UNIVERSITY OF CRETE

DEPARTMENT OF PHYSICS

MSc THESIS

Spectral classification of stars based on Machine Learning methods

Elias Kyritsis

Advisor(s): *Prof.* Andreas Zezas - University of Crete
Dr. Grigoris Maravelias - National Observatory of Athens

UNIVERSITY OF CRETE
DEPARTMENT OF PHYSICS

M S C T H E S I S

to obtain the title of

MASTER OF SCIENCE
OF
- ADVANCED PHYSICS -

Defended by
Elias KYRITSIS

Spectral classification of stars based on Machine Learning methods

COMMITTEE

Prof.	Andreas Zezas	-	UNIVERSITY OF CRERE
Prof.	Vasiliki Pavlidou	-	UNIVERSITY OF CRETE
Dr	Pablo Reig	-	UNIVERSITY OF CRETE

Date of the defense:
27/09/2019

Περίληψη

Τα Διπλά Συστήματα Εκπομπής Ακτίνων Χ Μεγάλης Μαζας (ΔΣΕΑΧΜΜ) αποτελούνται από ένα συμπαγές αντικείμενο(Αστέρα Νετρονίων ή Μαύρη Τρύπα) και ένα μαζικό συνοδό αστέρα φασματικού τύπου O- B-. Η γνώση των φασματικών τύπων αυτών των αστεριών είναι σημαντική επειδή μας παρέχει ένα πλούτο πληροφοριών για των σχηματισμό και την εξέλιξη των ΔΣΕΑΧΜΜ. Τα προηγούμενα χρόνια, μεγάλες έρευνες ήταν αφιερωμένες στην φασματική ταξινόμηση του συνοδού αστέρα αυτών των συστημάτων είτε στον στον Γαλαξία μας είτε στα Μαγγελανικά Νέφη. Βασιζόμενες κυρίως στον παραδοσιακό τρόπο φασματικής ταξινόμησης , εξέτασαν το φάσμα αυτών των αστεριών με οπτική επιθεώρηση και τα ταξινόμησαν συμφωνα με την παρουσία ή την απουσία χαρακτηριστικών φασματικών γραμμών. Στις μέρες μας, όπου ο αριθμός φασματοσκοπικών δεδομένων αυξάνεται συνεχώς αυτός ο τρόπος φασματικής ταξινόμησης είναι χρονοβόρος και εμπεριέχει υποκειμενικότητα. Έτσι, η ανάγκη μίας νέας αντικειμενικής αυτόματης μεθόδου για τον προσδιορισμό των φασματικών τύπων αυτών των αστεριών είναι πιο επίκαιρη από ποτέ. Σε αυτή την εργασία, χρησιμοποιούμε τον δημοφιλή επιβλεπόμενο αλγόριθμο μηχανικής μάθησης που ονομάζεται Τυχαία Δάση, με σκοπό να αναπτύξουμε έναν αυτόματο φασματικό ταξινομητή για αστέρια προγενέστερου φασματικού τύπου. Το δείγμα μας αποτελείται απο 777 αστέρια φασματικού τύπου OB από διαφορετικές έρευνες. Στα φάσματα αυτών των αστεριών μετράμε το Ισοδύναμο Πλάτος από 18 χαρακτηριστικές φασματικές γραμμές ακολουθώντας ένα σχήμα γραμμών που αναπτύχθηκε για την ταξινόμηση αυτών των πηγών. Βελτιστοποιούμε το μοντέλο μας αναζητώντας τις καλύτερες τιμές των υπερπαραμέτρων καθώς και τον καλύτερο συνδιασμό φασματικών γραμμών επιτυγχάνοντας ένα μοντέλο με ακρίβεια πρόβλεψης $\sim$ **70 %**. Τέλος, εφαρμόζουμε το μοντέλο μας σε ένα δείγμα από 28 πηγές οι οποίες βρίσκονται και στον Γαλαξία μας και στο Μικρό Νέφος του Μαγγελάνου και προηγουμένως έχουν ταξινομηθεί με οπτική επιθεώρηση , επιτυγχάνοντας ένα σκορ $\sim$ 60 %.

Abstract

High Mass X-Ray Binaries (HMXBs) are systems that consist of a compact object (Neutron Star or Black Hole) and a massive companion star with O- B-spectral type.The knowledge of the spectral types of these stars is crucial because it can provide us a wealth of information about the formation and evolution of HMXBs systems. Previous years, big surveys were dedicated in the spectral classification of the companions star in these systems either in the Galaxy or in the Magellanic Clouds. Spectral classification was performed through visual examination of their spectra ,according to the presence or the absence of characteristic spectral lines. Nowadays, where the number of spectroscopic data is continuously this approach is time consuming and suffers from subjectivity. Thus, the need of an new objective automated method is more timely than ever. In this work, we use the popular supervised machine learning algorithm Random Forest to develop an automated spectral classifier for early type stars. In our sample are included 777 OB stars from different surveys. We measure the Equivalent Width of 18 characteristic spectral lines (features) following a scheme developed for the classification of these sources. We optimized our model by searching for the best values of the hyperparameters as well as the best combination of spectral lines. We reached a prediction accuracy $\sim$ **70 %** using **14** out of the initial **18** lines in our scheme. Finally , we apply our model in a sample of 28 sources both from the Galaxy and the Small Magellanic Cloud, with known spectral types from visual inspection, reaching a success rate of $\sim 60\%$.

Acknowledgements

I would like to express my sincere gratitude to my supervisors Prof. Andrea Zeza and Dr. Grigori Maravelia, for their trust in my capabilities,their guidance and advices in every matter related to this work,but also in astronomy and astrophysics in general. I would like to acknwoledge also Dr. Paolo Bonfini for the very helpful discussions and advice offered to make this work better. In addition, I am very grateful to my family and my friends Myrto, Niko , George, Greg, Eva and Michael for their enthusiastic encouragement not only in the good times but especially in the difficult times of this work. Finally, I would like to express my sincere thanks to my partner, Pandora, for her patience and support throughout this journey.

Contents

List of Figures

List of Figures

List of Tables

1

Introduction

Definition and Classification of X-ray Binaries

An X-ray binary system (XRB) is composed of a compact object and a companion star. The compact object orbits around the companion star and accretes matter from it. Because of the enormous gravitational field of the compact object the matter from the donor star is accelerated to extremely high velocities. In this process the potential energy of the infalling matter is converted to the kinetic energy and eventually to heat (10^7 K to 10^8 K) radiation which lies in the region of X-rays in the electromagnetic spectrum. Thus, on these systems the compact object dominates on X-rays and the companion dominates on the optical domain (see the review by Reig 2011). Given the nature of the compact object we may have Black-Hole (BH), Neutron Star (NS) and White Dwarf (DW) systems. Depending on the mass of the companion star XRBs can be Low Mass X-Ray Binaries (LMXBs) with mass $M < 1M_{\odot}$ or High Mass X-Ray Binaries (HMXBs) with mass $M > 10M_{\odot}$. In our galaxy 300 high energy binary systems are known: 187 LMXBs and 114 HMXBs (respectively 62% and 38% of the total) (Liu et al., 2006).

High-Mass X-ray Binaries

Definition and Classification of HMXBs

A High-Mass X-Ray Binary (HMXB) system consists of a compact object and a massive companion star with a typical mass of $> 10M_{\odot}$. The companion star in these systems belongs to an early spectral type OB. Depending on the way the matter is accreted on the compact object the HMXBs are divided in two subcategories: Be/X-Ray Binaries(BeXRBs) and supergiant X-Ray Binaries (sgXRBs). In the sgXRBs the accretion is due to strong winds while in BeXRBs the accretion is due to a circumstellar disk that is formed because of the star's high (close to the critical limit) rotational velocity.

Formation and Evolution of HMXBs

In Figure 1.1 there is a schematic representation of the formation of a HMXB. Initially, we have a binary system with stars at Zero Age Main Sequence-ZAMS) with masses over $8M_{\odot}$. The primary star which is the most massive will evolve fast and in a few Myrs will fill its Roche-lobe, transfer most of its mass to its companion and finally will explode as a supernova. Under the condition that the binary system will survive from the explosion, the stellar remnant will remain in a wider and more eccentric orbit than the initial one, because of the gravitational potential of the system. After that the roles reverse and the secondary star in now the most massive. Depending on the nature of this star the accretion of matter to compact object can be either through stellar wind or an equatorial disk. At this point the system enters in HMXB stage. Depending on the available material and the geometry of the orbit of the systems, their X-Ray emission can be either persistent or variable in timescales of days up to several months. Their tyical luminosity ranges are between $\sim 10^{34}$ (for low-activity systems) up to $10^{38} ergs^{-1}$ (for outbursting systems). As the binary evolves the orbital period gradually becomes shorter because of the angular momentum loss. When the orbit shrinks to such a degree that the compact object is at the vicinity of the companion's envelope the binary enters the common envelope phase. This period is a very short-lived phase of a few $\sim 10^3$ yrs and the compact object spirals in and its orbital energy is deposited to the envelope. If the binary system has enough energy to eject the envelope before the compact object merges with the stellar core, the helium core of the secondary component of the binary evolves fast and leads to a second supernova explosion. The remnants of the binary are now two neutron stars and leads to potential gravitational wave system (Abbott et al., 2017).

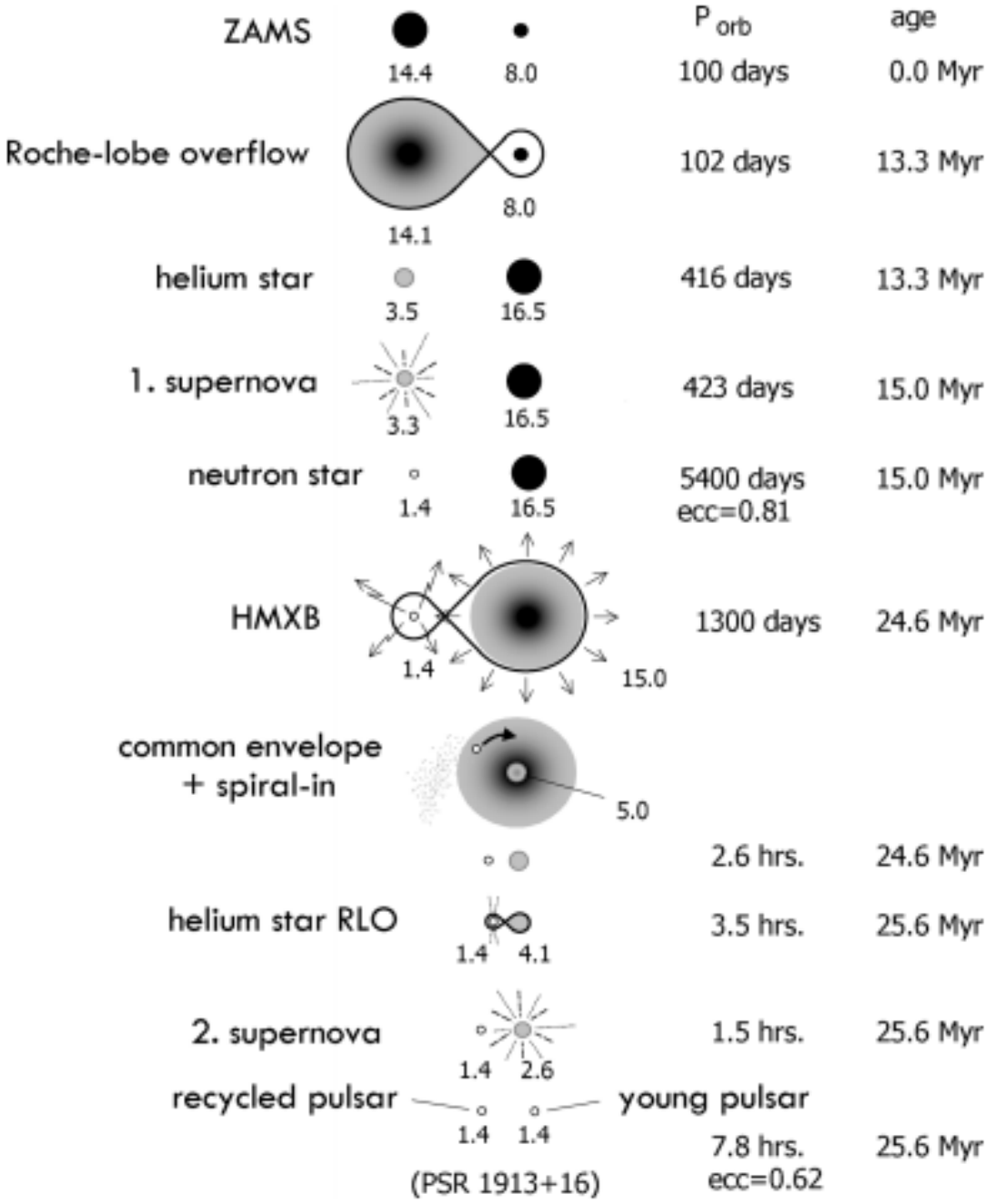

Figure 1.1: Formation and evolution of HMXBs - Schematic representation of the formation and evolution of a HMXB.The orbital period and the time evolution are also presented inline with the various evolution stages up to and after the HMXB phase.(Tauris & van den Heuvel, 2006)

Be X-ray Binaries

Be X-ray Binaries (BeXRBs) systems are a sub-class of HMXBs which consist of a a non-supergiant (with luminosity class III-V) B or late O type star and typically a neutron star.The main characteristic is that the donor's spectrum has, or had at some time, one or more Balmer lines in emission.That explains the symbol 'e'(from the word 'emission') after the spectral type. The emission is due to a decretion disk which forms as mass is lost from the stellar equator and accumulates in a geometrically thin, outward expanding disk (Porter & Rivinius, 2003). The mechanism that drives this mass loss is still unknown but it is believed that is a result of the stellar rotation (Townsend et al., 2004) and/or non-radial pulsations (Rivinius et al., 2013).

