This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e.g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

- This work is protected by copyright and other intellectual property rights, which are retained by the thesis author, unless otherwise stated.
- A copy can be downloaded for personal non-commercial research or study, without prior permission or charge.
- This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author.
- The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.
- When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.
Applying Machine Learning to Astronomical Surveys For Anomaly Detection and Population Analysis

Dennis A Crake

Doctor of Philosophy
The University of Edinburgh
04/2023
Abstract

This thesis explores the use of state-of-the-art computational analysis techniques in astronomy. This thesis begins by exploring the ability of machine learning techniques to identify stellar clusters within the Milky Way disc. Stellar clusters offer insight into stellar evolution and the history of our Galaxy. Using the latest Gaia photometry and astrometry, I identify \( \sim \) 2000 stellar clusters within the Galactic plane and provide age estimates. The challenges faced while identifying stellar clusters inspired a novel clustering approach with potential applications beyond stellar populations or astronomy in general. Thus, I continue the thesis by continuing to develop the approach to produce a generalised software suitable for various clustering tasks. The software, HEADSS, offers a user-friendly package that partitions and restitches the clustering analysis to reduce peak computational requirements while avoiding introduced artefacts. The principal achievements of this software are the scalability and repeatability of all results, which, until now, have not been possible with standardised software.

An intriguing insight stellar clusters offer is constraining the Initial-Final Mass Relation (IFMR) of stellar remnants, namely White Dwarf stars. An accurate IFMR of stellar remnants represents an empirical ground truth for stellar evolutionary models. Increasing the populous of remnants with known initial mass further constrains this relation, which further constrains the modelling of processes that drive extreme mass loss during the very late stages of stellar evolution. Current IFMRs contain White Dwarfs predominantly from a restricted sample of globular clusters. Building on earlier work within the thesis, I investigate increasing the IFMR population with White Dwarfs from within the Galactic place, representing a largely unexplored population.

Finally, the thesis pivots towards a new objective in outlier detection. Outlier...
detection is a powerful tool for identifying extreme or novel objects within the immense databases from current and future survey telescopes. The rapid and accurate identification of scientifically compelling patterns within the transient universe is vital for spectroscopic observations. I present an Unsupervised Random Forest to identify objects displaying a range of anomalous variability patterns. Combining the results with Gaia photometry verifies a relation between the Anomaly metric and the evolutionary stage of objects, allowing the identification of abnormal behaviours within specific classes. Furthermore, the analysis explores the physical mechanisms that drive the anomaly score, such as the orbital parameters of eclipsing binaries.
Lay Summary

The night sky has captured attention throughout human history. One of the earliest astronomical recordings is within a stone carving dating back over 5,000 years. In the following millennia, astronomy had no technological advancements until the invention of the telescope in the 17th century. The telescope revolutionised astronomy by revealing stars too faint to see by eye. Today, coupled with digital imagery, astronomy remains built on the same fundamental principles, with modern telescopes capturing faint objects in ever more detail. The advancements have captured incredible images and made significant discoveries, including other worlds orbiting alien stars, the birth of stars, the formation of galaxies containing billions of stars, and much more.

Astronomy is in the midst of another technological revolution. As the latest generation of telescopes shifts away from observing single objects, for hours at a time, towards focusing on large areas of the sky. The result is an enormous amount of data. We have detailed maps of our Galaxy, the Milky Way, containing billions of stars and images of the deepest depths of the universe. The result is that we now have many observations of billions of objects, providing increasingly complex data. This data has the potential to transform our understanding of the universe. Nevertheless, meaningfully analysing such vast volumes is a scientific and technological challenge.

Whilst always being a blend of a theoretical and observational field, astronomy has often relied on chance encounters with objects that challenge our understanding. With the current increase of available data, this practice cannot continue, as numerous fascinating discoveries will remain undiscovered. This thesis addresses this issue by presenting two projects that use cutting-edge machine learning techniques as a scalable approach. By focusing on stars within the Milky Way, they aim to address two demanding challenges within stellar astrophysics. Existing data is analysed to develop the methods while ensuring they can tackle
the challenges brought in by next-generation telescopes.

The first project focuses on identifying clusters of stars within the Milky Way. Stellar clusters form during the collapse of giant clouds of hydrogen found throughout the Galaxy’s disc. As the clouds collapse, regions become very dense and ignite, creating a population of stars with common properties. Chapter 3 focuses on training a series of machine learning models to remove unreliable data, identify and validate stellar clusters, and finally provide an age estimate of each cluster. This work relies on methods that can handle large volumes of data repeatably, i.e. do not rely on judgement or manual processes.

Chapter 2 expands on a novel clustering method developed in the previous chapter. Clustering algorithms require a lot of computation power that is often not feasible or available. The clustering approach reduces the requirements by splitting the data into standard partitions, offering a solution to a common problem faced by clustering big data. The chapter outlines the method with example datasets to demonstrate the potential beyond this thesis.

Chapter 4 showcases just one astrophysical use case of the stellar clusters identified earlier in the thesis. Once stars have exhausted their fuel reserves (hydrogen), they evolve by fusing heavier and heavier elements. Eventually, this process reaches a limit, and the star can no longer generate energy. During the final stages of evolution, the star loses its outer layers, leaving just the hot core. The isolated core is known as a White Dwarf star which has a much lower mass than the original star. Roughly 95% of all stars will eventually become a white dwarf star. As these final stages are very complex, calculating the original mass is challenging. This chapter uses the stellar clusters from Chapter 3 to understand the history of observed white dwarf stars allowing an estimate of the original mass.

Finally, Chapter 5 represents a second major project within this thesis. With the volume of data increasing at such a rate, identifying new or unknown objects is increasingly difficult. This chapter attempts to identify the most unusual objects within a dataset using a machine learning algorithm. Similar to the previous work within this thesis, the focus is to ensure the results do not rely on judgement or manual processes. The results discover a range of unusual objects and explore the processes that cause these distinctive behaviours. I identify a link between the likelihood of an object being unconventional and the evolutionary stage.
Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

Parts of this work have been published in:
Chapter 2 - This Chapter forms the main body of Crake et al. (2023), accepted in Astronomy and Computing.
Chapter 5 - This work forms the full text of Crake & Martínez-Galarza (2023) & an upcoming conference proceeding from the Machine Learning for Astrophysics conference. The unsupervised random forest method featured within this chapter was previously developed for Kepler light curves during my MPhys degree under the supervision of Dr Martínez-Galarza at the Harvard & Smithsonian Centre for Astrophysics (CfA). The analysis contributed to a wider collaboration comparing outlier detection methods within astronomy for time series analysis. This work was finalised and accepted as two submissions during my time at Edinburgh (Martínez-Galarza et al., 2021; Tirumala et al., 2021). The early work and Kepler results do not directly contribute towards this thesis, rather Chapter 5 represents a continuation of the collaboration with Dr Martínez-Galarza during my time in Edinburgh.

(Dennis A Crake, 04/2023)
Acknowledgements

I would like to express my deepest gratitude to both Professor Robert Mann & Dr Nigel Hambly. Without either of you, this endeavour would not have been remotely possible. With your expertise and support, you have allowed my creativity to flourish, without ever losing sight of the end goal. Throughout all the highs and lows, including a global pandemic, you have always helped me in ways I never expected. It has been a pleasure working with you both, I am ever grateful for the opportunity you have given me and I look forward to what the future brings.

I am also extremely grateful to Dr Juan Rafael Martínez-Galarza for your continued collaboration and friendship, your patience and guidance led to my first publication. It was our meeting during my MPhys research that led to the pursuit of this PhD, so, thank you for being such an inspiration and leading me to pursue this adventure.

A special thank goes to my family for always believing in me and teaching me how to overcome even the largest hurdles. To both my parents, beyond being fantastic and inspirational people, thank you for always being there when needed, I could ask no one better to have raised me and guided me along this crazy path. It goes without saying that I owe a huge debt to Caroline. Not only have you been there for every step, but always pushing me to aim higher and achieve greater things. I am ever grateful for the constant sanity checks, always making me laugh, and seeing the bright side of life.

I would like to extend my thanks to all my collaborators, I hope we have to opportunity to continue. A huge thank you goes to my cohort and office mates for all the fun and keeping me sane over the past few years. Finally, I would like to acknowledge the generous support of STFC who funded this work and allowed me to experience a range of conferences.
Contents

Abstract i
Lay Summary iii
Declaration v
Acknowledgements vi
Contents vii
List of Figures xii
List of Tables xix

1 Introduction 1

1.1 Sky Survey Datasets ................................................................. 1

1.1.1 A Brief History of Astronomical Catalogues ....................... 1

1.1.2 Astrometric Filters, Passbands & Spectra .......................... 4

1.1.3 Attenuation & Reddening ............................................ 7

1.1.4 Notable Modern Survey Telescopes .................................. 9

1.2 Stellar Evolution ...................................................................... 12

1.2.1 The Hertzsprung-Russell Diagram ................................. 12

1.2.2 Early Stellar Evolution ............................................... 14
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5</td>
<td>Removing Artefacts</td>
<td>69</td>
</tr>
<tr>
<td>2.6</td>
<td>Computational Performance</td>
<td>72</td>
</tr>
<tr>
<td>2.7</td>
<td>Broader Impact</td>
<td>74</td>
</tr>
<tr>
<td>2.8</td>
<td>Conclusions</td>
<td>76</td>
</tr>
<tr>
<td>3</td>
<td>A Scalable Approach to Identifying Stellar Clusters in the Milky Way Plane</td>
<td>77</td>
</tr>
<tr>
<td>3.1</td>
<td>Introduction</td>
<td>77</td>
</tr>
<tr>
<td>3.2</td>
<td>Classifying Spurious Astrometric Sources</td>
<td>78</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Identification of Training Data</td>
<td>81</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Training the Model</td>
<td>83</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Identification of All Spurious Sources</td>
<td>84</td>
</tr>
<tr>
<td>3.3</td>
<td>Identifying Stellar Clusters &amp; Co-moving Groups</td>
<td>88</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Clustering Preprocessing</td>
<td>89</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Clustering the Milky Way</td>
<td>93</td>
</tr>
<tr>
<td>3.3.3</td>
<td>Identifying Spurious or Contaminated Clusters</td>
<td>99</td>
</tr>
<tr>
<td>3.4</td>
<td>Cluster Age Determination</td>
<td>103</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Training Data</td>
<td>105</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Training Data Preprocessing</td>
<td>107</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Training the Network</td>
<td>111</td>
</tr>
<tr>
<td>3.4.4</td>
<td>Ageing Network Performance</td>
<td>115</td>
</tr>
<tr>
<td>3.4.5</td>
<td>Comparison To Recent Literature</td>
<td>118</td>
</tr>
<tr>
<td>3.5</td>
<td>Discussion</td>
<td>120</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Summary</td>
<td>122</td>
</tr>
</tbody>
</table>
Constraining the Initial-Final Mass Relation of White Dwarf Stars

4.1 Introduction

4.2 Identifying White Dwarf Stars
   4.2.1 Training Optimisation
   4.2.2 White Dwarf Detection Performance
   4.2.3 White Dwarf Detection for All Sources

4.3 White Dwarf Evolution
   4.3.1 The Cooldown Models

4.4 Initial-Final Mass Relation of White Dwarf Stars
   4.4.1 Calculating Initial Masses
   4.4.2 The Initial-Final Mass Relation
   4.4.3 The High-Mass IFMR

4.5 Summary & Conclusions

Outlier Detection

5.1 Introduction

5.2 Datasets
   5.2.1 TESS light curves
   5.2.2 Gaia Photometry and Astrometry

5.3 Anomalous Time Series Detection Algorithm
   5.3.1 Feature Extraction
   5.3.2 Feature Optimisation
   5.3.3 Optimisation of Hyperparameters
   5.3.4 Implementation and Hardware
5.4 Results ............................................................................. 160
5.4.1 The emergence of distinct populations ......................... 162
5.4.2 Consistency Across Sectors ........................................ 166
5.5 Linking anomaly scores to astrophysical properties: the Colour-Magnitude diagram .................................................. 167
5.6 Most Anomalous Objects Identified .................................. 170
5.6.1 Visual Inspection of Light Curves and Classification ........ 170
5.6.2 Composition of Outlying Population ............................ 172
5.6.3 Non-persistent anomalies ............................................. 176
5.7 Analysing Evolutionary Stages Across the Hertzsprung-Russel Diagram .............................................................. 178
5.7.1 TESS Objects of Interest ............................................. 179
5.7.2 Multi-Body Systems ................................................... 181
5.7.3 Pulsating Variable Stars .............................................. 183
5.7.4 Young Stellar Objects ................................................. 187
5.7.5 Giants ..................................................................... 188
5.7.6 White Dwarf Stars ..................................................... 191
5.7.7 Modulated light curves ............................................... 195
5.7.8 Artefacts ................................................................ 196
5.8 Discussion ......................................................................... 198
5.9 Conclusion ........................................................................ 200

6 Conclusion & Future Work .................................................. 203

A Definitions of Key Parameters and Data ............................ 207

Bibliography ........................................................................ 212
List of Figures

1.1 Visualisation of UBVRI Johnson passbands across the visible spectrum ........................................... 5

1.2 Simple spectra of the Sun with Atmospheric Absorption alongside a detailed echelle spectra highlighting the presence of absorption lines ................................................. 6

1.3 Observations of the dark cloud Bernard 68 in multiple passbands to highlight the difference in extinction ................................................................. 8

1.4 Visualisation of Gaia passbands in comparison to the visible spectrum ......................................................... 10

1.5 Example Colour-Absolute Magnitude Diagram from Gaia. The Main Sequence, Red Giant Branch and White Dwarf populations are labelled as the dominant structures ................................. 13

1.6 Comparison of the energy generation rates of nuclear fusion processes within stars, highlighting the higher temperatures required for helium fusion ........................................ 16

1.7 Example evolutionary track for a 6$M_\odot$ star, showing the changes in observational properties across the evolutionary track .......... 22

1.8 Example isochrone models for stellar masses ranging between 1.0–15.0$M_\odot$ .................................................. 25

1.9 Examples of galaxy types highlighting the differences between Elliptical, Lenticular, Spiral and Irregular types. Credit: NASA .................................................. 26

1.10 Velocity dispersion of the major stellar populations within the Milky Way .................................................. 27

1.11 Diagram of the Milky Way, denoting the approximate location of the thin disc, thick disc, galactic bulge, globular clusters and halo stars ................................................................. 28
1.12 Visualisation of distributed computational architecture, highlighting the capability of shared memory and worker nodes

32

1.13 Flowchart of a general machine learning workflow

38

1.14 Matrix of correlation between select Gaia features. This method is ideal for selecting training data when there is limited domain knowledge

43

1.15 An exaggerated example of feature importance histogram using the same example as Figure 1.14

44

1.16 Example Confusion matrix, showing the meaning of common terms for classification successes. A perfect classification is entirely True Positives and True Negatives, creating an Identity matrix

48

1.17 Demonstration of ROC feature space for model evaluation and thresholding curves for classification models

50

1.18 Graphic explaining the function of layers within a neural network. The layer types included are Fully Connected Dense Layers, Flatten, Max Pooling and Convolutional

52

1.19 Graphic of a decision trees classification process translated onto the effective splitting of feature space

53

1.20 Graphic demonstrating the step-by-step process of DBSCAN clustering with a simplified example dataset

55

1.21 Graphic explaining the process of Hierarchical clustering algorithms. An example dataset and the corresponding dendrogram are shown

56

2.1 2D visualisation of the splitting layers for an \( N = 2, 3 \& 4 \) base layer. For \( N = 2 \), the Quaternary layer remains unused

62

2.2 Visualisation of 2D stitching partitions that maximise the distance to a splitting boundary for \( N = 2, 3 \& 4 \). The colour represents the splitting layer hierarchy

62

2.3 Scaling of the number of partitions required for increasing \( N \) with 2 features and number of features with an \( N = 2 \) base layer, highlighting the importance of minimising the number of features

64

2.4 Example datasets used to evaluate the performance of HEADSS in conjunction with HDBSCAN

66

2.5 Clusters identified in the example datasets by HDBSCAN when evaluating the full dataset

68
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.6</td>
<td>Clusters identified in the example datasets by HDBSCAN with an $N = 2$ implementation of HEADSS</td>
</tr>
<tr>
<td>2.7</td>
<td>Clusters identified in the example datasets by HDBSCAN with an $N = 2$ implementation of HEADSS including the merger capabilities</td>
</tr>
<tr>
<td>2.8</td>
<td>Comparison of non-distributed and theoretical distributed run-times for HEADSS compared to the standard implementation of HDBSCAN for various values of $N$ highlighting the potential improvements for large datasets</td>
</tr>
<tr>
<td>3.1</td>
<td>Colour-Magnitude Diagram of uncorrected <em>Gaia DR3</em> sources. The presence of astrometrically spurious sources obscures the usual structure visible</td>
</tr>
<tr>
<td>3.2</td>
<td>Spatial distribution of Good and Spurious training sources. Good sources are selected from HEALPix pixels where no spurious sources are present, causing the two classes to occupy independent regions of the sky</td>
</tr>
<tr>
<td>3.3</td>
<td>Confusion matrix from the identification of spurious <em>Gaia</em> sources with accompanying ROC curve. The high performance is reflected by the area under the curve and a moderately liberal approach to false positives is identified</td>
</tr>
<tr>
<td>3.4</td>
<td>All sources spatial distribution of Good vs Spurious sources grouped by distances. The training data distributions are not reflected and confirm that spurious sources overpopulate crowded regions</td>
</tr>
<tr>
<td>3.5</td>
<td>All sources CAMD showing the distribution of Good vs Spurious sources split into distances revealing the recovered evolutionary populations</td>
</tr>
<tr>
<td>3.6</td>
<td>Velocity distribution of all sources within 1kpc in the local standard of rest, with the extended tail of thick disc and halo stars dominating above $50 - 100$ km/s</td>
</tr>
<tr>
<td>3.7</td>
<td>2D Visualisation of the Primary, Secondary and Tertiary layers used to split the feature space for clustering tasks</td>
</tr>
<tr>
<td>3.8</td>
<td>2D visualisation of the stitching boundaries for a 2x2 data split where the cluster centre points determine the entire clustering assignment</td>
</tr>
<tr>
<td>3.9</td>
<td>2D visualisation of partition clustering of the Milky Way disc for $</td>
</tr>
<tr>
<td>Number</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>3.10</td>
<td>Visualisation of the resultant clustering after the restitching the partitions created by the splitting process shown in Figure 3.9</td>
</tr>
<tr>
<td>3.11</td>
<td>Demonstration of Hough transformations, highlighting the difference between Cartesian and Hough feature space for a single point and a line of points</td>
</tr>
<tr>
<td>3.12</td>
<td>Example line detection with Hough transformations for a cluster identified with HDBSCAN</td>
</tr>
<tr>
<td>3.13</td>
<td>CAMD diagrams of randomly selected clusters grouped, by the calculated probability of true cluster, $P(C_G)$, by the cluster validation analysis</td>
</tr>
<tr>
<td>3.14</td>
<td>3D map of dust within the Milky Way from <em>dustmaps</em> package ([Green et al., 2019]) centred on Cygnus X</td>
</tr>
<tr>
<td>3.15</td>
<td>Figure to show the contrast between the observed and synthetic clusters in the training data for the neural network estimating cluster ages</td>
</tr>
<tr>
<td>3.16</td>
<td>Normalised performance metrics for ageing stellar clusters across model architectures outlined in Table 3.2</td>
</tr>
<tr>
<td>3.17</td>
<td>Scatter plots of true cluster ages vs predicted ages for real and synthetic clusters. Error bars represent the spread of age predictions</td>
</tr>
<tr>
<td>3.18</td>
<td>Comparison of cluster parameters identified in this work to matched clusters in <em>He et al., 2022</em></td>
</tr>
<tr>
<td>4.1</td>
<td>Representation of mass loss over the stellar lifetime to showcase the dramatic mass loss within the final stages of stellar evolution for mid to low-mass stars</td>
</tr>
<tr>
<td>4.2</td>
<td>Workflow for obtaining the IFMR, going from a database to the final IFMR. This approach includes the work in Chapter 3</td>
</tr>
<tr>
<td>4.3</td>
<td>Distribution of known white dwarfs against a sample of other sources on a CAMD</td>
</tr>
<tr>
<td>4.4</td>
<td>Spatial distribution of white dwarf detection training data with reliable photometric data</td>
</tr>
<tr>
<td>4.5</td>
<td>Feature importance of final features selected for final training model for the identification of white dwarf stars</td>
</tr>
<tr>
<td>4.6</td>
<td>Confusion matrix from WD detection training sets with an accompanying CAMD emphasising the regions of confusion</td>
</tr>
</tbody>
</table>
4.7 ROC curve for the identification of white dwarf stars. The large area under the curve reflects the high performance with the zoomed inset concluding that minimal improvements are possible by altering classification thresholds.

4.8 CAMD of WD vs non-WD populations as classified by the trained random forest.

4.9 White dwarf Mass-Radius relation for non-relativistic and relativistic models, demonstrating the origin of the Chandrasekhar mass.

4.10 Example cooldown curves for white dwarfs with hydrogen and helium dominated atmospheres. The tracks highlight the importance of reliable mass estimates.

4.11 Stellar lifetime estimates from PARSEC isochrones. Used to calculate the initial mass of white dwarf progenitors. Ages are estimated using a 1-dimensional spline.

4.12 Resulting IFMR split into DA (Top) and DB type stars (Bottom) with background IFMR reference from Cummings et al. (2018) shown.

4.13 CAMD of the white dwarfs featured in Figure 4.12 with matching colour coding. The distribution indicates that unresolved binarity is affecting the IFMR.

4.14 IFMR split into DA and DB-type stars, with unresolved binary systems corrected.

5.1 Visualisation of the observation segments (sectors) in equatorial coordinates. Each sector is observed for 27 days, with a cadence of 2 minutes. Overlaps at the poles result in targets being observed in multiple sectors.

5.2 Example real and synthetic datasets for both photometric flux and periodicity data. The non-linear sampling periods can be seen along the periodogram.

5.3 Importance score for each feature during the training of the random forest using data from Sector 1. Initial results revealed periods less than 4 hours have negligible importance.

5.4 Example light curves of outliers identified in Sectors 18–24 without well-defined classification labels.

5.5 Stacked histogram of all 24 Sectors colour-coded by sector. The overall distribution is consistent across all sectors, with 85% of objects assigned an Anomaly Score below 0.6.
5.6 Distribution of the standard deviation of anomaly scores for objects observed in multiple sectors. Overall low deviations are observed with $\bar{\sigma}(\sigma_{\text{Anomaly}}) = 0.0166$.

5.7 CAMD of all TESS objects, colour-coded by their average anomaly score across all sectors. There is a clear correlation between the Anomaly Score and stellar evolutionary stage.

5.8 The distribution of visual descriptions for objects with anomaly scores above 0.9 and lacking prior classification.

5.9 A sample of periodograms and the corresponding location on the CAMD for labelled outliers, unlabelled outliers and bulk objects. Noticeably, the three CAMD populations occupy distinct regions.

5.10 A visualisation of the standard deviation of anomaly score between sectors vs the average anomaly score. The intermediate population show a higher standard deviation due to the smaller population.

5.11 Distribution of descriptions for high standard deviation objects that have an anomaly score above 0.9 in at least one sector.

5.12 Examples of the most interesting object types within the high standard deviation population, including irregular patterns, transit events, long period variables, and artefacts.

5.13 Set of normalised histograms comparing orbital parameters for exoplanet systems within the bulk and outlying populations.

5.14 Stacked histogram of average anomaly scores for binary stars compared to TOI’s with example binary light curves.

5.15 Representation of candidate features that drive anomaly scores in binary systems. Feature analysed include primary transit depth, secondary transit depth and orbital period.

5.16 Histogram of Anomaly score for RR-Lyrae, Cepheid Variable, and other variable stars across all sectors.

5.17 Light curves of outliers within the pulsating type stars. Types represented include RR-Lyrae, Cepheid Variables, $\delta$-Scuti, and $\gamma$-Dor stars.

5.18 Histogram of anomaly scores for young stellar objects with example outlying light curves.

5.19 Histogram of anomaly scores for various Giant star types. The customarily bulk populations identify a small number of outliers showing particularly interesting light curves.
5.20 Example light curves of Giant stars from both the bulk and outlier populations. Examples include Red Giant, Red Supergiant and Blue Supergiant stars.

5.21 CAMD colour-coded by anomaly score, zoomed in on the white dwarf stars revealing little to no correlation along the length of the cooldown track.

5.22 Histogram of average anomaly scores for White Dwarf stars with light curves of identified example outliers. Interestingly, the light curves lack obvious defining features.

5.23 Example light curves of modulated light curves showing a range of patterns. The variability differs in both frequency and magnitude effects, suggesting multiple astrophysical mechanisms.

5.24 A CAMD colour-coded by average anomaly score with the position of modulated light curves highlighted indicating modulation patterns occur across many stages of stellar evolution.

5.25 Light curve examples showing the most common types of artefact found, including trends, spikes, and background removal issues.
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>HDBSCAN hyperparameters for complete dataset clustering</td>
<td>67</td>
</tr>
<tr>
<td>2.2</td>
<td>Optimised HDBSCAN hyperparameters for N=2 HEADSS region clustering</td>
<td>68</td>
</tr>
<tr>
<td>2.3</td>
<td>Merging hyperparameters for test datasets</td>
<td>70</td>
</tr>
<tr>
<td>3.1</td>
<td>$R/R_v$ values used to calculate dust extinction, [Wang &amp; Chen (2019)]</td>
<td>109</td>
</tr>
<tr>
<td>3.2</td>
<td>Number of Convolution, Max Pooling, Dense layer types and activation function within each model evaluated within the optimisation analysis</td>
<td>111</td>
</tr>
<tr>
<td>4.1</td>
<td>Final results for the IFMR of white dwarf stars</td>
<td>145</td>
</tr>
<tr>
<td>5.1</td>
<td>Top ten rows of the final results table, included are the key identifiers, anomaly score, sectors observed, classification within each sector alongside any observations on the light curves and a high variance flag</td>
<td>164</td>
</tr>
<tr>
<td>5.2</td>
<td>List of common descriptors used in the classification of anomalous objects</td>
<td>171</td>
</tr>
<tr>
<td>5.3</td>
<td>A list of known light curves of interest within the TESS literature and their associated Anomaly Scores</td>
<td>202</td>
</tr>
<tr>
<td>A.1</td>
<td>Descriptions of key Gaia DR3 columns in §3.2</td>
<td>208</td>
</tr>
<tr>
<td>A.2</td>
<td>Description of PySpark’s MultilayerPerceptronClassifier key parameters in §3.2</td>
<td>209</td>
</tr>
<tr>
<td>A.3</td>
<td>Description of key parameters of HDBSCAN in §3.3</td>
<td>209</td>
</tr>
<tr>
<td>A.4</td>
<td>Description of TensorFlow’s Sequential key parameters in §3.4.3</td>
<td>210</td>
</tr>
<tr>
<td>A.5</td>
<td>Descriptions of key Gaia DR3 columns in §4.2.1</td>
<td>210</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

This thesis concerns the use of machine learning to address several problems arising in the analysis of sky survey datasets. This Introduction starts by presenting a summary of the evolutionary histories followed by stars of different masses, see §1.2, thereby introducing many of the stellar classes that are encountered in later science chapters. §1.3 summarises current evidence for the distribution of different stellar populations in the Milky Way, while §1.1 describes the sky survey datasets that have facilitated the development of this picture of our Galaxy. The scale of sky survey datasets is increasing rapidly, therefore survey astronomy is facing a number of computational challenges that, as discussed in §1.4, must be addressed through a combination of hardware and software developments. Amongst the latter is the application of machine learning algorithms to address problems of classification and regression, as described in §1.5. Finally, §1.6 summarises the structure of this thesis, outlining the contents of each of the remaining chapters.

1.1 Sky Survey Datasets

1.1.1 A Brief History of Astronomical Catalogues

The first known stellar catalogues are attributed to Ancient Greece, namely Hipparchus of Nicaea (190 - 120 BC). Unfortunately, the contents of these catalogues are unknown, but it is understood to have contained the position
and brightness of around 850 stars. The oldest surviving stellar catalogue is from *Ptolemy (100 – 170 AD)* in the *Almagest*, which also is the first known use of the term “magnitude”. Ptolemy defined six “magnitude” levels ranging from the most bright “First Magnitude” stars to the faintest “Sixth Magnitude” stars. Besides documenting more extensive catalogues, the next breakthrough came with the invention of the telescope by *Galileo Galilei (1564 - 1642)* marking the dawn of the *Telescopic Age*. The telescope revealed previously invisible stars, marking the first discovery of stars fainter than Ptolemy’s Sixth Magnitude. Despite the improvements in completeness, significant improvements in accuracy continued with rigorous studies by *William Hershel (1738 - 1822)*, where he developed many catalogues and instruments.

The invention of photometers steered astronomers away from the visual magnitude scale to a mathematically defined system. In the mid 19th century, *Norman Pogson (1829 – 1891)* demonstrated first magnitude stars are $\sim 100$ times brighter than Sixth Magnitude stars, resulting in a factor of $\sqrt[5]{100} \approx 2.512$ between each magnitude band. The scale is logarithmic to match how the human eye responds to light intensity. This system is defined such that a higher numerical value represents a less bright star and allows negative magnitudes for objects brighter than the zero-point stars. This system measures the difference in brightness between *object 1* and *object 2* is formally defined as:

$$m_1 - m_2 = -2.5 \log_{10} \left( \frac{L_1}{L_2} \right),$$

where $L$ is the luminosity and $m$ is the apparent magnitude of *object 1* and *object 2* respectively.

Due to the intensity of light reducing proportionally to distance squared ($I \propto d^{-2}$), the magnitude system depends on the distance to the observer, with magnitudes measured from the observer’s position defined as *apparent magnitude*. In practice, comparing the intrinsic brightness of objects offers great scientific value, a measurement known as *absolute magnitude*, first published in *Kapteyn (1902)* by *Jacobus Kapteyn (1851 – 1922)*. Qualitatively, absolute magnitude is the magnitude of an object when observed from 10 parsecs, whereas the quantitative definition is:

$$m - M = 5 - 5 \log_{10}(d),$$

where $M$ is the absolute magnitude, $d$ is the distance in parsecs, $(m - M)$ is often
referred to as the distance modulus. The notion of absolute magnitude was widely accepted by 1905 after the work of Ejnar Hertzsprung (1873 – 1967). The author encourages the reader to refer to Hughes (2006) for a complete summary of the history of absolute magnitude. Nonetheless, while allowing a direct comparison of objects, the limitation of absolute magnitude is that it also requires an accurate distance measurement.

Measuring the distance to stars is non-trivial due to the vast distances involved. The origins of distance measurements date back to Nicolaus Copernicus (1473 – 1543) and the heliocentric model he developed. Copernicus developed this model independently from Aristarchus of Samos (310 – 230 BC), who first suggested this model some eighteen centuries earlier. This model presents a trigonometrical measurement of a star’s distance using the movement of Earth around the sun, called parallax. As more closer objects will be displaced by a greater angle, parallax measures the observed movement of a star compared to a background of extremely distant objects. A baseline of one astronomical unit (AU), defined as the distance between the Earth and Sun, and half the total angle ($\alpha$) between the apparent stellar position from the two observations taken six months apart, defines a triangle with a known angle and opposite side length. This results in the distance to an observed star being calculated by:

$$d = \frac{1\text{AU}}{p},$$

where $d$ is the distance to the observed star and $p$ is the parallax angle, defined as $p = \frac{1}{2}\alpha$. As the parallax angles are small, it is assumed that $\tan p = p$. It was not until the 19th century that a catalogue of reliable parallax measurements was published in Lundmark (1932), although all parallax measurements contained significant errors until the turn of the century.

With modern technologies came the invention of the charged coupled device (CCD) camera, marking the end of the reliance on photometric measurements using the human eye (even during the early 20th century, where photographic plates gained popularity). Modern CCDS offer high accuracy and vast capabilities, marking a particular improvement in photometry measurements. Yet, CCDS do not possess a similar response curve to light intensive as the human eye. While the eye responds logarithmically, the CCD pixels respond linearly with a particular dynamic range. This transition brought a new intensity metric known as flux, which is a linear measurement of observed brightness.
to the historical measurements described above, magnitudes remain prevalent throughout astronomy, therefore we convert from flux to magnitude using:

\[ m_x = -2.5 \log_{10} \left( \frac{F_x}{F_{x,0}} \right), \]

where \( m_x \) is the magnitude, \( F_x \) is the measured flux and \( F_{x,0} \) is the zero point flux for the used filter.

This brief history covers the evolution of astronomical observation and the origin of common practices within the field. For an in-depth review and further references, I refer the reader towards Miles (2007).

1.1.2 Astrometric Filters, Passbands & Spectra

Astronomical objects emit light across the entire spectrum, from Radio to X-ray. All telescopes are only sensitive to a window of the spectrum due to the range of wavelengths and energies. Yet, the ability to focus on specific wavelengths reveals otherwise unobtainable astronomical features. There are two main types, broad and narrowband filters. Broadband (or passband) filters allow a wide range of frequencies to pass through, often providing a measurement of a given colour. Broadband measurements are required for the calculation of physical properties, such as temperature, and for producing all colour images.

Within the broadband filter space, there are many systems with the most common being the Johnson-Cousins filters. These split the visible wavelengths into 5 distinct passbands, UBVRI which stand for ultraviolet, blue, visual, red, infrared (Bessell & Brett, 1988). This system has been expanded into the near-infrared with passbands JHKLM which correspond to regions of low atmospheric absorption (Bessell & Brett, 1988). Figure 1.1 offers a visualisation of the UBVRI transmission curves. While these systems are very commonly used, modern telescopes possess a unique passband system that is optimised for both the mission goal and instrument response. Thus, the passband response curve must be well understood to produce accurate measurements of targets and meaningful data.

Whereas, narrowband filters focus on specific emission lines which can act as proxy measurements for physical attributes. Photons from specific emission lines represent a known electron decay, allowing the measurement of specific elements or compounds within a source. Common examples include \( H\alpha \), emitted by
Figure 1.1  Visualisation of UBVRI Johnson passbands across the visible spectrum. A transmission of 100 equates to total transparency and zero corresponds to totally opaque.

molecular hydrogen, which acts as a trace for the star formation rate within galaxies as described in Glazebrook et al. (1999). Another common narrowband filter is $OIII$, emitted by doubly ionised oxygen, which measures gaseous structures such as planetary nebulae. These filters can be used for measurements of physical processes, measuring chemical presences and combined to create colour images that focus on specific targets.

An improvement on narrowband filters is spectroscopy. Spectroscopy takes measurements of narrow regions (known as the spectral resolution) across a wide spectrum often covering all visible wavelengths. This can be either an emission or absorption spectra depending on the source. Spectra are taken by spectroscopes which use diffraction gratings to split the light to measure the intensity at each wavelength. Figure 1.2 shows example spectra of the Sun.

Due to the splitting of the source, spectra require brighter sources than typical imagery but offer direct measurements of the source’s physical attributes. For example, stars (particularly degenerate stellar remnants) emit as near-perfect black body sources. Hence, Wien’s Law allows the direct measurement of the
Continuum Solar spectrum of the Sun with atmospheric absorption.

Figure 1.2  
Left: Simple visualisation of the Sun’s spectra, highlighting the peak intensity with (red) and without (yellow) the absorption from the Earth’s atmosphere. Credit: Penn State [CC BY-SA 3.0], via Wikimedia Commons. Right: Detailed echelle spectrum of the Sun representing wavelengths from 400 to 700 nm. The presence of absorption lines is clearly visible across the continuum. Credit: NASA

Surface temperature. Wien’s law is defined as:

$$\lambda_{\text{max}} T = b,$$

where $\lambda_{\text{max}}$ is peak intensity wavelength, $T$ is the surface temperature, and $b$ is the Wien’s constant ($2.897 \times 10^{-3}$ mK). Properties of stars are also related through the Stefan-Boltzmann Law, described as:

$$L = 4\pi R^2 \sigma T^4,$$

where $L$ is the luminosity, $R$ is the radius, $T$ is the temperature and $\sigma$ is the Stefan-Boltzmann constant ($5.67 \times 10^{-8}$ Wm$^{-2}$K$^{-4}$).

Combining Wien’s Law with the Stefan-Boltzmann Law allows a star’s radius to be inferred. Where the Stefan-Boltzmann Law is defined as:

$$L = 4\pi R^2 \sigma T^4,$$

where $L$ is the luminosity, $R$ is the radius, $T$ is the temperature and $\sigma$ is the
Stefan-Boltzmann constant ($5.67 \times 10^{-8}$ Wm$^{-2}$K$^{-4}$). Thus, combining these two relations allows the stellar radius to be inferred through observational properties alone. Further information provided by spectra includes the chemical composition, through the presence of emission/absorption lines, and line of sight velocities, through Doppler shifts, which in turn allow the measurement of rotation curves across galaxies.

Overall, much of modern astronomy is not possible without the use of passband filters, particularly spectra. The additional information provided by such measurements is vital for understanding both galactic and extra-galactic sources. In this work the majority of data involves broadband observations, however, some measurements are calculated using spectra or similar methods.

### 1.1.3 Attenuation & Reddening

Attenuation within astronomy accounts for the difference between the spectrum emitted by a source and the spectrum detected by an observer due to the effects of scattering and absorption. Scattering occurs due to particles, such as dust or molecular clouds, between the emitting object and the observer.

The intensity of light scattering, depends on the wavelength and size of the particles. If the size of the particle is much smaller than the wavelength of light, then the intensity of scattered light, $I_{\text{scatter}}$, is given by the Rayleigh Scattering Law:

$$I_{\text{scatter}} \propto \frac{1}{\lambda^4},$$

where $\lambda$ is the wavelength (Rayleigh 1899). It is this dependence on the wavelength that causes the reddening effects. As an aside, Rayleigh scattering is the effect that causes the sky to appear blue, where the scattered light is observed rather than the unscattered light. Dust particles are much larger than the molecules and are of comparable size to the wavelength of visible light ($\sim 1\mu m$). For these particles:

$$I_{\text{scatter}} \propto \frac{1}{\lambda}. $$
Figure 1.3 Observations of the dark cloud Bernard 68 in multiple passbands to highlight the difference in extinction. Notice that the region is completely opaque in shorter wavelengths, i.e. all light has been scattered, compared to the number of visible stars in the longest wavelengths. Credit: ESO

For these relatively larger particles, there is much less dependence on wavelength. Finally, if particles are much larger than the wavelength, the Mie Scattering Law dominates, which has no wavelength dependency [Hanslmeier (2023)].

Understanding the effect of light scattering explains how distinct passbands probe different objects and can determine the physical properties of observed clouds, such as particle size. Figure 1.3 demonstrates that shorter wavelengths identifies clouds of gas within the Milky Way, such as the observed dark cloud Bernard 68, whereas the cloud is not visible at longer wavelengths which allows the study of the stars behind the otherwise opaque cloud.

The interstellar medium is primarily (molecular) gas clouds and dust particles, meaning the scattering of astronomical observations disproportionately affects the shorter (bluer) wavelengths. As more blue light is scattered, observed spectra appear redder than the emitted spectra. This effect is known as reddening. Reddening is solely a change in magnitudes between passbands and does not affect the location of emission or absorption lines.

So far, we have considered light emitted by the target source scattered away from the observer. Yet, as the Milky Way contains numerous sources, light from
other sources will also be scattered towards the observer. In this situation, the scattering is described by the same processes introduced earlier but with the inverse relation to wavelength as the scattered light now reaches the observer.

Additionally, as the light passes through the dust and gaseous clouds, light can also be absorbed and reflected. Dust particles can be totally opaque in all or specific wavelengths to the observer, resulting in a proportion of the light being absorbed. Through the absorption, the clouds have a temperature resulting in the emission of light in the infrared spectrum from the cloud directly.

Thus, to accurately determine the total attenuation of an observed spectrum, one must consider the scattering of light emitted by the target source away from the observer, the scattering of light from other sources towards the observer, the absorption of light by particles along the line of sight, and the emission of light from particles along the line of sight. Understanding each of these effects is vital for correcting the observed spectra to match the originally emitted spectra from a source.

1.1.4 Notable Modern Survey Telescopes

Modern survey telescopes have mapped the sky in multiple bands with increasing detail and completeness. Such the success of recent telescopes such as the Optical Gravitational Lensing Experiment (OGLE), the Two Micron All-Sky Survey (2MASS) and the Wide-field Infrared Survey Explorer (WISE) amongst others, the focus is shifting from creating photometric and astrometric snapshots, but time series data of the whole sky. The flagship Vera C Rubin observatory, first light expected in 2024, is aiming to map the entire southern sky every three days over a ten-year window. This map represents an unparalleled map of the transient universe, with an expected 20 billion galaxies, 17 billion stars, and 6 million solar system bodies with 10 million nightly alerts. This represents a huge increase in data size and complexity to previous maps (Ivezić et al., 2019). The Vera C Rubin observatory is expected to revolutionise the field and has been a driving force behind the community interest and acceptance of machine learning techniques across almost all astronomical disciplines.

Of the current telescopes, this thesis largely analyses the data output from the Gaia telescope, with Chapter focusing on data from the Transiting Exoplanet Survey Satellite (TESS) telescope. Gaia was launched in 2013 with a primary
Figure 1.4 Visualisation of Gaia passbands in comparison to the visible spectrum. A transmissivity of 1 equates to total transparency and zero corresponds to totally opaque.

objective to observe one billion stars to further our understanding of the formation and evolution of our Galaxy. This is achieved by creating the most detailed 3D map of our Galaxy, containing both photometry and astrometry. The astrometry comprises accurate positions in the sky, ra and dec measurements, and distance through parallax measurements. Additionally, Gaia provides proper motion measurements, the velocity of a star relative to the expected trajectory from Galactic rotation, causing stars to gradually drift across the celestial sphere in ra and dec for nearly all observed sources. These measurements are taken through changes in parallax between differing epochs. Radial velocity, also known as line of sight proper motion, is measured by studying the red shift of spectra. As the radial velocity is derived from complex data products, Gaia DR3 only contains radial velocity measurements for 33 million sources (Gaia Collaboration et al., 2023). The photometry approach uses three broadband filters denoted G, BP, and RP, representing the combined result of filter and prism characteristics. The G-band extends slightly beyond the visible spectrum, whereas the RP and BP-bands represent approximately half the G-band range, covering the redder and bluer regions respectively. Figure 1.4 visualises the transmissivity curves of the Gaia broadband filters.
To date, *Gaia* has published three major data releases, with the latest in June 2022. This data release represents the photometry and astrometry of over two billion stars creating the most detailed map of the Milky Way with spectra of over 200 million objects. Further data releases are forecast to include additional data products including, time series data and spectra of nearly all objects. *Gaia DR3* provides a subset of these data products as an early release greatly expanding the potential for discovery from the *Gaia* observations. The data from *Gaia* is invaluable to the study of the local universe and is vital throughout all science chapters of this work. A key discovery from *Gaia* is the identification of a past galaxy merger with the Milky Way that occurred 10 billion years ago known as *Gaia Enceladus* (also named *Gaia Sausage*) which refers to this collection of stars found throughout the Milky Way that have significantly elongated trajectories and distinct chemical properties to the rest of the stars within the Galaxy, [Helmi et al. (2018)](https://doi.org/10.1038/s41586-018-0609-8). This result provides strong evidence of the role of major galaxy mergers in the formation of the Milky Way’s structure and the formation of a thick disc described in §1.3.

On the other hand, *TESS*, the successor to *Kepler*, is the leading telescope searching for exoplanet transits. The data products of *TESS* are time series data with a very short cadence providing an insight into the transient universe beyond the search for exoplanets. As of January 2023, *TESS* has detected nearly 300 confirmed exoplanets and over 6000 exoplanet candidates [NASA Exoplanet Archive](https://exoplanetarchive.ipac.caltech.edu). Nevertheless, the interest for this thesis is that, in many ways, the analysis performed on *TESS* can act as a precursor to the expected type of analysis required for the *Vera C. Rubin* observatory in the future.

