise-bound is probably suitable regardless of the specifics of the noising process.

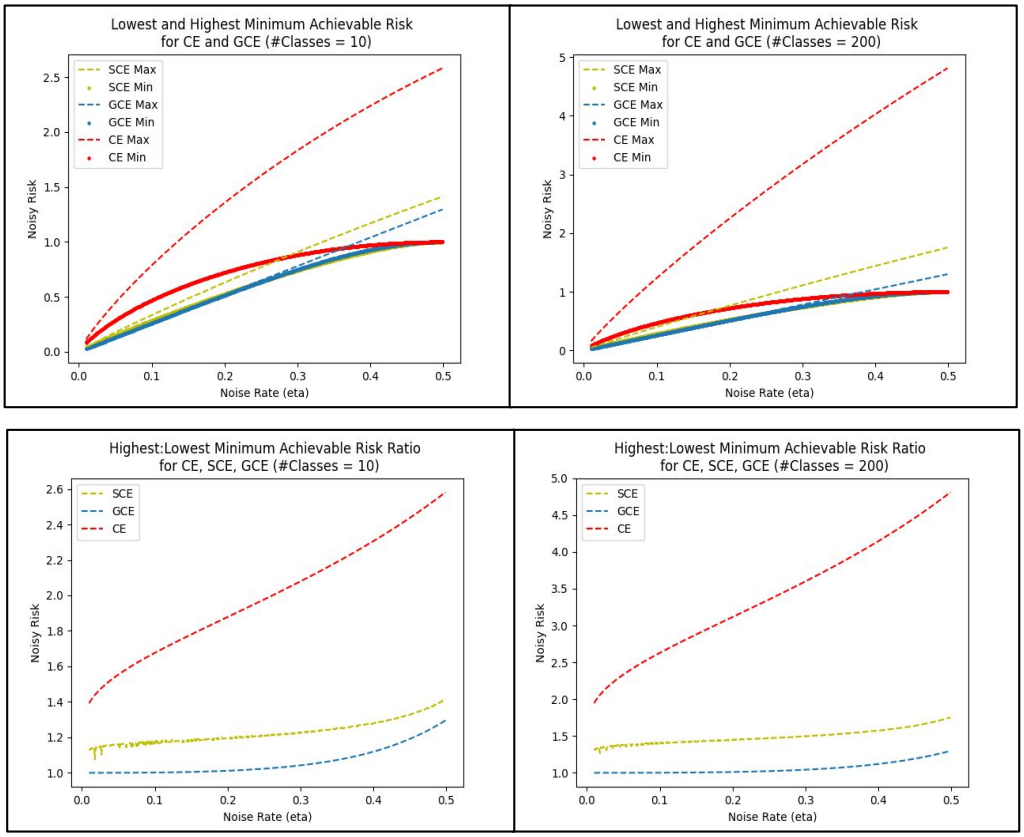


Figure C.1: On the top row, we plot the upper and lower limits of $A(\eta, c)$ for $\eta \in (0, 0.5]$ from Corollary C.2.1 for the CE (red), SCE (yellow) and GCE (blue) losses for ten classes (left) and 200 classes (right). On the bottom row, we plot a ratio of these upper and lower limits instead. We observe that the difference between these upper and lower limits is far greater for CE than the other losses. This is more pronounced for more classes.

On the top row of Figure C.1, we give a plot of the upper and lower limits of Equation C.6 for $\eta \in (0, 0.5]$ for $c = 10$ (left) and $c = 200$ (right) for GCE, SCE and CE. A dotted line gives the upper limit, while the lower limit is given by a filled line in the same colour. Each loss is scaled so they may be more easily compared. Similarly, in the row below, we plot the ratios of the upper and lower limits of Equation C.6 for each loss. These graphs show that the difference between the upper and lower limits is much greater for CE than for SCE and GCE. This difference is more pronounced when the number of classes is greater. The result is that on non-symmetric noise, our noise-bound (Definition 5.4.5) will generally be less suitable when used in conjunction with CE than when used with GCE or SCE.

C.2.2 Noise Model Plots

In Lemma 5.2.2, we showed that the SCE, GCE and FCE losses are generalised forward-correction losses and derived the corresponding functions f. (We derived f^{-1} as this was easier.) As discussed, these functions can be interpreted as noise models; $f(\boldsymbol{p}(y|x)) \approx \widetilde{\boldsymbol{p}}(\widetilde{y}|x)$. In section, we provide some plots of these noise models.

Properness While Definition 5.2.1 does not require the so-called base loss to be proper, Lemma 5.2.2 shows that GCE and SCE can be obtained by applying a non-linear correction to a proper loss. The defining characteristic of a proper loss is that the expected loss is minimised by setting $\boldsymbol{p} = \boldsymbol{q}$. Therefore,

$$H_{L_f}(\widetilde{\boldsymbol{p}}, \boldsymbol{q}) \coloneqq \widetilde{\boldsymbol{p}} \cdot L_f(\boldsymbol{q}) = \widetilde{\boldsymbol{p}} \cdot L(f(\boldsymbol{q})).$$

is minimised by setting $\widetilde{\boldsymbol{p}} = f(\boldsymbol{q}) \iff \boldsymbol{q} = f^{-1}(\widetilde{\boldsymbol{p}})$. We make this point because plotting f^{-1} as a function of $\boldsymbol{p}$ (which we do below) is the same as plotting $\arg\min_q H(\boldsymbol{p}, \boldsymbol{q})$ - this allows us to include the MAE loss on this plot even though it isn't a generalised forward-correction loss.

Plots In Figure 5.2, we present plots of f^{-1} for the SCE, GCE and FCE loss functions in the binary setting. The $x-$axis gives the probability of a noisy label being equal to one $\widetilde{p}(\widetilde{y} = 1 \mid x)$. On the $y-$axis we plot $p(y = 1 \mid x)$ where $\boldsymbol{p} = f^{-1}(\widetilde{\boldsymbol{p}})$. For proper losses, $f = id$, reflecting that they contain encode no noise model. The graphs for GCE and SCE are remarkably similar. Their graphs portray a noise model where label noise occurs more frequently at points where $\boldsymbol{p}$ contains higher intrinsic uncertainty. Conversely, no label noise occurs at anchor points. FCE requires a noise model to be fully specified; we assume symmetric label noise at $\eta = 0.4$. Varying η will change the steepness of the respective f^{-1}. Finally, we plot MAE. The graphs of SCE and GCE lie between those of MAE and CE. By varying the parameters of these losses, we can interpolate between them.

C.3 Further Experiments

In Chapter 5 we presented results for seven noisy datasets. Experiment results for two additional datasets, TinyImageNet and Animals are given in Table C.4.

C.3.1 Experiment Details

The number of training epochs was the same for each loss. For MNIST, FashionMNIST, TinyImageNet and Animals10N, we used 100 epochs; for all other datasets, we used 120 epochs. Each experiment in Tables 5.2,5.3 was run three times, and the mean and unbiased estimate of the standard deviation is given. We used a ResNet18 architecture for all experiments except TinyImageNet and Animals10N, where a ResNet34 was used. Each experiment is carried out on a single GeForce GTX Titan X. We used a batch size of 300 in all experiments except TinyImageNet and Animals10N, where this is reduced to 200. A learning rate of 0.0001 was used for all losses except MAE (lr = 0.001) and ELR, where we used their recommended learning rate of 0.01. We use a learning rate scheduler, which scales our learning rate by 0.6 at epoch 60. Our implementation of the *Truncated Loss* comes from the official GitHub implementation of GCE. Likewise, we use the official codebase for our implementation of ELR. Other losses are re-implementations based on details given in the respective papers. Our SCE loss used the recommended hyperparameter of $A = 8$. Our GCE loss used $a = 0.4$. FCE requires one to define a noise model. In each case, we assume noise is symmetric at the relevant rate. For Animals10N, this rate is set to 11%; the estimated noise rate.

C.3.1.1 CE with Prior

Cross-entropy with a 'prior' term (CEP) is one of the losses used in our experiments. We explain the motivation for this additional loss term and provide details on how it's implemented.

In Section 5.4.2 we assumed that the un-noised distribution $p(x,y)$ is separable (i.e. for each x, $p(y = k \mid x) = 1, p(i \neq k \mid x) = 0)$ for some $k \in \mathcal{Y}$. Thus, in the case of symmetric noise with a known noise rate η, the noisy label distribution $\widetilde{\boldsymbol{p}}(\widetilde{y}|x)$ is of the form for each x:

$$\widetilde{\boldsymbol{p}}(\widetilde{y} \mid x) = \left(\frac{\eta}{c-1}, \frac{\eta}{c-1}, \ldots, \underbrace{1-\eta}_{\text{k}^{\text{th}} \text{ position}}, \ldots, \frac{\eta}{c-1} \right) \tag{C.7}$$

Introducing a term to penalise our model when its outputs deviate from this distribution is reasonable. This is achieved through a regularisation term which measures the KL-divergence between our model probabilities and the desired distribution (Equation C.7). Let $\boldsymbol{p}_\eta := (p_1, p_2, \ldots, p_c) := (1-\eta, \frac{\eta}{c-1}, \ldots, \frac{\eta}{c-1})$ and let $q_1, q_2, \ldots, q_c$ denote the probabilities output by our model. We sort the q_i into descending order (which we denote as

$q_{\sigma(i)}$) and define our prior term as:

$$L_{prior}(\boldsymbol{q}, \boldsymbol{p}_\eta) := -\sum_{i=1}^{c} p_i \log(q_{\sigma(i)}) \tag{C.8}$$

Thus, overall we have $L_{CEP}(\boldsymbol{q}, i) := L_{CE}(\boldsymbol{q}, i) + L_{prior}(\boldsymbol{q}, \boldsymbol{p}_\eta)$. Tables 1 and 2 in Section 6 show that this additional term generally results in additional improvement over using the noise-bound alone. This prior acts as a feasible set reduction method: many different probability estimators achieve a training error equal to our noise-bound. Therefore, by introducing a prior term (Equation C.8), we can further restrict the set of admissible models.

C.3.2 Varying The Bound

We explore treating the bound 'B' as a hyperparameter to assess the proximity of the noise-bound to optimality. This consists of doing a grid search near the noise-bound for each loss function and recording how this impacts clean test performance. Tables C.2, C.3 include the result of these experiments, indicated with a star (e.g. CE+B*), together with the results of the other loss functions. When varying the bound from the noise-bound does not yield an improvement, the starred and unstarred accuracy values are the same. In slightly over half of our experiments, we find that we may achieve an improvement by perturbing the bound. This improvement is generally minor. Our assumption that the underlying clean dataset is separable means one should be able to improve performance by raising the bound to account for the additional randomness in the label distributions. Generally, we find this to be so. An exception to this pattern are the non-uniform and asymmetric datasets. In these cases, one typically benefits from marginally *lowering* the bound. This observation is consistent with our expectation; the noise-bound is a 'worst-case' entropy, attained only by uniform symmetric label noise. For other noise models, the noise-bound will be higher than strictly necessary to prevent overfitting and may benefit from being slightly decreased. The values of the optimal bounds may be found in a Table C.1.

C.3.2.1 Optimal Bounds

In our experiment tables in Section C.3.2, we give results using our noise-bounds. We additionally give results where the bound is treated as a hyperparameter. We do not search over the entire space; rather, we do a grid search near the noise-bound. For MNIST, FashionMNIST, EMNIST, CIFAR10 and CIFAR100, we search over

	MNIST		Fashion		EMNIST		CIFAR10		CIFAR100		ACIFAR100		NU-EMNIST
	0.4	0.6	0.2	0.4	0.2	0.4	0.2	0.4	0.2	0.4	0.2	0.4	0.6
FCE	-0.05	-0.05	0.0	-0.05	0.05	0.05	0.05	0.1	0.03	0.0	-0.1	-0.35	-0.2
GCE	0.0	0.0	0.05	0.0	0.03	0.05	0.05	0.05	0.05	0.0	0.05	0.05	0.0
SCE	0.0	0.05	0.0	0.2	0.2	0.1	0.0	0.2	0.2	0.2	0.0	0.0	0.0
CEB	0.0	0.0	0.0	0.0	0.05	0.05	0.05	0.05	0.1	0.1	0.2	0.0	-0.6
CEP	-0.15	-0.15	-0.15	-0.15	0.0	0.02	0.0	0.0	-0.08	-0.08	-0.08	-0.08	-0.1

Table C.1: table giving the offset of the 'optimal' bound from the noise-bound. A negative (blue) number means the bound is greater than the noise-bound. Positive (red) means the optimal bound is lower. Grey means that the optimal bound is zero, i.e. no offset.

Losses	MNIST		FashionMNIST		EMNIST				CIFAR10	
					0.2		0.4			
	0.4	0.6	0.2	0.4	Top 1	Top 5	Top 1	Top 5	0.2	0.4
MSE	93.3$_{\pm 0.47}$	85.8$_{\pm 0.95}$	84.8$_{\pm 0.22}$	80.6$_{\pm 0.84}$	82.9$_{\pm 0.29}$	98.1$_{\pm 0.04}$	80.2$_{\pm 0.19}$	97.1$_{\pm 0.07}$	78.7$_{\pm 1.51}$	56.4$_{\pm 0.11}$
MAE	97.9$_{\pm 0.08}$	96.4$_{\pm 0.08}$	83.2$_{\pm 0.10}$	82.2$_{\pm 0.37}$	49.8$_{\pm 2.83}$	52.2$_{\pm 0.10}$	50.4$_{\pm 1.14}$	51.4$_{\pm 0.96}$	88.6$_{\pm 1.34}$	78.9$_{\pm 5.95}$
NCE	97.8$_{\pm 0.06}$	96.0$_{\pm 0.25}$	87.7$_{\pm 0.26}$	**86.3**$_{\pm 0.14}$	84.5$_{\pm 0.25}$	97.9$_{\pm 0.05}$	82.6$_{\pm 0.81}$	96.7$_{\pm 0.03}$	**89.3**$_{\pm 0.40}$	**86.0**$_{\pm 0.81}$
MixUp	95.8$_{\pm 1.24}$	86.8$_{\pm 0.85}$	86.9$_{\pm 0.10}$	82.3$_{\pm 0.54}$	84.3$_{\pm 0.08}$	98.1$_{\pm 0.04}$	81.6$_{\pm 0.48}$	97.1$_{\pm 0.08}$	86.0$_{\pm 0.46}$	77.9$_{\pm 0.49}$
Spher.	95.0$_{\pm 0.41}$	88.1$_{\pm 0.82}$	87.2$_{\pm 0.04}$	84.1$_{\pm 0.75}$	84.6$_{\pm 0.12}$	98.3$_{\pm 0.05}$	83.2$_{\pm 0.29}$	98.1$_{\pm 0.58}$	86.6$_{\pm 0.01}$	72.1$_{\pm 0.80}$
Boot.	86.6$_{\pm 0.56}$	71.2$_{\pm 1.17}$	82.0$_{\pm 0.61}$	73.4$_{\pm 1.06}$	80.5$_{\pm 0.24}$	96.7$_{\pm 0.06}$	77.3$_{\pm 0.98}$	95.0$_{\pm 0.25}$	77.0$_{\pm 1.57}$	58.2$_{\pm 2.99}$
Trunc.	97.1$_{\pm 0.12}$	94.2$_{\pm 0.39}$	87.8$_{\pm 0.29}$	85.3$_{\pm 0.77}$	84.1$_{\pm 0.53}$	97.4$_{\pm 1.03}$	83.1$_{\pm 0.55}$	97.2$_{\pm 1.00}$	88.3$_{\pm 0.56}$	84.2$_{\pm 0.69}$
CL	82.7$_{\pm 0.57}$	67.5$_{\pm 1.83}$	81.2$_{\pm 0.34}$	73.1$_{\pm 0.66}$	79.6$_{\pm 0.17}$	96.4$_{\pm 0.05}$	75.1$_{\pm 0.67}$	94.2$_{\pm 0.24}$	76.0$_{\pm 2.16}$	59.4$_{\pm 4.20}$
ELR	98.1$_{\pm 0.04}$	97.8$_{\pm 0.07}$	85.3$_{\pm 0.23}$	83.4$_{\pm 0.02}$	81.8$_{\pm 0.26}$	97.5$_{\pm 0.21}$	76.6$_{\pm 0.10}$	96.5$_{\pm 0.11}$	88.1$_{\pm 0.82}$	85.7$_{\pm 0.06}$
FCE.	95.4$_{\pm 0.25}$	92.3$_{\pm 0.13}$	83.6$_{\pm 0.11}$	79.9$_{\pm 0.78}$	83.1$_{\pm 0.12}$	98.4$_{\pm 0.20}$	80.6$_{\pm 0.12}$	98.0$_{\pm 0.03}$	84.7$_{\pm 0.40}$	75.1$_{\pm 0.04}$
FCE+B	95.7$_{\pm 0.18}$	92.7$_{\pm 0.74}$	84.8$_{\pm 0.26}$	81.7$_{\pm 0.27}$	83.4$_{\pm 0.09}$	98.5$_{\pm 0.03}$	81.6$_{\pm 0.51}$	**98.1**$_{\pm 0.15}$	86.7$_{\pm 0.21}$	82.2$_{\pm 0.06}$
FCE+B*	96.7$_{\pm 0.17}$	94.3$_{\pm 0.50}$	84.8$_{\pm 0.26}$	83.3$_{\pm 0.22}$	84.4$_{\pm 0.06}$	**98.6**$_{\pm 0.13}$	83.1$_{\pm 0.42}$	98.1$_{\pm 0.10}$	87.2$_{\pm 0.20}$	82.2$_{\pm 0.06}$
GCE	94.4$_{\pm 0.36}$	83.8$_{\pm 1.14}$	86.4$_{\pm 0.24}$	81.6$_{\pm 0.37}$	84.3$_{\pm 0.13}$	98.4$_{\pm 0.08}$	82.7$_{\pm 0.07}$	97.9$_{\pm 0.02}$	81.1$_{\pm 0.72}$	60.0$_{\pm 1.31}$
GCE+B	96.6$_{\pm 0.22}$	94.0$_{\pm 0.13}$	86.5$_{\pm 0.56}$	85.5$_{\pm 0.13}$	84.1$_{\pm 0.29}$	98.4$_{\pm 0.04}$	82.8$_{\pm 0.28}$	98.0$_{\pm 0.06}$	86.1$_{\pm 0.22}$	79.0$_{\pm 1.17}$
GCE+B*	96.6$_{\pm 0.22}$	94.0$_{\pm 0.13}$	87.0$_{\pm 0.04}$	85.5$_{\pm 0.13}$	84.3$_{\pm 0.09}$	98.4$_{\pm 0.06}$	83.6$_{\pm 0.25}$	**98.2**$_{\pm 0.03}$	86.7$_{\pm 0.07}$	80.2$_{\pm 0.83}$
SCE	89.5$_{\pm 5.29}$	70.2$_{\pm 0.69}$	82.7$_{\pm 0.64}$	74.4$_{\pm 0.37}$	82.1$_{\pm 0.33}$	96.8$_{\pm 0.10}$	79.6$_{\pm 0.61}$	95.4$_{\pm 0.15}$	78.2$_{\pm 0.42}$	59.0$_{\pm 4.43}$
SCE+B	97.0$_{\pm 0.16}$	93.4$_{\pm 0.29}$	87.5$_{\pm 0.22}$	85.2$_{\pm 0.98}$	83.5$_{\pm 0.29}$	97.3$_{\pm 0.14}$	81.8$_{\pm 0.52}$	96.4$_{\pm 0.20}$	88.9$_{\pm 0.44}$	84.7$_{\pm 0.37}$
SCE+B*	97.0$_{\pm 0.16}$	93.7$_{\pm 0.52}$	87.5$_{\pm 0.22}$	85.8$_{\pm 0.67}$	83.6$_{\pm 0.03}$	97.4$_{\pm 0.02}$	81.8$_{\pm 0.52}$	96.5$_{\pm 0.26}$	88.9$_{\pm 0.44}$	84.9$_{\pm 0.20}$
CE	80.8$_{\pm 2.31}$	67.3$_{\pm 0.80}$	80.9$_{\pm 1.11}$	72.1$_{\pm 2.16}$	79.9$_{\pm 0.28}$	96.4$_{\pm 0.08}$	75.6$_{\pm 0.20}$	94.2$_{\pm 0.24}$	76.9$_{\pm 1.22}$	59.9$_{\pm 2.15}$
CE+B	96.2$_{\pm 0.32}$	93.0$_{\pm 0.09}$	87.9$_{\pm 0.10}$	84.7$_{\pm 0.37}$	80.8$_{\pm 0.08}$	97.0$_{\pm 0.04}$	78.9$_{\pm 0.12}$	96.1$_{\pm 0.26}$	84.5$_{\pm 0.73}$	76.0$_{\pm 1.13}$
CE+B*	96.2$_{\pm 0.32}$	93.0$_{\pm 0.09}$	87.9$_{\pm 0.10}$	84.7$_{\pm 0.37}$	81.5$_{\pm 0.11}$	97.3$_{\pm 0.02}$	79.0$_{\pm 0.09}$	96.2$_{\pm 0.01}$	84.8$_{\pm 0.55}$	78.6$_{\pm 1.28}$
CEP	97.5$_{\pm 0.08}$	92.1$_{\pm 0.44}$	87.8$_{\pm 0.12}$	84.8$_{\pm 0.23}$	85.5$_{\pm 0.10}$	98.1$_{\pm 0.07}$	84.3$_{\pm 0.22}$	97.6$_{\pm 0.14}$	84.2$_{\pm 0.51}$	58.2$_{\pm 2.94}$
CEP+B	95.6$_{\pm 0.32}$	85.5$_{\pm 0.77}$	88.1$_{\pm 0.31}$	84.2$_{\pm 0.33}$	**85.8**$_{\pm 0.12}$	98.3$_{\pm 0.02}$	**84.8**$_{\pm 0.10}$	98.0$_{\pm 0.04}$	88.5$_{\pm 0.32}$	85.1$_{\pm 0.20}$
CEP+B*	**98.5**$_{\pm 0.05}$	**97.9**$_{\pm 0.11}$	**88.4**$_{\pm 0.04}$	**87.2**$_{\pm 0.21}$	85.8$_{\pm 0.12}$	98.3$_{\pm 0.02}$	84.8$_{\pm 0.10}$	98.0$_{\pm 0.16}$	88.5$_{\pm 0.32}$	85.1$_{\pm 0.20}$