In Figure 1.2 we present a schematic representation of a BeXRB system. When the NS passes from the periastron (the closest to the star point of its orbit) it attracts matter from the disk and creates an accretion disk. This is the phase we have intense X-ray emission. Gradually, this emission decreases as the NS travells farther in its orbit and depending on the eccentricity and available material the emission can stop or remain undetected. Thus BeXRBs are transient systems and can be divided further in two subcategories depending on outbursting activity.

- *Type I outbursts*: These are regular and periodic each time the NS crosses the decretion disk at periastron. Their duration lasts a few days and their outburst luminosities are in the range of $10^{36} - 10^{37} ergs^{-1}$ Reig (2011).

- *Type II outbursts*: These correspond to giant outbursts which are aperiodic, with a dramatic expansion of the NS Chaty (2011).The flux in these cases is highly increased($10^3 - 10^4$ times the non-outbursting state) and makes these systems the brightest X-ray sources among the sky.

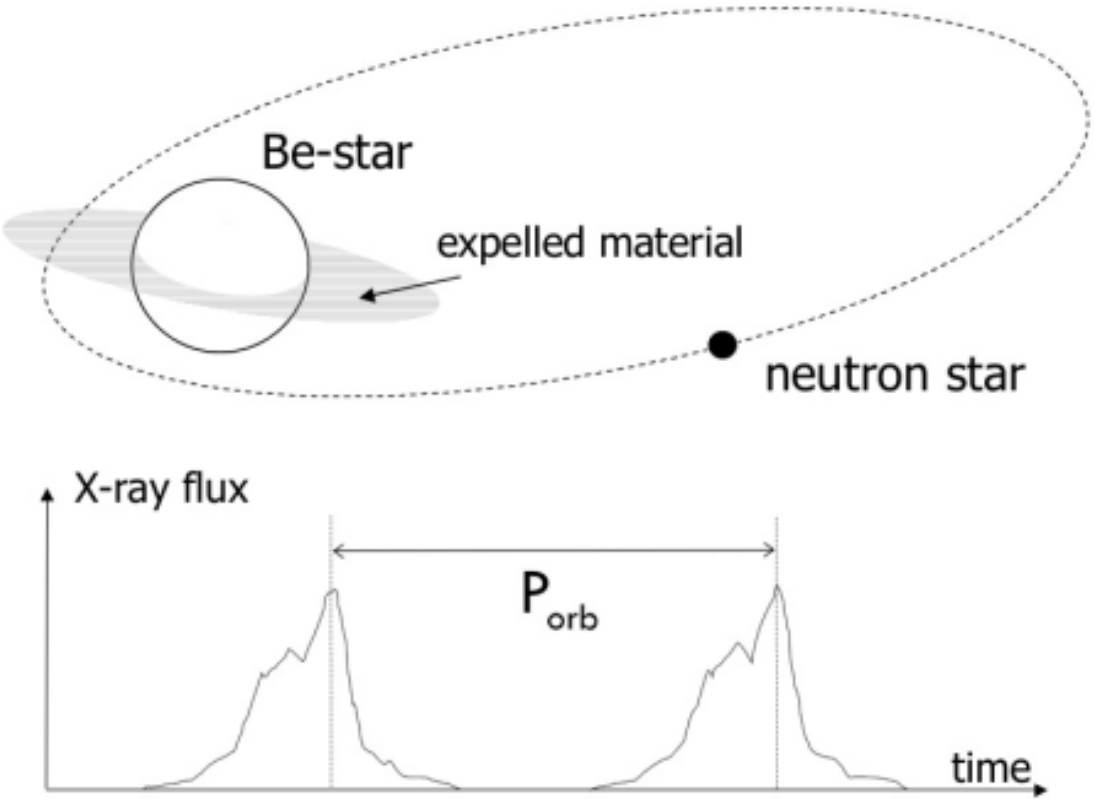

Figure 1.2: An illustration of a BeXRB system - Schematic representation of a BeXRB. As the NS passes close to the star, it accumulates material from the disk of the donor Be-star and enters an outburst event of Type I. The X-ray flux decreases (to a non-outbursting, often non-detectable phase) as the neutron star orbits away from the donor, and its flux raises again after one orbit when it approaches the periastron again (Tauris & van den Heuvel, 2006).

Optical spectroscopy as a tool for the study of OB stars

As we saw before, stars of spectral type OB are the companion stars of the HMXBs. The spectral classification of these systems is very important because it provides information regarding the physical parameters of the donor star and the evolution history of the binary. In particular, the spectral type correlates with the temperature (different temperatures give rise to different characteristic lines) as well as mass. For example, the typical spectral range of the optical counterparts is O8-B3 which corresponds approximately to 23-8 $M_{\odot}$(for Main Sequence stars; (Cox, 2000)). Given the mass we can estimate luminosities and from evolutionary models we can determine ages and formation scenarios for HMXBs systems.

In addition, statistical studies of large samples of HMXBs, with respect to their spectral-type/mass distributions allow us to investigate differences between different populations (e.g HMXBs in the Galaxy and the Maggelanic Clouds) and understand how the enviroment(e.g metallicity) can influence the evolution of these systems. Finally,the spectral distribution of these stars in different metallicity enviroments such as Maggelaninc Clouds (SMC has 1/5 $Z_{\odot}$ and LMC 1/2.5 $Z_{\odot}$) is important because there is an indication that low metallicity is asociated with higher formation efficiency of HMXBs and higher luminosity (Antoniou & Zezas, 2016; Antoniou et al., 2010). Consequently, optical spectroscopy of these stars is the absolute tool to confirm the real nature of these systems, and can offer us a way to study these populations.

The problem with optical spectral classification (not only in early type stars but in general) traditionally is done qualitative. It is based on the presence or absence of diagnostic lines and suffers from subjectivity, despite the power of human eye as a pattern recognition classifier. This is a time-consuming process and it is not easy to handle a big volume of data. Even worse, in the case of Be stars which exhibit Balmer lines in emmision we can not take advantage of these lines as diagnostic tool. The reason is that the variability of the size of the disk (in timescales of months) results in variable Balmer lines (Okazaki, 1997).In other words, when the circumstellar disk is absent the Balmer lines are in absorpion but when the circumstellar disk is fully developed the Balmer lines are in strong emission. Thus,we can not use the Balmer lines for spectral classification of Be stars and BeXRBs in particular which are the largest population of the HMXBs in general.

The need of an automated way of spectral classification

The best way to overcome the problem of the traditional way of stellar spectra classification is to use automated methods based on quantitative measurements of spectral features. Thus, the subjectivity of human factor is addressed and properly defined errors can be obtained. Driven by this idea, previous works have proposed different algorithms for spectral classification that have been based either on the evaluation of specific criteria (e.g spectral features) or on pattern recognition. In the first case, the algorithms imitates the way that a human assesses the presence of spectral lines, when visually examines a spectrum. In the latter case, the whole spectrum or parts of it is compared with a library of templates and then non-linear algorithms are used to identify the template that minimizes the distance from the observed spectrum (Navarro et al., 2012a). Nonetheless, the big challenge of all these automated methods is to classify a wide range of spectral types (O- to M-type), in which stars present many different characteristic spectral lines. But for both approaches there are certain restrictions. For the criteria-evaluation methods it is too difficult to account for all the spectral features in a such wide range of spectral types due to the fact that the indicator spectral lines change dramaticaly from the hottest stars to the coolest stars. Furthermore,the template-mathcing techniques, despite that they are more flexible, they need template and test data of similar qualities (wavelength coverage,resolution,SNR) that is not easy when we have data from different tesescopes/observational instruments or strategies. In addition, despite that stars may have the same spectral type they are actually unique and not well represented by a single template.

Machine Learning in Astronomy

The continuous increase of astronomical datasets both in size and complexity introduced the Astronomy in a new epoch of big data science (Pesenson et al., 2010). The need of a new way of data handling is a result of past, ongoing and future surveys, which produce massive datasets of all the observational techniques (e.g photometric data, spectroscopic data and time variability data) with a wealth of inforamtion to be extracted , analyzed and yet to be discovered. Big surveys such as Sloan Digital Sky Survey (SDSS,York & Adelman 2000), Gaia (Gaia Collaboration et al., 2016) and LAMOST (Zhao et al., 2012) are a few of the many different surveys that provide the community with a large volume of data and require the use of Machine Learning algorithms for the development of robust mining methods and proper statistical tools in order to explore and interpretate these datasets.

Today the field of Machine Learning can be used in order to address the above mentioned issues. More specifically, Machine Learning is the study of algorithms used by computer systems to progressively improve their performance on building statistical models of known data to predict on unknown data without being explicitly programmed. In general, Machine Learning algorithms are devided into two groups: Supervised and Unsupervised algorithms. The main difference between them is that the supervised agorithms use known and labeled data as input instead of unsupervised algorithms that try to learn the various groups of the data from the data itself. For this work we take advantage of supervised methods (for more details on unsupervised see (Baron, 2019)).

Supervised machine learning algorithms

Supervised machine learning algorithms are used to learn a relationship between a set of measurements and target variable using a set of provided examples (Baron, 2019). Once the relationship is determined and the model is defined then we are able to use this model for the prediction of the target variable of unknown data. Between these algorithms and the traditional model fitting techniques there is a main difference which is that in supervised algorithms the model is not predefined and is constructed based on the input dataset. Based on the terminology of machine learning the objects are the input dataset and each object has measured features and a target variable. In astronomy branch, the objects can be stars , galaxies, compact objects etc. and their features are measured properties as spectra or lightcurves as well as quanties as stellar mass or variability period. Supervised algorithms can be used in both tasks, regression

task (continuous target variables) and classification task (discrete target variables). For instance, there are works that using photometric measurements (continuous target variables) in order to estimate redshifts for galaxies and quasars (Ball et al., 2008; Wadadekar, 2005) as well as works that try to classify objects into stars or galaxies (discrete target variables) (Ball et al., 2006) or stars in their spectral types (Navarro et al., 2012b).

The application of supervised machine learning algorithms works into three stages. In *training stage*, that is the first stage,the model is trying to learn the parameters from a subset of the input dataset which called *training set*. The second stage is the *validation stage* where the model hyper-parameters are improved based on some predefined cost function and often a different subset of the input dataset is used that called *validation set*. Finally, the last stage is the *testing stage* where the trained model is used for the prediction of the target variable of a different subset of the input dataset that called *testing set* and thus we have an evaluation of the algorithm's performance. The most popular supervised algorithms are **Support Vector Machines (SVM)**, **Artificial Neural Networks** and **Random Forests** and all of them used in wide range in astronomical problems in the past years. Below, we present a brief summary of the first two algorithms and in Section 2.1 we describe in detail the Random Forest algorithm which was used in this work because it is similar to the human recognition.

Support Vector Machine algorithm

Support Vector Machine algorithm is used in a wide range of astronomical projects either for classification tasks or regression tasks (Hartley et al., 2017; Hui et al., 2018; Krakowski et al., 2016). Originally proposed by (Vapnik, 1979), SVM algorithm finds a hyper-plane in the N-dimensional space that best seperates the given classes given a dataset with N- features. In a two-dimensional feature space, the hyperplane is a line that divides the plane into two parts , where each class lies on a different side. The optimal hyperplane in this case is the plane that has the maximum margin. In other words, is the maximum distance between the plane and the data points that are called support vectors. The new data (objects) are classified according to their location with respect to the hypeplane.

Often, the classes in a dataset are not linearly seperable. In such cases, the classification problem can be solved with the SVM *kernel trick*. Instead, of constructing the desicion boundary in the input dataset , the dataset is mapped into a transformed feature spach of higher dimension and thus the linear separetion might be possible. Afterwards, the desicion boundary that previously was defined it is back-projected to

the original input space,resulting in a non-linear boundary. In order to apply the *kernel trick* the *kernel function* is needed. The *kernel function* is related to the non-linear feature mapping and the most popular being *Gaussian Radial Basis Function (RBF)*, *Polynomial* and *Sigmoid*.

