2021

Stellar classification of folded spectra using the MK Classification scheme and convolutional neural networks

John Magee
Technological University Dublin

Stellar classification of folded spectra using the MK Classification scheme and convolutional neural networks

John Magee

A dissertation submitted in partial fulfilment of the requirements of
Technological University Dublin for the degree of
M.Sc. in Computing (Data Analytics)

June, 2021

Declaration

I certify that this dissertation which I now submit for examination for the award of MSc in Computing (Data Analytics), is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

This dissertation was prepared according to the regulations for postgraduate study of the Technological University Dublin and has not been submitted in whole or part for an award in any other Institute or University.

The work reported on in this dissertation conforms to the principles and requirements of the Institute's guidelines for ethics in research.

Signed: John Magee

Date: 16 June 2021

Abstract

The year 1943 saw the introduction of the Morgan-Keenan (MK) classification scheme and this replaced the existing Harvard Classification scheme. Both stellar classification scheme are fundamentally grounded in the field of spectroscopy. The Harvard Classification scheme classified stars based on stellar surface temperature. The MK Classification scheme introduced the concept of a luminosity class that is intrinsically linked to the surface gravity of a star. Temperature and luminosity class values are estimated directly from the stellar spectrum.

Machine learning is a well-established technique in astronomy. Traditionally, a spectrum is treated as a one-dimensional sequence of data. Techniques such as artificial neural networks and principal component analysis are commonly used when classifying spectra. Recent research has seen the application of convolutional neural networks in this domain.

This research investigates the effectiveness of using convolutional neural networks with folded spectra. Robust experimental and statistical techniques were used to test this hypothesis. The result show that folded spectra and 2D convolutional neural networks obtained a higher average classification accuracy when compared to spectra processed with a 1D convolutional neural network. A ResNet V2 50 architecture was also included in this experiment, but the results show that it did not match the performance of shallower network architecture.

All data used in this research has been archived on github and is available by following this link https://github.com/D18124324/dissertation

Keywords: stellar classification, MK, convolutional neural network, folded spectra

Acknowledgments

I would like to express my profound thanks and appreciation to Patricia O'Byrne for her support and guidance over the last four months. I shudder to think how many cups of tea were thrown against the wall in frustration when reviewing the numerous drafts of this manuscript. They were not sacrificed in vain, and I hope that you are as proud of this final manuscript as I am.

I am forever grateful to my father Sean Magee and friends Thomas Finney and Ross Farrell for taking the time to help review this manuscript. Their feedback was invaluable.

I would also like to convey my thanks to all the Technological University Dublin staff that have supported me over the past three years.

Finally, I would like to express my thanks to my Daon colleagues Clive Bourke and Paul Kenny who have supported me on this journey to complete this MSc programme.

Contents

List of Figures

List of Tables

List of Acronyms

APOGEE	Apache Point Observatory Galactic Evolution Experiment
AU	Astronomical Unit
BD	Bonner Durchmusterung stellar catalogue
CFLib	The Indo-U.S. Library of Coudé Feed Stellar Spectra
CNN	Convolutional neural network
FITS	Flexible Image Transport System
HTTP	Hubble Tarantula Treasury Project
JHC	Jacoby-Hunter-Christian Atlas
LAMOST	Large Sky Area Multi-Object Fibre Spectroscopic telescope
HD	Henry Draper catalogue
MK	Morgan–Keenan stellar classification scheme
NGC	New General Catalogue
RGB	Red Giant Branch star
SC	Silva & Cornell Altas
SDSS	Sloan Digital Sky Survey
SNR	Signal-to-noise ratio

Chapter 1

Introduction

There are an estimated 200 billion stars in the Milky Way galaxy (Karttunen et al., 2006, p. 7). Such large numbers of stellar objects mean automated classification is a necessity for astronomers, machine learning being well established in the field of astronomy (Fluke & Jacobs, 2020). Given the large distances to stars, the only data available for study is the stellar spectrum. The MK Classification Scheme was devised in 1943 and is still in use today (Morgan et al., 1943). This scheme classifies a star based on a main class, sub-class and a luminosity class. Traditionally, classification is a manual process of extracting stellar indices from a spectrum, estimating the strength and width of emission lines, and comparing these to existing known stellar classes (Karttunen et al., 2006). The amount of data collected by astronomical surveys continues to grow at a rate that requires automated processing, as the manual process simply cannot scale to a level required to process all this new data. The application of machine learning to stellar classification is established and continues to expand, most recently using convolutional neural networks for stellar classification (Jiang et al., 2020; Sharma et al., 2019), but other methods like principal component analysis remain popular (Bazarghan, 2008).

1.1 Background

The first analysis of the stellar spectrum was by Gustav Kirchhoff in 1860 (Corbally & Gray, 2018). Since then, astronomers have come a long way and the classification scheme in use today is the MK Classification Scheme. Classification of stellar spectra is fundamentally grounded in the field of spectroscopy. Using a spectrograph, astronomers can deduce a large amount of information just from starlight. The most frequent features obtained from stellar spectra are chemical abundances and physical properties. Important physical properties include surface gravity (log g), surface temperature (T_{eff}) and metallicity (Fe/H) (Binney & Merrifield, 1998).

In 1918, the Henry Draper Catalogue (HD) was published (Cannon & Pickering, 1918) and this used the Harvard Spectral Classification scheme, the precursor of the MK Classification Scheme. In this scheme, stars are classified based solely on their surface temperature (T_{eff}). T_{eff} is derived through analysis of the strengths of spectral lines. Stars are assigned a main class and a sub-class. The main class is a single letter, and the sub-class is a number from 0 to 9. The sequence of the main class, in descending order of temperature, is O, B, A, F, G, K and M, class O represents the hottest stars and M represents the coolest. The 10 sub-classes indicate decreasing temperature (Karttunen et al., 2006).

In 1943 the Morgan-Keenan Classification (MK) system was introduced (Morgan et al., 1943). This extends the existing Harvard Classification scheme, and it includes surface gravity (log g) to estimate the luminosity class of a star. The scheme retains the main and sub classes defined by the Harvard Classification scheme and adds the luminosity class, to represent the intrinsic brightness of a star. The luminosity class is represented as Roman numerals from I (most luminous super-giants) to VII (white dwarfs) (Greene & Jones, 2004).

1.2 Research Project/problem

Modern astronomical surveys capture massive amounts of data. For example, the LAMOST telescope can capture spectra for up to 4000 different objects in a single

exposure (Wang et al., 2017). It is impossible for the traditional manual approach of stellar classification to scale to this level. It is also highly undesirable to leave such massive amounts of data unprocessed, there could be a discovery in the data waiting to be found.

Modern machine learning techniques have been applied to the domain of stellar classification, and this is an established field (Fluke & Jacobs, 2020). Popular methods include artificial neural networks and principal component analysis. More recent research is applying convolutional neural networks to this domain (Jiang et al., 2020; Sharma et al., 2019). Typically, stellar spectra is always treated as sequential data. For example, the first use of convolutional neural networks to this domain use a one-dimensional kernel to process the input spectrum.

Convolutional neural networks have achieved state-of-the-art classification accuracy when processing images (Kelleher, 2019). There are two aims for this research; one is to show that the advances in convolutional neural networks in the domain of computer vision can be applied to the domain of automated stellar classification. The second is to show that stellar spectra can be folded into a two-dimensional representation and processed by a two-dimensional convolutional neural network.

1.3 Research Objectives

The objectives of this research are:

1. Review the current literature in the domain of automated stellar classification using the MK Classification scheme.
2. Determine the state-of-the-art classifier.
3. Obtain the data set(s) necessary to reproduce the state-of-the-art classifier and attempt to reproduce the results in the literature.
4. Determine a suitable modification to the existing state-of-the-art classifier that can be tested experimentally in a robust manner and present this as the hypothesis.
5. Present the null hypothesis.
6. Design an empirical experiment to test the hypothesis.

7. Design and train a classification model to classify stellar spectra based on the MK Classification Scheme.
8. Evaluate the predictions obtained from the classification model and use robust statistical methods to determine the statistical confidence in the results.
9. Present the results and accept/reject the null hypothesis.

1.4 Research Methodologies

This research is empirical, quantitative, and secondary research with inductive reasoning. It involves systematic review, summary, and extension of existing knowledge in the published literature.

- Four data sets are used in this research, these are existing data sets collected as part of third-party research, so this is secondary research.
- Quantitative methods are used to evaluate the accuracy of classification models trained as part of this research.
- This research is Empirical Research as it is using data to answer a specific question, it is being tested using a hypothesis with a prediction and is being tested by experiment.
- Inductive Reasoning is used in this research as it is using neural networks to construct a model that generalizes the training data (going from the specific to the general)

1.5 Scope and Limitations

The scope of this research is the use of convolutional neural networks for stellar classification using the MK Classification scheme. Four data sets are used in this experiment. The first data set is the Jacoby-Hunter-Christian Atlas, created in 1984, and it contains 161 spectra. The second data set is the MILES data set, captured between 2000 and 2001, and it contains 985 spectra. The Indo-U.S. Library of Coude Feed Stellar

Spectra (CFLIB) data set was captured between 1995 and 2003 and it contains 1273 spectra. The ELODIE data set consists of 908 spectra. Issues were encountered obtaining the catalogue associated with this data set so a random selection from this data set is used in this research. Each data set represents spectra for all classes from O to M in the MK Classification system. Where object classifications are unclear, the SIMBAD database is used as a source of truth.

This research is limited to MK Classification of stars using normalised spectra based on the MK main class. The main classes are O, B, A, F, G, K and M. This research is using two dimensional CNN classifiers and folded spectra. Spectra are normally presented as a sequence of spectral flux values, whereas folded spectra represents spectra in a two-dimensional format, allowing them to be processed as if they were images. The assumption behind this strategy is that CNNs, being state-of-the-art image classifiers, will also improve the accuracy of stellar classification when spectra are represented in the folded representation. The spectra are processed using an existing procedure and it is assumed that it is appropriate for the data used in this research. The risks associated with this assumption appear low as this research is using an existing normalisation process with existing data sets. This research also assumes the published MK classification of objects in data sets is correct.

1.6 Document Outline

The remaining chapters of this research are structured as follows:

Chapter 2 - Review of existing literature

This chapter contains an introduction to the domain of stellar classification and classification schemes. It is intended to inform the reader of necessary background knowledge necessary to evaluate this research. It then presents the existing literature in the domain of automated stellar classification using the MK Classification scheme. Finally, it identifies gaps in the literature and presents the research question.

Chapter 3 - Design and Methodology

This chapter presents the experiment design. It includes the spectral data sets used in

this research, data cleaning, data normalization and the proposed experiment methodology. It concludes with the statistical evaluation methodology used to test the hypothesis.

Chapter 4 - Results, Evaluation and Discussion

This chapter presents the results of the experiment. The result of the proposed solution is compared to the baseline solution. The strengths and limitations of the proposed solution are discussed to highlight areas of improvement.

Chapter 5 - Conclusion

In this final chapter, a summary of the research is presented, including key findings, conclusions, and areas for future research.

Chapter 2

Review of existing literature

2.1 Introduction

This chapter introduces the research on stellar classification. Stellar classification is fundamentally grounded in the field of spectroscopy; therefore, this chapter begins with an introduction of spectroscopy, followed by stellar classification schemes. Additional background reading material on stars is contained in the Appendix in section A.4. A literature review of the techniques used for stellar classification is presented, followed by the gaps identified in the literature. Finally, the research question is presented.

2.2 Spectroscopy

Spectroscopy is the study of spectra. A spectrum is generated from a light source by passing the emitted electromagnetic radiation through a spectrograph, an instrument containing a glass prism or diffraction grating. The electromagnetic radiation is broken into its constituent wavelengths and this is recorded by a photo-sensitive medium, typically a charged couple device (CCD) in modern observatories, and the result is called a spectrum. In astronomy, wavelength is typically represented by the symbol λ and measurements are reported in units of Angstroms, Å, (10^{-10}m) (Greene & Jones, 2004; Ohanian, 1989).

In spectroscopy, electromagnetic radiation is referred to as spectral flux and the

amount of spectral flux recorded by the CCD is known as spectral flux density. Spectral flux density is typically measured in units of Watts (1W = $1Js^{-1}$), or Ergs (1erg = $1x10^{-7}J$), at a specific wavelength over a unit area ($1m^2$). This is typically reported in units scaled to the wavelength region of interest e.g. $Wm^{-2}\mu^{-1}$ for the infrared region or $Wm^{-2}nm^{-1}$ for the visible region. Relative spectral flux density is a process to scale the actual flux density relative to a reference value. This reference value is arbitrary and is chosen by the astronomer (Karttunen et al., 2006).

2.2.1 Types of Spectra

There are three distinct types of spectra:

1. Emission Spectrum. A sparse spectrum showing distinct emission lines that are represented as isolated bright lines. These lines characterise the elements responsible for the emission of the spectral flux. An example of an emission spectrum is shown at the top of Figure 2.1[1].

2. Absorption Spectrum. This spectrum shows distinct dark lines, indicating the presence of elements that have absorbed light. An example of an absorption spectrum is shown at the bottom of Figure 2.1.

3. Continuous Spectrum. This is a spectrum where spectral flux is emitted over an unbroken range of wavelengths. Continuous spectra are the spectra that are associated with starlight.

Emission and absorption spectra are both examples of line spectra as they are characterised by the presence or absence of lines at specific wavelengths. Figure 2.1 is a traditional representation of a spectrum, where spectral flux is represented in colours. Modern stellar spectra are plotted as graphs. The presence or absence of lines at different wavelengths that can be grouped together to identify an element are known as spectral indices. The most commonly used spectral indices were published

[1]Original image sourced from
http://khadley.com/Courses/Astronomy/ph$_2$05/*topics/lightTelescopes/spectroscopy.html*

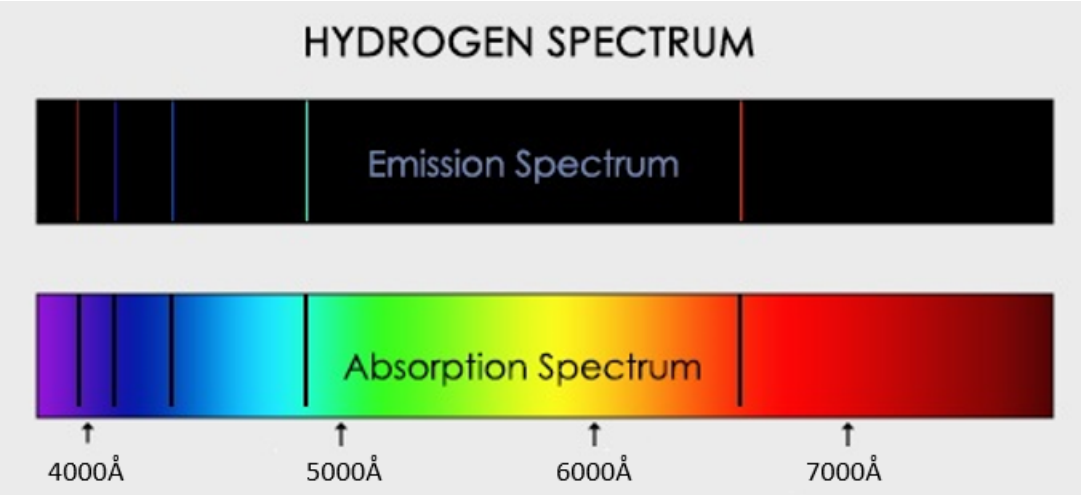

Figure 2.1: Emission and Absorption spectrum of Hydrogen. Image credit: khadley.com.

by Faber et al. (1985) and they define the indices wavelengths and expected intensity (Binney & Merrifield, 1998).

Another type of spectrum is the black body spectrum. This is a specific type of emission spectrum from a thermal source. A thermal source is defined as any object that emits electromagnetic radiation because of its temperature, the amount of electromagnetic radiation emitted increasing with temperature (Greene & Jones, 2004). Thermal sources are said to be opaque as the electromagnetic radiation emitted inside the object is not immediately released. Instead, it is re-absorbed and emitted many times before it eventually leaves the thermal source. Greene and Jones (2004) describes this as electromagnetic radiation being trapped within the thermal source. The effect of the continued emission and absorption effectively hides any spectral signature in the electromagnetic radiation. When a black body spectrum is plotted as a graph, it is represented as a perfect continuous curve, showing increasing spectral flux intensity that reaches a peak at a specific wavelength, before the spectral flux intensity decreases. Figure 2.2[2] shows black body spectra for objects of different temperatures. Black body spectra are very useful in astronomy as stars are very good approximations of black body objects (Karttunen et al., 2006).

[2]Original image sourced from https://cnx.org/contents/pZH6GMP0@1.409:OjoA1o37@3/Blackbody-Radiation

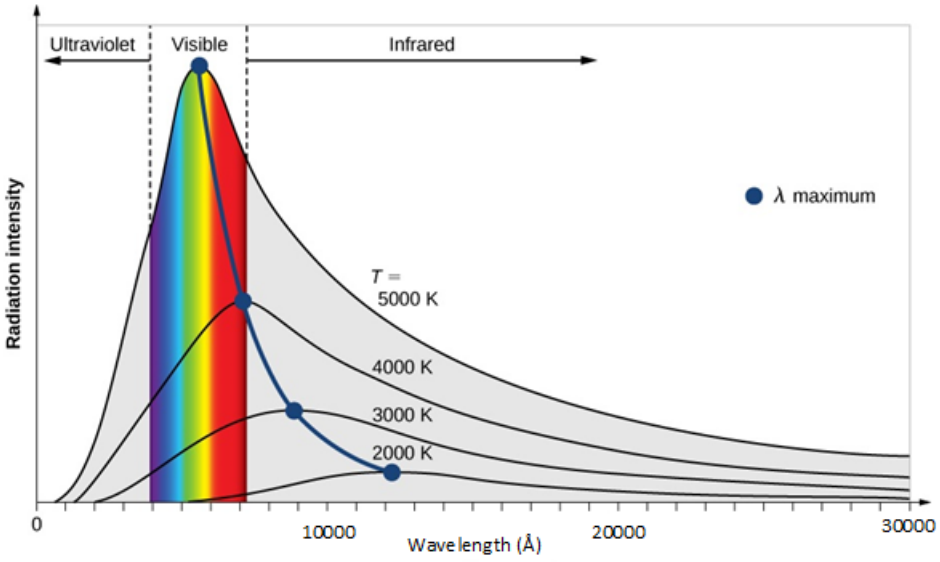

Figure 2.2: Examples of black body spectra. λ maximum represents the wavelength associated with the peak radiation intensity. T represents the temperature of the black body object, measured in Kelvin (K). Image credit: cnx.org.

2.2.2 Stellar Spectra

The last section introduced the different types of spectra and how each spectrum type can be interpreted e.g. peak temperature, element presence/absence etc. For stellar spectra, however, things are not so simple. Stars are a good representation of black body objects, but they do not produce perfect black body spectra. The atmospheres of stars are less dense than the core, and elements in the atmosphere generate absorption and emission lines that are visible in the spectrum. As a result, stellar spectra appear predominantly as black body spectra with distinct emission and absorption lines (Karttunen et al., 2006). Figure 2.3[3] is a graphic representing a black body spectrum showing Hydrogen absorption lines.

Figure 2.4 shows a real spectrum for star HD242908. This shows a black body curve with distinct absorption lines. In Figure 2.4, the spectral flux is measured in a unit called FLAM, a popular unit in astronomy defined as ergs per cm^2 per second per Angstrom (ergs/cm^2/s/Å) (Binney & Merrifield, 1998).

[3]Original image sourced from https://www.physicsforums.com/attachments/absorption-spectrum-of-hydrogen-gas-jpg.231851

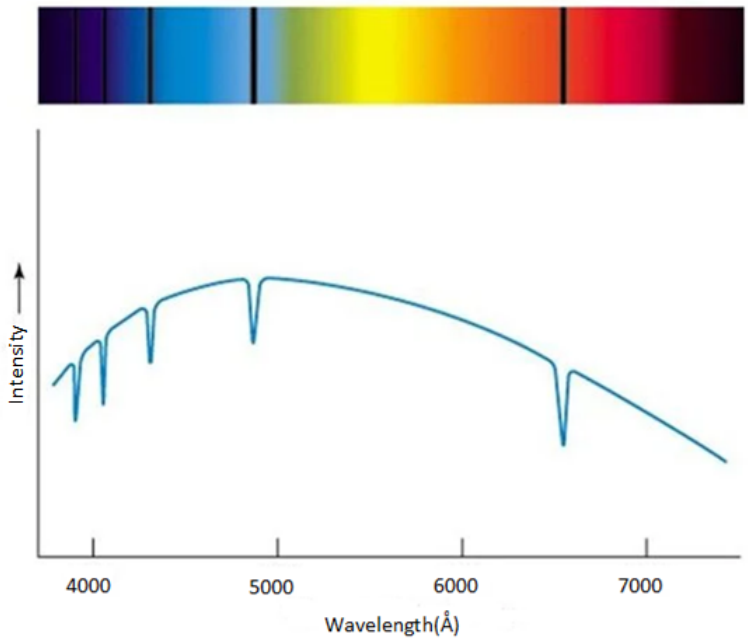

Figure 2.3: Black body spectrum showing Hydrogen absorption lines. Image credit: physicsforums.com.

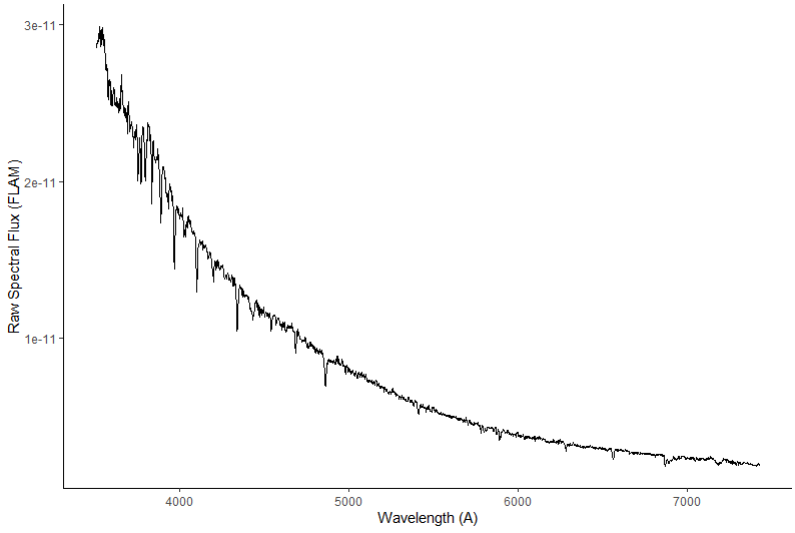

Figure 2.4: Spectrum of star HD242908 from the Jacoby-Hunter-Christian Atlas.

2.3 Stellar Classification Schemes

In 1860, Gustav Kirchhoff began interpreting dark lines in spectra generated from sun light. The spectra were captured by Joseph von Fraunhofer in Munich four years

earlier. Fraunhofer was the first ever observer of stellar spectra. These first tentative steps were the beginning of the process of stellar classification (Corbally & Gray, 2018).

Classification of stellar spectra is fundamentally grounded in the field of spectroscopy with two very different approaches existing, spectrum analysis and spectral classification. As the name suggests, spectrum analysis involves complex analysis of a spectrum whereas spectral classification simply groups together different spectra based on common morphologies (Binney & Merrifield, 1998). Spectral morphologies are based on absorption lines that vary in strength, width and position (Gulati et al., 1994). Grouping spectral morphologies resulted in the establishment of different classification schemes, like the MK classification scheme that is in use today (Morgan et al., 1943). The most frequent features obtained from stellar spectra are chemical abundances, e.g. Magnesium to Iron ratio (Mg/Fe) or Silicon to Iron (Si/Fe) ratio, and physical properties such a surface gravity (log g), surface temperature (T_{eff}) and metallicity (Fe/H) (Garcia-Dias et al., 2018).

2.3.1 Harvard Classification Scheme

In 1918, the Henry Draper Catalogue (HD) was published (Cannon & Pickering, 1918). This catalogue formulated the Harvard Spectral Classification scheme, sometimes also referred to as the Draper system (Corbally & Gray, 2018), especially in text books from that time (Fath, 1934; MacPherson, 1926). This scheme was derived mainly through the work of US astronomer Annie Jump Cannon and it classifies stars based on surface temperature (T_{eff}). T_{eff} is derived through analysis of the strengths of spectral lines. Stars are assigned a main class and a sub-class. The main class is a single letter, and the sub-class is a number from 0 to 9. The sequence of the main class, in descending order of temperature, is O, B, A, F, G, K and M, class O represents the hottest stars and M represents the coolest. The 10 sub-classes indicate decreasing temperature i.e. 0 is hotter than 9. Confusingly, sometimes a decimal value can be used for the sub-class e.g. 2.5, and O class stars sub-class range only goes from *5* to *9.5*. This means that O*5* class stars are the hottest stars known to exist based on this classification technique (Karttunen et al., 2006).

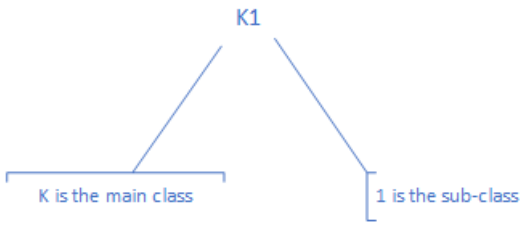

Figure 2.5: Harvard Classification for star HD249.

Spectral Class	**Max T_{eff}**	**Spectral Features**
O	35,000	Weak Hydrogen lines, Helium II visible, Ionized elements visible, Carbon III, Nitrogen III, Oxygen III, Silicon IV
B	15,000	Strong Helium I lines, Helium II lines absent, Strong Hydrogen lines, Carbon II, Oxygen II, Silicon III
A	9,000	Strong Hydrogen lines, Strong Magnesium II, Silicon II, Weak Calcium II
F	7,000	Weak Hydrogen lines, Strong Calcium II, 1st ionized state elements
G	5,500	String Calcium II lines, Hydrogen lines, Prominent neutral metals
K	4,000	Hydrogen lines, Neutral metallic lines
M	3,000	Strong Calcium I, Strong molecular bands

Table 2.1: The Harvard Spectral classification system, main class description overview.

The full classification for star HD249 using the Harvard Classification scheme is K1, as shown in Figure 2.5.

Table 2.1 provides an overview of the relationship of the main spectral class to temperature and the prominent spectral features common to that class (Binney & Merrifield, 1998).

2.3.2 MK Classification Scheme

In 1943 the Yerkes Spectral Classification system was introduced (Morgan et al., 1943) (MKK). This is an extension to the Harvard Classification scheme, and it includes the surface gravity (*log g*) as well as T_{eff} in the classification. In 1953 the Yerkes Spectral Classification system was modified and renamed the Morgan-Keenan (MK) system that is still in use today. The MK Classification is a description of how the spectrum of a star appears at medium resolution in the blue-green wavelength region of the visible spectrum (Corbally & Gray, 2018). The scheme retains the main and sub-classes defined by the Harvard Classification scheme and adds a luminosity class, to represent the intrinsic brightness of a star. The luminosity class is represented as Roman numerals from *I* (most luminous super-giants) to *VII* (white dwarfs). Table 2.2 gives more details of the luminosity classes. The MK Classification scheme has introduced new main classes for specific types of low temperature stars e.g. L, T, C, R and S, but these are considered sub-types of existing main class stars and represent less that 10% of known stars (Karttunen et al., 2006, p. 210). The full classification for star HD249 using the MK Classification scheme is K1*IV*, as shown in Figure 2.6. The relationship between the luminosity classes and the stellar atmospheric parameters is still an active area of research, Malkov et al. (2020) recently determined new statistical relations to estimate the atmospheric parameters (T_{eff}, log g) using the MK classification scheme.

Luminosity Class	**Description**
Ia$^+$	Hyper-giants or extremely luminous super-giants
Ia	Luminous super-giants
Iab	Luminous super-giants (intermediate)
Ib	Less luminous super-giants
II	Bright giants
III	Normal giants
IV	Sub-giants
V	Main sequence stars
VI	Sub-dwarfs
VII	White dwarfs

Table 2.2: The luminosity class introduced in the MK Classification scheme

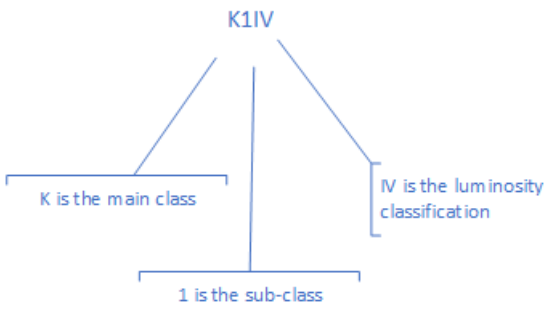

Figure 2.6: MK Classification for star HD249.

2.4 Convolutional Neural Networks

This section is intended to give an overview of the advances in convolutional neural networks as they have become state-of-the-art in the area of computer vision.