While this work is focused largely on visible wavelengths, there are other upcoming telescopes of note. The largest being the *Square Kilometre Array* (SKA), a radio array expected to produce colossal volumes of data to the tune of 300 Petabytes per year. These telescopes, current and future, represent an enormous volume of data and this reveals a host of challenges in physical analysis and software complexity. For more information on SKA, see [Aafreen et al. (2022)](https://doi.org/10.1051/0004-6361/202140527); [Braun et al. (2019)](https://doi.org/10.1051/0004-6361/201834166).
1.2 Stellar Evolution

This work uses approaches that can be applied to many research areas of astronomy, but the focus is centred on stellar data. Therefore, to fully understand the wider impact and the results of this work it is imperative to understand the major stages of stellar evolution. The following section explains how clouds of gas collapse creating groups of stars and the general stages of evolution they take depending on their mass with the internal processes briefly covered. This overview is based on publications from Pols (2011) & Maeder (2009) where more details can be found.

1.2.1 The Hertzsprung-Russell Diagram

The Hertzsprung-Russell diagram is one of the most impactful analysis tools available to study stellar populations. Ejnar Hertzsprung in (Hertzsprung, 1911) and Henry Norris Russell in Russell (1914) independently discovered that the distribution of luminosities against spectral types is strongly correlated to several physical properties. Later version replaced spectral type with surface temperature or colour, which later gave rise to the Colour-Magnitude Diagram (CMD). By convention, the HR diagram, and subsequent CMD, places the hottest blue stars closest to the Y-axis by reversing the orientation of the X-axis. The advantage of a CMD is that observations from multiple passbands can be directly used without conversion to accurate temperatures. Figure 1.5 shows a CMD using Gaia photometry.

Due to the position of a star revealing its evolutionary stage, the diagram uncovers many properties of stellar evolution processes. Figure 1.5 indicates the dominating populations and their positions, the main sequence where the majority of stars lie represents stars undergoing core hydrogen burning, such as our sun. The second population is the Red Giant Branch (RGB), where evolved stars experience hydrogen shell burning. Finally, the last populous region is the stellar remnants of white dwarf (WD) stars, where the stellar remnants remain while they cool. Notably, the density within each population indicates the relative timescales a star occupies each population. Stellar evolutionary stages are explored in detail within § 1.2, explaining the origin of additional features observed within the HR diagram.
Figure 1.5 Example Colour-Absolute Magnitude Diagram using \( \sim \) 200,000 stars from Gaia photometry. The Main sequence stretches from the bottom right to the top left, with the Red Giant Branch and the red clump extending to the right of the Main sequence. Above the red clump, we see the AGB track spreading across to the upper right. Finally, a small population of white dwarfs populate the bottom left corner.
Beyond the evolutionary stage analysis, there are many uses for the HR diagram. As a star’s lifetime is a function of mass, the position of the main sequence turn-off point indicates the age of stellar clusters and co-moving groups. Additionally, distances can be estimated using specific features of known brightness. For example, Müller et al. (2018) calibrates the brightest tip of the RGB branch to calculate the distance of stellar populations.

Despite being discovered over 100 years ago, the HR diagram remains a powerful tool when studying stellar populations and the properties of any observed star. Even today, with detailed measurements from Gaia photometry, there is new astrophysics revealed by the HR diagram. Recently it has been used to investigate the internal structure of stars and the cooling processes of degenerate stars in Jao et al. (2018); Tremblay et al. (2019) respectively. The work presented throughout this thesis utilises the HR diagram to test, validate and demonstrate the performance of machine learning models and more.

1.2.2 Early Stellar Evolution

Clustered Star Formation

Galaxies with active star formation are full of gas, found in enormous clouds usually distributed throughout the galactic disc. A small perturbation causing a gravitational collapse of a giant molecular cloud is the first stage of star formation, (Lada & Lada, 2003). Events that cause the initial perturbation include interactions with large bodies such as other clouds, nearby supernovae or proximity to active star formation, to name a few.

For the perturbation to cause the cloud to collapse under its gravity, the mass must exceed what is known as the Jeans Mass, defined as:

\[ M_J \propto \left( \frac{T^3}{\rho} \right)^\frac{1}{2}, \]

where \( M_J \) is the Jean’s Mass, \( T \) is the temperature and \( \rho \) is the density (Maeder, 2009).

\( M_J \approx 10^3 - 10^4 M_\odot \) is a typically accepted range for Jean’s Mass, which does not match the observed mass distribution of stars in the Milky Way. In reality, as the cloud contracts, stability conditions are violated leading to cloud fragmentation,
where sections of the cloud collapse upon themselves. Each fragmented part with sufficient mass to collapse and ignite thermonuclear reactions forms an individual protostar. Due to the fragmentation process, a single molecular cloud results in a significant population of stars with similar ages and velocities. Such stellar populations are known as stellar clusters or co-moving groups. The nature of fragmentation determines the distribution of stellar masses, known as the initial mass function (IMF), within the population. However, due to the extreme forces and complexity, the fragmentation process is complex with countless contributing factors meaning accurate modelling is extremely difficult. Nevertheless, there is extensive research on this process within the Milky Way in works such as Lada & Lada (2003); Portegies Zwart et al. (2010); Krumholz et al. (2019). The interest for this thesis is the presence of clusters and co-moving groups within the Galaxy structures described in § 1.3.

Main Sequence

After a star has formed, they evolve towards the Main Sequence (MS). At this stage, the stars are stable with the gravitational collapse balanced by the energy released from the thermonuclear fusion of hydrogen in the stellar core. Due to the abundance of fuel and structural stability, stars spend approximately 90% of their lifetime on the main sequence.

The mechanism driving hydrogen burning differs depending on the internal temperature and, consequently, the mass. Low-mass stars fuse hydrogen through a process called the p-p chain:

\[
4 \, ^{1}\text{H} \longrightarrow ^{3}\text{He} + 2 \, ^{1}\text{H}^{+} + 2 \nu_{e},
\]

where four hydrogen atoms are fused to create helium. Whereas, the fusion of hydrogen stars with mass above 1.5\(M_{\odot}\) is through the more complicated process known as the CNO cycle:
Figure 1.6 Comparison of energy generation rates for the p-p chain and CNO-cycles as a function of temperature. The triple-α process highlights the higher initial temperature required to fuse helium nuclei.

As shown, the CNO cycle uses a carbon atom as a catalyst, resulting in four hydrogen nuclei fusing into a single helium nucleus, as is the case with the p-p chain. Naturally, there are branches of this process that use intermediate products to create alternative isotopes of oxygen and fluorine.

In all stellar models, we assume hydrostatic equilibrium, which means the net acceleration of a gas element within the star’s interior is zero. The assumption
leads to the equation of hydrostatic equilibrium:

\[ \frac{dP}{dm} = - \frac{Gm}{4\pi r^4}, \]

where \( P \) is the pressure, \( r \) is the radius from the stellar centre to a small gas element in the interior, \( m \) is the mass of the gas element, and \( G \) is the gravitational constant. A physical interpretation is that the internal pressure of a star decreases radially from the centre. A consequence of this equation is the virial theorem which quantifies the total internal energy of a star as:

\[ E_{\text{int}} = -\frac{1}{2}E_g, \]

where \( E_g \) is the gravitational potential energy and defined as:

\[ E_g = -\int_0^M \frac{Gm}{r} dm = -\frac{GM^2}{R}, \]

where \( M \) is the stellar mass, \( R \) is the stellar radius. From the ideal gas equation, we get the following relation:

\[ \bar{T} \propto M. \]

As Figure 1.6 demonstrates, the dominant energy generation process in stars is determined by the internal temperature. The temperature is also strongly correlated to the mass. Therefore, there is a relation between the mass, \( M \), (or temperature, \( T \)) and luminosity, \( L \), (energy generated) approximated by:

\[ L \propto M^{2.35}, \]

which explains the physical origin of the main sequence seen on the Hertzsprung-Russell Diagram (Pols [2011]). The hydrogen burning phase is very stable and builds up a core of inert helium throughout this phase which causes the star to begin the next stage of evolution.
1.2.3 Post Main Sequence Stellar Evolution

Red Giant Branch

During the main sequence phase, the stellar core is depleted of hydrogen and creates a helium core. However, the central temperature is too low to ignite helium, leaving the core inert. Rather than the core burning, the hydrogen shell surrounding the core undergoes fusion in a phase known as hydrogen shell burning. Shell burning increases the mass of the inert helium core until it reaches the Schönberg-Chandrasekhar Limit, where the unsupported isothermal core contracts and heats (Schönberg & Chandrasekhar, 1942). Through a process called the mirror principle, the core collapse causes a rapid expansion of the outer shells to maintain the temperature in the still-fusing hydrogen shell. The rapid expansion decreases the surface temperature causing the red appearance and, over time, the luminosity increases. Due to their large size and red appearance, this type of star is called a Red Giant. Red Giant stars are identifiable on a HR diagram by a distinctive branch from the main sequence stretching towards the top right of the diagram, known as the Red Giant Branch (RGB).

As the envelope has dramatically expanded, the density and internal opacity are very low. The internal conditions support a deep convective layer, which encroaches into regions that have partially undergone fusion, bringing heavy metals from the CNO cycle (e.g. carbon and oxygen) to the surface. This process is called the First Dredge-Up and allows direct observation of the change of composition within a star. Overall, a star spends approximately 10% of the main-sequence lifetime on the RGB, accounting for the majority of the post-main sequence lifetime.

Meanwhile, the helium core continues to collapse and heat until it reaches $10^8 K$ igniting the helium core. In the intermediate-mass stars ($2 \lesssim M \lesssim 8 \text{M}_\odot$), the core remains non-degenerate halting core contraction and shell expansion, indicating the maximum luminosity reached during this phase. For low-mass stars ($M \lesssim 2 \text{M}_\odot$), the core is degenerate at this stage, hence when the core ignites, the star undergoes thermonuclear runaway causing a substantial increase of energy released known as the helium flash.
Helium Core Ignition

Currently, the star has two internal regions, the large cool convective hydrogen envelope and the hot helium core undergoing fusion. The mechanism for hydrogen burning is called the triple-α process occurring in the following steps:

\[ ^4\text{He} + ^4\text{He} \rightleftharpoons ^8\text{Be} + \gamma. \]

During the Beryllium’s lifetime (\(\sim 10^{-16}\)s) it may capture a third alpha particle leading to;

\[ ^8\text{Be} + ^4\text{He} \rightleftharpoons ^{12}\text{C}^*, \]

which is a reversible process but occasionally the Carbon nucleus can enter a lower energy state:

\[ ^{12}\text{C}^* \longrightarrow ^{12}\text{C} + \gamma. \]

This reaction releases roughly ten times the energy per nucleon than hydrogen burning, with an even stronger temperature dependence than the CNO cycle. In addition, there are also accompanying alpha processes occurring:

\[ ^{12}\text{C} + ^4\text{He} \longrightarrow ^{16}\text{O} \]
\[ ^{16}\text{O} + ^4\text{He} \longrightarrow ^{20}\text{Ne} \]

The energy released within the core during ignition causes an expansion, leading to the stellar envelope contracting, reversing conditions that drove the star onto the RGB. As the envelope contracts, the density, hence opacity, increases, causing the surface temperature to increase. The temperature change is dramatic, evolving the star along a feature of the HR diagram known as the Horizontal Branch. This evolutionary stage passes through a region of particular interest known as the Instability Strip, which is explored further in the following section.

During this stage, the helium core is depleted through the triple-α process, resulting in an increased abundance of carbon. As the helium abundance reduces, the nuclear rate decreases (\(\epsilon \propto Y^3\)), eventually causing the core to contract, expanding the envelope again through the mirror principle. However, the abundance of carbon drives the first of the accompanying alpha processes to become dominant, which creates oxygen through:

\[ ^{12}\text{C} + ^4\text{He} \longrightarrow ^{16}\text{O} + \gamma. \]
Evidently, this increases the abundance of oxygen, with the final oxygen/carbon increasing with the stellar mass. This process causes the envelope to contract, creating a loop in the HR diagram, seen later in Figure 1.7 referred to as the Blue Loop.

The Instability Strip

During helium burning, the star experiences periods of instability, leading to radial pulsations. These pulsations occur when the star evolves to a region in the HR diagram that stretches from the main sequence to higher luminosity, parallel to the giant branch. The Horizontal Branch and sufficiently large Blue Loops both pass through this region. This region is very diverse, with one of the brightest and most prominent stellar types being Cepheid Variable stars. Cepheid stars show a characteristic pulsation pattern and populate the intersection between the Instability Strip and the Blue Loop. Cepheid variables have periods directly linked to their dynamical timescale, hence luminosity. Thus, Cepheids follow a well-defined Period-Luminosity function allowing direct measurement of distance. Another notable stellar class is RR-Lyrae stars. They are Horizontal Branch stars and populate a region closer to the main sequence than Cepheids. RR-Lyrae also shows the characteristic pulsation pattern in brightness, albeit with a differing waveform and periodicity to Cepheids. Other stellar classes found in the Instability strip include but are not limited to; δ-Scuti, RV Tauri, and the rapidly oscillating Ap stars (Breger 2000; Gerasimović 1929; Kurtz 1982). The mechanism for thermal pulsations differs slightly between stellar classes. Nonetheless, at a fundamental scale, the physics is similar. Changes in density and opacity during periods of expansion and contraction create a repeating cycle known as the κ-mechanism. As a star contracts, the density increases, causing an increase in opacity, raising the internal heat and pressure as photons cannot as easily escape. The increased pressure causes an expansion, leading to a decrease in opacity, allowing an increase in energy escaping the stellar interior. The release of energy causes the interior to cool and cause a contraction, restarting the cycle (De Boer & Seggewiss 2008). In addition to the κ-mechanism, the ε-mechanism is also present. An increase in temperature increases the energy generated through fusion, causing additional heating during contractions. This process occurs at the active core, which at these timescales cannot cause significant oscillations at the surface during this evolutionary phase.
1.2.4 Low & Intermediate Mass Late-Stage Stellar Evolution

**Early-Asymptotic Giant Branch**

The final stages of stellar evolution are known as the *Asymptotic Giant Branch* (AGB), beginning when helium burning ceases in the core. As with the helium core, the lack of fusion in the carbon-oxygen (CO) core causes a contraction. The contraction causes the envelope to expand due to the re-ignition of a hydrogen shell. Until the helium shell below this fusion layer ignites, creating two layers of burning around an inert core. Yet, the helium shell causes the outer layers to contract again, ultimately extinguishing the hydrogen layer. This structure is now similar to the RGB, with a large radius and cool surface temperature, albeit with different chemical composition. In low-mass stars, the hydrogen burning continues at a restricted rate, which stops the convective surface from penetrating deep layers. If hydrogen burning is extinguished a second dredge-up event occurs, transporting heavier elements to the stellar surface. This dredge-up mixes a large volume of mass, reducing the total mass contributed to the core during the helium shell burning. Similar to the RGB phase, the CO core becomes degenerate. At this mass range \( M \lesssim 8M_\odot \) the core cannot reach the critical mass for carbon ignition hence a *Carbon-Flash* does not occur. Instead, the star evolves towards the *Thermally-Pulsating AGB* (TP-AGB) phase.

**Thermally-Pulsing Asymptotic Giant Branch**

In the TP-AGB phase, we have the CO core with a helium shell burning surrounded by a formerly/limited burning hydrogen shell. This structure extends \( \sim 1\% \) of the total radius. The remaining interior is an extended, predominantly hydrogen, convective envelope. As the helium shell burns, it approaches the extended hydrogen layer limiting the fuel available and causing a drop in the fusion rate. The reduction in energy production contracts the layers above, reigniting the hydrogen shell burning and extinguishing the helium shell. But hydrogen burning increases the abundance of helium available, eventually reigniting the helium layer and ceasing hydrogen burning. The flash of energy from the re-ignition of helium briefly causes convection between the hydrogen and helium layers, bringing further helium and heavier elements (from the CNO cycle) to the surface known as the *Third Dredge-Up*. The convection reduces the accretion of mass on the core reducing the final core mass.
While white dwarf progenitors can not exceed approximately $8M_\odot$, however, the white dwarfs themselves cannot exceed the Chandrasekhar Mass, $\sim 1.4M_\odot$, above which the internal pressures exceed those supported by electron degeneracy pressure. Therefore, there must be significant mass loss during this phase, with the thermal pulsations repeating with the two shells burning at different rates. Each thermal pulsation drives mass loss from the outer hydrogen-rich envelope from the top, while continued shell burning depletes the interior edge of the envelope. When the envelope is nearly exhausted, around $0.01M_\odot$, it shrinks and heats dramatically, marking the beginning of the Post-AGB phase.

**White Dwarf Formation**

At the start of the post-AGB phase, the surface has been super-heated, ionising the ejected mass from the envelope. The ejected material, no longer gravitationally bound, drifts from the core, becoming transparent, allowing the direct observation of the bright core surrounded by ejected gas layers. This structure is the formation of a Planetary Nebula, a short-lived phase until the remaining hydrogen burning ceases, resulting in a rapid drop in luminosity. The core begins to cool with a final thermal pulse often occurring, known as
the *Late Thermal Pulse* (Miller Bertolami & Althaus, 2007), until the core has transitioned into an inert white dwarf star.

The exact processes that occur are extremely volatile and sensitive to small perturbations. Factors such as metalicity, spin and magnetism affect the process dramatically in ways that are not fully understood. Due to the dependence on many factors, substantial mass loss and cataclysmic nature of the evolution, understanding the nature of white dwarf progenitors is particularly challenging.

### 1.2.5 High Mass Late-Stage Stellar Evolution

High-mass stars ($M \gtrsim 8 M_\odot$) evolve much faster than low-mass stars, with lifetimes of millions rather than billions of years. Therefore, there are significantly fewer high-mass stars within the Galaxy, an effect compounded by the power law of our Galaxy’s IMF (Chabrier, 2003). Considering the science contained within this thesis does not heavily feature high-mass stars, the following section is included for completeness and acts as a brief overview with many details omitted.

The determining factor of late-stage evolution is the stellar mass. In §1.2.3 I explored the formation of a carbon-oxygen core. High-mass stars reach the critical core mass of $1.06 M_\odot$ for the ignition of carbon burning within the core. For more massive stars, this process continues igniting increasingly heavy elements (neon, magnesium, silicon and sulphur) until the formation of an iron core. At this stage, the core forms layers of increasingly heavy elements closer to the centre, with the iron centre remaining inert and increasing in mass.

The inert iron core gravitational pressure is opposed by electron degeneracy pressure, meaning when the core exceeds the Chandrasekhar mass, the core collapses in milliseconds in an enormous release of energy. The star has gone supernova, releasing energy on the scale of galaxies rather than a star. During the collapse, the process of *neutronization* occurs, where protons combine with electrons resulting in a neutron-rich core. Now, the neutron-rich core remains degenerate, with the gravitational pressure opposed by neutron degeneracy pressure. The degenerate neutron mass is known as a *Neutron* star, one of the densest objects in the universe. Theories predict that for sufficiently massive stars, the gravitational pressure can exceed neutron degeneracy pressure resulting in a further collapse. However, this collapse produces no detectable signature, as the resulting body is a black hole. The transition into black holes signals the
breakdown of modern physics, with few predictions about physics beyond the event horizon.

Supernova explosions are some of the most catastrophic and energetic events in the Universe, and come in three broad categories:

- **Type Ia**: The result of a thermonuclear explosion caused by a WD exceeding the Chandrasekhar mass through the accretion of mass from a close companion star.
- **Type Ib**: Caused by core collapse in massive stars that have lost their hydrogen envelope.
- **Type II**: Caused by core collapse in massive stars with envelopes rich in hydrogen.

The presence of WD’s is known for Type Ia supernovae as they occur away from star-forming regions, whereas Type Ib and Type II occur within star-forming regions. The continued star formation indicates a requirement for short-lived high-mass stars.

Late-stage high-mass stellar evolution is over extremely short lifetimes meaning changes within the core are not observable as the envelope does not reflect the internal instability. The lack of observational data means this is a very active area of computational astrophysics. This summary provides an overview, ignoring the effects of mass loss, neutrinos and other relativist physics.

The information covered in this section provides an overview of stellar evolution details required to understand the astrophysical significance of this thesis. For an in-depth overview, the literature contains numerous books and articles, notably Pols (2011) & De Boer & Seggewiss (2008), both of which are used extensively throughout the above discussion.

### 1.2.6 Isochrone Models

Stellar evolution, overall, is well constrained with advanced hydro-dynamical simulations modelling the evolutionary tracks of a range of stellar masses. These models test the theoretical physics described above while demonstrating properties such as mass loss along these tracks. Due to the long timescales,
excluding rapid cataclysmic events, we will never observe a star evolve directly. Consequently, the analysis of stellar populations across the HR-Diagram provides snapshots revealing details of the total evolution tracks.

Observing populations of stars allows the study of stellar evolution by providing snapshots, revealing the global evolutionary patterns. These snapshots constrain evolutionary models to verify the effects of physical properties, such as metallicity and lifetimes. Today, the accuracy of these models allows many properties to be constrained by only fitting models to observed clusters. This ability reduces the reliance on the availability of high-quality spectra. Figure 1.8 visualises the evolutionary tracks for a range of stellar masses, where many of the evolutionary stages described earlier are present.

Despite the sophisticated models available, there are limitations. The most prominent occurs at the final stages of evolution. This transition is vital to understand as it determines the ultimate fate of stars. Evolving from the AGB towards the white dwarf population results in up to 85% of the total mass to be lost over in a short time frame, see § 1.2.4. The rapid pace of such evolution is challenging to model accurately due to several internal mechanisms and depends
on accurate observational constraints which are not yet fully available.

1.3 Milky Way Composition

In the 1920s, Edwin Hubble devised a process for classifying galaxies based on observable structure. He devised three major categories: elliptical, spiral, and barred type with an additional category for irregular galaxies that are not well described by the three definitions [Hubble 1926]. Elliptical galaxies have very little structure, are largely spherical and contain older stars with little star formation. Spiral galaxies have a central bulge which contains older stars (similar to an elliptical galaxy) with a large disc extending beyond. Within the disc, extended spiral arms are visible, with younger stars and higher star formation rates than the central bulge. Lenticular galaxies are the intermediate between elliptical and spiral galaxies. The structure is similar to a Spiral galaxy with a central bulge and extended disc, yet the disc does not contain distinctive spiral
arms. Finally, irregular galaxies account for those which do not fit into those described above and hence do not have a central bulge nor spiral structure. These are usually smaller or disturbed by their environment (e.g. colliding galaxies). Famous examples of irregular galaxies are the large and small Magellanic clouds. In practice, each class can be divided into sub-classes defined by the presence or strength of specific features, such as the eccentricity of elliptical galaxies. Figure 1.9 shows examples of these galaxy classifications.

The size of a galaxy is difficult to define due to the lack of a precise edge. Thus, astronomers often use brightness or mass profiles. A common method for disc galaxies is the Isophotal Diameter, which corresponds to the major axis diameter of a galaxy where the brightness level is at 25mag arcsec$^{-2}$ in the B-band. Chamba (2020) discusses the size of galaxies in detail. As our Galaxy, the Milky Way, is a spiral galaxy, we can estimate the size using the isophotal diameter. Goodwin et al. (1998) calculates a 25 B-mag arcsec$^{-2}$ diameter as, $D_{25} = 26.8 \pm 1.1$kpcs. Naturally, the stellar populations are not consistent across the Galaxy. There are four distinct stellar populations of note, the first being the older stars within the central bulge, although there is an ongoing debate over the discovery of a population of younger stars that appear to occur from a single star formation event Mauerhan et al. (2010). A second population of stars are known as the Halo stars. These stars do not have the same rotational velocity as the disc and are distributed spherically around the Galactic centre. Within
the halo are globular clusters, which are bound groups of thousands of halo stars. These clusters have been the focus of numerous stellar population studies and are extremely useful for determining the Milky Way’s formation history  [Helmi (2008)]. In total, halo stars account for \( \sim 1\% \) of the total Galaxy’s stellar mass.

The two remaining populations are within the galactic disc. These populations are known as the thick and thin disc. The thick disc has a scale height of 1kpc compared to the thin disc’s approximately 300 parsec scale height  [Gilmore & Reid, 1983; Hayden et al., 2017]. We define the scale height as the distance for which the density decreases by \( e^{-1} \)  [Carroll & Ostlie, 2006]. Two differences between these populations include average ages and metallicity. While younger and more metal-rich than the halo stars, the thick disc population is much older and has a lower metalicity than the thin disc. There also exists a difference in the velocity dispersion between the two populations, highlighted in Figure 16 of Rowell & Hambly (2011), included as Figure 1.10. There remains much debate about the distinction between the two populations and the process that formed the thick disc  [Bensby & Feltzing (2010)]. A leading hypothesis is that the thick disc formation requires a merger event with a Dwarf galaxy  [Villalobos & Helmi, 2009]. Yet, simulations by Schönrich & Binney (2009) challenge this,
where dispersion alone forms a thick disc over time. This work focuses on the thin disc population due to the increased star formation and the reduced velocity dispersion. The Sun is located approximately halfway along one the Orion spiral arm, roughly 8kpcs from the Galactic centre ([Camarillo et al., 2018]). Figure 1.11 summarises the distribution of stellar populations found within the Milky Way.

Stellar mass only represents a fraction of the total mass within the Galaxy. The Milky Way also contains significant volumes of dust and gas obscuring our view across the disc. Furthermore, the majority of a galaxy’s mass lies in the dark matter halo, which is only detectable through velocity measurements such as rotation curves. Therefore, the Milky Way has many unknown features and structures that are unobserved due to our position within the Galaxy, although many of these are beyond the scope of this thesis due to their nature or distance.

1.4 Computational Challenges

1.4.1 The Data Volume Challenge

The increasing data volumes from these telescopes bring incredible opportunities for discovery, however, they represent a complete overhaul in the analysis processes. To truly understand the extent of the challenges faced over the next decade, we must compare the past and current datasets against those forecast from future survey telescopes. Perhaps the most famous survey is the SDSS galaxy database, collecting spectra and imaging of several million galaxies across various instruments and projects over 10,000 degrees of sky ([Bundy et al., 2015]). Despite the relatively modest number of galaxies observed, the information per galaxy is substantial, creating an ideal dataset for citizen science. Not only is the scale of the challenge feasible, but the type of images are aesthetically pleasing, capturing public interest. Evidently, a dataset of colourful merging galaxies is far more engaging than endless graphs of abstract time series data. The Galaxy Zoo project has been a big success using citizen science to classify galaxies, collecting over five million classifications. Clearly, the success of citizen science projects depends on the number of sources and the nature of the data products rather than the total data volume. Databases with billions of objects (e.g. Gaia DR3) are unsuitable to attempt to utilise the public due to the required levels of engagement. By building ever larger surveys, citizen science is not suitable to
solve the upcoming challenges.

With the advancement of incoming data over the coming decade, it is tempting to assume that future technologies will solve the issues. Yet, the volume of data these surveys are forecast to produce is way beyond the best estimates of technological advancement. Expectations predict the Vera C. Rubin observatory to release 15 petabytes of compressed catalogue data over ten years, containing observational data of an estimated 40 billion objects \cite{lvezić2019}. To download and store just 1 petabyte of data takes 100 days to download on a dedicated 1Gbps connection and costs in the region of £50,000 in hardware using the latest storage solutions. While this is achievable at many facilities, this has disregarded the hundreds of petabytes from the image dataset, and the fact that difficulty arises when attempting to perform meaningful data analysis. The current and next generation of telescopes are driving astronomy to the cutting edge of advanced analysis and storage solutions.

The increases in data volume have already driven the shift to distributed and cloud computing. No longer can personal computing handle this data and nearly all research institutes have now developed high-powered compute clusters. Despite the outstanding resources available at these institutes, reliance on local storage is quickly becoming unfeasible. When dealing with 10’s of Petabytes or more, local storage is expensive and consumes significant amounts of energy. A shift towards open online services, which facilitate the analysis on a mission level, offers a more ecologically friendly, financially, and scientifically beneficial approach. For instance, the Gaia DMP project, \cite{GaiaCollaboration2022}, is bringing a cloud compute service to the community in preparation for future data releases. The project requires a shift in mentality around data handling, data analysis and funding allocations. Consequently, a universal computing system naturally forces the community to establish standards for supported languages, packages and even interfaces.

Fortunately, the language of choice in astronomy is Python, a convention that has naturally established itself through many specialised analysis tools, e.g. Astropy \cite{AstropyCollaboration2013}; Price-Whelan et al. \cite{Price-Whelan2018}, becoming available. The challenge posed by a single language is that the most commonly used packages and software often cannot handle big data efficiently. Many packages, such as Pandas, are non-distributed, which require extreme memory availability and computational architectures while offering unfavourable performance \cite{Team2020}. This limitation is a warning that commonly used tools
are not suitable for use on big data. Big data requires distributed programmes to maintain reasonable memory requirements and runtimes. It is unreasonable to expect a shift from Python anytime soon. Thus, there is a need to have software with a Python API, which is also capable of handling big data containing billions of columns.

1.4.2 Computing Architecture

While the development towards big data is new to astronomy, it is not in the tech sector and drives the profits of many of the biggest tech companies. Consequently, tools needed to handle the incoming databases exist. Yet, as these are private companies prioritising profits comes before the open-source nature of scientific research. Nonetheless, some publications provide insight into their techniques; back in 2008, Google published a paper outlining the methods they use to handle the scale of their data processing requirements. An example is the release notes for MapReduce, which reveals that during the month of November in 2007, Google handles over 400 petabytes of data, a scale Astronomy is only just approaching 15 years later (Dean & Ghemawat 2008). MapReduce can only handle this volume of data by distributing the workload across several concurrent workers. Even with technological advancements, any package attempting to handle comparable volumes of data must also implement a distributed approach. The issue within astronomy is that many of the most common and countless specialised software packages (such as Numpy, Pandas, and Matplotlib) must be updated.

Distributed Computing

Distributed computing has many subtleties, most of which are beyond the consideration of a system user. Yet, in the interest of completeness, at a fundamental level, a distributed system is a series of independent computers (nodes) connected to a global network. This network can run applications across multiple nodes concurrently with shared memory or independently on a single node as required. The cluster of smaller capacity nodes behaves flexibly, either as a large capacity distributed system or able to run independent tasks that each use a fraction of the total resources. Compared to other industries, as the number of professional astronomers simultaneously requiring a system is minimal, research institutes or specific databases can provide a single distributed system.
Figure 1.12  *An example of a distributed architecture showing shared memory across multiple nodes. The system shares resources across multiple nodes or divides resources to run independent applications concurrently.* Credit: Figure 1 of [van Steen & Tanenbaum (2016)](Computing 98, 967–1009, under CC BY 4.0).

able to handle the numbers of users provided the computational demands remain reasonable [van Steen & Tanenbaum (2016)](Computing 98, 967–1009, under CC BY 4.0).

Despite increased complexity and management requirements, high-end hardware does not increase linearly in price with performance. Instead, linking cheaper components into a network, such as in [Figure 1.12](Computing 98, 967–1009, under CC BY 4.0), reduces overall cost and increases utilisation. Furthermore, the system has failure resistance for both compute power and storage. Ultimately, distributed systems are now commonplace throughout research, yet the utilisation is focused on multiple users submitting single-threaded workflows. Nonetheless, the ability to scale resources offers the most meaningful advantage. Commercial systems, such as Amazon Web Services (AWS) and Azure, are now widespread as they only require a dedicated head node that scales the worker resources with demand, reducing overall energy and financial costs.

**Distributed Software**

The issue with Python being the standard language of choice within astronomy, with much of the specialised software either written in or using a Python API, is that, fundamentally, the language is built for single-threaded code. Distributed software lowers the peak computational requirements and can dramatically reduce run times. The package *multiprocessing* begins to distribute standard Python code, although this is an ad hoc solution that can be difficult to implement.
Whereas many distributed software packages are for a specific use case, e.g., MapReduce, that can lead to legacy issues when updates cease.

Apache Spark is a unified analytics engine providing an API designed to operate in a distributed manner [Zaharia et al., 2016] providing a solution to many of the issues highlighted above. Many billion-dollar companies, e.g. Netflix, already use a Spark deployment to run their businesses. An advantage is that Spark operates through a familiar API, even acting as a built-in extension of an existing Jupyter or Zeppelin notebook. Spark supports code written in Python, R, Scala, SQL and Java, complete with distributed plotting and machine learning libraries. The generality of the software ensures the foundations are universally available for big data analysis. However, the current Python interface, PySpark, requires a significant learning curve for those familiar with more common non-distributed data analysis tools in science, such as Pandas or Scikit-learn [Team, 2020; Pedregosa et al., 2011]. Future versions of PySpark promise compatibility with the Pandas syntax reducing the initial learning curve. Alternatively, individual packages are gaining distributed projects. Koalas attempts to offer a distributed version of Pandas [Liu & Hunter, 2019]. Yet, projects such as these require a substantial community effort with little to no dedicated funding. A notable alternative is Dask [Dask Development Team, 2016], which uses a familiar API for Pandas users but is less widespread.

On-Premise Vs Cloud Systems

As explored in the previous sections, it is common practice to use distributed systems for data analysis, with virtually all research institutes hosting on-premise systems. On-premise refers to the hardware being physically maintained in-house and requires traditional hardware connections. As expected, this is a fixed system with a significant fraction of costs incurred through purchasing hardware and software licenses during set-up. The advantages are that all resources are available at all times, with the ability to add functionality as required. With typically low running and maintenance costs, the approach is often an economical solution. The primary disadvantage is the fixed capacity, with a maximum capacity depending on the available hardware. Additional funding is necessary to purchase upgraded hardware to increase capacity while scaling down during periods of low demand is impossible.

Alternatively, commercially available cloud systems are hosted by institute
collaborations or private companies. Their advantages include economies of scale as resources are cheaper at scale and can adjust available resources to meet demand. The ability to match resources to demand is the greatest advantage of this approach, offering significantly reduced costs for individual institutes and continuing to improve with large numbers of users across multiple fields and institutes as the fluctuations in total demand are minimised. An individual project within a cloud system requires a small head node to remain online, with additional resources requested when demand increases and then will release those resources to other projects when demand subsides \cite{Fisher2018}. Due to the increased latency between worker nodes, even for advanced installations, such as Microsoft Azure, the cloud performance is slightly reduced compared to a similar on-premise system \cite{Györodi2019}. Nonetheless, a project will experience less downtime in a cloud system as resources seamlessly transfer across hardware in multiple sites.

Evidently, there are benefits to both systems. Ultimately, the design of future systems will depend on astronomical requirements and policy decisions. Is it reasonable for each institute to maintain and manage an on-site cluster, or equally, will a shared resource gain as much funding from institutes that will lose control of the specific capabilities? I believe that the data volumes will eventually drive the industry towards shared resources as it is unsustainable in terms of both bandwidth and storage capacities. Regrettably, this may make science less accessible due to consumption fees, effectively placing open-source data behind a paywall.

1.5 Machine Learning

The term machine learning covers a wide range of algorithms and techniques but generally is software that analyses or predicts a feature of data using results that are not directly programmed. Fundamentally, the computer is “learning” the relationship through statistical trends and patterns in the data rather than any physical interpretations. These approaches are most effective when dealing with large datasets with complex data points. For example, a database of stellar spectra will provide more accurate stellar classifications than a similarly sized database of photometric data. Nevertheless, machine learning is an increasingly used tool for challenges including but not limited to: classification, regression analysis, identification of structures (clustering), and object detection.
1.5.1 Supervised vs Unsupervised Learning

Most machine learning algorithms are characterised into two major classes, supervised and unsupervised learning. The key difference is that supervised learning relies on access to an accurately labelled training set of data, meaning all results reflect the information provided in the input labels. On the other hand, unsupervised learning identifies patterns in data, requiring the user to analyse the results to discover the physical meaning of the results. Each approach has benefits and drawbacks depending on the task at hand, and some algorithms lie between these classes, known as semi-supervised learning, which we will not cover in detail here. The following sections explore each approach in greater detail.

Supervised learning requires input labels (known as training data) to train the model on what features each class possesses. This trained model is then used to classify unseen data by matching those features to one of the classes learned from the input labels. As the training data represents a ground truth, the classification of unseen data is limited to the insights provided in the training dataset. After the training phase stage, the model classifies unlabelled datasets quickly and efficiently. The advantages include comprehensible outputs based on expert domain knowledge input during the training phase. Additionally, the known classifications within the training set allow the analysis of the model’s performance by retaining a subset of the labelled data to quantify performance, a process explored further in §1.5.5. The disadvantages of supervised learning are that the results are limited to information within the training data. This reliance on the training data often leads to large training sets that must be representative of the dataset to be analysed. Common approaches to select a training set are explored later in §1.5.3. Algorithms that use supervised learning include; support vector machines (Cortes & Vapnik 2009; Noble 2007), random forests (Breiman 2001) and k-nearest neighbour (Guo et al. 2003).

On the other hand, unsupervised learning evaluates the data with no prerequisite knowledge. This approach allows vastly more complex analysis and notably has the ability to discover unknown features within datasets with little intervention due to the lack of training data. The disadvantages of unsupervised learning are that the results are unpredictable and computationally heavy for large datasets. On top of this, the results can be challenging to interpret as they are purely statistical, requiring careful consideration to understand the physical interpretations. Nonetheless, this type of learning is particularly useful for
clustering data or outlier detection when the defining features are unknown.

**Active Learning**

As discussed above, both types of machine learning have drawbacks. Supervised learning requires a large training set representing the entire dataset, and unsupervised learning returns abstract and statistical results. Furthermore, an accurate training set is a rare commodity and creating one is time-consuming and requires specific expertise. Active learning is an approach to streamline the process by targeting the region in feature space with increased uncertainty, such as the boundary between multiple classifications.

Active learning creates a symbiotic relationship between the user and the algorithm, each contributing to the overall performance. Naturally, this requires a degree of domain knowledge and human-readable features. Commonly this approach is used with convolutional neural networks (CNNs) for computer vision or speech recognition, although the ML model and user do not necessarily have to view the same subset of features. The major advantage of active learning is the reduction in labelled training data which can dramatically improve results where unlabelled data is also limited.

While active learning offers a significant reduction in development times and training requirements, there are several drawbacks. The training approach is iterative, meaning there is the possibility of classification drift over time, exacerbated by the user directly influencing category boundaries. Thus, for active learning to be effective, the classifications must be well-defined and be illustrated in a user-friendly version (e.g. an image).

### 1.5.2 Common Machine Learning Practices

Machine learning is an evolving field with new approaches and techniques utilising the ever-growing capabilities of computer hardware. Machine learning is an area where academia does not lead the way, rather, companies such as Google, Apple, and Microsoft, along with other corporate giants, dominate the field with their capabilities. While these companies are profit-driven, the disparity in development between industry and academia has provided an opportunity as numerous advanced tools are publicly available. Yet, the scalable nature of these
Several common software packages are used for machine learning with varying capabilities and support aiming to simplify the application of machine learning models. The first is TensorFlow (Abadi et al., 2015) developed by a team within Google that offers a wide range of language support as well as integration with Keras (Chollet et al., 2015), which is a Python API as the most commonly used programming language. A second package is Scikit Learn (Pedregosa et al., 2011), also originating from Google, but now is developed through community support. This Python-based approach is built on standard packages (e.g. Scipy & Numpy), creating an easy-to-use interface. However, the familiarity and ease of the API can often hide several deep learning subtleties. A similar example is PyTorch (Paszke et al., 2019), which, as the name suggests, is a Python-based package (with C++ support) that even offers full Scikit-Learn interaction through the Skorch wrapper (Tietz et al., 2017). An astronomy-specific package is AstroML, which incorporates Astropy designed to act as a repository of tested machine learning code for use on astronomical data (Vanderplas et al., 2012). Finally, all of the mentioned packages are built around Pandas and Numpy arrays, which are intrinsically non-distributed. Thus, Apache Spark dataframes offer a distributed alternative, thus requiring a distributed machine learning package. Despite all algorithms not being possible to distribute, PySpark ML (Meng et al., 2016) offers a wide range of algorithms, including neural networks and random forests. PySpark ML is a growing package with scalability becoming vital alongside the growth of PySpark usage in industry and the convergence of PySpark ML with PySpark MLlib with version 3.0.0.

Despite several implementations, applications and data types, all machine learning pipelines follow the same main stages. The first stage is to collect data for analysis and evaluate the composition and intricacies of the data. This evaluation includes the ratios of each class or potential features for extraction and any bias or imbalances present in the data. Thus, a suitable training set can be constructed that accurately reflects the data. The exact requirements for training data are specific to the algorithm chosen and the specific application. Yet, all implementations rely on the training data being representative of the target data set, §1.5.3. After selecting and extracting features from the training data, it is common to normalise these to ensure a consistent range across all features for improved consistency during the training phase, §1.5.3. Finally, the model is trained using the training data and an independent subset, known as validation.
Figure 1.13  Flowchart of a general machine learning workflow.
data, until a selected evaluation metric is optimised, see §1.5.5. Figure 1.13 summarises the process from data to training a model as a flowchart. The following subsections explore the details and the most frequently used techniques (along with those found later in the science chapters of this thesis) for each stage. In these sections, the descriptions will address steps for classification tasks for clarity.

### 1.5.3 Training Data Creation

All machine learning models require a set of training data. In practice, the training data is split into a training and validation set with a ratio of 80/20 considered standard. The algorithm is trained on the training subset while withholding the validation set to evaluate the model performance. The split data ensures the model’s performance is referenced to “unseen data”, drastically reducing overfitting. The training phase is critical to a successful predictive model, consequently, any errors, misrepresentations, or biases within the training datasets will be present in the final model, meaning the selection of representative training data is crucial.

A representative dataset means all the intricacies of the target population are reflected. Consider a classification problem determining images of apples, oranges and pears. If the training data only contains apples and oranges, the performance will be excellent for those classes but entirely useless in determining pears. Furthermore, if the training data only contains images of green apples, the final analysis will be unlikely to perform accurately on images of red apples. Whilst this is a basic example, it represents the range of considerations when selecting and evaluating training data. Even for this example, it is imperative to ensure the images contain a range of backgrounds, orientations, sizes, and even lighting, as all these factors can introduce erroneous correlations. The model does not interpret the data. It simply identifies any pattern that optimises the results.

Initially, all known data must be identified, with many sampling methods possible to generate the final training data. The simplest being a random sample of available data. Random sampling can be effective when sufficient data is available that is representative of all classes as the nuances are more likely to be represented, provided the available data is from a suitable source. The definition of “sufficiently large” depends on the algorithm used and the complexity, similarity and number of classes. A model identifying cats and dogs
needs less training data than one to identify specific breeds of cats and dogs, as this represents similar and more complex classes.

*Stratified sampling* is effective when the collected training data does not represent the target population. The data is partitioned into groups, e.g. gender or tax bracket, and randomly sampled independently by each group. This approach allows the training data to better represent the target population in terms of this key identified metric and effectively increases the selection of underrepresented groups. For example, if age is a key factor (but not necessarily a selected feature) for a study, a sample of students would not match the population on the high street. By using stratified sampling to equate the age distributions, the sampled populations are also more representative in other metrics, such as net worth. The example above reduces selection bias and potentially increases the volume of usable training data. The downsides to stratified sampling are that it relies on a large populous and is limited by the least proportionally represented group.

Sometimes the described methods are not possible, or the available data is incomplete. This scenario is when synthetic data creation takes charge. An option to increase the training date is to manipulate the available data. Data manipulation is best understood in terms of image data. Simply rotating, stretching, subsampling or colour adjusting a single image creates new data points with sometimes vastly different characteristics. By switching colour channels, the green apples example above would be better suited to include red apples as the same class. Data manipulation is most effective in generalising models without increasing the volume of required labels, e.g. a rotated image should still be accurately classified. In extreme cases, theoretical models can create fully synthetic datasets. Such an approach allows scientific understanding to direct the model and can supplement particularly sparse data. For further information on creating training data, Borovicka et al. (2012) offers a summary of collecting representative data and dealing with possible pitfalls.