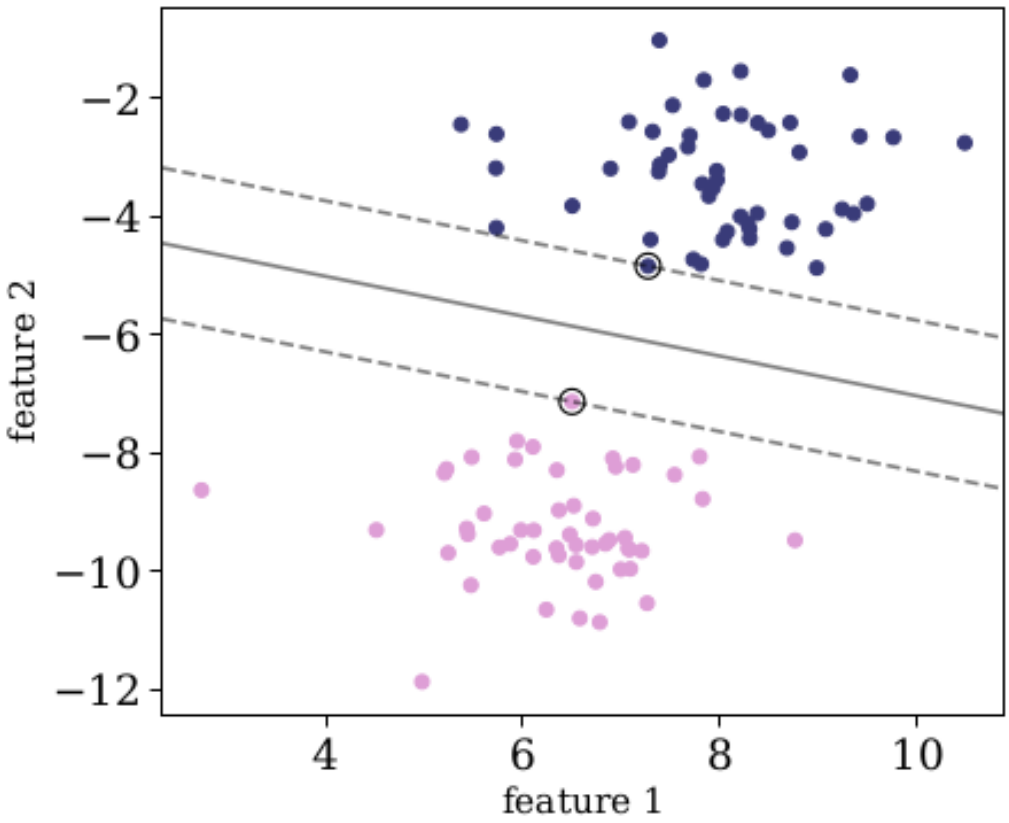

Figure 1.3: An illustration of SVM algorithm in a two-dimensional feature space. - The figure shows an illustration of the SVM best hyperplane for a two dimensional feature space with linearly seperable classes. Pink and purple circles are the two classes and the black circles are the support vectors. The hypeplane is marked with a solid gray line(Baron, 2019)

Artificial Neural Networks

As the SVM algorithm the Artificial Neural Networks are a set of supervised algorithms that used in a wide variaty of astronomical problems (Bilicki et al., 2018; Huertas-Company et al., 2018; Mahabal et al., 2017; Naul et al., 2018). The way that they work is inspired by the way that the human brain works and their flexible structure as well as thei non-linearity allows one to use them for different tasks including regression and classification.

A neural network consists of an *input layer*, an *output layer* and a lot *hidden layers*. Each of these layers contain neurons that transmit information to the neurons in the succeding layer. The values of every neuron in the network (except of the neurons in the input layer) is a linear combination of the neurons in the previous layers. In particular the value of the neurons in the first hidden layer are given by $\vec{x}_1 = f_1(W_1\vec{x}_0)$, where $\vec{x}_0$ is a vector that describes the values of the neurons in the input layer, W_1 is a weight matrix that describes the linear combination of the input values, and f_1 is a non-linear activation function (e.g RELU,TANH or softmax).The values of the second hidden layer are given by $\vec{x}_2 = f_2(W_2\vec{x}_1)$. The process follows the same logic and finally the values of the neurons in the output layer are given by $\vec{x}_3 = f_3(W_3\vec{x}_3) = f_3(W_3 f_2(W_2 f_1(W_1\vec{x}_0)))$. Thus, the input data is transmitted from the input layer, through the hidden layers and reaches the output layer where the target variable is predicted. For more information see lectures by M. Huertas-Company.

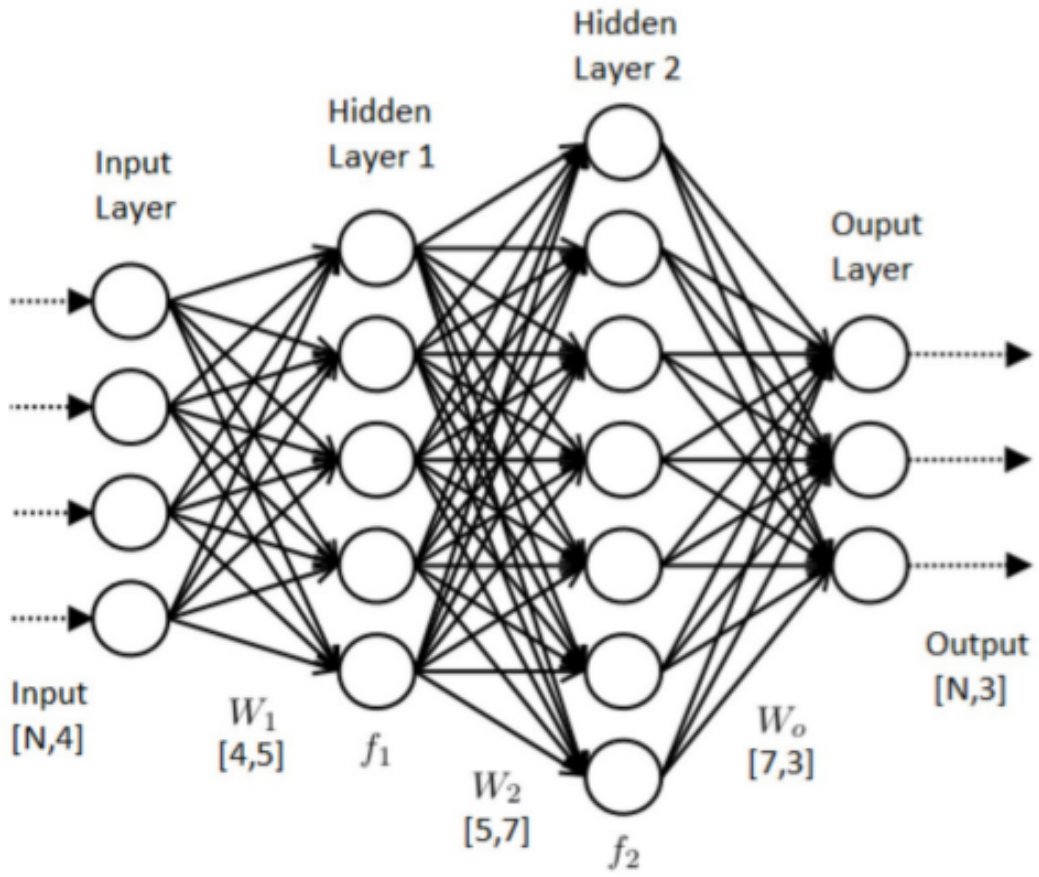

Figure 1.4: An illustration of a shallow neural network structure - The network consists of an input layer, two hidden layers, and an output layer. The input dataset is propagated from the input layer, through the hidden layers, to the output layer, where a prediction of a target variable is made. Each neuron is a linear combination of the neuron values in the previous layer, followed by an appli- cation of a non-linear activation function(Baron, 2019)

Aims of the project

The main goal of this project is the development of a new automated method of spectral classification of early type stars (OB-spectral type), the parent population of Oe/Be stars that can be found as donor stars in HMXBs systems. The knowledge of the spectral types of these stars is crucial because these massive stars can give us information about the formation and evolution of HMXBs systems. The continuously increased of the spectroscopic data can address the issues of a more quantitative spectral classification which is timely now since vast numbers of spectral produced by many surveys. Consequently, the aims of this work can be summarized as follows:

- Develop an automated spectral classifier using a popular supervised machine learning algorithm called Random Forests.
- Improve the performance of the algorithm by well established techniques.
- Apply our model to new datasets (which have been classified visually) in order to test the accuracy of our model, and compare it with the traditional way of spectral classification.

.

2 Methodology

Random Forest Algorithm

In this project we used a popular supervised learning algorithm which is the Random Forest algorithm (hereafter RF).

RF is a famous ensemble method mainly used as a supervised algorithm for classification and regression (Carliles et al., 2010; Vasconcellos et al., 2011). Ensemble methods are meta-algorithms that combine several supervised learning techniques in order to produce a single predictive model with a better performance than each individual supervised algorithm. These methods combine either different supervised learning algorithms or the information of a single algorithm that was trained on different subsets of the training set. RF is based on the production of a large number of decision trees during the training process.

Decision Trees

A decision tree is a non-parametric model constructed during the training process, which is described by a tree-like graph and is used for both,classification and regression tasks. The tree is building by following a deterministic procedure during which the training set is separated into a hierarchy of clusters of similar objects. More specifically a desicion tree is a set of sequent nodes , where each node represents a condition on one feature in the dataset. The conditions have the form $Y_j > Y_{j,th}$,where Y_j is the value of the feature at index j and $Y_{j,th}$ is the value of a threshold, both of which are determined during the training stage.
The easiest way to describe the construction of a decision tree is to consider the simple case of a classification task with two classes based on the description of (Reis & Baron, 2019). The training stage starts with the training dataset in the root node which is defined to be the fist node of the tree. The algorithm searches for the best combination of the feature Y_j and the feature threshold $Y_{j,th}$ that results in the best separation between the objects of the two classes. For the determination of the best separation in two classes the algorithm uses a model parameter with the typical choices being the *Gini impurity* and the $information\ gain$. The $information\ gain$ is defined to be the measurment of how much "information" a feature give us about the class and the algorithm always tries to maximize this value. Nevertheless, in most cases the parameter used for the best separation is the *Gini impurity*. The *Gini impurity* of a group is the probability that a randomly - selected object will be misclassified, if it is assinged with a randomly label from the distribution of the labels in the group (Reis & Baron, 2019). The *Gini impurity* in the case of a binary classification problem is :

$$G = 1 - (P_{n,A}^2 + P_{n,B}^2) \tag{2.1}$$

where the $P_{n,A}$ and $P_{n,B}$ are the fractions of objects of classes A and B within the group in the node n or the class probabilities.

The first step of the algorithm is to define the initial condition of the root node ,or in other words, to determine the feature and the value of the threshold at this node. Thus, the algorithm iterates over the full set of features and all possible thresholds. For each threshold the training dataset is devided into a $right$ and a $left$ group. Each group contains objects for which the feature values are $right$ and $left$ of the threshold respectively.The ultimate goal of the algorithm is to find the splitting threshold that results in the *minimal combined impurity* of the two groups which is defined as follows:

$$G_{right} \times f_{right} + G_{left} \times f_{left} \tag{2.2}$$

where the G_{right} and G_{left} are the Gini impurities of the two groups and f_{right}, f_{left} are the fractions of objects in each group such that:

$$f_{right} + f_{left} = 1 \tag{2.3}$$

After the choose of a specific feature and the corresponded feature value that result in the *minimal combined impurity* the initial condition for the root node is ready and the algorithm passe to the second step.
The second step is the splitting of the training dataset into two groups with respect to the value of the threshold. In other words in the $left$ node propagate objects with a value larger or equal than the threshold and in the *right* node propagate objects with a value smaller than the threshold. In each of these nodes, the algorithm searches again for the "best" threshold for the objects that propagated to it, with the same way. This process is repeated recursively, such that deeper nodes split generally smaller subsets of the original data.
The process terminates when the *combined impurity* of the two groups is not smaller anymore than the impurity of the node. The nodes that satisfy this are called *terminal nodes* or $leafs$ *nodes* and essentialy are the end of their corresponding tree branch. Each terminal nodes contain object of a single class (in our simple case objects of class A or class B). Thus the whole training process results in a tree-like structure where the intermediate nodes contain a condition (obtained from the features and the features values) and the terminal nodes contain the value of a class.

In Figure 2.1 we present an example of the structure of a decision tree taken from (Vasconcellos et al., 2011) where a desicion tree algorithm was used for classification between stars and galaxies.