Table C.2: Test accuracies obtained using different losses on the noisy MNIST/ FashionMNIST/EMNIST/CIFAR10 datasets. Losses implementing the noise-bound shaded in blue. When this bound provides benefit, the corresponding value is $\boxed{boxed}$. Overall top values in **bold**.

$\{-0.2,-0.15,-0.1,\ldots,0.15,0.2\}$ where e.g. 0.2 means that we add 0.2 onto our noise-bound ($B(\eta,c)\mapsto B(\eta,c)+0.2$). For Asymmetric CIFAR100 (ACIFAR100) and Non-uniform EMNIST (NU-EMNIST), this range is broadened to $\{-0.6,-0.55,\ldots 0.55,0.6\}$. The bounds which give the best results are given in Table C.1. When the optimal bound

Losses	CIFAR100				ASYM-CIFAR100				Non-Uniform-EMNIST	
	0.2		0.4		0.2		0.4		0.6	
	Top1	Top5	Top1	Top5	Top1	Top5	Top1	Top5	Top 1	Top 5
MSE	57.2$_{\pm 0.93}$	78.6$_{\pm 0.25}$	40.6$_{\pm 0.38}$	63.0$_{\pm 0.24}$	56.3$_{\pm 0.11}$	82.6$_{\pm 0.22}$	40.7$_{\pm 0.12}$	74.4$_{\pm 0.25}$	44.7$_{\pm 2.66}$	86.7$_{\pm 3.10}$
MAE	10.0$_{\pm 0.11}$	13.8$_{\pm 0.28}$	7.6$_{\pm 1.89}$	11.6$_{\pm 1.25}$	7.1$_{\pm 6.02}$	11.1$_{\pm 6.6}$	11.1$_{\pm 5.43}$	25.1$_{\pm 5.76}$	9.8$_{\pm 1.74}$	23.1$_{\pm 1.80}$
NCE	38.7$_{\pm 3.13}$	51.8$_{\pm 3.77}$	19.1$_{\pm 0.20}$	28.8$_{\pm 0.15}$	16.3$_{\pm 1.24}$	25.4$_{\pm 1.80}$	21.8$_{\pm 1.24}$	37.2$_{\pm 1.80}$	18.0$_{\pm 1.17}$	38.8$_{\pm 1.93}$
MixUp	59.6$_{\pm 0.31}$	81.5$_{\pm 0.39}$	51.3$_{\pm 8.63}$	75.8$_{\pm 8.09}$	61.2$_{\pm 0.88}$	86.0$_{\pm 1.12}$	47.2$_{\pm 0.60}$	81.3$_{\pm 0.23}$	**52.4**$_{\pm 0.80}$	95.5$_{\pm 0.08}$
Spher.	57.7$_{\pm 0.18}$	82.9$_{\pm 0.54}$	48.8$_{\pm 0.51}$	74.3$_{\pm 0.73}$	54.2$_{\pm 0.32}$	81.2$_{\pm 0.29}$	39.2$_{\pm 0.31}$	72.1$_{\pm 0.15}$	41.9$_{\pm 0.10}$	94.4$_{\pm 0.04}$
Boot.	54.0$_{\pm 0.37}$	76.4$_{\pm 0.39}$	37.7$_{\pm 0.89}$	60.9$_{\pm 1.52}$	56.0$_{\pm 0.34}$	83.8$_{\pm 0.03}$	43.2$_{\pm 0.35}$	78.3$_{\pm 0.20}$	49.1$_{\pm 0.29}$	95.3$_{\pm 0.42}$
Trunc.	58.1$_{\pm 0.36}$	82.7$_{\pm 0.37}$	50.9$_{\pm 1.17}$	77.2$_{\pm 0.59}$	56.3$_{\pm 0.62}$	82.3$_{\pm 0.61}$	45.2$_{\pm 0.81}$	75.6$_{\pm 0.29}$	23.7$_{\pm 0.98}$	40.1$_{\pm 1.24}$
CL	53.0$_{\pm 0.21}$	76.3$_{\pm 0.19}$	36.3$_{\pm 0.77}$	60.1$_{\pm 0.66}$	55.3$_{\pm 0.48}$	83.5$_{\pm 0.28}$	42.4$_{\pm 0.45}$	78.1$_{\pm 0.14}$	48.2$_{\pm 0.45}$	95.0$_{\pm 0.04}$
ELR	10.4$_{\pm 0.24}$	31.7$_{\pm 0.44}$	10.0$_{\pm 0.64}$	30.1$_{\pm 0.88}$	10.8$_{\pm 0.21}$	32.7$_{\pm 0.53}$	10.3$_{\pm 0.39}$	30.8$_{\pm 0.35}$	40.3$_{\pm 0.39}$	93.0$_{\pm 0.24}$
FCE	56.9$_{\pm 0.58}$	79.2$_{\pm 0.14}$	43.7$_{\pm 0.15}$	66.2$_{\pm 0.19}$	55.3$_{\pm 0.54}$	83.5$_{\pm 0.24}$	41.4$_{\pm 0.55}$	77.3$_{\pm 0.75}$	39.0$_{\pm 0.05}$	67.8$_{\pm 0.47}$
FCE+B	56.1$_{\pm 2.22}$	81.8$_{\pm 1.37}$	50.2$_{\pm 0.02}$	77.2$_{\pm 0.19}$	54.2$_{\pm 0.44}$	83.3$_{\pm 0.43}$	43.8$_{\pm 0.02}$	77.5$_{\pm 0.13}$	40.0$_{\pm 0.35}$	73.2$_{\pm 0.08}$
FCE+B*	56.1$_{\pm 2.22}$	82.2$_{\pm 0.39}$	50.2$_{\pm 0.02}$	77.2$_{\pm 0.19}$	54.2$_{\pm 0.44}$	83.4$_{\pm 0.24}$	45.1$_{\pm 0.37}$	79.9$_{\pm 0.24}$	43.1$_{\pm 0.40}$	79.4$_{\pm 0.12}$
GCE	60.0$_{\pm 0.13}$	82.6$_{\pm 0.63}$	44.9$_{\pm 0.07}$	67.2$_{\pm 0.34}$	53.8$_{\pm 0.55}$	81.6$_{\pm 0.14}$	39.4$_{\pm 0.44}$	74.0$_{\pm 0.36}$	44.8$_{\pm 0.62}$	91.2$_{\pm 0.70}$
GCE+B	59.4$_{\pm 0.02}$	83.5$_{\pm 0.24}$	50.3$_{\pm 0.11}$	75.3$_{\pm 0.64}$	55.4$_{\pm 0.55}$	83.0$_{\pm 0.35}$	46.5$_{\pm 1.44}$	77.7$_{\pm 0.35}$	47.1$_{\pm 0.20}$	93.5$_{\pm 0.43}$
GCE+B*	61.0$_{\pm 1.33}$	83.9$_{\pm 0.74}$	50.3$_{\pm 0.11}$	75.3$_{\pm 0.64}$	56.6$_{\pm 0.10}$	83.8$_{\pm 0.88}$	47.7$_{\pm 0.35}$	77.9$_{\pm 0.03}$	47.1$_{\pm 0.20}$	93.5$_{\pm 0.43}$
SCE	55.9$_{\pm 0.53}$	76.5$_{\pm 0.15}$	38.7$_{\pm 0.60}$	60.9$_{\pm 0.41}$	57.5$_{\pm 0.19}$	83.7$_{\pm 0.17}$	43.3$_{\pm 0.87}$	77.5$_{\pm 0.75}$	47.2$_{\pm 0.33}$	92.5$_{\pm 0.01}$
SCE+B	55.5$_{\pm 0.90}$	77.4$_{\pm 0.84}$	47.1$_{\pm 1.32}$	69.2$_{\pm 1.18}$	57.9$_{\pm 0.83}$	83.7$_{\pm 0.41}$	50.0$_{\pm 1.62}$	80.4$_{\pm 0.65}$	47.9$_{\pm 0.80}$	93.8$_{\pm 0.05}$
SCE+B*	56.6$_{\pm 1.07}$	78.5$_{\pm 0.88}$	47.3$_{\pm 1.16}$	69.6$_{\pm 0.90}$	57.9$_{\pm 0.83}$	83.7$_{\pm 0.41}$	50.0$_{\pm 1.62}$	80.4$_{\pm 0.65}$	47.9$_{\pm 0.80}$	93.8$_{\pm 0.05}$
CE	52.3$_{\pm 1.35}$	75.6$_{\pm 0.93}$	35.3$_{\pm 1.14}$	59.3$_{\pm 0.81}$	54.9$_{\pm 0.12}$	83.3$_{\pm 0.25}$	42.4$_{\pm 0.16}$	78.9$_{\pm 0.56}$	48.6$_{\pm 0.11}$	95.3$_{\pm 0.10}$
CE+B	50.9$_{\pm 1.01}$	76.5$_{\pm 0.86}$	39.9$_{\pm 1.02}$	65.8$_{\pm 1.19}$	52.9$_{\pm 1.86}$	83.2$_{\pm 0.88}$	34.7$_{\pm 2.51}$	73.4$_{\pm 1.50}$	45.5$_{\pm 5.11}$	93.0$_{\pm 0.16}$
CE+B*	50.9$_{\pm 1.01}$	78.2$_{\pm 1.16}$	39.9$_{\pm 1.02}$	68.1$_{\pm 0.63}$	53.3$_{\pm 0.89}$	83.2$_{\pm 0.88}$	45.9$_{\pm 0.40}$	79.7$_{\pm 0.29}$	50.2$_{\pm 0.35}$	**95.9**$_{\pm 0.14}$
CEP	58.8$_{\pm 0.87}$	78.6$_{\pm 0.38}$	43.5$_{\pm 0.24}$	65.1$_{\pm 1.27}$	59.4$_{\pm 0.08}$	82.2$_{\pm 0.03}$	46.5$_{\pm 0.17}$	76.4$_{\pm 0.25}$	48.2$_{\pm 0.05}$	95.4$_{\pm 0.07}$
CEP+B	**62.3**$_{\pm 0.87}$	**85.1**$_{\pm 0.46}$	54.3$_{\pm 0.86}$	79.2$_{\pm 0.93}$	63.0$_{\pm 0.92}$	**87.5**$_{\pm 0.32}$	53.0$_{\pm 0.28}$	82.8$_{\pm 0.13}$	45.0$_{\pm 0.48}$	95.0$_{\pm 0.08}$
CEP+B*	62.9$_{\pm 0.79}$	85.1$_{\pm 0.46}$	**55.3**$_{\pm 0.37}$	**79.8**$_{\pm 0.08}$	**63.0**$_{\pm 0.14}$	87.5$_{\pm 0.32}$	**55.6**$_{\pm 0.66}$	**83.8**$_{\pm 0.11}$	47.7$_{\pm 0.19}$	95.9$_{\pm 0.23}$

Table C.3: Test accuracies for different losses on the noisy CIFAR100/Asym-CIFAR100/Non-Uniform EMNIST datasets. Losses implementing the noise-bound shaded in blue. When this bound provides benefit, the corresponding value is *boxed*. Overall top values in **bold**.

is higher than the noise-bound, this is highlighted in blue. Otherwise, the cell is indicated in red. Our original table has columns for Top1 and Top5 accuracy, which often have slightly different optimal bounds. For brevity, we combine these by taking a mean of these values.

	TinyImageNet (0.2)		TinyImageNet (0.4)		Animals
Losses	Top 1	Top 5	Top 1	Top 5	
L2 (MSE)	42.91	67.02	29.42	53.13	80.97
MAE	3.86	5.58	3.94	5.54	54.67
NCE-MAE	7.63	10.24	6.29	10.70	80.85
Mix-Up	47.13	70.08	31.05	58.96	83.76
Bootstrap	40.04	61.94	25.69	46.65	82.11
Truncated	43.35	63.67	38.14	59.99	81.69
Mix-Up	47.13	70.08	31.05	58.96	83.10
Curriculum	41.81	64.53	27.57	48.84	81.68
ELR	44.95	66.65	34.66	55.72	82.62
FCE	43.81	64.97	48.85	29.92	81.82
FCE+B	51.18	73.79	46.34	69.92	82.40
GCE	39.81	60.51	26.93	45.17	81.13
GCE+B	47.40	71.37	39.13	63.75	81.37
SCE	39.81	60.51	26.93	45.17	82.59
CE	39.34	61.82	25.84	46.08	81.45
CE+B	38.47	61.85	30.00	52.61	80.72
CEP	44.39	64.56	33.33	51.45	82.06
CEP+B	47.85	71.00	40.56	65.15	81.79

Table C.4: Test accuracies obtained using different losses on the noisy TinyImageNet and Animals10N datasets. Losses implementing the noise-bound are shaded in blue. When this bound provides benefit, the corresponding value is $\boxed{boxed}$. Overall top values are in **bold**.

Appendix D

Early Stopping For Noisy Labels

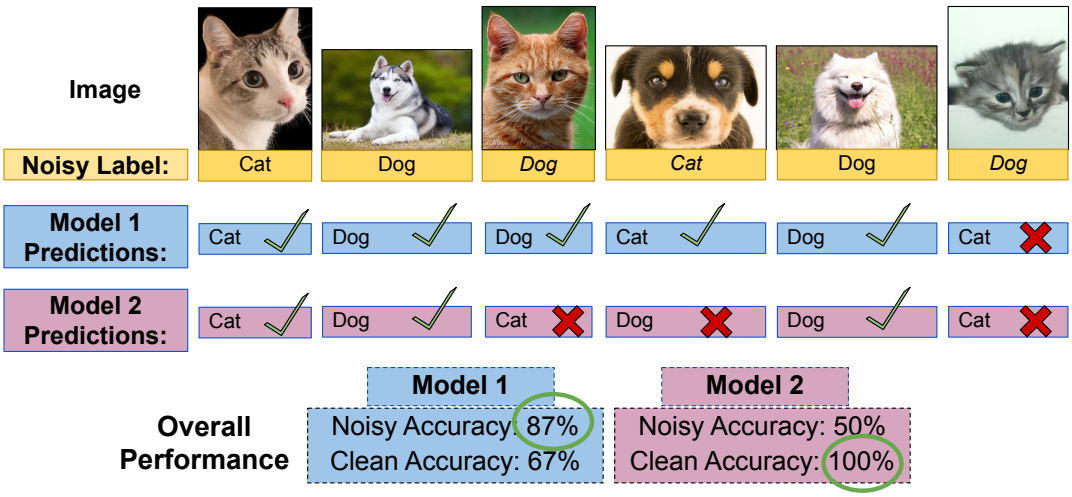


Figure D.1: Noisy and clean accuracies of two models on a small noisily-labelled dataset of cats and dogs. Model 1 (blue) attains a high noisy accuracy as it correctly predicts the *noisy* label for 5 out of 6 images. However, this translates to a clean accuracy of only 67%. In contrast, Model 2 (purple) attains a low noisy accuracy of only 50%. However, this translates to a clean accuracy of 100%. This example seeks to illustrate why, a priori, we might not expect noisy accuracy to be a reliable predictor of clean accuracy.

D.1 Theoretical Proofs

D.1.1 Fact 3 - Bayes-optimality

In this section, we give formal statements of the facts listed in Section 6.4 along with proofs. We begin by demonstrating the veracity of Fact 3.

Theorem D.1.1. *Let $p(x,y)$ be a data-label distribution and suppose that $\widetilde{p}(x,\widetilde{y})$ is a noisy version corrupted by class-preserving label noise. A probability estimator $\boldsymbol{q}^*$ is Bayes-optimal for the noisy distribution if and only if it is Bayes-optimal for the clean distribution. Equivalently;*

$$\boldsymbol{q}^* \in \arg\min_q R_{0\text{-}1}(q) \iff \boldsymbol{q}^* \in \arg\min_q R^{\eta}_{0\text{-}1}(q)$$

Proof. This follows immediately from the definition of class-preserving; indeed, the definition of class-preserving is constructed precisely as the weakest noise condition for which this theorem holds. Specifically, $\boldsymbol{q}^*$ is a global minimiser of the clean risk if and only if, for every $x \in \text{supp}(p(x))$,

$$\arg\max_{i\in\{1,2,\dots c\}} \boldsymbol{q}^*_i(x) = \arg\max_{i\in\{1,2,\dots c\}} p(y=i \mid x)$$

By the definition of class-preserving, the right-hand side is equal to $\arg\max_{i\in\{1,2,\dots c\}} \widetilde{p}(\widetilde{y}=i \mid x)$ and hence

$$\arg\max_{i\in\{1,2,\dots c\}} \boldsymbol{q}^*_i(x) = \arg\max_{i\in\{1,2,\dots c\}} \widetilde{p}(\widetilde{y}=i \mid x)$$

meaning that $\boldsymbol{q}*$ is a global minimiser of the noisy risk. □

D.1.2 Facts 1 and 4

We establish the following Lemma relating the noisy and clean risk of an estimator in terms of the covariance between the estimator and the noise model. Facts 1 and 4 (Section 6.4) then follow as corollaries.