**Normalisation**

An important consideration before training any network is the normalisation of features. While not required, it is standard practice to perform such data transformations. The benefits of normalising features are due to most algorithms’ initial parameter conditions being optimised for small data values. Normalisation can reduce training times and avoid unanticipated bias on specific features.
There are two main approaches for normalisation, the first being **Standardisation**. Standardisation removes the mean and sets the standard deviation to 1 (Shanker et al., 1996), which may be calculated either by column or globally. The most common approach involves treating each column independently using:

\[
X'[k, i] = \frac{X[k, i] - \mu_i}{\sigma_i},
\]

where;

mean \( \mu_i = \frac{1}{N} \sum_{k=0}^{n} X[k, i] \);

& standard deviation \( \sigma_i = \sqrt{\frac{1}{N-1} \sum_{k=0}^{N} (X[k, i] - \mu_i)^2} \).

Whereas, a global approach is more suitable when the correlations between features must remain the following equation is used:

\[
X'[k, l] = \frac{X[k, l] - \mu}{\sigma},
\]

where,

mean \( \mu = \frac{1}{N} \sum_{k=0}^{n} \sum_{l=0}^{N} X[k, l] \);

& standard deviation \( \sigma = \sqrt{\frac{1}{N-1} \sum_{k=0}^{N} \sum_{l=0}^{N} (X[k, l] - \mu)^2} \).

There are many variations of standardisation, such as removing the median rather than the mean, which are equally valid.

An alternative method is to use **Min-Max** scaling, where all data is scaled within a range of 0 - 1. The following equation is used to achieve such a transformation:

\[
X'(:, i] = \frac{X[:, i] - \text{min}(X[:, i])}{\text{max}(X[:, i]) - \text{min}(X[:, i])},
\]

The Min-Max approach is highly sensitive to outliers making it much less common. Similarly to standardisation, Min-Max scaling has numerous variations, such as scaling to the Nth percentile to reduce the effect of outliers.
1.5.4 Feature Selection & Training

Feature Selection Techniques

Feature selection is vital for all machine learning models. Successfully identifying concise features reduces training times, computational requirements and the possibility for unexpected learning, such as classifying by a noisy measurement (Najafabadi et al., 2015; Cai et al., 2018). The difficulty lies in selecting which parameters offer additional information without removing data that may reveal unknown or unexpected trends.

The challenges raise the issue of Feature Engineering, where observed features are manipulated to represent more concise or alternative projections depending on the problem to be solved. Such engineering can range from simple linear combinations or transforming into a log scale to using autoencoders to reduce greatly multidimensional data into a handful of abstract features (Fan et al., 2019). Each use case is unique and sensitive to differing influences, but in general, an engineered feature is an opportunity to introduce prior knowledge to the problem. Including domain knowledge can help direct the algorithm towards a known result but often limits the ability to identify unknown or unexpected trends.

Considering these fundamental points, there are several approaches to selecting optimal features. Guyon & Elisseeff (2003) provides an in-depth discussion of feature selection methods, with the following sections providing a summary of these ideas adapted towards astrophysical datasets where relevant. If there is good domain knowledge, a sensible starting point is an ad hoc selection for straightforward features, e.g. observation date would offer little astrophysical information. This type of selection speeds up more sophisticated selection methods that usually require iterative evaluations.

Feature Variance

If there is a limited or poor understanding of the domain, removing features should be treated with caution, but a good starting point for unlabelled data is exploring the individual feature variance. If a feature has a limited variance, that feature provides little to no information that differentiates sources, hence more
Figure 1.14  An example using a sample of 10,000 Gaia objects with the representative correlation between features. This example uses Pearson Rank and highlights the useful and linked features visually. Ideal for selecting training data when there is limited domain knowledge.

likely to train a model on noise. Variance is calculated by:

\[ \sigma^2 = \frac{\sum (x_i - \bar{x})^2}{N}, \]

where \( \sigma^2 \) is the variance, \( x_i \) is the values within distribution x, \( \bar{x} \) is the mean of distribution x and \( N \) is the number of objects in distribution x.

**Feature Correlation**

After an initial selection of potential features, a computationally light approach is to reduce redundancy between features. This method is best used in conjunction with domain knowledge to avoid introducing observational bias, such as from scanning laws or crowded regions. Additionally, for supervised learning, the feature correlation reveals which input features correlate with the output,
indicating the potentially useful features. A Pearson Rank calculation would suffice here using the following equation:

\[ r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \]

where \( r \) is the correlation coefficient, \( x_i \) is the values within the distribution \( x \), \( \bar{x} \) is the mean of distribution \( x \), \( y_i \) is the values within the distribution \( y \), \( \bar{y} \) is the mean of distribution \( y \).

Figure 1.14 demonstrates a simple example of feature correlation from a sample of Gaia data, where the output features are galactic coordinates (l,b). Even with a poor understanding of the data, it is evident that the (obviously) ideal features for training are the equatorial coordinate features (ra and dec). Please note an example this simple should never use machine learning in practice. This type of analysis has extensive limitations and is less favourable than user knowledge. Nonetheless, similar to the variance of individual features, the feature correlation can speed up the use of more sophisticated computationally heavy iterative methods.
Feature Importance

An effective metric to identify suitable features is using the feature importance. By studying the feature importance, those that do not contribute during training, or over-reliance on an unexpected feature, will be identified. For example, a model may rely on a non-physical feature, such as the observation date, which indicates the model is not “interpreting” the data as expected. This method is highly effective but relies on repeated use of the computationally heavy training phase for subsequent changes to the final model.

Using the previous example, training a simple classifier to calculate the galactic coordinates of objects. Starting with the same parameters as those in Figure 1.14 to determine the galactic coordinates (l, b) of a number of sources, we can see in Figure 1.15 that predictable only the equatorial (Ra and Dec) offer relevant information. This result confirms the result demonstrated in the earlier feature correlation approach. Additionally, the feature ecl_lat would likely be removed if there were further useful features. With more complex problems, the importance score is more complex possibly returning non-physical correlations, hence requires a more supervised approach.

Overall, all feature selection methods do not replace but supplement good domain knowledge. That knowledge ensures that relationships are physical and not artificial. The best usage of intensive tools such as feature importance is to track the training of the algorithm rather than allowing a black-box learning approach.

Class Weighting for Imbalanced Datasets

In truth, most databases are not ideal training sets, with a substantial issue for classification tasks being class imbalance. This issue with class imbalance is that the disparity biases the training as predictions are optimised to minimise a loss function. In extreme cases with a substantially dominating class, the loss function can reach a local minimum by classifying all objects as the dominating class. An example is when predicting medical data, where positive results are rare, a model may appear to achieve greater than 99% accuracy by simply assigning everything negative.

To counter the effect of class imbalance, implementing class weighting is a powerful tool to counter the disparity. Each scenario is different depending on
the importance of each outcome (such as is a false positive much worse than a false negative?), but a general approach is to use inverse frequency weightings. This is calculated using:

\[ W_j = \frac{N_{\text{total}}}{N_{\text{classes}} \times N_j}, \]

where \( W_j \) is the weighting of class \( j \), \( N_{\text{total}} \) is the total training data, \( N_{\text{classes}} \) is the number of classes and \( N_j \) is the \( N \) for class \( j \).

Alternatively, provided sufficient data is available, the sampling rate can be adjusted to over-sample the minority classes creating a more balanced dataset. As discussed in §1.5.3.

Class weighting is a very useful tool to negate issues in the datasets. This tool is particularly effective for the edge cases between classes as distant populations will remain unaffected as weighting will only adjust a decently working model or minimise the occurrences of local minima dominating the results. See Hashemi & Karimi (2018) for a detailed analysis of weighting in machine learning across various algorithms.

**Hyperparameter Optimisation**

Each machine learning algorithm has many parameters that impact its suitability for a specific task or dataset, referred to as hyperparameters. The exact parameters are distinct to any given algorithm but define the learning depth and how to handle particular decisions. These parameters balance performance and compute requirements with the available training data while aiming to avoid overfitting. Examples of hyperparameters include the number of trees, minimum node size and maximum tree depth for random forest algorithms or network architecture, learning rate and number of epochs for neural networks. However, these parameters are complex and nearly impossible to calculate unless there is an extraordinary understanding of the data and algorithm. In such a case, machine learning is likely to be not required. Therefore, optimal hyperparameters are identified through iterative processes, of which there are several techniques.

While there is no way of calculating the optimal parameters, following guides and knowledge of the algorithm can provide estimates of potential parameter ranges. The parameter ranges should initially be low resolution, spanning a wide range and be gradually refined. Yet, with often ten or more parameters to optimise, this
can be a computationally heavy task. Accordingly, there are two main approaches in the literature.

The so-called hyperparameter sweep is the most structured approach. It involves simply training a model on all possible combinations of the hyperparameters in the grid and selecting whichever achieves the best results. This approach is the most robust and will always identify the optimal parameters at the cost of long run times and requiring significant training hours. This approach optimising five parameters, each with just three possible values, requires 125 individual models to be trained. Considering training times can span significant time-frames, this approach is only suitable for relatively simple models (that are quick to train) and do not require extensive optimisation.

A more sophisticated approach is to use a random search. A random search uses the same parameter grid as the hyperparameter sweep but randomly selects values covering a fraction of the total possibilities. The optimal values are identified by analysing each parameter independently. Visualisation tools, such as HiPlot (Haziza et al., 2020), are convenient to identify the optimal and most influential hyperparameters. HiPlot is used to optimise hyperparameters layer in the thesis within §3.4.3. The advantage of a random search is that it allows more hyperparameter values to be analysed and reduces the complexity of the analysis returning significantly reduced training hours compared to a full hyperparameter sweep.

### 1.5.5 Evaluation Metrics

When dealing with supervised machine learning models, it is common to use metrics to evaluate the performance of a given approach. The most fundamental to this is known as the confusion matrix, which visualises the performance classification for each class. The confusion matrix defines various metrics that compare models, each signifying distinct performance goals. The terms true/false signal correct/incorrect classifications from a model with positives/negatives representing target class/non-target class as visualised in Figure 1.16. The performance metrics are explained in the following section, with the implications for a model highlighted. Note, false positives are also referred to as Type I errors, with false negatives referred to as Type II errors in some technical documentation.

In most cases, the innocuously named accuracy appears as an appropriate
Figure 1.16 Example Confusion matrix, showing the meaning of common terms for classification successes. TP = True Positive, FN = False Negative, FP = False Positive, TN = True Negative. A perfect classification is entirely True Positives and True Negatives, creating an identity matrix (if normalised).

A measure of model performance. Defined as:

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN},
\]

where \(TP\) is the true positives, \(TN\) is the true negatives, \(FP\) is the false positives and \(FN\) is the false negatives. This measure is appropriate where the correct identification of a result is vital in all classes, yet, the limitation occurs when there is an imbalance in class representation or the implications of false classification are imbalanced between classes. Accuracy aims to get the highest possible performance overall, placing equal weight on positive and negative results.

An alternative to accuracy is precision (also referred to as purity), which measures the quality of positive classification for a target class. Purity is defined as:

\[
\text{Purity} = \frac{TP}{TP + FP}.
\]

This measure is best suited towards situations where False Positives (FP) are particularly consequential. Such a scenario occurs when significant action requires a positive result, such as in a criminal court where a false positive would lead to unjust prosecution.

On the other hand, there are many situations where False Negatives (FN) hold the more significant consequence. In these situations, recall (also referred to as
completeness or sensitivity) is the suitable metric, defined as:

\[
Recall = \frac{TP}{TP + FN}.
\]

Recall is suited for situations where false positives lead to follow-up actions, maybe causing short-term distress, but a false negative could cause long-term irreversible consequences. Examples include metal detectors in airports and early medical screenings. In such cases, a positive result will lead to further screening, which will likely correct the result, whereas a false negative could have significant long-term implications.

Finally, in situations where there is no defined favourable false classification (e.g. classifying random images of cats vs dogs) and an imbalance in class representation, a more appropriate measure is the F-Score, defined as:

\[
F-Score = \frac{2 \times Recall \times Precision}{Recall + Precision} = \frac{2 \times TP}{2 \times TP + FP + FN}.
\]

The F-score is a normalised metric, with 1.0 being a perfect model, representing the harmonic mean of recall and precision, where the harmonic mean punishes extreme values the most. Often, the F-score is favourable to accuracy in real datasets due to the naturally occurring imbalance of classes.

In summary, the metrics used to evaluate classification problems are complex and need to be tailored to a given approach considering both the aim and implications for false classifications. This section has summarised the most common and remains a reference to convey the performance throughout this thesis.

**Classification Thresholds**

Classification algorithms within machine learning algorithms return the probability of an object belonging to each class and assign each object to the class with the highest likelihood. The standard threshold for assignment to a class is at least \(1/N\), where \(N\) is the number of classes. The most basic example is a binary classifier, where a probability above 0.50 signals that an object belongs to that class. However, this is often not the optimal location for the cut, particularly for unbalanced datasets. If purity is vital for the model, then a very high threshold will accomplish this, whereas achieving high recall requires a more liberal model, i.e. a lower threshold. Furthermore, refining the class thresholds can often
Figure 1.17 Demonstration of ROC feature space for model evaluation and thresholding curves for classification models. Left: ROC feature space split with example model performances. In general the closer to [0,1], the better the model. Right: Threshold curves in ROC feature space tracing the curves of three hypothetical models. Performance is measured by the area under the curve, with the random selection representing the random assignment of classes.

Identifying the ideal threshold values can be achieved by studying the Recall vs Precision curve. Essentially, this curve tracks the improvements in the true positive rate (recall) vs the false positives rate (purity). Plotting true positives against true negatives allows the comparison of models and if the model is conservative (lower recall) or liberal (lower purity) with assigning positive scores. Fawcett (2006). Figure 1.17 demonstrates the performance of four classifiers in ROC feature space, with A representing a near-perfect classifier, B is a more conservative model, whereas C is more liberal, and D offers a worse classification than randomly assigning a class label. A general rule is that the best model is closest to the coordinate [0, 1].

Refining class thresholds links to the ROC feature space by plotting the resultant curve from tracing the results across the full threshold range (0 – 1). The right side plot in Figure 1.17 shows three generic examples with the area under the curve (AUC) used as the metric for model performance Bradley (1997). Nonetheless, evaluation by the area under the curve places equal weight on recall and purity. Tuning the threshold towards specific metrics can be achieved.
by analysing the curve in ROC feature space or plotting threshold against
target metrics. As always with machine learning training, the optimal practice
requires knowledge of the classification challenge and the potential consequences
of incorrect classifications.

1.5.6 Common Algorithms

Neural Networks

Neural networks are a form of deep learning commonly used to classify complex
data. The term deep describes an algorithm that contains visible and hidden
layers. Visible layers are those the user controls and interacts with, usually the
input and output layers. Between these layers are the hidden layers. The user
defines the shape of these layers with each node an abstract weighted selection of
the nodes in the previous layer (Goodfellow et al., 2016). The presence of hidden
layers improves the ability to identify complex patterns and analyse multiple
data structures. However, due to its ability to analyse complex patterns and the
associated deep learning requirements, it is computationally expensive and often
requires a high-quality training set.

Furthermore, due to the hidden layers, it is difficult to establish the true nature
of the analysis. If not implemented carefully, seemingly excellent results can be
achieved, but the model relies on an unrelated feature. This outcome is why
machine learning, particularly deep learning, is often viewed as a “black box”.
This reputation is established further by detecting complex patterns that are
otherwise impossible to identify. Therefore, it is vital to isolate the desired
features as input as far as possible and critically assess the results at all stages.
Despite the requirements, they are the algorithm of choice for high-dimensionality
data such as images or spectra.

There are many layer types to create a neural network with the optimal
architecture depending on the input and output layers. Typically, a neural
network contains some variation of dense layers. These layers take input from
the previous layer into a defined number of nodes. Each new nodes hold a single
value which is a uniquely weighted combination of the previous layer. The exact
contributions are calculated during the training phase. Often, it is beneficial to
drop a fraction of nodes from each later, using a feature called drop out, but the
Figure 1.18  **Top Left:** A fully connected network showing two hidden layers with a single output.  **Top Right:** a flatten layer used to reduce the data into a single dimension often used after convolutional layers to act as the input for fully connected layers.  **Bottom Left:** A $2 \times 2$ max pooling layer with a stride of 2 and no padding.  **Bottom Right:** Convolutional layer with a stride of 1 and $3 \times 3$ random kernel, no padding. As the kernel passes over the matrix, it multiplies with the corresponding features to produce a convoluted matrix. All nine orange squares contribute to a single value in the resultant matrix.
extent depends on the problem and network size. For multidimensional data (such as images), CNNs are the most effective. CNNs have several convolutional layers, which run a kernel across the input data to extract useful features. Depending on the nature of this kernel, these layers highlight particular features such as sharpening edges or blurring the images. Convolution layers are often paired with pooling layers to reduce the sensitivity of feature position within the data. Max pooling is a filter that pools $N$ features into a single value. In 2D, this could be the rolling average of $N$ features or a rolling maximum. Both convolutional and max pooling layers can use padding around the matrix to maintain a constant feature size. Figure 1.18 offers a visualisation of the most popular layers found in neural networks.

**Tree Based Algorithms**

At a fundamental level, all tree-based algorithms are an ensemble of decision trees. A single decision splits data by a series of axis parallel cuts that group similar objects into a single node. Therefore, each node represents a region of feature space that contains similar objects, visualised in Figure 1.19. The optimal values for each cut are calculated during the training phase. The similarity of classes and the number of trees determines the depth of each tree (Number of...
cuts) required to separate the groups. Naturally, a single decision tree would require a much deeper structure than an ensemble of 100, which reduces the ability to generalise to new data leading to overfitting. An ensemble of trees (referred to as a random forest) calculates the final classification by either an average (for continuous variables) or a majority vote (for discrete variables) (Ho, 1995), (Breiman, 2001). In practice, the random forest is particularly robust as each tree is trained on a subset of the data and often a subset of the features. The ensemble makes them reliable, with an ability to handle incomplete data. Random forests are very commonly used for classification tasks and are much more user-friendly than neural networks.

Another form of tree-based learning is called an isolation forest (Liu et al., 2008). An isolation forest is the same as the random forest except now the feature space is split until each data point is isolated from all others. In this algorithm, the number of cuts required to isolate a given object is a direct measure of how isolated that object is from all others. Isolation forests are great for density analysis and outlier detection.

Random forest models are a naturally supervised algorithm as the tree must be trained on a labelled dataset, whereas the isolation forest is an unsupervised approach. As mentioned, they are excellent at classification and outlier detection problems and scale to large data excellently. Nevertheless, as demonstrated in Chapter 5, random forests can be trained in an unsupervised way.

**Clustering Algorithms**

Clustering algorithms, in data science, refer to acting in a given feature space, hence, can be used as a method of classification (Mouton et al., 2020; Erman et al., 2006; Almeida & C. Sousa, 2006; Zhang, Y. & Zhao, Y., 2004). However, in astronomy, we often desire a spatial clustering of objects dependent on their physical distances and interactions (Kelesis et al., 2022; Yu & Hou, 2022; Kounkel & Covey, 2019). There are two main classes of clustering algorithms, hierarchical and non-hierarchical.

Within these classes, there are several different types of algorithms. The most common centroid-based algorithm is called *k*-means, which iteratively identifies the centre of clusters. Initially, a number of centroids are placed across the feature space. Next, cluster membership is assigned to the nearest centroid. The centroid
Figure 1.20  Demonstration of the step-by-step process of DBSCAN clustering with a simplified example dataset. **Upper Left:** The initial step is to select a random point and identify all points within the distance $\epsilon$. This point becomes a core point if $N_{\text{min}}$ points are within the distance $\epsilon$ otherwise they become a border point, in the example $N_{\text{min}} = 2$. **Upper Right:** Next, if the selected point is a core point, evaluate all points within the $\epsilon$ radius. **Lower Left:** Repeat this process until there are no new core points. **Lower Right:** Finally, once no new points are evaluated in a cluster select a new unevaluated point. If this point does not reach the condition to be a core point, assign as an outlier.
Figure 1.21 Simple example of the associated dendrogram to a series of points to cluster. In this example, if the hyperparameters define clusters at “Cut 1”, the clusters are BC and DE with A, F and G classified as noise. However, if “Cut 2” is used, the clusters would be ABC and DEFG.

Position then moves to the centre of each identified cluster, and memberships are reassigned. This process repeats until equilibrium. The limitations of this method are the poor handling of noise, the preference for clusters with consistent shapes and densities, and the requirement that the number of centroids is known prior to clustering (Lucic et al., 2016; Balcan et al., 2013; Moon, 1996; Singh et al., 2011).

On the other hand, DBSCAN is a common density-based clustering algorithm. DBSCAN calculates the number of neighbours each point occupies within a predetermined radius ($\epsilon$). Figure 1.20 demonstrates the step by step clustering process of DBSCAN and how data points are assigned as a core point, border point or noise as defined by the distance parameter. Limitations of this algorithm are that it is not deterministic, meaning border points can belong to multiple clusters, it cannot handle differences in densities, and often selecting an effective radius is not intuitive. Nonetheless, the advantages are that it can handle irregular shapes, handles noise well, and crucially does not require prior knowledge of the number of clusters (Ester et al., 1996; Ali et al., 2010).

Both k-means and DBSCAN are non-hierarchical approaches. Hierarchical algorithms, on a fundamental level, approach clustering by creating a dendrogram and determining clusters by varying the depth of each cut, as shown in
This work uses HDBSCAN, a hierarchical version of DBSCAN. The method uses the same approach as DBSCAN but with multiple \( \epsilon \) values. The results are translated into a dendrogram based on the various epsilon values. The final clusters are selected from the hierarchy based on several hyperparameters, such as minimum cluster size. This approach is favoured as it allows the detection of clusters with varying sizes and densities and requires no prior knowledge of the number of groups expected [Distefano & Mindrila, 2013; Steinley, 2006; McInnes et al., 2017].

## 1.6 Outline of Thesis

This thesis explores applications of machine learning in astronomy over two major projects, each reliant on data from survey telescopes. Building on the topics discussed above, the first project explores the identification and characterisation of stellar clusters. Chapter 3 details the requirements for accurate clusters and introduced novel approaches for the clustering implementation and validation processes. Chapter 2 presents a formalisation of the clustering process introduced in the previous chapter. This work reflects a novel approach for clustering large data volumes which is a prevalent issue within the field. The approach is presented as a Python package for use cases beyond those within this thesis. Chapter 4 presents an Initial-Final Mass Relation for white dwarf stars. The relation links observed masses of white dwarf stars to the mass of the main sequence progenitor. This chapter represents just one use case of the results in Chapter 3 and by utilising the clusters identified this work presents the most populated version to date.

Chapter 5 contains the second project within this thesis. With the developments in volume and complexity of data produced by survey telescopes, the identification of scientifically interesting objects is increasingly difficult. This chapter explores an unsupervised algorithm to identify anomalous light curves observed by the TESS telescope. By introducing an anomaly metric, the analysis explores the link between anomalous behaviour and evolutionary stages. Furthermore, the chapter contains a deep dive into the mechanisms driving the anomaly score. Finally, conclusions are drawn in Chapter 6 alongside a summary of possible future projects that complement the science contained within this thesis.
Chapter 2

HEADSS: HiErArchical Data Splitting and Stitching Software for Non-Distributed Clustering Algorithms.

2.1 Introduction

As a result of continuous increases in data volume, scalable and distributed analysis techniques are essential. The plethora of incoming data affects all aspects of our lives, from personal data to scientific research. An illustration of this is in the field of astronomy. Over the last two decades, the volume of incoming data has scaled from a few Terabytes to hundreds of Petabytes, increasing to Exabytes over the next few years (Zhang & Zhao, 2015). In many fields, it is insightful to cluster these datasets for a range of applications including but not limited to outlier detection, pattern recognition and even physical clustering.

Unfortunately, clustering algorithms require significant computational resources as, generally, each data point must be evaluated against every other data point resulting in a complexity factor of $O(n^2)$, which is unsuitable for big data analysis. Clustering big data poses many challenges and no single approach performs well against all evaluation metrics (Ajin & Kumar, 2016). K-means clustering scales favourably with complexity $O(nk)$, where $n$ is the number of data points and $k$ is
the number of clusters (Na et al., 2010). There are also works that improve the scalability of K-means clustering through the construction of coresets in Lucic et al. (2016), or a distributed implementation in Balcan et al. (2013). However, K-means clustering has certain characteristics which can limit its suitability for certain datasets. The most prominent constraint is that the number of centroids must be defined, requiring a known number of clusters before analysis. Furthermore, K-means clustering is a form of the Expectation Maximisation (EM) algorithm (Moon, 1996), which is optimal for spherical clusters. There are approaches to reduce the impact of these limitations, but none fully eradicate these fundamental issues (Singh et al., 2011).

Another algorithm to consider is DBSCAN, which can identify an unspecified number of arbitrary-shaped clusters of a specific density. The full algorithm is outlined in Ester et al. (1996) and solves the major hurdles in K-means clustering. DBSCAN defines clusters by analysing the neighbouring points within a defined distance ($\epsilon$), and if the number of neighbours exceeds the selected minimum cluster size the object becomes a central cluster point. This process is repeated until the edge of the cluster is defined, otherwise, the object is classed as noise. However, the rigidity of $\epsilon$ reduces overall accuracy as DBSCAN cannot identify clusters of varying densities. Furthermore, DBSCAN has a complexity issue, even in the best case, it cannot achieve sub-$O(n \log n)$, with the worst-case scenario reaching $O(n^2)$. An issue exacerbated by the fact the full dataset must be loaded to memory and evaluated by a single node (Ali et al., 2010). McInnes et al. (2017) presents a Hierarchical Python implementation, known as HDBSCAN, allowing the identification of varying cluster densities through various values of $\epsilon$. Nevertheless, HDBSCAN continues to have the same heavy computational requirements as the standard DBSCAN. The complexity limitations of big data clustering with DBSCAN and the limitations of K-means clustering discussed above force users to decide between accuracy and scalability.

As clustering is a common task, there are methods for scaling specific algorithms within the literature. Such examples include methods for scaling Hierarchical Agglomerative Clustering (Sumengen et al., 2021), linkage-based hierarchical algorithms (Bateni et al., 2017) and several k-means or centroid-linkage algorithms in Lattanzi et al. (2020).

The astrophysical challenge that inspired the current work is identifying stellar clusters within the Milky Way. Clustering 5-dimensional stellar data (three positional and two velocity measurements), represents the first full-scale result
obtained by HEADSS. The full clustering details are to be described in §3.3 as part of Chapter 3. The challenge represents an ideal dataset for this algorithm, due to the clusters occupying relatively small regions of the total feature space. Furthermore, this type of clustering is representative of numerous physical clustering challenges. Research revealed a lack of easily repeatable approaches within the wider scientific community, with HEADSS aiming to correct this through a user-friendly package.

### 2.1.1 Splitting of Data

A simple method to avoid the complexity and memory requirements is partitioning the data into manageable regions known as “partitions” henceforth. Partitions are comparable to the concept of canopies for clustering and “blocking” in record linkage clustering, which identifies pairs of records that represent the same entity. Both techniques act as a preprocessing strategy designed to reduce the number of comparisons. Blocking acts as a preprocessing filter to partition data to avoid comparisons between distinctly unrelated records reducing computational requirements. Record linkage is commonly used for matching entries between catalogues and the blocking criteria can be trained through supervised techniques using a set of labelled datasets, as described in Michelson & Knoblock (2006) or identified by unsupervised methods as described in O’Hare et al. (2019). Canopy clustering can be considered a density-based analogue to HEADSS. The process of canopy creation is as follows:

- Select a random data point not associated with a canopy to act as a new canopy centre.
- All data points within a distance, $T_1$, are part of the canopy.
- All data points within a greater distance, $T_2$, are associated with the canopy but may join other canopies.
- Repeat these steps until all data is within a canopy.

The effectiveness of canopy clustering relies on suitable parameters and is subject to dimensionality issues (Kumar et al. 2014, Sagheer & Yousif 2021, McCallum et al. 2000). The approach of partitioning data using either of these methods or HEADSS allows the use of complex algorithms on big data with the same
hardware. Nonetheless, partitioning can cause issues to arise, most often on the extremities of a cluster, with several possible edge effects arising.

For this work, edge effects are defined as any artefact, loss of group members or inconsistency introduced by partitioning the data by axis parallel cuts that would otherwise not be present if the complete dataset was analysed. Additionally, an edge effect can be defined as any change in the result that disproportionately affects a specific region of the feature space, i.e. along the threshold of a partition. Typically, edge effects are caused by clusters spanning multiple partitions leading to potential data loss. As HDBSCAN requires a minimum cluster size to be defined, the partial clusters are at risk of dropping below the threshold resulting in incompleteness resulting in clusters with non-physical boundaries. Conversely, clusters large enough to be partially identified in multiple partitions are also problematic, as accurately differentiating valid mergers from neighbouring clusters is challenging at scale.

The prevention and handling of edge effects that occur from splitting data are the fundamental issues this work aims to address by setting a standard approach. The software, HiErArchical Data Splitting and Stitching (HEADSS henceforth), provides both the position for cuts and stitching boundaries that maximises the distance of any given point from an edge boundary, eradicating edge effects in a range of representative datasets and allowing compatibility with all clustering algorithms. Additionally, the software provides functionality to handle the split, clustering and stitching process using HDBSCAN as an example clustering algorithm. Furthermore, where individual clusters span a significant fraction of the feature space, it handles the processing mergers defined by three hyperparameters. HEADSS is an algorithm that works in conjunction with a variety of existing clustering algorithms, ensuring the clustering of big data remains accurate, reliable and repeatable with no direct user judgement required during the splitting or stitching phase.

2.2 Introducing HEADSS

In this Section, I outline the basic principles and assumptions of HEADSS. This work reduces complexity by partitioning the data, reducing the number of objects clustered in a given partition. This process is referred to as “splitting”. Splitting the data by a subset of features naturally establishes a hierarchy. This work
**Figure 2.1** 2D Visualisation of the splitting layers for an $N = 2$, 3 & 4 base layer (Top to bottom). For $N = 2$, the Quaternary layer remains unused. Darker colours indicate overlapping partitions in a single layer.

**Figure 2.2** Visualisation of 2D stitching partitions that maximise the distance to a splitting boundary for $N = 2$, 3 & 4 (left to right). The colour represents the splitting layer hierarchy. The white dashed line indicates the base layer cuts for reference.
splits data into evenly sized partitions across four complementary layers. Due to
the symmetry of the feature space and the repeating patterns, up to four layers
are required for any given $N$, where $N$ defines the number of partitions in the
base layer for each feature to be split. The base layer represents a rudimentary
partition by a simple $N^D$ grid, where $N$ is the number of splits in each feature and
$D$ is the number of features used for splitting. Hence, for 2 features, the feature
space is split into a $N \times N$ grid. This establishes multidimensional partitions
with each length equal to $FR/N$, where FR refers to the “Full Range” of a
given feature in the whole dataset. The Secondary layer offsets the base layer by
$FR/2N$, creating a $(N - 1)^D$ grid, centred where four partitions intersect in the
base layer. The Tertiary layer partitions centre where two base layer partitions
meet perpendicular to the nearest axis. Similarly, the Quaternary layer partitions
centre where two base layer partitions meet parallel to the nearest axis. A 2D
visualisation of the layers for $N = 2, 3 \& 4$ respectively can be seen in Figure 2.1.
Since the clustering partitions occupy an equal fraction of the total feature space,
assuming that the data does not predominantly occupy a small area of the full
feature space.

The amalgamation of these layers allows a single feature space where any point
resides at least $FR/2N$ distant from any boundary. HEADSS also establishes the
splitting layer that maximises the distance from any boundary for the full feature
space, a process referred to as “stitching”. This process selects cluster centroids
that maximise distance from a boundary while dropping repeated clusters. For
the selected centroids, the full cluster remains including the individual members
that span beyond the stitching boundary. An assumption at this stage is that
clusters do not span a large fraction of the total feature space (specifically less
than $FR/2N$). By ensuring the centroid, not the members, resides as far from
a cut as possible, the occurrence of partial clusters is minimised and avoids a
single point occupying multiple clusters provided the assumptions hold for the
underlying distribution. Figure 2.2 visualises the optimal stitching partitions for
the $N = 2, 3 \& 4$ splits seen in Figure 2.1. The central partitions cover identical
fractions of feature space, while the outer partitions extend to the edge of the full
feature space due to a lack of boundaries introduced by the splitting process.

Minimising the number of cuts and dimensionality (features) minimises overhead
requirements and artefacts. HEADSS has a complexity of $O(N_{\text{partitions}})$, where
$N_{\text{partitions}}$ is the number of partitions. $N_{\text{partitions}}$ scales as $(2N - 1)^D$, where
$N$ = number of base layer cuts and $D$ = number of features. The complexity
Figure 2.3  Scaling of the number of partitions required for increasing $N$ with 2 features (left) and number of features with an $N = 2$ base layer (right) highlighting the importance of establishing a small subset of features for the splitting process.

scales unfavourably with the number of features, $D$, but remains reasonable for increases in $N$. The extent of scaling is visualised in Figure 2.3, compounding the importance of reducing dimensionality. As each partition is analysed independently, the clustering can occur in parallel across multiple nodes, whereas the splitting and stitching can also be distributed with software such as Apache Spark.

In summary, the algorithm works as follows:

- Split the data into evenly sized partitions across four complementary layers, see Figure 2.1
- Cluster partitions independently using a user-selected clustering approach.
- Identify duplicate clusters and select the partition that maximises cluster centroid from an edge, see Figure 2.2

under the assumption that the clusters do not span a large fraction of the feature space ($< FR/2N$) and that a small number of clustering features (or a subset of features) exist to partition the data.

In the following sections, examples of the HEADSS API are shown before exploring the performance of the algorithm across a range of test datasets selected to challenge the assumptions, showcase the performance and highlight the results when the assumptions do not hold. In §2.5, a modified workflow is demonstrated that handles clusters that do span a significant fraction of the feature space.
2.2.1 Quick Start Guide

The GitHub¹ contains a series of user guides, including a quick start guide and all the code required to reproduce the analysis in this work. However, there are two major use cases to consider, those that use the integrated clustering algorithm and those which use alternative clustering algorithms which require greater user input but offer greater flexibility, such as running partitions concurrently. Currently, HDBSCAN is the sole integrated algorithm, with analysis performed with the following code:

```python
import numpy as np
import pandas as pd
import HEADSS

# import full dataset for clustering
data = pd.read_csv(filepath)

# Perform split, clustering & merge using HEADSS.
merge = headss_merge(df=data, N=2, merge=False,
                      split_columns=['col1', 'col2'],
                      cluster_columns=['col1','col2'],)

# clustering result returned as a pandas.DataFrame.
merged_df = merge.members_df
```

Listing 2.1: Simple HEADSS example.

All clustering parameters inherited from HDBSCAN and for merging, described in § 2.5, are omitted but shown in the user guide. The API is designed for the integrated use case, whereas the functionality for splitting and stitching is shown in the user guides. The most common function will be retrieving the partitioning and stitching boundaries which can be obtained using a few lines of code:

```python
import numpy as np
import pandas as pd
import HEADSS

# import full dataset for clustering
data = pd.read_csv(filepath)

# Calculate regions to split
head = headss_regions(N=2, df=data,
```

¹https://github.com/D-Crake/HEADSS

65
Figure 2.4  Example datasets used to evaluate the performance of HEADSS in conjunction with HDBSCAN

Listing 2.2: HEADSS example for returning splitting and stitching boundaries.

This code showcases two splitting approaches; either allow HEADSS to assign a “partition” column to the entire database, or for particularly large datasets, head.split_regions provides the split boundaries. If the dataset is too large for memory, HEADSS can calculate the suitable partitions using the maximum and minimum values for each feature. This approach allows the user to split and cluster by partition, while the head.stitch_regions provides the boundaries for stitching regions for the resulting cluster centroids.

2.3 Example Datasets

We’ve selected ten examples from Fränti & Sieranoja (2018) that demonstrate and push the limits of this software while representing various potential use cases. Figure 2.4 shows the selected datasets. They range from the highly-
Table 2.1  HDBSCAN hyperparameters for complete dataset clustering.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>min_samples</th>
<th>allow_single_cluster</th>
<th>cluster_method</th>
<th>min_cluster_size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregation</td>
<td>20</td>
<td>False</td>
<td>leaf</td>
<td>20</td>
</tr>
<tr>
<td>a3</td>
<td>20</td>
<td>False</td>
<td>leaf</td>
<td>20</td>
</tr>
<tr>
<td>flame</td>
<td>5</td>
<td>False</td>
<td>leaf</td>
<td>20</td>
</tr>
<tr>
<td>pathbased</td>
<td>10</td>
<td>False</td>
<td>eom</td>
<td>5</td>
</tr>
<tr>
<td>spiral</td>
<td>10</td>
<td>False</td>
<td>leaf</td>
<td>5</td>
</tr>
<tr>
<td>D31</td>
<td>20</td>
<td>False</td>
<td>leaf</td>
<td>20</td>
</tr>
<tr>
<td>birch1</td>
<td>10</td>
<td>False</td>
<td>leaf</td>
<td>200</td>
</tr>
<tr>
<td>jain</td>
<td>1</td>
<td>False</td>
<td>eom</td>
<td>50</td>
</tr>
<tr>
<td>t4.8k</td>
<td>10</td>
<td>False</td>
<td>eom</td>
<td>20</td>
</tr>
<tr>
<td>worms</td>
<td>200</td>
<td>False</td>
<td>leaf</td>
<td>200</td>
</tr>
</tbody>
</table>

dense birch1 and worms with approximately 100,000 data points to the sparsely populated flame, pathbased, spiral and jain each with around 300 data points. This selection of datasets contains examples with numerous compact clusters while others contain sparse clusters that span a large fraction of the feature space. This work has been developed with the physical clustering of astronomical data in mind, where a3, D31 and birch1 are the most representative and satisfy the assumptions in § 2.2. The test datasets are introduced in the following papers Gionis et al. (2007); Kärkkäinen & Fränti (2002); Fu & Medico (2007); Chang & Yeung (2008); Veenman et al. (2002); Zhang & Zhao (2015); Jain & Law (2005); Karypis et al. (1999); Sieranoja et al. (2019).

2.4 Performance

To evaluate the performance of HEADSS, a baseline with no splitting of the data is required. Due to the reasons outlined earlier in § 2.1, HDBSCAN is used throughout the remainder of this chapter, but any algorithm can be implemented if preferred. Using the hyperparameters in Table 2.1 the algorithm identifies the clusters shown in Figure 2.5. These parameters were optimised by visual inspection due to the lack of noise and optimal result being evident. Good agreement is seen with visual classifications for 7 of the datasets for the major clusters, with the exceptions being worms, pathbased & jain. Worms is a dataset with clusters of varying density and noise. Overall, the majority of clusters appear partially identified with a significant fraction lost as noise. Jain performs well separating the two curves, however, half of the structure in the upper curve is
lost due to low density. Finally, the worst performance by far is pathbased, the outer ring is poorly identified with sections merging with the central clusters.

Recall that this work does not aim to improve on the baseline results in Figure 2.5 as the aim is that HEADSS seamlessly assists implementation but does not impact the clustering results. The performance is analysed for an $N=2$ implementation due to the limited size of the example datasets. In this implementation, with two features, each partition covers 25% of the total feature space. In the datasets where clusters span a significant fraction (> 25%) of the feature space, the assumptions do not hold, meaning partial clusters are anticipated. To minimise data loss and account for the new projections, the clustering hyperparameters are adjusted to those seen in Table 2.2.
In Figure 2.6, the resulting clusters from the splitting and stitching process show promising results. Exploring the target datasets (a3, D31 and birch1), near-identical performance to the non-split clustering is observed. Considering these best represent the astronomical data that inspired this work and optimal use cases, the basic splitting and stitching functions are suitable when the above assumptions hold. Nonetheless, to broaden the potential impact of this work the seven remaining datasets must also be considered.

The noisy worms dataset predominantly identifies the same clusters as the baseline as the clusters do not span a sufficient fraction of the feature space to cause edge effects. Initial impressions of the remaining six datasets are that the known clusters are broadly identified but with excessive edge effects. The edge effects are directly caused by splitting the dataset and independently clustering each partition. Therefore, further evaluating these datasets is impossible without first correcting the significant edge effects.

2.5 Removing Artefacts

As mentioned above, the remaining six datasets (Aggregation, flame, pathbased, spiral, jain and t4.8k) all show a significant deviation from the baseline clusters. Typically, identifying cluster mergers is extremely difficult and often requires considerable human input or sophisticated algorithms when partitioning the data. Human input signifies the process is not scalable nor repeatable, whereas a sophisticated algorithm can interfere with the clusters or require additional
resources to be available. With HEADSS, potential mergers have a subset of mutual members from at least two independent splitting layers. This region of potential mutual members enables a numerical evaluation of similarity between neighbouring clusters.

The merging process applies as follows:

- Identify clusters that potentially overlap.
- Compare mutual members in overlapping clusters.
- Quantify defined overlapping parameters to determine suitable mergers.

This numerical evaluation requires a cross-match of members for all clusters that potentially overlap. As cross-matches are computationally expensive, it is vital to identify potential merges efficiently. HEADSS achieves this by evaluating the cluster limits for overlaps, which currently has complexity $O(k^2)$ where $k$ is the number of clusters rather than the number of data points. From this point, only the partitions that potentially overlap can be cross-matched as defined by the split and stitch boundaries described in §2.2. This process creates three further hyperparameters which quantify merges: "overlap_threshold", the fraction of mutual members within the overlap region, total_threshold, the fraction of mutual members within the whole cluster and minimum_members, the minimum mutual members to allow a merge. Each parameter protects from a few rogue points or small clusters falsely merging large sections of the feature space.

At this point, all clusters that span an overlapping area in all features are identified, despite not all interacting nor having common members. An example of this is a horseshoe or crescent pattern such as those in t4.8k. By iterating the list of merger identifiers, rather than the members, ensures that all chains merge to a
Figure 2.7  Clusters identified in the example datasets by HDBSCAN with an $N = 2$ implementation of HEADSS including the merger capabilities.

Figure 2.7 shows the final result from HEADSS with merging hyperparameters from Table 2.3.

HEADSS, with a subsequent merge, generally matches the baseline results in Figure 2.5 for the majority of the datasets, particularly those with high density. Merging has performed best on t4.8k, successfully connecting the six dominant clusters. Aggregation also performs well, but the rightmost conjoined clusters have merged and a small amount of data loss in the largest cluster. The merge has improved the results for spiral, but merging cannot reverse the loss of data in two of the spiral arms caused by the splitting process. Jain is a unique case where the completeness is improved but the ability to identify the full cluster is reduced compared to the baseline. Splitting the data successfully identifies the lost region in the upper curve while splitting the lower curve into several smaller clusters. The lower curve again shows the splitting of different densities first highlighted in a3. The merge reduces the divisions across partition boundaries in the lower curve, but cannot merge clusters split by the differing densities.

Finally, the flame and pathbased datasets must be evaluated. Neither return ideal performance, practically identifying a single large cluster due to their low density. The space between clusters is less defined when split into partitions, yielding clusters which span the gap causing the single cluster after merging. While those results are not ideal, neither dataset represents a realistic use case. For a sparse cluster to span a large fraction of the total feature space and remain identifiable, the dataset is either not very large (HEADSS would not be required)
or obscured by other clusters and noise.

These results show the merging capability provides a stark improvement in the impact and suitability of HEADSS over the results in Figure 2.6 where the merging capabilities are not used. Merging is best suited for clusters with well-defined boundaries, particularly dense clusters. Merging clusters in HEADSS is especially successful in two of the example datasets and somewhat in a further two, representing a wide range of potential use cases. The only use cases where merging was detrimental to the performance were designed to be challenging and do not represent a realistic clustering application where the complexity scaling with the number of members is a concern. Considering these results are solely evaluated with HDBSCAN, performance may improve on specific datasets with other clustering algorithms leading to further suitable use cases.