When the desicion tree is trained it can be used for the prediction of a class of unseen data. The input object is propagating along the tree based on its measured features and the conditions in the nodes. Then the predicted class of the object is the label of the terminal node.

Advantages and disadvantages of the Desicion tree

The first advantage of the desicion tree algorithm is that contains a few hyperparameters and is easily itepretable. In addition, a desicion tree does not require scaling and normalization of the data which means that requires much less effort during data

Figure 2.1: An example of a desicion tree structure - The figure shows a desicion tree with nodes that represent conditions which obtained based on three features of a specific dataset in order to classify the objects between stars and galaxies. The class with label 1 represents a star and the class with label 2 represents a galaxy. The terminal nodes contain an assigned label which is a result of each particular prath within the tree (Vasconcellos et al., 2011).

preparation for pre-processing. Last but not least, a desicion tree can determine the feature importance that represents the relative importance of different features during the training process (Baron, 2019). In particular,features that are higher in the tree (closer to the root node) are more important than other that are closer to the terminal nodes.

Althought a model which is built from a desicion tree classifier can has a very high performance on the training dataset it fails to predict correct new unseen data. That happens due to the fact that a single desicion tree is prone to overfit the data. Overifitting occurs when the training of the model is based on the noise which is contained in the dataset. In other words, noise and random fluctuations in the training data are learned as concepts by the model and they are applied to new data in detail. The result of overfitting is a model with very low performance and lack of the ability to generalize to a new dataset. Furthermore, another one disadvantage of this algorithm is that a small change in the data can cause a large change in the structure of the desicion tree resulting in an increase of instability. Finaly, a desicion tree is more adequate for classification task (prediction of descrete values) than a regression task (prediction of continuous values).

From the Desicion Trees to Random Forest

The solution to the data overfitting which is the main disadvatage of desicion trees is the use of RF algorithm. RF is a collection of a large number of desicion trees.A RF classifier constructs a number of desicion trees and during the training process of these trees it uses randomly-selected data subsets of the initial full dataset. In addition, during the training process random subsets of the features are used in each node of each desicion tree in order to find the appropriate conditions in the nodes. The final prediction of the RF algorithm is in the form of a majority vote. Each individual tree in the forest suggests a prediction for a class, thus the final prediction of the RF is this class that has been proposed from the majority of the trees in the forest. Also the number of trees that suggest the predicted class is a kind of measuremet of the certainty of the final prediction (Reis & Baron, 2019). Due to the randomness in which the RF algoritmh is builded the correlation between different trees is reduced and thus the structure and the conditions in the nodes of each tree are much different from tree to tree. That results to a model which can be generalized well to new unseen data and consequently to a better performance.

Advantages and disadvantages of Random Forest

The RF algorithm has several advantages. First of all can be applied to datasets with thousand of feautures without the need of data preparation. A normalization and a scaling of the data is not required. Also it can work well in cases where there is a large proportion of missing data which is usual in astronomical data. Astronomical data strongly depended on the observational conditions and the observational intruments where the possibility of missing data is high.

On the other hand, the use of a RF algorithm means more complexity in terms of computing time. It is much harder and time-comsuming to construct more than one desicion trees. But the most important disadvatage of RF is its inability (at least in its standard form)to take into account feauture and label uncertainties, which is again very usual in astronomical observations where it is difficult to measure a quantity or classify an object with high accuracy.

Samples

The samples that have been used in this project are a collection of different spectroscopic surveys for galactic and extra-galactic early type stars. Each survey that we used had a different coverage of wavelength range but we took into account only the range that it is more useful for spectral classification purposes of OB stars ,i.e in the optical regime $\sim 3900-4900$Å(Walborn & Fitzpatrick, 1990). The reason that we used samples from two different galaxies (the Galaxy and the SMC (has 1/5 of the galactic metallicity)) is to take into account a wider range of metallicities because when we apply our algorithm we want to be consistent (SMC).

Therefore, our galactic sample consists of spectra obtained from:

1. The Galactic O-Star Catalog (GOSC) which is a massive spectroscopic syrvey based on a high signal-to-noise ratio S/N $\sim$ 250, blue-violet digital observations from both hemispheres and a spectral coverage of $\sim 3900-5100$ Å(Maíz Apellániz et al., 2013, 2011).

2. The IACOB project which is a spectroscopic survey of massive OB stars under the supervision of Canary Islands Istitute of Astrophysics (Simón-Díaz et al., 2011, 2015). The wavelength range of the observations of this project is $\sim 3700-7300$ Åwith a signal-to-noise ratio S/N $\sim$ 200 .

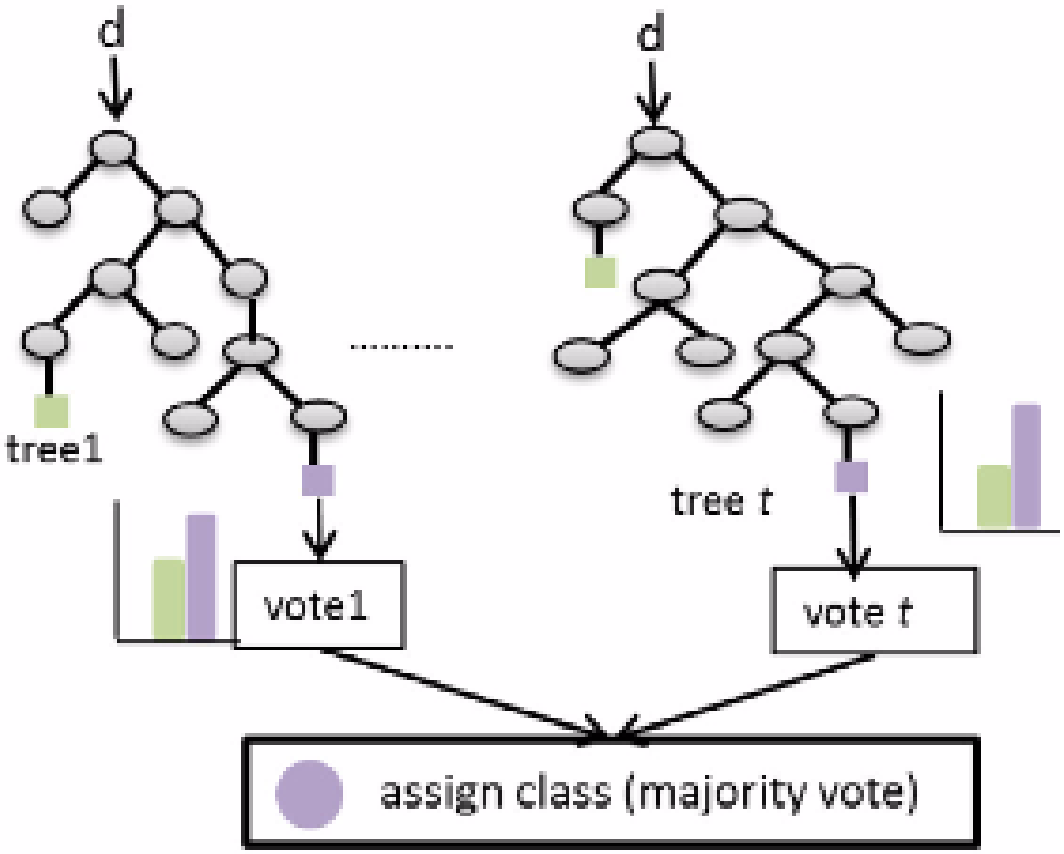

Figure 2.2: An example of a RF structure - The figure shows a random forest contains t number of trees. A d number of new data is going through each desicion tree and each tree suggests a classification class.The final assinged class is the majority of votes of each tree (Belgiu & Drăguţ, 2016)

3. The reference book by (Gray & Corbally, 2009), who discuss in detail the classification schemes for all types of stars.

Our extra-galactic sample consists of spectra obtained from:

1. The 2dF survey of the Small Magellanic Cloud: an extensive spectroscopic survey of O-,B- and A-type stars in a spectral range of $\sim 3900 - 4800$ Åand a signal-to-noise ratio S/N $\sim$ 20-150 (Evans et al., 2004).

One major limitation during the collection of the sample was the lack of a sufficiently large sample of publicly accesible spectra with available classifications as well as the need of a classical visual spectral classification instead of automated ways where the human factor is absent. At the end our final sample contains:

1. **584** Galactic sources from(Maíz Apellániz et al., 2013, 2011) in a spectral type range between the O2-O9.5 types.

2. **81** Galactic sources from (Simón-Díaz et al., 2011, 2015) in a spectral type range between the O4.5-B8 types.

3. **10** Galactic sources from the reference book of (Gray & Corbally, 2009) in a spectral range B1-B9 types.

4. **700** SMC sources from (Evans et al., 2004) in a spectral range between the O4-B9 types.

Spectral line selection

Given the Morgan-Keenan (MK) classification scheme (Gray & Corbally, 2009), we can classify a star into its spectral type based on the presence or the absence of characteristic spectral lines across the spectral type sequence as well as on their relative intensity. More specifically, the system is using the letters O, B, A, F, G, K, and M, as a sequence from the hottest (O type) to the coolest (M type). Each letter class is then subdivided using a numeric digit with 0 being hottest and 9 being coolest (e.g. A8, A9, F0, and F1 form a sequence from hotter to cooler).

In general, this system is a function of the temperature of the star where the temperature is decreasing as we move from early type stars to late type stars. The reason that it is based on the temperature of the star is that this physical parameter determines the relative strength of the spectral lines more robustly than any other physical parameter. In particular, the spectra of the hottest, O-types star are dominated by

the HeII lines, while as we move forward to B-type stars they disappear and the HeI lines start to prevail. Finally, in the spectrum of a late type star (further than A-type stars) the mettalic lines lines become apparent as we expect due to the decrease of temperature. In priciple,the most significant diagnostic tool in a spectral classification of stars is the Balmer series. In the MK system the hottest stars (O-type) have weak Hydrogen lines and they become stronger at lower temperature (A-type) stars. That happens because in high temperatures (O-type stars have surface temperatures of around 25000 K) almost all of the hydrogen is either ionized or has electrons in only very high energy levels.Thus the hydrogen lines are weak. On the other hand, in lower temperatures (A-type stars have surface temperature about 10,000 K), most of the hydrogen atoms have electrons in the second energy level and thus the hydrogen lines are stronger. However in our case we cannot take advantage of this diagnostic tool.The reason is that in BeXRBs the Balmer series exhibit strong variability due to the presence of the circumstellar disk (Porter & Rivinius, 2003).Additionally, (Reig & Zezas, 2014) showed that it is possible in some cases that the disk may affect the He I lines, although not as much as the Balmer lines.

For all the reasons that we mentioned above for our analysis we selected a scheme of characteristic spectral lines according to our classification criteria. The scheme is based on spectral lines derived from Galactic (Walborn & Fitzpatrick, 1990) and SMC (Evans et al., 2004; Maravelias et al., 2014) sources.

In Table 2.1 we present the classification criteria for B-type stars in SMC as the defined from previous works of (Maravelias et al., 2014) and (Evans et al., 2004).

Equivalent width measurments

From the scheme in Table 2.1 we selected a number of lines that help us distinguish the different classes. From these lines we acquire the Equivalent Width (EW) in order to quantify the itensity of these spectral lines.

Description of the method

EW is defined as the the width of the continuum region of a spectrum needed that contains the same flux as the spectral line examined. The formula is :

$$EW = \int_{\lambda_1}^{\lambda_2} \frac{F_{cont}(\lambda) - F_{line}(\lambda)}{F_{cont}(\lambda)} d\lambda = (\lambda_2 - \lambda_1) - \int_{\lambda_1}^{\lambda_2} \frac{F_{line}(\lambda)}{F_{cont}(\lambda)} d\lambda \qquad (2.4)$$

where λ_1,λ_2 are the initial and final wavelength over which the line flux is calculated, and F_{cont},F_{line} are the continuum and spectral-line flux density, respectively. In Figure

Table 2.1: Classification criteria for B-type stars in SMC from (Maravelias et al., 2014).

Line identifications	Spectral Type
HeII λ4200,HeII λ4541,HeII λ4686 present	earlier than B0
HeII λ4541 and HeII λ4686 present, HeII λ4200 weak	B0
HeII λ4200 and HeII λ4541 absent, HeII λ4686 weak	B0.5
HeII λ4686 absent, SiIV $\lambda\lambda$4088,4116 present	B1
SiIV λ4116 absent, SiIII λ4553 appear	B1.5
OII+CIII λ4640-4650 blend decreases rapidly	later than B1.5
SiIV and SiII absent, MgII λ4481 $<$ SiIII λ4553	B2
MgII λ4481 $\sim$ SiIII λ4553	B2.5
MgII λ4481 $>$ SiIII λ4553	B3
OII+CIII λ4640-4650 blend disappears,	later than B3
OII λ4415-4417, NII λ4631 disappear	
clear presence of HeI λ4471 and absence of MgII λ4481	earlier than B5
SiIII λ4553 absent, SiII λ4128 $-$ 4132 $<$ HeI λ4121,	B5
HeI λ4121 $<$ SiII λ4128 $-$ 4132 $<$ HeI λ4144,	B8
MgII λ4481 $\leq$ HeI λ4471	
HeI λ4471 $<$ MgII λ4481,	B9
FeII λ4233 $<$ SiII λ4128-4132	

2.3 is shown an illustrtion[1] of EW definition .