Lemma D.1.2. *Let $\widetilde{p}(x,\widetilde{y})$ be some noisy data-label distribution corrupted by class-preserving (possibly non-uniform) symmetric label noise. Let $\boldsymbol{q}$ be an arbitrary probability estimator model and let f be its plug-in classifier. Let*

$$g(x) \coloneqq p(y = f(x) \mid x)$$

This function gives the probability of our predicted class appearing at x. The expectation of $g(x)$ gives the proportion of labels predicted corrected (i.e. accuracy). Let σ_g denote the standard deviation of $g(x)$. Let $\eta(x)$ denote the noise rate at $x \in \mathcal{X}$ and let σ_η denote the standard deviation of $\eta(x)$ with respect to $p(x)$. The noisy risk and clean risks of $\boldsymbol{q}$ are related via

$$R^\eta(\boldsymbol{q}) = R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta + \frac{c}{c-1} Cov(g(x), \eta_x)$$

which leads to the inequality

$$R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta - \frac{\sigma_\eta \sigma_g c}{c-1} \leq R^\eta(\boldsymbol{q}) \leq R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta + \frac{\sigma_\eta \sigma_g c}{c-1}$$

Thus, in particular when the noise is uniform $\sigma_\eta = 0$ one has

$$R^\eta(\boldsymbol{q}) = R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta$$

Proof. Let $\boldsymbol{q}$ be a probability estimator and let $f : \mathcal{X} \to \{1, 2, \ldots, c\}$ be the associated plug-in classifier. The clean and noisy 0-1 risks of $\boldsymbol{q}$ can be expressed as

$$R(\boldsymbol{q}) = 1 - \mathbb{E}_{x,y\sim p(x,y)}[p(y = f(x) \mid x)]$$
$$R^\eta(\boldsymbol{q}) = 1 - \mathbb{E}_{x,\widetilde{y}\sim \widetilde{p}(x,\widetilde{y})}[p(\widetilde{y} = f(x) \mid x)].$$

We assume that the label noise is (non-uniform) symmetric label noise, meaning that, at each x, the $p(\tilde{y} \mid y)$ may be expressed by a matrix $T(x)$ of the following form

$$\begin{bmatrix} 1-\eta_x & \frac{\eta_x}{c-1} & \frac{\eta_x}{c-1} & \cdots & \frac{\eta_x}{c-1} \\ \frac{\eta_x}{c-1} & 1-\eta_x & \frac{\eta_x}{c-1} & \cdots & \frac{\eta_x}{c-1} \\ \frac{\eta_x}{c-1} & \frac{\eta_x}{c-1} & 1-\eta_x & \cdots & \frac{\eta_x}{c-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\eta_x}{c-1} & \frac{\eta_x}{c-1} & \frac{\eta_x}{c-1} & \cdots & 1-\eta_x. \end{bmatrix}$$

This allows us to write

$$\mathbb{E}_{x,\widetilde{y}\sim\widetilde{p}(x,\widetilde{y})}[p(\widetilde{y} = f(x) \mid x)] = \mathbb{E}_{x\sim p(x)}[\mathbb{E}_{\widetilde{y}\sim p(\widetilde{y}|x,y)}[p(\widetilde{y} = f(x) \mid x)]] \tag{D.1}$$
$$= \mathbb{E}_{x\sim p(x)}[(T(x)\boldsymbol{p}(y \mid x))_{f(x)}] \tag{D.2}$$

For any probability vector $\boldsymbol{p}$, the k^{th} component of $T(x)\boldsymbol{p}$ is equal to

$$(1-\eta_x)p_k + \frac{\eta_x}{c-1}\sum_{i\neq k} p_i = (1-\eta_x)p_k + (1-p_k)\frac{\eta_x}{c-1}$$
$$= p_k\left(1 - \eta_x - \frac{\eta_x}{c-1}\right) + \frac{\eta_x}{c-1}$$

Thus, using Equation D.2, the noisy risk can be written as

$$\begin{aligned} R^{\eta}(\boldsymbol{q}) &= 1 - \mathbb{E}_{x\sim p(x)}\left[p(y = f(x) \mid x)\left(1 - \eta_x - \frac{\eta_x}{c-1}\right) + \frac{\eta_x}{c-1}\right] \\ &= R(\boldsymbol{q}) - \frac{\eta}{c-1} + \mathbb{E}_{x\sim p(x)}\left[p(y = f(x) \mid x)\left(\frac{c\eta_x}{c-1}\right)\right] \end{aligned} \tag{D.3}$$

Using $g(x) \coloneqq p(y = f(x) \mid x)$ for brevity and identifying that

$$\mathbb{E}_{x\sim p(x)}\left[g(x)\eta_x\right] = \mathrm{Cov}(g(x), \eta_x) + \mu_g\mu_\eta$$

we can bound the final term of Equation D.3. Hence, after rearranging, we arrive at our first claim:

$$R^{\eta}(\boldsymbol{q}) = R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta + \frac{c}{c-1}\mathrm{Cov}(g(x), \eta_x).$$

Generally, we cannot expect that $g(x)$ will be independent of η_x. Using Cauchy-Schwarz on the random variables $X \coloneqq \eta_x - \eta$, $Y \coloneqq g(x) - \mu_g$ one obtains the following bound on the covariance

$$|\mathrm{Cov}(g(x), \eta_x)| \leq \sqrt{\mathrm{Var}(g(x))\mathrm{Var}(\eta_x)}$$

Denoting the standard deviations of η_x and $g(x)$ as σ_η, σ_g respectively we have

$$\sigma_\eta\sigma_g - \mu_\eta\mu_g \leq \mathbb{E}_{x\sim p(x)}\left[g(x)\eta_x\right] \leq \sigma_\eta\sigma_g + \mu_\eta\mu_g$$

Thus, we have the following upper and lower bounds on the noise risk

$$R(\boldsymbol{q}) - \frac{\eta}{c-1} + \frac{c}{c-1}\left(-\sigma_\eta\sigma_g + \mu_\eta\mu_g\right) \leq R^{\eta}(\boldsymbol{q}) \leq R(\boldsymbol{q}) - \frac{\eta}{c-1} + \frac{c}{c-1}\left(\sigma_\eta\sigma_g + \mu_\eta\mu_g\right)$$

μ_g is precisely the accuracy of the estimator $\boldsymbol{q}$ and thus $R(\boldsymbol{q}) = 1 - \mu_g$, likewise μ_η is the mean noise rate η. Putting this together, we obtain

$$\begin{aligned} R(\boldsymbol{q}) - \frac{\eta}{c-1} + \frac{c\eta}{c-1}(1 - R(\boldsymbol{q})) - \frac{\sigma_\eta\sigma_g c}{c-1} \leq R^{\eta}(\boldsymbol{q}) \leq R(\boldsymbol{q}) - \frac{\eta}{c-1} + \frac{c\eta}{c-1}(1 - R(\boldsymbol{q})) + \frac{\sigma_\eta\sigma_g c}{c-1} \\ R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta - \frac{\sigma_\eta\sigma_g c}{c-1} \leq R^{\eta}(\boldsymbol{q}) \leq R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta + \frac{\sigma_\eta\sigma_g c}{c-1} \end{aligned} \tag{D.4}$$

Which is precisely our second claim. Finally, note the two special cases:

Uniform: If $\eta_x = \eta = const.$ then this becomes

$$\begin{aligned} R(\boldsymbol{q}) - \frac{\eta}{c-1} + (1 - R(\boldsymbol{q}))\left(\frac{c\eta}{c-1}\right) \\ = R(\boldsymbol{q})\left(1 - \frac{c\eta}{c-1}\right) + \eta \end{aligned}$$

as claimed.

Independence: In the event that $g(x)$ and η_x are uncorrelated we have

$$\begin{aligned} R^{\eta}(\boldsymbol{q}) &= R(\boldsymbol{q}) - \frac{\eta}{c-1} + \frac{c}{c-1}\mu_g\mu_\eta \\ &= R(\boldsymbol{q}) - \frac{\eta}{c-1} + \frac{c}{c-1}(1-R(\boldsymbol{q}))\eta \\ &= R(\boldsymbol{q})\left(1 - \frac{\eta c}{c-1}\right) + \eta \end{aligned}$$

□

Fact 1: Symmetric Uniform Noise Lemma D.1.2 establishes Fact 1 in Section 6.4; that when the label noise is symmetric, uniform and class-preserving ($\eta < \frac{c-1}{c}$) the noisy and clean risk are related by a linear map. This is significant because it means that if $\boldsymbol{q}_1, \boldsymbol{q}_2$ are two models then $R(\boldsymbol{q}_1) \leq R(\boldsymbol{q}_2) \iff R^{\eta}(\boldsymbol{q}_1) \leq R^{\eta}(\boldsymbol{q}_2)$: If one has a set of probability estimators $\{\boldsymbol{q}_i\}_{i=1}^N$ and we select $\boldsymbol{q}_k$ which minimises the noisy risk then this will also minimise the clean risk.

Fact 4 Lemma D.1.2 also establishes Fact 4, that there is an affine relationship between the clean and noisy risk of a probability estimator when the noise model η_x and the model's accuracy function ($g(x) \coloneqq p(y = f(x) \mid x)$ where f is the plug-in classifier of the estimator) are uncorrelated (assuming non-uniform symmetric label noise). Thus, if one selects the minimiser of the noisy risk among a set of estimators $Q \coloneqq \{\boldsymbol{q}_i\}_{i=1}^N$, this model necessarily is also a minimiser of the clean risk.

D.1.3 Fact 2

We have shown in Lemma D.1.2 that, when label noise is uniform and symmetric and $\eta < \frac{c-1}{c}$ then for *any* two estimators $\boldsymbol{q}_1, \boldsymbol{q}_2$ and *any* data-label distribution $p(x,y)$

$$R^{\eta}(\boldsymbol{q}_1) \leq R^{\eta}(\boldsymbol{q}_2) \iff R(\boldsymbol{q}_1) \leq R(\boldsymbol{q}_2) \tag{D.5}$$

We now endeavour to show the converse, that this only holds for uniform symmetric noise below the specified threshold.

Assume we have some closed-set label noise model for which Equation D.5 holds for all distributions $p(x,y)$ and estimators $\boldsymbol{q}_1, \boldsymbol{q}_2$. Since Equation D.5 holds for all distributions, then it holds in particular when we set $p(x)$ to be a Dirac delta distribution at some point $x \in \mathcal{X}$, for all possible conditional distributions $\boldsymbol{p}(y \mid x) \in \Delta$. Given

two estimators $\boldsymbol{q}_1, \boldsymbol{q}_2$ we let $k_1 \coloneqq \arg\max_i(\boldsymbol{q}_1(x))_i$, $k_2 \coloneqq \arg\max_i(\boldsymbol{q}_2(x))_i$ denote their predicted labels at x. Then for any pair of predicted labels $k_1, k_2 \in \mathcal{Y}$, and for any $\boldsymbol{p}(y \mid x) \in \Delta$, Equation D.5 implies

$$(1 - \widetilde{p}(\widetilde{y} = k_1 \mid x) \leq 1 - \widetilde{p}(\widetilde{y} = k_2 \mid x)) \iff (1 - p(y = k_1 \mid x) \leq 1 - p(y = k_2 \mid x)).$$

This can be written equivalently as

$$(\widetilde{p}(\widetilde{y} = k_2 \mid x) \leq \widetilde{p}(\widetilde{y} = k_1 \mid x)) \iff (p(y = k_2 \mid x) \leq p(y = k_1 \mid x)). \tag{D.6}$$

Since we are modelling closed-set noise, we know the noise can be modelled by a transition matrix T at x. Thus, letting $\boldsymbol{p}$ be shorthand for the condition distribution at x; $\boldsymbol{p} \coloneqq \boldsymbol{p}(y \mid x)$, Equation D.6 is equivalent to

$$(T\boldsymbol{p})_{k_2} \leq (T\boldsymbol{p})_{k_1} \iff \boldsymbol{p}_{k_2} \leq \boldsymbol{p}_{k_1}. \tag{D.7}$$

Our goal is to demonstrate that, in order for Equation D.7 to hold $\forall \boldsymbol{p} \in \Delta$ and $\forall k_1, k_2 \in \mathcal{Y}$, that T must describe symmetric label noise. Equation D.7 implies that $(T\boldsymbol{p})_{k_2} = (T\boldsymbol{p})_{k_1} \iff \boldsymbol{p}_{k_2} = \boldsymbol{p}_{k_1}$ since $(\boldsymbol{p}_{k_2} \leq \boldsymbol{p}_{k_1}) \wedge (\boldsymbol{p}_{k_1} \leq \boldsymbol{p}_{k_2}) \implies \boldsymbol{p}_{k_1} = \boldsymbol{p}_{k_2}$. The first major implication of this is that T must be row stochastic as well as columns stochastic - rows sum to one. To see this let $\boldsymbol{p} = \left(\frac{1}{c}, \frac{1}{c}, \ldots, \frac{1}{c}\right)$; the uniform distribution over labels. Then $T\boldsymbol{p}$ must also be the uniform distribution, thus, the rows of T must all sum to 1 (since $(T\boldsymbol{p})_i = \frac{1}{c} \iff \sum_j T_{ij}\frac{1}{c} = \frac{1}{c} \iff \sum_j T_{ij} = 1$).

Now letting $\boldsymbol{p} = \boldsymbol{e}_k$ (the k^{th} coordinate vector), Equation D.7 implies that $(T\boldsymbol{p})_i = (T\boldsymbol{p})_j$ for all $i, j \neq k$. As $T\boldsymbol{p}$ is the k^{th} column of T then for all $i, j \neq k$, $T_{ik} = T_{jk}$. This allows us to write T as

$$\begin{bmatrix} T_{11} & a_2 & a_3 & \ldots & a_c \\ a_1 & T_{22} & a_3 & \ldots & a_c \\ \ldots & & & & \\ a_1 & a_2 & a_3 & \ldots & T_{cc} \end{bmatrix}$$

Since we know that the matrix is column and row stochastic, then, in particular, the k^{th} row and column sum to one, so

$$\begin{aligned} T_{kk} + \sum_{i \neq k} a_i = T_{kk} + (c-1)a_k = 1 \\ \iff \sum_{i=1}^{c} a_i = ca_k \end{aligned}$$

We can write this as a system of equations

$$\begin{bmatrix} 1-c & 1 & 1 & \dots & 1 \\ 1 & 1-c & 1 & \dots & 1 \\ \dots & & & & \\ 1 & 1 & 1 & \dots & 1-c \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_c \end{bmatrix} = \mathbf{0} \tag{D.8}$$

Thus, the $\boldsymbol{a}$ must lie in the nullspace of this matrix which may be computed. We identify that all vectors of the form $(\lambda, \lambda, \dots, \lambda)$ lie in the nullspace. However, as the rank of the matrix is $c-1$, then by the rank-nullity theorem, this is the entire nullspace.

Hence we know that $a_1 = a_2 = \dots = a_c =: a$ meaning that

$$T = \begin{bmatrix} T_{11} & a & a & \dots & a \\ a & T_{22} & a & \dots & a \\ \dots & & & & \\ a & a & a & \dots & T_{cc} \end{bmatrix}$$

By the condition that the rows and columns sum to one, this must be writable as

$$T = \begin{bmatrix} 1-a(c-1) & a & a & \dots & a \\ a & 1-a(c-1) & a & \dots & a \\ \dots & & & & \\ a & a & a & \dots & 1-a(c-1) \end{bmatrix}$$

then letting $a \coloneqq \frac{\eta}{c-1}$ we have

$$T = \begin{bmatrix} 1-\eta & \frac{a}{c-1} & \frac{\eta}{c-1} & \dots & \frac{\eta}{c-1} \\ \frac{\eta}{c-1} & 1-\eta & \frac{\eta}{c-1} & \dots & \frac{\eta}{c-1} \\ \dots & & & & \\ \frac{\eta}{c-1} & \frac{\eta}{c-1} & \frac{\eta}{c-1} & \dots & 1-\eta \end{bmatrix}$$

which is precisely the matrix describing symmetric label noise as desired.

Thus, for Equation D.6 to hold for all estimators and data-label distributions, the label noise must be symmetric at every x. It remains to show that the noise rate must be uniform. This may be derived using Lemma D.1.2 which states that for non-uniform symmetric label noise

$$R^{\eta}(\boldsymbol{q}) = R(\boldsymbol{q}) \left(1 - \frac{c\eta}{c-1}\right) + \eta + \frac{c}{c-1} \mathrm{Cov}(g(x), \eta_x).$$

Where $g(x) \coloneqq p(f(x) \mid x)$ gives the probability of the predicted label at x. It follows, that for any two estimators $\boldsymbol{q}_1, \boldsymbol{q}_2$

$$R^{\eta}(\boldsymbol{q}_1) \leq R^{\eta}(\boldsymbol{q}_2) \iff$$
$$R(\boldsymbol{q}_1) \leq R(\boldsymbol{q}_2) + \frac{1}{\frac{c-1}{c} - \eta} \left(\mathrm{Cov}(g_2(x), \eta_x) - \mathrm{Cov}(g_1(x), \eta_x)\right).$$

In order to ensure the difference between the covariances vanishes for all distributions $p(x,y)$ and estimators $\boldsymbol{q}_1, \boldsymbol{q}_2$, we must have $\eta_x = \eta = const$. Thus, the label noise model must be uniform symmetric noise as claimed—this establishes Fact 2.

D.1.4 Fact 5

With Fact 2 we saw that for any noise model other than uniform symmetric label noise, the minimiser of the noisy risk may not be a minimiser of the clean risk. In this section, we provide some worst-case bounds. Specifically, if

$$\boldsymbol{q}^{\eta}_* \coloneqq \arg\min_{\boldsymbol{q} \in Q} R^{\eta}(\boldsymbol{q})$$
$$\boldsymbol{q}_* \coloneqq \arg\min_{\boldsymbol{q} \in Q} R(\boldsymbol{q})$$

denote the minimisers of the noisy and clean risks within Q. Then, we seek to bound

$$|R(\boldsymbol{q}^{\eta}_*) - R(\boldsymbol{q}_*)|.$$

To simplify the mathematics, make some assumptions. We assume that the data distribution is separable and that every column of the transition matrix is a permutation of every other column.