2.4.1 2D Convolutional Neural Networks

The year 1989 saw the introduction of what we today would recognise as a convolutional neural network (CNN) (LeCun et al., 1989). The system, known as *LeNet-5*, was a fully interconnected CNN architecture with back propagation and was deployed to automatically detect hand written zip codes on letters processed by the US Post Office and for identifying hand-written numbers on checks in banks (Aggarwal, 2018). The network architecture used in *LeNet-5* is an example of a supervised learning algorithm. Machine learning has two types of learning algorithms, supervised and unsupervised learning. Supervised learning algorithms learn from labelled data and these algorithms are used for regression and classification tasks. Unsupervised learning algorithms learn without labelled data and these algorithms are typically used for clustering (Aggarwal, 2018; Kelleher, 2019).

It wasn't until 2012 that CNNs came to prominence when a system known as *AlexNet* dominated the annual ImageNet competition, more formally known as the *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC). The purpose of ILSVRC is to evaluate algorithms for computer vision and it has been dominated by CNNs ever since 2012. *AlexNet* used the RELU activation function and had a total of eight layers

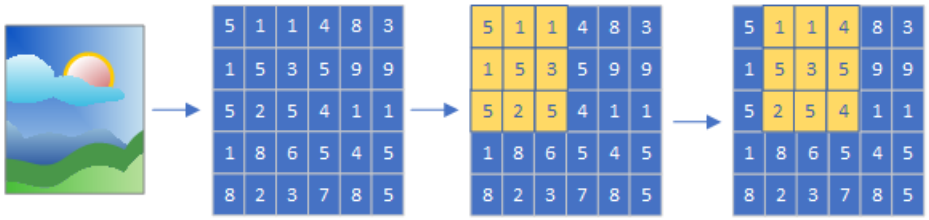

Figure 2.7: 2D CNN processing an image.

consisting of convolutional layers, dense layers and max-pooling layers (Krizhevsky et al., 2017). *AlexNet* achieved an error rate of 15.4%, 10% less then that achieved by the runner-up system (Aggarwal, 2018). In 2014 the Inception network from the *GoogLeNet* team won the ILSVRC. The Inception network contained 22 convolutional layers and reduced the ILSVRC error rate to 6.7% (Kiranyaz et al., 2021; Szegedy et al., 2015). In 2015, Microsoft developed *ResNet* that beat the benchmark set by the Inception network by utilising skip connections to allow even deeper learning (He et al., 2016a; Kelleher, 2019). Resnet reduced the ILSVRC error rate to 3.08% (Aggarwal, 2018). Ever since *AlexNet*, CNNs have dominated image classification tasks.

LeNet-5, *AlexNet* and the Inception network are all examples of 2D CNNs. 2D CNNs represent the input data in a matrix form and a kernel moves across the matrix in two dimensions, left to right and top to bottom. This is illustrated in Figure 2.7. As the kernel moves across the image, it is trying to identify spatial features that best represent the input data. Through the use of supervised learning, the network learns the kernel functions that best identify the spatial features of the input data that map to the provided class labels. The size of the kernel determines how many input values are used in the kernel function, and the depth describes the number of feature maps that store the spatial features. Other hyper-parameters influence the learning process, for example, stride determines the amount of overlap the kernel will have when moving across the input. If stride is set to a value of 1, then the kernel will move across 1 input value at a time. If the stride is less then the kernel size, then overlap will occur (Aggarwal, 2018).

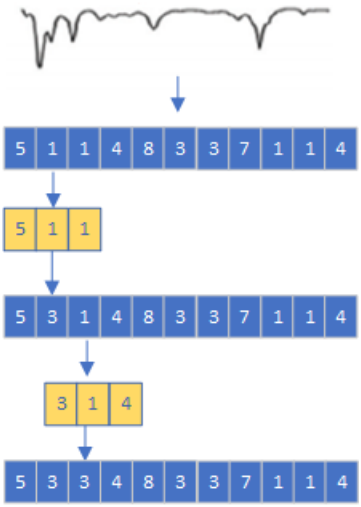

Figure 2.8: 1D CNN processing a signal.

2.4.2 1D Convolutional Neural Networks

Signal processing is a set of problems that require detection of features in a one dimensional domain e.g. an electrocardiogram (ECG), an ECG being an electrical signal that varies over time (Kiranyaz et al., 2016). Attempts to use 2D CNNs with ECG have drawbacks, including computational complexity. For example, attempts to use 2D CNNs for ECG classification requires a primary step to convert the signal into a log-spectrogram for training and classification (Ruiz et al., 2019). 1D CNNs are networks that are designed to process 1D signals. They are more efficient networks when compared to 2D CNNs because of the reduced complexity of the network. However, the learning process is the same as that used for 2D CNNs. Here the input is in the form of a vector and the kernel can only move across the input vector in one direction (left to right). This is illustrated in Figure 2.8. The size of the kernel represents the maximum number of input values used in the kernel. The same hyper-parameters, like stride, also apply to 1D CNNs.

1D deep networks still suffer from the same drawbacks as 2D deep networks, requiring the use of batch normalisation and dropout to overcome these limitations (Zhang et al., 2018). However, there are distinct advantages to 1D CNNs over 2D, they are less computationally complex and typically gain good results using shallow networks (Kiranyaz et al., 2021). In recent years, 1D networks have demonstrated superior

performance in areas such as early arrhythmia detection in electrocardiogram, fault detection (Abdeljaber et al., 2019; Kiranyaz et al., 2019) and stellar classification (Jiang et al., 2020; Sharma et al., 2019). It has also been shown that mixing 1D and 2D CNNs has helped increase efficiency and accuracy when processing hyperspectral images (Li et al., 2019).

2.5 Machine Learning in Astronomy

Ever since 1610 when Galileo first looked at Jupiter through a telescope and plotted the movement of the Jovian satellites, astronomers have been collecting data with modern astronomical surveys collecting terabytes or petabytes of data (Fluke & Jacobs, 2020). It is simply not possible to process all this data manually and computational processing is a vital part of astronomy, Data Mining in Astronomy is now established with several books being published in this area (Ivezić et al., 2020; Way, 2016).

Due to the immense distances involved in astronomical research, the data collected in astronomical surveys are either photographic images or spectra, with most surveys being ground based data collection projects e.g. the Sloan Digital Sky Survey (SDSS). Satellites have been used in recent years to collect data, especially in areas of the electromagnetic spectrum that are invisible to ground based detectors e.g. the Chandra X-ray Observatory. At this point in time, the use of machine learning in astronomy is considered an established field, specifically within the fields of solar astronomy, extra-solar planet detection, stellar classification, variable star classification, transient object detection and galaxy classification (Fluke & Jacobs, 2020).

2.5.1 Machine Learning in Stellar Classification

The application of machine learning in spectral classification is an established field and has evolved as new machine learning techniques are introduced. Gulati et al. (1994) pioneered an effort to apply the use of artificial neural networks to this domain. They used 3 different data sets, the Jacoby, Hunter, Christian Atlas (JHC) (Jacoby et al., 1984), Pickles Spectral Atlas (Pickles, 1985), and Silva & Cornell Atlas (SC) (Silva &

Cornell, 1992) spectral libraries for this purpose and reported an accuracy of 2 spectral sub-classes.

They also pioneered a method to harmonise spectral data from the 3 different libraries, a process that requires degrading the resolution of each library to that containing the lowest resolution, then converting the spectra to a higher resolution using interpolation. This is a process that is reused in more recent research (Sharma et al., 2019). The stellar classification is based on the MK Classification scheme, the MK classes being encoded in a novel method also pioneered by Gulati et al. (1994).

Weaver and Torres-Dodgen (1997) also used artificial neural networks to classify stellar spectra using the MK Classification scheme. The data used in this paper was captured by the authors using the 91cm MIRA telescope with a resolution of 15Å across a wavelength range from 5750 to 8950Å and consists of 817 unique spectra. This network used raw spectra as input. The result of their network shows an average error of 1.26 sub-classes across all stellar types. As part of the classification, they were attempting to classify spectral type simultaneously with the luminosity class and they speculate that this could be contributing towards the poor performance.

At this time, Principal Component Analysis (PCA) became a popular method to use as a data reduction technique when classifying spectra. Bailer-Jones et al. (1998) used an artificial neural network to classify stellar spectra using the MK Classification scheme. They used 5,000 spectra from the Michigan Spectral Survey and used PCA as the data reduction technique, stating that it compresses the spectra by a factor of over 30 while retaining 96% of the variance in the data. A total of 25 principal components are used in this experiment. The resultant classification error rate from their network was 0.82 sub-classes.

Singh et al. (1998) used an artificial neural network and PCA for stellar classification using the MK Classification scheme. They use the JHC data set exclusively for testing and the SC data set exclusively for training. PCA is used as the data reduction technique and a total of 20 principal components are used. They tested ten different ANN architectures containing between 1 and 2 hidden layers, and up to 256 nodes in the hidden layers. They also ran a test to compare the principal components

inputs against the raw spectral data as inputs. The architecture that provided the best results consists of 161 inputs, two hidden layers with 64 nodes each, and the output layer. Note that this architecture is based on the raw spectrum input, not the principal components. The best network has a correlation co-efficient r=99%.

Bazarghan (2008) developed a system using PCA and a Probabilistic Neural Network (PNN) for stellar classification of the JHC and ELODIE data sets. The best performance was achieved with only 26 principal components. Stellar class encodings used are those introduced by Gulati et al. (1994). They report a classification accuracy of 3.2 sub-classes.

As expert systems became popular, Manteiga et al. (2009) constructed a system known as STARMIND that used the MK Classification scheme to classify stellar spectra. This expert system evaluates stellar spectra based on spectral line ratios, line strengths and band fluxes utilising a simple IF-THEN rule structure. The first set of rules split the classification into two bins, the star is either binned as being class O, B, A or F, or binned as being class G, K or M. Further rules then refine the classification of the main class. The CFLIB data set is used to test STARMIND, which are spectra from the visible wavelength range (3465 to 9469Å). STARMIND achieved an accuracy of 79.5% when classifying to the sub-classes level, 92% within 1 sub-class and 98% within 5 sub-classes. They note that most of the errors in their system are what they describe as border problems, for example a spectrum is classified A0 but is labelled B9.

Also using the expert system approach is Gray and Corbally (2014), they built a system called MKCLASS. Data is fed through a series of modules in the inference engine that are designed to classify a specific stellar type based on the MK Classification scheme. This expert system implementation is similar to that used in Manteiga et al. (2009) as it uses direct comparison of a sample with MK standard spectra, considering the morphologies of existing classifications. The system processes spectra within the visible wavelength range (3800–5600Å). The first step is to identify the main stellar class by comparing the sample with the standard spectra of each stellar type. The initial classifications are based on following distinctions O, B, A, F-G, K-M and Non-

MK objects. Once the main class is identified, the sub-class is classified, resulting in the complete stellar class for the object. The system was tested using 960 labelled spectra from the NStars Project (Grey et al., 2003) and 280 published by Grey et al. (2001). These are manually labelled spectra. For testing purposes, O class stars are omitted due to the low number of samples available, but the system is designed to classify them. This system is reported to have a precision of 0.6 sub-classes, no accuracy figures are reported so it is difficult to compare the performance of MKCLASS to modern machine learning methods. The MKCLASS system was used to classify new spectra (Gray et al., 2015).

Kheirdastan and Bazarghan (2016) wrote a system for bulk spectral classification. Data from the SDSS catalogue (SEGUE-1 and SEGUE-2 releases, which consists of 400,000 spectra) is used to train and classify spectra, using Probabilistic Neural Network (PNN) (Specht, 1990), k-means and SVM. The training data consists of 100,000 labelled samples. The remaining 300,000 are used as the test data set. The SDSS release covers a wavelength range from 3850 to 8900Å and PCA is used as the data reduction technique. They use 280, 400 and 700 principal components to see what provides the best results. The use of PCA as a data reduction technique was shown to work very well in previous studies (Bailer-Jones et al., 1998). The study is limited to just 38 sub-classes of the MK system, rather than the full 70 sub-classes. Best classification accuracy was achieved using 700 principal components in all models (k-means, PNN and SVM), achieving an accuracy of 80%.

Liu et al. (2015) employed a support vector machine (SVM) for the purposes of MK classification using spectra from the LAMOST data set. They used a feature extraction process to extract line indices from the spectra. These line indices represent the prominent spectral features for each stellar class. They train the SVM to classify six main classes from the MK Classification scheme. In this case, classes O and B are merged into a single OB class as there are very few O class samples. They select 3,134 spectra and cross reference the objects IDs with the SIMBAD database for the appropriate MK classification, this is the data set for training and testing their model. Approximately 1,500 spectra are reserved for testing. They report different levels of

classification accuracy depending on the stellar class. For instance, classes A and G had an accuracy rate of over 90% whereas the O and B classes only achieved an accuracy of 44%. They cross reference their line indices with the same objects in the MILES data set and find that the indices from one survey are not completely compatible with another survey.

Wang et al. (2017) built a locally connected deep neural network for stellar classification and spectra recovery. They used 50,000 randomly selected spectra from the Large Sky Area Multi-Object Fibre Spectroscopic (LAMOST) telescope, also known as the Guo Shou Jing telescope, based in Xinglong Station, Hebei Province, China. The 50,000 spectra are labelled and only comprise of F, G and K class stars. It makes sense to choose these stellar classes as these are the most abundant stars, but it also makes it more difficult to compare the results presented in this paper to those existing in the literature. Wang et al. (2017) perform a data reduction technique by creating what they describe as synthesized pixels by computing the average every five pixels. In this way, they reduce the number of inputs to their network from 3601 to 721. They use the Pseudo-inverse learning (PIL) algorithm, originally proposed by Guo and Lyu (2004), stating that this is a more efficient learning process than the traditional back propagation algorithm. The use of the locally connected DNN leads to a significant decrease in the amount of network parameters to be trained but this is at the expense of missing connections in the data that are not local. The results reported are favourable, their network accuracy being better than other traditional methods like PCA, but they don't compare their results to existing DNN results in the literature.

Sharma et al. (2019) is heavily influenced by the approach pioneered by Gulati et al. (1994) but apply a 1D convolutional neural network and Random Forest to the domain of spectral classification. They use 4 data sets, harmonising these data sets as defined in (Gulati et al., 1994). The 4 data sets are the JHC, CFLIB, ELODIE and MILES data sets, the CFLIB data set retained exclusively for testing their trained classification model, an approach seen previously in Manteiga et al. (2009). Sharma et al. (2019) pre-process the data sets to harmonise the spectral data. As a result of this processing, they remove data from the data set. The criterion for removing

data includes low signal-to-noise ratios, and gaps in spectral coverage. Additional effort is made to retain data when an object classification is in conflict across the different data sets. To resolve this, they use the SIMBAD online spectral database as a source of truth and manage to retain more data in doing so. An auto-encoder is also used on SDSS data, approximately 60,000 labelled spectra are downloaded for this purpose. After training the auto-encoder, the trained weights are transferred to the classifier before training with the already prepared data set. It is notable that the data processing applied to the other data sets used in this paper are not applied to the SDSS data set. Also, there are currently 16 different data releases from SDSS, it is not stated what data release is used, meaning this step is impossible to recreate. Sharma et al. (2019) create separate classifiers for the stellar temperature class (main and sub-classes) and the luminosity class. They report a classification accuracy of 89% for the main spectral class and an error rate of 1.23 sub-classes.

Jiang et al. (2020) use both 1D and 2D CNNs for spectral classification. The use case is focused on a search for what they term unusual celestial bodies such as cataclysmic variables (compact binaries containing a white dwarf extracting atmospheric material from a companion star, usually a main sequence K or M class star (Gänsicke et al., 2009)). The use case in this paper is highly domain specific, but it boils down to classification of M class stars to the sub-class range 0 to 4. They use labelled data from the SDSS public Data Release 16 for training and testing. A total of 46,180 M class spectra are selected from the SDSS data set. The classifier is then applied to 1,234,445 spectra from the LAMOST Data Release 7. Data is harmonised across the two data sets, normalisation and resampling is applied, similar to that performed by Sharma et al. (2019). The result is normalised raw spectra in vector format covering a wavelength range of 5000Å. To format the data for the 2D CNN, the spectra is folded into a matrix of order 50x100. Their system is called multi-scale coded convolutional neural network (EMCCNN) and it uses an unusual architecture. It consists of two parallel CNNs to help extract features and reduce noise. The output of the two CNNs is merged and fed into a dense network for classification. For the 1D CNN architecture, one CNN uses a kernel of size 3 and is two layers deep, using the RELU activation

function between convolutional layers. The other CNN has a kernel of size 5 and is also two layers deep, again using the RELU activation function between convolutional layers. The 2D CNN architecture is the same as that used for the 1D architecture, but the kernels are 3x3 and 5x5 respectively. They compare the 1D and 2D EMCCNN architecture to ResNet and VGG16 architectures, also in 1D and 2D CNN flavours, using data sets with different SNR ranges. They find that the 2D CNN architectures outperform the 1D architectures in nearly all cases, the exception being that their 1D EMCCNN architecture out performs their 2D EMCCNN architecture. The 1D ResNet architecture outperformed the 2D equivalent when there was a high SNR, suggesting that it was over fitting the training data. They report that the use of folded spectra and 2D CNNs can achieve good classification results and are less likely to over-fit the training data.

Brice and Andonie (2019) developed a system to classify spectra using the MK classification schema with a single classifier. This approach computes a scaled spectrum for each object and then uses the k nearest neighbour (KNN) algorithm to classify objects. A total of 168,982 spectra are available from the SDSS public Data Release 14 and this covers 46 of the 420 possible MK classes. This data set is imbalanced, so to address this, different models are computed based on under-sampling, over-sampling and a hybrid approach of under-sampling and over-sampling. They achieve a precision score of 80% using just K=3 for the KNN model trained using over-sampling. Selecting a higher valuing of K does not increase the precision score. This is a fast algorithm that takes only 90 seconds to perform feature selection, 7 seconds to train and 3 seconds to test for K=3.

Dafonte et al. (2020) use the MK classification scheme and an ANN for the purpose of stellar classification. The approach taken in this paper is to split the MK classification scheme into its sub-components i.e., the main stellar class and the luminosity class are classified using two separate classifiers, the same approach taken by Sharma et al. (2019). Dafonte et al. (2020) use the same data sets used in Gulati et al. (1994) i.e., JHC, Pickles and SC atlases, but they bolster the number of available spectra by incorporating the 908 spectra from the ELODIE data set. Although data sets are

reused across these papers, these data sets are not without their deficiencies, for example, the same object can be assigned a different class in each atlas. The approach taken by Dafonte et al. (2020) is to simply remove these spectra from the data set which results in smaller data sets, For example, of 908 spectra incorporated from the ELODIE data set, only 500 are selected for use. They also completely disregard all O class stars in the atlases as they considered the small numbers present in the library as a reason not to include them in their data set. The maximum number of training instances for any of the remaining stellar classes is only 25, the minimum being 20, meaning this experiment is using only limited amounts of data for all stellar classes. Dafonte et al. (2020) do not use raw spectral data as input to their CNN, instead they apply a feature extraction process to extract spectral indices. Up to 25 spectral indices are used as inputs to the network (the actual number could be as low as 16, depending on the wavelength range coverage in the spectrum). They report an average accuracy rate of 75%, their best model obtaining an accuracy of 95%.

Sharma et al. (2020) turn their 1D CNN auto-encoder to outlier detection in spectral catalogues. This network is trained to recognise spectral classes O, B, A, F, G, K and M. The purposes of this system is to learn to reconstruct spectra, and assign an expected spectral class, thereby detecting mis-labelled data. The system is trained on 53,956 spectra and tested with 6,627 spectra. The accuracy is measured by computing the mean square error (MSE) and root mean square error (RMSE). They obtain a mean MSE of 0.0006 and a mean RMSE of 0.0222, indicating that their network is very capable of reconstructing spectra very accurately. This is a very welcome research as it can lead to more accurate catalogues and better classifiers that are trained on this data.

2.6 Summary

This section provides an overview of the literature review and presents the gaps identified in the literature.

2.6.1 Overview

Stellar classification was formalised in 1918 with the creation of the Harvard Classification Scheme, sometimes referred to as the Draper Classification scheme. The Harvard Classification Scheme divided stars into different classes based on surface temperature. These classes consist of a main class and a sub-class. In 1943, the MK Classification Scheme was created, it being an extension to the Harvard Classification Scheme to include a luminosity class alongside the existing main and sub-classes. Traditionally, stars are classified using the MK Classification Scheme manually through visual inspection of stellar spectra, a system ripe for automation.

The first application of neural networks to stellar classification was in 1994 (Gulati et al., 1994). Since then, stellar classification is now considered an established application of machine learning within astronomy (Fluke & Jacobs, 2020). Artificial neural networks, principal component analysis and expert system technologies have been used in this domain. There are two distinct approaches in the literature; one is to automate the estimation of stellar atmospheric parameters (T_{eff}, log g and Fe/H) which are then used in the MK classification process. The second approach, which was the focus of this chapter, is to automate the stellar classification directly. The advantage of the first approach (estimation of stellar atmospheric parameters) is that it allows the creation of catalogues of stellar parameters, which have further use outside of this domain, but the classification process is incomplete. The advantage of the second is that the stellar classification is estimated directly. This is really important given the vast amounts of spectral data generated by modern astronomical surveys. For example, the LAMOST telescope can capture up to 4000 spectra in a single exposure (Wang et al., 2017). CNNs are now being used in this domain. Most applications of CNNs in this domain concentrate on 1D CNNs, spectral data is always treated as a sequence of data. However, recent research has shown that a spectrum can be folded into a two dimensional representation and used to train 2D CNN classification models (Jiang et al., 2020).

2.6.2 Gaps in the literature

Recurrent Neural Networks (RNNs) are specific neural networks that have been shown to work with sequential and time series data. In astronomy, RNNs have been applied to the domain of variable star classification (Becker et al., 2020; Mackenzie et al., 2016; Naul et al., 2018). Traditionally in the domain of stellar classification, spectra are processed as sequential data. RNN architectures like Long Short Term Memory (LSTM) and Gated Recurrent Units (GRU) have the capacity to learn long-term dependencies, which exist in stellar spectra. A gap exists to use a RNN for stellar classification as the ability to learn long-term dependencies can help learning in a context where features are not local.

Hidden Markov models (HMM) is an algorithm that is also good for classification of sequential data. HMM have a proven track record in speech recognition and they could be applied to the domain of stellar classification (Bishop, 2006).

Sharma et al. (2019) construct a 1D CNN for the purposes of stellar classification. Their network architecture doesn't follow the heuristic of increasing the number of feature maps in deeper convolution layers. Scope exists here to address this concern and test their network architecture that better follows the CNN architectural best practices (Aggarwal, 2018).

2D CNNs have achieved state of the art results in image classification (Szegedy et al., 2015). 2D CNNs have been applied in other fields of astronomy, such as variable star classification (Mahabal et al., 2017). Jiang et al. (2020) applied 2D CNNs to a stellar classification task, but this was not the complete MK Classification scheme. They state that 2D CNNs show that they are a viable network for stellar classification. Therefore, a gap exists to use 2D CNNs with folded spectra (from 1D sequence into matrix format) for the purposes of stellar classification using the MK Classification Scheme.

Reinforcement learning is a different learning algorithm that could be used for stellar classification. While supervised learning uses examples of the target representation in order to learn, reinforcement learning takes the approach of attempting to learn the problem to be solved (Sutton & Barto, 1998). Prior to automated classification tech-

niques, stellar classification was a manual task based on identifying spectral indices. This problem seems highly appropriate for a solution based on reinforcement learning.

Optimization algorithms used in the papers presented in the previous section are used with their default settings. Recent research suggests that the default values for most optimizers do not produce the best results (Schmidt et al., 2020). Scope exists to refine the use of optimization algorithm through fine tuning of their parameters.

None of the reviewed papers have tried using semi-supervised learning techniques, a form of unsupervised learning that has achieved impressive performance learning image representations (He et al., 2020). He et al. (2020) have shown impressive use of Contrastive Self-Supervised Learning for unsupervised visual representation learning. This is a technique that could be applied to the domain of stellar classification through using folded spectra.

2.6.3 Research Question

The literature review and gap analysis resulted in the following research question:
To what extent can the accuracy of stellar classification using the MK Classification Scheme be improved by using 2D CNNs with folded spectra?

Chapter 3

Experiment design and methodology

This chapter introduces the research methodology of this research, a quantitative empirical study into the use of convolutional neural networks in the domain of stellar classification. This chapter states the null and alternative hypothesis, data preparation methods and the experiment design used to test the hypothesis.

3.1 Hypothesis

H_1: If a convolutional neural network with two-dimensional kernel is trained with folded stellar spectra, the stellar classification performance will result in an average accuracy score greater than 89% based on the main stellar class from the MK Classification scheme and this increased performance will be statistically significant at a 95% confidence level.

H_0: A convolutional neural network with one-dimensional kernel, trained with stellar spectra data for the task of stellar classification, provides the best possible average accuracy score of 89% based on the main class of the MK classification scheme.

3.2 Data Preparation

This section details the data sets and what data cleaning was performed on the different data sets. The term data set is used here to describe spectral data and the corresponding catalogue containing meta-data associated with each spectra. Spectral data is provided in the Flexible Image Transport System (FITS) file format.

3.2.1 Flexible Image Transport System

Spectra are supplied in a file format using the Flexible Image Transport System standard [1]. This file format is endorsed by the International Astronomical Union (IAU) for the interchange of astronomical data and defines a binary encoding for spectral and image data. All the data sources used in this research project supply data in this format. The Python library *astropy.io*[2] is used in this research to process FITS files.

3.2.2 Data Sets Overview

Four different data sets are used in this research. Each data set consists of stellar spectra in the FITS file format, and a catalogue that contains meta-data of each star, such as the spectral type. These data sets are:

1. The Indo-U.S. Library of Coudé Feed Stellar Spectra (CFLIB) (Valdes et al., 2004) [3].
2. MILES Spectral Library (Sanchez-Blazquez et al., 2006) [4].
3. ELODIE 3.1 Spectral Library (Prugniel et al., 2007) [5].
4. Jacoby-Hunter-Christian Atlas (JHC) (Jacoby et al., 1984) [6].

[1] https://fits.gsfc.nasa.gov/fits$_d$*ocumentation.html*
[2] https://www.astropy.org
[3] https://www.noao.edu/cflib/
[4] http://miles.iac.es
[5] http://www.obs-hp.fr/www/guide/elodie/elodie-eng.html
[6] https://www.stsci.edu/hst/instrumentation/reference-data-for-calibration-and-tools/astronomical-catalogs/jacoby-hunter-christian-atlas.html

Jacoby-Hunter-Christian Atlas (JHC)

The Jacoby-Hunter-Christian Atlas is the oldest data set used in this research. The spectra was captured in December 1980 over 26 nights from the No.1 90cm telescope at the Kitt Peak Observatory in Arizona, USA. Kitt Peak Observatory is in operation since 1958 and is located at an altitude of 2096 m, coordinates 31°5730N 111°3548W. The atlas was introduced in paper Jacoby et al. (1984) that describes a total of 161 spectra for spectral classes O-M. The spectra cover a wavelength range from 3510Å to 7427Å and was captured at a resolution of 4.5Å.

MILES Spectral Library

The MILES data set was captured in between 2000 and 2001 over a total of 25 nights from the 2.5 m Isaac Newton Telescope (INT) at the Roque de los Muchachos Observatory, La Palma, Spain. Roque de los Muchachos observatory is in operation since 1985 and is located at an altitude of 2396 m, coordinates 28°4549N 17°5341W. Sanchez-Blazquez et al. (2006) describe a total of 985 spectra for spectral classes O to M. The spectra cover wavelength range from 3525Å to 7500Å at a resolution of 2.3Å.

ELODIE Spectral Library

The ELODIE spectra are based on observations made on the 193cm telescope at the Haute-Provence Observatory, France. The Haute-Provence Observatory was established in 1958 and is located at an altitude of 650m, coordinates 43°5551N 5°4248E. The original ELODIE catalogue consists of 908 spectra corresponding to 709 different stars but no time frame is provided for their capture (Prugniel & Soubiran, 2001). An updated version of this catalogue was released in 2007 (Prugniel et al., 2007), called the ELODIE 3.1 release and is used in the research of Sharma et al. (2019). This new catalogue contained 1962 labelled spectra of 1388 objects.

The ELODIE 3.1 data set is not publicly available. The catalogue is cited in a paper that was never officially published, the paper only exists on the pre-print servers

[7] and the catalogue is not archived. This proved to be a major issue as this data set contains the majority of the data used in the training phase of Sharma et al. (2019). Requests to obtain the ELODIE 3.1 catalogue was issued to the lead authors of both relevant papers, to no avail.

Even though the ELODIE 3.1 catalogue is not publicly available, all data from the ELODIE survey is available for download from its website [8]. This spectra spans a wavelength range of 4000Å-6800Å which differs slightly to that reported in Sharma et al. (2019), 3900Å-6800Å. In order to proceed, a data set was constructed from the available ELODIE data, however, differences may exist between this data set and data used in Sharma et al. (2019). Section 3.2.4 explores this constructed data set in more detail and attempts to cross reference attributes of this data set against information published in Sharma et al. (2019).

The Indo-U.S. Library of Coudé Feed Stellar Spectra

The Indo-U.S. Library of Coudé Feed Stellar Spectra (CFLIB) spectra are based on observations made using the 0.9m Coudé Feed Telescope at Kitt Peak National Observatory, Arizona, U.S.A. This is the second catalogue that was created using instrumentation provided by the Kitt Peak National Observatory. The CFLIB provides spectra for 1273 stars captured between 1995 and 2003 at a resolution of 1Å.