2.6 Computational Performance

In addition to the standardisation of big data clustering, the ambition of the HEADSS software is to reduce the peak computational requirements. The two factors are the single-node memory requirements and computational runtime. The following section explores the theoretical reduction of these factors for an optimal use case.

Peak memory requirements are dependent on the number of objects within a single clustering calculation. For HEADSS, the optimal dataset is evenly distributed across the entire feature space, meaning the number of objects with each partition is proportional to the fraction of feature space covered. In this scenario, HEADSS offers an expected reduction in memory requirements by a factor of $O(N^D)$, where $N$ is again the number of splits in each feature, and $D$ is the number of dimensions. Considering many clustering algorithms have complexity between $n \log(n)$ and $n^2$, where $n =$ number of objects. When $N = D = 2$ HEADSS offers a $\sim 4 - 16\times$ reduction in peak requirements. The exact factor of reduction depends on the clustering algorithm used, the number of splits and the overall distribution of the data. If the data densely populates a sub-region of the feature space, consider splitting across an alternative set of features. If no such features exist, further partitions in the dense regions or an alternative approach, such as canopy clustering, should be considered.
Figure 2.8  Top: Non-distributed runtimes for HEADSS with multiple $N$ values compared to the standard HDBSCAN implementation for a 2-dimensional feature space with an increasingly large catalogue. The graph is scaled to demonstrate the runtimes of HEADSS improve as $N$ increases. Bottom: Theoretical distributed runtime for HEADSS (runtimes from the top plot are scaled by a factor of $N^D$), assumes a fully concurrent implementation and evenly distributed dataset.
Theoretical runtimes depend on many factors, including but not limited to the choice of clustering algorithm, the IO bandwidth and the dataset distributions. However, using empirical analysis, the impact of HEADSS on runtimes can be evaluated. The runtime of all clustering algorithms is dominated by the number of objects to cluster. As HEADSS facilitates the clustering of smaller partitions, for large datasets (number of objects > $10^6$) the reduced clustering runtimes easily offset the additional overheads introduced. The top panel of Figure 2.8 visualises the non-distributed runtimes of HEADSS compared to a baseline HDBSCAN implementation. The performance is demonstrated for a 2-Dimensional clustering, with HEADSS becoming required from roughly $10^6$ objects. Below this threshold, the overheads introduced by spitting and stitching the data account for a significant fraction of the runtime. Nonetheless, due to the short overall runtime, the increase for HEADSS runtimes is to the order of 10’s seconds. Above the threshold, the increase in clustering runtimes negates the overhead, allowing comparable or even better runtimes as early as $10^7$ objects, a comparatively small clustering task.

The analysis has shown that HEADSS successfully reduces peak memory requirements for a clustering task with comparable runtimes to HDBSCAN on a non-distributed deployment with reduced memory requirements. Yet, a substantial advantage of this package is the ability to run the independent clustering analysis concurrently, essentially distributing the clustering phase. In a deployment with concurrent clustering, the total runtime is determined by the most populated partition. The theoretical possible runtimes are shown in the bottom figure of Figure 2.8, which is the top plot scaled by a factor of $N^D$, which introduces an assumption that the data is evenly distributed and all computation is done in parallel. In conclusion, HEADSS offers orders of magnitude improvement in theoretical runtimes with as little as $10^7$ objects with a considerable reduction in peak memory requirements for a single process.

## 2.7 Broader Impact

Clustering is commonplace within science, with physical interpretations and theories forming from the outcome (Xu & Wunsch, 2010; Dumont et al., 2018; Kounkel & Covey, 2019). This work brings a standard implementation that produces reliable and repeatable results for any clustering algorithm. To my knowledge, this is the first available package that formalises this process. Due
to the careful consideration of complexity and memory requirements throughout, scalability has been ensured by reducing the data size of a given partition and reduced run times even in single-threaded implementations, see Figure 2.8. In testing the memory limit for HDBSCAN is reached, whereas, in HEADSS this limit represents the largest partition possible defining the minimum value of $N$. Furthermore, there is great potential for further improvements in run times through the utilisation of workflow managers such as Slurm [Yoo et al., 2003].

Scaling HEADSS for very large datasets is relatively straightforward. Sufficiently large datasets can be split across a proportionally large number of partitions, reducing the peak computational requirements to sensible levels. In practice, the optimal $N$ used for splits depends on many factors including the composition of features, number of compute nodes, size of compute nodes and the importance of runtime. In practice, a large dataset of $10^{12}$ entries with a fixed $N = 5$ split across two features results in 81 partitions, each with $\sim 10^{10}$ entries. This scale remains a considerable task, hence splitting over three or four features results in 729 or 6561 partitions respectively. Such a split over four features has reduced the peak requirements from clustering one trillion data points to roughly 200-300 million. Alternatively, if only two suitable splitting features exist, the data could be split with $N = 16$, creating 961 partitions, each with approximately 1 billion data points. The only caveat to these extremes is that the largest scale structure is likely to be lost, although it is likely possible to identify this structure using lower-resolution data. A good example would be using maps of stellar positions to identify galaxy filaments. In practice, the galaxy positions would be more suitable and represent a much smaller catalogue.

Returning to the challenge that inspired this work, identifying stellar clusters and co-moving groups within the Milky Way plane, HEADSS offers an opportunity to influence the clustering with prior domain and scientific knowledge. The clustering of stellar clusters requires both positional (coordinates) and velocity data (proper motions). Nonetheless, there is a fundamental understanding that the positional association outweighs the velocity connections, i.e. objects moving in the same direction but scattered across the Milky Way are not part of a common stellar cluster. The solution offered by HEADSS is splitting the data by the positional features and continuing to cluster using both positional and velocity feature types does not negatively affect the clusters identified. Essentially, the proper motion data increases the accuracy, but as a secondary subset of features for this dataset have a lower feature importance. Clustering the Milky Way plane
offers an excellent use case for HEADSS due to the natural feature importance hierarchy and natural distribution across the positional feature space. The results, which form part of Chapter 3, represent a true scientific use case with a dataset of $\sim 10^8$ objects. For use in other datasets, a general rule is to partition the data along features with the highest importance.

\section*{2.8 Conclusions}

This work presents HEADSS, a package designed to facilitate the accurate clustering of big data. I aimed to make any clustering algorithm scalable and repeatable while retaining its accuracy and reducing the size of required compute nodes. I have shown HEADSS represents an improvement in both scalability and run times, using HDBSCAN as an example, across a range of datasets representing a plethora of potential use cases. I also provide a process of merging clusters to expand the potential impact of this work way beyond the specific use cases first envisioned. I do not explore the limitations of HEADSS for high dimensionality as the process is optimised for splitting along a minimal number of features, whilst allowing the selected algorithm to cluster on additional features.
Chapter 3

A Scalable Approach to Identifying Stellar Clusters in the Milky Way Plane

3.1 Introduction

Advances in available data and maps of the Milky Way continue to advance our understanding of our Galaxy and how it has formed. The telescope leading these discoveries is Gaia, providing the most detailed map of the universe to date. As introduced in § 1.1.4, Gaia provides astrometry measurements of over 1.5 billion objects, revealing past galaxy mergers, stellar evolution and much more. A fundamental structure within the Galaxy is the stellar disc, where current star formation occurs. As explained in § 1.2.2, each star cluster originates from a single independent cloud collapse event. The study of such clusters offers an opportunity to understand many aspects of astrophysics, including the stellar initial mass function, stellar evolution, and the study of specific spectra types (Chabrier 2003; Bannister & Jameson 2005; Gaia Collaboration et al. 2018b).

Considering stars and gaseous clouds populate the disc, the evolution of stellar groups in this environment is complex. The earliest star formation within a collapsing cloud disrupts the collapse process of surrounding gas, driving the early evolution. Later, the cluster dynamics are driven by the dissipation of gas from the region resulting from the evolution of the stellar population (Allen...
et al. [2006]). These disruptions cause variations in the trajectory of the cluster members, along with potential tidal disruptions, which lead to the eventual dissipation of stellar clusters. The time frame for clusters to dissipate is not fully constrained and depends on their early evolution. Current understanding places the lifetime at hundreds of millions to low billions of years ($> 10^8$ years).

Nevertheless, observations of stellar populations constrain stellar evolution and the associated models. The invention of the Hertzsprung-Russell Diagram in the early 1900s can be considered the inception of this field. As described in §1.2.1 the HR-diagram is a powerful tool that directly correlates to a population’s physical properties. Due to its versatility, the observational HR-Diagram, also known as the Colour-Absolute Magnitude Diagram (CAMD), features throughout this thesis.

Kounkel & Covey [2019] (KC19 henceforth) present a leading modern paper in this field, demonstrating the incredible potential of *Gaia* when combined with modern analysis tools. This paper demonstrates a clustering algorithm, HDBSCAN (see §1.5.6), discovering thousands of potentially unidentified clusters with greater completeness than previous studies. This chapter builds on KC19 in several ways including state-of-the-art machine learning techniques to improve the data selection methods, develop a novel cluster validation approach and improve the scalability and repeatability of the techniques used. These improvements aspire to offer a repeatable solution for upcoming datasets, such as *Gaia DR4*.

The structure of this chapter is as follows, §3.2 outlines the methodology for identifying spurious sources in *Gaia DR3*. §3.3 discusses the clustering algorithm we deploy to identify, and the novel method developed to validate, the candidate clusters. §3.4 discusses the development and performance of a neural network able to calculate the age of clusters. Finally, §3.5 summarises the results and potential implications of this work while discussing the limitations and scope for improvements in the future.

### 3.2 Classifying Spurious Astrometric Sources

*Gaia DR3* contains 1.5 billion astrometric sources, representing the largest and most complete map of the Milky Way. Nonetheless, a fraction of the photometric data is spurious. I define spurious sources as any data point where the astrometric
Figure 3.1  Colour-Magnitude Diagram of a randomly selected set of Gaia DR3 sources. The presence of astrometrically spurious sources obscures the main sequence and renders giant populations and compact objects indistinguishable from the noise.

measurements are unreliable or non-physical [Figure 3.1] highlights the effect of spurious astrometric measurements on the database, causing populations such as the white dwarfs or Red Giant Branch to be indistinguishable from the central population. Spurious measurements principally originate from falsely associating astrometric source measurements that originate from different celestial objects. The association occurs on objects with minimal separation on the sky, e.g. a few arcseconds, leading to non-physical photometric results. Additional effects from scanning laws and global zero point errors lead to a complex spurious situation, [Fabricius et al. (2021)] provides a detailed analysis of the origin of spurious sources in Gaia DR3. As their origin is well understood, many spurious sources can be identified using axis-parallel cuts. For example, KC19 defines the genuine population as:

- $v_{\text{lsr}}^{\alpha,\delta} < 60 \text{ km/s}$.
- visibility_periods_used > 8.
- $\sigma_\varpi < 0.1\text{mas}$ or $\sigma_\varpi/\varpi < 0.1$, where $\varpi$ is the parallax.
- astrometric_sigma5d_max < 0.3.
• 1.0857/\text{phot\_g\_mean\_flux\_over\_error} < 0.03.

• \text{astrometric\_excess\_noise} < 1 \text{ or } (\text{astrometric\_excess\_noise} > 1 \text{ and } \text{astrometric\_excess\_noise\_sig} < 2).

Where \( v_{\text{lsr}} \) is the velocity in the local standard of rest reference frame, described later in § 3.3.1 and Table A.1 contains definitions for all other features. This approach is indeed transparent and adequate for many use cases. Nevertheless, sophisticated machine learning techniques offer greater completeness and accuracy of the resulting dataset than this comparatively simplistic method.

A high-quality cut of spurious sources prior to analysis will result in higher accuracy and more complete clusters. The major downside to the axis-parallel cuts approach is that it is indiscriminate to good astrometric sources with unusual behaviour and often errs on the side of caution, producing sufficient purity but poor completeness. Therefore, this work aims to use machine learning techniques to identify spurious sources taking inspiration from a method described in Rybizki et al. (2021) (CSAS henceforth), which expands on work described in Gaia Collaboration et al. (2021) (GCNS henceforth). The approach trains a neural network to identify spurious sources for either high or low signal-to-noise regimes. In contrast to CSAS, this work uses MultilayerPerceptronClassifier from PySpark ML (Meng et al., 2016) to train a fully connected model with an analogous four hidden layers also containing 64 neurons using a sigmoid activation function (Han & Moraga, 1995).

The network is optimised using the \text{l-bfgs} solver (Liu & Nocedal, 1989), producing a probability for each object to belong to a given class. A binary classifier is developed, with the likelihood of each class calculated by the algorithm. The selected features for training include:

- \text{ruwe}
- \text{pmra}
- \text{pmdec}
- \text{pmra\_error}
- \text{pmdec\_error}
- \text{parallax\_error}
- \text{parallax\_over\_error}
- \text{ipd\_frac\_odd\_win}
- \text{ipd\_frac\_multi\_peak}
- \text{astrometric\_gof\_al}
• visibility_periods_used
• astrometric_excess_noise
• astrometric_sigma5d_max
• ipd_gof_harmonic_amplitude

These features match those in CSAS and GCNS and, similarly, \( \text{parallax} \over \text{error} \) is abbreviated to SNR henceforth. Taking the absolute SNR avoids providing a flag for spurious sources appearing within the training data. Additionally, only features that are non-dependent on the position in the sky are selected. The remaining features contain information principally on the included error metrics and proper motions, see Table A.1 for detailed explanations of each feature. The decision to align with current literature rather than relying on the results of these other works allows the creation of a scalable pipeline with fully distributed compute from raw catalogue to ageing clusters.

### 3.2.1 Identification of Training Data

As with all supervised machine learning, accurate and representative training data is a fundamental requirement. Therefore, we require a reliable strategy to identify indisputably spurious sources. A naive approach would be to assume all negative parallaxes are spurious. However, this is an oversimplification due to negative parallaxes being a consequence of \textit{Gaia} measuring angles on the sky with finite precision. A more appropriate method is to take a more conservative threshold. Following CSAS, absolute spurious sources are those where:

\[
\text{parallax} \over \text{error}(\text{SNR}) < -4.5,
\]

returning 4.18 million sources.

Identifying a reliable and representative set of good astrometric sources is more challenging. There is no analogue for the previous cut at SNR as a distance-dependent feature. As overcrowding is the leading cause of spurious measurements, it is reasonable to deduce that reliable sources occupy regions that do not feature spurious sources. The sky is split into equal sections using a system called HEALPix\(^1\), from which \textit{Gaia Source IDs} are derived. HEALpix is an acronym for “Hierarchical Equal Area isoLatitude Pixelization of a sphere”, a system that maps a grid of equal area regions onto a sphere. In summary, HEALPix is “hierarchically tessellated into curvilinear quadrilaterals,” meaning

\(^1\)HEALPix: \url{https://healpix.sourceforge.io/index.php}
Figure 3.2  Spatial distribution of Good and Spurious sources. Spurious sources are selected as those with $\text{parallax\_over\_error} < -4.5$, whereas the good sources are from pixels with no spurious sources and are present in PS1, causing the two classes to occupy separate regions in the sky.

the initial level splits a sphere into 12 quadrilaterals of equal size known as pixels. Each subsequent layer splits each pixel into four, resulting in 12, 48, 192, and 768 pixels for the first four layers respectively. For more information on HEALPix, see Zonca et al. (2019); Górski et al. (2005).

Gaia Source IDs are defined by HEALPix level 12, where there are more than 200 million pixels. This level offers too much resolution to identify reliable sources. The following equation converts the source IDs to a HEALPix level, reducing the resolution:

$$\text{HEALPix level } N = \frac{\text{source\_id}}{2^{35}} \times 4^{(12-N)}.$$  

For this work, we only select good data from pixels in HEALPix level 6 that contain no sources with significantly negative SNR, $\text{parallax\_over\_error} < -4.5$. From a total of 49152 pixels, 4197 pixels satisfy this condition. From GCNS, we discover SNR is one of the most important features, but due to our selection of spurious sources, this feature would incorrectly act as a flag that all sources with $\text{SNR} < 4.5$ are good sources. To avoid this flag, we split the training into high/low SNR sources with SNR only included as a feature for a high SNR regime, defined as $\text{SNR} > 4.5$, for good sources. The definition of spurious sources remains unchanged for each regime. Due to a calibration error in the $\text{phot\_bp\_mean\_mag}$ passband ($G_{bp}$-band henceforth), causing faint objects to appear nonphysically blue (Riello et al., 2021), additional conditions must be satisfied. So, a cut of $G - \text{band} - G_{RP} < 1.8(1.5)$ for high (low) SNR regimes is applied. A final requirement is that all good training sources must be present in Pan-STARRS1 to remove any remaining spurious measurements.

Figure 3.2 reveals the distribution of good and bad sources. Due to the selection
of good sources using HEALPix restrictions, the two datasets populate discrete regions of the sky, with most spurious sources lying within the plane. The potential effects of the distributions are mitigated by ensuring the training features avoid positional dependency. Nonetheless, we must still consider the potential of indirect positional dependencies through mechanisms such as scanning laws or observational epochs. See Figure 8 & 9 from Lindegren et al. (2021) for examples.

3.2.2 Training the Model

The resulting database of good sources is separated into high/low SNR regimes, where high SNR represents $SNR > 4.5$, whereas low SNR represents object where $-3.0 < SNR < 4.5$. The difference between each regime is that $|SNR|$ is not a feature in the low regime. As explained in §1.5.2, each regime requires feature normalisation, with this section applying the standardisation method (§1.5.3) deploying the PySpark implementation of StandardScaler\(^1\) measured against each training dataset independently.

The two classifiers are trained independently on their representative datasets. The spurious catalogue contains 4,180,244 sources. The low SNR contains 3,399,861 good sources. The high SNR regime contains the low SNR good population supplemented by an additional high SNR 1,641,608 sources. The optimisation of hyperparameters is achieved using a parallel grid search with k-fold cross validation, as introduced in §1.5.4 through PySpark’s CrossValidator\(^2\). Due to the iterative nature, CrossValidator is computationally expensive resulting in only the most impactful parameters being able to be considered. The final values are:

- $maxIter$: 100
- $stepSize$: 0.03
- $blockSize$: 128
- $solver$: “l-bfgs”

We match the layers to those in CSAS, with fully connected layers with $[N_{\text{features}}, 64, 64, 64, 64, 2]$ nodes per layer. These parameters are defined in Table A.2.

---

\(^1\)StandardScaler Docs
\(^2\)CrossValidator Docs
3.2.3 Identification of All Spurious Sources

As can be seen in the confusion matrix, Figure 3.3a, without thresholding the optimised high SNR regime achieves a recall (a.k.a completeness) rate of 97.4% with a precision (a.k.a purity) of 96.2%. Similarly, the low SNR regime achieves a recall rate of 97.2% with 98.5% precision (see §1.5.5 for the definition of each metric). Figure 3.3b presents the ROC curve of the model, see §1.5.5 for an introduction into the ROC feature space. The large area under the curve confirms the high performance achieved. Nonetheless, the zoomed inset reveals that the model has a modest liberal approach to borderline false positives, with the default 0.5 classification thresholds, meaning the model can benefit from altering the classification threshold. Inspection reveals that the optimal classification threshold is 0.7, offering the greatest purity while minimising the impact to recall.

Despite the favourable results displayed by the confusion matrix, possible evidence of bias when applied to the entire database must be investigated. Recall that the distribution of the training data selected good and spurious sources from discrete regions of the sky, see Figure 3.2. As the distributions are an artefact from the selection method, the classification of the remaining sources should
Figure 3.4  All sources spatial distribution of Good vs Spurious sources grouped by distances. The training data distributions are not reflected and confirm that spurious sources overpopulate crowded regions. The figure represents a fraction of the total data, selected to maintain approximately 200,000 – 400,000 objects in each plot.
Figure 3.5  All sources CAMD showing the distribution of Good vs Spurious sources split into distances revealing the recovered evolutionary populations. Notice that the population of good sources becomes less defined at greater distances due to proportionally larger reddening effects. This figure represents the same fractions as those in Figure 3.4 to maintain 200,000 – 400,000 objects per plot.
not return discrete distributions. Instead, spurious sources should congregate in crowded regions, whereas good sources should reflect the distribution of stars within the Galaxy.

Figure 3.4 demonstrates the spatial distribution of objects across a range of distance thresholds. The most striking divergence between the two classes occurs at distances below 500 parsecs. Good sources populate the entire sky much more than spurious sources, which are almost exclusively within the crowded Galactic Plane. As the distances increase, the Galactic Plane becomes dominant in both classes due to the distribution of stars, a pattern increasingly prominent at 3kpcs or further. The distributions of each class demonstrate that the training data distributions are not reflected within the final result, indicating that all selected features do not include positional information.

While the distribution of each class is encouraging, statistically validating the performance beyond the confusion matrix is non-trivial. As the data represents observational data, several known structures must be present. The best visualisation is to produce a CAMD and compare the distributions to the lack of visible structure in Figure 3.1. Figure 3.5 presents a series of CAMDs, binned by distance to reduce the smearing due to uneven reddening effects. Analysis of the distribution of good sources within 500 parsecs (top two rows) reveals the expected structures, including a well-defined main sequence, giants branch and a population of white dwarf stars. Comparatively, no dominant structures beyond sparse traces of the main sequence are present in the spurious source population. As the main sequence population is so sparse, the fraction of false negatives is assumed to remain low across the entire dataset.

Due to the increasing reddening effect, the populations of good sources from distances further than 1000 parsecs show an increasingly less defined main sequence and giant branch. Additionally, notice the lack of lower magnitude stars due to the minimum sensitivity becoming an increasingly dominating factor. The spurious sources population, which shows a similar unstructured population at all distances, displays comparable reddening effects and increasing minimum magnitudes.

Both spatial and photometric distributions indicate that the network identifies spurious sources with acceptable accuracy and completeness. As the classifications act as a prior for subsequent models, false positives have potentially far-reaching consequences. Thus, there is a deliberate focus on precision over
completeness leading to the threshold remaining at 0.7. Based on the training data, the threshold change increases precision by \(1 - 2\%\) while returning \(\sim 0.98\%\) of good sources compared to the original threshold of 0.5.

### 3.3 Identifying Stellar Clusters & Co-moving Groups

The results from §3.2 provide a catalogue of sources with reliable photometry. The following section describes the processes involved with preparing the data (§3.3.1) prior to the clustering to identify stellar clusters and co-moving groups in the Milky Way thin disc (§3.3.2). Finally, the validation process is described (§3.3.3), before moving on to establishing cluster age estimates in §3.4.

The hierarchical clustering algorithm HDBSCAN \(^{\text{(McInnes et al., 2017)}}\) is utilised throughout this section. Recall HDBSCAN is a hierarchical version of DBSCAN, §1.5.6 for more details. The term hierarchical refers to the integration over various cluster radii (epsilon value). This approach allows the identification of an unknown number of clusters with varying densities and shapes. The integration creates a condensed minimum spanning tree, converted into a cluster hierarchy sorted by distance. The results are converted to a dendrogram where clusters are either defined by “Excess of Mass (eom)” or “leaf” clustering. The “eom” method identifies areas of high density across the condensed dendrogram, formally preventing a cluster split within another. The limitation of this method is the tendency to return a small number of large clusters, rather than searching the finer details of a population. The alternative “leaf” clustering method selects nodes (regions of density) from the minimum spanning tree, shifting the focus towards finer details without ignoring larger clusters. Usually, “eom” clustering is more reliable, yet, for this dataset, the “eom” method effectively identifies the Milky Way as a single cluster.

HDBSCAN is applied on the 5-d positional data available through Gaia. Specifically, right ascension (RA), declination (Dec), parallax and the proper motion in RA and Dec directions. These features identify stars in close proximity to each other that are also co-moving. The additional conditions increase the likelihood of identifying groups with a common formation history, minimising the spread of ages. Following suggestions from KC19, the minimum number of
sources for a cluster is 40, with 25 minimum samples per cluster, which affects the number of stars deemed noise, i.e. not part of a cluster.

The parameter selection has followed parameters found in KC19 due to the size and complexity of verifying results. However, the implementation differs in several ways. Firstly, KC19 discovered that HDBSCAN is highly attuned to distance in the feature space. To reduce the effect of distant clusters possessing a higher density, they ran the data on subset shells of data divided by distance. The post-clustering inspection was required to reconstruct clusters near the boundaries. While splitting the feature space into concentric shells reduces the distance-density effect, the effect still exists. Furthermore, removing all edge effects post-clustering is unattainable due to data loss by sections falling below the minimum size threshold. The most prominent issue is that cluster merging is excessively manual and subjective, leading to results that are difficult to reproduce and not scalable. To resolve the distance-density issue this work converts positional data to Cartesian coordinates. Additionally, a novel hierarchical approach to partition the data reduces computational requirements while counteracting potential edge effects.

3.3.1 Clustering Preprocessing

Beyond the distance restrictions, the members from the correct stellar population must be identified. Recall from §1.3 that there are three principal stellar populations beyond the Galactic bulge: the thin disc, the thick disc and the halo stars. These populations differ in age, velocity dispersion, metalicity and star formation rates. The purpose of clustering is to identify groups with a single star formation event, yet, clusters within the older populations will likely have since dispersed. The youngest population is the thin disc, therefore, isolating this population will remove a significant fraction of background noise.

Figure 5 from Schönrich et al. (2010) shows modelled kinematics in (U, V, W) components, suggesting a half-width, full-max value of approximately 30km/s. We also refer to a study of WDs in the Galactic Disc by Rowell & Kilic (2019), where they identify (U, V, W) components in the 10-40 km/s range. These results suggest a $1 - \sigma$ cut lies around 50km/s, and Figure 3.6 shows the distribution of our data. We can see the expected Gaussian with a long tail extending beyond 50–100 km/s. From this evidence and the missing tangential velocity component, we suggest a 75km/s cut in LSR coordinates will isolate the majority of thin disc
Figure 3.6  Velocity distribution of all sources within 1kpc in the local standard of rest. We see the extended tail at high velocities from the thick disc and halo stars begin to dominate between 50 – 100 km/s (2,500 – 10,000 $V_{lsr}^2$).

objects from the other populations. In addition to the isolation of the thin disc, the removal of high-velocity objects increases the contrast of proper motions within the dataset.

**LSR Conversion**

The conversion to LSR units begins with a conversion to galactic coordinates, which requires the position of the North Galactic Pole, whose position is denoted by ($\alpha_{NGP}, \delta_{NGP}$). Also, the longitude zero point is required, defined as “the new Galactic pole of the great circle passing through Sagittarius A” in Blaauw et al. (1960) and denoted by $\theta_0$. The values used henceforward are $\alpha_{NGP} = 192.729^\circ \pm 0.035^\circ$, $\delta_{NGP} = 27.084^\circ \pm 0.023^\circ$ and $\theta_0 = 122.928^\circ \pm 0.016^\circ$ which are values stated in Karim & Mamajek (2017). The conversion from equatorial to galactic units is given by the following equation:

$$
\begin{bmatrix}
\cos b \cos l \\
\cos b \sin l \\
\sin b
\end{bmatrix}
= T
\begin{bmatrix}
\cos \delta \cos \alpha \\
\cos \delta \sin \alpha \\
\sin \delta
\end{bmatrix},
$$

where $\alpha$ and $\delta$ are the right ascension and declination respectively in degrees, $l, b$ are the galactic latitude and longitude in degrees, and the transformation matrix,
\( \mathbf{T} \), is given by:
\[
\mathbf{T} = \begin{bmatrix}
\cos \theta_0 & \sin \theta_0 & 0 \\
\sin \theta_0 & -\cos \theta_0 & 0 \\
0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
-\sin \delta_{\text{NGP}} & 0 & \cos \delta_{\text{NGP}} \\
0 & -1 & 0 \\
\cos \delta_{\text{NGP}} & 0 & \sin \delta_{\text{NGP}}
\end{bmatrix} \begin{bmatrix}
\cos \alpha_{\text{NGP}} & \sin \alpha_{\text{NGP}} & 0 \\
\sin \alpha_{\text{NGP}} & -\cos \alpha_{\text{NGP}} & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

Using the values of \( \alpha_{\text{NGP}}, \delta_{\text{NGP}}, \theta_0 \) stated above, the transformation matrix becomes:
\[
\mathbf{T} = \begin{bmatrix}
-0.05646 & -0.87325 & -0.48397 \\
0.49253 & -0.44602 & 0.74731 \\
-0.86845 & -0.19617 & 0.45529
\end{bmatrix}
\]

The definition of the \( \mathbf{A} \) coordinate matrix is as follows:
\[
\mathbf{A} = \begin{bmatrix}
+ \cos \alpha \cos \delta & -\sin \alpha & -\cos \alpha \sin \delta \\
+ \sin \alpha \cos \delta & + \cos \alpha & -\sin \alpha \sin \delta \\
+ \sin \delta & 0 & + \cos \delta
\end{bmatrix}
\]

The galactic velocity components are given by:
\[
\begin{bmatrix}
U \\
V \\
W
\end{bmatrix} = \mathbf{T} \cdot \mathbf{A} \begin{bmatrix}
\rho \\
k \cdot \mu_\alpha / \varpi \\
k \cdot \mu_\delta / \varpi
\end{bmatrix},
\]
where \( \rho \) is the radial velocity in km/s, \( \mu_\alpha, \mu_\delta \) is the proper motion in corrected right ascension/declination respectively, in arcseconds/yr, \( \varpi \) is the parallax in arcseconds and \( k = 4.74057 \), which is a conversion factor to convert to km/s. These calculations appear in Johnson & Soderblom (1987), which covers the fine details of the conversion and provides examples in the Ursa major group.

To convert between the Galactic velocity calculated and the local standard of rest, \( V_{a,\delta}^{\text{LSR}} \), I combine the results with accurate measurements of the local velocity measurements from Schönrich et al. (2010), which measures the velocity of the sun as \( (U_\odot, V_\odot, W_\odot) = (11.1, 12.24, 7.25) \) km/s. Finally, to calculate \( V_{a,\delta}^{\text{LSR}} \), we use the following:
\[ V_{LSR}^{\alpha,\delta} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} U \\ V \\ W \end{bmatrix} - \begin{bmatrix} U_\odot \\ V_\odot \\ W_\odot \end{bmatrix} \]

A quality control cut is applied to only include objects that satisfy the condition: \( |V_{LSR}^{\alpha,\delta}| < 75 \text{km/s} \). After the final cut is applied, approximately 90 million objects remain in the catalogue suitable for the clustering method described in the following section. Due to the number of matrix manipulations to perform this calculation on this dataset, the distributed capabilities of \textit{Apache Spark} are required to maintain reasonable computational requirements.

### Coordinate System Conversions

As highlighted earlier, remaining in an equatorial coordinate system introduces a distance-density bias which makes clusters smaller but more dense as the distance increases. To accurately analyse the catalogue without splitting by concentric shells, the Cartesian coordinate system offers a suitable solution by removing the distance-density relation. Due to the initial careful selection of data, this conversion does not affect the overall size of the dataset used, and due to the accurate 3-dimensional position data available no loss in overall accuracy. However, due to the lack of reliable radial velocity measurements, the same transformation in velocity space does not offer similar benefits. For velocities, the distance dependence is removed by converting units from mas/yr to km/s.

The following transformation converts observations from spherical to Cartesian coordinates:

\[
\begin{bmatrix} X_{eq} \\ Y_{eq} \\ Z_{eq} \end{bmatrix} = \begin{bmatrix} \cos \alpha \cos \delta \\ \sin \alpha \cos \delta \\ \sin \delta \end{bmatrix}
\]

The conversion creates a Cartesian system with the Z-axis in the direction of the equatorial pole. Convention dictates that for a Cartesian system, X points in the direction of the Galactic centre, Y lies perpendicular within the plane, and Z points out of the plane towards the North Galactic Pole. Using \textit{Johnson & Soderblom} (1987), we use the following relation:
\[
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} =
\begin{bmatrix}
\cos l \cos b \\
\sin l \cos b \\
\sin b
\end{bmatrix} =
T
\begin{bmatrix}
\cos \alpha \cos \delta \\
\sin \alpha \cos \delta \\
\sin \delta
\end{bmatrix} =
T
\begin{bmatrix}
X_{eq} \\
Y_{eq} \\
Z_{eq}
\end{bmatrix},
\]

where \( l \) & \( b \) are galactic latitude and longitude respectively, and \( T \) is defined above as:

\[
T =
\begin{bmatrix}
\cos \theta_0 & \sin \theta_0 & 0 \\
\sin \theta_0 & -\cos \theta_0 & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
-\sin \delta_{NGP} & 0 & \cos \delta_{NGP} \\
0 & -1 & 0 \\
\cos \delta_{NGP} & 0 & \sin \delta_{NGP}
\end{bmatrix}
\begin{bmatrix}
\cos \alpha_{NGP} & \sin \alpha_{NGP} & 0 \\
\sin \alpha_{NGP} & -\cos \alpha_{NGP} & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

Similarly, to convert the proper motions from mas/yr to km/s using:

\[
\begin{bmatrix}
\mu_\alpha \\
\mu_\delta
\end{bmatrix} =
\begin{bmatrix}
k \cdot \mu_\alpha/\varpi \\
k \cdot \mu_\delta/\varpi
\end{bmatrix},
\]

where \( \mu_\alpha \) & \( \mu_\delta \) are the proper motion in corrected right ascension/declination respectively in arcseconds/yr. As with the positional coordinates, these are independent of their distance or coordinates, making them an appropriate choice. Even though velocities are from a different coordinate system than the positional data is of little concern due to the normalisation process and the algorithm treating each feature as independent sets of data, ensuring only values from the same coordinate system are compared during the clustering process.

### 3.3.2 Clustering the Milky Way

The results from the preprocessing algorithm, shown in Figure 3.5, show a considerable deterioration above 1kpc. The decreased signal-to-noise in parallax measurements and increased reddening effects cause smearing, making the populations challenging to distinguish from the main sequence. Additionally, due to the inverse square law and crowding, the data becomes increasingly incomplete with an increased brightness limit. So, for the clustering to remain accurate, it is decided to focus on the region within 1kpc to ensure sufficient completeness and accuracy to explore additional scientific opportunities.
After cutting objects further than 1kpc, approximately 50 million astrometrically reliable sources remain. Clustering at such scales with HDBSCAN is unfeasible due to the poor scalability (between $O(\log(N) - N^2)$) and the non-distributed nature. So, there are two possible approaches, reduce the clustering population or explore alternative algorithms. The latter refers to algorithms which scale more favourably with large datasets, namely K-means clustering with a complexity of $O(N)$ [Na et al., 2010]. The limitation of K-means algorithms is that the number of clusters must be known prior to the clustering. As the number of clusters cannot be accurately determined at this stage, a reduction in population is be required.

To not further restrict the scope of the analysis, we will not reduce the data further. Thus, the feature space must be partitioned, risking the introduction of non-physical edge effects. Considering the data represents physical positions, I conclude that a significant separation between coordinates is sufficient to infer a lack of association, therefore, does not require a more complex evaluation. This conclusion allows the feature space to be split by the three-dimensional positional features, creating a series of sub-regions.

However, as encountered in KC19, partitioning the feature space leads to a variety of edge effects and other inconsistencies surrounding boundary regions. Surprisingly, there is not a standard approach presented within the recent literature, with many implementations relying on user judgement, leading to non-repeatable, nor scalable, results. For instance, KC19 use concentric shells of increasing distance to partition the data, with nearby clusters evaluated for potential merges investigated by the authors manually. This technique indeed suffices at current scales, however, the potential for data loss is high due to the defined minimum cluster size and the difficulty of repeating published results. The following section aims to tackle the most prevalent issues that arise from the partition of feature space for clustering purposes.

**Partition Clustering**

Earlier in the chapter, it was mentioned that edge effects arise when splitting data. In this dataset, edge effects represent incomplete or missed clusters. These issues can arise due to split clusters falling below the minimum cluster size specified or due to the difficulty of accurately identifying and combining clusters that span a boundary (referred to as stitching henceforth). As introduced in Chapter 2, there
are two main approaches to splitting data, either feature space split or density approaches. Due to the increased incompleteness at greater distances, the density approach will create smaller nearby regions leading to a larger number of edge effects on the most accurate data. This section presents a scaled demonstration of the HEADSS software package that maximises the distance of clusters from any edge through a hierarchical split of the data, minimising the presence of edge effects.

As explored in detail in the previous chapter, the method is composed of a series of layers, each overlapping the boundary of the previous layer. The initial data split works with a base layer dividing the data into a simple $N^D$ grid, where $N$ is the number of base splits, and $D$ is the number of features (or dimensions) included in the split. It is best to keep $N$ and $D$ as small as possible to minimise the number of partitions and the potential for edge effect. The base layer covers the full feature space exactly once. The secondary layer contains another $N^D$ grid offset by 50% of a single partition length. Centring the secondary layer on the boundaries of the previous layer accounts for the boundaries perpendicular to the axis.

For this dataset $N = 2$ is possible on the available hardware, meaning the final boundary represents a single central region where all four partitions meet in the base layer. This area is covered by a Tertiary layer comprised of a single region. If $N > 2$, a Quaternary layer accounts for boundaries parallel to the axis. Figure 3.7 offers a 2D representation of the three layers used to split the data. The assumption this analysis is designed around is that for all clustering methods, the feature space must extend beyond the target area to ensure complete clusters.
Splitting the data is relatively simple compared to optimising the stitching of these results back together. This process focuses on three main goals:

- Avoiding split or repeated clusters.
- Returning complete clusters.
- Facilitating the largest clusters possible consistently across the feature space.

The first two are achieved by standardising the approach to ensure a cluster’s centre lies as far from a splitting boundary as possible, whereas the stitching process removes cluster duplication by only selecting the centre furthest from a splitting boundary. The methodology is easiest to visualise, as seen in Figure 3.8.

As detailed in Chapter 2, the Hierarchical Data Split and Stitch process allows for a cluster to span a minimum of 25% of a partition. Hence, fewer partitions allow for larger clusters.

This method transforms a large dataset into manageable sections while minimising duplicated clusters and maximising the completeness of the results. An overall reduction in peak computational requirements is observed, alongside a reduction in potential run times by allowing concurrent analysis. Furthermore, carefully
selecting features to split the data allows for a prioritisation (or hierarchy) of features. For this work, the feature space is split along the 3-D positional data in a $N = 2$ grid, resulting in 27 individual partitions. The most populous regions account for $1/5th$ of the total population, requiring a node with 86GB memory. The feature selection is merited by the fact that objects on opposite sides of our feature space may move in the same direction, but the physical separation ensures these objects do not belong to the same cluster. Due to this, it is beneficial to maintain fewer partitions and not split by proper motion features.

Each partition of the Milky Way data uses identical clustering parameters:

- $min\_samples$: 25
- $min\_cluster\_size$: 40
- $prediction\_data$: True
- $gen\_min\_span\_tree$: True
- $allow\_single\_cluster$: False
- $cluster\_selection\_method$: leaf

Figure 3.9 2D visualisation of partition clustering of the Milky Way disc for $|Z| < 500$ parsecs. **Top:** Visualisation of clusters split by the method described in Figure 3.7. Colour maps reflect the splitting layer and highlight individual partitions. **Bottom:** Exploded view of all layers emphasising the clusters identified and demonstrating the repeated clusters in overlapping regions. The colours are randomly assigned for clarity and do not represent any physical attribute.
These parameters are selected based on guidance from the documentation, e.g. *cluster_selection_method* returns very few large clusters when using ‘excess of mass’ over ‘leaf’ cluster selection method, and the discussion within KC19. The difference in these method lies in the cluster selection from the final dendrogram, with the ‘leaf’ methodology prioritising the final nodes (i.e. bottom up analysis), rather than the excess of mass which identifies regions of increased density across the whole population (i.e., top down approach). All parameters are defined in *Table A.3*. Ensuring the implementation remains consistent with previous works allows the evaluation of the more advanced source detection and clustering approaches.

The clustering identified 15,229 clusters from all partitions of the data. *Figure 3.9* shows the identified clusters, where each partition is displayed to represent their position in the full feature space. Visual analysis of these plots confirms that clusters do not dominate significant fractions of the feature space, hence merging clusters that span multiple regions is not required. *Figure 3.10* represents the final clustering result after the stitching process where 2,763 clusters remain. For this work, we do not attempt to identify major strings mentioned in KC19 due to the increased likelihood of contamination and multiple formation events.

The population distribution of these clusters is consistent with those in KC19. Approximately 75% contains fewer than 100 members and a median population of 62. Nonetheless, these clusters are statistically dense regions of the Galaxy,
yet, there is no verification of a common origin. To my knowledge, there are no standard processes existing in the literature. In the following section, we explore an automated approach to validate identified clusters by using the expected observational features of each cluster.

### 3.3.3 Identifying Spurious or Contaminated Clusters

The results from HDBSCAN are a statistical analysis of the positions and proper motions of the stellar population with no astrophysical verification the associations are not coincidental. Hence, there must be an independent astrophysical validation of clusters. Visual analysis methods in the literature, such as KC19, are labour-intensive, non-repeatable and non-scalable by relying on user judgement.

For meaningful age calculations, the clusters must originate from a single formation event. This scenario will reflect in the colour-magnitude diagram of each cluster and will contain a defined main sequence with a turn-off point.
and evidence of an RGB and WD population. Within 1kpc, the main sequence population is expected to dominate and can be used to identify our target clusters without signs of a complex formation history.

In the following section, we explore a numerical approach to remove unsuitable and non-physical clusters from our analysis.

**Hough Transformation Metrics**

The Hough transformation originates from the analysis of bubble chamber images to detect the traces of charged particles [Hough (1962)], with the form used today and described below first described in [Duda & Hart (1972)]. In mathematical terms, in Cartesian space, a line is defined as \( y = mx + c \), whereas in Hough space, a line is described in parametric terms, defined as \( r = x \cos \theta + y \sin \theta \), where \( r \) is defined as the length of the normal between the line and the origin and \( \theta \) is the angle between the normal and the x-axis. Defining the line in parametric terms allows the line in Cartesian space to be described by a single point in Hough space. Similarly, the reverse of this process transforms a single point in Cartesian space into a curve in Hough space. Therefore, when several points lie along a common line in Cartesian space (e.g. the main sequence). Each points is represented by an individual curve with the line being represented by the single point of overlap between all the curves in Hough space. [Figure 3.11] demonstrates a simple example of transforming populations from Cartesian to Hough projections. This process is computationally efficient for detecting lines and edges within images. The algorithm is used when real-time analysis is required, such as for self-driving algorithms [Waykole et al., 2021].

Within astronomy, Hough transformations have been used for several uses such as cleaning plate survey data and the identification and removal of artefacts such as satellite tracks or dust contamination [Storkey et al., 2004]. Identifying cold stellar streams in the Milky Way halo that originate from dwarf galaxy spheroidal mergers, providing a key insight to the formation of large galaxies [Shih et al., 2022]. Additional applications of the Hough transformation include the identification of debris in low earth orbit, detection of dust in the coma surrounding comets, and the detection of solar flares to name just a few [Bahcivan et al., 2022; Lemos, 2022; Patel et al., 2021]. Applying the Hough transformation to the result colour-magnitude diagrams from the clustering phase does not identify a line of best fit but rather defines the edges of populations such as
FIGURE 3.12 Example line detection with Hough transformations for a cluster identified with HDBSCAN. Left: Observed Colour-Magnitude diagram of the cluster NGC_2546. Middle: Hough transformation of the Colour-Magnitude diagram (left) with the 10 most dense regions highlighted by red dots. Right: Combined plot of the Colour-Magnitude diagram with edges detected by the Hough transformation (dark red) which correspond to the dots in the middle plot. A linear regression is also included for comparison (dashed blue line).

the main sequence. Edge detection is more reliable than a typical fitted line as edge detection is not affected by spurious members or artefacts. Our approach allows the identification of numerous (up to 10) edges and reducing each cluster to three parameters:

- **Mean Gradient** ($\bar{m}$)
- **Minimum Width** ($w_{\text{min}}$)
- **Variance of Gradients** ($\sigma_m$)

The mean gradient is simply the mean of all edges detected in the Hough transformation and is comparable to a linear fit and is an important metric as it allows the removal of clusters that do not feature a prominent main sequence. The minimum width measures the thickness of an area required to encompass 80% of the data points within the cluster. The 80% limit ensures that the width measurement is not penalised for clusters with a prominent giant branch, or a small number of contaminants, while reflecting significant contamination. Finally, the variance of all edge gradients reduces the information lost by simply averaging. Figure 3.12 visualises the spread of gradients and reflects the prominence and definition of the main sequence. Thus, a contaminated group or one with a complex formation history will be indicated by an unusual average gradient, reinforced by a large width and variance.
To accurately evaluate the dataset spurious trends in the dataset should not be part of the analysis. As introduced in §3.2, Gaia data is contaminated in the BP-band at the faint end of observations. Hence, the contaminants impact the Hough metrics described above. Due to these contaminants, the Hough transformation is applied only to the population above 18th magnitude. In astrophysical terms, this has little impact as the main sequence extends significantly above this threshold for the distances probed. To avoid potential low-population statistical consequences, clusters with less than 20 members (post-magnitude cut) were classified by default as spurious clusters.