Theorem D.1.3. *Consider a scenario with class-conditional label noise characterised by a transition matrix T. Define η_{max} as the maximum transition probability, i.e., $\eta_{max} \coloneqq \max_{j \neq i} T_{ij}$, and η_{min} as the minimum transition probability, i.e., $\eta_{min} \coloneqq \min_{j \neq i} T_{ij}$. Let $Q \coloneqq \{\boldsymbol{q}\}_{i=1}^{N}$ be a set of probability estimators. Denote $\boldsymbol{q}^{\eta}_*$ as the minimiser of the noisy risk and $\boldsymbol{q}_*$ as the minimiser of the clean risk, respectively:*

$$\boldsymbol{q}^{\eta}_* \coloneqq \arg\min_{\boldsymbol{q} \in Q} R^{\eta}(\boldsymbol{q}),$$
$$\boldsymbol{q}_* \coloneqq \arg\min_{\boldsymbol{q} \in Q} R(\boldsymbol{q}).$$

Denote the clean and noisy risks of these estimators as:

$$R_k := R(\boldsymbol{q}_*^\eta) \quad \textit{and} \quad R_*^\eta := R(\boldsymbol{q}_*^\eta),$$
$$R_* := R(\boldsymbol{q}_*) \quad \textit{and} \quad R_l^\eta := R(\boldsymbol{q}_*).$$

The difference between the optimal clean risk R_* and the clean risk achieved by $\boldsymbol{q}_*^\eta$ satisfies the following inequality:

$$|R_k - R_*| \leq \frac{R_*^\eta - \eta}{1-\eta-\eta_{max}} - \frac{R_l^\eta - \eta}{1-\eta-\eta_{min}}.$$

Thus, given that R_*^η is optimal, we have:

$$|R_k - R_*| \leq (R_*^\eta - \eta)\left(\frac{1}{1-\eta-\eta_{max}} - \frac{1}{1-\eta-\eta_{min}}\right). \tag{D.9}$$

Proof. Let $\boldsymbol{q}_j \in Q$ be an arbitrary estimator in our set. Let $A^+ \subseteq \text{supp}(p(x))$ denote the set upon which $\boldsymbol{q}_j$ correctly predicts the true label and we use A^- to denote the complement of A^+ in $\text{supp}(p(x))$ so that $A^+ \cup A^- = \text{supp}(p(x))$. We let $f(x)$ denote the plug-in classifier induced by $\boldsymbol{q}_j$. By the separability assumption, we know that

$$p(y = f(x) \mid x) = 1 \quad \text{for} \quad x \in A^+,$$
$$p(y = f(x) \mid x) = 0 \quad \text{for} \quad x \notin A^+.$$

Likewise,

$$\widetilde{p}(\widetilde{y} = f(x) \mid x) = 1-\eta \quad \text{for} \quad x \in A^+,$$
$$\widetilde{p}(\widetilde{y} = f(x) \mid x) \in [\eta_{min}, \eta_{max}] \quad \text{for} \quad x \notin A^+.$$

This allows us to write the following inequality lower bounding the noisy risk of $\boldsymbol{q}_j$;

$$\begin{aligned} R_j^\eta &= \int_{A^+} p(x)(1-\widetilde{p}(\widetilde{y} = f(x) \mid x))dx + \int_{A^-} p(x)(1-\widetilde{p}(\widetilde{y} = f(x) \mid x))dx \\ &\geq \mu(A^+)\eta + \mu(A^-)(1-\eta_{max}). \end{aligned}$$

Here μ denotes the measure associated with the density p, so that $\int_{A^+} p(x)dx = \mu(A^+)$. We can similarly deduce that

$$R_j^\eta \leq \mu(A^+)\eta + \mu(A^-)(1-\eta_{min}).$$

We note that $\int_{A^+} p(x)dx = \mu(A^+) = 1 - R_j$ thus, we have the following inequality between the noisy and clean risks of an arbitrary estimator $\boldsymbol{q}_j \in Q$

$$R_j^\eta \leq (1-R_j)\eta + R_j(1-\eta_{min}) = R_j(1-\eta_{min}-\eta) + \eta, \tag{D.10}$$
$$R_j^\eta \geq (1-R_j)\eta + R_j(1-\eta_{max}) = R_j(1-\eta_{max}-\eta) + \eta. \tag{D.11}$$

Rearranging this is equivalent to

$$\frac{R_j^\eta - \eta}{1-\eta_{min}-\eta} \leq R_j \leq \frac{R_j^\eta - \eta}{1-\eta_{max}-\eta}$$

Hence, in particular

$$\frac{R_l^\eta - \eta}{1-\eta_{min}-\eta} \leq R_* \leq \frac{R_l^\eta - \eta}{1-\eta_{max}-\eta},$$
$$\frac{R_*^\eta - \eta}{1-\eta_{min}-\eta} \leq R_k \leq \frac{R_*^\eta - \eta}{1-\eta_{max}-\eta}.$$

Putting these together, we conclude

$$\begin{aligned}
&\frac{R_l^\eta - \eta}{1-\eta-\eta_{min}} \leq R_* \leq R_k \leq \frac{R_*^\eta - \eta}{1-\eta-\eta_{max}} \\
\implies &|R_k - R_*| \leq \frac{R_*^\eta - \eta}{1-\eta-\eta_{max}} - \frac{R_l^\eta - \eta}{1-\eta-\eta_{min}} \\
\implies &|R_k - R_*| \leq (R_*^\eta - \eta)\left(\frac{1}{1-\eta-\eta_{max}} - \frac{1}{1-\eta-\eta_{min}}\right)
\end{aligned}$$

as desired. Note that if $\eta_{min} = \eta_{max}$ (i.e. the label noise is symmetric) that $|R_k - R_*| = 0$ which is consistent with Fact 1 (Section D.1.2). □

Corollary D.1.4. *For Pairwise label noise (i.e. where $T_{11} = 1-\eta$ and for some $i \neq 1$ $T_{1i} = \eta$, refer to Section 2.2.1) we have the following bound;*

$$|R_k - R_*| \leq \frac{\eta\left(R_k^\eta - \eta\right)}{(1-2\eta)(1-\eta)}$$

Note that for a noise rate η, a noisy risk below η is unattainable and $R_k \to R_*$ as $R_k^\eta \to \eta$.

D.1.4.1 Discussion

$$T = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} \tag{D.12}$$

Using some examples we can get a sense of how good the bounds we derived in Theorem D.1.3 and Corollary D.1.4. Start by considering a three-class dataset corrupted by label noise described by the transition matrix in Equation D.12. Suppose we train a classifier and that during training, we achieve a peak noisy accuracy of 40%. Plugging these numbers into Equation D.9 we get

$$|R_k - R_*| \leq \frac{1}{6}.$$

This means that, in the worst case, the optimal clean test accuracy attained during training could be 16.6% higher than the accuracy of our model selected by NES.

As a second example, suppose that we have a dataset corrupted by pairwise with a noise rate of $\eta = 40\%$, we train a neural network estimator on this dataset. The peak noisy validation accuracy attained during training is 50%. Then

$$|R_k - R_*| \leq \frac{1}{3}$$

Thus, in this setting, the difference in clean test accuracy between the NES-chosen model and the optimal model could be as high as 33.3%!

Neither of the bounds in these examples are particularly tight: Within the context of neural network classifiers trained on image datasets, a decrease in test accuracy of about 10% is large indeed. Empirically, we find that a Noisy Early Stopping policy generally permits us to attain a model within a single percentage point of optimal clean test accuracy. These bounds are therefore inadequate to explain why this performance is so good.

D.1.5 Lemma 6.6.1 and Generalisations

Lemma D.1.5. *Suppose we train a classifier on a dataset corrupted by (class-preserving) label noise for N epochs. Let $f^{(n)}$ denote the classifier obtained after training for n epochs. Let $\boldsymbol{g}^{(n)} = (g_1^{(n)}, g_2^{(n)}, \ldots, g_c^{(n)})$ denote the $g-$vector of the n^{th} classifier $f^{(n)}$. Suppose there exists an integer $T < N$ so that for all $i \geq 2$, $g_i^{(n)}$ is decreasing for $n \in \{1,2,\ldots,T\}$ and increasing for $n \in \{T, T+1 \ldots, N\}$ then the minimiser of the noisy risk within $\{f^{(n)}\}_{n=1}^{N}$ also minimises the clean risk.*

Proof. Since we are assuming that the data-label distribution is separable, we know that the clean conditional class distribution $\boldsymbol{p}(y \mid x) = \boldsymbol{e}_k$ for some k. It follows that, if $\boldsymbol{g}^{(n)}$ is the $g-$vector of $f^{(n)}$, then the clean accuracy of the n^{th} classifier $f^{(n)}$ is equal to $g_1^{(n)}$. We know that $\sum_{i=1}^{c} g_i^{(n)} = 1$ so clean accuracy can be expressed as $1 - g_2^{(n)} - g_3^{(n)} - \ldots - g_c^{(n)}$. Since by assumption, each of the $g_i^{(n)}$ attain their minimum simultaneously at $n = T$, then the clean accuracy is maximised as $n = T$. Our goal now is to show that the noisy accuracy is also maximised at this epoch.

The noisy accuracy may be expressed

$$\begin{aligned} &\int p(x)\widetilde{p}(\widetilde{y} = f^{(n)}(x) \mid x)dx = \int p(x) \left(T(x)\boldsymbol{p}(y \mid x)\right)_{f^{(n)}(x)} dx \\ &= \int p(x) \left(T\boldsymbol{e}_{k(x)}\right)_{f^{(n)}(x)} dx = \int p(x) \left(T_{k(x), f^{(n)}(x)}\right) dx. \end{aligned} \tag{D.13}$$

We assume that each of the columns of T are permutations of every other column. Suppose that each column of T is a permutation of the vector $(1-\eta, \eta_2, \ldots, \eta_c)$ where $\eta_i < \eta_j$ for $i < j$ and $\eta := \eta_2 + \eta_3 + \ldots \eta_c$. Then Equation D.13 can be written as

$$\begin{aligned}
&\int p(x) T_{k(x), f^{(n)}(x)} dx \\
&= \sum_{i=2}^{c} \int p(x) \eta_i p(T_{k(x), f^{(n)}(x)} = \eta_i \mid x) dx + \int p(x)(1-\eta) p(T_{k(x), f^{(n)}(x)} = 1-\eta \mid x) dx \\
&= \sum_{i=2}^{c} \eta_i \int p(x) p(T_{k(x), f^{(n)}(x)} = \eta_i \mid x) dx + (1-\eta) \int p(x) p(T_{k(x), f^{(n)}(x)} = 1-\eta \mid x) dx \\
&= \sum_{i=2}^{c} \eta_i g_i^{(n)} + (1-\eta) g_1^{(n)} \qquad \text{(D.14)} \\
&= (1-\eta, \eta_2, \eta_3, \ldots, \eta_c) \cdot (g_1^{(n)}, g_2^{(n)}, \ldots, g_c^{(n)})
\end{aligned}$$

Using the fact that $\sum_i g_i = 1$ we know that Equation D.14 can be written as

$$\begin{aligned}
\sum_{i=2}^{c} \eta_i g_i^{(n)} + (1-\eta) g_1^{(n)} &= \sum_{i=2}^{c} \eta_i g_i^{(n)} + (1 - \eta_2 - \eta_3 \ldots - \eta_c)(1 - g_2^{(n)} - g_3^{(n)} - \ldots - g_c^{(n)}) \\
&= \left(\sum_{i=2}^{c} g_i^{(n)} (\eta_2 + \ldots + 2\eta_i + \ldots + \eta_c - 1) \right) + (1 - \eta_2 - \eta_3 \ldots - \eta_c).
\end{aligned}$$

Since our label noise is class-preserving we know that $(1-\eta) > \eta_2 > \ldots > \eta_c$ so for every $i \geq 2$,

$$\eta_2 + \ldots + 2\eta_i + \ldots + \eta_c - 1 = \eta_i - (1-\eta) < 0$$

So, the noisy accuracy at epoch n may be expressed as

$$A_\eta^{(n)} = \sum_{i=2}^{c} \alpha_i g_i^{(n)} + const.$$

where $\alpha_i < 0$. By assumption, each of the $g_i^{(n)}$ are decreasing for $n \in \{1, 2, \ldots, T\}$ and therefore $A_\eta^{(n)}$ is increasing for $n \in \{1, 2, \ldots, T\}$. Likewise the $g_i^{(n)}$ are increasing for $n \in \{T, T+1, \ldots, N\}$ meaning that the noisy accuracy is decreasing for $n \in \{T, T+1, \ldots, N\}$. Hence the noisy accuracy attains its maximum at $n = T$ - the same epoch as the clean accuracy. □

Asymmetric We can extend Lemma 6.6.1 to accommodate uniform asymmetric label noise relatively easily by associating a 'g-matrix' with each classifier rather than a vector. In this matrix, the i-j^{th} entry, denoted g_{ij}, represents the average probability that the model assigns to class j given that the true class is i, computed as $g_{ij} :=$

$\mathbb{E}_{x\sim p(x|y=i)}[\boldsymbol{q}(x)_j]$. The '$g-$matrix' is closely related to the confusion matrix for the classifier, although subtly different. (Hendrycks et al. (2018) use this matrix to estimate the noise transition matrix T.) The noisy accuracy equals the Hadamard product of the noise transition matrix and this g-matrix. Under the assumption that each off-diagonal entry g_{ij} (for $i \neq j$) reaches its minimum simultaneously, the epochs at which the noisy and clean accuracies are maximised coincide.

Non-Uniform Generalising Lemma 6.6.1 to the case of non-uniform label noise is more non-trivial. Much of the proof of Lemma 6.6.1 holds in the non-uniform case, however the η_i now vary during training. More significantly, if η denotes the average noise rate on the subset of dataspace upon which our classifier predicts the clean label and η_i similarly denotes the average noise rate on the subset of dataspace upon which our classifier predicts the i^{th} most likely noisy label, then we no longer have $\eta = \sum_{i>1}\eta_i$. This prevents us from concluding that the coefficients of each of the $g_i^{(n)}$ are negative without further assumptions. We add in the assumption that the classifier tends to predict more accurately (with respect to the clean distribution) in regions of dataspace with low noise rate. This added assumption allows us to extend the proof to the non-uniform case. We do not write this out in full and may be seen as an exercise for the reader.

Non-Separable We have assumed that the data distribution is separable. Handling the non-separable case is challenging - however, we can reason informally that Lemma 6.6.1 should be extendable to the non-separable case by noting that label-noise and aleatoric randomness are mathematically indistinguishable. Specifically, suppose that $p(x,y)$ is non-separable. This means that some of the clean class conditionals $\boldsymbol{p}(y \mid x)$ are not expressible as a coordinate vector $\boldsymbol{e}_k$ for some k. We can conceptualise the clean distribution $p(x,y)$ therefore as being a noised version of some separable distribution $p^\circ(x,y)$. The label noise associating these distributions would generally be non-uniform, asymmetric and must be class-preserving. Suppose now we have a noisy distribution $\widetilde{p}(x,\widetilde{y})$ which is obtained by applying label noise to $p(x,y)$ then, by concatenation of these two label noise processes, $\widetilde{p}(x,\widetilde{y})$ can be thought of as a noised version of $p^\circ(x,y)$. Our results tell us therefore that if we were to train a classifier on samples drawn from $\widetilde{p}(x,\widetilde{y})$, then the $\widetilde{p}(x,\widetilde{y})-$accuracy should peak at the same time as the $p^\circ(x,y)-$accuracy (under the relevant assumptions). With $p^\circ(x,y)-$accuracy and $\widetilde{p}(x,\widetilde{y})-$accuracy occurring at the same epoch, we might expect the $p(x,y)-$accuracy to also peak at this epoch as it is an intermediate distribution between $\widetilde{p}$ and p°.

Relaxing Simultaneity The key condition of Lemma 6.6.1 is that each of the $g_i^{(n)}$ achieve their minima at the same epoch during training. In practice this is not always exactly satisfied although each of the g_i generally attain their minima within a couple of epochs of one another (See Figure 6.9). Lemma 6.6.1 can be generalised to handle this. If we let $T_1 < T_2$ be two integers such that each of the g_i (for $i > 1$) achieve their minima at epochs $n \in \{T_1, T_1 + 1, \ldots, T_2\}$ then both the noisy and clean accuracies are attained in this short interval. Thus, if we stop training when the noisy accuracy is maximised, we are fairly close to the point during training when clean accuracy is maximised if $T_2 - T_1$ is small.

D.2 Algorithm Details and Code

D.2.0.1 Symmetric Noise in Image Datasets

In symmetric noise models, we alter a proportion η of labels in datasets such as CIFAR-10, CIFAR-100, MNIST, and FashionMNIST. For each affected label, a new label is chosen uniformly at random. In CIFAR-10 and CIFAR-100, the new label is selected from all possible labels, excluding the original. In MNIST and FashionMNIST, the original label remains a possible choice. The datasets are split into 70% training, 15% noisy validation, and 15% clean test sets to evaluate the effects of label noise.

D.2.0.2 Non-Uniform Noise in EMNIST

In the *Non-Uniform EMNIST* setup, we introduce label noise based on a linear classifier's predictions. After training the classifier on EMNIST, we modify the label of each training point with probability η to the classifier's output. This method creates x-dependent noise, where the label alterations depend on the classifier's performance across different regions of the data space. The classifier attains an accuracy of 28%, meaning the new label is different from the original label 72% of the time.

D.2.0.3 Asymmetric Label Noise

CIFAR10 The asymmetric label noise for CIFAR10 is consistent with that used in (Patrini et al., 2017) and most other noise robust literature, as described in Section 4.5.

FashionMNIST The asymmetric label noise for FashionMNIST follows the methodology used in (Z. Zhang & Sabuncu, 2018), as detailed in Section 4.5.

CIFAR100 The asymmetric label noise for CIFAR100 follows the methodology used in (X. Li et al., 2021), which consists of symmetric noise within each of the 20 superclasses as detailed in Section 4.5.

MNIST The asymmetric label noise for MNIST deviates slightly from the literature. We divide the labels into 4 groups: {0,1,2}, {3,4,5}, {6,7,8}, {9}. Within each of the sets of cardinality three, label noise as described by the following transition matrix where $T_{ij} \coloneqq p(\tilde{y} = i \mid y = j)$

$$T = \begin{bmatrix} 1-\eta & \eta & \eta \\ \eta & 1-\eta & \eta \\ 0 & 0 & 1-2\eta \end{bmatrix}$$

D.2.1 Creation of the Noised Datasets

D.2.1.1 Symmetric Noise in Image Datasets

In symmetric noise models, we alter a proportion η of labels in datasets such as CIFAR-10, CIFAR-100, MNIST, and FashionMNIST. For each affected label, a new label is chosen uniformly at random. In CIFAR-10 and CIFAR-100, the new label is selected from all possible labels excluding the original. In MNIST and FashionMNIST, the original label remains a possible choice. The datasets are split into 70% training, 15% noisy validation, and 15% clean test sets to evaluate the effects of label noise. The noise rates used in the experiments shown in Table 6.1 are MNIST ($\eta = 0.8$), FashionMNIST ($\eta = 0.6$), NoisedCIFAR10 ($\eta = 0.6$) and NoisedCIFAR100 ($\eta = 0.6$).

D.2.1.2 Non-Uniform Noise in EMNIST

In the *Non-Uniform EMNIST* setup, we introduce label noise based on a linear classifier's predictions. After training the classifier on EMNIST, we modify the label of each training point with probability η to the classifier's output. This method creates x-dependent noise, where the label alterations depend on the classifier's performance across different regions of the data space. The classifier attains an accuracy of 28%, meaning the new label is different from the original label 72% of the time.

D.2.1.3 Asymmetric Label Noise

CIFAR10 The asymmetric label noise for CIFAR10 is consistent with that used in (Patrini et al., 2017) and most other noise robust literature. The label noise is constructed by mapping pairs of classes into each other, specifically, with probability η one transitions $Truck \rightarrow Automobile$, $Bird \rightarrow Plane$, $Deer \rightarrow Horse$, $Cat \leftrightarrow Dog$ (Patrini et al., 2017). This label noise is used in all papers which use the CIFAR10 dataset with asymmetric label noise except (X. Li et al., 2021). In (X. Li et al., 2021) they use circular label noise, flipping each class to the following class with probability η. Our experiments in Table 6.1 uses $\eta = 0.4$.