3.2.3 Data Cleaning

This section outlines what cleaning was applied to the catalogues. Cleaning specifically applies to catalogues containing spectra that are not classified using the main class from the MK Classification scheme. The approach is to attempt to classify all objects, thus maximising the amount of data available. Objects that have no official classification are removed.

[7] http://arxiv.org/abs/astro-ph/0703658
[8] http://www.obs-hp.fr/www/guide/elodie/elodie-eng.html

Jacoby-Hunter-Christian Atlas (JHC)

All objects in the JHC data set have assigned stellar classes [9] so no cleaning is necessary. The distribution of the main spectral class within the JHC data set is shown in Figure 3.1.

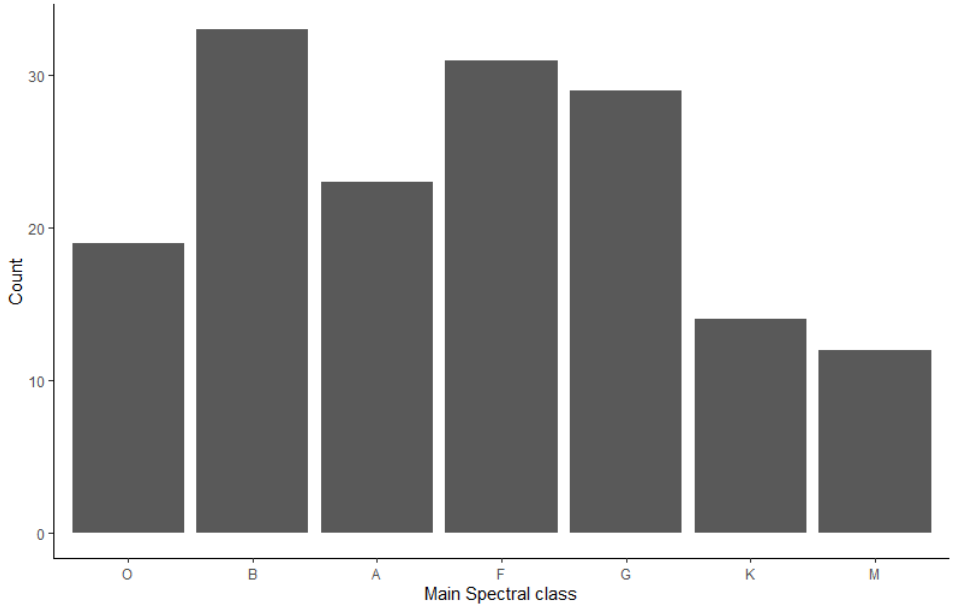

Figure 3.1: Main class distribution of the Jacoby-Hunter-Christian data set.

MILES

The stellar classification of the MILES catalogue is incomplete, as can be seen in Figure 3.2. There are 76 entries with no stellar class, 7 entries that are classified as S, which is a sub-type of M. R class stars are a subclass of K class stars. Entries of class I is not a valid main class, and may indicate a labelling error.

[9] https://www.stsci.edu/hst/instrumentation/reference-data-for-calibration-and-tools/astronomical-catalogs/jacoby-hunter-christian-atlas.html

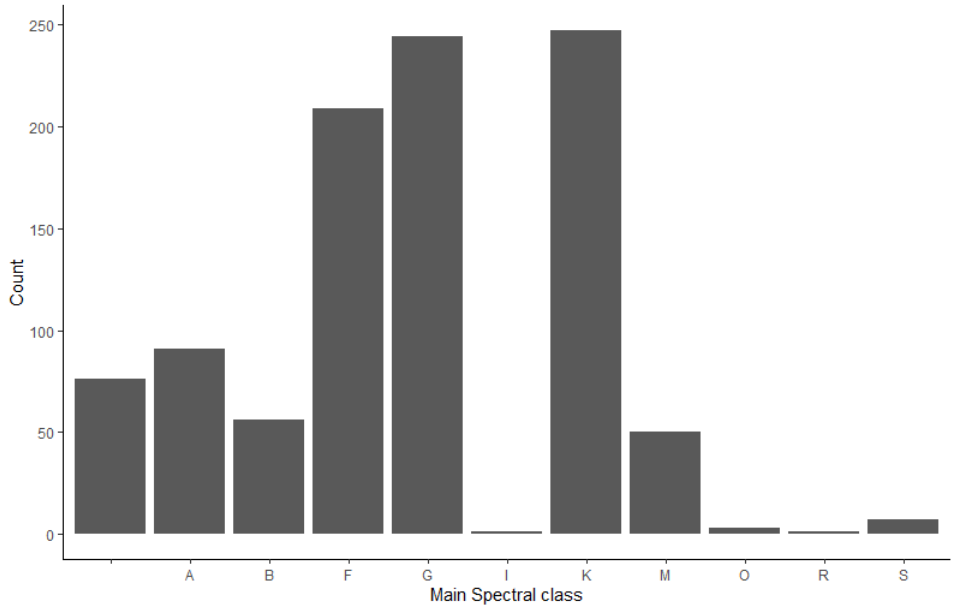

Figure 3.2: Main class distribution in the MILES data set.

SIMBAD [10] is an online spectral database maintained by the Centre de Données Astronomiques de Strasbourg (CDS). This database contains information on approximately 5.5 million stars, including the stellar class where available. The SIMBAD database is queried using simple HTTP requests to retrieve data. For the purposes of this research, the SIMBAD classification is given higher priority to that assigned in the MILES catalogue. The methodology to assign classifications is as follows:

1. Query the SIMBAD database based on the object name in the MILES catalogue.
2. If MILES and SIMBAD classifications are the same, there is high degree of confidence in the MILES classification and it is retained.
3. If no classification exists in MILES and SIMBAD, remove spectra from the catalogue.
4. If MILES and SIMBAD classifications both exists but are different, the SIMBAD classification is preferred over the MILES classification.

[10] http://simbad.u-strasbg.fr/simbad/

5. If no SIMBAD classification exists but the object is classified in the MILES catalogue, the MILES classification is retained.

The results of this processing is that:

1. 75 objects are not classified in MILES and SIMBAD, these are removed.
2. 737 object classifications are the same in MILES and SIMBAD.
3. 94 SIMBAD classifications are preferred over the MILES classification.
4. 77 object have no SIMBAD classification, therefore the MILES classification is retained.

What remains are 908 classified objects that are used in this research. The new main class distribution is shown in Figure 3.3.

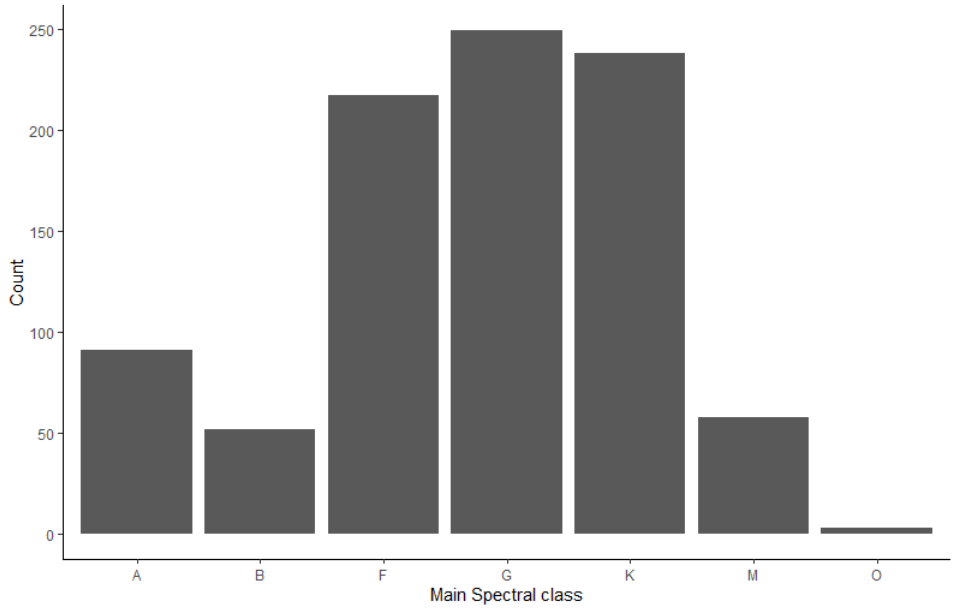

Figure 3.3: Main class distribution in the MILES data set after cleaning.

CFLIB

There are nine objects in the CFLIB catalogue that are not assigned a stellar class. These objects do not exist in the JHC or MILES catalogues. Cross referencing the object names with the SIMBAD database shows that three of these stars are of class

S and six are of class C, both of these classes are subclass of the M class, therefore the classification assigned to these objects is M.

Further filtering is applied to the data set for spectra with the following issues

1. The wavelength coverage does not include the full range from 4000Å to 6800Å.
2. Spectra containing gaps greater than 50Å.

A total of 110 spectra meet this criteria and are removed, resulting in a catalogue containing 1160 objects. The impact of this step is much smaller than that reported in Sharma et al. (2019), this is addressed in Section 3.2.4. The resulting distribution is shown in Figure 3.4.

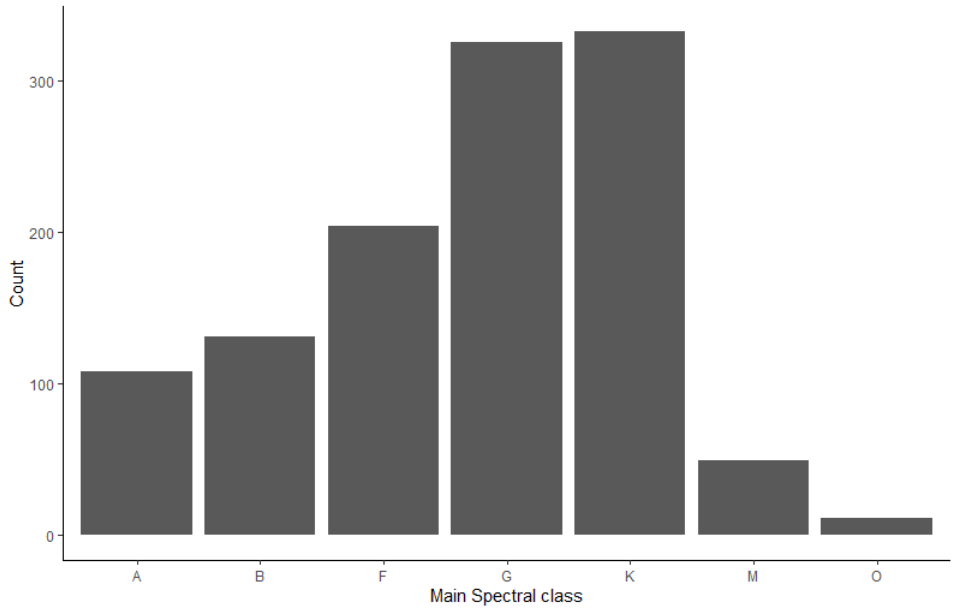

Figure 3.4: Main class distribution in the CFLIB data set after cleaning.

ELODIE

No cleaning is necessary for the ELODIE data set as only labelled spectra are included in the data set. The distribution of the main class is shown in Figure 3.5.

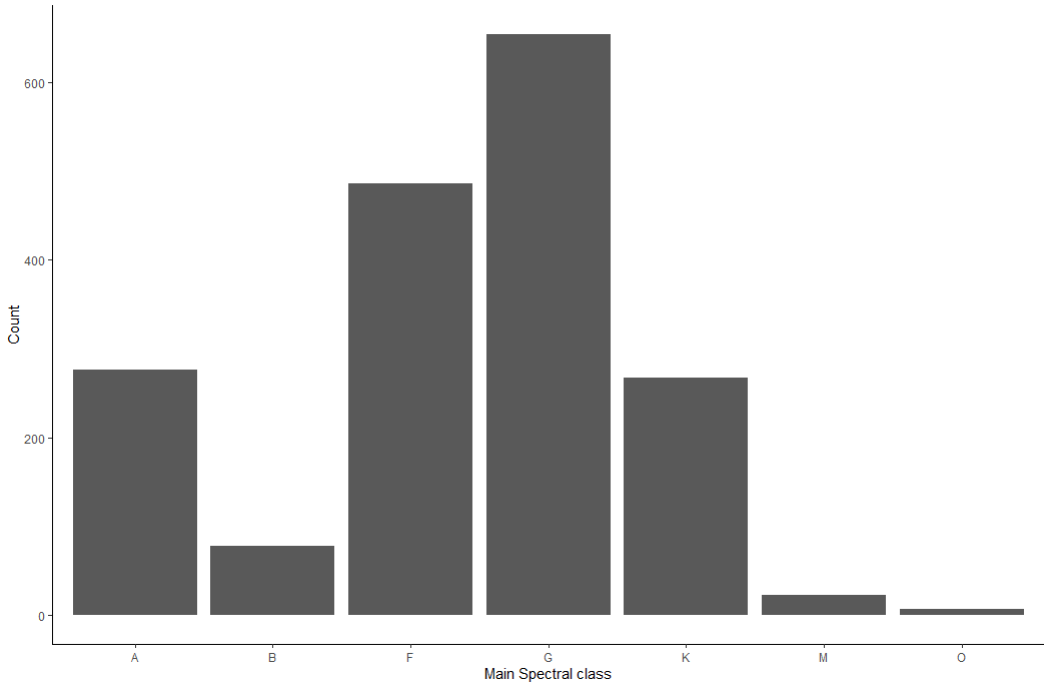

Figure 3.5: ELODIE main class distribution.

A Note on Class Imbalance

The cleaning process above will have highlighted the fact that each data set is imbalanced i.e. equal representation for each class does not exist. A summary of the class distributions across the data sets is presented in Table 3.1. These distributions are not the result of poor or biased surveys, but represent the relative abundances of different stellar classes in the Milky Way i.e. there are less super-massive O class stars than main sequence G class stars (Karttunen et al., 2006). Over-sampling is a potential technique that would be useful in this situation (Kelleher et al., 2015). However, Sharma et al. (2019), like other papers reviewed in Chapter 2, did not try to balance the number of stars in each class. As a result, this research will use the classes are they are in order to best follow the experimental procedure as outlined by Sharma et al. (2019).

Main Class	JHC	MILES	ELODIE	CFLIB
O	19	3	7	11
B	33	52	82	131
A	23	91	316	107
F	31	217	626	201
G	29	249	526	313
K	14	238	369	330
M	12	58	36	42

Table 3.1: The main class distribution across the data sets

3.2.4 Data Consistency

Signal To Noise Ratio Comparison

As stated previously, the ELODIE data set used in this research differs from that used in Sharma et al. (2019). To try and gain insight into the difference between the data sets, the signal-to-noise ratio (SNR) distribution of the ELODIE data used in this research is compared to Sharma et al. (2019). This is not an exact comparison, for reasons that will be outlined below, but it is a useful guide. The differences are reported in Table 3.2 and it shows that 90% of the the data used by Sharma et al. (2019) and 91% of the data used in this research are within the same SNR region (SNR >15 and SNR <200). Sharma et al. (2019) makes no reference to the distribution of the SNR values in the remaining data whereas the remaining 9% of data used in this research has SNR <15.

SNR Range	Sharma et al. (2019)	This Research
<15	Not reported	9%
15-100	70%	67%
100-200	20%	24%

Table 3.2: SNR comparison of the ELODIE data used in this research and that used in Sharma et al. (2019).

Reasons why the SNR values cannot be considered an exact comparison include:

1. Due to the unavailability of the ELODIE 3.1 catalogue, this comparison can never be exact.

2. In Sharma et al. (2019), data is removed due to missing classifications, resulting in a smaller data set that could skew the results.

3. The percentages quoted in Sharma et al. (2019) are for the entire training data set (ELODIE, MILES JHC). However, there are no published SNR values for data sets other than ELODIE, so it is unclear how these figures were obtained.

4. Sharma et al. (2019) makes claims about the SNR distribution of the CFLIB data set. This data set is generated by merging captures across different gratings before normalising. In this context, it doesn't make any sense to discuss SNR, for instance, what capture does it refer too? Unless they reconstructed the CFLIB data set from the original captures, measuring the SNR values for this data set is not actually possible. This claim needs to be treated with caution.

It is encouraging that the data used in this research is similar to that reported in Sharma et al. (2019) based on the SNR distribution, but this should be treated with caution as it isn't entirely clear what the figures reported by Sharma et al. (2019) actually mean or how they were obtained.

CFLIB Data Set Comparison

As stated in Section 3.2.3, 110 spectra are removed from the CFLIB data set, compared to 423 by Sharma et al. (2019). This is not a trivial difference, and, in fact, is critical as this impacts directly on how the metrics are generated in this research (classification accuracy). It is unclear why an additional 313 spectra are removed from the CFLIB data set in Sharma et al. (2019) as no justification can be found to do so and this will have a direct effect on the classification accuracy measurements. The difference in the resulting distributions of the main classes is shown in Table 3.3.

Main Class	Sharma et al. (2019)	This research
A	90	107
B	98	131
F	155	201
G	251	313
K	221	330
M	28	42
O	7	11
Total	850	1135

Table 3.3: Shows the difference in the main class distribution between the CFLIB data set reported in Sharma et al (2019) and that used in this research.

3.2.5 Data Processing

Each spectra is processed according to the process as outlined in Sharma et al. (2019) and this is based on an existing approach in the literature (Gulati et al., 1994). The steps applied are:

1. Degrade all the spectra in the ELODIE, MILES and CFLIB data sets to the same resolution as JHC by convolving with a Gaussian kernel.

2. Normalize flux values in all spectra to unity at 5550Å.

3. Use a cubic spline to interpolate available flux values over a wavelength range from 4000Å to 6800Å at a resolution of 1Å.

4. Wavelength information is discarded and the data is saved as a vector of order 1x2800.

The Gaussian kernel above is computed based on the resolution of the individual data set. The Full Width Half Maximum value (FWHM) value is computed based on the Formula 3.1, where R is the native resolution of the data set.

$$FWHM = \sqrt{4.5^2 - R^2} \tag{3.1}$$

The interpolated spectrum is saved to file to be used in the training and testing phases.

3.3 Neural Network Architecture

This section outlines the neural network architectures used in this research. These architectures are trained to produce classification models. An overview of the hyper-parameters that are configurable in a convolutional neural network architecture are shown in Table 3.4. More details on the description of each layer is available on the Keras API documentation website [11], the list in Table 3.4 is only the subset relevant to this research.

Layer	Name	Description
Convolution	Activation function	The activation function
	Filters	The dimensionality of the output space
	Kernel	The dimensionality of the kernel
	Padding	The padding to use
	Strides	The length of the stride
Dense	Activation function	The activation function
	Units	Dimensionality of the output space
Max Pooling	Pool size	Size of the pooling window
	Padding	The padding to use
Dropout	Rate	Fraction of the input units to drop

Table 3.4: Available hyper-parameters in each layer in a CNN.

3.3.1 Baseline Network

The baseline network architecture is shown as Part A in Figure 3.6. While this research attempts to reproduce this as accurately as possible, a design choice is required at this point, specifically, the number of classes the model will predict. The reason this choice is required is because Sharma et al. (2019) only report results (classification accuracy, precision etc) for the main class (7 classes) but they train their network based on the sub-class level (70 different classes). The design choice for this research is to classify based on the main class only (7 classes) to best compare results with those presented in Sharma et al. (2019).

This network architecture consists of a 1 dimensional kernel of width 4, this kernel width is constant across all CNN layers. There are four convolutional layers that

[11] https://keras.io/api/layers/

are interleaved with pooling layers. The output of the convolutional layers are fed into a flattening layer before being passed into a two layer dense network. The first dense layer has 64 nodes, the second has 32 nodes. The output of the dense layers is the output layer that uses the *softmax* activation function to determine the predicted main class based on the MK Classification scheme. The loss function is the mean squared logarithmic loss. The optimizer is Adam, this being a stochastic gradient descent method that is based on adaptive estimation (Aggarwal, 2018). Adam is a popular optimizer but some have expressed concern that models created using Adam generalise worse than regular stochastic gradient descent (Kingma & Ba, 2015; Wilson et al., 2017). Adam is used with default configuration, as shown in Table 3.5.

Parameter	Value
Learning Rate	0.001
beta_1	0.9
beta_2	0.999
epsilon	1 e^{-7}

Table 3.5: Adam optimisation configuration parameters and default values.

All parameters in the baseline network are fixed as per the specification in Sharma et al. (2019). All hyper-parameters are documented in Table A.5. Activation function RELU is shorthand for the Rectified Linear Unit activation function.

3.3.2 Proposed Network

The network architecture proposed in this research is shown as part B in Figure 3.6. This architecture is closely related to that in Sharma et al. (2019) but differs in some significant ways:

1. A two dimensional convolutional kernel is used instead of a one dimensional kernel. Different kernel sizes will be used during the hyper-parameter tuning stage.

2. The number of convolutional and max pooling layers is reduced. This is necessary as the dimensions of the input data to the proposed network architecture are not

large enough to undergo four max pooling layers. To apply four such layers would reduce the input shape to 1x1 before input data reached the dense layer. The design decision is to only use half of the input layers of the baseline network architecture in the proposed architecture.

3. Batch normalisation layers are included between convolutional layers. These are absent in the baseline architecture.

4. Dropout layers are included between dense layers. These are absent in the baseline architecture.

Batch Normalisation is a technique that prevents vanishing and exploding gradients while Dropout is a technique that randomly drops connections in a dense network (Aggarwal, 2018). There is discussion about whether Batch Normalisation and Dropout layers actually work together constructively when applied to image recognition (He et al., 2016a; Ioffe & Szegedy, 2015). As the premise of this research is that a CNN will perform better if the spectra is treated like an image compared to a one-dimensional vector, this is something that will be evaluated when fine tuning the hyper-parameters of this network. Hyper-parameters of the network will be tuned as part of the training process, see Section 3.4.4 for more details.

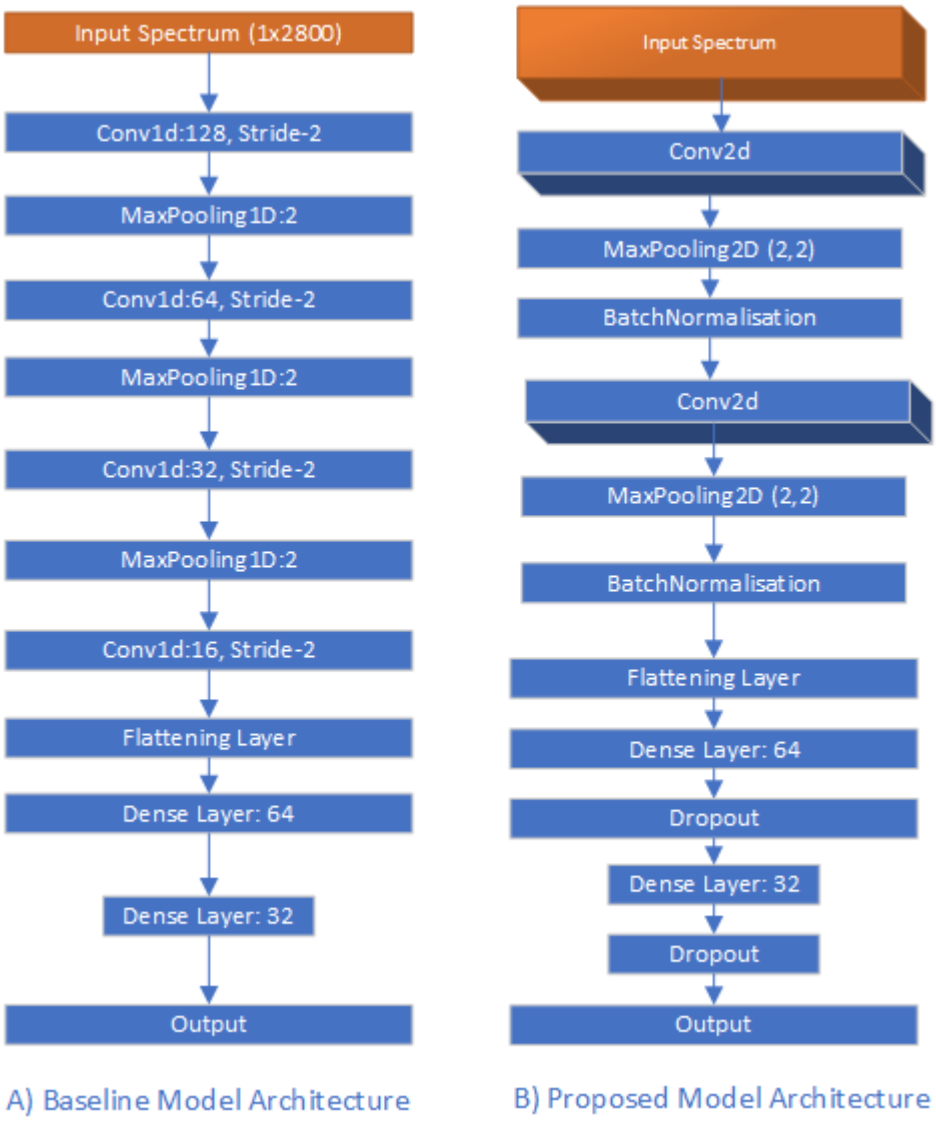

Figure 3.6: CNN architectures used in this research. 'A' represents the baseline architecture defined in Sharma et al. (2019) and 'B' represents the architecture proposed by this research.

3.3.3 ResNet

The ResNet architecture was developed by Microsoft and it won the ILSVRC in 2015 (Aggarwal, 2018; He et al., 2016a). The insight provided by He et al. (2016a) was that deep networks were hard to train as the optimisation process becomes too difficult. To address this, they introduced the concepts of residual units and skip connections (Géron, 2017). A residual unit is a self-contained unit of two convolutional layers

and the RELU activation function. The skip connection allows the input data from a residual unit to be merged with the output of the residual unit, thus ensuring the residual units only needs to learn small changes to the input. By chaining together these residual units, they were able to build very deep network architectures that were also extremely efficient in terms of computational complexity. The original ResNet architecture contained 34 layers. A ResNet 50 architecture was defined, also by He et al. (2016a), using blocks that contain three convolutional layers.

As part of this research, a ResNet50 model will be trained and compared to the proposed model. A ResNet50 V2 model is trained using the input shape that shows the best accuracy performance. ResNet V2 was introduced in 2016 and includes changes to reduce the complexity of training and improve generalisation (He et al., 2016b). The ResNet V2 50 implementation is provided by the Keras.io framework[12]. Due to the depth of the ResNet architecture, the time to train it is much higher than the baseline or proposed network architectures. As a result, ResNet classification models will be trained for 200 epochs rather than 1000.

3.4 Experimental Procedure

This section outlines the procedure used to build and evaluate the models.

3.4.1 Data Sampling

A mixture of hold-out sampling and K-fold cross validation is required to build and test the classification models, and the data sets are setup accordingly. Hold-out sampling, as shown in Figure 3.7, is a process to separate the test data set from all other data used in the training process. This process ensures the test data is not available to the modeller, resulting in a test process that provides an honest evaluation of the model performance (Kelleher et al., 2015; Schorfheide & Wolpin, 2012). K-fold Cross Validation is a process where the training and validation data sets are merged and segregated into folds, that are used iteratively to build multiple models where each

[12]https://keras.io/api/applications/resnet/resnet50-function

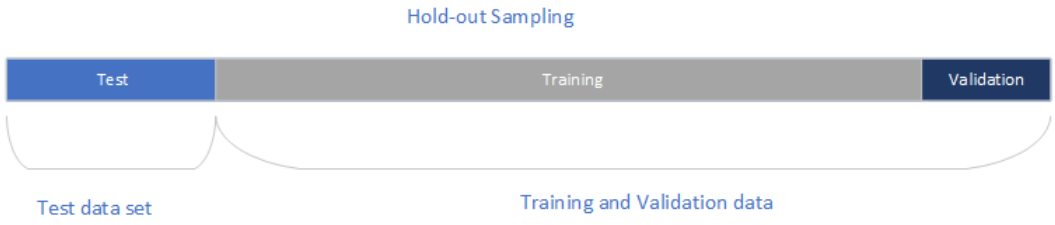

Figure 3.7: Hold-out Sampling.

model uses a different fold as the validation data set. All remaining folds make up the training data set. Figure 3.8 shows how each successive fold is used as the validation data set for different classification models.

3.4.2 Data Setup

The CFLIB data set is retained exclusively for experimenting on the trained classification models while JHC, MILES and ELODIE data sets are used exclusively in the training phase. For the purposes of hyper-parameter tuning, 15% of training data is used as the validation data set. During the training phase, when K-Fold Cross Validation is used, ten folds are used (K=10), this being the best practice (Kelleher et al., 2015).

A custom implementation of the Keras Dataset Preprocessing class[13] is used to access the data sets. Keras provides out of the box implementations for image, text and time-series data sets, none of which can be used directly on the data sets used in this research. This custom implementation is responsible for loading and reshaping the data sets and it can randomly split data into training and validation data sets, the random controller being seeded, meaning the random selection is repeatable as long as the same seed is used. The seeding ability means this preprocessing class is used to generate repeatable data sets for experimentation.