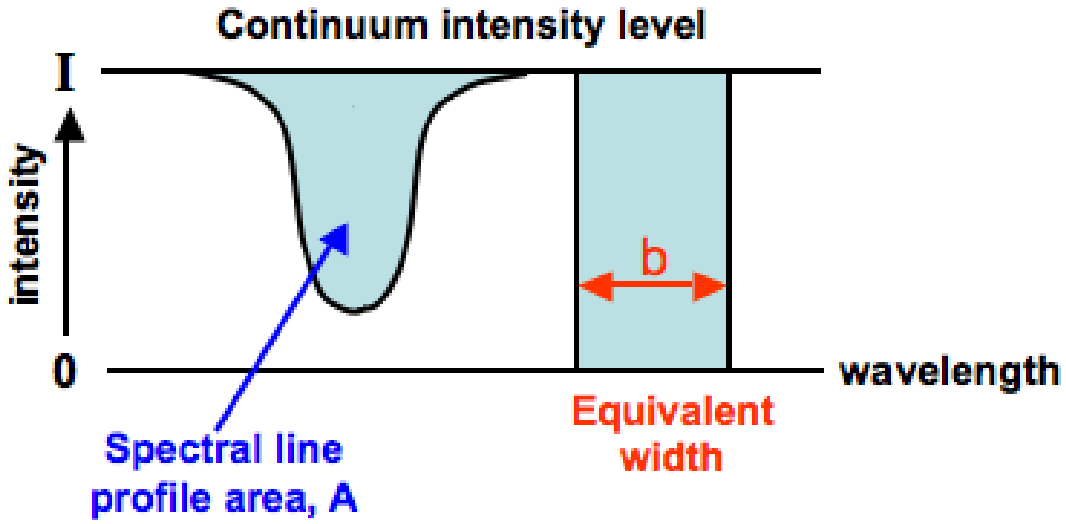

Figure 2.3: An illustration of EW definition - The area, A, of a spectral line measured below the continuum level is related to a rectangular line profile with the same area, and equivalent width, b.

However, in our case the spectral density is measured in individual pixels and their relation with the wavelength is given by the dispersion of the spectrum (d). Thus we can transform the previous formula as:

$$EW = d \times N - \sum_{i=1}^{N} \frac{F_{line_i}}{F_{cont_i}} d = d \times N - d \times \frac{1}{C} \sum_{i=1}^{N} F_{line_i} \tag{2.5}$$

where the flux of the continuum is considered constant (C) over the wavelength range of N pixels of the line. Thus, we can measure the total flux of the spectral line as the sum of the pixel values included within the λ_1-λ_2 region. To estimate the continuum flux density we use the value of the central wavelength of the line obtained from linear interpolation of the continuum intenisity from regions at the blue and red sides of the line:

$$C = C_{blue} + \frac{C_{red} - C_{blue}}{\lambda_{red} - \lambda_{blue}} (\lambda_{line} - \lambda_{blue}) \tag{2.6}$$

where C_{blue} and C_{red} are the average values for the continuum flux density at the blue and red sides (in Å/px), and λ_{blue},λ_{red} and λ_{line} are the central wavelengths of the blue and the red continuum, and the line regions respectively. Therefore, after visual inspection from the appropriate range of the wavelengths for the spectral lines and their

[1] `http://astronomy.swin.edu.au/cosmos/E/Equivalent+Width`

continuum regions were defined making sure that they did not include other lines or artifacts (Maravelias, 2014). In Table 2.2 we present the full scheme of the spectral lines with their central wavelengths and their corresponding spectral ranges are presented in Table 2.3. After that we can calculate the mean continuum intensity at the center of each continuum side and based on the equation 2.6 we calculate the continuum flux density at the center of each line (see Figure2.4). The final step is to use this value to equation 2.5 in order to measure the EW of each spectral line. All the procedure is being automatically based on an algorithm developed by (Maravelias, 2014).

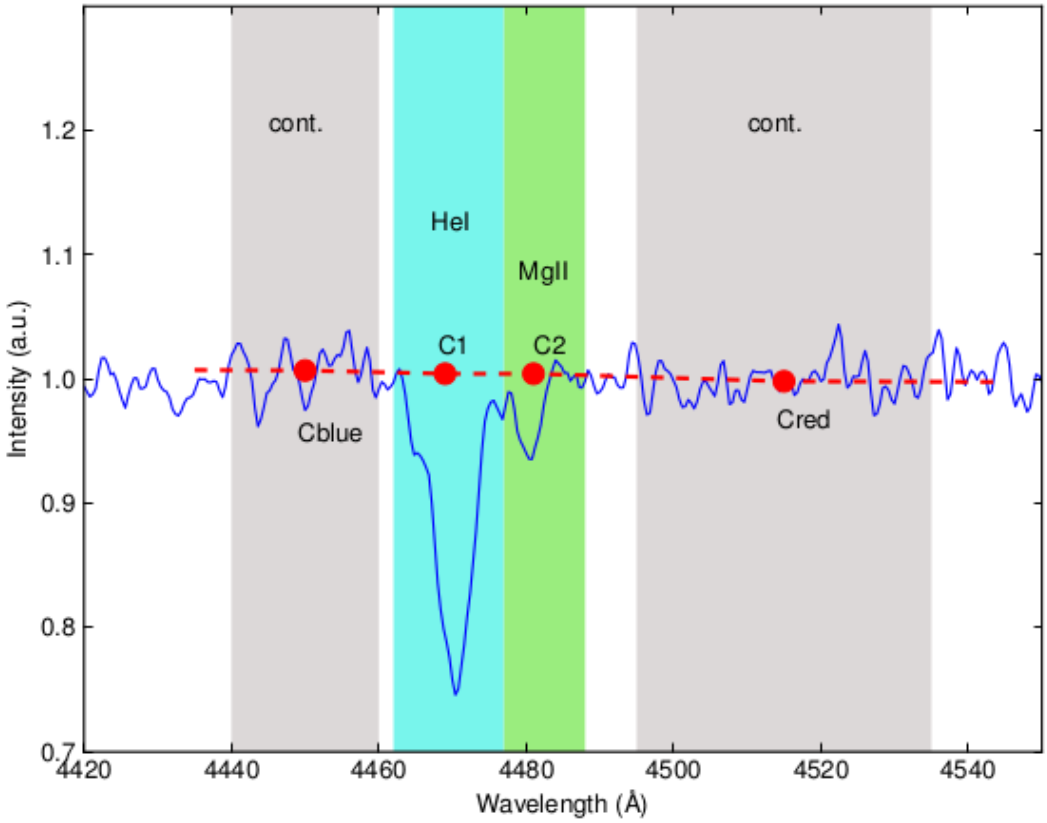

Figure 2.4: Example of the measuremet of EW - The figure shows the regions used to measure the EW for the lines HeI λ4471(cyan) and MgII λ4481 (green) and their corresponding blue and red continuum (gray) (Maravelias, 2014)

Table 2.2: The full set of the the characteristic spectral lines and their central wavelength for which we measured their EW.

Line ID	λ_{central} (Å)
CaIIK	3928
HeI	4009
HeI	4026
SiIV	4088
SiIV	4116
HeI	4121
SiII	4130
HeI	4144
HeII	4200
FeII	4233
HeI	4387
OII	4416
HeI	4471
MgII	4481
HeII	4541
SiIII	4553
OII+CIII	4645
HeII	4686

Table 2.3: Wavelength ranges for the spectral line measurements and their continuum regions (Maravelias, 2014).

Line ID/	Spectral line		Continuum blue		Continuum red	
$\lambda_{central}$ (Å)	λ_{start} (Å)	λ_{end} (Å)	λ_{start} (Å)	λ_{end} (Å)	λ_{start} (Å)	λ_{end} (Å)
CaIIK/3928	3924	3932	3908	3922	3935	3955
HeI/4009	4004	4016	3935	3955	4035	4060
HeI/4026	4017	4035	3935	3955	4035	4060
SiIV/4088	4084	4091	4035	4060	4150	4190
SiIV/4116	4113	4118	4035	4060	4150	4190
HeI/4121	4118	4125	4035	4060	4150	4190
SiII/4130	4125	4135	4035	4060	4150	4190
HeI/4144	4140	4150	4035	4060	4150	4190
HeII/4200	4190	4207	4150	4190	4238	4260
FeII/4233	4229	4237	4150	4190	4238	4260
HeI/4387	4378	4395	4360	4380	4398	4411
OII/4416	4412	4421	4398	4411	4440	4460
HeI/4471	4462	4477	4440	4460	4495	4535
MgII/4481	4477	4488	4440	4460	4495	4535
HeII/4541	4537	4547	4495	4535	4580	4620
SiIII/4553	4548	4558	4495	4535	4580	4620
OII+CIII/4645	4635	4655	4600	4630	4660	4670
HeII/4686	4679	4692	4660	4670	4737	4747

3 Code implementation and Results

The general idea behind the use of machine learning techniques is to split the sample into two sets one used for training the algorithm and one for testing the performance. When the best model is determined we can use this model to predict new unseen data. With this in mind we implemented our code and in this chapter we present all the procedure that we followed as well as the results we achieved. The main tool that we used in our analysis is the python library for machine learning `scikit-learn` version 0.21.3 (Pedregosa et al., 2011).This library contains the RF and a multitude of other algorithms as well as a set of differnet metrics for the evaluation of the performance.

Data preparation

Before we run the RF algorithm we prepared and organized the data. First, for each spectral line we measured its EW. These are the features used in our method.

EW measurments of the characteristic spectral lines and sample size

As we mentioned in Section 2.2 the sample that we managed to collect consisted of 1378 stars. The first step of our analysis was to measure the EWs of each spectral

line from Table 2.2 in order to calculate their values which correspond to the features for each star used in RF. For this reason we followed the method that we described in Subsection 2.4.1 based on a code from (Maravelias, 2014). After we ran the code, we examined our results and we found out that in a non-negligible number of cases the EW of some spectral lines had not be measured. Although, the augmentation of data in most cases is based on averaging values, for our case this is not an acceptable solution. Given the lack of any other appropriate way to solve this as well as that the current version of the algorithm does not handle missing values , we opted to take into account only sources with the full set of features measured. Consequently, we continued our analysis with 777 objects out of the initial 1378 objects.

Sample binning

In Figure 3.1 we present the initial spectral type distribution of our sample. As it is shown in our sample we had a number of diffrerent spectral types and their sub spectral types in the range O4-B9. However, for a few classes we had only a few objects and theit number is not respresentative of their class. For example the classes O4.5,O5 or B0.7 and B6 had less than 5 objects each one. The RF algorithm cannot be efficient with such small sample sizes. Thus, we binned our sample in such a way to have a physical meanining for the spectral classification and also in order to decrease the number of the classes.

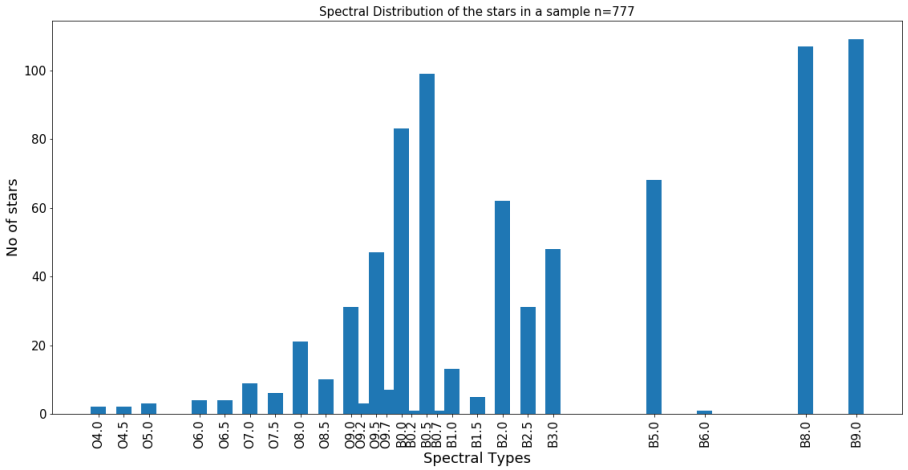

Figure 3.1: - The figure shows the initial spectral types distribution without binning the sample.