FashionMNIST The asymmetric label noise for FashionMNIST follows the methodology used in (Z. Zhang & Sabuncu, 2018). This synthetic label noise is constructed similarly to CIFAR10 and MNIST, via pairwise transitions; $Boot \rightarrow Sneaker$, $Sneaker \rightarrow Boot$, $Sneaker \rightarrow Sandals$, $Pullover \rightarrow Shirt$, $Coat \leftrightarrow Dress$. Our experiments in Table 6.1 uses $\eta = 0.4$.

CIFAR100 The asymmetric label noise for CIFAR100 follows the methodology used in Patrini et al. (2017). In Patrini et al. (2017) the authors introduce asymmetric noise to the CIFAR100 dataset by implementing circular noise within each of the 20 'superclasses.' The CIFAR100 dataset is organised into groups of 5 classes, termed as superclasses. For instance, the 'Aquatic Mammals' superclass comprises Beaver, Dolphin, Otter, Seal, and Whale (Patrini et al., 2017). Within each superclass, labels are cyclically permuted (e.g., Beaver $\mapsto$ Otter $\mapsto$ Dolphin, and so forth) with a probability of η. This label noise is also used in the following papers Z. Zhang and Sabuncu (2018), Ma et al. (2020), L. Feng et al. (2021), Y. Wang et al. (2019), Reed et al. (2014), Patrini et al. (2017), Englesson and Azizpour (2021b), T. Zhou et al. (2020), S. Liu et al. (2020). Our experiments in Table 6.1 uses $\eta = 0.6$.

MNIST The asymmetric label noise for MNIST deviates slightly from the literature. We divide the labels into 4 groups: {0,1,2}, {3,4,5}, {6,7,8}, {9}. Within each of the sets of cardinality three label noise as described by the following transition matrix where $T_{ij} \coloneqq p(\widetilde{y} = i \mid y = j)$

$$T = \begin{bmatrix} 1-\eta & \eta & \eta \\ \eta & 1-\eta & \eta \\ 0 & 0 & 1-2\eta \end{bmatrix}$$

Our experiments in Table 6.1 uses $\eta = 0.2$.

D.2.1.4 NonUniformMNIST Dataset

The NonUniformMNIST dataset is a modified version of the standard MNIST dataset. It introduces non-uniform noise into the labels based on a noise level parameter, η. Two distinct class-conditional noise models are applied across the dataset, each influencing separate segments of dataspace. This segmentation is governed by the principal component analysis (PCA) of each class, ensuring that exactly half of the samples from each class are noised according to the first model and the remaining half according to the second model.

Label Noise Process Each image label in the dataset is subject to modification according to the following probabilistic rule:

- With probability $1-\eta$, the label remains unchanged.
- With probability η, the label is altered using one of two predefined stochastic transition matrices, denoted as `matrix1` and `matrix2`.

Transition Matrices

- **`matrix1` (Symmetric Noise):** This matrix introduces uniform symmetric noise across all labels, constructed such that each incorrect label assignment occurs with equal probability.
- **`matrix2` (Symmetric Noise with Subsets):** This matrix applies noise only within predefined subsets of labels: (0,1,2), (3,4,5), and (6,7,8), with label 9 unchanged. Noise within these subsets is also symmetric.

Selection of Transition Matrix For each class in the dataset:

1. **Statistical Computation:**
    - Calculate the mean of all samples within the class.
    - Determine the first principal component (PC1) of the class samples.
2. **Label Adjustment Criterion:**
    - Center each sample by subtracting the class mean.
    - Project the centred sample onto PC1.
    - The sign of the resulting scalar product determines which transition matrix is used:
        - If negative, `matrix1` is used to modify the label.
        - If positive, `matrix2` is employed.

Our experiments in Table 6.1 uses $\eta = 0.3$.

D.2.1.5 Synthetic Noisy Dataset

D.2.1.5.1 Dataset Creation The synthetic dataset used in the experiment shown in Figure 6.3 was generated using the `make_classification` function from `scikit-learn`, adept at creating complex multiclass classification problems. The dataset consists of 2,000 samples, each with 20 features, where 10 are informative and directly influence the class labels, and the remaining 10 are designed to mimic irrelevant data aspects found in real-world scenarios. The data is divided into three classes, each centred around a single cluster to simplify the classification task and emphasise the impact of label noise over inter-class separation. To standardise the dataset, feature scaling was performed using `StandardScaler` from `scikit-learn`.

D.2.1.5.2 Noise Introduction Label noise was introduced at a rate of 41% using a pairwise transition method, where each label is cyclically shifted to the next. The introduction process was controlled using a fixed random seed (`random_state=42`), ensuring consistent and reproducible noisy labels across different experiments.

D.2.1.5.3 Model Description The neural network used in the experiment comprised two hidden layers with 64 neurons each and a ReLU activation function followed by an output layer for three-class classification. The model is trained using Stochastic Gradient Descent (SGD) with a learning rate of 0.1.

Appendix E

Noise-Tolerant Loss Functions

E.1 The Noise-Tolerance Theorem

In this section, we prove the 'Noise-Tolerance Theorem' (Theorem 7.1.2).

Theorem E.1.1. *Let T be a stochastic matrix describing class-conditional label noise. A loss function L is Noise Tolerant to T if and only if there exists $\lambda > 0$ and $\boldsymbol{c} \in \mathbb{R}^c$ such that for all $\boldsymbol{q} \in \Delta$,*

$$T^T \boldsymbol{L}(\boldsymbol{q}) = \lambda \boldsymbol{L}(\boldsymbol{q}) + \boldsymbol{c} \tag{E.1}$$

Proof. **Notation:** For this proof, we will use a tilde $\widetilde{R}_L$ to denote noisy risks rather than R_L^η., which avoids overly cluttered notation.

($\implies$) We begin with the forward direction, which is to say that we assume we have a loss function L, which is Noise Tolerant to some class-conditional noise model with transition matrix T. By Definition 7.1.1, there exists an increasing function f (defined on $Dom(R_L(\boldsymbol{q}))$) so that for any $p(x,y)$ and $\boldsymbol{q}$

$$\widetilde{R}_L(\boldsymbol{q}) = f(R_L(\boldsymbol{q})).$$

Begin by supposing we have some distributions $p_1(x,y), p_2(x,y)$ and an estimator $\boldsymbol{q}$ and let $R_L^{(1)}(\boldsymbol{q}), R_L^{(2)}(\boldsymbol{q})$ denote the risk obtained by $\boldsymbol{q}$ on $p_1(x,y), p_2(x,y)$ respectively. The noisy risks of $\boldsymbol{q}$ can be expressed as

$$\widetilde{R}_L^{(1)}(\boldsymbol{q}) = f(R^{(1)}(\boldsymbol{q}))$$
$$\widetilde{R}_L^{(2)}(\boldsymbol{q}) = f(R^{(2)}(\boldsymbol{q})).$$

Suppose that we construct a mixture distribution $p_\lambda(x,y) = \lambda p_1(x,y) + (1-\lambda)p_2(x,y)$. The clean risk obtained by $\boldsymbol{q}$ on p_λ is a mixture of the risks on the respective distributions, likewise for the noisy risk;

$$\begin{aligned} R_L^{(\lambda)}(\boldsymbol{q}) &= \lambda R_L^{(1)}(\boldsymbol{q}) + (1-\lambda)R_L^{(2)}(\boldsymbol{q}) \\ \widetilde{R}_L^{(\lambda)}(\boldsymbol{q}) &= \lambda \widetilde{R}^{(1)}(\boldsymbol{q}) + (1-\lambda)\widetilde{R}_2(\boldsymbol{q}). \end{aligned}$$

Using the relation from Equation 7.1 one obtains

$$\begin{aligned} & f(R_L^{(\lambda)}(\boldsymbol{q})) = \widetilde{R}_L^{(\lambda)}(\boldsymbol{q}) \\ \iff & f(\lambda R_L^{(1)}(\boldsymbol{q}) + (1-\lambda)R_L^{(2)}(\boldsymbol{q})) = \lambda \widetilde{R}_L^{(1)}(\boldsymbol{q}) + (1-\lambda)\widetilde{R}_L^{(2)}(\boldsymbol{q}) \\ & \qquad = \lambda f(R_L^{(1)}(\boldsymbol{q})) + (1-\lambda) f(R_L^{(2)}(\boldsymbol{q})) \\ \therefore & f(\lambda R_L^{(1)}(\boldsymbol{q}) + (1-\lambda)R_L^{(2)}(\boldsymbol{q})) = \lambda f(R_L^{(1)}(\boldsymbol{q})) + (1-\lambda) f(R_L^{(2)}(\boldsymbol{q})). \end{aligned}$$

We may conclude from this that f is affine, that is

$$f(x) = mx + k$$

for all $x \in Dom(R_L(\boldsymbol{q}))$. To demonstrate this let $a, b \in Dom(R(\boldsymbol{q}))$. Then

$$\begin{aligned} & f(\lambda a + (1-\lambda)b) = \lambda f(a) + (1-\lambda) f(b) \\ \iff & f(\lambda(a-b)+b) = \lambda(f(a) - f(b)) + f(b). \end{aligned}$$

Letting $x \in [a,b]$ then

$$f(x) = \left(\frac{x-b}{a-b}\right)(f(a) - f(b)) + f(b)$$

which is of the form $mx+k$ since $f(a), f(b)$ are just constants where $m > 0$ by assumption that f is increasing.

$$\begin{aligned} & \widetilde{R}_L(\boldsymbol{q}) = f(R_L(\boldsymbol{q})) \\ \iff & \widetilde{R}_L(\boldsymbol{q}) = mR_L(\boldsymbol{q}) + k \end{aligned}$$

The right-hand side of this is equal to the risk one would obtain from using the loss function $\boldsymbol{L}' \coloneqq m\boldsymbol{L} + \boldsymbol{k}$, by this we mean that for any distribution $p(x,y)$ and estimator $\boldsymbol{q}$,

$$R_{L'}(\boldsymbol{q}) = mR_L(\boldsymbol{q}) + k.$$

However, in order for the relation

$$\widetilde{R}_L(\boldsymbol{q}) = R_{L'}(\boldsymbol{q})$$

to be satisfied for all $\boldsymbol{q}$ and all distributions $p(x,y)$ we know by Lemma A.1.1 that

$$\begin{aligned} \boldsymbol{L}' &= T^{-T}\boldsymbol{L} \\ \implies T^T(m\boldsymbol{L}+\boldsymbol{k}) &= \boldsymbol{L} \\ \implies T^T\boldsymbol{L} &= \lambda\boldsymbol{L}+\boldsymbol{c} \end{aligned}$$

where $\lambda \coloneqq \frac{1}{m} > 0$ and $\boldsymbol{c} = \frac{-T^T\boldsymbol{k}}{m}$.

($\Longleftarrow$) For the converse direction, we start by assuming that we have a loss function which satisfies

$$T^T\boldsymbol{L}(\boldsymbol{q}) = \lambda\boldsymbol{L}(\boldsymbol{q}) + \boldsymbol{c}.$$

for some $\lambda > 0$ and $\boldsymbol{c} \in \mathbb{R}^c$. Given an arbitrary distribution $p(x,y)$ and an estimator $\boldsymbol{q}$ we can write out the noisy risk as

$$\begin{aligned} \widetilde{R}(\boldsymbol{q}) &\coloneqq \int p(x)(\boldsymbol{p}(y \mid x)^T T^T)\boldsymbol{L}(\boldsymbol{q}(x))dx \\ &= \int p(x)(\lambda\boldsymbol{p}(y \mid x)^T\boldsymbol{L}(\boldsymbol{q}(x)))dx + c \\ &= \lambda\int p(x)(\boldsymbol{p}(y \mid x)\cdot\boldsymbol{L}(\boldsymbol{q}(x)))dx + c \\ &=: \lambda R(\boldsymbol{q}) + c \end{aligned}$$

where $c \coloneqq \int p(x)(\boldsymbol{p}(y \mid x)\cdot\boldsymbol{c})dx$ □

Remark E.1.2. *To aid clarity and exposition, we have not gone into the detail of rigorously defining $Dom(R_L)$. We assume that $Dom(R_L)$ consists of all risks which can be obtained by some choice of $p(x,y)$ and $\boldsymbol{q}$ where for simplicity, we assume that $p(x)$ has compact support.*

Remark E.1.3. *The proof of Theorem E.1 relies on the fact that if f is defined for x_1, x_2 then it is defined for all $y \in [x_1, x_2]$. Since, by assumption, f is defined on the domain of R_L, then this is equivalent to saying that $Dom(R_L)$ is path-connected. This may be established by identifying that given two risks in $Dom(R_L)$, an intermediate risk may be obtained by mixing the respective distributions. We omit the details for brevity.*

E.2 Partial Fisher Consistency

In Section 7.2, we showed how, aside from symmetric label noise, no loss function can be simultaneously Noise Tolerant and Fisher consistent. This prompted us to introduce a notion of partial Fisher consistency. We showed that Noise-Tolerant loss functions exist that satisfy this weakened notion of consistency, specifically in the case of binary labels. In this section, we present the proofs of the various claims and lemmas from Section 7.2.

Lemma E.2.1. *Let L be a $A-$Fisher consistent loss function which is Noise Tolerant to class-conditional label noise described as by the transition matrix in Equation 7.5 (without loss of generality letting $b \geq a$). Then*

$$A \subseteq \left[0, \frac{1}{2}\right] \cup \left[\frac{b}{a+b}, 1\right].$$

Note that when $a = b$, this corresponds to the interval $[0,1]$ since fully Fisher consistent loss Noise-Tolerant loss functions do exist for symmetric label noise.

Proof. For each $p \in [0,1]$ take $q^* \in \arg\min_{q\in[0,1]} H(p,q)$ and record whether $q^* < \frac{1}{2}$ or $q^* \geq \frac{1}{2}$. I.e. we record the class predicted by q^*. In the case where there are two or more elements of $\arg\min_{q\in[0,1]} H(p,q)$ which induce different predictions, then L is degenerate. Assume that L is non-degenerate. In this case we can partition $[0,1]$ into disjoint sets B_0, B_1 consisting of all those p where every $q^* \in \arg\min_{q\in[0,1]} H(p,q)$ predicts class 0 and all those where each $q^* \in \arg\min_{q\in[0,1]} H(p,q)$ predicts class 1 respectively (by thresholding at a half). Since L is Noise Tolerant we know that for all $p \in \Delta$,

$$\arg\min_{q\in[0,1]} H(p,q) = \arg\min_{q\in[0,1]} H(T(p),q).$$

(Where Tp is shorthand for taking the first component of the vector $T(p, 1-p)$). We know, therefore, that B_0 and B_1 must each be closed under application of T or T^{-1} since otherwise, we have, e.g. $p \in B_0$ where the equality above fails. One can show that the sets $\left[0, \frac{b}{a+b}\right)$ and $\left(\frac{b}{a+b}, 1\right]$ are mapped to themselves by T - this is demonstrated by using the fact that T is linear and $T(\frac{b}{a+b}, \frac{a}{a+b}) = (\frac{b}{a+b}, \frac{a}{a+b})$ [1]. Since B_0, B_1 are closed

[1] we must additionally assume $1-a-b > 0$ but we showed previously that this must hold if L is Noise Tolerant to T

under T, T^{-1} this leaves four possibilities:

$$\begin{aligned}
&\text{Case 1:} \quad B_0 = \emptyset \quad \text{and} \quad \left[0, \frac{b}{a+b}\right) \cup \left(\frac{b}{a+b}, 1\right] \subseteq B_1, \\
&\text{Case 2:} \quad B_1 = \emptyset \quad \text{and} \quad \left[0, \frac{b}{a+b}\right) \cup \left(\frac{b}{a+b}, 1\right] \subseteq B_0, \\
&\text{Case 3:} \quad \left[0, \frac{b}{a+b}\right) \subseteq B_0 \quad \text{and} \quad \left(\frac{b}{a+b}, 1\right] \subseteq B_1, \\
&\text{Case 4:} \quad \left[0, \frac{b}{a+b}\right) \subseteq B_1 \quad \text{and} \quad \left(\frac{b}{a+b}, 1\right] \subseteq B_0.
\end{aligned}$$

In order for L to be $A-$consistent the set A must satisfy the condition that when $A \ni p < \frac{1}{2}$, all minimisers of $H(p,q)$ satisfy $q < \frac{1}{2}$. It must also satisfy the condition that for $A \ni p \geq \frac{1}{2}$, all minimisers of $H(p,q)$ satisfy $q \geq \frac{1}{2}$. If we use A_0 to denote $A \cap \left[0, \frac{1}{2}\right)$ and $A_1 \coloneqq A \cap \left[\frac{1}{2}, 1\right)$ then this can be expressed succinctly as

$$\begin{aligned}
A_0 \cap B_1 &= \emptyset, \\
A_1 \cap B_0 &= \emptyset.
\end{aligned}$$

With reference to our four cases, one may establish that

$$A = A_0 \cup A_1 \subseteq \left[0, \frac{1}{2}\right] \cup \left[\frac{b}{a+b}, 1\right].$$

□

Below is the example from Section 7.2.2 with a proof of the claim that it is $A-$Fisher consistent on the claimed set.

Example Let $b > a$ where $1 - a - b > 0$, and L be defined

$$\begin{aligned}
L(q,1) &= 1 - q, \\
L(q,2) &= \frac{b}{a} q.
\end{aligned}$$

Then L is Noise Tolerant with respect to label noise with transition matrix in Equation 7.5. Moreover, L is $A-$Fisher consistent where A is as defined on the right-hand side of Equation 7.8. To demonstrate this to oneself, write out the expected loss of a prediction q:

$$\begin{aligned}
H(p,q) &\coloneqq p(1-q) + (1-p)\frac{bq}{a} \\
&= q\left(\frac{b}{a} - \frac{bp}{a} - p\right) + const. \\
&= q\left(\frac{b}{a} - p\left(\frac{b}{a} + 1\right)\right) + const.
\end{aligned}$$

If $\left(\frac{b}{a} - p\left(\frac{b}{a}+1\right)\right) < 0 \iff (b - p(b+a)) < 0 \iff p > \frac{b}{a+b}$ the expected loss is minimised by setting $q = 1$. Otherwise, by setting $q = 0$. Using the language of Lemma 7.2.1 we have $B_0 := \left[0, \frac{b}{a+b}\right)$ and $B_0 := \left(\frac{b}{a+b}, 1\right]$. Hence, by setting $A := \left[0, \frac{1}{2}\right] \cup \left[\frac{b}{a+b}, 1\right]$ we ensure that for any $p \in A$ that minimising the pointwise $L-$risk (i.e. the expected loss) aligns with Bayes-optimal decisions.