Training the Validation Model

Reliably identifying the target clusters through axis parallel cuts on the metrics derived by the Hough transformation is not possible due to the complex structures within the data. Thus, a more complex approach must be taken but for such a small number of input features (three) a neural network is excessive. Hence, a random forest is selected as it offers a sophisticated but less complex solution than a neural network. The importance of the three metrics has been, in theory, justified above as they each represent a separate astrophysical property. Initial testing, using a small sample of manually labelled data, confirms that all three features show higher importance than the average membership probability metric within HDBSCAN. $\bar{m}$, $w_{\text{min}}$ & $\sigma_m$ show a feature importance of 0.54, 0.32, 0.11 respectively compared to probability’s importance score of 0.03. While sufficient to evaluate the feature importance, the small training set is unsuitable for the final model.

The output from the Hough transformation is abstract and not human-readable. Thus, the performance of any model is evaluated through the CAMD, despite the random forest having no direct access to this data. Furthermore, considering the random forest is evaluating just two classes and that the majority of data shows a defined main sequence, a randomly selected training set will heavily bias towards astrophysically meaningful clusters. Instead, as described in §1.5.1 this dataset lends itself towards active learning, due to the fact it focuses additional training data to be sampled from the region of feature space where the classification is most ambiguous. The small initial training data is identified from an initial study of the CAMD providing ~30 data points to train the initial model. After each evaluation of the data, the 16 most uncertain observations are visually
classified, again using the CAMD, to improve the existing training data. After four iterations, approximately 100 labelled clusters, the results stabilise, returning consistent classifications.

**Cluster Validation Results**

Continuing the evaluation of the model using photometric data, Figure 3.13 demonstrates the evolution of clusters as a function of the probability that a cluster is physical, as calculated by the trained random forest described above \(P(C_G)\). Despite the small training set, compared to traditional supervised training sets, there is a visible correlation in Figure 3.13. 55% of clusters have a \(P(C_G) > 0.99\), with just 0.56% returning \(P(G_C) < 0.2\). The over-representation of clusters with a well defined main sequence and turn off (indicating members have a common stellar age), compared to highly contaminated clusters, is a validation of the clustering method and a demonstration of the effectiveness of active learning for this dataset.

Analysing the clusters in Figure 3.13 leads to the conclusion that a reasonably conservative threshold for spurious clusters is \(P(G_C) < 0.6\). This threshold represents the removal of 9.8% of all clusters, which is in line with the 5 – 10% estimate in KC19. The remaining clusters represent those that, according to their photometry at least, have a common formation history and have not undergone major perturbations or disturbances during their history. While the threshold certainly removes some valid clusters, as with the treatment of spurious sources, the focus remains on accuracy rather than achieving maximum completeness. The following section explores the approach used to determine the associated ages of these clusters.

### 3.4 Cluster Age Determination

Assuming each cluster identified by the validation process originates from a single formation event, evolutionary models can estimate the time since formation, known as the cluster’s age. The leading evolutionary models are PARSEC (Bressan et al., 2012) and MIST (Choi et al., 2016; Dotter, 2016). These models are sophisticated hydro-dynamical models produced to track the evolutionary curves of stars across a range of masses. Inversely, by sampling a population of
Figure 3.13  CAMD diagrams of randomly selected clusters, grouped by the calculated probability of true cluster, $P(C_G)$, by the cluster validation analysis. $P(C_G)$ increases downwards in non-linear steps to better represent the distribution of scores.
stars with an estimated mass distribution at a specific age, the resultant curve, known as an isochrone, represents a theoretical CAMD for a stellar cluster. By fitting these models to observed clusters, estimates of physical parameters are made. The most significant feature during this process is the main sequence turn-off point. This feature represents the mass of stars leaving the main sequence and acts as a strong correlation tool for isochrone fitting (Hills et al., 2015).

Typically, MCMC Bayesian fitting is used to identify a posterior isochrone for an observed cluster. The leading software is BASE-9 which achieves estimates within 10% of the true value, alongside constraining other physical parameters, such as metallicity, helium abundance, distance modulus and line-of-sight absorption (von Hippel et al., 2006; Hills et al., 2015). Nevertheless, BASE-9 fitting uses complex iterative processes, resulting in runtimes at the order of hours for a single cluster (von Hippel et al., 2014).

To run the BASE-9 software on each cluster validated in §3.3.3 would take over five months. Such a task is evidently, not feasible nor scalable. Once more, machine learning offers a solution by training a model focused on identifying cluster ages. Kounkel et al. (2020) (Follow-up paper to KC19) includes a neural network, Auriga, which estimates the age, distance modulus and extinction of clusters with promising results. Nonetheless, by correcting well understood effects such as distance and dust extinction removes potential degeneracy. Thus, removing the requirement of calculating these measurable parameters, the age estimates can be improved while reducing the volume of required training data.

### 3.4.1 Training Data

A suitable dataset to sample training data for an ageing model is the Milky Way Stellar Clusters Catalogue (MWSC), containing approximately 3000 known stellar clusters (Scholz et al., 2015). These clusters are discovered using 2MASS data and located within 1.8 kpcs. Cantat-Gaudin et al. (2018); Scholz et al. (2015) provide corresponding age estimates for the clusters within the MWSC. The set of observational training data is collected by cross-matching photometric observations from Gaia & 2MASS (Marrese, P. M. et al., 2019). Due to 2MASS not achieving the same depth as Gaia, the resultant training set contains 1229 clusters with age estimates and photometry in all selected bands. Unfortunately, a sample of this size is insufficient to train a network as complex as the one required to determine cluster ages. This is due to the data being unable to fully
populate the entire feature space, particularly when split into training, testing and validation sets. Increasing the data volume either requires additional catalogues or data manipulation. Alternative catalogues are not necessarily compatible with the MWSC as the ageing techniques are not strictly defined. Moreover, additional datasets will not contain adequate data volume to train a network. So, data augmentation is required.

Despite the limited number of clusters, they, on average, have many more members than those identified in §3.3. Hence, by sub-sampling these clusters to match the population distribution of the target dataset, each MWSC cluster can represent many data points increasing the volume of training data. While the sampled clusters contain the same population of celestial bodies, each combination selects a unique group of objects and the number of members. The subsampling reduces the chance of overfitting the model. By randomly sampling cluster members (weighted by the published probability each star is a member of the target cluster), the 1229 clusters are sub-sampled to over 150,000 combinations. The number of members within sub-sampled clusters matches the distribution of members in the clusters identified in §3.3.

Data augmentation by sampling clusters successfully increases the data volume, but it does not improve the distribution of ages. Due to the limited volume and data quality, accurate interpolation between the observed ages is problematic. Fortunately, the models that constrain the observed cluster ages can also construct simulated data. The synthetic data aims to improve age estimates in areas insufficiently represented by the observed data. Furthermore, due to contamination in observed clusters, a representative synthetic dataset without contamination is expected to increase overall accuracy with finely sampled accurate age estimates.

**Synthetic Clusters**

There is a good agreement between stellar evolutionary models, with the differences deriving from the handling of particular chemical elements. The PARSEC models are more prominent within the community and offer evolutionary tracks for various telescopes passbands and extinctions. While the MIST models are an option, they are cited less within the literature. Hence, the PARSEC models are selected to align best with the literature. The synthetic data begins by producing a database of observational properties for a range of stellar masses at various
stellar ages. The database created for this work includes stellar masses ranging from $0.09 - 62.67 M_\odot$ across an age range of $6.6 - 10.1$ dex.

Interpolating the grid with a 2D spline allows the creation of a synthetic dataset with any age within the $6.6 - 10.1$ dex. For the initial synthetic dataset, 5000 clusters are sampled with random ages, each with 1000 members of various masses sampled from the Chabrier Prior, a distribution following a broken power law, built into the isochrones python package. Due to the fact the model is theoretical, the resultant clusters contain no contamination and limited spread of ages. While this provides a clean training set, it is unrepresentative of the observed data. Hence, to improve the representation of the synthetic dataset, a small spread of ages is introduced of 0.1 dex. Similar to the real clusters, the synthetic data is sub-sampled to increase the volume of data.

The synthetic data offers an accurate theoretical result to train the model, however, they remain insufficiently representative of the target data. Likewise, the real training data is much more representative of the target data but does not offer a sufficient resolution on the target feature of cluster age. Thus, combining the two datasets creates a more accurate and representative training set for this complex task. Determining the volume and ratio of real and synthetic data is part of the hyperparameter optimisation process in §3.4.3.

### 3.4.2 Training Data Preprocessing

Four main parameters affect the observations of a cluster; age, metallicity, extinction and distance. Despite the data creation techniques, the training set remains imperfect. Correcting the photometry to remove the effects of distance and extinction does not affect the final age estimate, due to both being observer effects, but does reduce the required volume of training data. Considering clusters identified earlier in this work do not have reliable metallicity measurements, the training data must represent a range of metallicities. The real data is also observational and hence assumed representative, whereas the synthetic clusters are sampled from a range of metallicities. While metallicity does impact the evolutionary timescales, the consequence is limited. Moreover, the largest impact on age estimates is assumed to originate in the contamination of observed clusters.

Ageing clusters requires accurate photometry across several passbands. Gaia passbands cover the visible spectrum but do not offer enough data alone.
Combining the observed data with 2MASS and WISE allows a more accurate age estimate. Yet, the datasets of WISE & 2MASS do not match the depth achieved by Gaia. This disparity is largely due to the comparatively longer passbands of 2MASS and WISE corresponding to a less bright region of typical stellar spectra, as shown in Figure 1.2 and Fouesneau et al. (2022). Hence, to ensure acceptable levels of completeness, only 2MASS data is included as this represents the lesser reduction of completeness. Furthermore, as explained in §3.2, there is a systematic calibration error most prominent in the $G_{BP}$-band of Gaia DR3, causing the faint objects to appear non-physically red (Riello et al., 2021). Such a prominent effect renders the entire passband unusable for this model, representing a significant loss of the available data.

Hence, it is worth considering the use of Gaia DR2 data to counter the loss of data. While this dataset also contains a slight bias compared to SDSS photometry in faint objects, 3 – 4 mmag at 16th magnitude in the G-band, the effect is much less prominent than those in DR3. This effect, known as “hockey stick” is believed to originate from a not fully successful background subtraction (Evans et al., 2018). As the hockey stick does not disproportionately affect a single photometry band and is less prominent than the issue in DR3, this dataset is more suitable for the ageing training data as all three passbands can be used. Thus, I accept the slight reduction in accuracy and for the ageing model use Gaia DR2 data, rather than attempt to mix two data releases.

**Extinction**

As extinction is caused by the presence of dust between the observer and source, the total observed extinction depends on the distance and position in the sky. Dust extinction causes reddening as shorter wavelengths are scattered to a greater extent, see §1.1.3. While the network may be able to estimate the effects of dust, this requires a representative sample, which isn’t available, or would rely on some simulated component to expand the training set. As this option will most likely reduce the ageing accuracy while increasing the complexity of training the network, the decision to remove the effects of extinction is the preferential methodology.

Fortunately, several astronomical surveys map dust in the Milky Way, allowing an analytical approach to calculating the effects of reddening. Dustmaps provides several two and three-dimensional maps through a Python API (Green, 2018).
This package calculates either the line of sight extinction or exact extinction dependent on the availability of distances. The map selected is described in Green et al. (2019) and covers a region with declination $> -30^\circ$, derived from over 800 million stars using photometric data from 2MASS & Pan-STARRS 1 and parallax measurement from Gaia. Figure 3.14 displays a section of this map centred on Cygnus X.

The extinction of individual objects for a given passband, $A(E)$, is given by:

$$A(E) = E R,$$

where E is the reddening, and $R$ is the “extinction vector”. The extinction vector varies for each passband, found in Table 3.1.

### Distance

As with extinction, correcting the effects of distance avoids relying on the algorithm to interpret that the distance does not affect cluster age. Fortunately,
due to the requirement for clustering, all the objects have reliable parallax measurements, meaning we can correct the magnitude using the distance modulus:

$$m - M = 5 \log(1/\varpi) - 5,$$

where $\varpi$ is parallax in arcseconds. Converting all measurements to absolute magnitude removes the need to augment clusters to various distances, reducing the volume of training data and minimising the model complexity.

This correction of distance and reddening from dust extinction removes the dominant extrinsic observational effects. Additionally, not requiring the model to predict these measured parameters simplifies the feature space, increasing the relatively limited training data coverage.

Figure 3.15 contains a small sample of corrected real and synthetic clusters. Despite the removal of these observational effects, a notable contrast between the two datasets is that the synthetic data returns a tighter spread along the main sequence. This effect reflects the reduced age distribution and lack of additional observational features, such as those within the cluster. As the inclusion of synthetic data strives to improve the interpolation of ages from the real clusters, the impact on overall performance is expected to be minimal.
<table>
<thead>
<tr>
<th>model</th>
<th>Convolution</th>
<th>Max Pooling</th>
<th>Dense</th>
<th>activation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>relu</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>7</td>
<td>2</td>
<td>relu</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>relu</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>relu</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>other</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>relu</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>relu</td>
</tr>
</tbody>
</table>

**Table 3.2** Number of Convolution, Max Pooling, Dense layer types and activation function within each model evaluated within the optimisation analysis. Dense layers always occur after a Flatten layer, with the final layer reducing the model to the final output layer containing a single node.

### 3.4.3 Training the Network

All neural networks contain the same basic structure; an input layer, a series of hidden layers and a final layer. The input layer of the ageing model requires a 2D array of size $[250 \times 6]$, corresponding to six passbands from Gaia & 2MASS for up to 250 celestial objects. The `Sequential` function from TensorFlow is used throughout this section to build the model architecture. Only objects observed in all six bands are included because neural networks cannot accept null data values, and zero-point values often cause complications. Similarly, as most sub-sampled clusters contain less than 250 members, a consideration is how to pad the input arrays with less than 250 objects. After initial tests, the most effective approach is to repeat members to fill the empty rows. The selection of repeated members is random, weighted by the probability of an object belonging to the cluster as given by the MWSC database. The resultant input data removes any missing data and decreases the impact of spurious members. The network output is a final layer containing a single node, corresponding to a cluster’s age estimate.

As discussed throughout this chapter, machine learning models require hyperparameter tuning to return accurate results. Neural networks are no exception to this, and due to the complexity of the task, the tuning has greater significance than the models covered previously within this work. Packages such as TensorFlow ([Abadi et al., 2015](#)) contain tools to optimise hyperparameters through multiple training iterations. However, combining the real and synthetic training datasets brings additional hyperparameters to evaluate, meaning these tools cannot evaluate these additional parameters and the parameters of the...
neural network simultaneously. Instead, for this work I use a random search of parameter space to evaluate the parameters deemed the most impactful.

The identified hyperparameters to optimise are:

- Hidden layer architecture
- real_perc
- syn_weight
- normalise_by
- epochs
- early_stop
- sample_weight
- patience of early_stop

Effective optimisation of all hyperparameters requires an unfeasible number of training iterations. Focusing on the most impactful parameters in isolation significantly reduces the requirements. The most impactful parameter is the architecture of the hidden layers. Several architectures are considered, each placing differing importance placed on convolution, max pooling and dense layers, §1.5.6 contains definitions of each layer type. Table 3.2 outlines the layers in each model. To accurately measure the impact of the model, all other parameters are set to initial estimates. Table A.4 describes the hyperparameters that originate from TensorFlow’s neural network algorithm, whereas real_perc is defined as the percentage of real clusters in training data, syn_weight defines the weighting of synthetic clusters during training (requires sample_weight to be True). Finally, normalise_by refers to normalising the data globally or by columns as described in §1.5.3.

Model Architecture

Each model architecture is run twice with the mean square error (mse) loss function. Four performance metrics are averaged across both runs to evaluate each model; $\chi^2$, mean square error, mean absolute error (mae) and the Pearson correlation coefficient (r). As shown in Figure 3.16 the model selection can account for a $\sim 50\%$ reduction in errors. Conversely, the alternative activation function does not result in a meaningful improvement. Model 2 presents the optimal results across all metrics. The superiority continues when isolating the performance for the real dataset, whereas model 2 is only the third best model for synthetic data. Interestingly, the additional dense layers of models 5 and 6 are detrimental to the performance, as are the additional convolution and max
Normalised performance metrics for ageing stellar clusters across model architectures outlined in Table 3.2. Mean square error (mse), mean absolute error (mae), and $\chi^2$ (chi) are calculated such that lower is better, whereas higher is better for Pearson’s rank ($r$).

The performance of the real data leads to the selection of model 2 for further optimisation. Models with even less complex hidden layers are not considered to maintain sufficient flexibility in the model. As the training data is sub-optimal, there is considerable importance for the model to maintain high resilience.

The network has the following architecture. An input matrix of $[250, 6]$, where the six columns represent G, BP-band, and RP-band Gaia passbands and J, K, H passbands from 2MASS. The network performs two $(3 \times 3)$ 2D convolution layers through 8 channels, each followed by a $(2, 2)$ Max Pooling layer. The resulting output has a shape $(62 \times 1 \times 8)$ which is passed through two further $(3 \times 3)$ 2D convolution layers, this time through 16 channels, each followed by a $(2, 1)$ Max Pooling layer, resulting in an output matrix of shape $(25 \times 1 \times 16)$. The output is flattened and passed a single fully connected layer with 64 nodes before a final fully connected layer with a single node representing the age of a cluster in dex.

The network contains 19,641 trainable parameters and uses the \textit{relu} activation function (defined as $\text{max}(0, x)$ for all $x$).
Hyperparameter Optimisation

Following the model architecture selection, optimising the remaining parameters can now begin. As alluded, existing approaches cannot optimise the additional parameters created by the combination of observed and synthetic training data. The identified hyperparameters are assessed across the following values:

- sample_weight: [False, True]
- real_perc: [0.25, 0.5, 0.75, 0.99]
- syn_weight: [0.1, 0.5, 0.75, 0.99]
- normalise_by: [column, global]
- patience: [10, 20, 30]
- epochs: [100, 200, 300]
- early_stop: [False, True]

This relatively small distribution of values accounts for over 1,100 possible combinations meaning a grid search is impractical. A random search of parameters allows a wider search but is less likely to identify the absolute optimal setup. § 1.5.4 provides details on these techniques. Using the random search approach over 25 iterations sufficiently covers each parameter, allowing for the identification of their optimal values. Analysis with HiPlot identifies a strong correlation across all evaluation metrics, indicating the optimisation of just one metric is reflected in the other evaluation metrics.

Further, analysis reveals that allowing early stop and the number of epochs does not significantly affect the final performance, whereas the normalisation approach (normalize_by), the weighting of clusters (sample_weighting), the weight of synthetic clusters (syn_weight), percentage of real clusters (real_perc) and patience parameters all impact the final performance. Identifying the optimal parameters is not as simple as minimising the loss function due to the synthetic data within the training set. Using interactive plots, created using HiPlot (See § 1.5.4 Haziza et al. (2020)), with domain knowledge to ensure the optimisation remains on real, rather than synthetic, clusters. The final parameters selected are:

- sample_weight = True
- real_perc = 0.75
- syn_weight = 0.5
- normalise_by = column
- patience = 20
- epochs = 300
- early_stop = True
Further optimisation of these hyperparameters is difficult to verify due to the complexity of the training data, hence, these are the final hyperparameters used.

### 3.4.4 Ageing Network Performance

The challenges encountered while optimising hyperparameters persist while evaluating the final model’s performance. Figure 3.17 shows the distribution of predicted ages against their true age split by real and synthetic clusters. The model performs well on the synthetic data due to the clean data, known membership and larger training size. Through inspection, the predictions are accurate within 0.5 dex. However, the performance on synthetic data is reflecting the unrealistic dataset, with little nuance or contamination. Turning to the performance of the real clusters, it is evident there is much greater uncertainty. The prominent trend is that the accuracy is more reliable for older clusters, with a tendency to underestimate the age of older clusters (greater than $10^8$ years) and overestimate the age of younger clusters (less than $10^8$ years). The spread of ages is much greater than that observed in the synthetic data due to the lower quality of observations and contamination of clusters.

Aside from the handful ($\sim 5 – 10$) prominent outliers, the older clusters are reliable to at least 1 dex, whereas the age estimations of young clusters are less
reliable. Initially, the assumption is that the effect is a bias within the model. For this effect to be the result of the model, it must originate as an intrinsic behaviour of the data, an issue with the model itself or from within the MWSC. Investigating the MWSC, a model was trained with only synthetic data resulting in highly unreliable results with estimates, as far as three dex for some models. Unfortunately, the performance is inadequate to draw meaningful conclusions, however, it seems unlikely that the MWSC would be the source of these biases due to the results originating from multiple methodologies. The possibility that the model itself is biased at these ages is difficult to determine, yet, if this is the source, then the synthetic dataset would expect to present a similar pattern. Thus, a property within the clusters themselves must be intrinsically causing the overestimate in age.

To understand what may cause such behaviour let’s explore the features within the observations that determine the age. In studying the CAMD of a stellar population, the main sequence is the dominant feature, followed by the red giant branch. For under-populated clusters, any additional features, such as AGB or white dwarfs, are not well defined due to their lower populations at these timescales. Recall, from §1.2.2, that the lifetime of a star is inversely proportional to the mass, approximated by:

$$\tau \propto 10^{10} \left( \frac{M}{M_\odot} \right)^{-2.5},$$

where $\tau$ is the stellar lifetime, $M$ is the stellar mass, $M_\odot = 1$ solar mass and that the luminosity also increases with mass, approximately as:

$$L \propto M^{2.35},$$

where $L$ is the luminosity and $M$ is the stellar mass as before.

The combination of these relations indicates that luminosity and lifetime are intrinsically linked. As explained in detail within §1.2, a star does not significantly evolve while on the main sequence until it begins the shell-burning phase, evolving into a red giant star. Since stellar lifetimes are inversely proportional to their initial mass, hence proportional to luminosity, the absolute magnitude of stars leaving the main sequence indicates the cluster’s age. This point is known as the “main sequence turn-off.” In practice, stellar evolution models, such as PARSEC, provide detailed functions of these relations. Thus,
the main sequence turn-off point is likely a direct indicator of the cluster age for
the trained network. Linking this to the overestimation of young clusters’ ages
requires an understanding of stellar mass distributions. Salpeter (1955) defines
the Salpeter Initial Mass Function (IMF) as:

\[ \xi(m) \Delta m = \xi_0 \left( \frac{m}{M_\odot} \right)^{-2.35} \left( \frac{\Delta m}{M_\odot} \right), \]

for a \( \Delta m \) range of masses, where \( \xi_0 \) is the constant. This relation has since been
improved with Chabrier (2003) providing a more accurate IMF for stars below
1\( M_\odot \). A one solar mass star has a lifetime of roughly \( 10^{10} \) years, which is beyond
the observed lifetime of stellar clusters, meaning the Salpeter IMF is sufficient
for understanding the observed behaviour in Figure 3.17.

The IMF indicates that high-mass stars are much less common than mid to low-
mass. Since the main sequence turn-off point is a decisive indicator of a cluster’s
age, the likelihood of a young population containing a well-defined turn-off point
is much lower than an older cluster, due to the relative scarcity of high-mass
stars. Without this population, the age is difficult to determine with adequate
accuracy. The trend in Figure 3.17 suggests difficulty below \( 10^8 \) years and a
complete incapacity to estimate ages below \( 10^7 \) years. Evidently, the turn-off
point will be increasingly unpopulated at these young ages.

An additional complexity is the systematic effect of unresolved binary systems.
§ 5.7 of GCNS discusses the presence of unresolved binaries that bifurcate the
main sequence locus. The population of unresolved binaries raise the position
of the main sequence by 0.75 mag, see GCNS’s Figure 32. The presence and
effect of unresolved binaries are also discussed in Belokurov et al. (2020). Due
to the nature of deep learning, it is difficult to predict how this affects the age
estimates. But, raising the magnitude of the main sequence turn-off point in
observed clusters would lead to a tendency to underestimate the cluster ages.
Therefore, this effect may, at least partially, explain the observed underestimation
of older cluster ages. Comparing our work to the Auriga model for the MWSC
(Figure A1 in Kounkel et al. (2020)), they also under-perform for young clusters,
overestimating their ages, along with the tendency to underestimate the age of
older clusters.
3.4.5 Comparison To Recent Literature

The identification of stellar clusters in the Milky Way is an active field of study, leading to numerous results being available. Beyond KC19, recent literature represents a variety of approaches for the identification of stellar clusters, from Sim et al. (2019) using a visual inspection of 5-dimensional space to the more complex approaches of Castro-Ginard et al. (2020, 2022), He et al. (2022) relying on DBSCAN or Liu & Pang (2020) utilising the Friend of a Friend (FoF) algorithm Fu et al. (2010). However, these works apply a very simplistic filter on the data prior to clustering with cuts of various metrics in Gaia, notably in magnitude. As stated earlier, to adequately remove spurious data sources, a non-negligible fraction of reliable data is falsely removed. In particular, this approach is the least reliable for low-brightness objects, such as white dwarf stars.

Of the works highlighted, He et al. (2022) (He22 henceforth) represents the best comparison to this work due to having the largest overlap in feature space, while offering an ideal comparison with established methodologies. He22 uses DBSCAN to identify clusters to a distance of 1.2kpcs of the sun over the full sky and uses all stars with G-Band magnitude $> 18$. Of the 886 clusters identified, 676 lie within the feature space explored by this work. Identifying clusters from He22 also found in this work relies on matching member IDs. 57% of clusters in He22 are recovered when setting the threshold to 50% of the cluster members match. The proportion increases to 68% when the member match threshold is relaxed to 25%. Due to different clustering algorithms and parameters, contrasting approaches to identifying spurious sources and cluster validation, I do not expect this work to perfectly align with the cluster members or recover every cluster presented in He22.

Due to the differences in approaches, I consider all clusters with more than 25% members from He22 identified within a cluster in this work to be a match. This threshold may appear low, however, these works assume that all members originate from a common formation history, meaning the cluster age estimates will correlate. Figure 3.18 reveals a correlation between the cluster parameters from this work and He22. Firstly, exploring the number of members, it is evident the clusters in this work contain more members, implying that the utilisation of HDBSCAN returns a higher degree of completeness. This result is expected as HDBSCAN can handle clusters with various densities, whereas DBSCAN cannot. This effect also explains how this work recovers more than three times (2128 vs. 1188)
Figure 3.18  Comparison of the number of cluster members (left) and cluster age estimates (right) identified in this work with corresponding matches with He et al. (2022). This plot provides further evidence of an overestimation of cluster ages in this work and that the performance for young clusters is less reliable than for older clusters. Whereas the number of clusters suggests the utilisation of HDBSCAN increases the number of members recovered compared to DBSCAN in He et al. (2022).

676) the number of clusters in the same area, despite the higher minimum cluster size. Analysing the distributions, the clusters in this work contain significantly more members for those with less than 100 members in He22. Furthermore, most clusters in this work that do not correlate to a cluster in He22 contain less than 100 members, meaning they are likely below the minimum cluster threshold when using DBSCAN.

Moving onto the cluster ages, where the two approaches are significantly divergent, a correlation between the two works remains evident. Cluster ages in He22 are derived through isochrone fitting, whereas this work utilises a trained neural network. Similarly to the results for real clusters in Figure 3.17, we notice a weaker correlation in young clusters, whereas the performance is much stronger for cluster ages above 8.0 dex. However, this work consistently overestimates the cluster age by approximately 0.5 dex compared to the results from isochrone fitting. As observed in the plot of members within each cluster, this work contains significantly more members for clusters in He22 with less than 100 stars, possibly indicating the matching is less reliable. Thus, the red data points in the comparison of ages isolate these clusters, revealing no significant difference in the correlations of the least and most populous clusters.
The consistency in distributions assures that the disparity observed in cluster members for this population is a reflection of the more complete clusters identified in this work. Nonetheless, the comparison to He22 provides confidence that the complex machine-learning approaches used throughout this chapter are returning astrophysically meaningful results in line with recent literature.

3.5 Discussion

This work presents a pipeline to identify and verify stellar clusters and co-moving groups within the Milky Way Plane using astrometry and photometry measurements from the latest Gaia data release. A validation process ensures that the clusters originate from a single star formation event with a novel approach to counter the computational requirements of clustering algorithms at this scale.

The identification and ageing of clusters rely on accurate astrometric and photometric measurements. The method begins by deploying a neural network to identify spurious sources within the Gaia database (§ 3.2). This approach aims to identify measurements that do not correspond to a genuine astrophysical signal, such as measurements corresponding to more than one celestial body due to an overcrowding of the observed area. The model evaluates the population up to 10 kpcs, recovering the previously obscured populations across the CAMD.

The methodologies in this work are comparable to those within CSAS and GCNS. Nonetheless, by deploying state-of-the-art compute clusters, I utilise the distributed capabilities of PySpark, SparkML and the development of HEADSS to demonstrate the advantage of these new technologies without detrimental effects on the scientific results. This work achieves similar distributions of spurious vs reliable astrometric sources (see Figure 3.5 with greatly reduced computational requirements.

With a reliable dataset, § 3.3 describes the identification of stellar clusters and co-moving groups. By exploiting the ability of available clustering algorithms, namely HDBSCAN, and a novel partitioning approach to minimise the computational demands, § 3.3 identifies 2128 clusters within 1 kpc. With the implementation of alternative clustering methods and coordinate systems, the distribution and number of clusters are in line with those identified within Kounkel & Covey (2019). Despite having a clean dataset to 10 kpcs, the completeness at these distances and overall levels of noise makes identifying
clusters substantially less reliable. §3.3.3 explores the cluster validation process by a novel application of Hough transformations to train a random forest algorithm. Using metrics from the Hough transformation, the random forest evaluates the prominence of the main sequence. The validation process identifies 9.8% of all clusters to show insufficient astrophysical evidence of a single formation history. This level of contamination is also aligned with the estimated range in Kounkel & Covey (2019).

Finally, the age of validated clusters is estimated using a neural network. §3.4 outlines the model that evaluates cluster ages based on the photometry from Gaia and 2MASS. The training data contains a combination of previously observed and synthetic clusters, either from the Milky Way Stellar Clusters (MWSC) catalogue or sampled using PARSEC isochrone models. The inclusion of the model concludes the pipeline developed that has automated the process of identifying, validating and ageing stellar clusters from just an input catalogue of stars provided by Gaia only relying on additional photometry to improve the age estimates. Kounkel et al. (2020) presents the only other known attempt at ageing stellar clusters with machine learning. Their Auriga package, attempting to constrain several physical parameters beyond age, performs with similar accuracy. However, by focusing on age and correcting the effects of measurable parameters, this work does not rely on training data formed by multiple datasets. As estimating clusters ages is directly dependent of the stellar evolution model used, the reliance on a single catalogue provides more confidence in any conclusions that are derived from the age estimates presented in this chapter.

As the clusters identified in this chapter rely on a series of several machine learning algorithms, there is potential for small errors to propagate into a significant divergence from recent literature. He et al. (2022) represents a more established methodology, hence has been identified as an ideal work to compare with my results in §3.4.5. By identifying common clusters in the two works, I identify a correlation between both the number of members and the cluster ages. The correlation provides confidence in the results of this work but also identifies that the clusters contain significantly more members. This feature confirms the advantage of selecting HDBSCAN and the benefits of HEADSS to reduce computational demands. The comparison of cluster ages indicates an overestimation of approximately 0.5 dex in this work, which should be a consideration, yet the correlation reinforces the earlier confidence in these estimates.
3.5.1 Summary

In summary, this chapter presents the result of developing a series of machine learning models that result in a pipeline to identify stellar clusters within the Milky Way plane, validate the aforementioned clusters originate from a single formation history, and provide accurate age estimates requiring only astrometric and photometric data from Gaia. Although, additional photometric measurements, from WISE, improve the final age estimates. By providing this dataset of clusters, there are many scientifically compelling potential use cases, some of which form the discussion later in this thesis. Nevertheless, here the main findings of this chapter are summarised:

- Despite the overall high-quality data from Gaia, a machine learning model is required to identify the majority of objects with spurious measurements. This filter improves the identification of specific populations and removes a significant source of contamination while identifying stellar clusters.

- Utilising the clustering algorithm HDBSCAN through a novel implementation, the work identifies 2128 clusters within 1 kpc. The novel clustering approach to partition the feature space greatly reduces the computation demands. As this approach is applicable to algorithms and use cases beyond those explored in this work, the method is developed and generalised to allow a wider impact in Chapter 2.

- Validating the identified clusters originate from a single formation history is achieved by transforming a cluster’s CAMD into Hough space, efficiently identifying the main sequence. A random forest, trained on key metrics extracted from the Hough transformation, flags 9.8% of the identified clusters as non-physical or excessively contaminated. Any previous attempts to automate the validation of clusters are not known to the author, particularly one involving a Hough transformation.

- The ageing model developed to establish cluster ages successfully handles clusters older than $10^8$ years. Below this mark, it is suspected that the population around the main sequence turn-off point is insufficient to gain meaningful estimates for the somewhat underpopulated clusters identified.

- The key astrophysical result of this work is the catalogue of 1920 validated stellar clusters with age estimates. This catalogue allows the exploration of
several scientifically compelling use cases. Chapter 4 explores constraining the initial-final mass relation of stellar remnants utilising the clusters identified within this chapter.

- Comparisons with He et al. (2022) return a correlation between this work and the established methods in recent literature. The greatest improvement offered by this work returns a greater number of more complete clusters due to the utilisation of HDBSCAN. However, the Neural Network developed overestimates the cluster ages by approximately 0.5 dex compared to the more conventional isochrone fitting method.
Chapter 4

Constraining the Initial-Final Mass Relation of White Dwarf Stars

4.1 Introduction

Despite advancements in modelling capabilities, allowing ever increasingly complex models, stellar evolution remains a complex challenge. The final stages are challenging to model due to rapid evolution with high dependence on difficult-to-model processes, such as convection overshoot, dredge up, pulsations, and mass loss \(^\frac{1}{2}p.124\) Marigo & Girardi (2007); Pastorelli et al. (2019); Doherty et al. (2014).

Nonetheless, there are a number of popular models simulating these processes, e.g. PARSEC, MIST or YaPSI evolutionary tracks (Bressan et al., 2012; Choi et al., 2016; Spada et al., 2017). However, the most uncertain aspect of these models is the final stages. As visualised in Figure 4.1 these stages result in dramatic mass loss, as much as 85% of the total mass, through incredibly disruptive processes outlined in \(\frac{1}{2}p.126\). Through the study of white dwarf stars, it is possible to infer the total mass loss across the stellar lifetime from the mass during the main sequence stage (initial mass or \(M_{\text{initial}}\)) until the mass of the resultant white dwarf star (final mass or \(M_{\text{final}}\)). Using a sample of white dwarfs, with varying masses, an initial-final mass relation (IFMR) can be derived. Not only does a well-constrained IFMR significantly constrain the models, but also allows improved studies into the star formation of low to mid-mass stars through the
observation of the resultant white dwarfs. Furthermore, the IFMR is a crucial aspect of white dwarf cosmochronometry, a method to measure the age of stellar populations using the WD luminosity function (Wood & Oswalt, 1998; Mukadam et al., 2013).

Attempts to constrain the IFMR date back to Weidemann (1977), where the upper mass limit for white dwarf progenitor stars are first defined. The minimum mass required for a carbon core detonation supernova determines the maximum mass of the progenitor star. This limit currently stands at approximately \(8M_\odot\). Since then, the field has advanced dramatically using advancements in volume and accuracy of observational data. While obtaining the final mass is possible through spectroscopic observations (Bédard et al., 2017), the progenitor mass requires establishing the total and cooldown lifetime of the white dwarf. The cooldown age represents the time since white dwarf formation, estimated by cutting-edge cooldown models. The total age is the time since star formation. The difference between the two ages estimates the stellar lifetime, revealing the initial mass through the relations between stellar lifetime and mass. Cummings et al. (2018) (JC18 henceforth) presents the current leading IFMR, identifying 80 white dwarfs within 13 stellar clusters, where the cluster age defines the total lifetime. Zhao et al. (2012) presents an alternative approach, identifying 10 white dwarf-main sequence (WD-MS) wide-binary systems, with the chromospheric activity of the main sequence companion used to estimate the total age.

By building on the clusters identified in Chapter 3, this chapter also relies on
cluster ages to determine the total lifetime. Figure 4.2 summarises the full workflow including the steps within Chapter 3. This work aims to improve the semi-empirical IFMR by utilising the sheer volume of Gaia data available. By the nature of this work, the results are less precise than alternative methods which rely on precise follow-up studies. Nonetheless, with the increase in census and identification of white dwarfs from a larger sample of stellar clusters, the resultant IFMR represents a greater variety of total lifetimes and environments. Additionally, the methods discussed in this chapter are subject to continuous improvement by virtue of future data releases.

The structure of this chapter is as follows; § 4.2 delves into a method to identify white dwarf stars within such a large dataset as Gaia. § 4.3 presents the leading cooldown models within the current literature. Finally, in § 4.4 the contribution of this work to the Initial-Final Mass Relation is presented.
Accurate and reliable identification of white dwarf stars is a vital requirement of this chapter. White dwarfs are the dying remnants of stars below $8M_\odot$. For more massive stars, the electron degeneracy pressure is insufficient to oppose the gravitational forces, resulting in a neutron star or black hole, as described in §1.2.5. As an exposed inert core, the surface temperature is much higher than a main sequence star of similar mass. White dwarfs typically have a surface temperature between $\sim 6,000 - 30,000K$ but have been observed to reach as hot as $100,000K$ (Eisenstein et al., 2006). However, due to a $1M_\odot$ white dwarf radius comparable to Earth’s, the reduced surface area results in a low absolute magnitude, forming a distinct region of faint hot objects observed on a Hertzsprung-Russel Diagram, see Figure 1.5.

Due to their distinct observational properties (in colour and magnitude), it is reasonable to identify white dwarfs through these properties alone. Fusillo et al. (2021) (GF21 henceforth) offers the leading example in the current literature by relying on cuts in colour, magnitude, and quality flags to reduce the Gaia DR3
database that possesses a SDSS spectra. In GF21, white dwarfs are identified by their association with a locus on the CAMD. The authors acknowledge that this approach can only be complete for single white dwarfs, WD-WD binary systems or WD-MS binaries where the companion star is very low luminosity and does not dominate the observed colour of the system. This work continues from previous studies of Gaia DR1 and DR2 (Gentile Fusillo et al., 2015, 2019), by providing the probability of being a white dwarf and calculating various physical parameters including surface gravity, temperature and composition through analysis of the SDSS spectra available. An alternative approach is from Jiménez-Esteban et al. (2018), which adopts Monte Carlo techniques to create synthetic population models, however, this is less complete than methods using spectra. A machine learning approach for white dwarf identification appears in §5.8.1 of Gaia Collaboration et al. (2021) (GCNS henceforth) and describes a random forest algorithm to detect white dwarfs within 125pcs. GCNS identifies 250 candidates that do not appear in GF21, suggesting the initial cuts restrict the completeness of GF21, particularly for less bright examples.

This section presents a machine learning method to scale the identification of white dwarf stars in large catalogues to greater distances and independent of available spectra. The approach seeks to improve the completeness of white dwarf catalogues by not relying on spectra availability. This work only focuses on white dwarfs identification. Bergeron et al. (2019); Genest-Beaulieu & Bergeron (2019) present an approach to obtain physical properties, namely mass, where spectra are unavailable.

4.2.1 Training Optimisation

As with all machine learning approaches, success relies on a representative and reliable training set. Fortunately, both of these are available for the detection of white dwarfs. Introduced earlier, GF21 offers a high-quality and reliable training set of white dwarfs. A limitation of this training set is the selection of spurious sources. Therefore, for this work, all spurious sources identified in §3.2 are removed from the training set. Due to the sample returning less bright sources generally, the threshold for spurious clusters is relaxed to 0.5 to ensure sufficient completeness. Using the metric probability white dwarf ($P_{wd}$), as defined in GF21, the resulting training set contains 330,791 white dwarfs selected as those with $P_{wd} > 0.75$, and 188,681 non-white dwarfs selected
as those with $P_{wd} < 0.7$. The boundaries follow recommendations from the authors and establish a distinction between the two classes. Figure 4.3 shows the distribution on a CAMD of both training sets. Additionally, Figure 4.4 offers the spatial distribution, demonstrating consistent sky coverage between each dataset.

As a classification endeavour with minimal training features, this task lends itself well towards a random forest. Specifically, the RandomForestClassifier algorithm from PySpark ML is used, (Meng et al., 2016). Initial features are selected to reflect the knowledge that white dwarf stars are photometrically separate from other types, see Figure 1.5. Nonetheless, with various potentially informative photometric passbands and combinations, the final feature selection is determined by the feature importance of initial testing (See § 1.5.2) to select the most informative features and remove redundant ones. The features thought to be significant include:

- $\text{parallax}$
- $\text{phot}_g\_\text{mean}_\text{flux}$
- $\text{phot}_{bp}\_\text{mean}_\text{flux}$
- $\text{phot}_{rp}\_\text{mean}_\text{flux}$
- $\text{phot}_g\_\text{mean}_\text{mag}$
- $\text{phot}_{bp}\_\text{mean}_\text{mag}$
- $\text{phot}_{rp}\_\text{mean}_\text{mag}$
- $b_p\_r_p$
- $b_p\_g$
- $g\_r_p$
- $m\_g$

Photometric features include apparent magnitudes and flux measurements, with parallax measurements also present to provide distance information. Alternatively, the magnitude measurements could be converted to absolute magnitudes using the distance modulus, defined in § 1.1.1. Thus, the absolute G-band
magnitude (denoted as $m_g$) is also provided. Table A.5 defines all parameters inherited from Gaia.

The feature importance selection method determines which features contribute towards identifying white dwarf stars, described in §1.5.4. Those that return low importance scores ($<1\%$) in early models are removed until all features contribute a sufficient importance. The results reveal that the most impactful features are those that reflect colour, absolute G-band magnitude, and parallax as shown in Figure 4.5. The selection is consistent with those expected by the scientific understanding of these objects. Finally, the hyperparameters are optimised using the same CrossValidator as §3.2.2 resulting in:

- numTrees: 100
- impurity: “entropy”
- featureSubsetStrategy: “sqrt”
- minInstancesPerNode: 1

Table A.6 describes all selected parameters, and all other parameters remain the default values. The number of trees is limited to 100 as more trees provide minimal gains while increasing training computational and time requirements. A 80 : 20 split for training and validation datasets is selected.

### 4.2.2 White Dwarf Detection Performance

The resulting confusion matrix, Figure 4.6a, reveals the model performs at very high accuracy, with False Positives accounting for $\sim 1.95\%$ of the total
Figure 4.6  

(a) Confusion matrix from WD detection training sets. As with previous Confusion matrices in this thesis, the results are normalised by True Class to emphasise recall of the random forest.  