To use this custom implementation, spectra for different classes are arranged in a directory structure, one directory per stellar class e.g. all A class spectra are added into the A directory, etc. The custom dataset preprocessing class automatically loads and labels data based on their folder name. The training data sets of JHC, MILES

[13]https://keras.io/api/preprocessing/

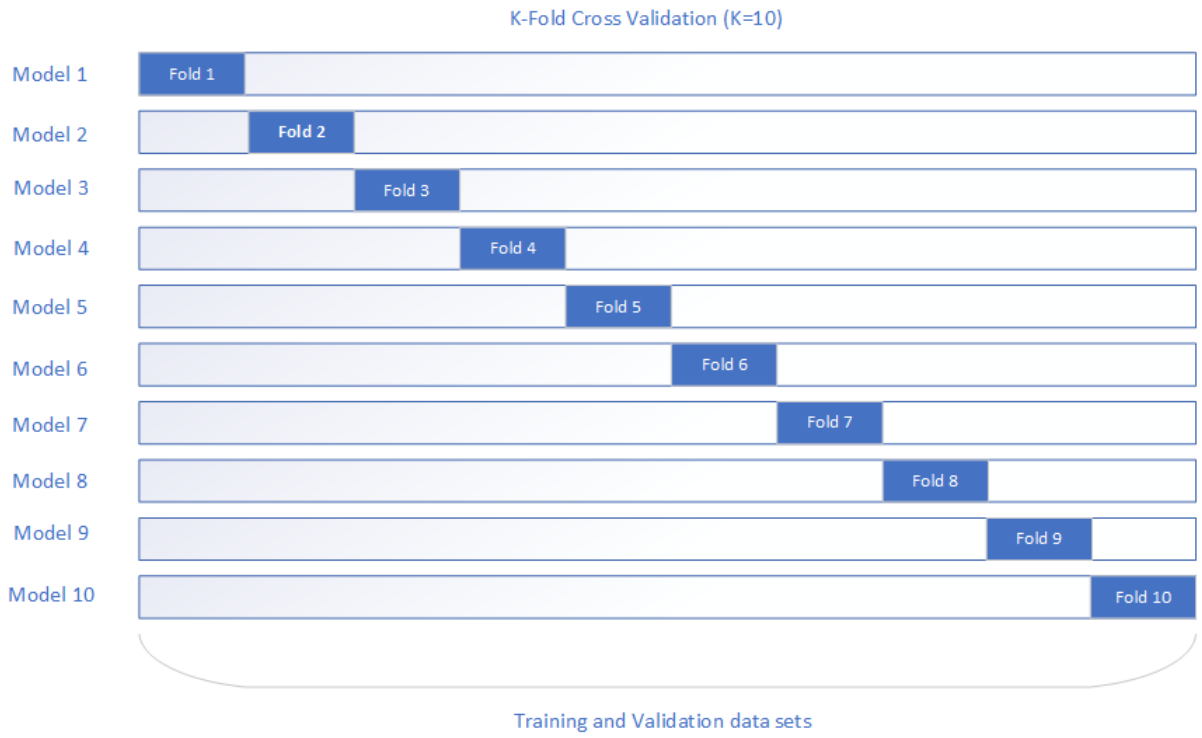

Figure 3.8: K-Fold cross validation.

and ELODIE are merged into this directory structure and CFLIB is copied into a separate, but identical, directory structure.

3.4.3 Spectra Input Dimensionality

Different input dimensions are used in this research. Figure 3.9 shows how the spectra for K class star HD38 is represented using the different dimensions proposed by this research.

1. Vector of order 1x2800. This is the shape of the data used by Sharma et al. (2019).

2. Matrix of order 50x56. This is a two dimensional representation of 2800.

3. Matrix of order 40x70. This is also a two dimensional representation of 2800.

In order to test the effects of the chosen input dimensions on the accuracy of the classification models, models with the opposite dimensions are also trained and tested.

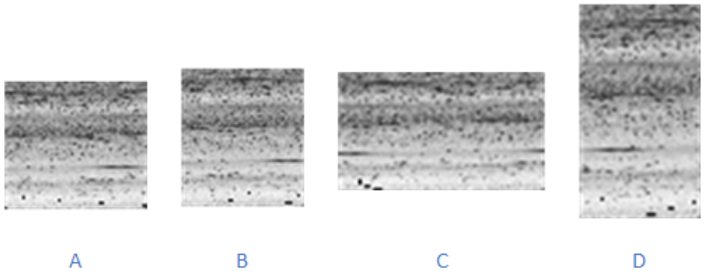

Figure 3.9: Different representations of the spectrum of star HD 38. A is a 50x56 matrix, B is 56x50, C is 40x70 and D is 70x40.

1. Matrix of order 56x50.

2. Matrix of order 70x40.

3.4.4 Hyper-parameter Tuning

The proposed CNN architecture is defined in Section 3.3.2. This network has several hyper-parameters in the Dense and CNN layers that require tuning, as previously described in Table 3.4. In the Dense layer, the dropout frequency value is configurable while in the convolutional layer, filter size and kernel size are tuned. The activation function used is RELU for all layers except for the output layer, where *softmax* is used. The padding used is *same*. The specific hyper-parameters that will be tuned as part of this research are shown in Table 3.6.

Layer	Hyper-parameter
Convolution layer 1	kernel size filter size stride
Max Pooling layer 1	pool size
Convolution layer 2	kernel size filter size stride
Max Pooling layer 2	pool size
Dropout layer 1	frequency
Dropout layer 2	frequency

Table 3.6: Proposed model hyper-parameters tuned as part of this research.

In total, 10 hyper-parameters are tuned as part of this research. The hyper-parameters that provide the best training and validation accuracy are incorporated into the architecture specification and used to training the classification models.

3.4.5 Training Classification Models

All classification models are trained using Python 3.8, Keras 2.4.3 with Tensorflow 2.3.0 backend and the Nvidia CUDA Deep Neural Learning library. The Scikit Learn KFold [14] module is used when KFold Cross Validation is used. A DELL XPS 9500 with Nvidia GeForce GTX 1650 Ti GPU that has 4GB dedicated RAM and 16GB of shared RAM is used to train the classification models.

The loss function is an important part of the learning configuration, the mean squared logarithmic loss function is used in this research alongside the Adaptive Momentum (Adam) optimiser. Adam is used with its default configuration settings as shown in Table A.5. Batch size is 256 and the number of epochs is 1000. Early stopping is not used as this was not used in Sharma et al. (2019). The following Keras callbacks are used when training the classification models:

1. ModelCheckPoint. Only models with a better validation accuracy are saved. This is important to ensure the best models are retained, there is no guarantee that the final model after 1000 epochs will be the best model.

2. Tensor Board Callback to store Tensor Board logs.

K-fold cross validation is used in the training phase. Cross validation is used to produce 10 classification models (K=10) based on different selections of data within the training data set. The ten best performing classification models (highest training and validation accuracy) are used in the evaluation phase.

3.4.6 Evaluating Models

The input to the evaluation stage are the ten classification models produced by the K-Fold Cross Validation process. Each of these classification models are tested using

[14] https://scikit-learn.org/stable/modules/generated/sklearn.model$_s election.KFold.html$

the same data sets. The average classification accuracy of each model is then used to determine the best performing model and input shape.

It is important that the data sets used for testing are consistent to ensure the classification accuracy results are comparable regardless of the fold or input dimension used. To achieve this, 100 random seeds are generated using the Python random library. These seeds are stored in a file and used to control the preprocessing class described in Section 3.4.2. This preprocessing class uses the supplied seed to select data consistently, resulting in 100 experimental data sets, each data set contain 8% of the spectra in the CFLIB data set. Classification accuracy is a measure of how good a model is at correctly classifying input, and is calculated based on Equation 3.2 (Kelleher et al., 2015) where TP is the number of True Positive classifications, FP is the number of False Positive classifications, TN is the number of True Negative classifications and FN is the number of False Negative classifications.

$$ClassificationAccuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{3.2}$$

The Scikit Learn [15] function *accuracy score* is used to automatically compute the classification accuracy for each experiment. The classification accuracy is stored to file for statistical evaluation. Other metrics reported for each architecture are Precision, F_1 Score and Recall.

Precision is a measure of how often a positive classification is correct compared to the total number of positive classifications. It is the ratio of true positive classifications to the total number of true positive (correctly classified) and false positive (incorrectly classified as positive) classifications for a target class, as shown in Equation 3.3 (Kelleher et al., 2015). A high precision reflects a models ability to make correct predictions.

$$Precision = \frac{TP}{(TP + FP)} \tag{3.3}$$

Recall is a measure of the rate at which the model correctly classifies a target

[15] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html

class. It is the ratio of true positive classifications to the total number of true positive (correctly classified) and false negative (incorrectly classified as a different class) classifications for a target class, as shown in Equation 3.4 (Kelleher et al., 2015). A high recall reflects a models ability to distinguish between different classes.

$$Recall = \frac{TP}{(TP + FN)} \tag{3.4}$$

F_1 Score is a performance metric that combines precision and recall, as shown in Equation 3.5 (Kelleher et al., 2015). It is also referred to as the harmonic mean and is less sensitive to large outliers. Higher F_1 Score values indicate better classification performance.

$$F_1 Score = 2 * \frac{(precision * recall)}{(precision + recall)} \tag{3.5}$$

3.4.7 Statistical Evaluation

An α value of 0.05 is defined for the significance test. It is assumed that the classification scores generated follow a normal distribution as this is required to use parametric methods. It is necessary to verify this assumption of normality using the appropriate statistical methods such as checking the skew and kurtosis of the distribution and performing Levene's test for homogeneity of variance. Assuming the classification accuracy scores are normally distributed, a independent T-test will compare the classification accuracy ranges from the baseline model and proposed models to compute the t-statistic with sample size N=91, degrees of freedom=90. The t-statistic has an associated p-value which is compared to the pre-defined α value for statistical significance. The zeta squared value is used as the size effect that will ultimately accept or reject the null hypothesis. If any of the assumptions required for the parametric independent t-test are not met, a Mann-Whitney U Test for non-parametric distributions is used. The size effect is estimated by converting the z statistic into Pearson's r (Field et al., 2012).

3.5 Summary

Presented in this chapter are the neural network architectures used to test the hypothesis that 2D CNNs can provide higher accuracy performance than that of a 1D CNN in the domain of stellar classification using the main class of the MK Classification scheme. The baseline model is that defined in Sharma et al. (2019). The proposed model is a 2D CNN version of that in Sharma et al. (2019), but with only two convolutional layers. Four different input shapes are defined, resulting in four slightly different flavours of the proposed model. The input shapes are 40x70, 70x40, 50x56 and 56x50. Each of these shapes are whole number representations of a sequence of length 2800 folded into a matrix format. The ResNet50 architecture is also used to test this hypothesis. The ResNet V2 50 architecture is only applied to the input shape that shows best classification accuracy.

Four data sets are used in this research. These are The Indo-U.S. Library of Coudé Feed Stellar Spectra, the MILES Spectral Library, ELODIE spetral data and the Jacoby-Hunter-Christian Atlas. These spectral libraries represent four different spectral surveys across three different locations over a duration of almost 20 years. Each library has a different resolution and cover different wavelength ranges. As a result, the individual spectra are pre-processed and normalised.

The experiment design ensures robust evaluation of the classification models. 10-fold cross validation is used to produce ten classification models for each input shape. The classification model with the best validation accuracy score is retained for testing. The testing procedure uses a reproducible method to select data from the CFLIB data set. Each selection represents a test set and is approximately 8% of the total data set. There are one hundred test data sets and these data sets follow a distribution of main stellar class that is very close to the CFLIB data set.

A robust statistical procedure is defined to test the generated accuracy distributions from the models. Independent T-test (parametric) and the Mann-Whitney U Test (non-parametric) are used depending on the distributions of the accuracy scores. Levene's test for homogeneity of variance is applied to check for homogeneity of vari-

ance, a necessary step to understand if parametric tests are suitable.

3.5.1 Strengths

The strengths of this research are:

- **Availability of labelled data**: There are numerous data sets available in astronomy. The data sets used in this research are well known and in the public domain. they are used extensively in the literature.
- **Independent source of truth**: The SIMBAD database ensures that a trusted source of truth is available if the MK Classification of any object is in question.
- **Documented methodology**: The process of merging different data sets that span different wavelength ranges and resolutions exists in the literature since 1994.
- **Availability of CNN framework**: Using the Keras.io framework ensures the coding aspect of this research is focused on the production of models using a tried and tested code base. The likelihood of bugs being introduced into the code is reduced and the amount of code needed to produce and test the classification models is modest. The use of an Nvidia CUDA library increases the computational efficiency and thus reduces the overall time required to train and test the classification models. Other frameworks such as astropy.io and scikit-learn are also used in this research.
- **Domain knowledge**: Deep domain knowledge of spectroscopy is not required for this research. The spectra used are already classified by professional astronomers.
- **Hyper-parameter tuning**: In the course of this research, the hyper-parameters of the neural network are tuned to find the values that work best with this data.
- **Robust Statistical Evaluation**: This research will use robust statistical methods to compare the accuracy distributions. The independent T-test and the

Mann-Whitney U Test are robust statistical procedures used in this research to determine the difference of the means across two different distributions.

3.5.2 Limitations

The limitations of this research are:

- **Catalogue unavailability**: The ELODIE catalogue used by Sharma et al. (2019) is no longer available. This means the spectra used in this research will differ slightly to that used by Sharma et al. (2019), possibly impacting on accuracy of the classification models.

- **Limited comparison** Sharma et al. (2019) report an average accuracy score of 89% for main MK class classification. However, there is no distribution of accuracy scores provided. For example, the minimum or maximum accuracy scores are not documented, nor are the number of accuracy scores in their distribution.

- **No optimiser tuning**: This research will tune the hyper-parameters for the neural network, but the optimiser is using default settings. This is following the procedure as outlined in Sharma et al. (2019) but default optimiser settings are not always the best settings to use.

Chapter 4

Results, evaluation and discussion

As the amount of astronomical data being captured year after year increases, the requirement for an automated classification system for stars is necessary. This research is focused on improving existing classification techniques specifically through the application of convolutional neural networks on stellar spectra.

This chapter presents the results of the experiments and compares the performance of the baseline classification model to the one proposed in this research. The difference in the results is evaluated and the limitations of this study and findings are presented.

4.1 Experiment Data Distribution

The results presented below are produced after generating 100 experimental data sets from the CFLIB data set, each data set containing 91 randomly selected instances (8% of the total data set). As stated previously, the data sets are not balanced as they do not contain the same number of instances for each class. Therefore randomly selecting from this data set could result in an even more imbalanced distribution of stellar classes over the course of 100 experiments. A comparison of the distributions is shown in Table 4.1.

Main Class	CFLIB		Test	
	Count	%	Count	%
A	107	9.4	852	9.4
B	131	11.5	1034	11.4
F	201	17.7	1597	17.5
G	313	28.0	2559	28.1
K	330	29.0	2644	29.0
M	42	3.7	327	3.6
O	11	0.9	87	0.9
Total	1135		9100	

Table 4.1: This table shows the distribution of the stellar classes in the CFLIB data and in the 100 experimental data sets randomly selected from CFLIB.

Stellar classes G and K have the highest representation in the CFLIB data set and this is also true in the experimental data sets. Stellar class O has the lowest representation. The number of stellar classes, as percentages of the total, is very similar in CFLIB and the experimental data set, so it is clear that the generated data sets used in the experiments are a very good representation of the CFLIB data set.

4.2 Experimental Results

A high level overview of the experiment results are presented in Table 4.2. This table includes the minimum, maximum and average accuracy measurements of each classification model. A detailed review of the results for each classification model is presented in the following sections. A boxplot chart of these distributions is shown in Figure 4.1.

Model	Input Size	Min	1st Qrt	Mean	3rd Qrt	Max
Baseline Model	1x2800	0.58	0.73	0.77	0.80	0.91
Proposed Model	50x56	0.65	0.77	0.80	0.84	0.92
	70x40	0.66	0.79	0.82	0.85	0.93
	40x70	0.58	0.75	0.78	0.82	0.92
	56x50	0.62	0.75	0.79	0.84	0.93
ResNet V2 50	70x40	0.52	0.66	0.71	0.75	0.87

Table 4.2: Overview of the experimental accuracy scores for each model.

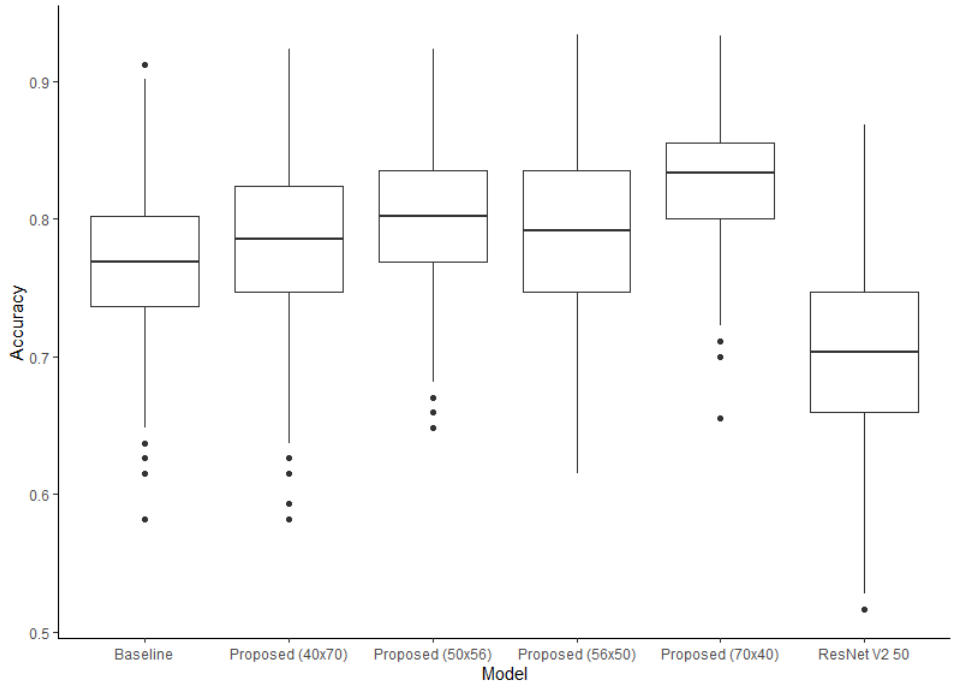

Figure 4.1: Boxplot of the accuracy scores obtained from all models created in this research.

4.2.1 Baseline Model Results

The experiments on the baseline model generated 1000 accuracy scores (0.76, sd=0.05) across the 10 models generated using the K-Fold Cross Validation process. The minimum accuracy score achieved was 0.58, and the maximum was 0.91. A histogram showing the distribution of the accuracy scores from the baseline models is shown in Figure 4.2. The accuracy distribution has a skew value of -0.12 and a kurtosis value of -0.15. The standardised skew is -1.61 and the standardised kurtosis is -1.00. As both the standardised skew and kurtosis are within a range of ± 2, this distribution can be considered to follow the normal distribution and can be analysed using parametric statistical methods. The breakdown of the distribution for each model is shown using boxplots in Figure 4.3.

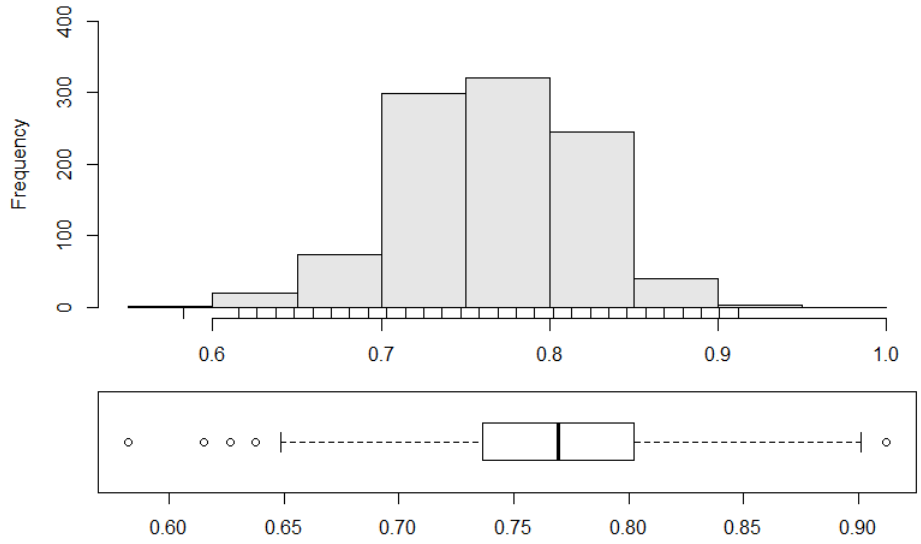

Figure 4.2: Histogram and boxplot of the classification accuracy scores obtained by testing the baseline models.

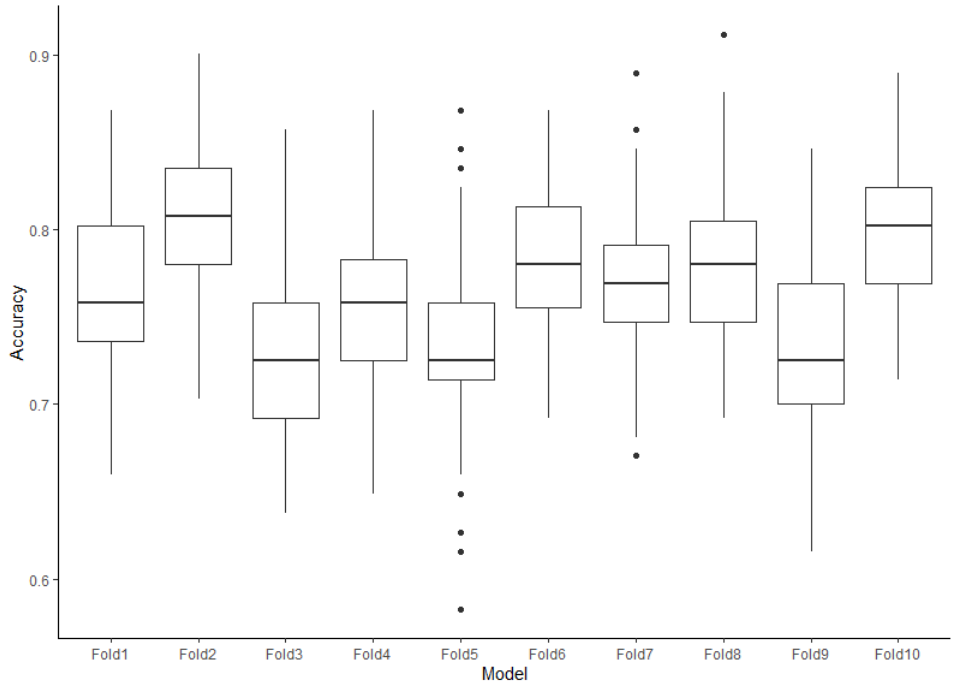

Figure 4.3: Classification accuracy results for the 10 fold baseline models.

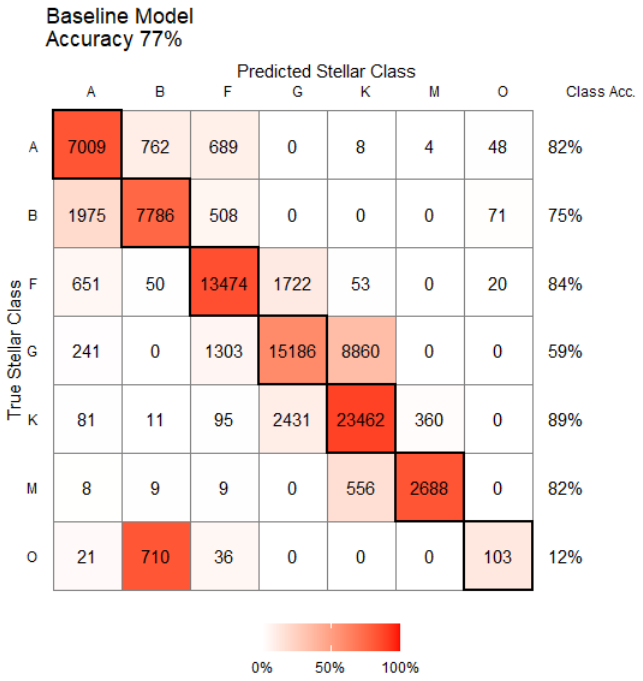

Figure 4.4: Confusion matrix showing the test results for the baseline model.

Main Class	Precision	Recall	F1 Score
A	0.71	0.82	0.76
B	0.84	0.75	0.79
F	0.84	0.84	0.84
G	0.80	0.59	0.67
K	0.72	0.89	0.79
M	0.88	0.82	0.84
O	0.24	0.12	0.15

Table 4.3: Evaluation scores for the baseline model.

Table 4.3 shows the evaluation statistics of the baseline model, precision, recall and F_1 score, broken down by stellar class. Figure 4.4 shows the confusion matrix [1] for the baseline model and the accuracy per class.

[1]Confusion matrix derived from code on stackoverflow.com https://stackoverflow.com/questions/23891140/r-how-to-visualize-confusion-matrix-using-the-caret-package

4.2.2 Proposed Model

The first step for the proposed models was to determine the hyper-parameters that produced the best validation results. To obtain these, the training data was separated into training and validation data sets, with 15% of training data being retained exclusively for validation. The proposed models are generated using different input shapes, matrix of order 50x56, matrix of order 56x50, matrix of order 40x70 and matrix of order 70x40. The hyper-parameters are tuned separately for each input shape. The experiments on the proposed models followed the same process as that of the baseline model, generating 1000 accuracy scores across the 10 classification models using the K-Fold Cross Validation process.

Model with input shape 50x56

The hyper-parameters that provided the best validation accuracy are shown in Table A.1. The best validation accuracy was obtained by gradually increasing the filter size in each convolutional layer, an opposite approach to the baseline model. This also dramatically reduced the time required to train the model. Figure 4.5 shows the histogram of the 1000 accuracy scores generated from the experiments on the proposed model using input size 50x56 (0.80, sd=0.05). The minimum accuracy was 0.65 and the maximum was 0.92. The accuracy distribution has a skew of -0.06 and a kurtosis of -0.34, standardised skew of -0.81 and standardised kurtosis of -2.19. As the standardised kurtosis is not within a range of ± 2, more investigation is necessary to ascertain if the data distribution can be considered to follow the normal distribution. The next step is to see if 95% of the standardised accuracy scores falls within a standard score. As there are more than 80 observations, the score of $\pm$ 3.29 is used. There are no standardised accuracy scores that fall outside a score of $\pm$ 3.29, therefore this data can be considered to follow the normal distribution and can be analysed using parametric statistical methods. The breakdown of the distribution for each model is shown using boxplots in Figure 4.6.

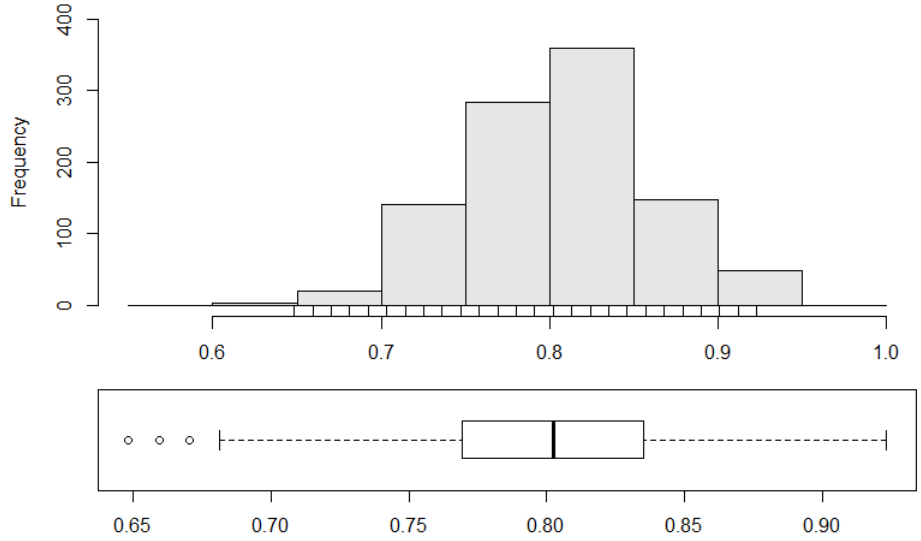

Figure 4.5: Histogram and boxplot of the classification accuracy scores obtained by testing the proposed classification models with input shape 50x56.