Lemma E.2.2. *Suppose that the transition matrix T satisfies the property that $\exists \boldsymbol{p} \in \Delta$ such that $\arg\max_i (T\boldsymbol{p})_i \neq \arg\max_i \boldsymbol{p}_i$. Then, there are no Fisher consistent loss functions which are Noise Tolerant with respect to T.*

Proof. A loss function is Noise Tolerant with respect to T if, for any probability estimator $\boldsymbol{q}$, the noisy and clean risks are related via an expression of the form

$$R(\boldsymbol{q}) = \lambda \widetilde{R}(\boldsymbol{q}) + k,$$

where $\lambda > 0$, $k \in \mathbb{R}$. It follows that for any $\boldsymbol{p}, \boldsymbol{q}$ we have the following relationship between the pointwise risks:

$$H(\boldsymbol{p}, \boldsymbol{q}) = \lambda H(T\boldsymbol{p}, \boldsymbol{q}) + k.$$

Thus, $\forall \boldsymbol{p} \in \Delta$,

$$\arg\min_{\boldsymbol{q} \in \Delta} H(\boldsymbol{p}, \boldsymbol{q}) = \arg\min_{\boldsymbol{q} \in \Delta} H(T\boldsymbol{p}, \boldsymbol{q}).$$

Let $\boldsymbol{p}'$ satisfy the condition that $\arg\max_i (T\boldsymbol{p}')_i \neq \arg\max_i \boldsymbol{p}'_i$. Then it cannot be simultaneously true that a minimiser of the pointwise risk

$$\boldsymbol{q}^* \in \arg\min H(\boldsymbol{p}, \boldsymbol{q}) (= \arg\min_{\boldsymbol{q}} H(T\boldsymbol{p}, \boldsymbol{q}))$$

satisfies both

$$\arg\max_i \boldsymbol{q}^*_i = \arg\max_i \boldsymbol{p}_i$$
$$\text{and} \quad \arg\max_i \boldsymbol{q}^*_i = \arg\max_i T\boldsymbol{p}_i$$

Hence L is not Fisher consistent. □

E.3 Multiclass Setting

E.3.0.1 Conjecture Proof for $c = 3$

Given class-conditional label noise expressed by some transition matrix T, assume that we have a Noise-Tolerant loss function L. Since L is Noise Tolerant it lies in a translate of one of the the eigenspaces of T^T. Assume, by contradiction, that this eigenspace has dimension $1 (= c-2)$. Hence $\boldsymbol{L}(\boldsymbol{q})$ lies within a line contained in $\mathbb{R}^3$. It follows that there exists some real-valued function $f(\boldsymbol{q})$ and constants a_1, a_2, a_3 and b_1, b_2, b_3 such that for all $\boldsymbol{q} \in \Delta$,

$$\boldsymbol{L}(\boldsymbol{q}) = (a_1, a_2, a_3) + f(\boldsymbol{q})(b_1, b_2, b_3).$$

This allows us to write out the expected loss of a forecast $\boldsymbol{q}$ with respect to a probability vector $\boldsymbol{p} \in \Delta$,

$$H(\boldsymbol{p}, \boldsymbol{q}) = \boldsymbol{p} \cdot \boldsymbol{L}(\boldsymbol{q}) = f(\boldsymbol{q}) \left(\sum_{i=1}^{3} p_i a_i \right) + \sum_{i=1}^{3} p_i b_i.$$

Crucially, one should see that the minima of $H(\boldsymbol{p}, \boldsymbol{q})$ therefore depends *only* on the sign of $\boldsymbol{p} \cdot (a_1, a_2, a_3)$. Letting $Q_-, Q_+ \subseteq \Delta$ be defined

$$Q_- := \arg\min_{\boldsymbol{q} \in \Delta} f(\boldsymbol{q}),$$
$$Q_+ := \arg\max_{\boldsymbol{q} \in \Delta} f(\boldsymbol{q}),$$

then for $\boldsymbol{p} \in \Delta$,

$$\text{if} \quad \boldsymbol{p} \cdot (a_1, a_2, a_3) > 0 \quad \text{then} \quad \arg\min_{\boldsymbol{q} \in \Delta} H(\boldsymbol{p}, \boldsymbol{q}) = Q_-$$
$$\text{if} \quad \boldsymbol{p} \cdot (a_1, a_2, a_3) < 0 \quad \text{then} \quad \arg\min_{\boldsymbol{q} \in \Delta} H(\boldsymbol{p}, \boldsymbol{q}) = Q_+.$$

It must, therefore, be true that L is degenerate. Specifically, one of the following cases holds:

1. There exist $\boldsymbol{q}^{(1)}, \boldsymbol{q}^{(2)} \in Q_-$ such that

$$\arg\max_i \boldsymbol{q}_i^{(1)} \neq \arg\max_i \boldsymbol{q}_i^{(2)}.$$

2. There exist $\boldsymbol{q}^{(1)}, \boldsymbol{q}^{(2)} \in Q_+$ such that

$$\arg\max_i \boldsymbol{q}_i^{(1)} \neq \arg\max_i \boldsymbol{q}_i^{(2)}.$$

3. For some $k \in \{1,2,3\}$, no $\boldsymbol{q}$ in either Q_- or Q_+ satisfies

$$k = \arg\max_i \boldsymbol{q}_i.$$

Case 3 means that there is some class $k \in \{1,2,3\}$ where, for every $\boldsymbol{p} \in \Delta$, the minimiser of the expected loss never predicts k. Cases 1 and 2 mean that for all $\boldsymbol{p} \in \Delta$, at least two minimisers of the expected loss predict different classes. In all cases, L is degenerate.

E.3.0.2 Improperness of Noise-Tolerant Losses

Definition E.3.1. *A map $f : A \to f(A)$ is a* homeomorphism *if it is bijective (both* injective *and* surjective*),* continuous*, and its inverse f^{-1} is also continuous. This ensures a one-to-one correspondence where small changes in A correspond to small changes in $f(A)$, and the mapping can be smoothly inverted.*

Lemma E.3.2. *Let $\boldsymbol{L} : \Delta \to \boldsymbol{L}(\Delta)$ be a homemorphism. Then L is improper, which means that for all but one choice of $\boldsymbol{p} \in \Delta$, the expected loss $H(\boldsymbol{p}, \boldsymbol{q})$ is uniquely minimised by $\boldsymbol{q}$ in the boundary of the simplex; $\boldsymbol{q} \in \partial\Delta$.*

Proof. Under our assumptions, we may conclude that $\boldsymbol{L}$ has a continuous inverse and is, therefore, a homeomorphism. Since $\boldsymbol{L}$ is a homeomorphism, it maps the boundary of Δ to itself ($\partial\boldsymbol{L}(\Delta) = \boldsymbol{L}(\partial\Delta)$). Geometrically, the expected loss

$$H(\boldsymbol{p}, \boldsymbol{q}) = \boldsymbol{p} \cdot \boldsymbol{L}(\boldsymbol{q}),$$

represents the projection of $\boldsymbol{L}(\boldsymbol{q})$ along the vector $\boldsymbol{p}$. Therefore for any $\boldsymbol{p} \in \Delta$ where $\boldsymbol{p} \not\perp \boldsymbol{L}(\Delta)$,

$$\arg\min_{\boldsymbol{x} \in \boldsymbol{L}(\Delta)} \boldsymbol{p} \cdot \boldsymbol{x} \subseteq \partial\boldsymbol{L}(\Delta).$$

As $\partial\boldsymbol{L}(\Delta) = \boldsymbol{L}(\partial\Delta)$ then for any $\boldsymbol{p} \in \Delta$ where $\boldsymbol{p} \not\perp \boldsymbol{L}(\Delta)$,

$$\arg\min_{\boldsymbol{q} \in \Delta} \boldsymbol{p} \cdot \boldsymbol{L}(\boldsymbol{q}) = \boldsymbol{L}^{-1}\left(\arg\min_{\boldsymbol{x} \in \boldsymbol{L}(\Delta)} \boldsymbol{p} \cdot \boldsymbol{x}\right) \subseteq \boldsymbol{L}^{-1}(\partial\boldsymbol{L}(\Delta)) = \partial\Delta.$$

as claimed. In the case where $\boldsymbol{p} \perp \boldsymbol{L}(\Delta)$,

$$\arg\min_{\boldsymbol{q} \in \Delta} \boldsymbol{p} \cdot \boldsymbol{L}(\boldsymbol{q}) = \Delta,$$

meaning that all forecasts $\boldsymbol{q}$ attain the same expected loss at this point. This can happen for *at most* one setting $\boldsymbol{p} \in \Delta$ since otherwise we have $\boldsymbol{p}_1 \cdot \boldsymbol{L}(\Delta) = 0$ and $\boldsymbol{p}_2 \cdot \boldsymbol{L}(\Delta) = 0$, which means that $\boldsymbol{L}(\Delta)$ has a dimension strictly less than $c-1$ which is a contradiction. □

Corollary E.3.3. *There are no strictly proper Noise-Tolerant loss functions.*

Proof. To be Noise Tolerant $\mathcal{L}(\Delta)$ must lie entirely within one of the eigenspaces of T^T. We know that no eigenspace can have dimension more than $c-1$ since we assume that T describes non-trivial label noise; $T \neq I$. We know that if the dimension is strictly less than $c-1$, L cannot be Fisher consistent, precluding strict properness. Thus, we need only consider the case where the dimension is $c-1$. Thus,L satisfies an equation of the form

$$\boldsymbol{L}(\boldsymbol{q}) \cdot \boldsymbol{\alpha} = a.$$

This allows us to write the expected loss of a forecast $\boldsymbol{q} \in \Delta$ with respect to $\boldsymbol{p}$ as;

$$\begin{aligned}
H(\boldsymbol{p},\boldsymbol{q}) = \boldsymbol{p} \cdot \boldsymbol{L}(\boldsymbol{q}) &= \sum_{i=1}^{c} p_i L_i(\boldsymbol{q}), \\
&= \sum_{i=1}^{c-1} p_i L_i(\boldsymbol{q}) + p_c(a - L_1(\boldsymbol{q}) - \ldots - L_{c-1}(\boldsymbol{q})), \\
&= \sum_{i=1}^{c-1} p_i L_i(\boldsymbol{q}) + (1 - p_1 - p_2 - \ldots - p_{c-1})(a - L_1(\boldsymbol{q}) - \ldots - L_{c-1}(\boldsymbol{q})), \\
&= \sum_{i=1}^{c-1} L_i(\boldsymbol{q})(p_1 + p_2 + \ldots + 2p_i + \ldots + p_{c-1} - 1).
\end{aligned}$$

For brevity we denote $\beta_k \coloneqq (p_1 + p_2 + \ldots + 2p_k + \ldots + p_{c-1} - 1)$. We see that, given two vectors $\boldsymbol{p}^{(1)}, \boldsymbol{p}^{(2)}$ that if there exists $\lambda > 0$ so that

$$(\beta_1^{(1)}, \beta_2^{(1)}, \ldots, \beta_{c-1}^{(1)}) = \lambda(\beta_1^{(2)}, \beta_2^{(2)}, \ldots, \beta_{c-1}^{(2)}),$$

then $H(\boldsymbol{p}^{(1)},\boldsymbol{q}), H(\boldsymbol{p}^{(2)},\boldsymbol{q})$ have the same minima. This would violate strict properness since strict properness requires that $\boldsymbol{p} = \arg\min_{\boldsymbol{q} \in \Delta} H(\boldsymbol{p},\boldsymbol{q})$. However, we can find such a pair of vectors $\boldsymbol{p}^{(1)}, \boldsymbol{p}^{(2)}$. For example, let

$$\begin{aligned}
\boldsymbol{p}^{(1)} &= (\varepsilon, \varepsilon, \ldots, \varepsilon, 1 - (c-1)\varepsilon) \\
\boldsymbol{p}^{(2)} &= (\delta, \delta, \ldots, \delta, 1 - (c-1)\delta)
\end{aligned}$$

where ε, δ are distinct small positive numbers. Here we set $\lambda = \frac{c\delta - 1}{c\varepsilon - 1}$. □

Non-Uniform Noise We provide proof of our claim in Lemma 7.3.1 that no Noise-Tolerant loss function exists for non-uniform label noise. This proof relies on Conjecture 7.2.5.

Lemma E.3.4. *If a loss function L is Noise Tolerant to some label noise model then this label noise must be uniform.*

Proof. We begin by assuming that we have a loss function which is Noise Tolerant to some label noise model represented by a stochastic matrix $T(x)$. We wish to demonstrate that this noise model must be uniform $T(x) = T$. By the definition of Noise Tolerance, we know that for any distribution $p(x,y)$ and estimator $\boldsymbol{q}$, the clean and noisy risks are related by the expression;

$$R^{\eta}(\boldsymbol{q}) = f(R(\boldsymbol{q})).$$

Following the proof of Theorem 7.1.2 one may establish that f must be linear so, for some $m > 0, k$,

$$R^{\eta}(\boldsymbol{q}) = m(R(\boldsymbol{q})) + k.$$

Consider two arbitrary locations $x_1, x_2 \in \mathcal{X}$. Define two distributions $p_1(x,y), p_2(x,y)$ where $p_1(x) = \delta(x_1)$ and $p_2(x) = \delta(x_2)$. The noisy risks can be expressed as

$$\begin{aligned} R_1^{\eta}(\boldsymbol{q}) &= (T(x_1)\boldsymbol{p}_1) \cdot \boldsymbol{L}(\boldsymbol{q}) = mR_1(\boldsymbol{q}) + k = m(\boldsymbol{p}_1 \cdot \boldsymbol{L}(\boldsymbol{q})) + k, \\ R_2^{\eta}(\boldsymbol{q}) &= (T(x_2)\boldsymbol{p}_2) \cdot \boldsymbol{L}(\boldsymbol{q}) = mR_2(\boldsymbol{q}) + k = m(\boldsymbol{p}_2 \cdot \boldsymbol{L}(\boldsymbol{q})) + k. \end{aligned}$$

These equalities hold for all $\boldsymbol{p}_1, \boldsymbol{p}_2$. So for all $\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{q} \in \Delta$

$$\begin{aligned} \boldsymbol{p}_1^T (T^T(x_1) - mI) \cdot \boldsymbol{L}(\boldsymbol{q}) = k, \\ \boldsymbol{p}_2^T (T^T(x_2) - mI) \cdot \boldsymbol{L}(\boldsymbol{q}) = k. \end{aligned}$$

We assume without loss of generality that $k = 0$ (otherwise we may let $L(\boldsymbol{q}, i) \mapsto L(\boldsymbol{q}, i) - \frac{k}{1-m}$) so,

$$\begin{aligned} \boldsymbol{p}_1^T T^T(x_1) \boldsymbol{L}(\boldsymbol{q}) = m\boldsymbol{p}_1^T \boldsymbol{L}(\boldsymbol{q}), \\ \boldsymbol{p}_2^T T^T(x_2) \boldsymbol{L}(\boldsymbol{q}) = m\boldsymbol{p}_2^T \boldsymbol{L}(\boldsymbol{q}). \end{aligned}$$

Setting $\boldsymbol{p}_1 = \boldsymbol{e}_k$ for each $k \in \{1, 2, \ldots, c\}$, we can establish that for every $\boldsymbol{q} \in \Delta$, $\boldsymbol{L}(\boldsymbol{q})$ must lie entirely within the eigenspace of $T^T(x_1)$ associated with the eigenvalue m. We can conclude the same for $T^T(x_2)$. Hence, $\boldsymbol{L}(\Delta)$ lies in the intersection of these subspaces. However, in order for L to be non-degenerate we know that the dimension

of $\boldsymbol{L}(\Delta) \geq c-1$ (Conjecture 7.2.5). It follows that both $T^T(x_1)$ and $T^T(x_2)$ must share a $(c-1)$-dimensional eigenspace for the eigenvalue m in addition to sharing the eigenspace of dimension 1 spanned by $(1,1,\ldots,1)$. Consequently, the matrices are equal: $T^T(x_1) = T^T(x_2)$ and therefore, since x_1, x_2 were arbitrary the label noise is uniform. □

Theorem E.3.5. *Assume that a (non-degenerate) loss function L is Noise Tolerant to some noise model. This noise model must be class-conditional and given by a matrix T, which can be written in the following form:*

$$\begin{bmatrix} 1-\widehat{\eta_1} & \eta_1 & \ldots & \eta_1 \\ \eta_2 & 1-\widehat{\eta_2} & \ldots & \eta_2 \\ \ldots & & & \\ \eta_c & \eta_c & \ldots & 1-\widehat{\eta_c} \end{bmatrix} \tag{E.2}$$

for some $\eta_1, \eta_2, \ldots, \eta_c$ *such that* $\sum_{i=1} \eta_i > 1$. *Where* $\widehat{\eta_k} := \sum_{i=1}^{c} \eta_i - \eta_k$. *Note that* $\eta_i = \frac{\eta}{c-1}$ *gives a matrix corresponding to symmetric label noise.*

Proof. We know from Lemma 7.3.1 that loss functions can only be Noise Tolerant to uniform class-conditional label noise. We assume that conjecture 7.2.5 holds. Thus, the matrix T^T must have two eigenspaces: one of dimension 1, for the eigenvalue 1 and spanned by the vector $\mathbf{1} = (1,1,\ldots,1)$ and another of dimension $c-1$ for some eigenvalue $\lambda > 0$. We let $\boldsymbol{v_2}, \boldsymbol{v_3}, \ldots, \boldsymbol{v_c}$ be a basis of the larger eigenspace. We can assume that $\boldsymbol{v}_i \perp \mathbf{1}$ for $i > 2$ and that the $\boldsymbol{v}_i$ form an orthonormal set of vectors and we denote $\beta := \boldsymbol{v}_2 \cdot \mathbf{1}$. We now attempt to write a general form for the matrix T^T in terms of λ, c, β. Consider some arbitrary vector

$$\boldsymbol{v} = a_1\mathbf{1} + a_2\boldsymbol{v}_2 + a_3\boldsymbol{v}_3 + \ldots + a_c\boldsymbol{v}_c.$$

$\boldsymbol{v} \cdot \mathbf{1} = ca_1 + a_2\beta$ and $\boldsymbol{v} \cdot \boldsymbol{v}_2 = a_1\beta + a_2$. Hence,

$$a_1 = \frac{1}{c-\beta^2}(\boldsymbol{v} \cdot \mathbf{1} - \beta \boldsymbol{v} \cdot \boldsymbol{v}_2)$$

It follows that

$$\begin{aligned} & T^T(\boldsymbol{v} - a_1\mathbf{1}) = \lambda(\boldsymbol{v} - a_1\mathbf{1}), \\ \iff & T^T(\boldsymbol{v}) = a_1(1-\lambda)\mathbf{1} + \lambda\boldsymbol{v} \\ & \qquad = \frac{(1-\lambda)}{c-\beta^2}(\boldsymbol{v} \cdot \mathbf{1} - \beta\boldsymbol{v} \cdot \boldsymbol{v}_2)\mathbf{1} + \lambda\boldsymbol{v}. \end{aligned}$$

The expression $(\boldsymbol{v}\cdot\mathbf{1})\mathbf{1}$ can be written out in matrix form as a matrix of all 1's whereas $\boldsymbol{v}\cdot\boldsymbol{v}_2$ can be expressed as a matrix where each row is $\boldsymbol{v}_2$. Therefore T^T can be written as

$$\lambda I + \frac{(1-\lambda)}{c-\beta^2}A$$

where

$$A = \begin{bmatrix} 1-\beta v_{21} & 1-\beta v_{22} & \dots & 1-\beta v_{2c} \\ 1-\beta v_{21} & 1-\beta v_{22} & \dots & 1-\beta v_{2c} \\ \dots & & & \\ 1-\beta v_{21} & 1-\beta v_{22} & \dots & 1-\beta v_{2c} \end{bmatrix}$$

and v_{2i} sum to β. We remark that the rows of A sum to $c-\beta^2$. Using this observation and reparameterising, T^T can equivalently be written in the form

$$\lambda I + (1-\lambda)A$$

where

$$A = \begin{bmatrix} u_1 & u_2 & \dots & u_c \\ u_1 & u_2 & \dots & u_c \\ \dots & & & \\ u_1 & u_2 & \dots & u_c \end{bmatrix}$$

where the u_i sum to 1. We will do a final reparameterisation to simplify this form further. We let $\eta_i := (1-\lambda)u_i$. The condition that u_i sum to 1 implies that $\lambda = 1-\sum_{i=1}^{c}\eta_i$. Since the eigenvalue must be positive, this imposes the condition that $\sum_{i=1}^{c}\eta_i > 0$. Using this reparameterisation, we conclude

$$T^T = \lambda I + (1-\lambda)A = \begin{bmatrix} \eta_1 + (1-\sum_{i=1}^{c}\eta_i) & \eta_2 & \dots & \eta_c \\ \eta_1 & \eta_2 + (1-\sum_{i=1}^{c}\eta_i) & \dots & \eta_c \\ \dots & & & \\ \eta_1 & \eta_2 & \dots & \eta_c + (1-\sum_{i=1}^{c}\eta_i) \end{bmatrix}$$

which may be written in the form Equation 7.9 using the relevant notation after taking the transpose. □

Bibliography

Adomaityte, U., Defilippis, L., Loureiro, B., & Sicuro, G. (2024). High-dimensional Robust Regression Under Heavy-tailed Data: Asymptotics and Universality. *Journal of Statistical Mechanics: Theory and Experiment*, *2024*(11), 114002.