In Figure 3.2 we present the final spectral type distribution of our sample. We followed an adaptive binning where we binned together the spectral sub-types from O4.5 to O7 at the spectral sub-type O4. For the rest we binned them to 1 type. Thus all the intermediate cases (e.g O8.5,O9,O9.5,O9.7 etc.) were binned in such a way that all the objects with a sub spectral type for example B0.5 or B0.7 were counted as type B0 following the convention [B0,B1). By doing that we decreased the classes to be predicted from 26 at 10.

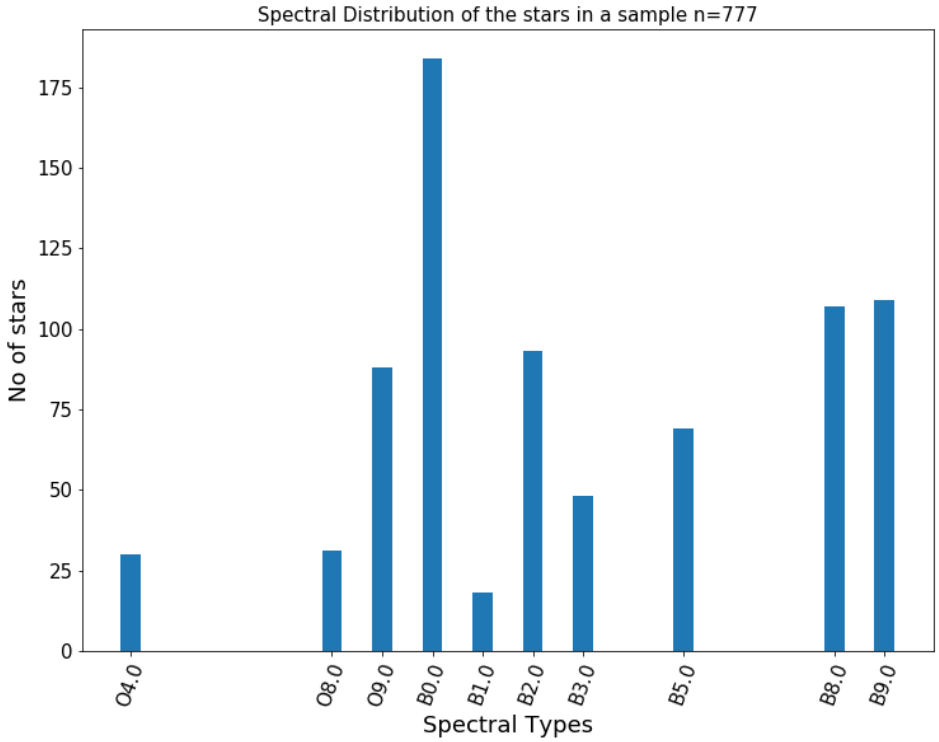

Figure 3.2: - The figure shows the final spectral types distribution based on adaptive binning.

Running the RF algorithm

After the preparation and the organization of the data we applied the RF algorithm.

First, we randomly shuffled our complete dataset and we splitted it in two data subsets. For the training data we considered the **70%** of the full set and for the testing data the **30%**. In Table 3.1 we present a summary of the sample's demographics based

on this split.

Table 3.1: A summary of the initial sample of stars.

Total sample	Training sample	Testing sample
777	543	234

After, the splitting of the data we ran the RF algorithm with its default hyperparameters (a parameter whose value is set before the training process begins) in order to have a first idea of its behavior and its performance. We used all the characteristic spectral lines in Table 2.2.

Confusion Matrix and metrics evaluation

In the machine learning and especially in the problem of statistical classification the confusion matrix is a specific table layout that allows visualization of the performance of an algorithm.Each row of this table represents the instances in an actual class while each column represents the instances in a predicted class. The confusion matrix shows the number of objects in each class versus the number of objects predicted by the model to belong to a particular class.In an ideally confusion matrix where the prediction ability of our model is 100% correct we would expect all the objects of each class to be in the diagonal of the matrix.

In order to interpretate a confusion matrix and evaluate the performance of a model with different metrics, the definition of some basic terms is needed.

- **True Positives (TP)**: The number of objects that has been predicted to belong to a class A, and they actually belong in this class.
- **True Negatives (TN)**: The number of objects that has been predicted to not belong in a class A, and they actually do not belong in this class.
- **Fasle Positives (FP)**: The number of objects that has been predicted to belong in a class A, and actually they do not belong in this class.
- **False Negatives (FN)**: The number of objects that has been predicted to not belong in a class A, but actually they belong in this class.

The most important metrics that used in the evaluation of the results of a confusion matrix are:

- **Accuracy**: Accuracy is an estimation of how often the classifier predicts correct.

$$\frac{TP + TN}{total} \tag{3.1}$$

- **Missclassification rate**: An estimation of how often the algorithm predicts incorrect.

$$\frac{FP + FN}{total} \tag{3.2}$$

  or

$$1 - Accuracy \tag{3.3}$$

- **Precision**: Presicion is an estimation of how precise is a model. Out of those objects that have been predicted positive, how may of them are actually positive.

$$\frac{TP}{TP + FP} \tag{3.4}$$

In Figure 3.3 an example of a confusion matrix is shown taken from (Mahabal et al., 2017), who trained a deep learning model to distinguish between 7 classes of cariable stars. Each class of stars corresponds to numbers 1,2,4,5,6,8 and 13 in the diagram.

Now that we defined the confusion matrix and the metrics that we used we present in Figure 3.4 the confusion matrix that we came up after the first running. As it is shown the Accuracy of RF in this case is $\sim$ **64%**. Furthermore, in Table 3.2 we present the results of the two important metrics of RF algorithm for each class. The accuracy and the presicion. The fourth column represents the testing sample for each class.

As it is obvious the RF model seems to work but it can be improved.

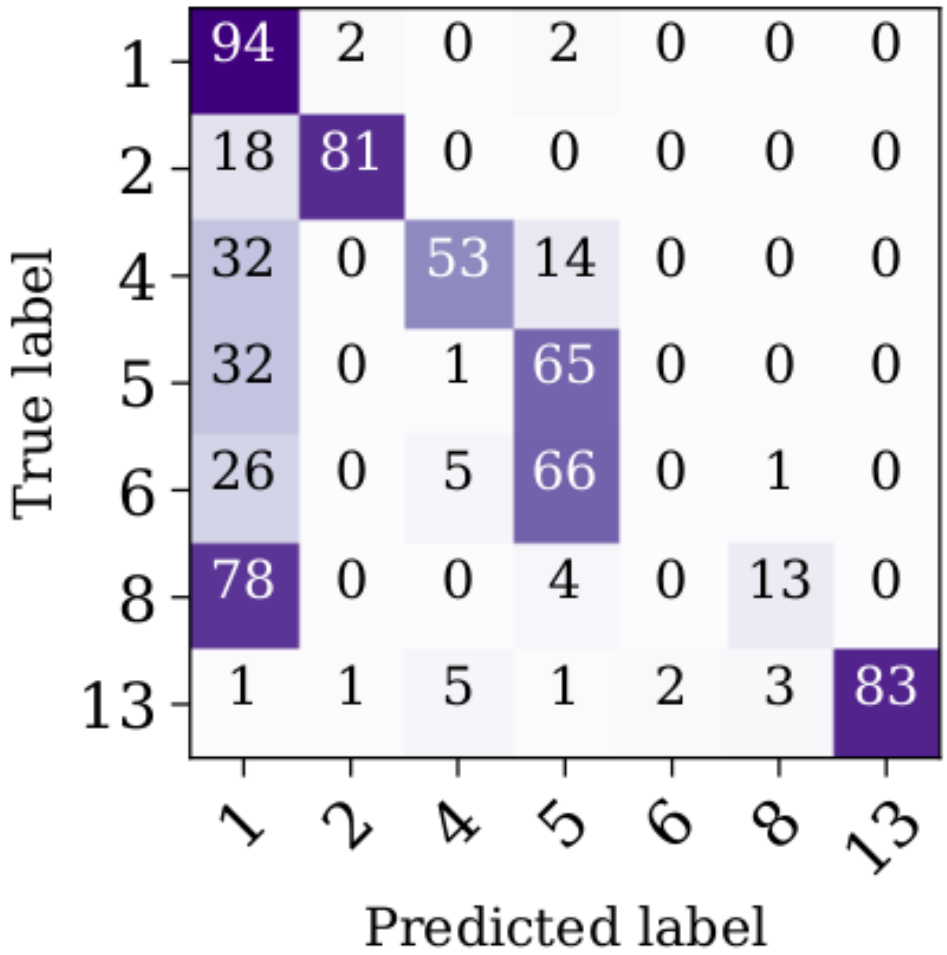

Figure 3.3: Confusion Matrix example - The figure shows a confusion matrix from (Mahabal et al., 2017) for a classification task for variable stars.

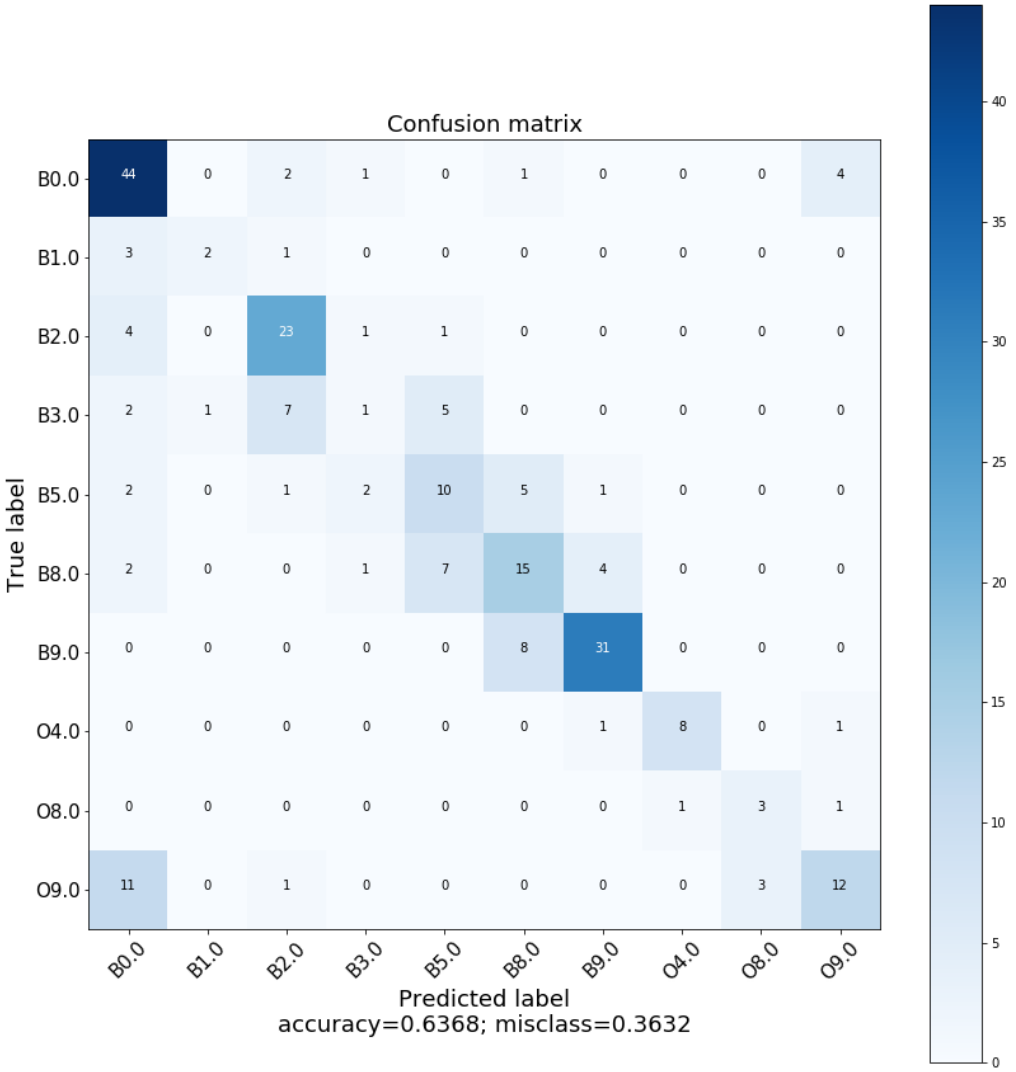

Figure 3.4: Confusion Matrix with default RF hyperparametes - The figure shows the confusion matrix from the 1^{st} run of RF algorithm with the default values of hyperparameters and the full set of features.The x-axis is the predicted label and the y-axis is the true label of the the testing dataset.The accuracy is $\sim 64\%$ and the Misclassification rate is $\sim 36\%$.

Table 3.2: A classification report for the running of RF with its default hyperparameters and the total set of features.