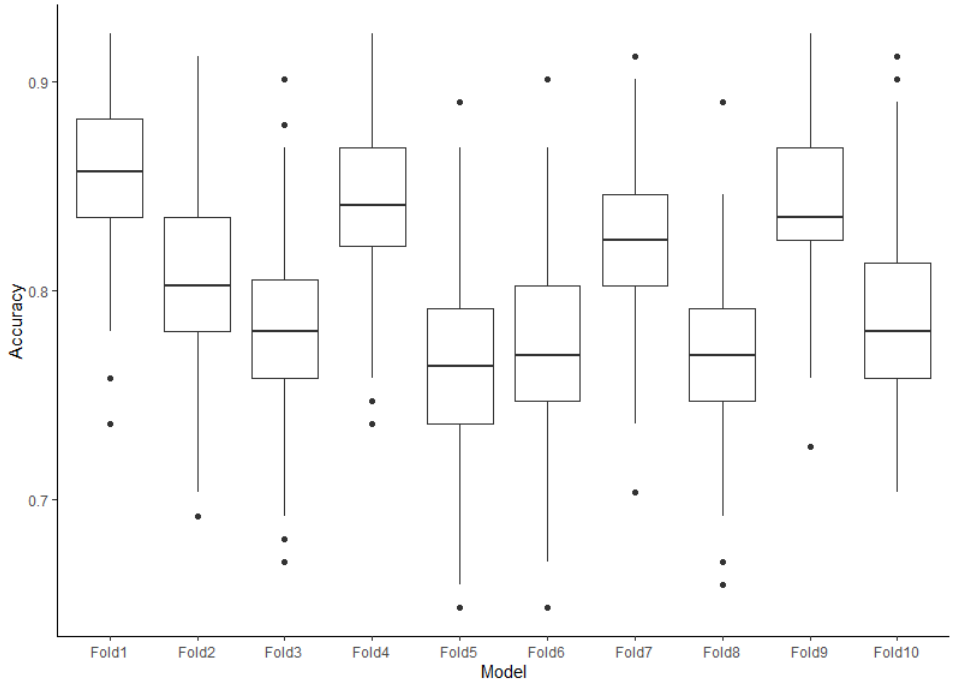

Figure 4.6: Classification accuracy results for the 10 fold proposed models using input shape 50x56.

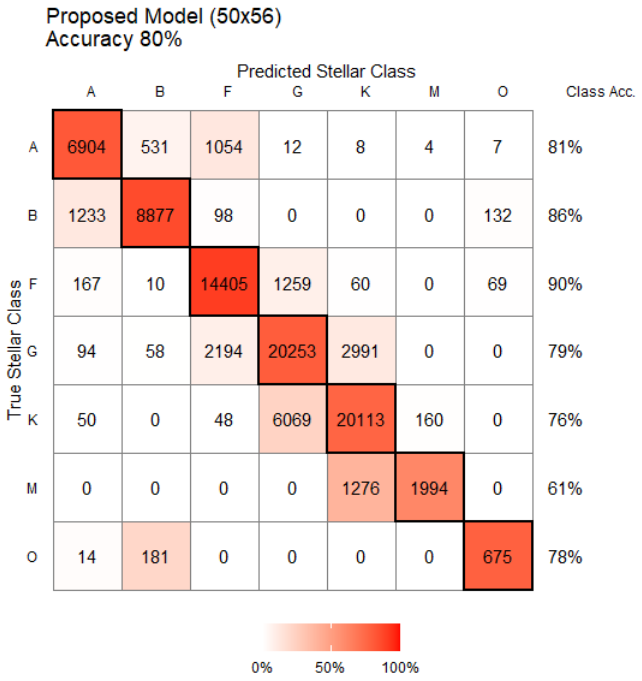

Figure 4.7: Confusion matrix showing the test results for the proposed model for input shape 50x56.

Main Class	Precision	Recall	F1 Score
A	0.82	0.81	0.81
B	0.92	0.86	0.89
F	0.81	0.90	0.85
G	0.75	0.79	0.76
K	0.83	0.76	0.78
M	0.92	0.61	0.71
O	0.76	0.78	0.76

Table 4.4: Model evaluation scores for proposed model for input shape 50x56.

Table 4.4 shows the evaluation statistics of the model broken down by stellar class. Figure 4.7 shows the confusion matrix for the model and the accuracy per class.

Model with input shape 70x40

The hyper-parameters that provided the best validation accuracy are shown in Table A.2. This model also obtained better validation accuracy when using increased number of filters in each subsequent convolution layer. Figure 4.8 shows the histogram of the 1000 accuracy scores generated from the experiments on the proposed model using

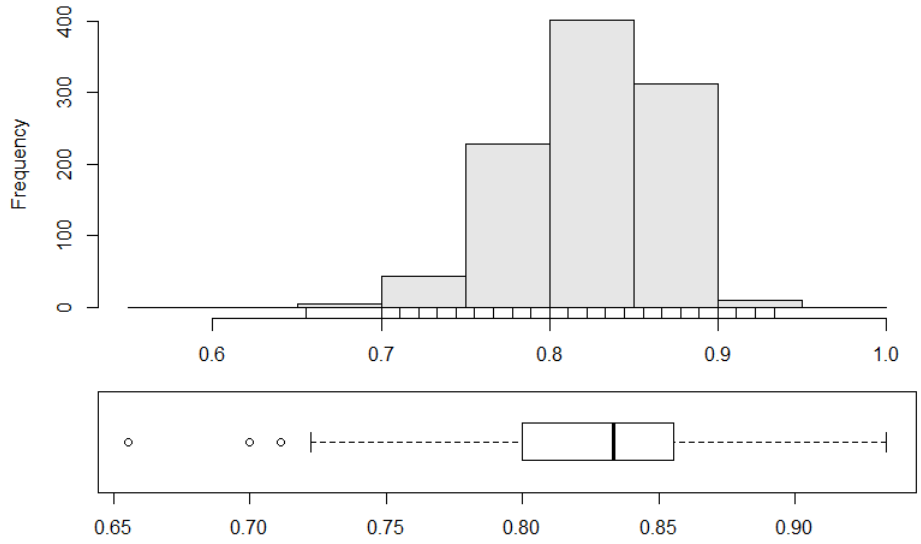

Figure 4.8: Histogram and boxplot of the classification accuracy scores obtained by testing the proposed classification models with input shape 70x40.

input size 70x40 (0.82, sd=0.04). The minimum accuracy was 0.66 and the maximum was 0.93. The accuracy distribution has a skew of -0.25 and a kurtosis of -0.06, indicating a negative skew and kurtosis. The standardised skew has a value of -3.24 and the normalised kurtosis has a value of 0.42. As the standardised skew is not within the range of ± 2, more investigation is necessary to ascertain if the data distribution can be considered to follow the normal distribution. The next step is to see if 95% of the standardised the accuracy scores falls within a standard score. As there are more than 80 observations, the score of $\pm$ 3.29 is used. There are 2 observations that fall outside the score of $\pm$ 3.29, representing 0.2% of the total. As this is less than 5%, this data can be considered to follow the normal distribution and can be analysed using parametric statistical methods. The breakdown of the distribution for each model is shown using boxplots in Figure 4.9.

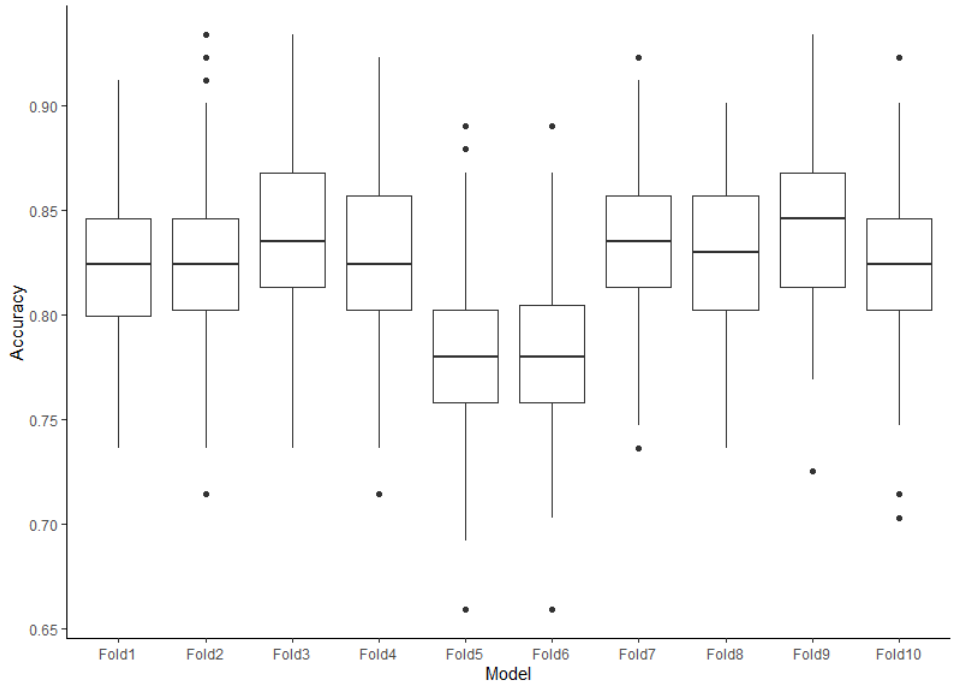

Figure 4.9: Classification accuracy results for the 10 fold proposed models using input shape 70x40.

Main Class	Precision	Recall	F1 Score
A	0.86	0.84	0.84
B	0.94	0.89	0.91
F	0.85	0.87	0.85
G	0.73	0.85	0.78
K	0.88	0.74	0.79
M	0.91	0.76	0.82
O	0.90	0.79	0.78

Table 4.5: Model evaluation scores for proposed model for input shape 70x40.

Table 4.5 shows the evaluation statistics of the model broken down by stellar class. Figure 4.10 shows the confusion matrix for the model and the accuracy per class.

Model with input shape 40x70

The hyper-parameters that provided the best validation accuracy are shown in Table A.3. This model also obtained better validation accuracy when using increased number of filters in each subsequent convolution layer. Figure 4.11 shows the histogram of the 1000 accuracy scores generated from the experiments on the proposed model using

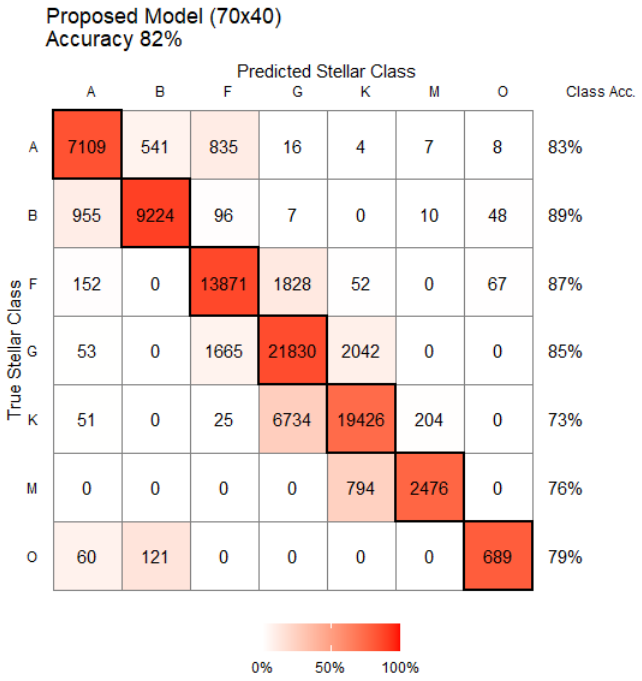

Figure 4.10: Confusion matrix showing the test results for the proposed model for input shape 70x40.

input size 40x70 (0.78, sd=0.06). The minimum accuracy was 0.58 and the maximum was 0.92. The accuracy distribution has a skew of -3.30 and a kurtosis of -0.12, indicating a negative skew and kurtosis. The standardised skew has a value of -4.26 and the normalised kurtosis has a value of -0.79. As the standardised skew is not within a range of ±2, more investigation is necessary to ascertain if the data distribution can be considered to follow the normal distribution. The next step is to see if 95% of the standardised the accuracy scores falls within a standard score. As there are more than 80 observations, the score of ± 3.29 is used. There are 2 observations that fall outside the score of ± 3.29, representing 0.2% of the total. As this is less than 5%, this data can be considered to follow the normal distribution and can be analysed using parametric statistical methods. The breakdown of the distribution for each model is shown using boxplots in Figure 4.12.

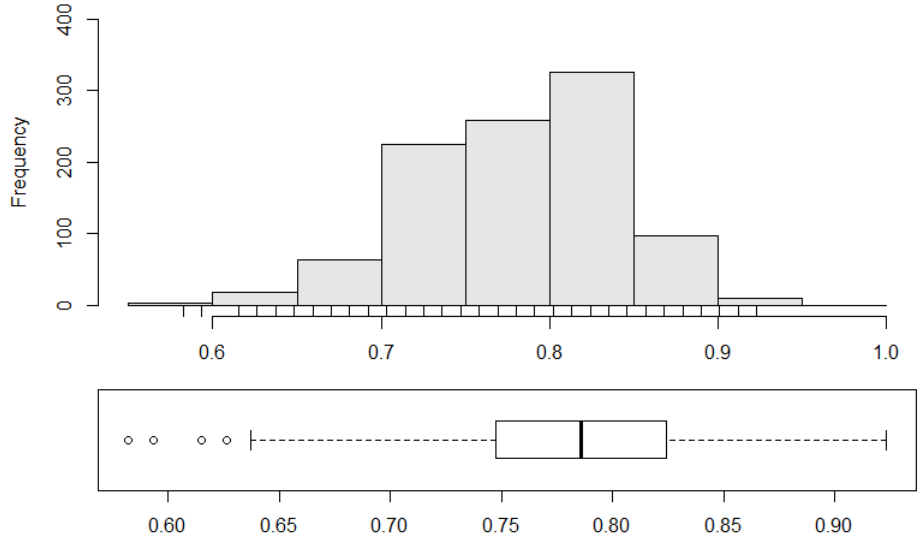

Figure 4.11: Histogram and boxplot of the classification accuracy scores obtained by testing the proposed classification models with input shape 40x70.

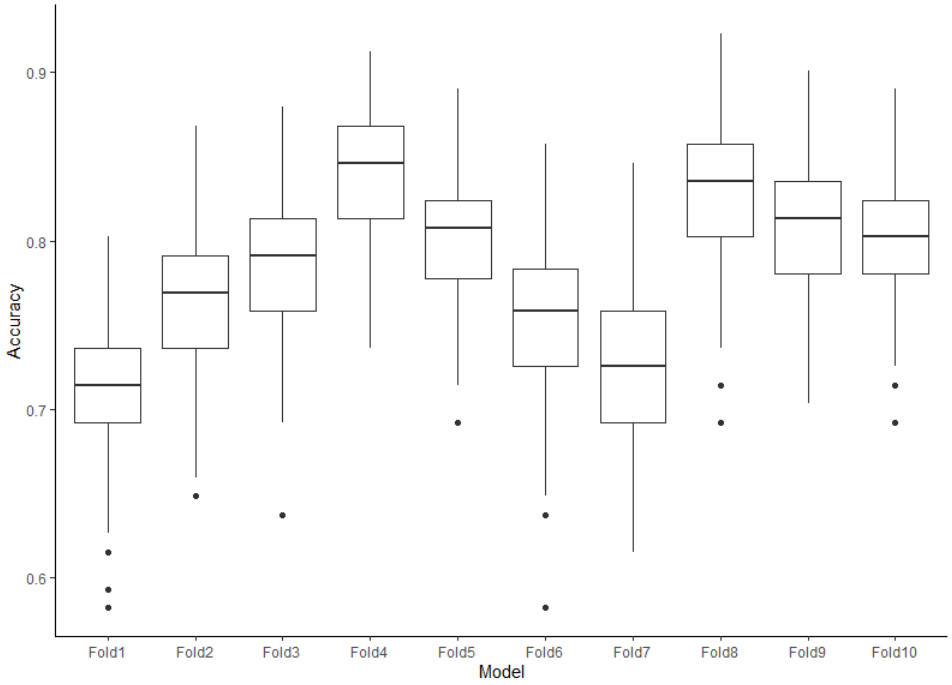

Figure 4.12: Classification accuracy results for the 10 fold proposed models using input shape 40x70.

Main Class	Precision	Recall	F1 Score
A	0.74	0.83	0.77
B	0.91	0.84	0.87
F	0.84	0.81	0.82
G	0.73	0.73	0.71
K	0.77	0.74	0.73
M	0.76	0.60	0.65
O	0.94	0.58	0.65

Table 4.6: Model evaluation scores for proposed model for input shape 40x70.

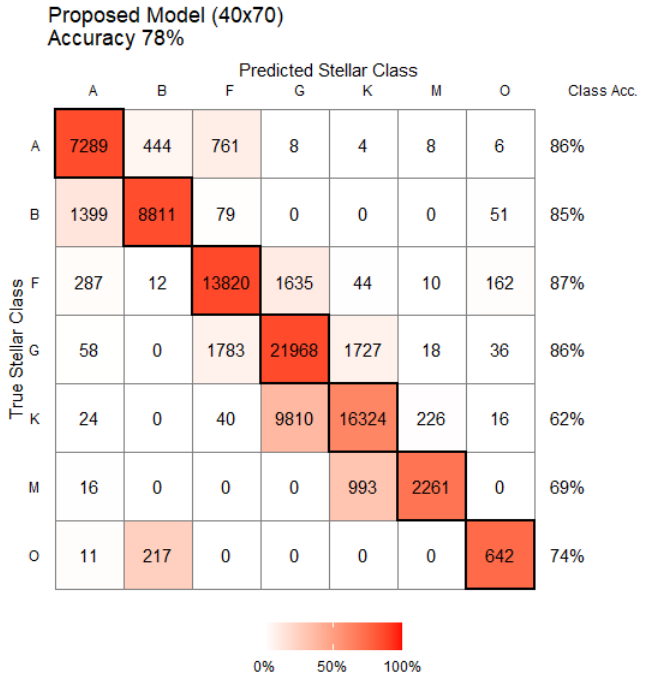

Figure 4.13: Confusion matrix showing the test results for the proposed model for input shape 40x70.

Table 4.6 shows the evaluation statistics of the model broken down by stellar class. Figure 4.13 shows the confusion matrix for the model and the accuracy per class.

Model with input shape 56x50

The hyper-parameters that provided the best validation accuracy are shown in Table A.4. This model also obtained better validation accuracy when using increased number of filters in each subsequent convolution layer. Figure 4.14 shows the histogram of the 1000 accuracy scores generated from the experiments on the proposed model using

input size 56x50 (0.79, sd=0.05). The minimum accuracy was 0.62 and the maximum was 0.93. The accuracy distribution has a skew of -0.21 and a kurtosis of -0.25, indicating a negative skew and kurtosis. The standardised skew has a value of -2.69 and the normalised kurtosis has a value of -1.62. As the standardised skew is not within a range of ±2, more investigation is necessary to ascertain if the data distribution can be considered to follow the normal distribution. The next step is to see if 95% of the standardised the accuracy scores falls within a standard score. As there are more than 80 observations, the score of ± 3.29 is used. There are no observations that fall outside the score of ± 3.29, indicating that this data can be considered to follow the normal distribution and can be analysed using parametric statistical methods. The breakdown of the distribution for each model is shown using boxplots in Figure 4.15.

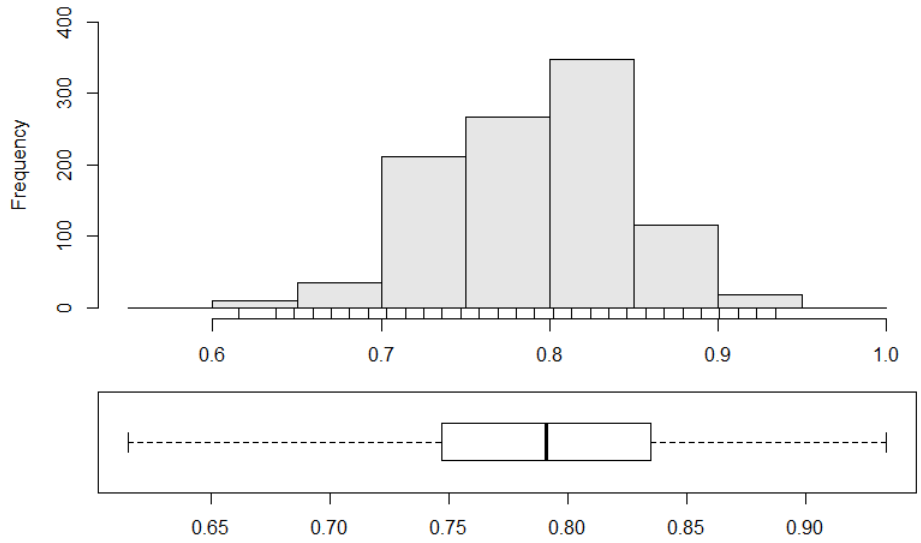

Figure 4.14: Histogram and boxplot of the classification accuracy scores obtained by testing the proposed classification models with input shape 56x50.

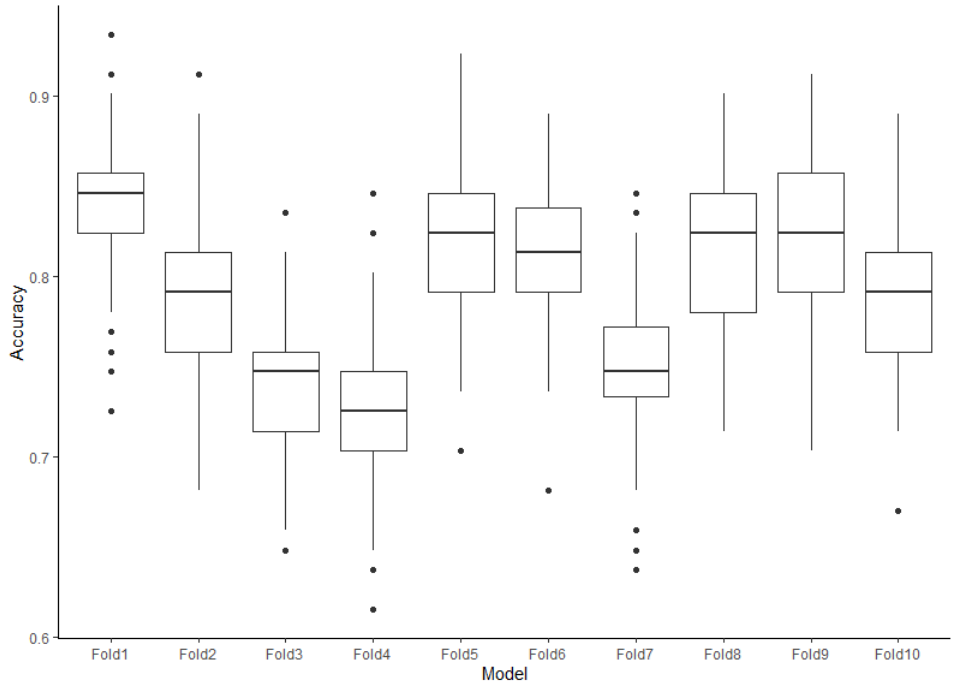

Figure 4.15: Classification accuracy results for the 10 fold proposed models using input shape 56x50.

Main Class	Precision	Recall	F1 Score
A	0.73	0.83	0.76
B	0.87	0.80	0.83
F	0.81	0.83	0.82
G	0.65	0.81	0.71
K	0.79	0.56	0.63
M	0.80	0.50	0.57
O	0.81	0.47	0.52

Table 4.7: Model evaluation scores for proposed model for input shape 56x50.

Table 4.7 shows the evaluation statistics of the model broken down by stellar class. Figure 4.16 shows the confusion matrix for the model and the accuracy per class.

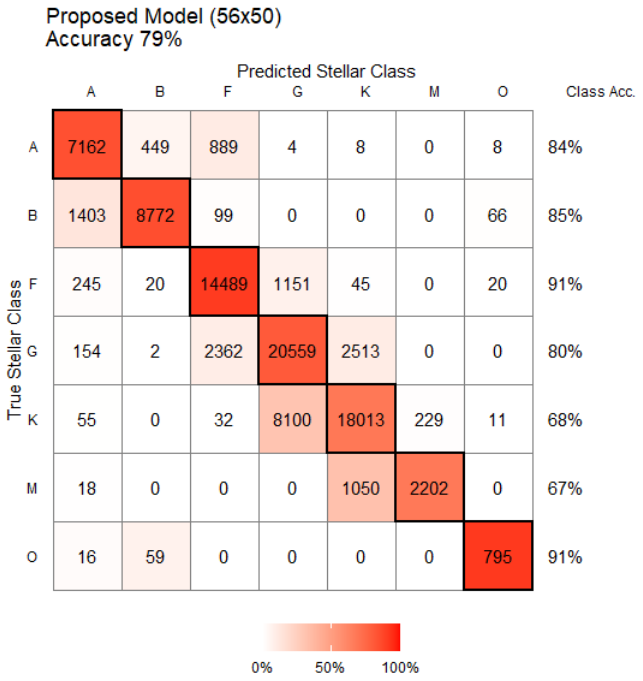

Figure 4.16: Confusion matrix showing the test results for the proposed model for input shape 56x50.

ResNet V2 50 model

There is no hyper-parameter tuning for the ResNet V2 50 model, this is supplied by the Keras.io framework [2]. 10 fold cross validation was applied to this model. ResNet was only trained with an input shape of 70x40 as this input shape provided the best mean accuracy performance of all the input shapes used in this research. Figure 4.17 shows the histogram of the 1000 accuracy scores generated from the experiments on the ResNet model using input size 70x40 (0.71, sd=0.06). The minimum accuracy was 0.52 and the maximum was 0.87. The accuracy distribution has a skew of -0.07 and a kurtosis of -2.66, indicating a negative skew and kurtosis. The standardised skew has a value of -0.88 and the normalised kurtosis has a value of -1.72. As the standardised skew and kurtosis are within a range of ± 2, this data can be considered to follow the normal distribution and can be analysed using parametric statistical methods. The breakdown of the distribution for each model is shown using boxplots in Figure 4.18.

[2]https://keras.io/api/applications/resnet/resnet50v2-function

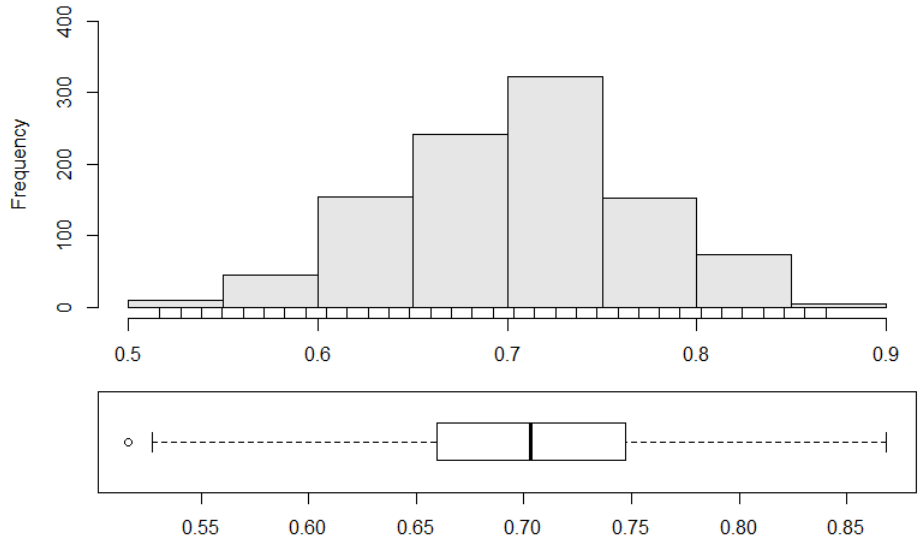

Figure 4.17: Histogram and boxplot of the classification accuracy scores obtained by testing the ResNetV250 model with input shape 70x40.

Table 4.8 shows the evaluation statistics of the ResNet V2 50 model broken down by stellar class. Figure 4.19 shows the confusion matrix for the model and the accuracy per class.

Main Class	Precision	Recall	F1 Score
A	0.67	0.80	0.72
B	0.84	0.69	0.75
F	0.79	0.85	0.82
G	0.63	0.78	0.69
K	0.75	0.56	0.61
M	0.91	0.53	0.67
O	0.42	0.31	0.36

Table 4.8: ResNet V2 50 model mean evaluation scores.

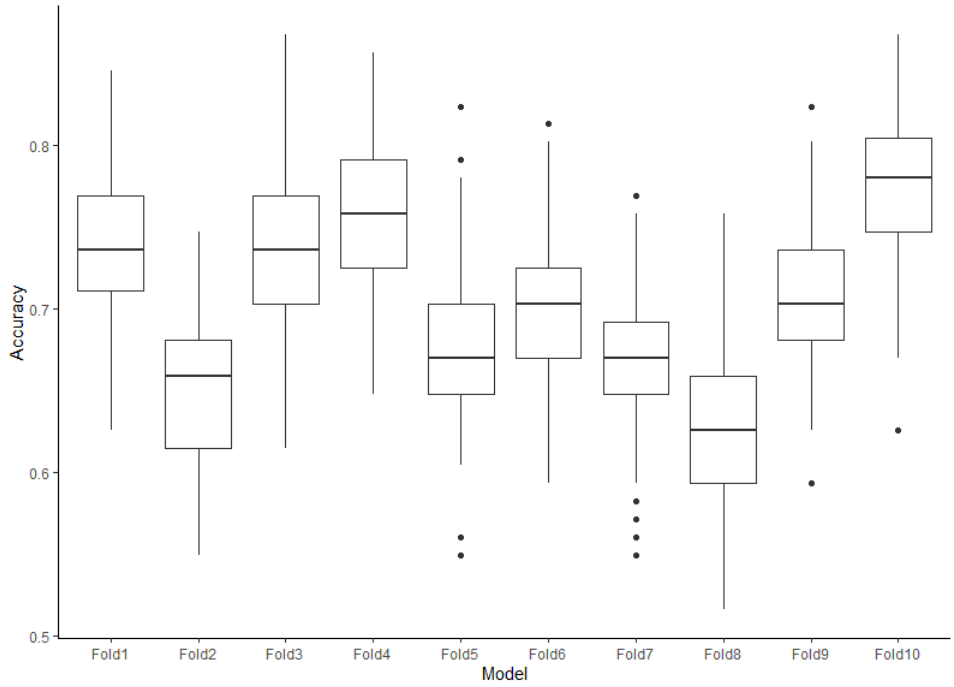

Figure 4.18: Classification accuracy results for the 10 fold ResNet V2 50 models using input shape 70x40.