Alemohammad, S., Casco-Rodriguez, J., Luzi, L., Humayun, A. I., Babaei, H., LeJeune, D., & Siahkoohi, R., Ali & Baraniuk. (2023). Self-Consuming Generative Models Go Mad. In *The Twelfth International Conference On Learning Representations.*

Algan, G., & Ulusoy, I. (2020). Label Noise Types and Their Effects On Deep Learning. *Arxiv Preprint Arxiv:2003.10471*.

Algan, G., & Ulusoy, I. (2021). Image Classification With Deep Learning in the Presence of Noisy Labels: a Survey. *Knowledge-Based Systems*, *215*, 106771.

Amid, E., Warmuth, M. K., Anil, R., & Koren, T. (2019). Robust Bi-tempered Logistic Loss Based On Bregman Divergences. *Advances in Neural Information Processing Systems*, *32*.

Angluin, D., & Laird, P. (1988). Learning From Noisy Examples. *Machine Learning*, *2*, 343–370.

Arazo, E., Ortego, D., Albert, P., O'Connor, N., & McGuinness, K. (2019). Unsupervised Label Noise Modeling and Loss Correction. In *International Conference On Machine Learning* (pp. 312–321).

Aristeidou, M., Herodotou, C., Ballard, H. L., Young, A. N., Miller, A. E., Higgins, L., & Johnson, R. F. (2021). Exploring the Participation of Young Citizen Scientists in Scientific Research: the Case of Inaturalist. *Plos One*, *16*(1), e0245682.

Arpit, D., Jastrzębski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., ... Bengio, Y. (2017). A Closer Look At Memorization in Deep Networks. In *International Conference On Machine Learning.*

Authors. (2021). Enhanced Brain Tumor Classification With Inception V3 and Xception Dual-channel Cnn. *Neural Computing and Applications*. doi: 10.1007/s00521-020-05424-4

Bai, Y., Yang, E., Han, B., Yang, Y., Li, J., Mao, Y., ... Liu, T. (2021). Understanding and Improving Early Stopping for Learning With Noisy Labels. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, & J. W. Vaughan (Eds.), *advances in Neural Information Processing Systems* (Vol. 34, pp. 24392–24403). Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2021/file/cc7e2b878868cbae992d1fb743995d8f-Paper.pdf

Banko, M., & Brill, E. (2001). Scaling to Very Very Large Corpora for Natural Language Disambiguation. In *proceedings of the 39th Annual Meeting of the Association for Computational Linguistics* (pp. 26–33).

Bar, N., Koren, T., & Giryes, R. (2021). Multiplicative Reweighting for Robust Neural Network Optimization. *Arxiv Preprint Arxiv:2102.12192*.

Beede, E., Baylor, E., Hersch, F., Iurchenko, A., Wilcox, L., Ruamviboonsuk, P., & Vardoulakis, L. M. (2020). A Human-centered Evaluation of a Deep Learning System Deployed in Clinics for the Detection of Diabetic Retinopathy. In *proceedings of the 2020 Chi Conference On Human Factors in Computing Systems* (p. 1–12). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3313831.3376718 doi: 10.1145/3313831.3376718

Beigman, E., & Klebanov, B. B. (2009). Learning With Annotation Noise. In *proceedings of the Joint Conference of the 47th Annual Meeting of the Acl and the 4th International Joint Conference On Natural Language Processing of the Afnlp* (pp. 280–287).

Board, N. T. S. (2016). *Preliminary Report: Highway Hwy16fh018* (Preliminary Report No. HWY16FH018). Washington, D.C..

Bodapati, J. D., Shaik, N. S., & Naralasetti, V. (2021). Deep Convolution Feature Aggregation: an Application to Diabetic Retinopathy Severity Level Prediction. *Signal, Image and Video Processing*, *15*, 923–930.

Bootkrajang, J., & Kabán, A. (2012). Label-noise Robust Logistic Regression and Its Applications. In *joint European Conference On Machine Learning and Knowledge Discovery in Databases* (pp. 143–158).

Botta-Dukát, Z. (2005). Rao's Quadratic Entropy As a Measure of Functional Diversity Based On Multiple Traits. *Journal of Vegetation Science*, *16*(5), 533–540.

Byerly, A., Kalganova, T., & Dear, I. (2021). No Routing Needed Between Capsules. *Neurocomputing*, *463*, 545–553.

Bylander, T. (1994). Learning Linear Threshold Functions in the Presence of Classification Noise. In *proceedings of the Seventh Annual Conference On Computational Learning Theory* (pp. 340–347).

Cannings, T. I., Fan, Y., & Samworth, R. J. (2020). Classification With Imperfect Training Labels. *Biometrika*, *107*(2), 311–330.

Chang, H.-S., Learned-Miller, E., & McCallum, A. (2017). Active Bias: Training More Accurate Neural Networks By Emphasizing High Variance Samples. *Advances in Neural Information Processing Systems*, *30*.

Charoenphakdee, N., Lee, J., & Sugiyama, M. (2019). On Symmetric Losses for Learning From Corrupted Labels. In *international Conference On Machine Learning* (pp. 961–970).

Chen, X., & Gupta, A. (2015, December). Webly Supervised Learning of Convolutional Networks. In *proceedings of the Ieee International Conference On Computer Vision (iccv)*.

Cheng, S., Chen, W., Liu, W., Zhou, L., Zhao, H., Kong, W., ... Fu, M. (2024). Dynamic Training for Handling Textual Label Noise. *Applied Intelligence*, 1–16.

Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., ... Pringle, M. (2013). The cancer imaging archive (tcia): Maintaining and operating a public information repository. *Journal of Digital Imaging*, *26*, 1045–1057.

Cohen, E. (1997). Learning Noisy Perceptrons By a Perceptron in Polynomial Time. In *Proceedings 38th Annual Symposium On Foundations of Computer Science* (p. 514-523). doi: 10.1109/SFCS.1997.646140

Cortes, C., & Vapnik, V. (1995). Support-vector Networks. *Machine Learning*, *20*, 273–297.

Cortinhas, S. (2022). *Muffin vs Chihuahua Image Classification.* `https://www.kaggle.com/datasets/samuelcortinhas/muffin-vs-chihuahua-image-classification`. (Accessed: 05/09/2024)

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., & Le, Q. V. (2019). Autoaugment: Learning Augmentation Strategies From Data. In *Proceedings of the IEEE/cvf Conference On Computer Vision and Pattern Recognition* (pp. 113–123).

Cubuk, E. D., Zoph, B., Shlens, J., & Le, Q. V. (2020). Randaugment: Practical Automated Data Augmentation With a Reduced Search Space. In *proceedings of the Ieee/cvf Conference On Computer Vision and Pattern Recognition Workshops* (pp. 702–703).

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum Likelihood From Incomplete Data via the Em Algorithm. *Journal of the Royal Statistical Society: Series B (methodological)*, *39*(1), 1–22.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: a Large-scale Hierarchical Image Database. In *2009 Ieee Conference On Computer Vision and Pattern Recognition* (p. 248-255). doi: 10.1109/CVPR.2009.5206848

Deng, L. (2012). The Mnist Database of Handwritten Digit Images for Machine Learning Research. *IEEE Signal Processing Magazine*, *29*(6), 141–142.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., & Gelly, S. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference On Learning Representations.*

Drenkow, N., Sani, N., Shpitser, I., & Unberath, M. (2021). A Systematic Review of Robustness in Deep Learning for Computer Vision: Mind the Gap? *Arxiv Preprint Arxiv:2112.00639*.

Englesson, E., & Azizpour, H. (2021a). Consistency Regularization Can Improve Robustness to Label Noise. In *International Conference On Machine Learning ICML workshops, 2021 workshop on uncertainty and robustness in deep learning.*

Englesson, E., & Azizpour, H. (2021b). Generalized Jensen-shannon Divergence Loss for Learning With Noisy Labels. *Advances in Neural Information Processing Systems*, *34*, 30284–30297.

Englesson, E., & Azizpour, H. (2024). Robust Classification via Regression for Learning With Noisy Labels. In *ICLR 2024-the Twelfth International Conference On Learning Representations, Messe Wien Exhibition and Congress Center, Vienna, Austria, May 7-11th, 2024.*

Fatras, K., Damodaran, B. B., Lobry, S., Flamary, R., Tuia, D., & Courty, N. (2019). Wasserstein Adversarial Regularization (war) On Label Noise. *Arxiv Preprint Arxiv:1904.03936*.

Feng, C., Tzimiropoulos, G., & Patras, I. (2021). S3: Supervised self-supervised learning under label noise. *Corr*, *abs/2111.11288*. Retrieved from `https://arxiv.org/abs/2111.11288`

Feng, C., Tzimiropoulos, G., & Patras, I. (2024). Noisebox: Towards More Efficient and Effective Learning With Noisy Labels. *IEEE Transactions on Circuits and Systems for Video Technology*, 1-1. doi: 10.1109/TCSVT.2024.3426994

Feng, L., Shu, S., Lin, Z., Lv, F., Li, L., & An, B. (2021). Can Cross Entropy Loss Be Robust to Label Noise? In *proceedings of the Twenty-ninth International Conference On International Joint Conferences On Artificial Intelligence* (pp. 2206–2212).

Ferianc, M., Bohdal, O., Hospedales, T. M., & Rodrigues, M. R. (2024). Navigating Noise: a Study of How Noise Influences Generalisation and Calibration of Neural Networks. *Transactions On Machine Learning Research*, 1–44.

Frenay, B., & Verleysen, M. (2014). Classification in the Presence of Label Noise: a Survey. *IEEE Transactions on Neural Networks and Learning Systems*, *25*(5), 845-869. doi: 10.1109/TNNLS.2013.2292894

Freund, Y., & Schapire, R. E. (1999). Large Margin Classification Using the Perceptron Algorithm. In *machine Learning: Proceedings of the Eleventh Annual Conference* (pp. 209–217). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Fu, X., Huang, K., & Sidiropoulos, N. D. (2018). On Identifiability of Nonnegative Matrix Factorization. *IEEE Signal Processing Letters*, *25*(3), 328-332. doi: 10.1109/LSP.2018.2789405

Fürnkranz, J. (1997). Noise-Tolerant Windowing. In *IJCAI* (pp. 852–859).

Ghosh, A., & Kumar, H. (2017). Robust Loss Functions Under Label Noise for Deep Neural Networks. In *proceedings of the Aaai Conference On Artificial Intelligence* (Vol. 31).

Ghosh, A., Manwani, N., & Sastry, P. (2015). Making Risk Minimization Tolerant to Label Noise. *Neurocomputing*, *160*, 93–107.

Gneiting, T., & Raftery, A. E. (2007). Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, *102*(477), 359–378.

Goldberger, J., & Ben-Reuven, E. (2017). Training Deep Neural-networks Using a Noise Adaptation Layer. In *proceedings of the International Conference On Learning Representations.*

Goodfellow, I. (2016). *Deep Learning.* MIT press.

Gouk, H., Frank, E., Pfahringer, B., & Cree, M. J. (2021). Regularisation of Neural Networks By Enforcing Lipschitz Continuity. *Machine Learning*, *110*, 393–416.

Guo, D., Li, Z., Zhao, H., Zhou, M., & Zha, H. (2022). Learning to Re-weight Examples with Optimal Transport for Imbalanced Classification. *Advances in Neural Information Processing Systems*, *35*, 25517–25530.

Guo, Y., Wang, W., & Wang, X. (2023). A Robust Linear Regression Feature Selection Method for Data Sets With Unknown Noise. *IEEE Transactions on Knowledge and Data Engineering*, *35*(1), 31-44. doi: 10.1109/TKDE.2021.3076891

Han, B., Yao, Q., Liu, T., Niu, G., Tsang, I. W., Kwok, J. T., & Sugiyama, M. (2020). A Survey of Label-noise Representation Learning: Past, Present and Future. *Arxiv Preprint Arxiv:2011.04406*.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., & Sugiyama, M. (2018). Co-Teaching: Robust Training of Deep Neural Networks With Extremely Noisy Labels. *Advances in Neural Information Processing Systems*, *31*.

Harutyunyan, H., Reing, K., Steeg, G. V., & Galstyan, A. (2020, 13–18 Jul). Improving Generalization By Controlling Label-noise Information in Neural Network Weights. In H. D. III & A. Singh (Eds.), *proceedings of the 37th International Conference On Machine Learning* (Vol. 119, pp. 4071–4081). PMLR.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. In *proceedings of the Ieee Conference On Computer Vision and Pattern Recognition* (pp. 770–778).

Hendrycks, D., Mazeika, M., Wilson, D., & Gimpel, K. (2018). Using Trusted Data to Train Deep Networks On Labels Corrupted By Severe Noise. *Advances in Neural Information Processing Systems*, *31*.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer Feedforward Networks Are Universal Approximators. *Neural Networks*, *2*(5), 359–366.

Huang, Y., & Chen, Y. (2020). Autonomous Driving With Deep Learning: a Survey of State-of-art Technologies. *Arxiv Preprint Arxiv:2006.06091*.

Huber, P. J. (1973). Robust Regression: Asymptotics, Conjectures and Monte Carlo. *The Annals of Statistics*, 799–821.

Iscen, A., Valmadre, J., Arnab, A., & Schmid, C. (2022). Learning With Neighbor Consistency for Noisy Labels. In *proceedings of the Ieee/cvf Conference On Computer Vision and Pattern Recognition* (pp. 4672–4681).

Ishida, T., Yamane, I., Sakai, T., Niu, G., & Sugiyama, M. (2020). Do We Need Zero Training Loss After Achieving Zero Training Error? In *International Conference On Machine Learning* (pp. 4604–4614).

Janocha, K., & Czarnecki, W. M. (2016). On Loss Functions for Deep Neural Networks in Classification. *Schedae Informaticae*, *25*, 49.

Jiang, L., Zhou, Z., Leung, T., Li, L.-J., & Fei-Fei, L. (2018). Mentornet: Learning Data-driven Curriculum for Very Deep Neural Networks On Corrupted Labels. In *International Conference On Machine Learning* (pp. 2304–2313).

Johnson, J. M., & Khoshgoftaar, T. M. (2022). A Survey On Classifying Big Data With Label Noise. *ACM Journal of Data and Information Quality*, *14*(4), 1–43.

Jost, L. (2006). Entropy and Diversity. *Oikos*, *113*(2), 363–375.

Khanal, B., & Kanan, C. (2021). How Does Heterogeneous Label Noise Impact Generalization in Neural Nets? In *advances in Visual Computing: 16th International Symposium, Isvc 2021, Virtual Event, October 4-6, 2021, Proceedings, Part Ii* (pp. 229–241).

Kim, H., Chang, H. S., Cho, K., Lee, J., & Han, B. (2024). Learning With Noisy Labels: Interconnection of Two Expectation-Maximizations. *Arxiv Preprint Arxiv:2401.04390*.

Kim, T., Ko, J., Choi, J., & Yun, S.-Y. (2021). Fine samples for learning with noisy labels. *Advances in Neural Information Processing Systems*, *34*, 24137–24149.

Krizhevsky, A., Nair, V., & Hinton, G. (2009). *CIFAR-10 Canadian Institute for Advanced Research).* Available online. Retrieved from `http://www.cs.toronto.edu/~kriz/cifar.html` (Accessed: 2024-09-14)

Krizhevsky, A., Sutskever, I., & Hinton, G. (2012, 01). Imagenet Classification With Deep Convolutional Neural Networks. *Neural Information Processing Systems*, *25*. doi: 10.1145/3065386

Kumar, A., & Amid, E. (2021). Constrained Instance and Class Reweighting for Robust Learning Under Label Noise. *Arxiv Preprint Arxiv:2111.05428*.

Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., … Kolesnikov, A. (2020). The Open Images Dataset V4: Unified Image Classification, Object Detection, and Visual Relationship Detection At Scale. *International Journal of Computer Vision*, *128*(7), 1956–1981.

Lachenbruch, P. A. (1966). Discriminant Analysis When the Initial Samples Are Misclassified. *Technometrics*, *8*(4), 657–662.

Larsen, J., Nonboe, L., Hintz-Madsen, M., & Hansen, L. (1998). Design of Robust Neural Network Classifiers. In *proceedings of the 1998 Ieee International Conference On Acoustics, Speech and Signal Processing, Icassp '98 (cat. No.98ch36181)* (Vol. 2, p. 1205-1208 vol.2). doi: 10.1109/ICASSP.1998.675487

Lawrence, N., & Schölkopf, B. (2001). Estimating a Kernel Fisher Discriminant in the Presence of Label Noise. In *18th International Conference On Machine Learning ICML 2001)* (pp. 306–306).

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based Learning Applied to Document Recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324. doi: 10.1109/5.726791

Lee, K., Yun, S., Lee, K., Lee, H., Li, B., & Shin, J. (2019, 09–15 Jun). Robust Inference via Generative Classifiers for Handling Noisy Labels. In K. Chaudhuri & R. Salakhutdinov (Eds.), *Proceedings of the 36th International Conference On Machine Learning* (Vol. 97, pp. 3763–3772). PMLR. Retrieved from `https://proceedings.mlr.press/v97/lee19f.html`

Li, J., Socher, R., & Hoi, S. C. (2020). Dividemix: Learning With Noisy Labels As Semi-supervised Learning. In *International Conference On Learning Representations.*

Li, M., Soltanolkotabi, M., & Oymak, S. (2020). Gradient Descent With Early Stopping Is Provably Robust to Label Noise for Overparameterized Neural Networks. In *international Conference On Artificial Intelligence and Statistics* (pp. 4313–4324).

Li, X., Liu, T., Han, B., Niu, G., & Sugiyama, M. (2021). Provably End-to-end Label-noise Learning Without Anchor Points. In *International Conference On Machine Learning* (pp. 6403–6413).

Li, X.-C., Xia, X., Zhu, F., Liu, T., yao Zhang, X., & lin Liu, C. (2023). *Dynamic Loss for Learning With Label Noise.*

Liu, S., Niles-Weed, J., Razavian, N., & Fernandez-Granda, C. (2020). Early-learning Regularization Prevents Memorization of Noisy Labels. *Advances in Neural Information Processing Systems*, *33*, 20331–20342.