Class	Accuracy	Presicion	Support sample
B0	0.84	0.65	52
B1	0.50	0.67	6
B2	0.79	0.66	29
B3	0.43	0.17	16
B5	0.47	0.43	21
B8	0.51	0.52	29
B9	0.79	0.84	39
O4	0.80	0.89	10
O8	0.60	0.50	5
O9	0.44	0.67	27

Improving the RF algorithm

Despite the fact that the RF seems to work an accuracy of 64% it is not optimal. Thus, we tested some usual techniques in machine learning in order to increase the score of our model. We focused on the values of the most important hypeparameters that are used from RF algorithm as well as we investigated the features importances and we tried to see if any specific features combination results in a better score.

K-Fold Cross-Validation

Cross-Validation (CV) or rotation estimation (Kohavi, 1995) is a way to estimate how accurately a predictive model will performe in new unseen data. The characteristic of this method is that it takes into account the whole sample to define the accuracy of a model. The goal of CV is to test the model's ability to predict new data that was not used in estimating it, in order to avoid problems like overfitting. In addition, with this test we can estimate our model after a number of runs.The algorithm is very simple and it can be described as follows.

1. An initial dataset is splitted into k smaller datasets or as they called in the terminology "folds".
2. For each of the k-folds a model is training using $k-1$ of the folds as a training set.
3. Then the resulting model is validated on the remaining data and reports an accuracy value.
4. The final value of the accuracy is then the average of the values that previous computed in the loop.

In Figure 3.5 we present an example of how the K-Fold CV works. The initial sample is devided into 5 folds. At split 1 the model is trained based on the training set of 4 folds and is tested on the remaining fold. At split 2 the previous testing fold is replaced by another fold from the training dataset and is now one of the 4 training folds. At each split an accuracy value is computed and the final accuracy of the model is the average of these values. The procedure ends when all of the folds were used as a testing set.

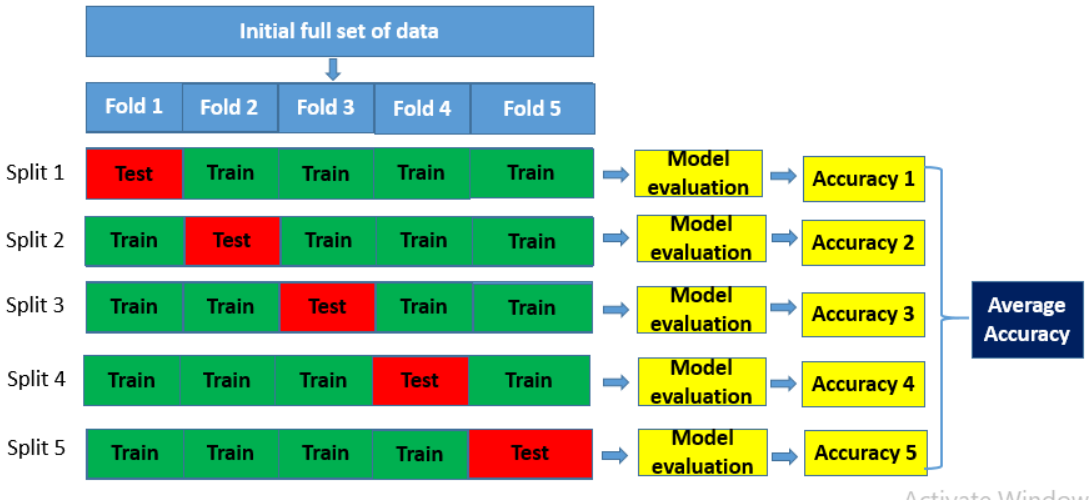

Figure 3.5: - An example of K-Fold cross-validation

Driven by the above advantage of the K-Fold CV we applied this test in our case. In particular, we used a variation of the original K-Fold algorithm the `StratifiedKFold` [1]. This algorithm follows exactly the same procedure that we previous described with the only difference that each set contains approximately the same percentage of samples for each target class as the complete set. Thus, by using this CV test we ran again the RF algorithm considered again the default values for the hyperparameters and and the full set of features.

[1] `https://scikit-learn.org/stable/modules/cross_validation.html`

In Table 3.3 we present the results from this run. We chose to split the initial sample into 5 folds because for more folds each subset would constist of an even smaller number of objects due to the relatively small number of the full sample. The average accuracy is now $\sim$ **63%**.

Table 3.3: K-Fold report for the RF algorithm.

Split	Accuracy
1	0.62
2	0.60
3	0.63
4	0.72
5	0.57
Average accuracy	**0.63**

Tuning hyperparameters

The next step in order to improve the performance of our algorithm was to search more methodically for the values of the most important hyperparameters of RF. As a hyperparameter is defined a parameter whose value is set before the training process begins. Different model training algorithms require different set of hyperparameters. Given these hyperparameters, the training algorithm learns the parameters from the data. In our case, we have a large number of hyperparameters but the most important are:

- **n_estimators**: The number of trees in the forest. The default value is 10.
- **max_depth**: The maximum depth of the tree or in other words the max number of levels in each decision tree. The default value is *None* which means that the nodes are expanded until all leaves are pure.
- **min_samples_split**:The minimum number of samples required to split an internal node. The default value is 2.
- **min_samples_leaf**:The minimum number of samples required to be at a leaf node. The default value is 1.
- **max_leaf_nodes**: This hyperparameter affects the way that trees grow in order to have the best result. Best nodes are defined as relative reduction in impurity.

The default value is `None` which means that an ulimited number of leaf nodes is derived.

- **min_weight_fraction_leaf**: The minimum weighted fraction of the sum total of weights (of all the input samples) required to be at a leaf node.The default value is 0. Then samples have equal weight.

For more details see at `RandomForestClassifier`[1] documentation.

Validation Curves

The first step was to plot different values of each hyperparameter versus the algorithm's score , i.e. the validation curves. We investigated only the most important hyperparameters that were presented in Section 3.2. We determined a specific range of values for each hyperparameter(starting from their default values) and we ran the RF model for each value using a CV test similar with that has been previously described. More specifically we used 5 folds and a range of values for each hyperparameters, while keeping locked the others. Finally, in Figure 3.6 we plot the validation curve of each hyperparameter. By investigating each plot seperately we saw that each hyperparameter behaves different. The n_estimators increases rapidly for low values and after a threshold it remains almost constant. The same happens for the max_depth and the max_leaf_nodes. In contrast, the min_samples_leaf and the min_weight_fraction_leaf show higher scores for low values (closer to their default) and gradually the score decreases. Finally, for the min_samples_split we observed that it seems to results in a constant score between the values 0.65-0.67 and after 20 for the hyperparameter value it has a general decreasing trend.

Driven by these results we decided to investigate with higher accuracy the values of the hyperparameters that result in the best performance of our model. For this reason we applied another technique for the tuning of the hyperparameters which called Grid Search.

[1] `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html`

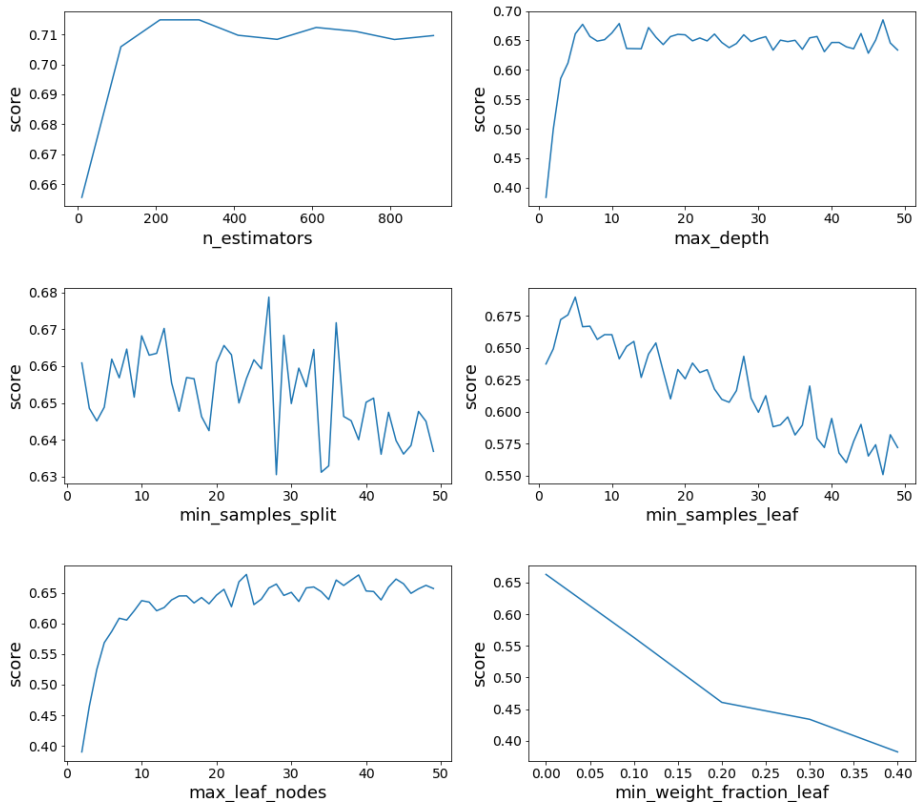

Figure 3.6: - The validation curves of the most important RF hyperparameters.

Grid Search

The Grid Search is one of the most robust techniques in machine learning to find the optimal hyperparameters of a model which results in improved predictions. `GridSearchCV`[1] builds a model for every combination of hyperparameters specified and evaluates each model. In particular Grid Search takes as an input a specific range of values for each hyperparameter and it tries to find the best combination of these values for each run. During each run it uses again a CV test. It is an expensive technique in terms of computational time but it is powerful and accurate method for the hypeparameters tuning. Consequently, in order to save some computational time we applied the Grid Search method using a range of values determined by the validation curves shown in Section 3.3.2.1 and they are presented in Table 3.4.

Table 3.4: Value ragnes for the hyperparameters that were used in the Grid Search method.

Hyperparameter	Range	Step
n_estimators	100-700	50
max_depth	5-30	5
min_samples_leaf	1-15	3
min_samples_split	2-20	4
max_leaf_nodes	10-40	3

For the n_estimators , max_depth and the max_leaf_nodes we selected the region where the performance of the algorithm becomes constant. For the min_samples_split and min_samples_leaf we took into account a smaller region close to the default value because the performance of the algorithm is better there than at higher values of these hyperparameters. Finaly, we did not consider the hyperpararmeter min_weight_fraction_leaf for further analysis with the Grid Search method because from the corresponding validation curves the highest scores are achieved for the default values. The best values we found are presented in the Table 3.5. For the CV test the number of folds that we used was 3 in order to decrease computational time.

[1] `https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html`

Table 3.5: Best Values for the hyperparameters as they were obtained from the Grid Search method.

Hyperparameter	Best value
n_estimators	400
max_depth	15
min_samples_leaf	4
min_samples_split	2
max_leaf_nodes	35

Feature selection

Until now we have tried to determine the values of the most important hypeparameters that result in the best performance of the RF algorithm. The final step for the improvement of RF was to study if there is any specific combination of features that can result in a better score. For this reason we used an Sequential Feature Selection algorithm in a attemp to improve the computational efficiency and reduce the generalization error of the model by removing irrelevant features or noise.

Sequential Feature Selection algorithms (SFAs) are a family of search algorithms that are used to reduce an initial d-dimensional feature space to a k-dimensional feature subspace where $k < d$ (Pudil et al., 1994). They are based on the wrapper method which marries the feature selection process to the type of model being built, evaluating feature subsets in order to detect the model performance between features, and subsequently select the best performing subset. Despite the fact that there are different versions of algorithms that belong to this family we used the Sequential Forward Floating Selection (SFFS) for our analysis. More specificaly,we used the Mlxtend (machine learning extensions) which is a python library and includes the `SequentialFeatureSelector`[1].

[1] `https://rasbt.github.io/mlxtend/user_guide/feature_selection/SequentialFeatureSelector/`

Sequential Forward Floating Selection

The Sequential Forward Floating Selection algorithm can be described as follows:

- The SFFS algorithm takes as an input the whole set of features of size d.

$$\mathrm{Y}=\{\mathrm{y}_1, y_2, ..y_d\}$$

- The returned output of the algorithm is a subset of the feature space of a specified size.

$$\mathrm{X}_k = x_j | j = 1, 2, ..., k; x_j \in Y$$

  where $k < d$.

- The initialization of the algorithm requires

$$\mathrm{X}_0 = 0, k = 0$$

- The first step of the algorithm is to evaluate the model by using each individual feature. For the evaluation of the model it uses again a CV test.Afterwards, it selects the feature which results to the best performance of the model.

- The second step is to evaluate again the model with all the possible combinations of the selected feature and a subsequent feature and select these that result in the best performance of the model.

- This procedure is being sequentially and terminates when the size of the output subset of feautures X_k reaches the number k that we predefined in the SFFS algorithm.