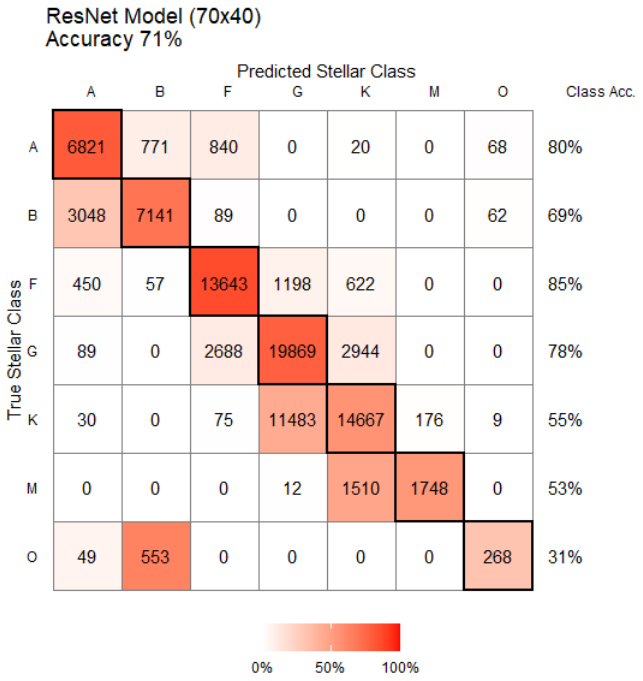

Figure 4.19: Confusion matrix showing the test results for the ResNetV250 classification models for input data 70x40.

4.3 Statistical Evaluation

All the accuracy distributions obtained during the experiments have been shown to follow the normal distribution. Levene's test for homogeneity of variance is applied to each accuracy distribution of the proposed model and the baseline model, to ascertain if a t-test or a Mann-Whitney U test was the most appropriate. An α value of 0.05 is defined for the significance test. Additional supporting information about the statistical evaluation performed in this research is provided in the Appendix in Section A.3.

Proposed Model with input 50x56

A Levene's test for homogeneity of variance was conducted and indicated equality of variance for accuracy scores for the baseline model and the proposed model with input shape 50x56 ($F(1,1998)=0.82$, $p=.36$). An independent-samples t-test was conducted to compare accuracy scores for the baseline model and proposed model. A significant difference in the accuracy scores was found ($M=0.80$, $SD=.05$ for the proposed model and $M=0.77$, $SD=.05$ for the baseline model), ($t(1998)=-0.04$, $p<.05$). The eta square statistic also indicated a small effect size (.12).

Proposed Model with input 70x40

A Levene's test for homogeneity of variance was conducted and indicated inequality of variance for accuracy scores for the baseline model and the proposed model with input shape 70x40 ($F(1,1998)=27.89$, $p<.05$). After performing a Mann-Whitney U test, the accuracy scores in the baseline model ($Mdn=0.77$) did differ significantly from the proposed model ($Mdn=0.82$), ($U=214579$, $z=22.14$, $p<.05$, $r=.49$).

Proposed Model with input 40x70

A Levene's test for homogeneity of variance was conducted and indicated inequality of variance for accuracy scores for the baseline model and the proposed model with input shape 40x70 ($F(1,1998)=14.18$, $p<.05$). After performing a Mann-Whitney U

test, the accuracy scores in the baseline model (Mdn=0.77) did differ significantly from the proposed model (Mdn=0.78), (U=415458, z=6.56, p<.05, r=.14).

Proposed Model with input 56x50

A Levene's test for homogeneity of variance was conducted and indicated inequality of variance for accuracy scores for the baseline model and the proposed model with input shape 56x50 (F(1,1998)=4.03, p<.05). After performing a Mann-Whitney U test, the accuracy scores in the baseline model (Mdn=0.77) did differ significantly from the proposed model (Mdn=0.79), (U=368320, z=10.22, p<.05, r=.22).

ResNet V2 50 Model with input 70x40

A Levene's test for homogeneity of variance was conducted and indicated inequality of variance for accuracy scores for the baseline model and the ResNet V2 50 model with input shape 70x40 (F(1,1398)=34.77, p<.05). After performing a Mann-Whitney U test, the accuracy scores in the baseline model (Mdn=0.77) did differ significantly from the ResNet50 model (Mdn=0.70), (U=769840, z=20.92, p<.05, r=.49).

4.4 Discussion

The best performing neural network architecture was the architecture proposed by this research, achieving a mean accuracy value of 0.82. The baseline neural network architecture achieved a mean classification accuracy of 0.77. After using the appropriate statistical tests, this result is confirmed as a statistically significant result and the size effect is deemed as medium, r=.49. The criteria for a medium size effect is an r value >.3 and <.5. The result of this research indicates that the use of the 2D architecture did contribute to the increase in accuracy performance.

4.4.1 Strengths

This research demonstrated the ability of the proposed neural network architecture to learn the necessary features from, and generalise across, the different data sets.

Jiang et al. (2020) indicated that 2D CNN network architectures could be used for this purpose and this work reinforces this finding. The results presented here show that the approach for merging different data sets, first introduced by Gulati et al. (1994), does work. The validation accuracy observed while training the classification models indicates that this procedure can be improved, a point addressed in the next section. During training, the best validation accuracy achieved was 0.85 for the baseline architecture, 0.84 for the proposed architecture and 0.79 for the ResNet V2 50 architecture. Each network was trained using 1,805 spectra across the 7 different main stellar classes, indicating that the best baseline classification model failed to learn to classify 270 spectra, which is poor.

Sharma et al. (2019) was the first application of CNNs to the domain of stellar classification using the MK Classification scheme, producing very good results. Jiang et al. (2020) then applied 2D CNNs to a sub-domain of stellar classification, indicating that 2D CNNs could outperform 1D CNNs. In this research, the proposed 2D convolutional neural network architecture out performed both the baseline and ResNet network architectures. The proposed network architecture used two convolutional layers, this shallow approach out performing the deeper network architectures used in the baseline architecture and the ResNet V2 50 architecture. ResNet architectures are trained for many of thousands of epochs, something that was infeasible to do in this research (He et al., 2016a). It is possible that the ResNet architecture will perform better after more training.

4.4.2 Limitations

The data described in Sharma et al. (2019) are different to that used in this research due to the unavailability of the ELODIE 3.1 catalogue. While the ELODIE data is used in this research, there is no way to know how this compares to the data used by Sharma et al. (2019).

The data sets used is imbalanced, meaning that stars of each class are not equally represented, previously shown in Table 3.1. The confusion matrix for each classification model generated by the 10 fold cross validation process shows that some classification

models were unable to identify O class stars at all. This was mainly a problem for the ResNet V2 50 architecture where 4 of the 10 classification models could not identify O class stars, but it also affected the proposed architecture. The small number of O class stars in the test data set means that this issue didn't impact the over all classification accuracy score as much as it should have if the test data set was balanced.

The process for normalising and merging different data sets used in Sharma et al. (2019) was first introduced in 1994 by Gulati et al. (1994). This process is designed to degrade spectra to a common resolution. The common resolution is that of the data set with the lowest resolution. Interpolation using a cubic spline is then used to generate spectra with a resolution of 1Å. When this process was introduced in 1994, 3 of the 4 data sets used in this research did not exist. The only existing data set, JHC, was used by Gulati et al. (1994) in their research. There are limitations in this process, however, as it doesn't seem to account for the exposure of each spectra. Figure 4.20 shows samples of processed spectra (after interpolation) for all four data sets used in this research, each one of these images represents an O class star. It is clear to see that the exposure, or brightness, of the spectra are highly variable which would impact on a neural networks effectiveness to learn common features across these images. Contrast these images with that presented in Figure 3.9, where the features of the star are more prominent.

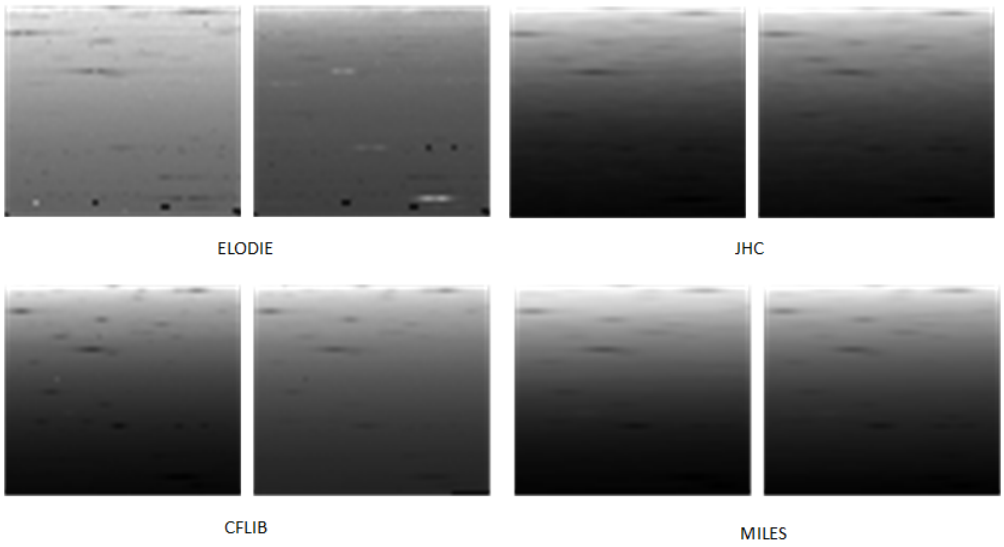

Figure 4.20: Different O class spectra showing different levels of brightness.

The architecture of the neural network used by Sharma et al. (2019) uses a smaller number of feature maps for each subsequent convolution layer. While there is no hard and fast rule about this, the heuristic is that CNN architectures increase the number of feature maps for deeper convolutional layers (Aggarwal, 2018). The reason for this is to maximise the models ability to learn features in deeper layers. The MaxPooling layers are used to decrease the input data dimensions, meaning each subsequent layer has less data to learn from, therefore, increasing the number of learned features is important. It is not clear why Sharma et al. (2019) configured the network architecture this way. The proposed network architecture used in this research followed a different approach. During hyper-parameter tuning, it was quickly observed that using the approach used by Sharma et al. (2019) did not give better accuracy results. The approach taken by Sharma et al. (2019) also increased the computational time required to train the classification models by front loading all the computations into the first convolutional layer (large input and deep feature maps).

Another point about the architecture used by Sharma et al. (2019) is that it contains 4 convolutional layers. He et al. (2016a) pointed out that deeper network architectures lead to higher training errors. Reviewing the training errors for the baseline network, it is clear to see that this is indeed the case here. Shown in Figure 4.21 is an example of the training and validation loss for a single classification model based on the baseline architecture. This shows the distinct pattern of diverging training and validation loss.

The ResNet V2 50 architecture didn't perform well on this type of data, this result is consistent with that reported by Jiang et al. (2020). Of the three architectures used in this research, it provided the poorest classification accuracy results. The mean accuracy for the baseline classification model was 0.77, the best performing proposed classification model obtained a mean accuracy of 0.82, but the ResNet classification model only achieved a mean accuracy of 0.71. When the ResNet architecture was introduced, it was trained and tested on images with dimensions 224 pixels wide by 224 pixels high. This represents significantly more input pixels (80,176) than the input used in this research (2,800). It is possible that the ResNet architecture doesn't work

so well with smaller input dimensions. The ResNet classification model was also only trained for 200 epochs compared to 1000 for the other classification models. This may also have impacted on the overall performance.

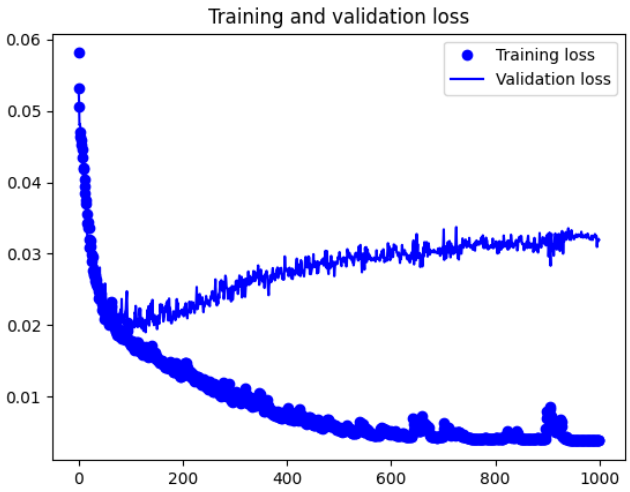

Figure 4.21: Training error rates for one of the baseline classification models.

The proposed approach uses four different input shapes when folding the spectra into the matrix format, 50x56, 40x70, 70x40 and 56x50. All classification models obtained statistically significant results. The effect size for classification models for three of the four input shapes is small, the remaining effect sizes are medium. It is unclear why an input shape of 70x40 should produce a bigger effect size than the other input shapes. In reality, the input shapes are somewhat arbitrary, so it is not clear why this input shape out performed the other three, or whether this shape is meaningful when applied to other data sets. Jiang et al. (2020) fold spectra into a matrix of order 50x100, which is larger than any matrix orders used in this research. This is by necessity as there was insufficient data to create an input shape the same as Jiang et al. (2020).

The accuracy results reported by Sharma et al. (2019) could not be replicated.

4.5 Summary

In this section, the experimental results are presented, including statistical testing, for the baseline network architecture, the proposed architecture and the ResNet V2 50 architecture. The proposed network architecture produced the classification models that obtained the highest average accuracy of 0.82. When this is compared to the baseline classification models, the result is statistically significant with a medium effect size.

Chapter 5

Conclusion

This chapter discusses the overall dissertation, the results obtained by this research and compares these results to those reported by Sharma et al. (2019). It also attempts to understand the difference in results and the reasons why these differences occurred. Finally, the contributions and impact of this work is discussed.

5.1 Research Overview

The aim of this research was to apply modern machine learning techniques to the problem of stellar classification using the MK Classification scheme and produce a state-of-the-art classification model. The current state-of-the-art classification model uses a 1D CNN to directly classify stellar spectra without the need for an explicit feature extraction process. Introduced in 1918, the Harvard Classification Scheme classifies stars based on surface temperature (T_{eff}). In 1943, the MK Classification Scheme was introduced, introducing the concept of a stellar luminosity class. This is derived from stellar surface gravity (log g), since the luminosity is correlated with the surface density of a star. Other stellar attributes are used to further refine classifications. Traditionally, stars were classified manually, the process requiring extraction of stellar indices from a spectrum, these are then classified by comparing them against previously classified spectra. As modern astronomical surveys begin to capture more and more data, the manual classification approach simply cannot keep pace with the

rate at which data is being collected.

5.2 Problem Definition

There are an estimated 200 billion stars in the Milky Way galaxy (Karttunen et al., 2006, p. 7). Not all stars are the same, they differ widely in terms of size, colour, distance and brightness. To discuss stars, it is necessary to have a classification scheme to be able to discuss the star in context. The MK Classification Scheme provides this context and has been in use since 1943.

Given the vast distances to stars the only information astronomers have about them is the light they emit. Through the application of spectroscopy, astronomers have been able to gain vast amounts of information about individual stars. Astronomers use an instrument called a spectrograph to produce a spectrum from a light source. The light source is passed through a diffraction grating or prism and a spectrum is created based on a physical process called diffraction. The spectrograph measures the spectral flux density at specific wavelengths. Each spectrograph has a defined wavelength range and a resolution. The resolution determines how many wavelength points are recorded over the wavelength range. Stars are classified by extracting features from their spectrum. Common features are surface temperature (T_{eff}), surface gravity (log g) and metallicity (Fe/H). The presence or absence of lines at different wavelengths that can be grouped together to identify an element are known as spectral indices and these are also used in the classification process.

Machine learning is an established technology in the domain of stellar classification and has been used to both predict stellar atmospheric properties and to directly classify stars based on the MK Classification Scheme. Recent research has focused on the application of CNNs to the domain of stellar classification. Supervised learning techniques allow the training of classification models directly from the stellar spectra without the need for specialized knowledge required for complex spectral analysis. There are also numerous data sets and databases available with labelled data that can be used for this process. Traditionally, spectra are considered as sequential data,

meaning that the wavelength data must be in a sequence for it to be meaningful. 1D CNN was the first CNN architecture applied to stellar spectra classification. 1D CNNs also have the benefit of being less computationally complex than 2D CNNs.

This research makes the claim that spectra data can be treated in a two-dimensional (folded) representation for the purposes of stellar classification and this claim is backed up by research by Jiang et al. (2020). The results of this research are compared with the state-of-the-art 1D CNN classification model.

The research question posed was: To what extent can the accuracy of stellar classification using the MK Classification Scheme be improved by using 2D CNNs with folded spectra?

5.3 Design/Experimentation, Evaluation & Results

This section discusses the experiment design, how the results were evaluated and the overall results. The final conclusion is presented about this research.

5.3.1 Design/Experimentation

The experiment uses four existing data sets as set out by Sharma et al. (2019). These data sets are the CFLIB, JHC, ELOIDE and MILES data sets. The CFLIB data set is retained exclusively for testing the trained classification models. Each of these data sets contain labelled stellar spectra. The ELODIE 3.1 catalogue was unavailable, so it was necessary to recreate a data set from the publicly available data.

Each data set was captured at a different resolution and have different wavelength ranges. It is necessary to harmonise these data sets to use them together. The harmonisation process consisted of degrading each spectra the lowest resolution in the data sets, then using interpolation with a cubic spline to create spectra at a common resolution of 1Å.

The first classification model created was that of the baseline model. This is a 1D CNN model, the specification is outlined in Sharma et al. (2019). All the hyperparameters for this model are fixed, as is the input shape of the data. 10-fold cross

validation was used to train the baseline classification models. The baseline models were trained for 1000 epochs using the root mean squared error as the loss function. The baseline classification model with the best validation and training accuracy for each fold was retained for testing.

The proposed 2D CNN architecture contained ten tuneable hyper-parameters, as discussed in Section 3.4.4. In order to determine the hyper-parameters that provide the best classification accuracy, the training data was split into a training and validation data set. Different hyper-parameter values were used to train classification models, and these models were evaluated with the validation data set. The hyper-parameter that provided the best classification accuracy on the validation data set were retained. The model hyper-parameters are determined for four different data input shapes, these are 50x56, 40x70, 70x40 and 56x50. Once the hyper-parameters were fixed, 10-fold cross validation was used to train the classification models. Models were trained for 1000 epochs using the root mean squared error as the loss function. The classification models with the best training accuracy for each fold are retrained for testing.

The testing procedure consists of generating 100 different test data sets from the CFLIB data set, this is selection with replacement, meaning that the same sample will be reused multiple times. Each selection contains 8% of the total CFLIB data set. There are 10 models for each input shape, and each of these models are tested with the 100 test data sets, resulting in 1,000 classification accuracy scores for each input shape. The average of the 1,000 accuracy scores is then used to determine what input shape model performs best. The best model is then compared to the results reported by Sharma et al. (2019). A ResNet V2 50 model was trained using the input shape that provided the best average classification accuracy. The performance of this model is compared to the other classification models.

5.3.2 Evaluation

The output of the experiment was a list of 1,000 classification accuracy scores for each of the baseline classification models, the proposed models for each input shape (4x1,000 accuracy scores), and for the ResNet V2 50 model. An accuracy score can have a value

anywhere between 0 and 1, inclusive, so this is a ratio data type. The distributions for each classification accuracy set was checked to see if they conformed to the normal distribution, this is a pre-requisite before parametric statistical methods can be used on the data. Each data set was found to conform to the normal distribution.

The statistical evaluation process compared the classification accuracy scores of the baseline model with the 4 proposed classification models and the ResNet model. A Levene's test for homogeneity of variance was conducted on the baseline model accuracy score against each of the proposed classification models and ResNet. Where equality of variance was confirmed, an independent t-test was used to compare the two distributions. Where inequality of variance was confirmed, a Mann-Whitney U-test was used to compare the two distributions. The appropriate effect size was calculated depending on the type of test performed on the data.

5.3.3 Results

This section discusses the results of this research and then compares these to the results presented in Sharma et al. (2019).

Results of this research

All the results obtained by this research were statistically significant. This is to be expected as there are 1,000 results for each classification model, a sufficient amount to guarantee statistical significance (Field et al., 2012). However, the effects size for the models is small, except for the classification model that used the input shape 70x40 and the ResNet model.

Based on the experiments, the best performing neural network architecture was the architecture proposed by this research using the input shape 70x40, achieving a mean accuracy value of 0.82. The baseline neural network architecture achieved a mean classification accuracy of 0.77. After using the appropriate statistical tests, this result is confirmed as a statistically significant results and the size effect is deemed as medium, r=.49. The criteria for the size effect is an r value $>.3$ and $<.5$, so the result of this research indicates that the use of the 2D architecture did contribute to the

increase in accuracy performance. Interestingly, the ResNet architecture performed worse than the baseline classification model (m=.71), and the size effect was medium (.49).

Comparing results to Sharma et al. (2019)

Sharma et al. (2019) report a mean accuracy of 89% for their classification model. Their model was reproduced in this research, labeled the baseline model, and used to compare with the results from the proposed model. This research was unable to reproduce the mean classification accuracy obtained by Sharma et al. (2019). Differences were introduced over the course of this research that diverged from the process as outlined by Sharma et al. (2019) that were unavoidable. For example, Sharma et al. (2019) used the ELODIE 3.1 catalogue in their research, but this catalogue was never officially published and existing links to this are now dead, so this could not be used.

Sharma et al. (2019) also performed weight transfer from an auto-encoder model before the training process. This step was not replicated in this research as it was not possible to replicate the data set used in this step. The data set generated from the SDSS archive; there are currently 16 different data releases from SDSS, it is not stated what data release is used. Their data set consisted of 60,000 spectra, however, these objects are not document. Further more, the SDSS data is not normalised in the same way that the CFLIB, ELODIE, JHC or MILES data sets are, so it is not clear why this weight transfer is of any benefit in this context.

Another significant difference was in the cleaning process of the test data set. Sharma et al. (2019) removed 313 more samples from this data set than were removed in this research. Although the same criteria was applied for the removal of data, no justification could be found for the removal of the extra 313 samples. This must impact on the classification accuracy scores and go somewhere to explaining the difference in the results of this research and those published by Sharma et al. (2019). Table 5.1 outlines the F_1 scores published by Sharma et al. (2019) and those achieved by this research. It is clear to see that the results reported by Sharma et al. (2019) for every stellar class out perform those obtained in this research, especially the baseline model.

Class	Sharma	Baseline	50x56	40x70	56x50	70x40	ResNet
A	0.90	0.76	0.81	0.77	0.76	0.84	0.72
B	0.93	0.79	0.89	0.87	0.83	0.91	0.75
F	0.88	0.84	0.85	0.82	0.82	0.85	0.82
G	0.85	0.67	0.76	0.71	0.71	0.78	0.69
K	0.88	0.79	0.78	0.73	0.63	0.79	0.61
M	0.93	0.84	0.71	0.65	0.57	0.82	0.67
O	1.00	0.15	0.76	0.65	0.52	0.78	0.36

Table 5.1: Comparing the F_1 scores for this research and the values reported by Sharma et al. (2019)

5.3.4 Summary

This research has shown that folded stellar spectra outperforms the traditional 1D representation, regardless of the input shape used. The network architecture used in this research was also less complex than that used in the state-of-the-art classifier (Sharma et al., 2019). This research, however, was unable to reproduce the original results presented by Sharma et al. (2019), obtaining only an average classification accuracy of 77% for the baseline classification model, compared to 89% reported in Sharma et al. (2019). The data used in this research did differ from that used in Sharma et al. (2019). This is especially significant when the differences affect the test data set which can explain the difference in the reported classification accuracy results.

Based on the results generated by this research, the conclusion is that there is not enough evidence to support rejecting the null hypothesis.

5.4 Contributions and impact

This research builds on work from Sharma et al. (2019) and Jiang et al. (2020) to show that CNNs can be applied to the domain and that 2D CNNs have the ability to learn features in data that was previously considered sequential. By folding spectra into a matrix representation, it is possible to get higher classification accuracy using a 2D CNN. This research also showed that the ResNet V2 50 architecture provided a lower classification accuracy result, meaning that very deep neural networks might not be a

suitable approach in this domain.

5.5 Future Work & recommendations

The data processing procedure introduced by Gulati et al. (1994) works well but limitations were observed in the spectra produced. An opportunity exists here to extend this procedure to take into account the exposure levels of different surveys. For example, it was observed in this research that spectra with lower exposures tend to have a lower mean pixel value compared to those with longer exposures, building this check into the process is an area worth further investigation. Another area for future work is to use larger data sets e.g. SDSS data, that would not require data processing at all. Spectra that cover a wider wavelength range would also allow researchers to better understand the behaviour of deeper neural network architectures, such as ResNet, which didn't perform favourably in this research. Finally, other domains where one-dimensional data is classified, e.g. E.C.G, may benefit from this technique of classifying folded data with 2D CNNs.

Bibliography

Abdeljaber, O., Sassi, S., Avci, O., Kiranyaz, S., Ibrahim, A. A., & Gabbouj, M. (2019). Fault detection and severity identification of ball bearings by online condition monitoring. *IEEE Transactions on Industrial Electronics*, *66*(10), 8136–8147. https://doi.org/10.1109/TIE.2018.2886789

Aggarwal, C. C. (2018). *Neural networks and deep learning: A textbook*. Springer.

Bailer-Jones, C., Irwin, M., & Hippel, T. (1998). Automated classification of stellar spectra - ii. two-dimensional classification with neural networks and principal components analysis. *Monthly Notices of the Royal Astronomical Society*, *298*, 361–377.

Bazarghan, M. (2008). Automated classification of elodie stellar spectral library using probabilistic artificial neural networks. *arXiv: Astrophysics*.

Becker, I., Pichara, K., Catelan, M., Protopapas, P., Aguirre, C., & Nikzat, F. (2020). Scalable end-to-end recurrent neural network for variable star classification. *Monthly Notices of the Royal Astronomical Society*, *493*(2), 2981–2995. https://doi.org/10.1093/mnras/staa350

Binney, J., & Merrifield, M. (1998). *Galactic astronomy*. Princeton University Press.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Brice, M. J., & Andonie, R. (2019). Automated morgan keenan classification of observed stellar spectra collected by the sloan digital sky survey using a single classifier. *The Astronomical Journal*, *158*(5), 188. https://doi.org/10.3847/1538-3881/ab40d0

Cannon, A., & Pickering, E. (1918). The henry draper catalogue, 1–290.

Chandra, V., & Schlaufman, K. C. (2021). Searching for low-mass population III stars disguised as white dwarfs. *The Astronomical Journal*, *161*(4), 197. https://doi.org/10.3847/1538-3881/abe535

Corbally, C. J., & Gray, R. O. (2018). The spectral classification of stars over the last 200, 100, 75 years and in the future. *Proceedings of the International Astronomical Union*, *13*, 489–493. https://doi.org/10.1017/S1743921319000656

Dafonte, C., Rodríguez, A., Manteiga, M., Gómez, Á., & Arcay, B. (2020). A blended artificial intelligence approach for spectral classification of stars in massive astronomical surveys. *Entropy*, *22*(5), 518. https://doi.org/10.3390/e22050518

Faber, S. C., Friel, E., Burstein, D., & Gaskell, C. (1985). Definition of a system of spectral indices. *57*.

Fath, E., Arthur. (1934). *The elements of astronomy* (Third Edition). McGraw-Hill Book Company, Ic.

Field, A. P., Miles, J., & Field, Z. (2012). *Discovering statistics using r* [OCLC: ocn760970657]. Sage.

Fluke, C. J., & Jacobs, C. (2020). Surveying the reach and maturity of machine learning and artificial intelligence in astronomy. *WIREs Data Mining and Knowledge Discovery*, *10*(2). https://doi.org/10.1002/widm.1349

Gänsicke, B. T., Dillon, M., Southworth, J., Thorstensen, J. R., Rodríguez-Gil, P., Aungwerojwit, A., Marsh, T. R., Szkody, P., Barros, S. C. C., Casares, J., de Martino, D., Groot, P. J., Hakala, P., Kolb, U., Littlefair, S. P., Martínez-Pais, I. G., Nelemans, G., & Schreiber, M. R. (2009). SDSS unveils a population of intrinsically faint cataclysmic variables at the minimum orbital period. *Monthly Notices of the Royal Astronomical Society*, *397*(4), 2170–2188. https://doi.org/10.1111/j.1365-2966.2009.15126.x

Garcia-Dias, R., Prieto, C. A., Almeida, J. S., & Ordovás-Pascual, I. (2018). Machine learning in APOGEE: Unsupervised spectral classification with *K-means*. *Astronomy & Astrophysics*, *612*. https://doi.org/10.1051/0004-6361/201732134

Géron, A. (2017). *Hands-on machine learning with scikit-learn and TensorFlow: Concepts, tools, and techniques to build intelligent systems* (First edition) [OCLC: ocn953432302]. O'Reilly Media.