(b) CAMD reveals the regions of confusion within the results. False Positives are stars falsely classified as white dwarfs, likewise, False Negatives are the opposite.

Figure 4.7  

ROC curve for the identification of white dwarf stars. The performance identified in Figure 4.6a is reflected by the large area under the curve. The zoomed inset reveals that little performance can be gained by adjusting the classification thresholds.
white dwarf population. Despite this level of accuracy, using the location of inaccurate scores in Figure 4.6b, there is a distinct region of concentrated errors. This region lies along the boundary of the two populations, with the greatest concentration occurring towards the hot bright end of the cooldown track. This region represents the youngest white dwarf stars where the cooldown times are the most rapid explaining the lower population and steeper gradient. The uncertainty in this region highlights the limitation of this method which is likely improved by including additional passbands.

As in §3.2.3, thresholding must be explored for potential improvements in white dwarf identification. Figure 4.7 presents the resulting ROC curve for the trained model. The large area under the curve reflects the high performance eluded to by the confusion matrix in Figure 4.6a. The zoomed region reveals that the model is marginally more liberal towards false positives, however, the potential improvements are minimal. In fact, the potential improvements are less than the expected variation between subsets of data, therefore, adjusting the classification thresholds is not advised.

4.2.3 White Dwarf Detection for All Sources

With the hyperparameters and classification thresholds optimised, the trained model can now evaluate all sources observed by Gaia for their likelihood of being a white dwarf. Nevertheless, the model performance requires reliable photometric measurements. To ensure a reliable dataset all spurious sources identified in §3.2 are removed. Furthermore, due to the selection method in GF21, the model is only defined for a feature space that satisfies $G_{\text{abs}} > 6 + 5 \times (G_{BP} - G_{RP})$. The training data is not representative beyond this region. Therefore, the model is only applicable within this area. Despite this restraint, white dwarfs all satisfy this condition, so this limitation should not decrease the completeness of the result.

The resultant database contains 800,000 objects. The model identifies around 350,000 white dwarfs, corresponding to approximately 40% of the analysed population. The dataset is comparable to the size of the training dataset from GF21. Considering the dataset is used as the training data, the intersection of each dataset is misleading. Rather, analysing Figure 4.8 reveals this approach achieves better completeness for less bright objects, presumably due to the requirement for spectra not applying to this work. GCNS observes a similar
The method successfully identifies white dwarf stars within \textit{Gaia} that complement examples within the literature. Nonetheless, the model also confirms that the results within GF21 offer a reliable database with high recall for hotter white dwarfs. Concerning the IFMR, the cooler white dwarfs correspond to longer timescales due to being further down the cooldown track. Due to the increased lifetimes, it is less viable that the additional candidates identified remain within a gravitationally bound cluster, implying the initial mass will remain undetermined. Furthermore, the IFMR relies on accurate mass and composition estimates which require available spectra to meet the required accuracy. Due to these limitations and the limited number of new candidates, for the remainder of this chapter, the white dwarf population contains those identified by GF21 with a $P_{\text{wd}} > 0.75$. Within this dataset of over 350,000, 176 are associated with 103 clusters identified in Chapter 3.
4.3 White Dwarf Evolution

The degenerate core of white dwarfs has low opacity due to photon absorption requiring an electron to change its energy state. However, for this to happen, the higher energy state must be free, which is only true for a slim fraction of the electrons in a degenerate environment. Additionally, degenerate electron gases are good conductors of heat, meaning white dwarfs have a uniform heat distribution throughout the internal structure [Kutner (2016)]. Nonetheless, left over from the progenitor star is a layer of either mostly hydrogen or helium. This layer is not in a degenerate state, therefore, does not reach thermal equilibrium with the internal structure. Due to the lack of thermonuclear activity, white dwarf evolution is characterised by cooling, assuming they are undisturbed by neighbouring objects. Theoretically, white dwarfs cool until they no longer radiate energy, i.e. they reach thermal equilibrium with the surroundings, although the expected timeframes for this exceed the age of the universe. Otherwise, the cooling rate is primarily split into two phases, initially, a neutrino radiation-dominated phase which evolves into a thermal radiation phase.

Initially, when the core is at its hottest, white dwarf cooling is dominated by neutrino emission. The source of neutrino emission in white dwarf stars is the decay of the plasmons. This process has a luminosity, \( L_\nu \propto T^3 \), where \( T \) is the internal temperature [Kantor & Gusakov (2007)]. This form of cooling dominates when the internal temperature is very high, e.g. \( \sim 10^8 \) K, whereas, a typical surface temperature is \( \sim 10^4 \) K due to differences in composition and degeneracy. Cooling from neutrino emission is much faster because neutrino-matter interactions are extremely weak, resulting in an exceptionally low opacity [Bédard et al. (2020)].

Due to not reaching thermal equilibrium, the outer layer cools to a much lower temperature than the degenerate core. As radiative transfer dominates the second cooling phase, the rate of energy lost is equal to the luminosity, \( L \), given by:

\[
L = \sigma A T^4,
\]

where \( \sigma \) is the Stefan-Boltzmann constant, \( A \) is the surface area in \( m^2 \), and \( T \) is the surface temperature in Kelvin. As the luminosity depends on the cooler surface temperature, the cooling rate slows and continues to slow as the surface temperature reduces. The cooler outer layer effectively insulates the hot core.
explaining the vast cooldown ages for white dwarf stars, theorised to exceed the age of the universe.

In addition to the surface temperature, radiative cooling depends on the surface area. Generally, a more massive object will have a greater surface area. For example, the mass-radius relation for main sequence stars can be approximated by the power law $R \propto M^{0.7}$. Nevertheless, white dwarfs do not follow the same power law. Originating from the Pauli exclusion principle, which states that no two electrons can occupy the same energy state, electron degeneracy pressure opposes the gravitational forces in white dwarfs. Additional mass compresses the degenerate matter raising the electron’s energy states (by forcing them to the next unoccupied energy level). By modelling the core as a non-relativistic degenerate electron gas, the resulting mass-radius relation is $R \propto M^{-\frac{4}{3}}$. This relation states that a more massive white dwarf corresponds to a reduced radius. However, the limitation of this model is that as the mass increases, electrons occupying the highest energy states approach the speed of light. Hence, modelling of the core must assume a relativistic, degenerate electron gas.

Figure 4.9 presents a simplified model of the relativistic and non-relativistic white dwarf Mass-Radius relation, where relativistic effects dominate at higher masses, tending the radius towards zero at approximately $1.44M_{\odot}$. This limit is known as the Chandrasekhar mass, after the astronomer Subrahmanyan Chandrasekhar (Chandrasekhar, 1931, 1935), and represents the theoretical upper limit of white
Figure 4.10 Example cooldown curves for white dwarfs with hydrogen and helium-dominated atmospheres. The tracks refer to a range of masses, highlighting the importance of reliable mass estimates. 

**Top:** Cooldown curves for white dwarfs with hydrogen-dominated atmospheres, **Bottom:** Cooldown curves for white dwarfs with helium-dominated atmospheres.

dwarf masses. Pols (2011) provides derivations of the relations stated within this section and is referenced throughout this section. Returning to the effects of the cooldown, the inverse nature of the mass-radius relation further restricts the cooldown of more massive white dwarfs through radiative transfer.

Furthermore, as the white dwarf cools, a crystallisation process begins in the non-degenerate carbon and oxygen ions. This process releases binding energy in the form of heat, which affects the cooling curves. The temperature at which this process occurs depends on the composition but can delay the cooling process by about one billion years (Hansen & Liebert 2003; Tremblay et al. 2019). Considering that this is not a comprehensive list of events that determine the cooldown curve of white dwarf stars, modelling the evolution is beyond the scope of this work. Gladly, such models are available within the literature.
4.3.1 The Cooldown Models

Bédard et al. (2020) presents the leading cooldown models for white dwarfs with hydrogen and helium-dominated atmospheres, known as DA/DB type. Figure 4.10 visualises the cooldown curves, revealing the dependence on the atmospheric composition. To estimate an observed white dwarf’s cooldown age, the surface temperature and mass, or surface gravity (log($g$)), must be known.

The cooldown models are sampled for surface temperatures between 1250–110000 K with log$g$ between 7.0 – 9.0. The resultant grid provides adequate coverage to perform a 2D spline to determine the cooldown age of observed white dwarfs. Using tools within SciPy (Virtanen et al., 2020), cubic interpolations are used for the properties given in GF21. 150 of the 176 white dwarfs associated with clusters return estimates for cooldown age. The remaining 26 are missing data for the required properties in GF21, i.e. the surface gravity and surface temperature. Estimates rely on the $\chi^2$ fitting of the observed spectra to be either a DA or DB type, also provided by GF21.

A prominent issue surrounding this approach for determining the cooldown age is the question of unresolved binary stars. As described in GF21, a non-white dwarf companion star will dominate the observed signal, meaning that the system will remain unidentified due to the selection parameters. The alternative WD-WD binary systems are worth considering. As both members are white dwarfs, assuming no interaction such as mass transfer, the evolution should remain unaffected, as will the colour. Hence, the unresolved binarity of white dwarfs will have a minimal effect on the calculated cooldown age, although it does effect the observed mass and must be considered during the final results.

4.4 Initial-Final Mass Relation of White Dwarf Stars

4.4.1 Calculating Initial Masses

With the cooldown models of Bédard et al. (2020) providing estimates for the cooldown age of each white dwarf and the neural network in §3.4 providing an

\footnote{https://www.astro.umontreal.ca/~bergeron/CoolingModels/}
estimate for the total age of each cluster, the difference in these ages provides an estimate for the stellar lifetime. This method assumes that all members of the cluster originate from a common formation history, as validated in §3.3.3.

Identifying white dwarfs within stellar clusters allows an initial mass to be estimated from observational properties, marking the final parameter required to create the IFMR.

Converting the stellar lifetime to an initial mass requires turning to isochrone models. Sampling a range of initial masses, the time frames required to evolve into late-stage AGB stars provide the stellar lifetime-mass relation. Recall that the neural network in §3.4 is trained using a mix of synthetic and observed clusters. To some degree, these cluster parameters are derived from isochrone models, either directly, in the synthetic case, or fitted, in the observed clusters. Although any appropriate model is acceptable, to maintain consistency, the same PARSEC model from the earlier analysis is used to determine the initial mass. A sample of isochrones covering ages of $6.6 - 10.1$ dex in evenly spaced $0.1$ dex intervals, with a fixed initial metallicity of $Z = 0.014$, is sampled. The selected range covers the entire training set range in ages with a solar metallicity selected due to models described in Ekström et al. (2012). The stellar lifetimes for progenitor stars are derived using a one-dimensional spline across the models, as shown in Figure 4.11.

**Figure 4.11** The resulting 1-dimensional spline of PARSEC isochrones used to calculate the initial mass of the white dwarf progenitor star. The area shaded in grey roughly indicates the mass range for which the progenitor star is too massive to form a white dwarf star.
Errors in the initial mass estimates originate from the uncertainty in cluster ages, originating from the performance of the trained neural network, rather than the cooldown models due to the cooldown models demonstrating a greater accuracy. Due to the use of neural networks, each evaluated cluster does not have a measured error estimate. Instead, the spread of sub-sampled training clusters provides the best estimate. By focusing on those clusters with ages greater than 7.5 dex, the average error is approximately 0.2 dex. Using a fixed error estimate causes the initial mass estimates to be less reliable for those with a considerable disparity between the stellar lifetime and total cluster age. Within this dataset, such a disparity is more likely for white dwarfs with higher initial mass estimates.

### 4.4.2 The Initial-Final Mass Relation

The raw IFMR, shown in Figure 4.12, reveals that there appears to be a bifurcation of the population. Additionally, a group of the observed white dwarfs have masses that are too low to be present in this population. The timescales to produce white dwarfs below 0.5\(M_\odot\) exceeds the expected lifetime of stellar clusters within the Galactic plane. An explanation of these observed phenomena would be the presence of unresolved WD-WD binaries Kilic et al. (2018).
Figure 4.13  CAMD of the white dwarfs featured in Figure 4.12, with matching colour coding, with a heat-map of the Gaia white dwarf population in grey for reference. The distribution confirms that the bifurcation observed in the IFMR is reflected in the CAMD, suggesting unresolved binarity is the source. The red dashed line indicates the threshold for binary system classification in this work.

Unresolved binaries would affect the observations by appearing brighter. Due to the inverse mass-radius relation, Figure 4.9, the higher observed luminosity results in a lower mass estimate. Identifying these binary systems requires exploring the CAMD, as these systems will produce a secondary locus brighter than the central WD locus. A similar population is present along the main sequence, where a secondary locus appears alongside the main locus, with a 0.75 increase in magnitude, as shown in Figure 32 of Gaia Collaboration et al. (2021). Assuming that the systems are sufficiently wide binaries, such that they do not significantly impact each other’s evolution, the secondary locus of unresolved binaries will remain a similar distance from the central white dwarf locus. This bifurcation is also noted in El-Badry et al. (2018) where they suggest binary systems may be the cause but do not attempt to correct such systems.

Figure 4.13 visualises the identified white dwarfs in the IFMR, shown in Figure 4.12 plotted against the population within GF21. Continuing the colour coding from the IFMR, it is evident that those observed with uncharacteristically low mass tend to occupy the region above the central white dwarf locus. This
location corresponds to the zone expected to be occupied by unresolved binary systems.

To correctly address the effect of binarity, there are some assumptions. Firstly, all of the objects brighter than the threshold line ($y < -1.15x^2 + 5.43x + 11.45$, where $y$ is the absolute G-magnitude and $x$ is the G-RP magnitude) are indeed unresolved binaries. This threshold is calculated by a line of best fit to the entire population, offset by 0.37 magnitude to correlate to the midpoint between the bulk and the increase due to a doubling of brightness. Secondly, without significant additional data, the best assumption is that these systems are equal mass binaries Hogeveen (1992); Andrews et al. (2015). Therefore, those systems will appear twice as bright, resulting in an $\sqrt{2}$ reduction in radius (surface area = $4\pi R^2$ for a sphere). Using the relativistic mass-radius relation, shown in Figure 4.9, the mass of each member is scaled. Figure 4.14 presents the corrected IFMR for 90 white dwarf stars, with the scaled binary systems indicated. In this version, a considerable reduction in the bifurcation of the population is evident, and the results lie broadly in line with those presented in Cummings et al. (2018), albeit with the expected greater errors resulting in a greater spread of results.

To analyse the details, the IFMR is split into three distinct populations, low, medium and high-mass progenitor stars. Low mass white dwarfs are those with an initial mass below approximately $3M_\odot$, medium mass as those with an initial mass above $3M_\odot$, but below $5M_\odot$, and high mass as those with an initial mass above $5M_\odot$.

The Low-Mass IFMR

Containing 27 DA and 21 DB stars, the low mass population accounts for 50% of the total population. This work returns a much higher population in this region than in previous studies. There is a distinct increase in the spread of this population, indicating less correlation than the rest of the IFMR, centred on an initial mass of $2.5M_\odot$. JC18 also observed a lack of correlation for low-mass stars. The low mass population in JC18 is split into two groups, with one centred at $1M_\odot$ and another at $2M_\odot$. The lower mass population is not present in this work due to the stellar ages exceeding the expected lifetimes of the clusters within the Milky Way plane. As for the second population, there is agreement within the observed errors between this work and this population. Accounting for the general
Figure 4.14  IFMR split into DA and DB-type stars with corrected unresolved binary systems indicated. Median errors are presented for clarity, with lighter colours indicating a lack of error upper initial mass estimates due to the estimated cooldown age exceeding the lower limit of the cluster age. Background IFMR (grey crosses) with corresponding 3 piece fit (black dashed line) for reference from Cummings et al. (2018). Theoretical IFMR from Choi et al. (2016) is also shown (blue dashed line). Vertical lines indicate the boundary’s selected for the low, mid, and high mass populations discussed below.
<table>
<thead>
<tr>
<th>dr3_source_id</th>
<th>(M_{\text{ini}} (M_\odot))</th>
<th>(M_{\text{final}} (M_\odot))</th>
<th>Type</th>
<th>Binary?</th>
</tr>
</thead>
<tbody>
<tr>
<td>66697547870378368</td>
<td>3.408149</td>
<td>1.087786</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>119677790531181056</td>
<td>6.680132</td>
<td>0.613956</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>192780809672948096</td>
<td>2.420836</td>
<td>0.811862</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>19404965038502464</td>
<td>2.468703</td>
<td>0.767579</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>194103350362955520</td>
<td>2.638367</td>
<td>0.673280</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>194235429196028032</td>
<td>3.092276</td>
<td>0.781677</td>
<td>DA</td>
<td>True</td>
</tr>
<tr>
<td>208870822314832256</td>
<td>2.458807</td>
<td>1.170213</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>244003693457188608</td>
<td>5.168726</td>
<td>1.087200</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>250116630209371520</td>
<td>4.003255</td>
<td>0.679995</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>260666929614243840</td>
<td>6.768720</td>
<td>0.817559</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>262361788128063488</td>
<td>2.402913</td>
<td>0.997212</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>2623931929901696</td>
<td>2.843655</td>
<td>0.796176</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>269008576436255872</td>
<td>2.671051</td>
<td>0.495765</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>287058228037678976</td>
<td>4.435130</td>
<td>0.628582</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>334175565462472960</td>
<td>2.812114</td>
<td>0.648671</td>
<td>DA</td>
<td>True</td>
</tr>
<tr>
<td>34247935215348160</td>
<td>2.482583</td>
<td>0.973893</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>34252364652426368</td>
<td>2.433551</td>
<td>0.461093</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>39628763445638848</td>
<td>2.778547</td>
<td>1.086202</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>4273497312800848</td>
<td>2.708182</td>
<td>0.805892</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>435725089313589376</td>
<td>4.416282</td>
<td>1.028864</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>44155455417487232</td>
<td>2.581138</td>
<td>0.824130</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>451811867843759616</td>
<td>2.738901</td>
<td>0.770443</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>45220597404300064</td>
<td>2.425502</td>
<td>0.567164</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>45808289805529472</td>
<td>3.010258</td>
<td>1.060983</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>45877892757347168</td>
<td>3.136423</td>
<td>0.932264</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>469401408186159744</td>
<td>2.797429</td>
<td>0.702134</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>481116601501789952</td>
<td>2.986328</td>
<td>0.962934</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>482831942720970624</td>
<td>3.046877</td>
<td>0.758893</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>483864491516348416</td>
<td>2.425022</td>
<td>0.597301</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>494588715616761856</td>
<td>2.662488</td>
<td>0.810374</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>497069591803837440</td>
<td>3.272351</td>
<td>0.946339</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>dr3_source_id</td>
<td>M_init (M_☉)</td>
<td>M_final (M_☉)</td>
<td>Type</td>
<td>Binary?</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>---------------</td>
<td>------</td>
<td>---------</td>
</tr>
<tr>
<td>506862078583709056</td>
<td>2.957126</td>
<td>0.973730</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>507054806657042944</td>
<td>2.954227</td>
<td>1.107400</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>507105143670906624</td>
<td>4.091605</td>
<td>0.871318</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>507128332197081344</td>
<td>3.976839</td>
<td>0.924578</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>507221863701989248</td>
<td>5.283317</td>
<td>0.939773</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>507277870080186624</td>
<td>3.274946</td>
<td>0.915830</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>507312161094123392</td>
<td>3.644611</td>
<td>0.844737</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>50736201277541552</td>
<td>2.890149</td>
<td>0.908988</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>507414067782288896</td>
<td>2.802444</td>
<td>0.830197</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>507555904779576064</td>
<td>2.960006</td>
<td>0.844170</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>507899399087944320</td>
<td>4.622093</td>
<td>0.780344</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>511159317926025600</td>
<td>3.751200</td>
<td>0.773756</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>53444968861249056</td>
<td>4.104940</td>
<td>0.558504</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>547682581635128576</td>
<td>2.356054</td>
<td>0.687308</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>564142408141685120</td>
<td>2.740128</td>
<td>0.792514</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>564799989112513536</td>
<td>3.088655</td>
<td>0.768047</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>602672216156639488</td>
<td>6.156138</td>
<td>0.407068</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>654999486351234432</td>
<td>3.129534</td>
<td>0.646050</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>662998983199288032</td>
<td>2.744051</td>
<td>0.793849</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>673157439847406080</td>
<td>3.315819</td>
<td>0.753007</td>
<td>DA</td>
<td>True</td>
</tr>
<tr>
<td>865551143842417536</td>
<td>2.933584</td>
<td>0.474734</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>938594982022602224</td>
<td>2.685107</td>
<td>0.835408</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>941610148733039104</td>
<td>3.845480</td>
<td>0.601788</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>942029642481986048</td>
<td>2.616424</td>
<td>1.195687</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>962203176887102208</td>
<td>2.977926</td>
<td>0.836993</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>980134317454471680</td>
<td>2.476586</td>
<td>0.876274</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>986561856272291968</td>
<td>5.155876</td>
<td>0.712664</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>991944790323171968</td>
<td>3.199474</td>
<td>0.570344</td>
<td>DA</td>
<td>True</td>
</tr>
</tbody>
</table>
Table 4.1

<table>
<thead>
<tr>
<th>dr3_source_id</th>
<th>M_ini ( (M_\odot) )</th>
<th>M_final ( (M_\odot) )</th>
<th>Type</th>
<th>Binary?</th>
</tr>
</thead>
<tbody>
<tr>
<td>993288019870226304</td>
<td>3.583334</td>
<td>0.789540</td>
<td>DA</td>
<td>True</td>
</tr>
<tr>
<td>99604952173667712</td>
<td>2.958566</td>
<td>0.588642</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>1002498693201406336</td>
<td>8.294896</td>
<td>0.913262</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>1004850376774050432</td>
<td>3.580408</td>
<td>0.779384</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>1107525830592772608</td>
<td>3.746239</td>
<td>0.851558</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>1755387012581031936</td>
<td>4.531718</td>
<td>0.889103</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>1764788902156456704</td>
<td>3.824128</td>
<td>0.813740</td>
<td>DA</td>
<td>True</td>
</tr>
<tr>
<td>1803140554930981120</td>
<td>4.510330</td>
<td>0.758835</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>180422879655313920</td>
<td>2.015740</td>
<td>0.515788</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>1842064164196699136</td>
<td>3.039487</td>
<td>0.945539</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>191462948787181312</td>
<td>3.984829</td>
<td>0.798292</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>195157448723149248</td>
<td>3.300115</td>
<td>0.559144</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>1957448284504262912</td>
<td>3.183657</td>
<td>0.735182</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>1962167835086179840</td>
<td>3.584902</td>
<td>0.558709</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>1973010153610517760</td>
<td>2.791167</td>
<td>1.018396</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>1989303335384960768</td>
<td>2.529365</td>
<td>0.918564</td>
<td>DA</td>
<td>True</td>
</tr>
<tr>
<td>2012912117779224240</td>
<td>2.961311</td>
<td>1.051077</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>2073285549656153600</td>
<td>4.175674</td>
<td>0.805492</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>2138558740321558016</td>
<td>2.329404</td>
<td>0.471770</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>2170776080281869056</td>
<td>4.143807</td>
<td>0.954969</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>218149827716067888</td>
<td>2.354757</td>
<td>0.667056</td>
<td>DB</td>
<td>True</td>
</tr>
<tr>
<td>2272159302996746240</td>
<td>3.175614</td>
<td>0.757160</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>2879428195013510144</td>
<td>2.353810</td>
<td>0.706107</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>3049577928650396672</td>
<td>2.734060</td>
<td>0.520160</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>3089161755999033344</td>
<td>2.401379</td>
<td>0.374356</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>332421847522997760</td>
<td>2.727749</td>
<td>0.898280</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>34577882892312132</td>
<td>2.855271</td>
<td>0.76528</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>4229037383344579072</td>
<td>2.206013</td>
<td>0.826882</td>
<td>DA</td>
<td>True</td>
</tr>
<tr>
<td>543348313670068336</td>
<td>2.737441</td>
<td>0.634669</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>5437085789626872832</td>
<td>2.135169</td>
<td>0.592740</td>
<td>DB</td>
<td></td>
</tr>
<tr>
<td>6242539938863247360</td>
<td>2.350191</td>
<td>0.868228</td>
<td>DB</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.1

(1) - source_id from Gaia DR3
(2) - Progenitor mass (aka Initial mass)
(3) - Observed white dwarf mass (Final mass)
(4) - Type of white dwarf (DA/DB)
(5) - Suspected unresolved binary flag

Final results for the IFMR of white dwarf stars. Those flagged as suspected binary systems contain the corrected final estimates and all required observational properties can be found in [Fusillo et al. (2021); Hambly et al. (2021)].
underestimation of cluster ages from the neural network in Chapter 3, adjusting ages by 0.20 dex would further align the two IFMR populations. However, no such adjustments are made.

The agreement between studies suggests there is an astrophysical explanation for this lack of correlation. These stars relate to the longest stellar lifetimes, which are the least sensitive to age estimate errors in either the cluster or cooldown age. In theory, these are the most reliable results on the IFMR. JC18 suggest that, as their sample is from a limited number of clusters, the lack of resolution in cooldown ages of bright young white dwarfs causes the lack of resolution at the lowest masses. Yet, this hypothesis does not hold for this work, as multiple white dwarfs rarely occupy the same cluster. As final mass estimates remain large, the observed lack of correlation at low initial masses may be non-physical, but such an assumption risks ignoring a genuine astrophysical phenomenon.

To observe this trend, either the network used to calculate cluster ages is significantly overestimating the cluster ages of these stars, or the cooling lifetime is longer than suggested by the cooling curve for these objects. The evidence suggests that the cluster ageing model developed in §3.4 underestimates cluster ages, particularly for the longer lifetime of low-mass stars. The cooldown models assume that white dwarfs remain inert and do not have an internal energy source. Recent studies, Chen et al. (2021, 2022), have challenged this assumption by identifying an overabundance of hot white dwarfs within the globular cluster M13, slowing the cooling process of approximately 70% of the population by about one billion years. The authors suggest that this is evidence of stable hydrogen burning, achieved by missing the Third Dredge-Up within the Horizontal Branch evolutionary phase, where the majority of the remaining shell is depleted. If this is the mechanism causing the overabundance of high-mass white dwarfs associated with the low-mass progenitor stars, this work suggests this phenomenon also occurs for helium-dominated atmospheres, a previously unobserved result.

Such a claim has far-reaching implications for the study of white dwarfs, meaning it must be considered that non-physical explanations may also exist. It may be the case that this sample is falsely associated with their “host” cluster or that the age estimates are unreliable in these cases. Considering this effect is not observed in other studies of the IFMR, this is a possibility. However, as shown in Chen et al. (2021), the presence of delayed cooling occurs in some clusters and not others. By exploring a range of clusters, this work may be increasing the likelihood of selecting such populations. Additionally, not rejecting samples
based on their spectra during the initial filters, such as in [Cummings et al. (2018)],
could explain why this phenomenon has remained unknown. Evidently, further
study of these populations is needed to confirm if this is a physical property before
major conclusions are made.

Furthermore, slowing the cooldown process accounts for the population lying
above the bulk of JC data, with a higher-than-expected final mass population
observed within the low initial mass region. Correcting the unresolved binaries
has brought this population (particularly for DB stars) closer in line with those
in JC18. Nonetheless, a small population remains below the bulk of JC18 data.

### The Mid-Mass IFMR

Containing 14 DA and 21 DB stars, the mid-mass population accounts for
approximately 40% of the total population. After correcting the unresolved
binarity, this population is in agreement with JC18. To confidently constrain
the gradient of this region requires a reliable low-mass fit, which is difficult to
obtain with a population with potentially active fusion. Due to this population,
the fitting is likely unreliable. This highlights that this work is a demonstration
of the challenges that come with machine learning analysis and the pipeline is
suitable to adapt to Gaia DR4, offering a path of continual improvement.

As mentioned earlier, the results obtained in this work return higher error
estimates than JC18, which generally underestimates the initial mass compared
to theoretical models. Additionally, the ageing process tends to underestimate
the cluster ages leading to higher initial mass estimates. These factors explain the
disparity between this work and the literature. Overall, excluding the unusual
high-final mass white dwarfs with low initial mass estimates, the two studies show
good agreement between the low and mid-mass populations. This consistency
suggests that combining the results of the two studies enables further constrains
the IFMR for white dwarf stars with initial masses below 5\(M_\odot\).

### 4.4.3 The High-Mass IFMR

Containing just 4 DA and 3 DB stars, the high-mass population accounts for 10% of
the total population. Due to the short stellar lifetimes, this region reflects the
objects with the largest errors. High-mass stars have a stellar lifetime of up to
approximately $10^8$ years. As the majority of clusters return a cluster lifetime of at least $10^{8.5}$ years, an underestimate of 0.1 dex in cluster age accounts for the entire stellar lifetime. This example demonstrates why the initial mass estimates of high-mass white dwarfs can be so unreliable due to the disparity of ages probed. To accurately evaluate high-mass stars extremely accurate ages must be obtained or younger clusters identified to reduce the disparity in the cluster and stellar ages. Unfortunately, the ageing model is unreliable for suitably young clusters, meaning alternative approaches are required for this population.

Due to this effect, I expect many white dwarfs associated with a cluster but not present on the IFMR presented to correspond to shorter stellar lifetimes within older clusters. As shown, this combination requires extremely accurate cluster ages and the tendency to underestimate the ages within this work worsens the issue. The high mass population that does feature in Figure 4.14 is the least reliable with initial mass errors spanning several (2 – 3) solar masses, therefore, does not add much value to constraining the region of the IFMR above $6M_\odot$.

### 4.5 Summary & Conclusions

An astrophysical use case for the clusters identified using a scalable machine-learning pipeline is presented in this chapter. By continuing with the same principles, this work successfully produces a population of white dwarf stars that further constrain the observed Initial-Final Mass Relation (IFMR). The population of white dwarfs covers an initial mass range of $2.0 - 8.1M_\odot$ and represents the first IFMR created by an open search rather than targeting specific clusters made possible by the accuracy of \emph{Gaia}. Comparing the results to the recent literature, they lie broadly in line with the analysis of Cummings et al. (2018) despite the larger errors. Furthermore, compared to theoretical models in Choi et al. (2016); Marigo & Girardi (2007), this work indicates that the low initial mass population is more strongly correlated to the initial mass than related works. By investigating the potential source of the major discrepancies, we have identified avenues for future research and understand the potential improvements offered by future surveys offering even greater accuracy. Here, the main findings are summarised.

- Using a random forest algorithm, the white dwarf population provided by Fusillo et al. (2021) is validated to return a high degree of completeness (up
to 1 kpc), particularly at the brighter end of the cooldown track.

- Maintaining the scalable philosophy, the pipeline identifying, validating and providing age estimates of stellar clusters has been extended to constrain the IFMR of white dwarf stars using the largest number of stellar clusters to date. Identifying 90 members from within the Milky Way plane, presented in [11] this work represents the largest population of white dwarfs stars with initial mass estimates compared previous leading works such as Zhao et al. (2012); Andrews et al. (2015); El-Badry et al. (2018); Cummings et al. (2018).

- Identified by the bifurcation of the initial IFMR, a number of likely unresolved WD-WD binaries are identified and corrected using the theoretical mass-radius relation. These results successfully bring the observed systems within a reasonable range and represent the first attempt at including unresolved WD-WD binary systems in an IFMR.

- A population of white dwarfs return a non-physically low initial mass estimates that correspond to a reduced cooling rate. A potential explanation is that these stars have experienced active fusion within the core, supporting the recent results of Chen et al. (2021, 2022). This work identifies indications that fusion has occurred in both DA and DB-type stars, a previously undiscovered phenomenon.

- The results of this work are promising and confirm that relying on machine learning models can successfully recover astrophysical results and potentially discover previously unknown phenomena. The methods discussed are transferable to future data releases and telescopes, meaning these results act as a proof of concept ready for considerable improvement as greater volumes of increasingly accurate data are released.
Chapter 5

Outlier Detection

5.1 Introduction

Recent, current, and upcoming sky surveys are transforming the field of time-domain astronomy by producing light curves for millions of astronomical objects, providing constraints to cosmological models, enabling the discovery of unexpected phenomena, and challenging our data storage and processing capabilities. The Sloan Digital Sky Survey (SDSS) has provided insight into the large scale of the universe by providing millions of images and spectra of approximately one-third of the entire sky \cite{York2000}, and is about to become the most extensive time-domain spectral variability survey with the advent of SDSS-V \cite{Kollmeier2019}. As a prelude to the eagerly anticipated Vera C. Rubin Telescope’s Legacy Survey of Space and Time (LSST), which promises to be a generational step in the observations of the transient universe, the Zwicky Transient Facility (ZTF) is currently surveying the northern sky at a pace of more than 3750 square degrees per hour, discovering energetic transients every single night \cite{Bellm2014}.

Space-based surveys have focused on stellar targets. The KEPLER/K2 mission observed more than 500,000 stellar objects across nine years of operation, increasing the number of confirmed transiting exoplanets from hundreds to just under three thousand. Additionally, providing valuable information about intrinsic stellar variability \cite{Overbye2018}. The next-generation exoplanet finding sky survey came online in April 2018, following the launch of the
Transiting Exoplanet Sky Survey (TESS), with the primary goal of continuing the search for exoplanets, now focusing on smaller, rockier worlds in a much larger area of the sky with respect to Kepler and looking at comparatively brighter stars.

These and other similar surveys are producing an unprecedented onslaught of time-domain data. Yet, the challenge remains to effectively analyse the resulting plethora of data to accurately classify sources, select the most promising objects for spectroscopic follow-ups, and identify anomalies that require new theoretical frameworks, expanding our knowledge horizons. The latter is of particular interest within the exploration approach to astronomy research, in which we look for the unknown unknowns by expanding the parameter space of observables, or by dissecting existing datasets in novel ways to find true astrophysical anomalies.

A rapidly increasing number of publications deals with the problem of anomaly detection in astronomical datasets, and in time-domain datasets in particular astro-nomaly (Lochner & Bassett, 2021), while others include; Giles & Walkowicz (2019); Margalef-Bentabol et al. (2020); Doorenbos et al. (2021); Škoda et al. (2020); Baron & Poznanski (2017). Anomaly detection is relevant because finding anomalous objects is a direct avenue to discovery, particularly when those discoveries challenge existing models, motivating the formulation of new paradigms. A recurring theme of this research relates to the very definition of what constitutes an anomaly. In general, the anomalous nature of a given object in a large dataset depends on both the features used to represent the objects and the method used to quantify the anomalies, and crucially, it also depends on the specific domain knowledge in a particular field, in that not all rare objects represent astrophysical anomalies from a purely scientific point of view.

In Martínez-Galarza et al. (2021) (MG21 hereafter), we have provided a plausible definition for anomalous behaviour in the context of Kepler light curves: given a representation constructed by the light curve points (fluxes or magnitudes at each time snapshot) and its power spectrum (Lomb-Scargle periodogram). An anomalous light curve has features that appear to be sampled from a different distribution in the multi-dimensional space of those features compared to the bulk of the data. This definition is based on the results of applying a particular anomaly detection method (an adaptation of the Unsupervised Random Forest described in Baron & Poznanski (2017)) to the Kepler light curves. We have demonstrated that the method successfully identifies bona-fide anomalies, including Boyajian’s star, as well as a variety of pulsators including δ-Scuti stars,
RR Lyrae stars, long period variables, among other types.

To better understand anomalous variability in large time-domain surveys, one might ask if the anomalous nature of a given object is not only a function of the features and the method but also of the physical properties embedded in the dataset itself. Specifically, given a set of similar features from two similar datasets, are the anomalies similar and do they reveal any information about the physical properties of the objects in the dataset? For example, are RR Lyrae stars with larger periods more or less anomalous than their more rapid counterparts?

One possible way to answer this question, in the context of regular, well-sampled light curves, is by applying the same anomaly detection method to both *Kepler* and *TESS* light curves and investigating the types of anomalies recovered in each case. By doing so, one can learn about the physical properties that result in anomalous variability in each case. In addition, by studying the properties of anomalous light curves, regardless of whether they belong to a known class, one can identify the relationship between the anomaly score and specific physical properties of the system.

In this chapter, we apply the MG21 anomaly detection method to a set of SPOC-reduced *TESS* light curves observed between July 2018 and May 2020 in the *TESS* archive. Based on the results, we do the following: i) We provide a catalogue of unreported anomalies for each of the *TESS* sectors included in the analysis, classify them according to their variability trends, and provide anomaly statistics; ii) We investigate how the anomaly score relates to basic stellar parameters given by the location of the objects in the Hertzsprung-Russell (HR) diagram and how it relates to specific parameters of the system for anomalies of known class; iii) We investigate the prevalence of instrumental and analysis artefacts in our list of anomalies; iv) finally, we provide a discussion on how the overall properties of the population affect the selection of anomalies by comparing the anomalies found in this work versus those found in MG21 for *Kepler* light curves.

In §5.2, we investigate in detail the selected datasets and further information we use to analyse the results. §5.3 describes the fundamentals of the algorithm and the improvements made during this research. We also discuss how this is implemented with the selected databases. §5.4 explores what is identified as an anomalous object by the algorithm and the implications of this. §5.5 explains the patterns identified within the data, linking this to astrophysical classes through the combination of anomaly score with *Gaia* data to create a colour-coded Colour-Magnitude Diagram. §5.6 explores the astrophysical process in detail behind the
patterns identified. Finally, § 5.8 concludes our results and the future potential of the algorithm presented.

5.2 Datasets

The two datasets that enable this research are presented in this section. The first dataset consists of the TESS light curves, reduced using the TESS Science Processing Operations Center (SPOC) pipeline. The second dataset consists of Gaia DR2 photometric and astrometric data for TESS, where available.

5.2.1 TESS light curves

TESS has an ambitious science case, with exoplanet detection being the main scientific goal. But many other astrophysical phenomena can be studied using the sensitivity and cadence of TESS light curves, including astroseismology (Handberg et al., 2021), the dynamics of binary and multiple stars (Justesen & Albrecht, 2021), stellar rotational dynamics (Martins et al., 2020), and magnetic activity on the surface of main sequence stars (Cunha et al., 2019).

While both Kepler and TESS have similar science objectives, their input catalogue of targets differ in several ways. TESS observes 85% of the entire sky, an area that is 400 times larger than the Kepler field. It aims at stars that are on average 10 times closer (and 30 to 100 times brighter) than those observed by Kepler. To enhance the probability of observing small planets, the TESS 2-minute cadence targets are specifically selected to be bright, cool dwarf stars (Stassun et al., 2018). This implies that, on average, the TESS targets analysed here are cooler, older, and less massive than the Kepler targets, with populations of giants, such as the red clump, being much less prominent in TESS, and white dwarfs more represented in comparison with Kepler (Berger et al., 2020). These astrophysical differences in the target list impact the anomaly detection results, as is later demonstrated in this chapter.

TESS uses four identical cameras, each with four $2k \times 2k$ CCDs with a pixel scale of 21 arcseconds, resulting in a total field of view of $24 \times 24$ degrees. The design results in the simultaneous observation of a $24 \times 90$ degree stripe that is scanned in steps over the sky to provide coverage of almost the entire celestial
sphere, as seen in Figure 5.1. Each observation stripe constitutes a sector and is typically observed repeatedly over approximately 27 days, producing full-frame images with a cadence of 30 minutes. Additionally, 2-minute cadence light curves are obtained for a subset of brighter and typically nearby stars. The TESS light curves are processed by the Science Processing Operations Center (SPOC), including a pre-processing pipeline that corrects for systematics and performs photometric corrections. We use the light curves processed with the Presearch Data Conditioning (PDC) algorithm described in Fausnaugh (2018). We further re-process the PDC light curves to remove single-pixel spikes unlikely to be of astrophysical origin and normalize the light curves by dividing the fluxes by the mean value.

Each TESS sector contains about 20,000 PDC light curves with a 2-minute cadence. Objects located in the overlapping zones between sectors near the ecliptic poles, contribute more than one light curve to the dataset at different epochs. The TESS database contains 2-minute cadence light curves associated with over 225,000 individual stars. For this work, we analyse the first 24 sectors, corresponding to approximately two years of observations. In each sector, there are gaps in the light curves where the spacecraft is offline for maintenance, with the largest intermissions caused by the inability to observe during data
transmissions (Fausnaugh, 2018). These gaps typically span from one to a few days. In comparison with a Kepler quarter, the total observation time in a TESS sector is roughly three times longer (90 vs 27 days), the cadence is 15 times shorter (2 vs 30 minutes) and the number of targets is much smaller (20,000 vs 167,000), which is reflected in the data volume of a TESS sector being about half of a single Kepler quarter (5 − 6GB versus 11GB).

### 5.2.2 Gaia Photometry and Astrometry

The *Gaia* data release 2 (DR2) consists primarily of astrometric (parallax and proper motions) and photometric (G, BP and RP bands) data for over 1 billion stars (Marrese, P. M. et al., 2019), or about 1% of the total Milky Way stellar population (Perryman et al., 2001). The *Gaia* all-sky survey provides photometric measurements for the majority of both *Kepler* and *TESS* databases.

From the *Gaia* DR2 catalogue, we collect the following values for matching TIC sources: *phot\_G\_mean\_mag* (92.5% matching success with the TIC), *bp\_rp* (90.3%) and *parallax* (90.0%). The absolute *G* − band magnitude is obtained from the apparent magnitude and the parallax-estimated distance to each source. Details about the *Gaia* DR2 observations can be found in Gaia Collaboration et al. (2018a).

### 5.3 Anomalous Time Series Detection Algorithm

We perform anomaly identification using the Unsupervised Random Forest (URF) algorithm, first used in astronomical datasets in Baron & Poznanski (2017) in the context of SDSS galaxy spectra, and previously explored by the authors in the context of *Kepler* data in MG21. Given a set of features from the light curves the URF method assigns anomaly scores in two steps. First, using the data features as inputs, a random forest is trained to identify the original dataset from a synthetic dataset constructed by sampling the marginal distribution of the original features. Secondly, the population of terminal nodes is analysed for the original dataset to calculate an anomaly metric for each object. The origin of the anomaly score lies in Baron & Poznanski (2017), but for this work, we use the modified version in MG21. The anomaly score is calculated by analysing the populations of each terminal node and creating the similarity score as the average
Figure 5.2 Example real and synthetic datasets for both photometric flux and periodicity data. The non-linear sampling periods can be seen along the periodogram. **Top Left:** Example of an arbitrary real light curve. **Bottom Left:** An example of an arbitrary synthetic light curve, notice the δ-like spikes caused by the random selection of data from many separate light curves. **Top Right:** Example of a real periodogram for the light curve displayed Top Left. **Bottom Right:** An example of a Synthetic periodogram again selected at random. Notice the lack of trends within the synthetic datasets.

fraction of the total population in a given object’s terminal node across all trees. The final *Anomaly score* is defined as: \(1 - S\), where \(S\) is the similarity score. The normalised anomaly score, therefore, reflects how many similar objects exist within the data set. An anomaly score of 1 signals that the object is the sole occupant of its terminal node across all trees, while a score of 0 indicates the entire dataset occupies the same terminal node across all trees.

We now describe how the light curve features that act as inputs for the URF are extracted and the selection of the URF hyperparameters.

### 5.3.1 Feature Extraction

Following MG21, we construct the feature vector for the *TESS* light curves by concatenating the vector of normalized light curve points and the vector of Lomb-Scargle spectral power values evaluated for a logarithmic range of frequencies (corresponding to periods spanning 4 hours to 27 days). The Lomb-Scargle method constructs the power spectrum of regular or irregular time series by using the equivalence between the least-square fit to a periodic signal using sinusoidal
functions and the classical Fourier transform. Specifically, the power spectrum is a simple function of the $\chi^2$ value for each frequency of the sinusoidal model. A fit to the data is performed by adjusting the amplitude and the phase, resulting in an estimation of the harmonic content of the signal (the periodogram). This approach is computationally much less expensive than the Fourier transform (VanderPlas, 2018). The resulting vectors contain information on both the relative amplitude and the frequency properties of each light curve and are defined for the same set of times and frequencies for all light curves.