Liu, T., & Tao, D. (2015). Classification With Noisy Labels By Importance Reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, *38*(3), 447–461.

Liu, Y., Cheng, H., & Zhang, K. (2023, 23–29 Jul). Identifiability of Label Noise Transition Matrix. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *proceedings of the 40th International Conference On Machine Learning* (Vol. 202, pp. 21475–21496). PMLR. Retrieved from `https://proceedings.mlr.press/v202/liu23g.html`

Liu, Y., & Guo, H. (2020, 13–18 Jul). Peer Loss Functions: Learning From Noisy Labels Without Knowing Noise Rates. In H. D. III & A. Singh (Eds.), *proceedings of the 37th International Conference On Machine Learning* (Vol. 119, pp. 6226–6236). PMLR. Retrieved from https://proceedings.mlr.press/v119/liu20e.html

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The Extended Cohn-kanade Dataset (ck+): a Complete Dataset for Action Unit and Emotion-specified Expression. In *2010 Ieee Computer Society Conference On Computer Vision and Pattern Recognition-workshops* (pp. 94–101).

Ma, X., Huang, H., Wang, Y., Erfani, S. R. S., & Bailey, J. (2020). Normalized Loss Functions for Deep Learning With Noisy Labels. In *proceedings of the 37th International Conference On Machine Learning*. JMLR.org.

Ma, X., Wang, Y., Houle, M. E., Zhou, S., Erfani, S., Xia, S., … Bailey, J. (2018, 10–15 Jul). Dimensionality-driven Learning With Noisy Labels. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th International Conference On Machine Learning* (Vol. 80, pp. 3355–3364). PMLR. Retrieved from https://proceedings.mlr.press/v80/ma18d.html

Mahsereci, M., Balles, L., Lassner, C., & Hennig, P. (2017). Early Stopping Without a Validation Set. *Arxiv Preprint Arxiv:1703.09580*.

Majidi, N., Amid, E., Talebi, H., & Warmuth, M. K. (2021). Exponentiated Gradient Reweighting for Robust Training Under Label Noise and Beyond. *Arxiv Preprint Arxiv:2104.01493*.

Malach, E., & Shalev-Shwartz, S. (2017). Decoupling" When to Update" From" How to Update". *Advances in Neural Information Processing Systems*, *30*.

Manwani, N., & Sastry, P. S. (2013). Noise Tolerance Under Risk Minimization. *IEEE Transactions on Cybernetics*, *43*(3), 1146-1151. doi: 10.1109/TSMCB.2012.2223460

Meng, D., Zhao, Q., & Jiang, L. (2015). What Objective Does Self-paced Learning Indeed Optimize? *Arxiv Preprint Arxiv:1511.06049*. Retrieved from https://arxiv.org/abs/1511.06049

Menon, A., Rooyen, B. V., Ong, C. S., & Williamson, B. (2015, 07–09 Jul). Learning From Corrupted Binary Labels via Class-probability Estimation. In F. Bach & D. Blei (Eds.), *Proceedings of the 32nd International Conference On Machine Learning* (Vol. 37, pp. 125–134). Lille, France: PMLR. Retrieved from https://proceedings.mlr.press/v37/menon15.html

Menon, A. K., Rawat, A. S., Reddi, S. J., & Kumar, S. (2020). Can Gradient Clipping Mitigate Label Noise? In *international Conference On Learning Representations*.

Mnih, V., & Hinton, G. E. (2012). Learning to Label Aerial Images From Noisy Data. In *proceedings of the 29th International Conference On Machine Learning ICML-12)* (pp. 567–574).

Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., & Sutskever, I. (2021). Deep Double Descent: Where Bigger Models and More Data Hurt. *Journal of Statistical Mechanics: Theory and Experiment*, *2021*(12), 124003.

Natarajan, N., Dhillon, I. S., Ravikumar, P. K., & Tewari, A. (2013). Learning With Noisy Labels. *Advances in Neural Information Processing Systems*, *26*.

Nettleton, D. F., Orriols-Puig, A., & Fornells, A. (2010). A Study of the Effect of Different Types of Noise On the Precision of Supervised Learning Techniques. *Artificial Intelligence Review*, *33*, 275–306.

Nguyen, D. T., Mummadi, C. K., Ngo, T. P. N., Nguyen, T. H. P., Beggel, L., & Brox, T. (2019). Self: Learning to Filter Noisy Labels With Self-ensembling. In *International Conference On Learning Representations.*

Nigam, N., Dutta, T., & Gupta, H. P. (2020). Impact of Noisy Labels in Learning Techniques: a Survey. In *advances in Data and Information Sciences* (pp. 403–411). Springer.

Nishi, K., Ding, Y., Rich, A., & Hollerer, T. (2021, June). Augmentation Strategies for Learning With Noisy Labels. In *Proceedings of the IEEE/cvf Conference On Computer Vision and Pattern Recognition CVPR* (p. 8022-8031).

Norris, J. R. (1998). *Markov Chains*. Cambridge: Cambridge University Press.

Ovcharov, E. Y. (2018). Proper Scoring Rules and Bregman Divergences. *Bernoulli*, *24*(1), 53–79.

Patrini, G., Rozza, A., Krishna Menon, A., Nock, R., & Qu, L. (2017). Making Deep Neural Networks Robust to Label Noise: a Loss Correction Approach. In *Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition* (pp. 1944–1952).

Pechenizkiy, M., Tsymbal, A., Puuronen, S., & Pechenizkiy, O. (2006). Class Noise and Supervised Learning in Medical Domains: the Effect of Feature Extraction. In *19th Ieee Symposium On Computer-based Medical Systems (cbms'06)* (p. 708-713). doi: 10.1109/CBMS.2006.65

Pereira, E., Carneiro, G., & Cordeiro, F. R. (2022). A Study On the Impact of Data Augmentation for Training Convolutional Neural Networks in the Presence of Noisy Labels. In *2022 35th Sibgrapi Conference On Graphics, Patterns and Images (sibgrapi)* (Vol. 1, p. 25-30). doi: 10.1109/SIBGRAPI55357.2022.9991791

Pi, T., Li, X., Zhang, Z., Meng, D., Wu, F., Xiao, J., & Zhuang, Y. (2016). Self-paced Boost Learning for Classification. In *ijcai* (pp. 1932–1938).

Prechelt, L. (2002). Early Stopping-but When? In *neural Networks: Tricks of the Trade* (pp. 55–69). Springer.

Reed, S., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., & Rabinovich, A. (2014). Training Deep Neural Networks On Noisy Labels With Bootstrapping. *Arxiv Preprint Arxiv:1412.6596*.

Ren, M., Zeng, W., Yang, B., & Urtasun, R. (2018). Learning to Reweight Examples for Robust Deep Learning. In *International Conference On Machine Learning* (pp. 4334–4343).

Rice, L., Wong, E., & Kolter, Z. (2020, 13–18 Jul). Overfitting in Adversarially Robust Deep Learning. In H. D. III & A. Singh (Eds.), *proceedings of the 37th International Conference On Machine Learning* (Vol. 119, pp. 8093–8104). PMLR. Retrieved from https://proceedings.mlr.press/v119/rice20a.html

Rolnick, D., Veit, A., Belongie, S., & Shavit, N. (2017). Deep Learning Is Robust to Massive Label Noise. *Arxiv Preprint Arxiv:1705.10694*.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning Representations By Back-propagating Errors. *Nature*, *323*(6088), 533–536.

Sachdeva, R., Cordeiro, F. R., Belagiannis, V., Reid, I., & Carneiro, G. (2021, January). Evidentialmix: Learning With Combined Open-set and Closed-set Noisy Labels. In *Proceedings of the IEEE/cvf Winter Conference On Applications of Computer Vision (wacv)* (p. 3607-3615).

Savage, L. J. (1971). Elicitation of Personal Probabilities and Expectations. *Journal of the American Statistical Association*, *66*(336), 783–801.

Scott, C., Blanchard, G., & Handy, G. (2013, 12–14 Jun). Classification With Asymmetric Label Noise: Consistency and Maximal Denoising. In S. Shalev-Shwartz & I. Steinwart (Eds.), *Proceedings of the 26th Annual Conference On Learning Theory* (Vol. 30, pp. 489–511). Princeton, NJ, USA: PMLR. Retrieved from `https://proceedings.mlr.press/v30/Scott13.html`

Shanthini, A., Vinodhini, G., Chandrasekaran, R., & Supraja, P. (2019). A Taxonomy On Impact of Label Noise and Feature Noise Using Machine Learning Techniques. *Soft Computing*, *23*, 8597–8607.

Shen, Y., & Sanghavi, S. (2019, 09–15 Jun). Learning With Bad Training Data via Iterative Trimmed Loss Minimization. In K. Chaudhuri & R. Salakhutdinov (Eds.), *proceedings of the 36th International Conference On Machine Learning* (Vol. 97, pp. 5739–5748). PMLR. Retrieved from `https://proceedings.mlr.press/v97/shen19e.html`

Shu, J., Xie, Q., Yi, L., Zhao, Q., Zhou, S., Xu, Z., & Meng, D. (2019). Meta-weight-net: Learning an Explicit Mapping for Sample Weighting. *Advances in Neural Information Processing Systems*, *32*.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, *abs/1409.1556*.

Song, H., Kim, M., & Lee, J.-G. (2019). SELFIE: Refurbishing Unclean Samples for Robust Deep Learning. In *International Conference On Machine Learning* (pp. 5907–5915).

Song, H., Kim, M., Park, D., & Lee, J. (2019). How Does Early Stopping Help Generalization Against Label Noise? Arxiv 2019. *Arxiv Preprint Arxiv:1911.08059*.

Song, H., Kim, M., Park, D., Shin, Y., & Lee, J.-G. (2023). Learning From Noisy Labels With Deep Neural Networks: a Survey. *IEEE Transactions on Neural Networks and Learning Systems*, *34*(11), 8135-8153. doi: 10.1109/TNNLS.2022.3152527

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a Simple Way to Prevent Neural Networks From Overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929–1958.

Stempfel, G., & Ralaivola, L. (2009). Learning Svms From Sloppily Labeled Data. In *artificial Neural Networks–icann 2009: 19th International Conference, Limassol, Cyprus, September 14-17, 2009, Proceedings, Part I 19* (pp. 884–893).

Stephenson, C., & Lee, T. (2021). When and How Epochwise Double Descent Happens. *Arxiv Preprint Arxiv:2108.12006*.

Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and Policy Considerations for Deep Learning in nlp. *Corr*, *abs/1906.02243*. Retrieved from `http://arxiv.org/abs/1906.02243`

Suganyadevi, S., Seethalakshmi, V., & Balasamy, K. (2022). A Review On Deep Learning in Medical Image Analysis. *International Journal of Multimedia Information Retrieval*, *11*(1), 19–38.

Sukhbaatar, S., Bruna, J., Paluri, M., Bourdev, L., & Fergus, R. (2015). Training Convolutional Networks With Noisy Labels. In *3rd International Conference On Learning Representations, Iclr 2015.*

Sukhbaatar, S., & Fergus, R. (2014). Learning From Noisy Labels With Deep Neural Networks. *Arxiv Preprint Arxiv:1406.2080*, *2*(3), 4.

Sun, X., Zhang, S., & Ma, S. (2024). Prediction Consistency Regularization for Learning With Noise Labels Based On Contrastive Clustering. *Entropy*, *26*(4), 308.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., … Rabinovich, A. (2015). Going Deeper With Convolutions. In *proceedings of the Ieee Conference On Computer Vision and Pattern Recognition* (pp. 1–9).

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. In *2016 Ieee Conference On Computer Vision and Pattern Recognition CVPR* (p. 2818-2826). doi: 10.1109/CVPR.2016.308

Tanveer, M. S., Khan, M. U. K., & Kyung, C.-M. (2021). Fine-tuning Darts for Image Classification. In *2020 25th International Conference On Pattern Recognition (icpr)* (pp. 4789–4796).

Van Rooyen, B., Menon, A., & Williamson, R. C. (2015). Learning With Symmetric Label Noise: the Importance of Being Unhinged. *Advances in Neural Information Processing Systems*, *28*.

Veselovsky, V., Ribeiro, M. H., & West, R. (2023). Artificial Artificial Artificial Intelligence: Crowd Workers Widely Use Large Language Models for Text Production Tasks. *Arxiv Preprint Arxiv:2306.07899*.

Vyas, N., Saxena, S., & Voice, T. (2020). Learning Soft Labels via Meta Learning. *Arxiv Preprint Arxiv:2009.09496*.

Wang, D.-B., Wen, Y., Pan, L., & Zhang, M.-L. (2021). Learning From Noisy Labels With Complementary Loss Functions. In *proceedings of the Aaai Conference On Artificial Intelligence* (Vol. 35, pp. 10111–10119).

Wang, Q., Yao, J., Gong, C., Liu, T., Gong, M., Yang, H., & Han, B. (2021). Learning With Group Noise. In *proceedings of the Aaai Conference On Artificial Intelligence* (Vol. 35, pp. 10192–10200).

Wang, X., Hua, Y., Kodirov, E., Clifton, D. A., & Robertson, N. M. (2023). IMAE for Noise-robust Learning: Mean Absolute Error Does Not Treat Examples Equally and Gradient Magnitude's Variance Matters. In *ICLR 2023 Workshop On Trustworthy and Reliable Large-scale Machine Learning Models.*

Wang, Y., Liu, W., Ma, X., Bailey, J., Zha, H., Song, L., & Xia, S.-T. (2018, June). Iterative Learning With Open-set Noisy Labels. In *proceedings of the Ieee Conference On Computer Vision and Pattern Recognition CVPR.*

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., & Bailey, J. (2019, nov). Symmetric Cross Entropy for Robust Learning With Noisy Labels. In *2019 IEEE/cvf International Conference On Computer Vision (iccv)* (p. 322-330). Los Alamitos, CA, USA: IEEE Computer Society. Retrieved from https://doi.ieeecomputersociety.org/10.1109/ICCV.2019.00041 doi: 10.1109/ICCV.2019.00041

Wei, H., Tao, L., Xie, R., & An, B. (2021). Open-Set Label Noise Can Improve Robustness Against Inherent Label Noise. *Advances in Neural Information Processing Systems*, *34*, 7978–7992.

Wei, J., Liu, H., Liu, T., Niu, G., Sugiyama, M., & Liu, Y. (2021). To Smooth Or Not? When Label Smoothing Meets Noisy Labels. *Arxiv Preprint Arxiv:2106.04149*.

Whitla, P. (2009). Crowdsourcing and Its Application in Marketing Activities. *Contemporary Management Research*, *5*(1).

Williams, R. (2023, June 22). The People Paid to Train Ai Are Outsourcing Their Work to Ai. *Technology Review*. Retrieved from https://www.technologyreview.com/2023/06/22/1075405/the-people-paid-to-train-ai-are-outsourcing-their-work-to-ai/

Wolpert, D., & Macready, W. (1997). No Free Lunch Theorems for Optimization. *IEEE Transactions on Evolutionary Computation*, *1*(1), 67-82. doi: 10.1109/4235.585893

Xia, X., Han, B., Wang, N., Deng, J., Li, J., Mao, Y., & Liu, T. (2023). Extended T: Learning With Mixed Closed-set and Open-Set Noisy Labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(3), 3047-3058. doi: 10.1109/TPAMI.2022.3180545

Xia, X., Liu, T., Han, B., Gong, C., Wang, N., Ge, Z., & Chang, Y. (2021). Robust Early-learning: Hindering the Memorization of Noisy Labels. In *international Conference On Learning Representations 2021* (pp. 1–15).

Xia, X., Liu, T., Wang, N., Han, B., Gong, C., Niu, G., & Sugiyama, M. (2019). Are Anchor Points Really Indispensable in Label-noise Learning? *Advances in Neural Information Processing Systems*, *32*.

Xiao, T., Xia, T., Yang, Y., Huang, C., & Wang, X. (2015, June). Learning From Massive Noisy Labeled Data for Image Classification. In *Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition CVPR*.

Xu, Y., Cao, P., Kong, Y., & Wang, Y. (2019). L_dmi: a Novel Information-theoretic Loss Function for Training Deep Nets Robust to Label Noise. *Advances in Neural Information Processing Systems*, *32*.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). Xlnet: Generalized Autoregressive Pretraining for Language Understanding. *Advances in Neural Information Processing Systems*, *32*.

Yao, Y., Liu, T., Han, B., Gong, M., Deng, J., Niu, G., & Sugiyama, M. (2020). Dual T: Reducing Estimation Error for Transition Matrix in Label-noise Learning. *Advances in Neural Information Processing Systems*, *33*, 7260–7271.

Yu, Q., & Aizawa, K. (2020). Unknown Class Label Cleaning for Learning With Open-set Noisy Labels. In *2020 IEEE International Conference On Image Processing (icip)* (p. 1731-1735). doi: 10.1109/ICIP40778.2020.9190652

Yuan, S., Feng, L., & Liu, T. (2024). Early Stopping Against Label Noise Without Validation Data. In *the Twelfth International Conference On Learning Representations.*

Yun, S., Oh, S. J., Heo, B., Han, D., Choe, J., & Chun, S. (2021). Re-labeling Imagenet: From Single to Multi-labels, From Global to Localized Labels. In *Proceedings of the IEEE/CVPR Conference On Computer Vision and Pattern Recognition* (pp. 2340–2350).

Zhang, C., Bengio, S., Hardt, M., Recht, B., & Vinyals, O. (2021). Understanding Deep Learning (Still) Requires Rethinking Generalization. *Communications of the Acm*, *64*(3), 107–115.

Zhang, H., Cheng, N., Zhang, Y., & Li, Z. (2021). Label Flipping Attacks Against Naive Bayes On Spam Filtering Systems. *Applied Intelligence*, *51*(7), 4503–4514.

Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2017). Mixup: Beyond Empirical Risk Minimization. *Arxiv E-prints*, arXiv–1710.

Zhang, Y., Zheng, S., Wu, P., Goswami, M., & Chen, C. (2021). Learning With Feature-dependent Label Noise: a Progressive Approach. In *international Conference On Learning Representations.*

Zhang, Z., & Sabuncu, M. (2018). Generalized Cross Entropy Loss for Training Deep Neural Networks With Noisy Labels. *Advances in Neural Information Processing Systems*, *31*.

Zheng, G., Awadallah, A. H., & Dumais, S. T. (2019). Meta Label Correction for Learning With Weak Supervision. *Corr*, *abs/1911.03809*. Retrieved from `http://arxiv.org/abs/1911.03809`

Zheng, S., Wu, P., Goswami, A., Goswami, M., Metaxas, D., & Chen, C. (2020). Error-Bounded Correction of Noisy Labels. In *international Conference On Machine Learning* (pp. 11447–11457).

Zhou, T., Wang, S., & Bilmes, J. (2020). Robust Curriculum Learning: From Clean Label Detection to Noisy Label Self-correction. In *International Conference On Learning Representations.*

Zhou, X., Liu, X., Jiang, J., Gao, X., & Ji, X. (2021, 18–24 Jul). Asymmetric Loss Functions for Learning With Noisy Labels. In M. Meila & T. Zhang (Eds.), *proceedings of the 38th International Conference On Machine Learning* (Vol. 139, pp. 12846–12856). PMLR. Retrieved from `https://proceedings.mlr.press/v139/zhou21f.html`