Gilmore, G., Randich, S., Asplund, M., Binney, J., Bonifacio, P., Drew, J., Feltzing, S., Ferguson, A., Jeffries, R., Micela, G., Negueruela, I., Prusti, T., Rix, H. .-W., Vallenari, A., Alfaro, E., Allende-Prieto, C., Babusiaux, C., Bensby, T., Blomme, R., ... Gaia-ESO Survey Team. (2012). The Gaia-ESO Public Spectroscopic Survey. *The Messenger*, *147*, 25–31.

Gray, R. O., & Corbally, C. J. (2014). An expert computer program for classifying stars on the mk spectral classification system. *The Astronomical Journal*, *147*(4), 80. https://doi.org/10.1088/0004-6256/147/4/80

Gray, R. O., Corbally, C. J., Cat, P. D., Fu, J. N., Ren, A. B., Shi, J. R., Luo, A. L., Zhang, H. T., Wu, Y., Cao, Z., Li, G., Zhang, Y., Hou, Y., & Wang, Y. (2015). LAMOST OBSERVATIONS IN THE *KEPLER* FIELD: SPECTRAL CLASSIFICATION WITH THE MKCLASS CODE. *The Astronomical Journal*, *151*(1), 13. https://doi.org/10.3847/0004-6256/151/1/13

Greene, S., & Jones, M. H. (2004). *An introduction to the sun and stars* (S. J. Burnell, Ed.; Co-published ed). Open University ; Cambridge University Press.

Greene, S., & Lambourne, R. (2006). *Introducing Astronomy*. The Open University ; Cambridge University Press.

Grey, O., R., Corbally, C., J., Garrison, R., F., McFadden, M., T., & Robinson, P., E. (2003). Contributions to the nearby stars (NStars) project: Spectroscopy of stars earlier than m0 within 40 parsecs: The northern sample. i. *The Astronomical Journal*, *126*(4).

Grey, O., R., Napier, M., G., & Winkler, L., I. (2001). The physical basis of luminosity classification in the late a-, f-, and early g-type stars. i. precise spectral types for 372 stars. *The Astronomical Journal*, *121*(4).

Gulati, R. K., Gupta, R., Gothoskar, P., & Khobragade, S. (1994). Stellar spectral classification using automated schemes. *The Astrophysical Journal*, *426*, 340. https://doi.org/10.1086/174069

Guo, P., & Lyu, M. R. (2004). A pseudoinverse learning algorithm for feedforward neural networks with stacked generalization applications to software reliability growth data. *Neurocomputing*, *56*, 101–121. https://doi.org/10.1016/S0925-2312(03)00385-0

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. *arXiv:1911.05722 [cs]*. http://arxiv.org/abs/1911.05722

He, K., Zhang, X., Ren, S., & Sun, J. (2016a). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. https://doi.org/10.1109/CVPR.2016.90

He, K., Zhang, X., Ren, S., & Sun, J. (2016b). Identity mappings in deep residual networks. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer vision – ECCV 2016* (pp. 630–645). Springer International Publishing. https://doi.org/10.1007/978-3-319-46493-0_38

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In F. Bach & D. Blei (Eds.), *Proceedings of the 32nd international conference on machine learning* (pp. 448–456). PMLR.

Ivezić, Ž., Connolly, A., Vanderplas, J. T., & Gray, A. (2020). *Statistics, data mining, and machine learning in astronomy: A practical python guide for the analysis of survey data*. Princeton University Press.

Izmodenov, V. V., & Alexashov, D. B. (2020). Magnitude and direction of the local interstellar magnetic field inferred from voyager 1 and 2 interstellar data and global heliospheric model. *Astronomy & Astrophysics*, *633*, L12. https://doi.org/10.1051/0004-6361/201937058

Jacoby, G. H., Hunter, D. A., & Christian, C. A. (1984). A library of stellar spectra. *The Astrophysical Journal Supplement Series*, *56*, 257. https://doi.org/10.1086/190983

Jiang, B., Wei, D., Liu, J., Wang, S., Cheng, L., Wang, Z., & Qu, M. (2020). Automated classification of massive spectra based on enhanced multi-scale coded

convolutional neural network. *Universe*, *6*(4), 60. https://doi.org/10.3390/universe6040060

Karttunen, H., Kroger, P., Oja, H., Poutanen, M., & Donner, K., J. (2006). *Fundamental astronomy* (Fifth Edition). Springer.

Kelleher, J. D. (2019). *Deep learning*. The MIT Press.

Kelleher, J. D., Mac Namee, B., & D'Arcy, A. (2015). *Fundamentals of machine learning for predictive data analytics: Algorithms, worked examples, and case studies*. The MIT Press.

Kheirdastan, S., & Bazarghan, M. (2016). Sdss-dr12 bulk stellar spectral classification: Artificial neural networks approach. *Astrophysics and Space Science*, *361*, 1–8.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In Y. Bengio & Y. LeCun (Eds.), *3rd international conference on learning representations, ICLR 2015, san diego, ca, usa, may 7-9, 2015, conference track proceedings*. http://arxiv.org/abs/1412.6980

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., & Inman, D. J. (2021). 1d convolutional neural networks and applications: A survey. *Mechanical Systems and Signal Processing*, *151*, 107398. https://doi.org/10.1016/j.ymssp.2020.107398

Kiranyaz, S., Gastli, A., Ben-Brahim, L., Al-Emadi, N., & Gabbouj, M. (2019). Real-time fault detection and identification for MMC using 1-d convolutional neural networks. *IEEE Transactions on Industrial Electronics*, *66*(11), 8760–8771. https://doi.org/10.1109/TIE.2018.2833045

Kiranyaz, S., Ince, T., & Gabbouj, M. (2016). Real-time patient-specific ECG classification by 1-d convolutional neural networks. *IEEE Transactions on Biomedical Engineering*, *63*(3), 664–675. https://doi.org/10.1109/TBME.2015.2468589

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, (6), 84–90. https://doi.org/10.1145/3065386

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recogni-

tion. *Neural Computation*, *1*(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

Li, J., Cui, R., Li, B., Song, R., Li, Y., & Du, Q. (2019). Hyperspectral image super-resolution with 1d–2d attentional convolutional neural network. *Remote Sensing*, *11*(23). https://doi.org/10.3390/rs11232859

Liu, C., Cui, W.-Y., Zhang, B., Wan, J.-C., Deng, L.-C., Hou, Y.-H., Wang, Y.-F., Yang, M., & Zhang, Y. (2015). Spectral classification of stars based on LAMOST spectra. *Research in Astronomy and Astrophysics*, *15*(8), 1137–1153. https://doi.org/10.1088/1674-4527/15/8/004

Mackenzie, C., Pichara, K., & Protopapas, P. (2016). Clustering-based feature learning on variable stars. *The Astrophysical Journal*, *820*(2), 138. https://doi.org/10.3847/0004-637X/820/2/138

MacPherson, H. (1926). *Modern astronomy. its rise and progress*. Oxford University Press.

Mahabal, A., Sheth, K., Gieseke, F., Pai, A., Djorgovski, S. G., Drake, A. J., & Graham, M. J. (2017). Deep-learnt classification of light curves. *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–8. https://doi.org/10.1109/SSCI.2017.8280984

Malkov, O., Kovaleva, D., Sichevsky, S., & Zhao, G. (2020). Statistical relations between stellar spectral and luminosity classes and stellar effective temperature and surface gravity. *Research in Astronomy and Astrophysics*, *20*(9), 139. https://doi.org/10.1088/1674-4527/20/9/139

Manteiga, M., Carricajo, I., Rodríguez, A., Dafonte, C., & Arcay, B. (2009). STARMIND: A FUZZY LOGIC KNOWLEDGE-BASED SYSTEM FOR THE AUTOMATED CLASSIFICATION OF STARS IN THE MK SYSTEM. *The Astronomical Journal*, *137*(2), 3245–3253. https://doi.org/10.1088/0004-6256/137/2/3245

Morgan, W., Keenan, P., & Kellman, E. (1943). *An atlas of stellar spectra, with an outline of spectral classification* (1st). The University of Chicago press.

Naul, B., Bloom, J. S., Pérez, F., & van der Walt, S. (2018). A recurrent neural network for classification of unevenly sampled variable stars. *Nature Astronomy*, *2*(2), 151–155. https://doi.org/10.1038/s41550-017-0321-z

Ohanian, H. C. (1989). *Physics* (2. ed., expanded) [OCLC: 830868594]. Norton.

Pickles, A., J. (1985). Spectral atlas.

Placco, V. M., Roederer, I. U., Lee, Y. S., Almeida-Fernandes, F., Herpich, F. R., Perottoni, H. D., Schoenell, W., Ribeiro, T., & Kanaan, A. (2021). SPLUS j210428.01004934.2: An ultra metal-poor star identified from narrowband photometry. *The Astrophysical Journal Letters*, *912*(2), L32. https://doi.org/10.3847/2041-8213/abf93d

Prugniel, P., & Soubiran, C. (2001). A database of high and medium-resolution stellar spectra. *Astronomy & Astrophysics*, *369*(3), 1048–1057. https://doi.org/10.1051/0004-6361:20010163

Prugniel, P., Soubiran, C., Koleva, M., & Borgne, D. L. (2007). New release of the ELODIE library: Version 3.1. *arXiv:astro-ph/0703658*. http://arxiv.org/abs/astro-ph/0703658

Ruiz, J. T., Pérez, J. D. B., & Blázquez, J. R. B. (2019). Arrhythmia detection using convolutional neural models. In F. De La Prieta, S. Omatu, & A. Fernández-Caballero (Eds.), *Distributed computing and artificial intelligence, 15th international conference* (pp. 120–127). Springer International Publishing.

Sanchez-Blazquez, P., Peletier, R. F., Jimenez-Vicente, J., Cardiel, N., Cenarro, A. J., Falcon-Barroso, J., Gorgas, J., Selam, S., & Vazdekis, A. (2006). Medium-resolution isaac newton telescope library of empirical spectra. *Monthly Notices of the Royal Astronomical Society*, *371*(2), 703–718. https://doi.org/10.1111/j.1365-2966.2006.10699.x

Schmidt, R., M., Schneider, F., & Hennig, P. (2020). Descending through a crowded valley – benchmarking deep learning optimizers. https://arxiv.org/pdf/2007.01547

Schorfheide, F., & Wolpin, K. I. (2012). On the use of holdout samples for model selection. *American Economic Review*, *102*(3), 477–481. https://doi.org/10.1257/aer.102.3.477

Sharma, K., Kembhavi, A., Kembhavi, A., Sivarani, T., Abraham, S., & Vaghmare, K. (2019). Application of convolutional neural networks for stellar spectral classification. *Monthly Notices of the Royal Astronomical Society*, stz3100. https://doi.org/10.1093/mnras/stz3100

Sharma, K., Singh, H. P., Gupta, R., Kembhavi, A., Vaghmare, K., Shi, J., Zhao, Y., Zhang, J., & Wu, Y. (2020). Stellar spectral interpolation using machine learning. *Monthly Notices of the Royal Astronomical Society*, *496*(4), 5002–5016. https://doi.org/10.1093/mnras/staa1809

Silva, R., D., & Cornell, E., M. (1992). Spectral atlas.

Singh, H. P., Gulati, R. K., & Gupta, R. (1998). Stellar spectral classification using principal component analysis and artificial neural networks. *Monthly Notices of the Royal Astronomical Society*, *295*(2), 312–318. https://doi.org/10.1046/j.1365-8711.1998.01255.x

Specht, F., D. (1990). Probabilistic neural networks. *Neural Networks*, *3*(1), 109–118. https://doi.org/10.1016/0893-6080(90)90049-Q

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. MIT Press.

Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9. https://doi.org/10.1109/CVPR.2015.7298594

Valdes, F., Gupta, R., Rose, J. A., Singh, H. P., & Bell, D. J. (2004). The indo-US library of coude feed stellar spectra. *The Astrophysical Journal Supplement Series*, *152*(2), 251–259. https://doi.org/10.1086/386343

Wang, K., Guo, P., & Luo, A. (2017). A new automated spectral feature extraction method and its application in spectral classification and defective spectra recovery. *Monthly Notices of the Royal Astronomical Society*, *465*, 4311–4324.

Way, M. J. (2016). *Advances in machine learning and data mining for astronomy.* [OCLC: 957680888]. CRC Press.

Weaver, W. B., & Torres-Dodgen, A. V. (1997). Accurate two-dimensional classification of stellar spectra with artificial neural networks. *The Astrophysical Journal*, *487*(2), 847–857. https://doi.org/10.1086/304651

Wilson, A. C., Roelofs, R., Stern, M., Srebro, N., & Recht, B. (2017). The marginal value of adaptive gradient methods in machine learning. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 4151–4161.

Zhang, W., Li, C., Peng, G., Chen, Y., & Zhang, Z. (2018). A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load. *Mechanical Systems and Signal Processing*, *100*, 439–453. https://doi.org/10.1016/j.ymssp.2017.06.022

Appendix A

Additional content

A.1 Tuned Hyper-parameters

The tables in the following pages contain the hyper-parameters used to train the CNNs in this research. Static values for the activation function (RELU) padding (SAME) are used for all models.

Layer	Hyper-parameters	Values
Convolution layer 1	Filters	16
	Kernel	5
	Strides	1,1
Max Pooling layer 1	Pool size	2,2
Convolution Layer 2	Filters	32
	Kernel	3
	Strides	2,2
Max Pooling layer 2	Pool size	1,1
Dense Layer 1	Units	64
Dropout 1	Rate	0.01
Dense Layer 2	Units	32
Dropout 2	Rate	0.01

Table A.1: Proposed model hyper-parameters for input shape 50x56.

Layer	Hyper-parameters	Values
Convolution layer 1	Filters	16
	Kernel	7
	Strides	1,1
Max Pooling layer 1	Pool size	2,2
Convolution Layer 2	Filters	32
	Kernel	5
	Strides	1,1
Max Pooling layer 2	Pool size	2,2
Dense Layer 1	Units	64
Dropout 1	Rate	0.01
Dense Layer 2	Units	32
Dropout 2	Rate	0.01

Table A.2: Proposed model hyper-parameters for input shape 70x40.

Layer	Hyper-parameters	Values
Convolution layer 1	Filters	16
	Kernel	5
	Strides	1,1
Max Pooling layer 1	Pool size	2,2
Convolution Layer 2	Filters	32
	Kernel	5
	Strides	1,1
Max Pooling layer 2	Pool size	2,2
Dense Layer 1	Units	64
Dropout 1	Rate	0.01
Dense Layer 2	Units	32
Dropout 2	Rate	0.01

Table A.3: Proposed model hyper-parameters for input shape 40x70.

Layer	Hyper-parameters	Values
Convolution layer 1	Filters	16
	Kernel	5
	Strides	1,1
Max Pooling layer 1	Pool size	2,2
Convolution Layer 2	Filters	32
	Kernel	5
	Strides	1,1
Max Pooling layer 2	Pool size	2,2
Dense Layer 1	Units	64
Dropout 1	Rate	0.01
Dense Layer 2	Units	32
Dropout 2	Rate	0.01

Table A.4: Proposed model hyper-parameters for input shape 56x50.

A.2 Baseline model hyper-parameters

Layer	Hyper-parameters	Values
Convolution layer 1	Activation function	RELU
	Filters	128
	Kernel	4
	Padding	Same
	Strides	2
Max Pooling	Pool size	2
Convolution Layer 2	Activation function	RELU
	Filters	64
	Kernel	4
	Padding	Same
	Strides	2
Max Pooling	Pool size	2
Convolution Layer 3	Activation function	RELU
	Filters	32
	Kernel	4
	Padding	Same
	Strides	2
Max Pooling	Pool size	2
Convolution Layer 4	Activation function	RELU
	Filters	16
	Kernel	4
	Padding	Same
	Strides	2
Flatten	N/A	
Dense Layer 1	Activation function	RELU
	Units	64
Dense Layer 2	Activation function	RELU
	Units	32
Output Layer	Activation function	softmax
	Units	7

Table A.5: Baseline network hyper-parameter settings.

A.3 Statistical Information

This section provides more information about the statistical checks that were performed on the classification accuracy score distributions for each model. These checks include checks for normality, Levene's test for homogeneity of variance, t-test and Mann-Whitney U test outputs. QQPlots and histograms of the standardised accuracy

scores are included for each accuracy score distribution. The QQPlot is used as a visual reference to see how the accuracy score distribution varies from a theoretical normal distribution (Field et al., 2012).

A.3.1 Baseline model

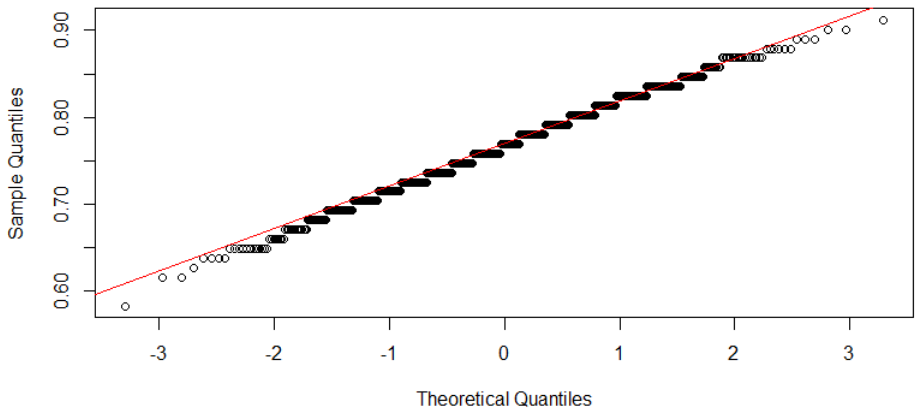

Figure A.1: QQPlot of the accuracy scores for the baseline model.

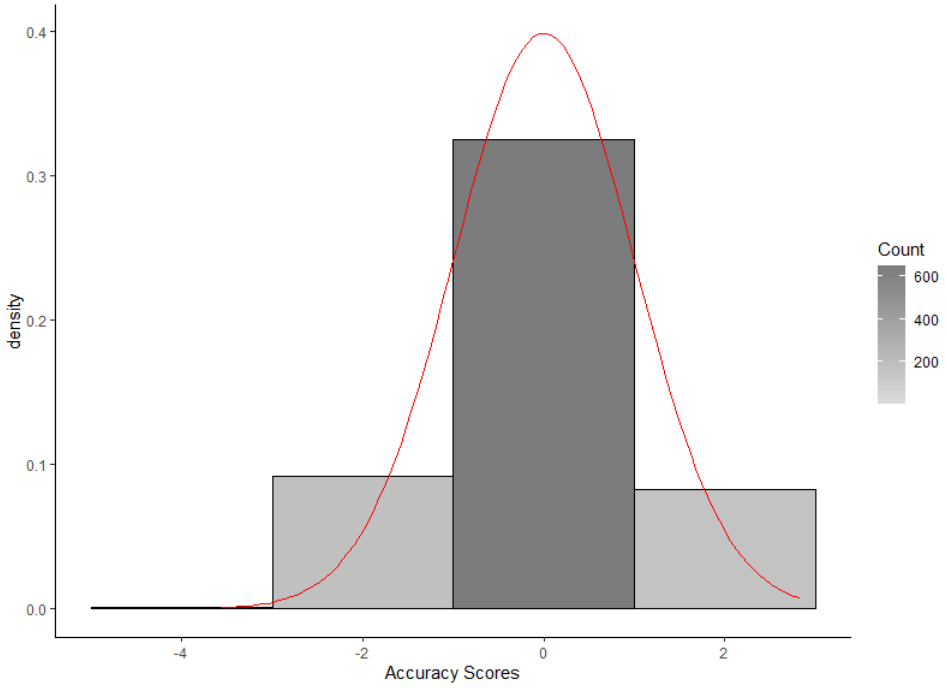

Figure A.2: Histogram of the standardised accuracy scores for the baseline model.

A.3.2 Proposed model with input shape 50x56

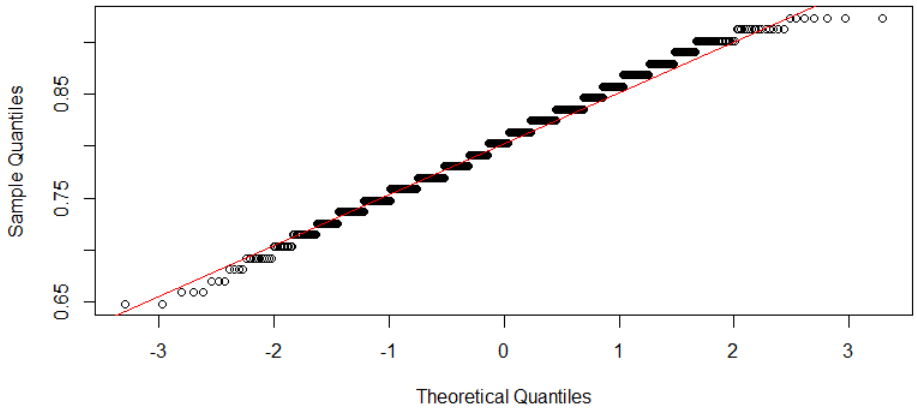

Figure A.3: QQPlot of the accuracy scores for the proposed model with input shape 50x56.

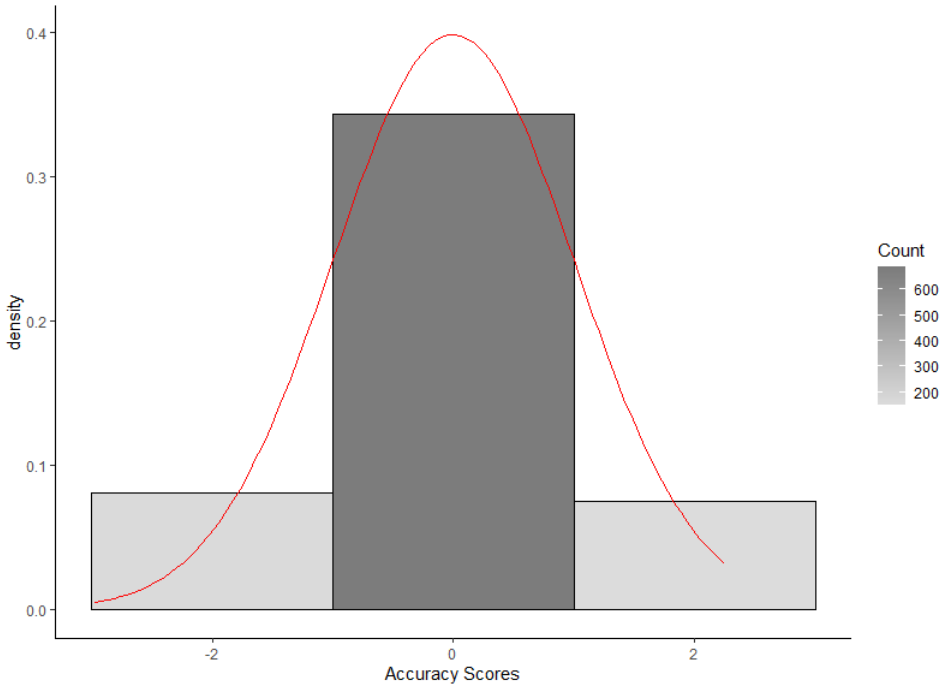

Figure A.4: Histogram of the standardised accuracy scores for the proposed model with input shape 50x56.

```
Levene's Test for Homogeneity of Variance (center = median)
       Df F value Pr(>F)
group   1  0.8232 0.3644
      1998
```

Figure A.5: Output of the Levene's test for homogeneity of variance for the model with input shape 50x56.

```
        Two Sample t-test

data:  Accuracy by Model
t = -16.582, df = 1998, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.04317007 -0.03403872
sample estimates:
mean in group REF mean in group PRO
        0.7660220         0.8046264
```

Figure A.6: Output of the independent t-test for the model with input shape 50x56.

A.3.3 Proposed model with input shape 70x40

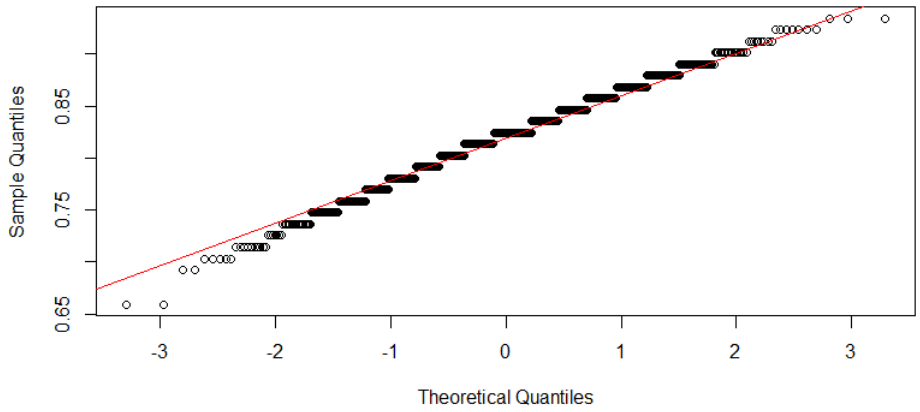

Figure A.7: QQPlot of the accuracy scores for the proposed model with input shape 70x40.

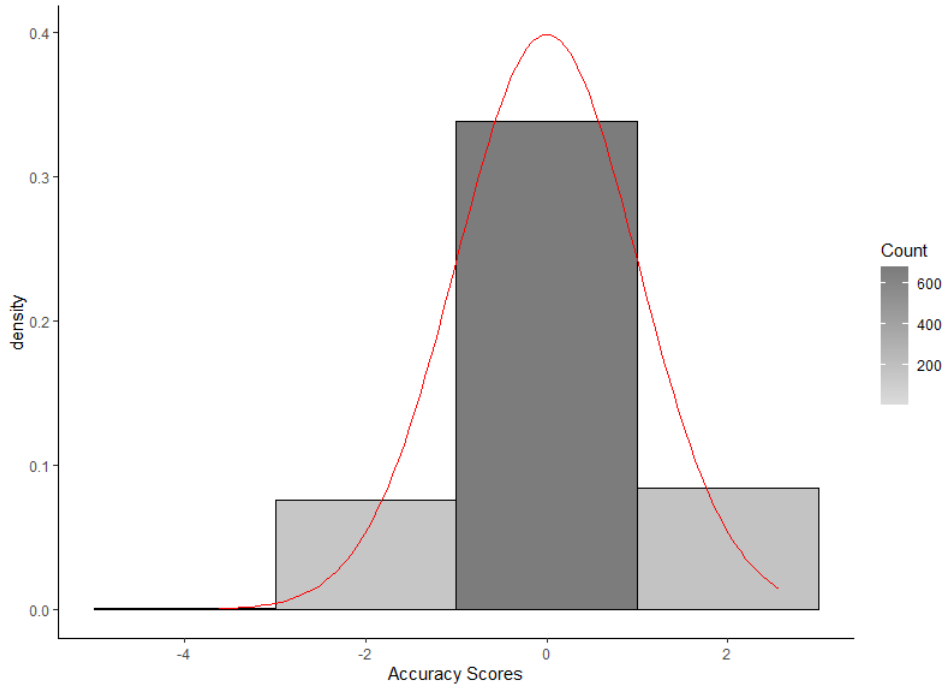

Figure A.8: Histogram of the standardised accuracy scores for the proposed model with input shape 70x40

```
Levene's Test for Homogeneity of Variance (center = median)
        Df F value    Pr(>F)
group    1  27.893 1.422e-07 ***
      1998
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure A.9: Output of the Levene's test for homogeneity of variance for the model with input shape 70x40.