In MG21, we have thoroughly studied how the selection of these features affects the anomaly detection algorithm and have demonstrated that passing this set of feature vectors to the URF algorithm results in the successful identification of anomalies, including bona-fide objects. In particular, we have demonstrated that the distribution of anomaly scores obtained from the analysis shows a clear, distinct peak of anomalous objects that are assigned a significantly higher anomaly score. In the Kepler data, this population of anomalies is dominated by rare periodic and non-periodic variable stars, with large normalized amplitude variations, and characteristic timescales of either a few hours or a few months.

We now describe the method used to identify the optimal range of frequencies for the URF algorithm for the specific case of the TESS light curves.

### 5.3.2 Feature Optimisation

At a 2-minute cadence over a range spanning 27 days, a typical TESS short-cadence light curve contains over 19,000 points. Using the entire length of the light curves for this analysis is both scientifically unnecessary for our purposes and computationally prohibitive, as the computing time required by the URF scales linearly with the number of features used and quadratically with the number of objects considered. Consecutive light curve points at this time resolution tend to be highly correlated, and most stellar variability phenomena occur at timescales longer than 1 hour (Eyer & Mowlavi, 2008). The correlation between points typically leads to overfitting when a random ensemble method, such as the URF, is applied to the data. Therefore, we opt for reducing the number of light curve points used in the analysis to 3000 points, which is achieved by performing a uniform and regular sampling of the original light curves. The resulting light curves have a time resolution of about 10-15 minutes or between a third and a half of the Kepler cadence.
Figure 5.3 Importance score for each feature during the training of the random forest using data from Sector 1. Data points to the left of the dotted line represent flux data, whereas those to the right represent the periodogram. As seen in the first section of periodogram data the higher frequencies hold negligible importance and why the minimum period is 4 hours rather than extended to sample near the cadence of the telescope.

To investigate which frequencies contain information relevant to anomaly detection, we performed test runs of the URF algorithm using a different number of periodogram points, and frequency boundaries, while keeping the number of light curve points at 3000. The resulting URF scores did not change significantly for more than approximately 1000 elements in the periodogram, or for a lower bound of the frequencies corresponding to timescales shorter than a few hours. We, therefore, set the frequency boundaries to values corresponding to timescales between 4 hours and 27 days, which correspond to the duration of the observation for each sector. The frequencies are distributed logarithmically between these boundaries. We end up with a feature vector of length 4000, with the first 3000 points corresponding to the light curve points, and the last 1000 corresponding to the periodogram. Our Lomb-Scargle periodograms are computed before the subsampling to minimise the loss of information.

In Figure 5.3 we show the feature importance for each of the 4000 features, resulting from performing the random forest classification on the first step of the URF algorithm. We note that importance distributes more or less evenly among light curve points, while it peaks for the spectral power features at frequencies corresponding to timescales of a few days. This is as opposed to the importance distribution for the Kepler light curves, where important timescales for anomaly
detection were either of the order of a few hours or a few weeks. The distinctions here reflect the differences that we have discussed between the Kepler and TESS input catalogues, with dwarf stars overrepresented in the latter and giant stars underrepresented. Pulsation modes and variability types typical of giant stars are present in the Kepler light curves and are unlikely to be found in the present dataset, whereas stellar flares, rotational patterns, and other features of dwarf stars are more likely to be found in the present study.

5.3.3 Optimisation of Hyperparameters

The optimisation of the URF hyperparameters is done using k-fold cross-validation. Specifically, we employ the Random Search Cross Validation implementation in scikit-learn [Pedregosa et al., 2011] with 3-fold validation to maximise the accuracy of the random forest classifier. We randomise the following parameters:

- min_samples_leaf: [1, 2]
- Bootstrap: [True, False]
- warm_start: [True, False]
- max_features: [sqrt, log2]
- min_samples_split: ['None', 2, 4, 7, 10]
- n_estimators: 10 even steps from 50 - 200
- max_depth: [100, 300, 500, 700, 900, 1000, “None”]

Selected parameters are explained in Table A.7. During the random search, I look for hyperparameters that maximise the validation accuracy of the RF classifier while avoiding over-fitting. Once the parameters that maximise this accuracy are identified, we fine-tune the specific hyperparameters near these values to increase the contrast in anomaly scores between objects. Fine-tuning is necessary because the parameters that maximise classification accuracy are not necessarily the same parameters that maximise the detection of anomalies. For example, by allowing a small amount of overfitting, we can improve the ability of the method to find anomalies as the more stringent isolation of light curves in the terminal
nodes reduces the range of light curve shapes that are considered non-anomalous. We have therefore tuned the hyperparameters to obtain a sweet spot between classification accuracy and the resulting fraction of anomalies.

The final parameters used for this study are: \( n_{\text{estimators}}=100, \) \( \text{warm\_start}=\text{True}, \) \( \text{bootstrap}=\text{False}, \) \( \text{min\_samples\_leaf}=2, \) \( \text{max\_features}=\text{‘auto’}, \) \( \text{min\_samples\_split}=4, \) \( \text{max\_depth}=700. \)

5.3.4 Implementation and Hardware

The computation of the anomaly scores uses 96 GB of memory and approximately 16 hours of run time per sector. The periodograms, URF training and anomaly scores were calculated by Cuillin Computing Cluster at the Institute of Astronomy in the Royal Observatory Edinburgh. As described in MG21 and earlier in this chapter, we use a modified version of the original URF algorithm that significantly reduces the computation time to calculate the anomaly score. Specifically, we do not use a pairwise match for each pair of light curves, as this is redundant. Instead, we shift to the analysis of the terminal node populations. The updated method scales linearly with the number of final nodes bringing increasing returns the larger the input catalogue.

The uncertainty in the anomaly scores originates from the random nature of the ensemble methods and the finite number of trees. This uncertainty is inversely related to the number of trees used and results in a variance associated with each computed anomaly score. The values provided here for the anomaly score of each light curve correspond to the average score over ten independent realisations of the method.

5.4 Results

Results from the URF are published alongside this work, with Table 5.1 showing the format. Columns 1–5 represent identifiers for the object, Column 6 represents the anomaly score from this work, columns 7–10 explain the sectors observed and the population of each light curve, e.g. the anomalous, bulk or intermediate populations, and columns 11 & 12 are observations also from this work and products of upcoming analysis in § 5.6.1 & § 5.6.3. Figure 5.4 shows randomly
Figure 5.4 Examples of outliers identified in Sectors 18–24 without well-defined classification labels. Anomalous behaviour decreases from left to right, with a random sample taken from each 20th percentile. The light curves are in blue, with the black light curve representing a very non-anomalous light curve, TIC 261337074, for reference on magnitude variations. Plots from Sector 23 reveal the nature of the detrimental trends identified.
selected light curves from evenly spaced percentiles from a handful of sectors.

5.4.1 The emergence of distinct populations

Figure 5.5 shows the distribution of URF anomaly scores for the 24 TESS sectors considered in this work. The distribution consists of two distinct peaks, one centred at a score of about 0.35, and another at about 0.93. A sparsely populated region of objects with intermediate scores lies between them. The bulk of the population (objects with anomaly scores \( \lesssim 0.6 \)) corresponds to what we can consider normal light curves, that is, objects whose light curve features are sampled from a parent population that represents the most common forms of variability (or lack thereof). On the other hand, objects with anomaly scores greater than 0.9, represent light curves with outlying amplitude and frequency properties sampled from a different distribution with respect to the bulk of the objects. These are the ones that we will consider anomalies in this work. Up to this point, they only represent anomalies from a purely data-centred perspective. Whether they constitute truly novel astrophysical objects, observational and/or pipeline artefacts, or only rarer classes that are underrepresented in the dataset,
<table>
<thead>
<tr>
<th>tic_id</th>
<th>designation</th>
<th>ra</th>
<th>dec</th>
<th>main_type</th>
<th>anomaly_score</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIC 471013946</td>
<td>Gaia DR2</td>
<td>25130218554931769344</td>
<td>33.821292</td>
<td>0.244569</td>
<td>WD*</td>
</tr>
<tr>
<td>TIC 166463090</td>
<td>Gaia DR2</td>
<td>6670392146960681984</td>
<td>298.835020</td>
<td>-48.282248</td>
<td>gammaDor</td>
</tr>
<tr>
<td>TIC 177932603</td>
<td>Gaia DR2</td>
<td>3052092481686246016</td>
<td>107.658294</td>
<td>-7.122892</td>
<td>deltaCep</td>
</tr>
<tr>
<td>TIC 444535842</td>
<td>Gaia DR2</td>
<td>51099133315346688</td>
<td>19.837507</td>
<td>62.301401</td>
<td>NaN</td>
</tr>
<tr>
<td>TIC 30317282</td>
<td>Gaia DR2</td>
<td>4661514377893838464</td>
<td>74.272331</td>
<td>-68.414734</td>
<td>EB*</td>
</tr>
<tr>
<td>TIC 439869954</td>
<td>Gaia DR2</td>
<td>249789505310247040</td>
<td>43.384666</td>
<td>-0.562453</td>
<td>WD*</td>
</tr>
<tr>
<td>TIC 142867174</td>
<td>Gaia DR2</td>
<td>5054768262723451952</td>
<td>50.837600</td>
<td>-32.270100</td>
<td>NaN</td>
</tr>
<tr>
<td>TIC 178366477</td>
<td>Gaia DR2</td>
<td>3071240270519385856</td>
<td>123.327106</td>
<td>-1.057903</td>
<td>CataclyV*</td>
</tr>
<tr>
<td>TIC 149894385</td>
<td>Gaia DR2</td>
<td>5587836255503080576</td>
<td>116.769000</td>
<td>-34.372400</td>
<td>NaN</td>
</tr>
<tr>
<td>TIC 444000734</td>
<td>NaN</td>
<td>131.365000</td>
<td>10.914900</td>
<td>PM*</td>
<td>0.986069</td>
</tr>
<tr>
<td>tic_id</td>
<td>sectors</td>
<td>weirdness$\geq 0.9$</td>
<td>weirdness$\geq 0.6$</td>
<td>Notes</td>
<td>high_variance</td>
</tr>
<tr>
<td>---------------</td>
<td>---------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>-------</td>
<td>---------------</td>
</tr>
<tr>
<td>TIC 471013946</td>
<td>[4]</td>
<td>[4]</td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>TIC 166463090</td>
<td>[13]</td>
<td>[13]</td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>TIC 177932603</td>
<td>[7]</td>
<td>[7]</td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>TIC 444535842</td>
<td>[24]</td>
<td>[24]</td>
<td></td>
<td>[artefact ]</td>
<td>-</td>
</tr>
<tr>
<td>TIC 30317282</td>
<td>[7]</td>
<td></td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>TIC 439869954</td>
<td>[4]</td>
<td>[4]</td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>TIC 142867174</td>
<td>[4]</td>
<td>[4]</td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>TIC 178366477</td>
<td>[7]</td>
<td>[7]</td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>TIC 149894385</td>
<td>[8]</td>
<td>[8]</td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>TIC 444000734</td>
<td>[7]</td>
<td>[7]</td>
<td></td>
<td>0</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5.1

(1) - TIC ID from Tess catalogue
(2) - Designation ID from Gaia DR2
(3) - Right Ascension
(4) - Declination
(5) - Main object type from SIMBAD
(6) - Average Weirdness Score from this work
(7) - Sectors the object is present
(8) - Sectors where the Weirdness Score $\geq 0.9$
(9) - Sectors with $0.6 \leq$ Weirdness Score $\leq 0.9$
(10) - Sectors with Weirdness Score $\leq 0.6$
(11) - Notes from visual inspection, discussed in §5.6.1
(12) - Flag for high variance objects, discussed in §5.6.3

Top ten rows of the final results table, included are the key identifiers, anomaly score, sectors observed, classification within each sector alongside any observations on the light curves a high variance flag.
is the subject of our investigation in the upcoming sections.

Overall, about 82% of the TESS targets have an average anomaly score $< 0.6$; 7% of them have an intermediate average score between 0.6 and 0.9, and 11% have average anomaly scores higher than 0.9. The average is calculated over multiple observations of the same object, which is more likely to occur for targets located at high ecliptic latitudes. The distribution of anomaly scores is very similar from sector to sector, apart from a different normalisation in each case, due to differences in the total number of targets detected in each sector. In particular, the population of outliers with scores $> 0.9$ represents a similar fraction of the cases in all sectors, with small variations. For example, in Southern Hemisphere sectors, the fraction of anomalous light curves is $0.11^{+0.03}_{-0.02}$, whereas, for Northern Hemisphere sectors, it is $0.08^{+0.007}_{-0.018}$. Sector 23 is an outlier with a fraction of anomalous objects of only 0.02. The variation in the fraction of outliers between hemispheres is understood to be due to different fractions of the sector field containing the galactic plane. The average value of the anomaly score is very similar for all sectors, with a standard deviation of only 0.02 for the 24 averages. This all suggests that each sector is representative of the entire dataset and that the anomalous nature of the objects is related to astrophysical or observational phenomena that are common to all sectors, which favours an interpretation according to which most of the anomalies - but not all of them - are objects of a known class with particularly outlying variability parameters.

In our results (An example of which can be found at Table 5.1), we present the average anomaly score for each anomalous object in our dataset, computed across all sectors in which the object is observed. For this work, we define anomalous objects as those for which the average score is higher than 0.9. To indicate the dispersion of each object’s score across sectors, we also list sector IDs where the object is detected and the sectors where the score is higher than 0.9 or below 0.6. We justify providing the average score as most objects show nominal variance between sectors. Additionally, we provide a flag (high variance) for the objects that contradict this assumption. We dedicate part §5.6.3 to discuss the prominent cases where a large dispersion in anomaly scores is observed across sectors, as those objects represent anomalies that are not persistent in time.
5.4.2 Consistency Across Sectors

Our dataset contains over 400,000 light curves for more than 200,000 unique targets. Despite an average of 2 light curves per object, less than half (\(\sim 70,000\)) are observed in multiple sectors. The majority of these objects have similar anomaly scores across all observations. However, there is a small fraction where the anomalous variability pattern occurs during a particular observation. This event leads to an increased anomaly score in that sector, with the other sectors remaining unaffected. Naturally, this leads to an increased variance for that object, indicating the anomalous light curve may be an exceptional or rare behaviour that is not persistent in time, for example, a transient event, ASASSN-21lw (Shappee et al., 2014).

Nonetheless, we compute the standard deviation of the anomaly scores for all objects observed more than once. We show the resulting distribution of standard deviations in Figure 5.6. We find consistency between the scores across observations, with 95% of the objects having \(\sigma_{\text{Anomaly}} < 0.1\). The mean and median of the \(\sigma_{\text{Anomaly}}\) distribution are 0.027 and 0.017 respectively. Implying...
the average anomaly score is a good diagnostic for the anomalous nature of the vast majority of the sources. We now discuss the cases where this is not true.

Only about 0.38% of the objects have a standard deviation higher than $\sigma > 0.25$ between sectors. These might indicate short-lived transient events, or periodic signals such as dips with a period significantly longer than the duration of the TESS light curves. Astrophysical transients such as cataclysmic variables, stellar flares, microlensing events and other unknown explosive events could also be part of this group. Additionally, data processing or instrumental artefacts, such as sudden drops in the baseline level of the light curve, can also be identified in this fashion. We discuss this group of anomalies in §5.6.3.

A total of 44,889 light curves in our sample have anomaly scores greater than 0.9, representing 25,858 unique objects. As is usually the case in astronomical anomaly detection, the list of anomalies is too large to allow for the characterisation of each light curve. Our approach for prioritisation here is to explore the anomalous objects to understand the astrophysical mechanisms driving the anomaly score, followed by methods probing beyond the anomaly metric alone. Such methods potentially isolate the rarest or most extreme anomalous behaviours, such as objects likely to be non-persistent anomalies which indicate the observed anomalous behaviour is restricted to a single sector, as opposed to behaviour that results from repeating or periodic patterns. Nevertheless, we provide the list of anomalies and provide a description based on the overall light curve type, determined using visual inspection for sectors 13 onward. Additionally, in the next section, we incorporate information related to the astrophysical properties of these sources, such as luminosities and effective temperatures, to assess the relationship between anomalous nature and evolutionary stage for stars in the TESS input catalogue.

5.5 Linking anomaly scores to astrophysical properties: the Colour-Magnitude diagram

The discussion of whether a particular light curve represents a true astrophysical novelty or relates to an understood phenomenon within a particular data set is non-trivial due to the required domain knowledge that may require modelling or other tools of inference. An initial step involves identifying the objects with
a known class. Only about 30% of the anomalies identified in this work have a known class described in independent studies. Since the vast majority of the TIC objects are stars, we can rely on additional information from their Gaia observables to constrain their physical properties. In Figure 5.7 we show the Gaia-generated Colour-Magnitude diagram for our targets, colour-coded by the average anomaly score across all sectors. This diagram allows us to relate the anomaly scores to particular spectral types or evolutionary stages, setting a general framework for the interpretation of our results. Our cross-match indicates that ~90% of all targets considered here have reliable distance and photometry measurements from Gaia, due to the selection of only bright nearby objects for TESS high cadence targets. This fraction is similar in all sectors.

There are several interesting trends in Figure 5.7. A prominent region of the HR diagram with overall high anomaly scores is the white dwarf branch. The instability strip is also a prominent region of relatively high anomaly scores, potentially driven by the radial pulsations of stars occupying this part of the diagram. Other regions with relatively high anomaly scores include; supergiants, subdwarf B stars, M-dwarfs and ultra-cool dwarfs. There are also objects whose optical variability is likely to be extrinsic, rather than intrinsic, or a combination of both. These include young stellar objects (YSOs), which show variation due
to internal stabilisation and obstructions from dusty halos, explained in greater
detail in § 5.7.4. Additionally, objects that occupy sparsely populated regions
of the HR diagram, e.g. the coolest region of the M-dwarf branch, are often
classified as an outlier. Our results demonstrate the isolation observed in the HR
diagram, due to the extreme luminosities or temperatures, is reflected in their
variability patterns.

We observe an increase in the anomaly score as we move into cooler, low-mass
dwarf stars, from convective envelopes into fully convective M-type stars. In this
regime, it has been proposed as part of basic dynamo theory that stellar magnetic
activity should increase inversely to mass, accounting for the proportionally
deeper convective envelope (Radick, 1992; Brun & Browning, 2017). A significant
part of the variability in these stars is the result of chromospheric activity, such as
sunspots (Baliunas et al., 1995; Schuessler et al., 1996; Barnes & Collier Cameron,
2001; Barnes et al., 2011).

Focusing now on the right panel of Figure 5.7, where we have limited the anomaly
score range to only the most anomalous scores, we note that the highest anomaly
scores are distributed among stars in different regions of the HR diagram, with
an increased density of anomalous objects near the instability strip where the MS
meets the giant branch. Low-mass stars and white dwarfs consistently have lower
anomaly scores, whereas stars in the giant branch generally do not make the > 0.9
score mark to be included in the anomalous group, which was not the case for
Kepler targets. In the particular case of white dwarfs, variability is dominated
by rapid flux variations that deviate between 1% and 2% from the mean flux. In
§ 5.6.2 we describe the variability trends in each case and investigate potential
physical mechanisms for the anomalous behaviour.

The MS remains, at best, sparsely populated by anomalous light curves, likely
the result of the relatively stable hydrogen-burning cycle of MS stars. In this
region, the observed anomalous time-domain behaviour is caused predominantly
by extrinsic phenomena, such as eclipses. As for RGB stars, they are largely
stable throughout this phase of evolution at the timescales probed by the TESS
data, and they are sufficiently represented in the TIC not to be considered
a rare population. The timescales of their spectrally dominant variability
modes are longer (hundreds and thousands of days) than the timescales of TESS
observations. The assumption of variability over long timescales does not hold
for all types of supergiant stars, with some of them undergoing different stages of
evolution, particularly members of the blue supergiant population. We expand
our exploration of giant stars further in §5.7.5. We note here, however, that there are signs of increased anomaly score in the cool extremes of the red giant branch, where we observe variability with periods between \( \sim 30 - 80 \) days which can be considered “long period variability” compared to the observation timescales.

In what follows, we explore the set of anomalous light curves, both in terms of their value (i.e., are there any true astrophysical one-offs?) and how the anomaly score relates to specific physical parameters for objects belonging to known classes. This will allow an investigation of the distribution of physical parameters for objects of specific types (e.g. eclipsing binaries) and the most extreme among those. Our methodology is as follows: first, we perform a visual inspection of the anomalous light curves and categorise them according to their overall variability pattern. We then use additional metrics, such as the standard deviation in the anomaly score across observations, to search for non-persistent anomalies. Finally, we study anomalies of specific known classes (e.g. white dwarfs, eclipsing binaries) and investigate what the anomaly score tells us about their specific configurations.

### 5.6 Most Anomalous Objects Identified

#### 5.6.1 Visual Inspection of Light Curves and Classification

We perform a visual inspection of all anomalous (score \( \geq 0.9 \)) light curves in sectors 13 through 24 that do not have an unambiguous classification in the SIMBAD database. This includes objects that only had uninformative labels in SIMBAD, such as “Star”. Rather than identifying all potential unique objects, our initial goal is to understand the diversity and composition of the landscape of TESS anomalous light curves.

We inspect each anomalous light curve individually, regardless of whether the associated object has multiple light curves. This work has analysed over 12,500 light curves. As a result of the inspection, we exclude sectors 22 and 23 from any further analysis due to a disproportionate population of artefacts not corrected by the PDC pipeline, leading to inconsistent outlier populations. Sector 23 is the most affected with only 0.8% of light curves classified as anomalous, compared to the typical 5-10% in other sectors. Although we exclude them from the analysis, we still include results from these sectors in our catalogue of anomalies.
<table>
<thead>
<tr>
<th>Shorthand</th>
<th>Descriptive</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>s</td>
<td>sinusoidal</td>
<td><img src="image" alt="Sinusoidal Example" /></td>
</tr>
<tr>
<td>m</td>
<td>modulated</td>
<td><img src="image" alt="Modulated Example" /></td>
</tr>
<tr>
<td>e</td>
<td>artefact</td>
<td><img src="image" alt="Artefact Example" /></td>
</tr>
<tr>
<td>p</td>
<td>peaked</td>
<td><img src="image" alt="Peaked Example" /></td>
</tr>
<tr>
<td>dd</td>
<td>dips</td>
<td><img src="image" alt="Dips Example" /></td>
</tr>
<tr>
<td>i, I</td>
<td>irregular</td>
<td><img src="image" alt="Irregular Example" /></td>
</tr>
<tr>
<td>mb</td>
<td>multi-body system</td>
<td><img src="image" alt="Multi-body System Example" /></td>
</tr>
<tr>
<td>t</td>
<td>transit</td>
<td><img src="image" alt="Transit Example" /></td>
</tr>
<tr>
<td>LPV</td>
<td>long period variable</td>
<td><img src="image" alt="Long Period Variable Example" /></td>
</tr>
<tr>
<td>-</td>
<td>No Label</td>
<td><img src="image" alt="No Label Example" /></td>
</tr>
</tbody>
</table>

**Table 5.2** List of the most common descriptors used in the classification of anomalous objects. The examples represent a typical object from these classes. In reality, these labels cover a range of patterns and are often used in conjunction with each other, e.g. Modulated Sinusoidal. Additional labels are used to separate the main labels (those with examples above) into subcategories.
for completeness and to highlight results such as these are beneficial to understand the limitations and improve the PDC pipeline.

Our classification is based on the morphological properties of the light curves rather than on astrophysical types for two main reasons: first, we would require a vast amount of domain knowledge and consideration to correctly assign accurate astrophysical classes based on the light curve alone. In addition, since we expect some anomalies do not belong to any known class, our best efforts can only describe their morphology. We converged to the final classification in Table 5.2 iteratively by initially setting a set of labels based on Sector 24 alone. These were updated as more analysis revealed slightly different patterns in different sectors. Finally, we reclassified sector 24 to reduce potential inconsistencies from inexperience and missing labels during the initial classification.

As an example, RR-Lyrae stars are classified as “sinusoidal” due to their periodic pattern resembling a sinusoidal function despite not truly being sinusoidal. When appropriate, additional labels are included to refine the class, so “sinusoidal asymmetrical” indicates a distinctive pattern in which there is a sinusoidal pattern with asymmetrical peaks or troughs indicating more rapid increases in brightness than decreases or vice versa. Objects that do not fit in any morphological types are left unlabelled. If the astrophysical nature of the variability pattern is clear from the morphology, this is reflected in the label, as is the case for “transit events”, which primarily include eclipsing binary systems. Overall, descriptive labels are used in ambiguous cases to minimise mislabelling and class overlap. In Table 5.2 we compile a summary of the most common labels and provide light curve examples. The only exception to this morphological system of classification is white dwarfs, which we label according to the external catalogue provided by Gentile Fusillo et al. (2019). This divergence in approach is required as the white dwarf population systematically returns higher than average anomaly scores that are difficult to determine visually. We believe this is due to rapid pulsations described in §5.7.6.

5.6.2 Composition of Outlying Population

Out of the 12,500 light curves inspected, 4,128 are classified using the morphological system summarised in Table 5.2. An additional 3,063 objects have been previously classified as white dwarfs (WDs) by Gentile Fusillo et al. (2019), with only ~ 1% falling in one of the morphological categories. We also identify several
Figure 5.8 The distribution of descriptions for objects with anomaly scores above 0.9 and lacking prior classification. The dataset includes $\sim 30\%$ of objects in sectors 13 - 21 and 24, with the remaining objects lacking defining features upon visual inspection or catalogue comparisons.

Figure 5.9 Top: A random sample of $\sim 30$ periodograms from the labelled outliers, unlabelled outliers that show an increased variability magnitude, and objects with anomaly score less than the bulk population peak from Sector 13. Bottom: Full CAMDs for the same distinct populations as the periodograms above (Unsampled). The distributions highlight the type of objects in each class. White Dwarf stars are deliberately removed from the sample as these are classified using an alternative approach.
objects with minor artefacts, e.g., single-pixel spikes and trends in small sections of the data. Since these minor features are not likely to dominate the anomaly score, we classify these objects based on the morphology of unaffected parts of the light curve.

In what follows, we refer to light curves with a repeating pattern of short but statistically significant dips, including transits and suspected eclipses, as “multi-body” systems. The breakdown of light curve classification is as follows: 3063 are classified as white dwarfs, 1275 as sinusoidal, with an additional 1244 showing additional patterns to the sinusoidal baseline (alt_sinusoidal), such as modulation, short periods (less than \( \sim 8 \) hours) or asymmetrical peaks. There are 500 multi-body systems, 423 periodically peaked light curves showing repeated peaks of emission, 264 irregular light curves and 237 dominated by pipeline artefacts. In addition, 129 are modulated patterns and 43 show long-period variability (LPV), defined in this work as a pattern with an apparent period \( \geq 26 \) days. 13 light curves do not fit any of the morphological classes but have a unique description. Figure 5.8 provides a visualisation of this breakdown.

There is a group of light curves for which the only noticeable feature is a relatively higher amplitude than a non-variable noise-dominated light curve that populates the low-anomaly end of the distribution. The majority of these higher amplitude light curves are associated with low-luminosity, cool M-dwarf stars. As the light curve shows little evidence of anomalous behaviour, we must explore the frequency space and investigate their power spectrum. Figure 5.9 shows the periodograms and location in the HR diagram for three types of objects: anomalies that fit in any of our morphological types (left), anomalies that show the higher amplitude feature (centre), and non-anomalous light objects (right).

We note that the periodograms of M-dwarf anomalies are relatively featureless compared to the morphologically classified objects, which indicates that they do not have well-defined periodic features. However, similar to the morphological anomalies, they have a greater fraction of the power contained at low-frequency modes than the bulk objects. This pattern indicates dominating variability at relatively long (weeks) timescales. Evidently, these objects are anomalous due to both having higher amplitude and features within their frequency spectrum, supporting results from MG21, where we found low-frequency modes of variability are associated with anomalous light curves in *Kepler* data.

The stellar rotation might be related to this anomalous behaviour. As demon-
Figure 5.10  A visualisation of the standard deviation of anomaly score between sectors vs the average anomaly score. The colour bar represents the number of sectors observed. As observed, we expect the intermediate population to show an average higher standard deviation due to the smaller population within this region. Nonetheless, the objects that occupy the region above the bulk are of the greatest interest. The boundary line represents the boundary between the bulk and the high standard deviation sources for this section of analysis.

Strated in the observational study by Popinchalk et al. (2021), M-dwarfs of earlier spectral types (M4-M6) are rapid rotators, with rotational periods of less than one day to as short as a few hours. M-dwarfs become cooler as they evolve, causing their rotational period decreases, possibly due to magnetic effects (Garraffo et al., 2018), until it stalls for late types. The reduction in rotation explains why the majority of field M-dwarfs with measured rotation periods are greater than 10 days, with non-periodic variability timescales typically a few weeks to months in the later types. This non-periodic excess of spectral power at longer timescales is what makes their light curve anomalous. The fact that we observe a gradient in the anomaly score along the main M-dwarf portion of the main sequence, with cooler M-dwarf stars being more anomalous, is indicative that our method is sensitive to the disappearing rotation signature. These results, once more, suggest a correlation between the anomaly scores and the object’s evolutionary stage, at least for certain types of objects.
5.6.3 Non-persistent anomalies

A small fraction of the objects show a high variance in the anomaly score between sectors, which may indicate a particularly elevated score in only one of the observed epochs. These are candidate non-persistent anomalies, i.e., anomalous behaviour that is non-repeating and could signal single-event transients. About 95% of the objects show a standard deviation below 0.1 in their scores, and just 0.38% of them have a standard deviation above 0.25 (see Figure 5.6). In this subsection, we investigate these high-variance objects.

In Figure 5.10, we plot the average anomaly score against the score’s standard deviation for each object. Symbols are colour-coded by the number of observations. Most objects populate a relatively dense region below the polynomial dashed line, with those above the dashed line considered “high variance objects” in this work. The threshold is defined as $17.92x - 69.63x^2 + 129.07x^3 - 112.37x^4 + 36.77x^5 - 1.72$, where the curve is derived through regression analysis and intersect increased by $2\sigma$ from the best fit location. In total, we find a population of 1789 high variance objects, with the most extreme variances ($\sigma_{\text{Anomaly}}$) reaching 0.35. The population generally shows a higher variance in the intermediate region of anomaly scores (0.6-0.9), possibly due to the reduced population in this area compared to the extremes of the anomaly range. We now investigate the differences in variability in the high variance population to the outliers discussed in §5.6.2.
We concentrate here on objects with high variance for which at least one of the individual scores is higher than 0.9. We identify 636 such light curves, corresponding to 410 unique objects. We examine the recorded classifications, similarly to the approach in §5.6.1, and inspect their light curves to evaluate their variability properties. Compared to the total outlier population, we find a greater fraction of these objects classified as “other” in the population of non-persistent outliers. Sinusoidal types still dominate, but there is also an increased fraction of “artefacts” and “Long Period Variables”. The fraction of objects with multiple transits remains approximately the same. Remarkably, white dwarfs are not represented in this group at all. Figure 5.11 visualises the breakdown of classes identified within the high variance population.

Further investigation reveals that the sinusoidal light curves found in this group have longer periods when compared to the sinusoidal anomalies in the general outlier group, with practically no light curve showing periods shorter than one day. Perhaps the most interesting objects in this group belong to the “irregular” and “multi-body” system categories. In Figure 5.12, we show examples of light curves in this group. The upper left panel shows an irregular light curve corresponding to high proper motion star TIC 370327409, which has been
classified as a binary \cite{Eggleton2008} and has spectral type K2. The lower left light curve corresponds to TIC 156462093 and shows a light curve similar to those of heartbeat stars previously found by \textit{Kepler} \cite{Thompson2012}.

In the second column of Figure 5.12 we identify two transit events of interest (\textit{Top: TIC 230125546, Middle: TIC 350298314}). These are likely binaries with orbital periods long enough to be captured during transit only once during the \textit{TESS}, with the transit visible in the plotted light curves. In fact, both objects are identified in \textit{Prša et al.} (2022), a dedicated search for binary stars. TIC 350298314 is measured to have a period of 47.72 ± 3.516 days, placing it in the top 20 longest periods identified out of 4584 confirmed systems. Because of the limitations on inclination required for a transit to occur at such long orbital periods, it is evident this represents an extreme example of an outlying class within our sample. The identification of this extreme system confirms the effectiveness of exploring non-persistent outliers in the case of transits with large member separation.

Other examples in Figure 5.12 include LPVs and artefacts. We note that in the case of artefacts, there is a subclass of objects that look like repeating, regular flaring events (bottom right panel). We have included them as artefacts because, in the majority of cases, we have determined they originate from an improper background subtraction, where the background area contained a source with repeating transits. Nonetheless, gravitational lensing is a potential astrophysical phenomenon that could theoretically cause this behaviour. If a compact, undetected object orbits a star, the resulting lensing pattern can look like the bottom right plot in Figure 5.12. Thus, a number of this flaring event type of anomaly may have an astrophysical origin.

### 5.7 Analysing Evolutionary Stages Across the Hertzsprung-Russel Diagram

In this subsection, we investigate the relationship between the anomaly score and the physical properties of anomalies of known class. The motivation is that even if the general astrophysical type is known, there can be members of a particular group with outlying physical parameters that could indicate anomalous configurations. Examples include orbital periods that are either too short, too
long or very deep transiting events. Below we perform this analysis for TESS objects of interest, stellar systems with multiplicity greater than 1, such as eclipsing binaries, pulsating stars, young stellar objects, giants, and white dwarfs.

### 5.7.1 TESS Objects of Interest

The TESS mission team regularly releases a list of TESS objects of interest (TOI’s hereafter) containing the most promising exoplanet candidates. For detailed follow-ups with several transits per object, candidates with estimated orbital periods of less than ten days orbiting bright host stars have priority (Guerrero et al., 2021). At the time of download, the list of TOIs contained 2,647 exoplanet candidates. We now turn to the question of whether transiting exoplanets show up as anomalies in our analysis. Since we do not analyse folded or stacked light curves, it is unlikely that this particular anomaly detection method will detect most of the candidates.

We look at the 1,262 exoplanet candidates from the TOI list that overlap with the
sectors analysed in this chapter. For TOIs, 87% are bulk objects, with anomaly scores less than 0.6, 6.3% are anomalous objects, with anomaly scores above 0.9, and the mean anomaly score is 0.44, with a standard deviation of 0.16. As we suspected, these values are not significantly dissimilar to the general population, indicating that exoplanet transits do not dominate the anomaly score of the stellar system in this work. Nevertheless, we have the data to explore how the orbital period, transit duration and transit depth affect the anomaly score. This analysis is relevant in the study of the most prominent exoplanet transits and population studies of eclipsing binary stars.

In Figure 5.13 we show the distributions of transit parameters for TOIs in two different ranges of anomaly scores: anomalous with an anomaly score above 0.9 and non-anomalous objects with an anomaly score below 0.6. We note that anomalies clearly show deeper transits than the bulk objects, with anomalous transits being close to 15-20 ppt in depth versus < 1 for the majority of the bulk population. Furthermore, there are indications of a difference in the distribution of orbital periods, with exoplanet anomalies generally having orbital periods shorter than the bulk objects, with the outlier orbital periods primarily remaining below five days. Although in this case, the difference is not as prominent as with the transit depths.

In the upper panel of Figure 5.14 we compare the anomaly score distributions for TOIs and eclipsing binaries. We note that, at just 6.3%, anomalies represent a smaller fraction of the total in the case of TOIs than binary stars, for which the vast majority of light curves are anomalous. The distribution suggests that dim exoplanet transits are best inferred from phased, stacked light curves and are not picked up as anomalies when individual, unphased light curves are considered. Overall, it appears that in transits, regardless of whether they are produced by eclipsing stellar companions or exoplanets, anomalies are selected to be deeper eclipses with somewhat shorter orbital periods. This has implications for the typical exoplanet size. For example, for fixed semi-major axes, inclination and stellar properties for a K4V star, the radius of the exoplanet would need to be about three times larger to create a transit with the average depth of the anomalous objects, with respect to the size of the typical “normal transit” (see https://ccnmtl.github.io/astro-simulations/exoplanet-transit-simulator/). As deeper transits are more anomalous in this work, we conclude transits produced by larger planets will be considered more anomalous than those of smaller planets in the TESS dataset.
The main effect of the orbital period on the shape of the light curve is the number of transits that occur during a single observation. At approximately 27 days, the data is optimised for orbital periods less than roughly ten days. The effects would most prominently reflect on harmonics shown in the light curve periodogram for exoplanet transits. On the other hand, the transit duration as a fraction of the orbital period is more informative of the system. For a given size planet, the transit duration depends on multiple factors including inclination and orbital distance. Yet, in the upper right panel of Figure 5.13, there is not a major difference observed in the distribution of transit durations between the anomalous and non-anomalous populations. This may be due to the selected features or, most likely a strong selection effect in the orbital parameter space of TIC objects.

5.7.2 Multi-Body Systems

We call multi-body systems TESS sources whose light curves show regular dips that can be associated with eclipses. They are predominantly eclipsing binaries or single exoplanet transits, but systems with multiplicity greater than two that show more than one frequency in the eclipses are also included in this category.
Between 20 – 80% of all stars are predicted to have companions, with the actual fraction depending on the spectral type (Traven et al., 2020; Duchêne & Kraus, 2013). However, to show up as eclipsing light curves in TESS data, the orbital inclination and periods must be within specific narrow ranges. As a result, eclipsing light curves are rare, accounting for about 1% of the entire dataset. This results in the majority of them (67%) being identified as anomalies by our method.

In § 5.7.1, we have reported a correlation between the depth of the transit in exoplanet light curves and the anomaly score. We now investigate if this correlation extends to eclipsing binary stars. The top panel of Figure 5.14 shows that eclipsing binaries are labelled as anomalous at a significantly higher rate than exoplanets, with the vast majority of binaries having scores higher than 0.9. Presumably, this is because they result in deeper transits. In Figure 5.15, we compare the anomaly score to key transit parameters from Avvakumova et al. (2013): primary transit depth, secondary transit depth and period.

The results indicate that the primary transit depth is a driver of the anomaly score. Practically all the transits with depths larger than 1.5 mag have anomaly
scores higher than 0.95 as opposed to shallower primary transits, for which none has an anomaly score < 0.95. Secondary transit depths, on the other hand, have no discernible effect on the anomaly score, likely because of their shallower profile. A much better predictor of the anomaly score is the binary orbital period, with longer periods having a clear tendency to be more anomalous. The fact that longer orbital periods that result in fewer transits during the \textit{TESS} observation are rare in the TIC is a selection effect due to the mission specifications and not necessarily an astrophysical trend \cite{Prsa2022}. Nevertheless, our results imply that deep transits observed fewer times during a \textit{TESS} light curve are among the rarest variability type in this particular dataset. Our catalogue of anomalies contains previously unidentified objects of this rare class. In the lower panel of Figure 5.14, we show examples of binary transits with various orbital periods and transit depths.

The anomaly score can therefore be used as a probe of the physical parameters of binary and eclipsing systems, and in general, as a detector of these transits in a \textit{TESS}-like dataset. In our catalogue of anomalies, Table 5.1, these systems are labelled in column 12 as either “multi-body system”, or “Transit”.

### 5.7.3 Pulsating Variable Stars

Pulsating variables show periodic intrinsic variability that originates from the expansion and contraction of their surface layers as they transition out of the main sequence. The specifics of those oscillatory modes, as well as their periods and amplitudes, depend on the underlying physics. They occupy a specific region of the Hertzsprung-Russel diagram, known as the instability strip. \textcite{Gautschy1996} present a review of stellar pulsations along the HR diagram, emphasising that pulsations can occur at any mass and in various evolutionary stages. Cepheids and RR-Lyrae stars are among the most represented types within this class, of which \textcite{Clementini2019} identify 140,874 Cepheids and 9,575 RR-Lyrae stars in Gaia data. These objects represent less than 0.01% of the entire Gaia DR2 catalogue. To assess what physical mechanisms drive the anomaly score in pulsating stars, we rely on the independently determined classifications documented in SIMBAD. The resulting catalogue of known types of pulsating stars contains 156 RR-Lyrae stars, 468 Cepheid variables, and an additional 887 $\delta$-Scuti and $\gamma$-Dor stars. Also included are all other stellar types whose labels contain \textit{Pulsating Variable Star}, even if the specific types
Figure 5.16  
**Top Left:** Anomaly score of RR-Lyrae stars across all sectors.  
**Top Right:** Anomaly score of Cepheid Variable Stars in all sectors.  
**Bottom:** Anomaly score of other variable stars across all sectors. (e.g. δ-Scuti type variable.)

are unknown.

Figure 5.16 shows the distribution of anomaly scores for the most prominent pulsating types. The top left panel shows the distribution of scores for RR-Lyrae stars. 96% of the members of this class have an anomaly score greater than 0.9, with 98.1% of them having a score above 0.8. The classification is due to the RR-Lyrae variability pattern being uncommon in the TIC with a highly variable light curve, meaning our method will flag RR-Lyrae stars by assigning them a high anomaly score. Their pulsation periods range from \( \sim 0.2 \) days to 1 day [Soszyński et al., 2014, 2019], which corresponds to the range of frequencies extracted for the periodogram features. The few RR Lyrae stars with an anomaly score below 0.8 lack the expected RR-Lyrae pattern in their light curves, indicating a potential data issue most likely caused by either an incorrect classification or a cross-match error between catalogues. In the context of automatic source classification, the fact that false classifications can be identified based on their outlying anomaly score can be used to improve existing training sets, increasing their purity.
The top right panel of Figure 5.16 shows the anomaly score distribution for Cepheid variables, split into the following sub-classes: δ-Cepheid, W-Virginis and those classified simply as Cepheids. δ-type and W-Virginis types are identified as anomalies in about 90% of the cases. On the other hand, those classified as Cepheids are identified as anomalies only ∼ 50% of the time. As we had discussed previously (see Figure 5.9), anomalies show more structure in their periodograms, with clearly indicated peaks in the power spectrum. In particular, this holds for pulsating stars, with different types having unique characteristic frequencies. Cepheid stars with comparatively lower anomaly scores do not show a distinct spectral signature in the periodogram or higher power contained in long timescale (> 1 day) variability. Cepheids have periods that can exceed 100 days, whereas delta and W-Virginis periods are not known to exceed ∼ 20 – 30 days (Soszyński et al., 2017), and this can be another reason why a smaller fraction of Cepheids are detected as anomalous since their periods are outside the range of frequencies probed here. Nevertheless, the fraction of anomalous Cepheid variables (∼ 50%) remains significantly higher than the fraction of stars identified as anomalous in the entire catalogue (∼ 10%).

The bottom panel of Figure 5.16 shows the distribution of anomaly scores for the remaining types of known pulsators, including the ambiguous pulsating variable class. Compared to RR-Lyrae and Cepheids, a significantly smaller fraction of these objects corresponds to anomalous light curves. For δ-Scuti stars, the fraction is 24.2%, whereas for plain pulsating stars, β-Cephei and γ-Dor stars the fractions are respectively 21.3%, 42.7%, and 45.2%. For δ-Scuti and β-Cephei stars, this is somewhat due to their short pulsation periods, ranging from 0.02 to 0.25 days (Ziaali et al., 2019), placing the majority of these objects outside the range of frequencies probed by our periodogram features. γ-Dor stars show multiple oscillation modes with periods ranging between 0.5 and 3 days, resulting in the multi-modal pattern seen in the bottom right panel of Figure 5.17 (Pietrukowicz et al., 2020; Tkachenko et al., 2013). The pulsating stars without a specific class have a distribution of anomaly scores similar to the distribution for δ-Scuti stars, suggesting a similar distribution of their frequency properties. The periodograms of those with anomaly scores below 0.7 show little to no structure across the entire range and therefore have no discernible oscillatory modes within the range of periods probed here (4 hours to 27 days). While a verification by anomaly score can not be used in lieu of classification for these objects, it at least suggests that unclassified variable pulsating objects with anomaly scores below

\(^2\text{Classifications defined by the SIMBAD Astronomical Database}\)
Figure 5.17 Light curves of objects identified as outliers in the pulsating type stars with their distributions found in Figure 5.16. Target light curves are shown in blue, with the black light curve representing a non-anomalous light curve, TIC 261337074, for reference on magnitude variations. From the top down, other V* types are: delta-Scuti, delta-Scuti, Pulsating Variable and Gamma Dor type Variables.