```
        Wilcoxon rank sum test with continuity correction

data:  Accuracy by Model
W = 214579, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0
```

Figure A.10: Output of the Mann-Whitney U test for the model with input shape 70x40.

A.3.4 Proposed model with input shape 40x70

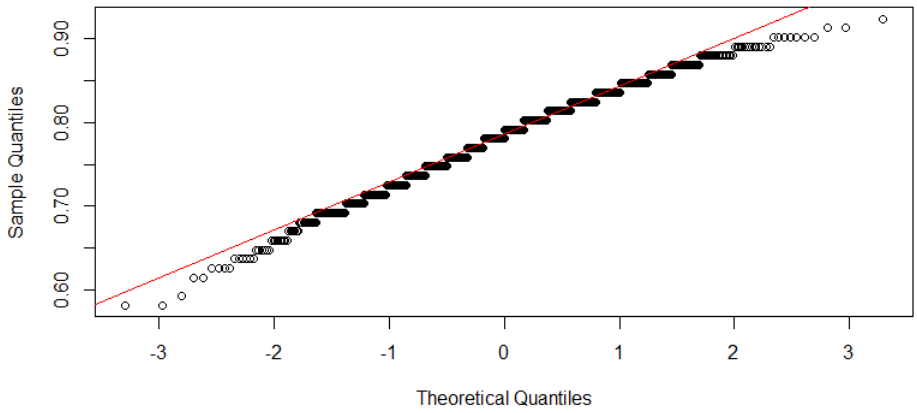

Figure A.11: QQPlot of the accuracy scores for the proposed model with input shape 40x70.

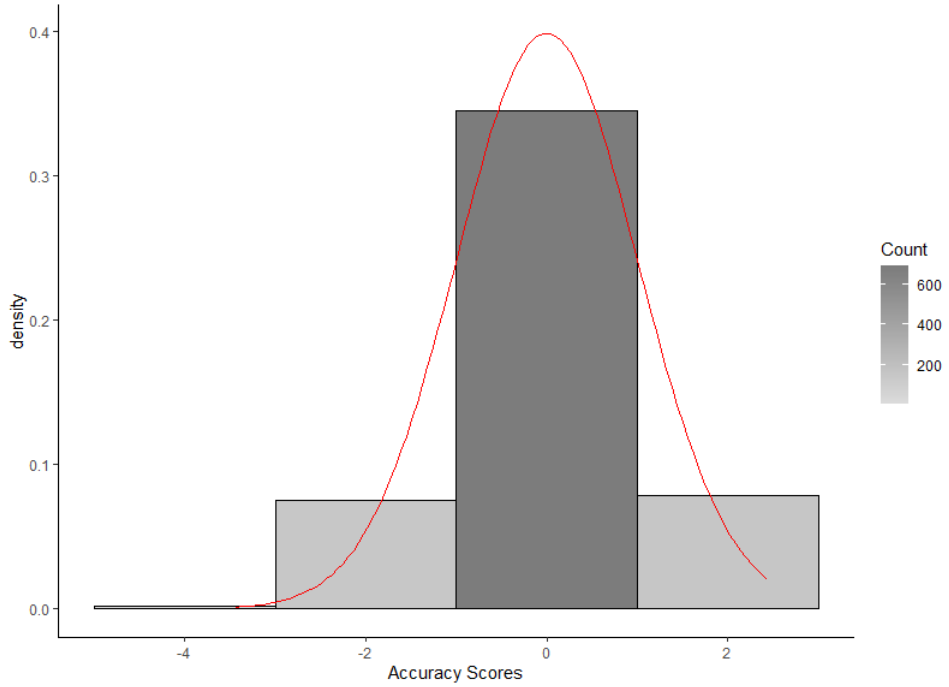

Figure A.12: Histogram of the standardised accuracy scores for the proposed model with input shape 40x70.

```
Levene's Test for Homogeneity of Variance (center = median)
       Df F value    Pr(>F)
group   1  14.183 0.0001707 ***
      1998
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure A.13: Output of the Levene's test for homogeneity of variance for the model with input shape 40x70.

```
        Wilcoxon rank sum test with continuity correction

data:  Accuracy by Model
W = 415458, p-value = 5.431e-11
alternative hypothesis: true location shift is not equal to 0
```

Figure A.14: Output of the Mann-Whitney U test for the model with input shape 40x70.

A.3.5 Proposed model with input shape 56x50

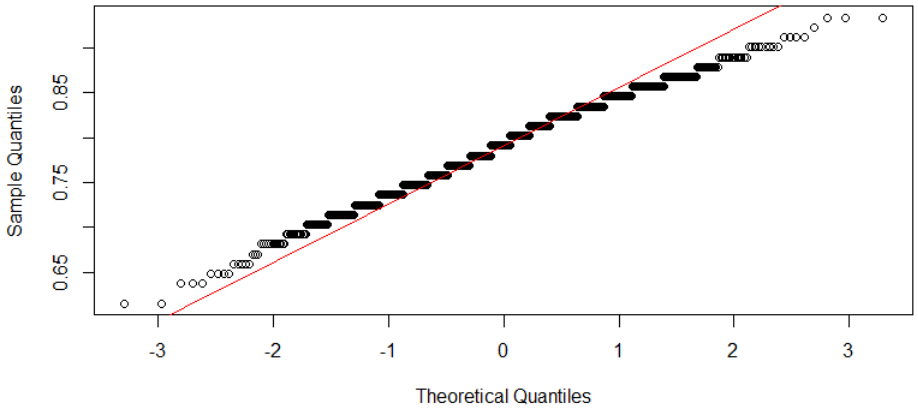

Figure A.15: QQPlot of the accuracy scores for the proposed model with input shape 56x50.

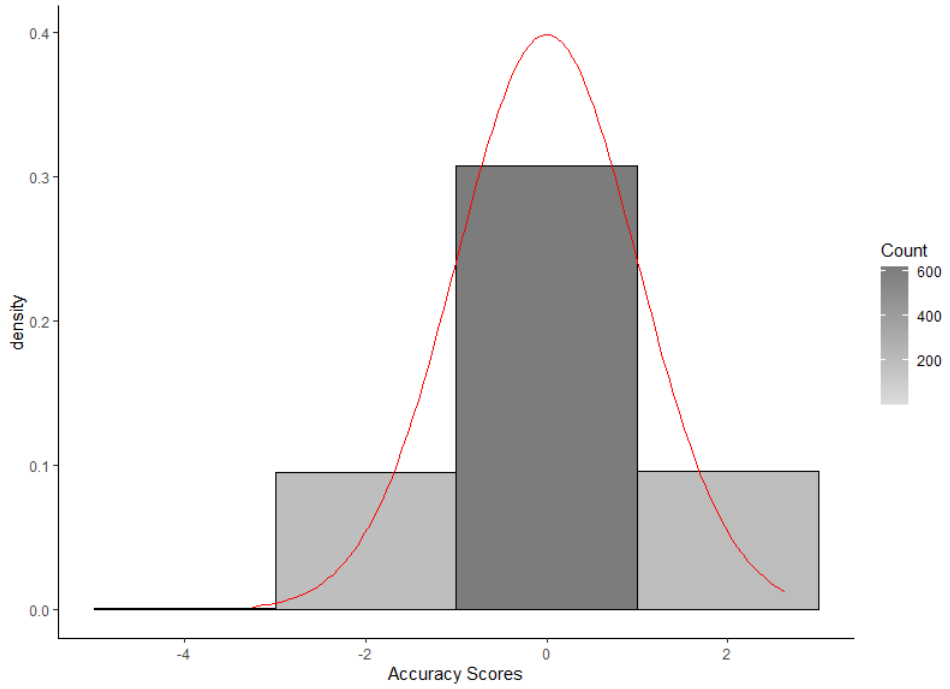

Figure A.16: Histogram of the standardised accuracy scores for the proposed model with input shape 56x50.

```
Levene's Test for Homogeneity of Variance (center = median)
        Df F value  Pr(>F)
group    1  4.0322 0.04477 *
      1998
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure A.17: Output of the Levene's test for homogeneity of variance for the model with input shape 56x50.

```
        Wilcoxon rank sum test with continuity correction

data:  Accuracy by Model
W = 415458, p-value = 5.431e-11
alternative hypothesis: true location shift is not equal to 0
```

Figure A.18: Output of the Mann-Whitney U test for the model with input shape 56x50.

A.3.6 Resnet model with input shape 70x40

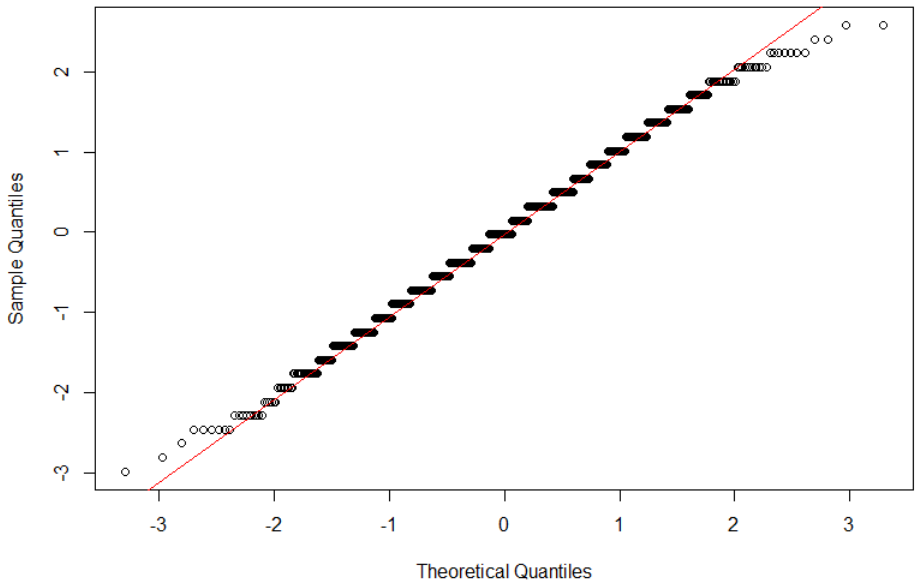

Figure A.19: QQPlot of the accuracy scores for the ResNet model with input shape 70x40.

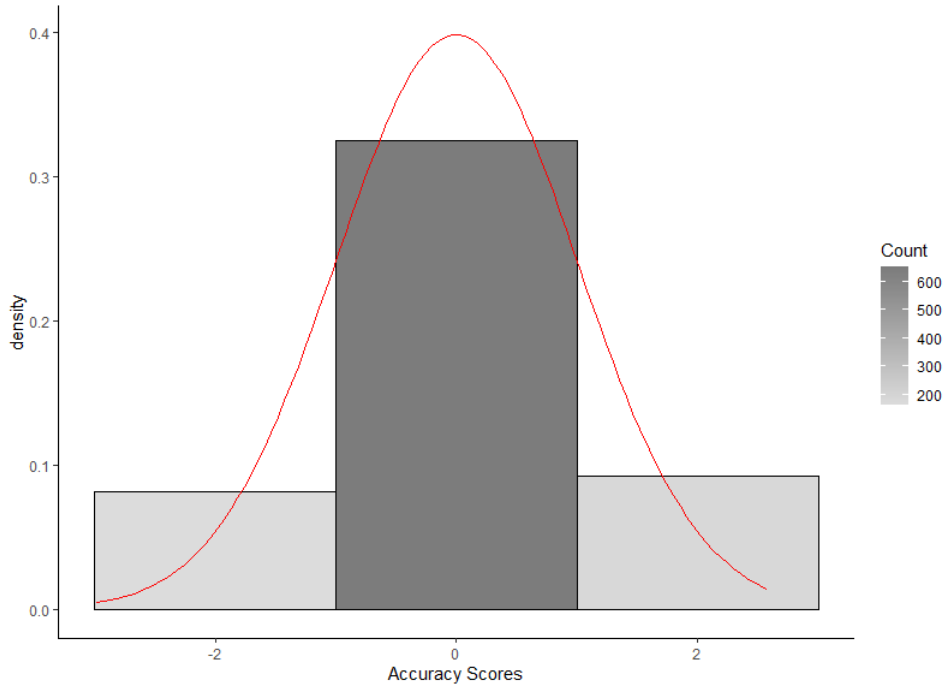

Figure A.20: Histogram of the standardised accuracy scores for the ResNet model with input shape 70x40.

```
Levene's Test for Homogeneity of Variance (center = median)
        Df F value    Pr(>F)
group    1   34.77 4.347e-09 ***
      1998
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure A.21: Output of the Levene's test for homogeneity of variance for the ResNet model with input shape 70x40.

```
        Wilcoxon rank sum test with continuity correction

data:  Accuracy by Model
W = 769840, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0
```

Figure A.22: Output of the Mann-Whitney U test for ResNet model with input shape 70x40.

A.4 The Stars

This section gives an overview of stars, their attributes and how astronomers measure stellar attributes.

A.4.1 The stellar life cycle

Stars are formed when molecular clouds undergo gravitational collapse, the size of the star depending on the amount of material within the molecular cloud. The gravitational force that causes the collapse becomes so large that nuclear fusion begins in the core of the proto-star, resulting in the core being extremely hot. As a result of the fusion process, radiation is emitted, and it is the radiation pressure from the fusion process that counteracts the gravitational collapse. From this point, the stars core is in a constant fight between gravity and radiation pressure, these two forces cancelling each other out (Greene & Lambourne, 2006).

Hydrogen is the primary source of fuel for the nuclear fusion process, but the amount of hydrogen is limited. Once this fuel source is extinguished, heavier elements begin to fuse, but this requires more and more energy. Eventually the star reaches a point where iron is the main source of fuel in the core. At this point, fusion stops, because fusion of iron requires more energy as input than it can output. The radiation pressure disappears and the force of gravity takes over, causing the stars core to collapse. This collapse can be dramatic in the case of large stars, a supernova explosion destroying the star. If the star is 11 times the mass of the Sun or more, this can result in a stellar remnant in the form of a neutron star or even a black hole, if anything survives at all. For less massive stars the explosion simply blows most of the atmosphere into space and leaves a the exposed stellar core, known as a white dwarf, that slowly grows cold and fades. The atmospheric material ejected from the star forms a planetary nebula, a cloud of dust and gas, that will eventually collapse again and form new stars. The life time of a star can be as little as 1 million years for super massive stars, but much longer for less massive stars (the estimated life time of the Sun is around 9 billion years, it being halfway through this cycle already) (Binney & Merrifield, 1998; Greene & Jones, 2004).

A.4.2 Stellar Attributes

A prerequisite for any classification scheme is an understanding of measurable attributes of the objects to be classified. This section presents an overview of stellar attributes. Some of these attributes are intrinsic to the star being measured e.g. brightness, but some are relative e.g. distance.

Colour

Looking up at the night sky, it is not always obvious that stars shine in different colours. This is partially because of light pollution, but also because stars appear in the sky as points of light that are difficult to compare. A simple photograph of the night sky will reveal the intrinsic colours of stars. The colour of a star is related to its surface temperature, cooler stars appear red while hot stars appear white/blue (Greene & Lambourne, 2006).

Distance

The distance to the stars from Earth are almost incomprehensible when measured in standard units. For example, the average distance between the Earth and the Sun, our nearest star, is $1.5\text{x}10^{11}$m. To make these numbers user friendly, astronomers have come up with their own units to represent the vast distances to, and between, objects in space (Greene & Jones, 2004).

The average distance between the Earth and the Sun is used as a measurement unit, and is known as a Astronomical Unit (AU). A light year (ly) is the distance travelled by electromagnetic radiation in a vacuum in one year. This is $9.46\text{x}10^{15}$m. Another common unit of measurement is the parsec (pc). A parsec is approximately 3.25 light years, and is the distance if an angle of 1 arc-second is projected onto a width of 1AU, as shown in Figure A.23. The AU, light year and the parsec are not standard units (SI units), but these are used by astronomers all the time to keep distance values manageable (Greene & Lambourne, 2006).

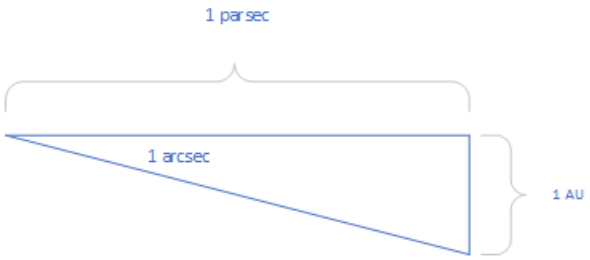

Figure A.23: The definition of a parsec.

Size

Probably the most basic attribute of a star is its size. This is important to understand as the life cycle of a star is directly influenced by this attribute, smaller stars fade away whereas large stars end their lives spectacularly in massive explosions known as novas. When thinking about the size of a star, two important attributes need to be considered, its mass and radius. Counter-intuitively, these two attributes are not always correlated. Red Giant stars are an example of a star that has reached the point in its life cycle where its surface undergoes massive expansion, increasing its radius, to maintain the nuclear fusion process, but its mass remains constant (Greene & Lambourne, 2006).

Motion

Despite their appearance, all stars in the sky are moving relative to each other. The observed motion of a star without the effects of the movement of the observer is known as proper motion. In reality, all observers are in motion relative to the stars they observe. The measured motion of a star is known as its space velocity. The space velocity vector is a combination of the star's radial velocity (movement along the line of sight) and transverse velocity (change in velocity perpendicular to the line of sight). The transverse velocity is inversely proportional to the distance, so this measurement is useful when determining distances to stars (Greene & Jones, 2004).

Brightness & Temperature

All stars have an intrinsic brightness, also known as luminosity, that is relative to their surface temperature. Surface temperature is also related to the colour of the star. When referring to brightness, the convention is to refer to the region known as the photosphere, the area of the star that corresponds to the region that we see visually. On the Sun, this is a region approximately 500 Km deep, where temperatures vary from 9000K at the bottom to 4500K at the top. Most of the light is emitted from a region where the temperature is approximately 5600K, and this is used by convention to refer to the surface temperature (Greene & Jones, 2004). This is used with caution as there is no real definition of a surface for a star, there being no solid component that can be referenced as a surface.

The apparent brightness of a star is not uniform across its surface, the brightness reducing as a function of distance from its equator. This is known as limb darkening and is related to the observers view of the stellar surface rather than an actual phenomenon on the surface. Magnetic activity can also affect the perceived brightness of a star, sunspots and flairs are examples of this (Greene & Jones, 2004).

Stars in the sky are very faint, making the measurement of brightness difficult. To help resolve this problem, astronomers use a technique called Standard Candles. Standard candles are objects with a known brightness. When the brightness of an object is known, it can be used to determine the distance to an object by comparing the actual brightness and the measured brightness. Certain types of variable stars are Standard Candles. Cepheid variables were used by Edwin Hubble in 1922 to determine the distance to the Andromeda Galaxy. Up to that point, the Andromeda Galaxy was classified as a nebula within the Milky Way Galaxy. After Hubble's discovery, it was clear that the universe was made up of multiple galaxies. Type 1a supernovae are also standard candles. As these violent explosions are so massive, they are also very bright, meaning they can be used to estimate distances for objects that are at great distances. Type 1a supernova are commonly used to estimate distances to far away galaxies. Studying distant type 1a supernova is the main purpose of the James Web telescope, due to be launched in October 2021 (Binney & Merrifield, 1998; Greene &

Lambourne, 2006).

Chemical composition

The chemical composition of a star depends on when the star was formed, and the contents of the stellar medium from which it formed. Stars are categorized based on the percentage content of heavy elements in the stellar constituents, this is known as the stellar metallicity. The term 'heavy elements' refers to any element heavier than Helium. Depending on the measured metallicity, stars are assigned to distinct Populations; Population I represents young to moderately ages stars, these contain a metallicity value of up to 3%. Population II stars are older stars and have a metallicity value of less than 1%. Theoretically Population III stars can exist, these being stars made from the original primordial material leftover after the big bang (Karttunen et al., 2006). There is no observational evidence for the existence of Population III stars but this is still an active area of research (Chandra & Schlaufman, 2021; Placco et al., 2021).

Magnetic field

Stars are made of rotating plasma which generates a vast magnetic field, stretching many astronomical units. Stars do not have a solid surface, meaning that areas closer to the equator rotate faster than the polar regions. This is known as differential rotation and is the result of centrifugal forces generated by the rotation of the star. As a result, magnetic field lines become twisted and stretched, leading to very interesting phenomena such as magnetic field re-connection and solar flares (Greene & Jones, 2004). To understand the true size of the Sun's magnetic field, after 36 years travel, the Voyager 1 space craft, on 25th August 2012, entered interstellar space, a region of space outside the influence of the Sun, including its magnetic field (Izmodenov & Alexashov, 2020). The Voyager 1 space craft had travelled a distance of 122 AU to reach this point.

A.4.3 Measuring Stellar Attributes

This section details the measurable attributes of a star, how they are measured and the units of each measurement.

Right Ascension and Declination

Stars are mapped onto the celestial sphere, an imaginary sphere used as a projection surface. The Earths equator is projected onto the celestial sphere to create a plane known as the celestial equator. From here it is a simple exercise to create a coordinate system, containing celestial north and south poles. Any line passing from the celestial north pole, through the equator to the celestial south pole is known as a meridian. The angular separation of a star from the celestial equator is known as the declination and is denoted by the symbol δ. Declination is analogous to geographical latitude. The analogue of longitude in celestial terms is known as right ascension (R.A.). Right ascension is measured as the angular distance, counterclockwise, from a meridian from both poles passing through a point on the celestial equator known as the vernal equinox, which is in the constellation of Aries. Using declination and right ascension, the positions of stars in the sky can be accurately mapped. As these measurements are both angular separations, they are measured in terms of degrees, hours, minutes, and seconds (Greene & Jones, 2004).

Distance

Different techniques are used to measure the distance to stars, depending on how far away they are. The simplest technique is known as stellar parallax. This technique uses the Earth's own orbit as a basis to perform different observations, 6 months apart. Due to the movement of the Earth along its orbit, the position of nearby stars appear to move when compared to the stars in the background. The apparent shift of position of the star is calculated using the small angle formula as shown in Equation A.1:

$$D = \frac{1}{p} \tag{A.1}$$

where D is the distance and p is the angular displacement measured in arc seconds (Greene & Jones, 2004).

For objects that are further away, standard candles can be used to determine their distance. This distance is calculated using the known luminosity, expressed as magnitude, and the measured luminosity, using the Equation A.2 where M is the stars absolute magnitude, m the apparent magnitude and d is the distance measured in parsecs (Greene & Jones, 2004).:

$$M = m - 5logd + 5 \tag{A.2}$$

The standard candle technique is especially useful as there are several different standard candles, including certain types of super-nova, and these can be seen across great distances. Equation A.3 shows how the distance, d, to an object with known luminosity is calculated, where L is the luminosity and F is the spectral flux density (Greene & Jones, 2004).

$$d = \sqrt{\frac{L}{4\pi F}} \tag{A.3}$$

Combining accurate measurements of distance with right ascension and declination has produced amazing three dimensional maps of the local galaxy. For example, the GAIA space craft has accurately measured the positions of the closest one million stars (Gilmore et al., 2012) and the information is publicly available as an app [1].

Radius

Computing the radius of a star depends on the proximity of the star to the observer. If the star is close enough, and the distance between the star and observer is known, then it is computed using the small angle formula where d is the distance to the star and α is the angular diameter of the star, as shown in Equation A.4 (Greene & Jones, 2004).

[1] https://zah.uni-heidelberg.de/gaia/outreach/gaiasky

$$R = \left(\frac{\alpha}{2}\right) * d \tag{A.4}$$

Computing the angular diameter is simple in theory, but not so simple in practice. Very large telescopes are required to capture images at the necessary resolution. Ground based telescopes suffer from effects of atmospheric distortion, meaning images may not be crisp enough to determine the angular diameter. Astronomers have developed a system known as active optics to help work around atmospheric distortion. Active optics is a system that uses a targeting laser and computers to deform the primary mirror in a telescope to reverse the effects of atmospheric distortion (Karttunen et al., 2006).

Stellar radius values, R, are reported relative the radius of the Sun ($6.96\text{x}10^8$m), denoted by the symbol $R_\odot$, as shown in Equation A.5 (Greene & Jones, 2004)

$$R = \frac{(\alpha/2) * d}{R\odot} \tag{A.5}$$

It is not possible to use direct imaging to measure the radius of stars that are the same size of the Sun, or smaller. Astronomers have developed methods using interferometry as a way around this problem. Occultation, the transit of another body across the line of sight of a star, can also be used to measure its radius (Karttunen et al., 2006).

Luminosity

Stellar luminosity is a measure of the power a star radiates over all wavelengths and is measured in Watts. Stellar luminosities are reported relative to the luminosity of the Sun, denoted by the symbol $L_\odot$, ($3.84\text{x}10^{26}$W). The luminosity of a star is estimated using Equation A.6 where L is the estimated luminosity, R is the radius, σ is the Stefan-Boltzmann constant and T is the temperature (Greene & Jones, 2004).

$$L = 4\pi R^2 \sigma T^4 \tag{A.6}$$

The problem with Equation A.6 is that the radius of the object is required and in

practice this is very difficult to measure. A more robust method to estimate luminosity based on spectral flux density is shown in Equation A.7 where F is the spectral flux density and d is the distance to the object (Greene & Jones, 2004).

$$L = \frac{4\pi d^2}{F} \tag{A.7}$$

Equation A.7 removes the need to know the stellar radius, only the distance between the observer and object is required (Greene & Lambourne, 2006).

Light curve

The light curve of a star is the plot of the luminosity of a star over time. The light curve is a useful tool to detect changes in luminosity. Variable stars are detected this way, the first Cepheid variable being detected in 1784. Variable stars are often used as standard candles (Greene & Jones, 2004).

Mass

The mass of a star is a measure of how much material is contained within the star and is measured in Kilograms (kg). When dealing with mass on a stellar scale, the numbers become very large. For example the mass of the Sun, denoted by the symbol $M_{\odot}$, is 1.99x10^{30} kg. For convenience, astronomers report stellar masses relative to the mass of the Sun as shown in Equation A.8 where Ms is the mass of the star, $M_{\odot}$ is the mass of the Sun and M is reported relative mass (Greene & Jones, 2004).

$$M = \frac{Ms}{M\odot} \tag{A.8}$$

Mass can be measured by direct observations of binary systems by measuring the period of the rotation of a system, the sum of the masses can be determined using Equation A.9 where M and m represent the masses of the object in the binary system, r is the radius of the distance between the two objects, G is the Gravitational Constant and P is the orbital period (Greene & Jones, 2004).

$$M + m = \frac{4^2 r^3}{GP^2} \tag{A.9}$$

Temperature

Objects that emit continuous spectra are known as thermal sources. Thermal sources have a property that produce a specific type of continuous spectrum known as a black-body spectrum (also sometimes referred to as a Planck Spectrum). Black-body spectra produce a very distinctive thermal light curve, appearing as a smooth line with an obvious peak at a specific wavelength. This peak wavelength is directly proportional to the temperature of the object. The peak wavelength and temperature are related using Wien's Displacement Law, shown in Equation A.10 where λ is the peak wavelength and T is the temperature in Kelvin (Greene & Jones, 2004).

$$T = \frac{2.90x10^{-3}}{\lambda} \tag{A.10}$$

Continuous spectra from stars are not perfect black-body spectra, but they are a very good approximation. Using Wien's Displacement Law shown in Equation A.10, the surface temperature can be approximated with very good accuracy from spectra. However, if the luminosity and radius are known, then the temperature can be estimated directly as shown in Equation A.11, which is a rearrangement of Equation A.6

$$T = \left(\frac{L}{4\pi R^2 \sigma} \right)^{1/4} \tag{A.11}$$

The effective surface temperature is denoted with the abbreviation T_{eff} (Binney & Merrifield, 1998; Greene & Jones, 2004).

Surface gravity

The force of gravity at the surface of a star depends on the mass of the star and its radius. As already outlined in section A.4.2, these values are not fixed, and as a result of this two stars with the same mass can have different surface gravity values.

Equation A.12 calculates the surface gravity for a star where g is the force of gravity, G is the Gravitational Constant, M is the stellar mass and R is the stellar radius (Greene & Jones, 2004).

$$g = \frac{GM}{R^2} \tag{A.12}$$

The surface gravity exerts pressure on the stellar plasma, and thus influences the density of this material, and the densities at which the spectral lines are formed. This can be directly measured from the spectrum. Surface gravity is reported in a log scale, typically reported in units of cms^{-1} and denoted with the abbreviation *log g* (Binney & Merrifield, 1998).

Metallicity

The chemical composition of a star is measured by comparing the abundance of Hydrogen (H) to any other heavy element in that star. Heavy element in this context means an element heavier than Helium (He). Stars rich in iron (Fe) also tend to be rich in other heavy elements, therefore the metallicity of a star is typically reported as the abundance of Hydrogen to Iron using the abbreviation Fe/H. It can be measured directly from the stellar spectra and is reported relative to the metallicity value of the Sun, as shown in Equation A.13. As this is a ratio, it does not have any units (Binney & Merrifield, 1998).

$$[Fe/H] = log_{10}\left(\frac{n(Fe)}{n(H)}\right)_{\text{star}} - log_{10}\left(\frac{n(Fe)}{n(H)}\right)_{\text{sun}} \tag{A.13}$$

Motion

The space velocity of a star is a vector that is broken into two components, its radial velocity and its transverse velocity. The radial velocity is measured directly from the spectrum. This is commonly referred to as the red shift if the star is moving away from the observer, or blue shift if the star is moving towards the observer. The word 'shift' gives a clue to how this is actually measured, spectral lines for known elements

appear to be shifted to either the blue or red parts of the spectrum. This phenomena is known as the Doppler effect (Greene & Lambourne, 2006). The magnitude of this shift is proportional to the radial velocity of the star. Spectral shifting can also be influenced by the expansion of space (dark energy) but this is not a concern for objects within a galaxy. The radial velocity is calculated using Equation A.14 where v_r is the radial velocity, $\lambda^{'}$ is the measured wavelength, c is the speed of light in a vacuum and λ is the actual wavelength (Greene & Jones, 2004).

$$v_{\mathrm{t}} = c\left(\frac{\lambda^{'} - \lambda}{\lambda}\right) \tag{A.14}$$

To measure the transverse velocity, the angular distance the star has travelled is used in Equation A.15

$$v_{\mathrm{r}} = d\mu \tag{A.15}$$

where d is the distance and μ is the angular displacement. This is the small angle formula in use again. Velocities are reported in standard SI units of Kms^{-1} (Binney & Merrifield, 1998).