0.9 are more likely to be similar to δ-Scuti or γ-Dor than RR-Lyrae or Cepheid variables.

We conclude that periodic pulsations are a driving factor behind the anomalous nature of TESS light curves when both the light curves and the power spectrum are considered together. A similar behaviour is observed for the Kepler light curves in MG21, where we also found pulsations to be among the main drivers of anomaly score. However, a smaller fraction of δ-Scuti stars are classified as anomalous in TESS than in Kepler, where practically all δ-Scuti stars were anomalous. The discrepancy between the two datasets is most likely due to a difference in the parent population of stars in both surveys, with M dwarfs being much more represented in both the general population of TESS targets and the group of anomalous objects. The anomaly score alone is not enough to distinguish between anomalous pulsators. The typical κ-mechanism driven stars show a similar distribution of anomaly scores to the non-radial, gravity mode pulsations in γ-Dor stars Kaye et al. (1999). In Table 5.1 the majority of these pulsating sources are noted as sinusoidal, with additional notes including asymmetrical
Figure 5.18  **Top Panel:** Histogram of anomaly scores for young stellar objects. Note that the scores are from individual sectors.  **Bottom four panels:** Example light curves of young stellar objects showing a weirdness score above 0.9.

or rapid depending on the specific shape. γ-Dor stars show a distinct pattern predominantly noted as irregular peaked.

### 5.7.4 Young Stellar Objects

Young stellar objects (YSOs) are stars in their earliest evolutionary stage, post-cloud collapse, but before they enter the Main Sequence. The youngest protostars are within a thick envelope of gas and dust arising from the original birth cloud, and the resulting obscuration manifests in the light curve of these young stars as an irregular variability pattern. As the primordial cloud dissipates, magnetic activity intensifies, producing extreme stellar flares as dissipated magnetic energy converts into kinetic energy. These flares often interact with the remaining circumstellar disk. In addition, a non-uniform distribution of starspots [Bhardwaj et al., 2019] can cover up to 40% of their surface [Herbst et al., 1994; Kesseli et al., 2016; Guo et al., 2021]. These conditions all add to the stochasticity of the light curves of young stars. Furthermore, their relatively short rotation periods (typically 1 to a few 10’s of days [Froebrich et al., 2021]) allow semi-periodic signals to be present.
The top panel of Figure 5.18 presents the distribution of anomaly scores for previously known YSOs in our dataset. We note that YSO light curves tend to rank highly in the anomaly score distribution, with objects making the threshold of 0.9 being considered an anomaly. The four lower panels of Figure 5.18 show representative YSO light curves. The irregular pattern, with relatively long-term baseline variability, as well as the presence of dips are likely contributors to the anomaly score. In fact, among the types of light curves evaluated as part of this work, the YSO light curves are the closest analogues of a remarkable light curve, namely Boyajian’s star. While most of the objects classified in our nomenclature as ‘irregular’ probably are unclassified YSOs, we believe that if any true analogues of Boyajian’s star have been detected, they would belong to this group.

Out of a total of 190 YSOs, 67% have anomaly scores above 0.9 and 81% above 0.6. The remaining objects, which are part of the bulk population, show significantly less structure in their light curves. As low-mass YSOs evolve towards the main sequence, both the accretion rate and the size of the disk reduce, causing less variability observed in the corresponding light curve (Chapter 6 in Schultz [2005] provides a detailed explanation of YSO evolution). Hence, the reduction in variability observed in the YSOs in the bulk population indicates a more evolved system than their anomalous counterparts, with less obscuration, accretion, or magnetism-related variability imprinted in their light curves. Furthermore, when studying a Hertzsprung-Russell diagram, these normal YSOs also align closer to the main sequence than the anomalous YSOs, further indicating they correspond to a more evolved system. These observed young stars are expected to be of a low mass, as the timescales and radiative feedback in high-mass young stars generally prevent the formation of a long-lasting accretion disk (Shepherd [2003]). Therefore, our method consistently flags the irregular light curves that result from a combination of accretion activity and obscuration corresponding to young low-mass stars as anomalies. These are the closest analogues we find to the unique behaviour shown by Boyajian’s star.

In Table 5.1, these systems are predominantly classified as Irregular Dips, with those previously classified as YSOs marked as such in the main_type column.

### 5.7.5 Giants

Up to this point, we have focused on identifying objects of known classes with anomalous light curves and understanding the underlying mechanisms that drive
Figure 5.19  Histograms of anomaly scores for different stages of giant stars. Note the dominant spike for the class in all cases and a small number of outliers in each plot, demonstrating an alternative approach to identifying outliers than simply a high anomaly score.

Figure 5.20  Example light curves from giant stars in blue, with a non-anomalous base light curve from TIC 261337074 shown in grey to highlight the magnitude of variations. The light curves cover a range of weirdness scores, with the selection reflecting the distribution seen in Figure 5.19. From top to bottom, Red giant weirdness = [0.9, 0.4, 0.3], red supergiants weirdness = [0.8, 0.4, 0.3], and blue supergiant weirdness = [0.9, 0.8, 0.3].
their classification as anomalies. However, we might miss additional insightful information by solely focusing on this anomalous/non-anomalous dichotomy. For instance, the anomaly score of predominantly non-anomalous classes may reveal extreme examples or variability patterns of interest that do not quite reach the anomalous threshold. Furthermore, the anomaly score may still correlate to physical processes within these classes. In this subsection, we explore the variability properties and distribution of anomaly scores for giant stars, namely the post-main sequence Red Giants and Red Supergiants, as well as the main sequence Blue Supergiants. These giants show mixed behaviour in terms of their mean anomaly score and are, therefore, a good target for understanding the relationship between the score and light curve attributes.

Stars occupying the giant branch are post-main-sequence stars with large surface areas resulting in an increased brightness. We focus here on previously identified red giants, red supergiants and blue supergiant stars. Figure 5.19 shows the distribution of anomaly scores for Giant stars. Red Giants and Red Supergiants are predominantly classified as non-anomalous, whereas the majority of Blue Supergiants have high anomaly scores, with most having a score higher than 0.9. This is also apparent in the Colour-Magnitude Diagram of Figure 5.7, where the RGB stars prominently show low anomaly scores.

Anomalous Blue Supergiants, the majority of stars of this type, are characterised by shorter variability timescales and larger variability amplitudes compared to most red giants (which have low anomaly scores), as shown in the right-hand column of Figure 5.20. They can also show a combination or rapid variability modulated by high amplitude variations in intermediate timescales (centre-right panel). As the anomaly score decreases, they start to resemble any other giants with a low score, and the light curve becomes dominated by a constant, low amplitude noisy continuum. These Blue Supergiants are undergoing an unstable stage in their evolution; they are young high-mass stars with high circumstellar dust fractions and low-frequency gravity waves (de Wit et al., 2014; Bowman et al., 2019). Red Giants and Supergiants, on the other hand, are intrinsically more stable, with variability patterns observed over months to decades dictated by slow pulsation mechanisms (Kiss et al., 2006). Anomalous luminous stars in the TIC are, therefore, likely to be Blue Supergiants.

Starting in late 2019, a Red Supergiant star, Betelgeuse, experienced a well-documented period of instability. This period consisted of a significant dimming spanning several months, leading to several proposed hypotheses to explain the
cause of the dimming (Joyce et al., 2020; Castelvecchi, 2020; Levesque & Massey, 2020). Remarkably, one of the two light curves of red supergiants that appear as anomalous outliers in the distribution shown in the upper right panel in Figure 5.19 corresponds to an observation of Betelgeuse in Sector 6 (the remaining outlier is an artefact), and has an anomaly score of 0.82. Betelgeuse’s light curve shows regular increases in brightness, as shown in the middle top panel of Figure 5.20. The observation predates the great dimming event by approximately ten months.

Unfortunately, Betelgeuse is not observed again by TESS until Sector 33, which occurs after the dimming event has concluded and the unusual periodic increases in brightness have subsided. No observations with different facilities and similar cadence or sensitivity are available during the time preceding and following the dimming event, and therefore we cannot verify when this anomalous pattern began. We note that the regular increases in luminosity prior to the dimming event suggest an intrinsic variability, which contradicts the dusty veil model presented in Montargès et al. (2021). Determining if the anomalous behaviour we have identified here is related to the major dimming event would require additional observations and comparisons to evolutionary models, which is outside the scope of this work. It is nonetheless remarkable that we can detect anomalous behaviour in the light curve of Betelgeuse shortly before the major dimming event.

As this approach relies on external classifications, the potential for discovery is restricted by the available databases. However, with the growing number of automated classifiers, e.g. Hinners et al. (2018), with Narayan et al. (2018) aiming for real-time transient classification for LSST data, the databases will soon provide sufficiently reliable results allowing the type of analysis in this work immediately after each data release, drastically increasing the effectiveness of this analysis.

5.7.6 White Dwarf Stars

White dwarfs are one of the individual groups of stars with consistently high anomaly scores. We have cross-matched our list of anomalies against the catalogue of white dwarfs in Gentile Fusillo et al. (2019), identifying 2195 examples with a high (> 0.7) probability of being a true white dwarf. This population represents less than 1% of the objects in our dataset. Of these, 90.8% (1995 examples) have an anomaly score above 0.9, and 98.1% (2153 examples)
Figure 5.21  CAMD colour-coded by anomaly score, zoomed in on the white dwarf population. There appears to be little to no correlation along the length of the cooldown track, indicating no correlation between the cooling stage and anomaly score.

Figure 5.22  Top Panel: Histogram of average anomaly scores for White Dwarf stars found in [Gentile Fusillo et al. (2019)]. Bottom four panels: Example light curves of White Dwarf stars showing an anomaly score above 0.9 despite a lack of visually defining features.
have a score above 0.8. The high prevalence of anomalous light curves among white dwarfs suggests a prevalent physical mechanism driving a unique type of variability among them.

In Figure 5.21, we show an inset of the HR diagram including only the WD branch, colour coded by anomaly scores higher than 0.9. There is no evidence of increasing anomaly with temperature or luminosity. White dwarf stars evolve through cooling, becoming less bright and redder over timescales comparable to the age of the universe. The lack of a correlation along the cooling track suggests that the mechanism causing an anomalous variability is not related to evolutionary changes in surface temperature. Amplitude variations are likely to be part of the reason for the WD anomaly, with variations of up to 10% in the normalized flux, as shown in the lower panel of Figure 5.22. These amplitude variations are significantly higher than those observed in MS stars of similar luminosities.

The lower panels of Figure 5.22 show anomalous white dwarf light curves. They reveal a rapid variability pattern of high amplitude and lack of periodicity, which is also confirmed by the periodograms. Althaus et al. (2010) explain in detail the pulsation periods of white dwarf stars, and Córnsico et al. (2019) further classifies a range of white dwarf pulsation types. These pulsations typically have short periods spanning 2-100 minutes, with amplitudes between 0.4 mmag and 0.3 mag. The periodogram shows little structure or clear power peaks, which indicates that the power spectrum is not sensitive to these short-period pulsations at the TESS cadence. However, the amplitude of the magnitude variations in these relatively dim stars are consistent with the pulsation types described in Córnsico et al. (2019), representing a likely candidate for the mechanism driving high anomaly scores in white dwarf stars.

The above discussion and the fact that white dwarfs are not present in the high variance population, described in §5.6.3, suggests that, at any luminosity or absolute magnitude, our method can identify additional white dwarf candidates. We can define white dwarf candidates as those with a high anomaly score above 0.9, BP-RP colour smaller than 1, and do not return a high variance between observations.
Figure 5.23  Example light curves of modulated light curves. These show a range of patterns with variations in both frequency and magnitude, suggesting multiple astrophysical mechanisms drive the variability.

Figure 5.24  A CAMD colour-coded by average anomaly score with the position of modulated light curves highlighted in green. Evidently, modulation patterns are present in many evolutionary stages and mass ranges, signalling multiple mechanisms causing such a variability pattern.
5.7.7 Modulated light curves

A fraction of the light curves in our sample show a combination of a higher frequency pulsation modulated by a lower frequency pulsation, as in the examples shown in Figure 5.23. We refer to these as modulated light curves. In total, we have identified 132 of these objects. There seems to be no preferential location for these objects in the HR diagram, except that no examples are found along the giant branch, as shown in Figure 5.24.

In Figure 5.23, we group objects in this class in sub-categories, as follows: the rapid sub-class shows fast variations within a broad modulation envelope; the modulated sub-class is similar to the previous sub-class, but shows a longer period (\( \sim 10 \text{ days} \)) for the lower of the two frequencies; finally, the sinusoidal sub-class shows a dominant, long-period sinusoidal pattern combined with a higher frequency variation of low amplitude. These sub-classes have somewhat blurry boundaries, with some of the light curves fitting in none or more than one sub-class. If more than one sub-class is suitable to describe an object, we label it with multiple types, e.g. rapid modulated sinusoidal.

Given that objects with this modulated pattern are widespread over most of the HR diagram, as well as the substantial differences between the sub-classes, it is unlikely that a single, intrinsic physical mechanism is behind the observed variability. These sources have a wide range of masses and evolutionary stages. Those which lie on the instability strip are likely RR-Lyrae stars undergoing the Blažkho effect (Blažko, 1907), a long-term modulation effect first observed in RW Draconis that have since been observed in other RR-Lyrae type stars. Despite being discovered over a century ago, this effect is poorly understood, with available models unable to fully explain the observed behaviour (Benkő et al., 2011; Skarka et al., 2020).

In Saylor et al. (2018), the authors search for low mass modulators in the SUPERBLINK catalogue observed during the Kepler K2 mission. The modulation pattern of GKM-dwarf-type stars is understood to be caused by the presence of starspots on fast rotators. This effect is well-modelled, but a limited number of examples have been found, with Saylor et al. (2018) identifying only 508 candidates from a catalogue containing 3 million objects. This study highlights the rareness of these types of phenomena and the importance of additional methods and catalogues like the ones presented here to identify further
candidates. Other studies also suggest sunspots as a possible cause for modulation in higher mass main sequence stars. See Balona (2011) for a detailed discussion.

While we do not explore other sources of modulation that certainly exist, we have highlighted that frequency modulation is related to astrophysical phenomena of interest. We also identify many candidates in the TIC using our unsupervised method, starting from a very general collection of light curves. In our catalogue (Table 5.2), we assign the *modulated* label to these candidates, with sub-class labels *sinusoidal* or *rapid* when appropriate.

### 5.7.8 Artefacts

Artefacts are rare features in light curves that do not have an astrophysical origin and are instead related to unintended instrumental or processing effects. Examples include hot pixels, baseline trends, detection gaps, or pipeline errors. Although artefacts do not represent astrophysical events, their identification is crucial in assessing the quality of catalogue and data releases (Lochner & Bassett, 2021). An unsupervised anomaly detection method cannot independently differentiate an artefact from a rare astrophysical phenomenon. To better
characterise the anomalies, it, therefore, becomes necessary to understand the artefacts associated with the data in question. TESS light curves are subject to several sources of artefacts, ranging from single pixel spikes to calibration issues that span significant fractions of the light curve. Discontinuity in the light curve due to a sudden change in the detector throughput is an example of the latter. The PDC pipeline produces the released TESS light curves by correcting for the typical sensor response curve, yet some artefacts inevitably remain. Despite our best efforts to filter out single-pixel spikes, some remain in the final light curves that we have analysed. In Figure 5.25, we show examples of common artefacts in TESS light curves.

Due to their often extreme effects and rarity within the dataset, artefacts can be assigned a very high anomaly score. This is particularly true when they result in discontinuities with timescales comparable with the duration of the observation, such as those seen in the left panels of Figure 5.25. Due to their distinct features (discontinuities, sudden variations in flux, etc.), these dominating patterns are identifiable by a visual inspection. On the other hand, single-pixel events do not dominate the anomaly score, as they usually account for a single feature. In fact, in §5.6.1 we have identified that light curves that resemble each other have very similar scores, even when one is affected by a single pixel spike.

Finally, some artefacts result from imperfect background subtraction in crowded fields. When a crowded field contains a variable star, the background subtraction process can be adversely affected. The background subtraction in such a field results in an inverted imprint of the variability onto the target light curve. For example, transiting systems (such as binaries or exoplanets) incorrectly subtracted with the background will appear as repeating flare-like features in the target light curve. With the correct symmetry, these artefacts can be indistinguishable from genuine astrophysical novelties, such as gravitational lensing events. Therefore, they need to be identified and individually assessed. The final column of Figure 5.25 contains examples of this phenomenon.

Fortunately, most artefacts are sector-dependent, particularly those associated with the detector response. Since artefacts result in a high anomaly score, they increase the standard deviation in the score for observations of the same object. It is, therefore, wise to identify artefacts from searches of anomalies that occur in single sectors, as we have discussed in §5.6.3. In Table 5.1, we have labelled identified artefacts in the Notes column.
5.8 Discussion

This work has exploited the ability of a particular unsupervised anomaly detection algorithm (the Unsupervised Random Forest, URF) to detect rare events in a set of PDC-processed TESS light curves with a varied and diverse form and astrophysical origin. We have compiled a catalogue of the anomalies found.

We have identified a broad range of anomalous variability patterns on TESS light curves. The method remains general and identifies many variability patterns without targeting a specific variability signature or astrophysical class. The resulting anomalies are also independent of whether the object is subject to previous studies or whether the astrophysical mechanisms are understood. Therefore, we have found a mix of unusual astrophysical behaviours (see §5.7.5), objects of known classes that show extreme or rare physical parameters (e.g. §5.7.2), as well as processing artefacts (§5.7.8) that are important to identify to produce a list of candidate astrophysical anomalies. The final catalogue can act as a repository of objects with outlying properties that can either form new stellar classes or inform astrophysical models of variability in known classes by adding examples of extreme cases.

Statistical analysis of the URF in §5.4 - 5.6 reveals a bimodal distribution of anomaly scores, with approximately 10% of objects identified as anomalous. This population is not necessarily only astrophysical oddities, rather they possess a behaviour that is statistically distant from the general population. The anomalous objects include a population of known stellar classes that show extreme variability patterns (e.g. eclipsing binaries with particularly deep eclipses, instability strip objects and irregular light curves in YSOs, described respectively in §5.7.2, 5.7.3 & 5.7.4); non-astrophysical anomalies (e.g. non-linear detector response curves, described in §5.7.8) and a fraction are the true unknown unknowns that demonstrate peculiar astrophysical behaviour, one such example is discussed in §5.7.5. Analysis of individual sources across multiple sectors reveals that the identification of periodic anomalous signals is consistent across all observations while providing a strategy to identify rare events that occur during a specific observation (see §5.6.3). These objects may include catastrophic events (such as supernovae) and variability (periodic or not) with timescales exceeding the duration of a single sector observation, such as an eclipsing binary with a large orbital separation or long-period variables.
We find a link between certain anomalous behaviours and their stellar evolutionary stage as traced by *Gaia* data, see §5.5. The population of *TESS* anomalous objects differs from those found in *Kepler* data in MG21. For example, red giants and supergiants score significantly higher in anomaly score in *Kepler* with respect to *TESS*. On the other hand, instability strip objects and blue supergiants appear anomalous in both datasets, while most of the main sequence stars have low anomaly scores for both *Kepler* and *TESS*. The comparison of both populations underscores an important aspect of anomaly detection; given a method for anomaly identification, the anomalous nature of a particular object is a function of the entire population and the specific observational parameters of the survey. Specifically, the high cadence *TESS* population is composed of stars typically closer and intrinsically brighter than those in the *Kepler* (see §5.2.1 for further details). This selection makes red giants less anomalous in the present work. Thus, even though, in both cases, we look at the periodograms and light curves as the features, the difference in the spectral range covered due to different cadences also affects the anomaly score of specific objects. Therefore, the analysis of anomaly scores in light curves should always be in the context of the population studied.

We also explored the link between the anomaly score and specific astrophysical configurations and found that a high anomaly score can indicate a rare or extreme configuration. For example, in subsection 5.7.2, we show that the primary transit depth and orbital period in binaries dominate the final anomaly score. In the context of *TESS* light curves, longer periods and deeper primary transits cause a higher anomaly score, which allows this work to increase the census of extreme examples within this object class. We also showed in subsection 5.7.4 that the irregular light curves resulting from obscuration in embedded YSOs have a high anomaly score and are the closest candidates to being analogues to the light curve of Boyajian’s star.

We further confirm our ability to detect genuine astrophysical anomalies by comparison with recent literature. Table 5.3 lists several anomalous objects recently identified. The majority of these “bona fide” anomalies are assigned an anomaly score of 0.9 or higher, with 90% of them having a score higher than 0.7. These objects range from quasi-periodic oscillations in a white dwarf to complex transits in multiple stellar systems. Only one of these objects is not assigned a high anomaly score: a rapidly oscillating Ap pulsator (TIC 264509538). This object has a pulsation period of 7.52 minutes (Shi et al., 2020), which is too short...
for our spectral analysis to pick it up.

Our work has also uncovered anomalous behaviour in well-studied sources that can be associated with their evolutionary stage or particular events during their evolution. One example is Betelgeuse, where we discovered an unusual variability pattern months before a major dimming event, see §5.7.5. The anomaly score for the Betelgeuse light curve is 0.85, much higher than other stars at a similar evolutionary stage. It remains unclear whether the anomalous variability is related to the subsequent dimming event, but the fact it is present so close to the event may indicate abnormal photospheric activity ten months before the major dimming event started.

## 5.9 Conclusion

We have presented the results of applying the Unsupervised Random Forest (URF) anomaly detection method to a large amount of TESS light curves characterised by their light curve points and frequency power spectrum, building on our previous work with Kepler data. We have provided a catalogue of anomalous light curves, classified those anomalies according to their variability characteristics, and associated their anomalous nature to any particular evolutionary stage or astrophysical configuration. For anomalies belonging to known classes (e.g., eclipsing binaries), we have also investigated what physical parameters drive the anomaly score. By comparing our results with those from the Kepler study, we have also studied how the anomalous nature of particular light curves depends on the characteristics of the general population itself. Here we summarise our main findings.

- Nearly 10% of the studied TESS light curves form a separate group of objects with enhanced anomaly scores. They include a mix of previously unclassified objects and objects with previously assigned classes with outlying properties and/or configurations.

- High amplitude variability, pulsations, rapid periodic patterns and long-term variability timescales that dominate the frequency spectrum are among the dominating properties that set the anomaly score for the different types of objects. As such, the method can serve as a detector of pulsating stars.
• Pulsating objects along the instability strip as well as M-dwarfs displaying flares, sunspots and other magnetic-related activity, are consistently among the most anomalous objects. Irregular light curves from young stars with significant circumstellar obscuration also rank high in the list of anomalies and constitute the closest analogues to Boyajian’s star.

• TIC stars are on average cooler, older, and less massive than those targeted by Kepler. This results in populations of giants, such as the red clump stars, being underrepresented in TESS and white dwarfs overrepresented in comparison with Kepler [Berger et al., 2020]. As a result, the distribution of anomalous objects over classes differs between the two populations. For example, the Giant Branch has a much lower average anomaly score in the TESS dataset, despite the method to find these anomalies being very similar in both cases.

• Anomalous eclipsing binaries and exoplanet transits have significantly deeper transits and longer orbital periods than their “normal” counterparts. The method can also detect deep transits with orbital periods longer than the typical light curve duration.

• Among giant stars, only Blue Supergiants show a high anomaly score, mostly driven by more irregular, high amplitude light curves, likely associated with their young evolutionary stage. Additionally, white dwarfs are consistently identified as anomalies, with no apparent relation between anomaly score and surface temperature or other physical parameters other than the amplitude of their variability, which is consistent with unresolved, short-period pulsations as described in [Corrêa et al., 2019].

• Instrumental and processing artefacts, such as single-pixel spikes and discontinuities in the light curve, are readily identified by our method, with the latter having a substantial impact on the anomaly score. They are sufficiently different from astrophysical phenomena to be easily identified, except for inverted transits that result from the inclusion of transiting objects in the background region of nearby targets, which could be misidentified as a lensing event.
Table 5.3  A list of known light curves of interest within the TESS literature and their associated Anomaly Scores.  

(1) - TIC ID from TESS catalogue  
(2) - Anomaly Score (this work)  
(3) - Notes on the physical nature of the system  
(4) - Previous references to the system

<table>
<thead>
<tr>
<th>TIC ID</th>
<th>Anomaly Score</th>
<th>Notes</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIC 21505340</td>
<td>0.96</td>
<td>Quasi-periodic oscillations in WD star.</td>
<td>Littlefield et al. (2021)</td>
</tr>
<tr>
<td>TIC 157376469</td>
<td>0.96</td>
<td>Binary system of K-type main sequence stars.</td>
<td>Pan et al. (2021)</td>
</tr>
<tr>
<td>TIC 63328020</td>
<td>0.95</td>
<td>dipole pulsation mode in the eclipsing binary.</td>
<td>Rappaport et al. (2021)</td>
</tr>
<tr>
<td>TIC 264509538</td>
<td>0.50</td>
<td>Pulsations of the Rapidly Oscillating Ap Star.</td>
<td>Shi et al. (2020)</td>
</tr>
<tr>
<td>TIC 229804573</td>
<td>0.95</td>
<td>compact hierarchical quadruple system.</td>
<td>Borkovits et al. (2021)</td>
</tr>
<tr>
<td>TIC 234523599</td>
<td>0.96</td>
<td>A giant planet, transiting an M3 dwarf star.</td>
<td>Bakos et al. (2018)</td>
</tr>
<tr>
<td>TIC 257459955</td>
<td>0.91</td>
<td>Pulsating helium-atmosphere white dwarf.</td>
<td>Bell et al. (2019)</td>
</tr>
<tr>
<td>TIC 278659026</td>
<td>0.73</td>
<td>g-mode hot B subdwarf pulsator.</td>
<td>Charpinet et al. (2019)</td>
</tr>
<tr>
<td>TIC 38586082</td>
<td>0.86</td>
<td>Peculiar variable star of alpha2 CVn type</td>
<td>Khalack et al. (2019)</td>
</tr>
</tbody>
</table>
Chapter 6

Conclusion & Future Work

This thesis presents projects demonstrating the typical challenges brought by the volume of incoming data from survey telescopes. Chapter 3 presents a machine learning pipeline to identify, validate, and provide age estimates of stellar clusters from astrometry and photometry provided by Gaia. By adopting machine learning models throughout, the final pipeline is the first fully scalable and repeatable approach. The strict focus on maintaining scalability resulted in the development of novel solutions, the most notable being the clustering approach with applications beyond those contained within this thesis. More than 2000 clusters are identified, within the Milky Way disc, with accompanying age estimates. These clusters offer many scientific use cases.

Nevertheless, there is continual potential to improve this work. The most beneficial is to include radial velocity measurements in the clustering analysis. The additional velocity dimension would reduce false membership within clusters by providing the entire three-dimensional picture. Unfortunately, radial velocity measurements for most Gaia sources are unavailable, although anticipated to be part of the upcoming data release, expected in 2024. Additionally, a further improvement to increase the potential scope of the results is to train the ageing model to estimate more physical properties, i.e. metallicity. Although, this does require a greater number of well-constrained clusters and potentially an observation campaign to measure properties directly. Despite the potential improvements, the results remain high quality, and by restricting all approaches to scalable methods, the pipeline is available when additional and updated data becomes available.
Chapter 2 expands on the novel clustering approach implemented during the previous chapter. By presenting this work in a general way, it highlights the application of this methodology on data beyond astronomy. Hence, the chapter focuses on the model as a stand-alone Python package. The package tackles the computational requirements for clustering vast datasets, which otherwise limits the repeatability of results. The package, HEADSS [Crake et al., 2023], is publicly available providing the ability to split, cluster and stitch the results through integration with the clustering algorithm HDBSCAN.

The additional functionality, removing the most likely edge effects for large clusters spanning a significant fraction of the feature space, represents a substantial improvement on the implementation used during the initial version identifying stellar clusters. Nonetheless, further improvements to the package HEADSS include integrating additional clustering algorithms and an option to partition the data by density. Nevertheless, the package offers the community a standard solution to a common issue that increases the transparency and repeatability of clustering big data.

Chapter 4 explores constraining the Initial-Final Mass Relation (IFMR) of white dwarf stars, presenting an astrophysical use case for the clusters identified in Chapter 3. I present an IFMR containing 108 new initial mass estimates, doubling the current census. This work contains previously unobtainable initial mass estimates by focusing on relatively small clusters within the Galaxy plane. As each cluster contains only a few white dwarfs, the examples probed offer a wider distribution of environments than previous studies. After presenting the first known correction for unresolved WD-WD binarity, the IFMR shows respectable agreement with the current literature and theoretical models covering an initial mass range of $2.1 - 8.1 \, M_\odot$ [Cummings et al., 2018; Choi et al., 2016; Marigo & Girardi, 2007]. The IFMR also identifies a population of white dwarfs that indicate an interrupted cooldown period, which suggests active fusion may have occurred within the hydrogen or helium layers. This observation is a significant result as it addresses an ongoing discussion within the community and offers the first evidence of a potential fusion history within DB-type white dwarfs.

To further constrain the IFMR and verify the presence of features, addressing the improvements discussed for Chapter 3 provides more accurate cluster age estimates, leading to more reliable initial mass estimates. Furthermore, a targeted campaign to improve the measured observational properties of white dwarf stars identified within stellar clusters by collecting high-resolution spectra would
significantly benefit the final IFMR. These measurements would allow verification of the population of white dwarfs with apparent ongoing fusion. Alternatively, the methods are readily transferable to future data releases offering greater accuracy and allowing a more distant search, further constraining the IFMR.

The clusters identified in Chapter 3 offer a range of potential avenues. An exciting possibility would be to attempt to constrain the IFMR of other stellar remnants, namely Neutron stars. Achieving such a relation would require extremely accurate cluster ageing estimates due to only high-mass stars evolving into Neutron stars. Nonetheless, a similar approach to those contained in Chapter 3 & 4 would assist in constraining the equation of state of degenerate neutron gas \cite{Lattimer2011, Sumiyoshi2022}.

Finally, in Chapter 5 I present a second project exploring the detection of anomalous behaviour in stellar light curves produced by TESS. The analysis attempts to quantify how anomalous a given light curve is with respect to the rest of the population. I find that nearly 10% of light curves belong to a population with enhanced anomalous behaviour. Combining the anomaly metric with photometric data identifies a strong correlation between the evolutionary stage and the anomaly score, indicating that the unsupervised approach is indeed probing genuine astrophysical phenomena. This discovery led the analysis to explore the astrophysical mechanisms that drive anomalous behaviour. Searching within particular stellar classes enhances the possibility of identifying genuinely abnormal behaviours. The most significant example is the identification of Betelgeuse as an anomalous object several months prior to a considerable dimming event \cite{Joyce2020, Castelvecchi2020}. This work led to the publication \cite{Crake2023}.

The major limitation of this work is the requirement for each light curve to be sampled identically in time. Due to the design of TESS light curves, many objects have identical sampling rates, yet, this is not the standard approach for other telescopes. Removing the requirement for identical time sampling would allow the methodology to apply to a range of datasets, including the upcoming Vera C. Rubin and Gaia DR4.

As for the outlier detection approach, this methodology brings an abundance of possibilities. As discussed, addressing the requirement of identical time sampling would allow the examination of several time-domain datasets beyond those from TESS. Furthermore, analysing stellar spectra with the Unsupervised Random
Forest (URF) offers an alternative science use case. Within Baron & Poznanski (2017), the URF is used to identify anomalous galaxy spectra. With Gaia releasing low-resolution spectra for all objects within the next data release, the URF has the potential to identify objects with a unique spectral signature. Spectra are particularly suited to this type of analysis as all sources are observed with identical wavelengths.

Overall, the era of survey telescopes and big data astronomy promises to revolutionise our understanding of the universe around us. By applying many of the techniques described within this thesis, I believe the field will be able to adapt and overcome the challenges presented by the sudden increase in data. Of course, there are many additional challenges to address, and the work contained within this thesis alone has discussed several possible avenues and projects for further investigation.
Appendix A

Definitions of Key Parameters and Data
<table>
<thead>
<tr>
<th>Feature</th>
<th>Feature Description</th>
<th>unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>parallax_error</td>
<td>Standard error of stellar parallax</td>
<td>Angle [mas]</td>
</tr>
<tr>
<td>parallax_over_error</td>
<td>Parallax divided by its standard error</td>
<td>-</td>
</tr>
<tr>
<td>pmra</td>
<td>Proper motion in right ascension</td>
<td>Angular Velocity [mas/year]</td>
</tr>
<tr>
<td>pmdec</td>
<td>Proper motion in declination</td>
<td>Angular Velocity [mas/year]</td>
</tr>
<tr>
<td>pmra_error</td>
<td>Standard error of proper motion in right ascension direction</td>
<td>Angular Velocity [mas/year]</td>
</tr>
<tr>
<td>pmdec_error</td>
<td>Standard error of proper motion in declination direction</td>
<td>Angular Velocity [mas/year]</td>
</tr>
<tr>
<td>phot_g_mean_flux_over_error</td>
<td>G-band mean flux divided by its error</td>
<td>-</td>
</tr>
<tr>
<td>astrometric_sigma5d_max</td>
<td>The longest semi-major axis of the 5-d error ellipsoid</td>
<td>Angle [mas]</td>
</tr>
<tr>
<td>astrometric_excess_noise</td>
<td>Excess noise of the source</td>
<td>Angle [mas]</td>
</tr>
<tr>
<td>astrometric_excess_noise_sig</td>
<td>Significance of excess noise</td>
<td>-</td>
</tr>
<tr>
<td>visibility_periods_used</td>
<td>Number of visibility periods used in Astrometric solution</td>
<td>-</td>
</tr>
<tr>
<td>ruwe</td>
<td>Renormalised unit weight error</td>
<td>-</td>
</tr>
<tr>
<td>astrometric_gof_al</td>
<td>Goodness of fit statistic of model wrt along-scan observations</td>
<td>-</td>
</tr>
<tr>
<td>ipd_gof_harmonic_amplitude</td>
<td>Amplitude of the IPD GoF vs position angle of scan</td>
<td>-</td>
</tr>
<tr>
<td>ipd_frac_odd_win</td>
<td>Percent of transits with truncated windows or multiple gate</td>
<td>[%]</td>
</tr>
<tr>
<td>ipd_frac_multi_peak</td>
<td>Percent of successful Image Parameters Determination windows with more than one peak</td>
<td>[%]</td>
</tr>
</tbody>
</table>

**Table A.1**  
(1) - Feature from Gaia Catalogue  
(2) - Column description from *Hambly et al.* (2021)  
(3) - Units  
*Descriptions of key Gaia DR3 columns in §3.2. See §13.1 of *Hambly et al.* (2021) for further description of all Gaia columns.*
<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>maxIter</td>
<td>Maximum number of iterations</td>
<td>100</td>
</tr>
<tr>
<td>blockSize</td>
<td>Block size for stacking input data in matrices</td>
<td>128</td>
</tr>
<tr>
<td>stepSize</td>
<td>Step size for each iteration of optimisation</td>
<td>0.03</td>
</tr>
<tr>
<td>solver</td>
<td>Solver algorithm for optimisation</td>
<td>l-bfgs</td>
</tr>
<tr>
<td>layers</td>
<td>Number of nodes per layer from input to output</td>
<td>-</td>
</tr>
</tbody>
</table>

**Table A.2** (1) - Hyperparameter  
(2) - Column descriptions from Meng et al. (2016)  
(3) - Default values.  
Description of PySpark’s MultilayerPerceptronClassifier key parameters in §3.2. See PySpark docs for further description of all available parameters.

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>min_cluster_size</td>
<td>The minimum size of clusters</td>
<td>5</td>
</tr>
<tr>
<td>min_samples</td>
<td>Number of nearby samples for a point to be a core point</td>
<td>None</td>
</tr>
<tr>
<td>allow_single_cluster</td>
<td>Allow the return of a single cluster</td>
<td>False</td>
</tr>
<tr>
<td>cluster_selection_method</td>
<td>The method to select clusters from the condensed tree</td>
<td>eom</td>
</tr>
<tr>
<td>prediction_data</td>
<td>generate data for predicting labels of new unseen points</td>
<td>False</td>
</tr>
<tr>
<td>gen_min_span_tree</td>
<td>generate the minimum spanning tree for later analysis</td>
<td>False</td>
</tr>
</tbody>
</table>

**Table A.3** (1) - Hyperparameter  
(2) - Column descriptions from McInnes et al. (2017)  
(3) - Default values.  
Description of key parameters of HDBSCAN in §3.3. See HDBSCAN docs for further description of all available parameters.
<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>sample_weight</td>
<td>Training weights to calculate the loss function</td>
<td>None</td>
</tr>
<tr>
<td>epochs</td>
<td>Number of training epochs</td>
<td>1</td>
</tr>
<tr>
<td>early_stop</td>
<td>Stop training when the loss metric stops improving</td>
<td>False</td>
</tr>
<tr>
<td>patience</td>
<td>Number of epochs with no improvement before training stopped</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table A.4**

(1) - Hyperparameter
(2) - Column descriptions from Abadi et al. (2015)
(3) - Default values.
Description of TensorFlow’s Sequential key parameters in §3.4.3. See Keras Sequential docs & Keras.Callbacks.EarlyStopping docs for further description of all available parameters.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Feature Description</th>
<th>unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>parallax</td>
<td>Absolute stellar parallax $\pi$</td>
<td>Angle [mas]</td>
</tr>
<tr>
<td>phot.g.mean_flux</td>
<td>Mean flux in the G-band</td>
<td>Flux [es$^{-1}$]</td>
</tr>
<tr>
<td>phot.bp.mean_flux</td>
<td>Mean flux in the BP-band</td>
<td>Flux [es$^{-1}$]</td>
</tr>
<tr>
<td>phot.rp.mean_flux</td>
<td>Mean flux in the RP-band</td>
<td>Flux [es$^{-1}$]</td>
</tr>
<tr>
<td>phot.g.mean_mag</td>
<td>Mean magnitude in the G-band</td>
<td>Magnitude [mag]</td>
</tr>
<tr>
<td>phot.bp.mean_mag</td>
<td>Mean magnitude in the BP-band</td>
<td>Magnitude [mag]</td>
</tr>
<tr>
<td>phot.rp.mean_mag</td>
<td>Mean magnitude in the RP-band</td>
<td>Magnitude [mag]</td>
</tr>
<tr>
<td>bp_rp</td>
<td>BP - RP colour</td>
<td>Magnitude [mag]</td>
</tr>
<tr>
<td>bp_g</td>
<td>BP - G colour</td>
<td>Magnitude [mag]</td>
</tr>
<tr>
<td>g_rp</td>
<td>BP - RP colour</td>
<td>Magnitude [mag]</td>
</tr>
</tbody>
</table>

**Table A.5**

(1) - Feature from Gaia Catalogue
(2) - Column description from Hambly et al. (2021)
(3) - Units
Descriptions of key Gaia DR3 columns in §4.2.1. See §13.1 of Hambly et al. (2021) for further description of all Gaia columns.
<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>numTrees</td>
<td>Number of trees in the forest</td>
<td>20</td>
</tr>
<tr>
<td>featureSubsetStrategy</td>
<td>Number of features evaluated for a split</td>
<td>“auto”</td>
</tr>
<tr>
<td>impurity</td>
<td>Measure of the homogeneity of the labels within a node</td>
<td>“gini”</td>
</tr>
<tr>
<td>minInstancesPerNode</td>
<td>Minimum samples for a child node after a split</td>
<td>1</td>
</tr>
</tbody>
</table>

Table A.6 (1) - Hyperparameter  
(2) - Column descriptions from Meng et al. (2016)  
(3) - Default values.  
Description of key parameters from PySpark’s RandomForestClassifier in § 4.2.1. See PySpark docs for further description of all available parameters.

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>n_estimators</td>
<td>Number of trees in the forest</td>
<td>100</td>
</tr>
<tr>
<td>max_features</td>
<td>Number of features evaluated for a split</td>
<td>“sqrt”</td>
</tr>
<tr>
<td>max_depth</td>
<td>Maximum depth of a single tree</td>
<td>None</td>
</tr>
<tr>
<td>min_samples_split</td>
<td>Minimum samples for an internal node</td>
<td>0</td>
</tr>
<tr>
<td>min_samples_leaf</td>
<td>Minimum samples for a final (leaf) node</td>
<td>1</td>
</tr>
<tr>
<td>Bootstrap</td>
<td>Bootstrap samples when building trees?</td>
<td>True</td>
</tr>
<tr>
<td>Warm Start</td>
<td>If True, reuse solution from previous training</td>
<td>False</td>
</tr>
</tbody>
</table>

Table A.7 (1) - Hyperparameter  
(2) - Column description from Pedregosa et al. (2011)  
(3) - Default values.  
Description of scikit-learn’s RandomForestClassifier key parameters in § 2.3.2. See scikit-learn docs for further description of all available parameters.
Bibliography


Avvakumova E. A., Malkov O. Y., Kniazev A. Y., 2013, Astronomische Nachrichten 334, 860


Blažko S., 1907, Astronomische Nachrichten, 175, 325


Breger M., 2000, in Delta Scuti and Related Stars. p. 3


Carroll B. W., Ostlie D. A., 2006, *An introduction to modern astrophysics and cosmology*

Castelvecchi D., 2020, Mysterious faded star Betelgeuse has started to brighten again, https://www.nature.com/articles/d41586-020-00561-z


Chollet F., et al., 2015, Keras, https://keras.io


Hertzsprung E., 1911, Publikationen des Astrophysikalischen Observatoriums zu Potsdam, 63


Hough P. V. C., 1962, Method and means for recognizing complex patterns


Hughes D. W., 2006, Journal of Astronomical History and Heritage, 9, 173


Kapteyn J. C., 1902, Publications of the Kapteyn Astronomical Laboratory Groningen, 11, 1
Karypis G., Han E.-H., Kumar V., 1999, Computer, 32, 68
Kounkel M., Covey K., 2019, AJ, 158, 122

219


Lindgren L., et al., 2021, \textit{A&A} \textbf{649}, A4


Liu D. C., Nocedal J., 1989, Mathematical programming, 45, 503


Lochner M., Bassett B., 2021, \textit{Astronomy and Computing}, 36, 100481


Lundmark K., 1932, \textit{Handbuch der Astrophysik} \textbf{5}, 210


Marrese, P. M. Marinoni, S. Fabrizio, M. Altavilla, G. 2019, \textit{A&A} \textbf{621}, A144


Rayleigh L., 1899, XXXIV. On the transmission of light through an atmosphere containing small particles in suspension, and on the origin of the blue of the sky, doi:10.1080/14786449908621276, https://doi.org/10.1080/14786449908621276


Russell H. N., 1914, Popular Astronomy, 22, 275

Shanker M., Hu M. Y., Hung M. S., 1996, Omega, 24, 385
Sieranoja S., Fränti P., 2019, Pattern Recognition Letters, 128
Singh K., Malik D., Sharma N., 2011, IJCEM International Journal of Computational Engineering & Management ISSN, 12, 2230


Steinley D., 2006, \textit{The British journal of mathematical and statistical psychology}, 59, 1


VanderPlas J. T., 2018, \textit{ApJS}, 236, 16


Waykole S., Shiwakoti N., Stasinopoulos P., 2021, \textit{Sustainability}, 13


224

Xu R., Wunsch D. C., 2010, IEEE reviews in biomedical engineering, 3, 120


Zhang Y., Zhao Y., 2015, *Data Science Journal*, 14, 11


van Steen M., Tanenbaum A. S., 2016, Computing, 98, 967