Thus, we used the above algorithm to test the full set of features which is the scheme of Table 2.2. The number of folds for the CV test that we used in the model evaluation was 5. The extra step added was the search for the best combination of features for all possible sizes of the subset X_k. So we repeatedly executed the SFFS algorithm and each time the size k of the requesting subset was varying within the range 2-18. Then, we plotted the size of these subsets ,i.e. the number of the features versus the best score that was suggested from the model evaluation and we present it in Figure 3.7. As we can see from the plot the SFFS algorithm suggests that the best combination of features is not the full set of 18 lines but a smaller subset which consists of 14 and results in

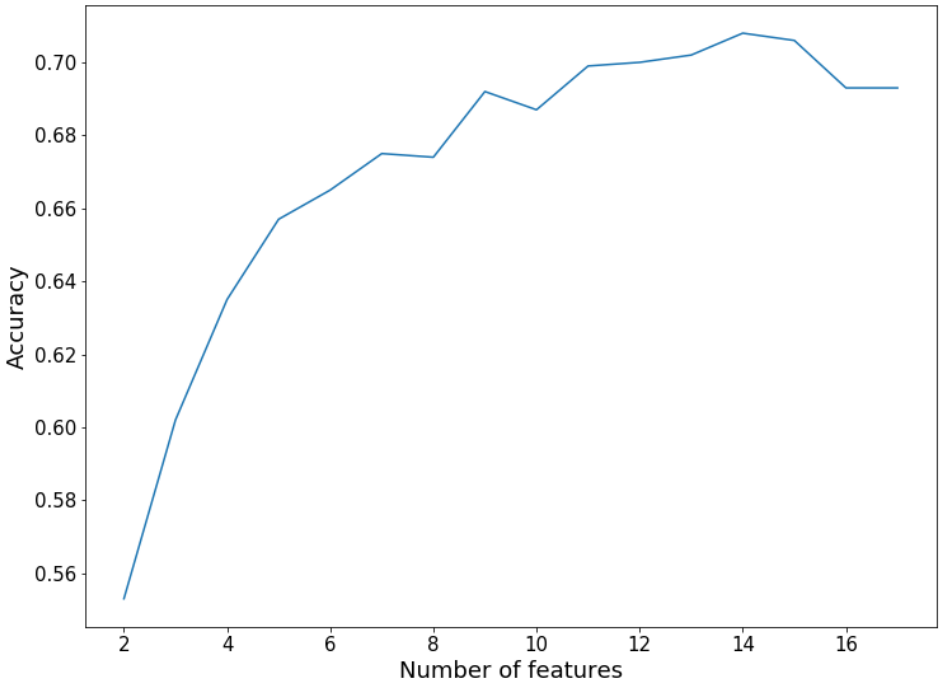

Figure 3.7: - The number of best features versus the accuracy of the RF algorithm

an accuracy of $\sim$ 70%. Furthermore, we see that for a small number of features the accuracy is low which is that we expected, since the model has not enough information in order to classify correctly the objects in each class. The features selected from the SFFS algorithm are presented highlighted in the Table 3.6.

Table 3.6: The best combination of features that was obtained from the SFFS algorithm.

Line ID	λ_{central} (Å)
CaIIK	3928
HeI	4009
HeI	4026
SiIV	4088
SiIV	4116
HeI	4121
SiII	4130
HeI	4144
HeII	4200
FeII	4233
HeI	4387
OII	4416
HeI	4471
MgII	4481
HeII	4541
SiIII	4553
OII+CIII	4645
HeII	4686

Feature importance

After finding the best combination of feautures (**14** out of **18**) we wanted to investigate how important each one is. For this reason we used the `feature_importances_`[1] which is a method of `RandomForestClassifier` that calclulates which of the input features is more important during the training process. In Figure 3.8 we present a plot of the importance of each feature. As we can see characteristic spectral lines such as HeII/4686 or the blend of OII+CIII/4645 are important for the performance of the algorithm as we expected from our previous experience with the visual spectral classification. In general, when we classify an early type star based on visual inspection of its spectrum the most distinct lines are HeII lines. In contrast, lines such as SiIII or SiIV are less prominent and thus more difficult to recognize them. Therefore, someone would expect these lines to not be significant, something that is verified by the feature analysis we did in Figure 3.8 latest lines to not be importnat for the algorithm which it also seems from the plot.

[1] `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.feature_importances_`

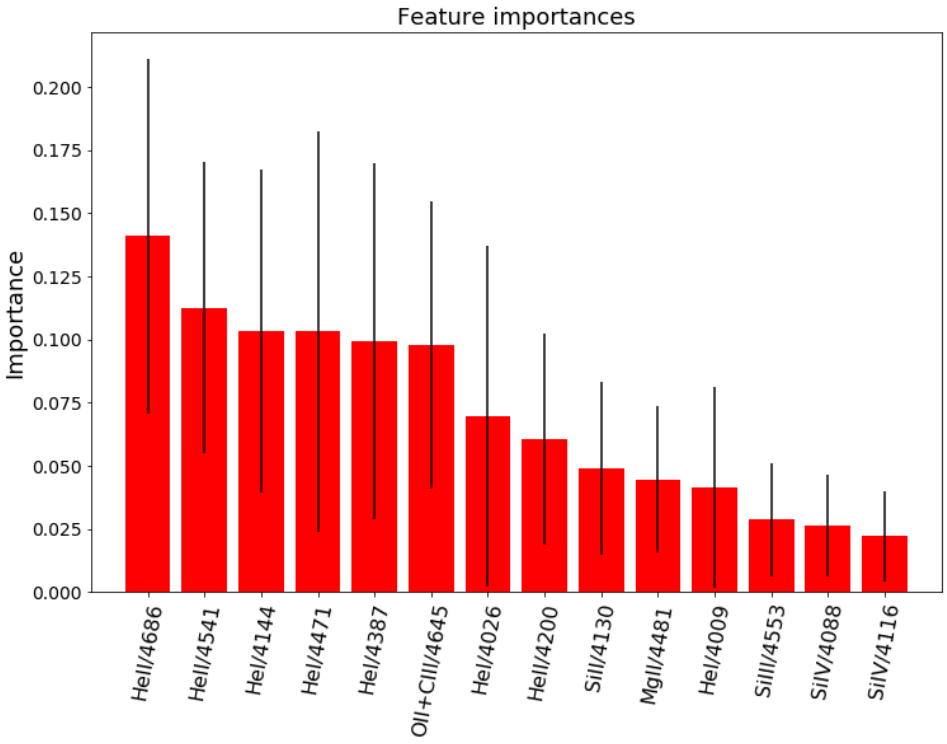

Figure 3.8: - Feature importances for the best combination of features

Evaluating the performance of RF algorithm based on the best model

In the previous sections we determine the best combination of the hyperparameters and the best combination of the features. The next step was to use all the results and rerun our model. In order to determine the final performance of the model we splitted again the initial set of data by using the the **70%** of them for training and the **30%** for testing. In Figure 3.9 we present the final confusion matrix. In addition, in Table 3.7 we present again the results of the two important metrics of RF algorithm for each class, accuracy and the presicion. The fourth column represents the testing sample for each class. As it is shown the final accuracy of our model is $\sim$ **72%** and the misclassification rate is $\sim$ **28%**.

Table 3.7: The classification report for the run of RF with the best values of the hyperparameters and the best combination of features.

Class	Accuracy	Presicion	Support sample
B0	0.93	0.76	60
B1	0.50	0.50	4
B2	0.68	0.68	25
B3	0.62	0.50	13
B5	0.48	0.56	21
B8	0.64	0.70	33
B9	0.94	0.80	34
O4	0.80	1.00	10
O8	0.25	0.50	10
O9	0.67	0.64	24

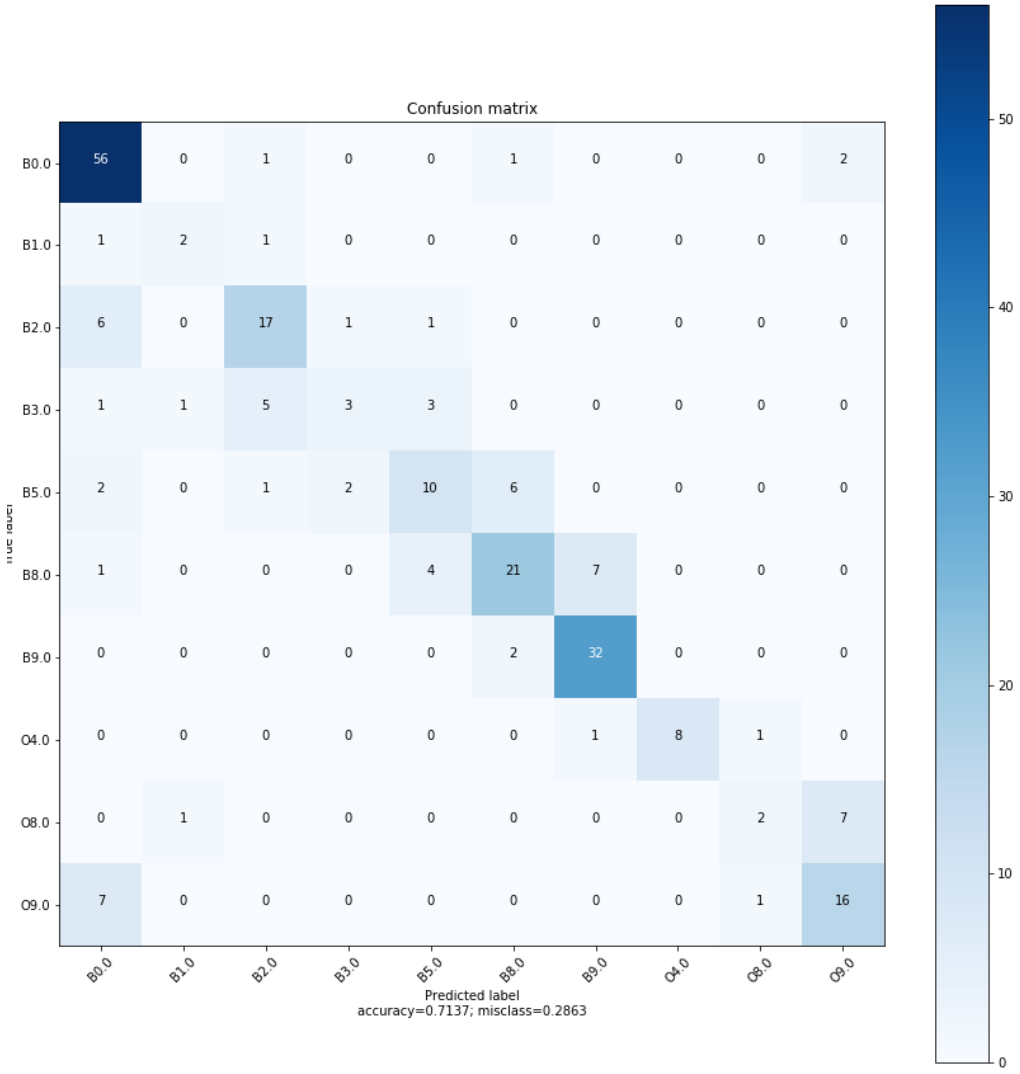

Figure 3.9: Final confusion matrix based on the best model - The figure shows the confusion matrix from the final run of RF algorithm with the best values of hyperparameters and the best combination of features.The x-axis is the predicted label and the y-axis is the true label of the the testing dataset.The accuracy is $\sim 72\%$ and the Misclassification rate is $\sim 28\%$.

In addition, we ran our best model again based on the K-Fold method by using 5 folds in order to estimate the stability of our algorithm and determine the accuracy when all sample is taken into account. In Table 3.8 we present the accuracy scores of each split and the average accuracy. As it is shown the average accuracy is $\sim$ **70%**.

Table 3.8: K-Fold final report for the RF algorithm based on the best model.

Split	Accuracy
1	0.74
2	0.71
3	0.65
4	0.67
5	0.70
Average accuracy	**0.70**

This means that we achieved an improvement of the prediction ability of our algorithm almost $\sim$ **7%** in comparison to the previous model. At this point we have to emphasize that the construction of the confusion matrix is a stochastic procedure , because of the randomness in the selection of the training and the test samples, which means that in each run we will receive a different accuracy score. Nonetheless, the accuracy of the confusion matrix is consistent with the average accuracy from the K-Fold cross validation test.

4 Discussion

Comments on the results

For the interpretation of our results, firstly we applied a K-Fold cross validation test on the initial and the final model. In Figure 4.1 we present the scores from this test for the initial (blue) and the final (orange) model,respectively. In the final model the accuracy is more stable.

The Random Forest predicts a probability distribution of the object to belong in each class and the final predicted class is the class that corresponds to the maximun probability of this distribution. In other words, when an object is assinged with a specific spectral type this spectral type has the highest probability value in the probability distribution. In Figure 4.2 we plotted the number of test sources that have been predicted correctly and incorrectly versus the probability. In particular, from each source we take into account the class corresponding to max proability. In Figure 4.3 an example of this probability distribution is shown.

We see that above the threshold of probability $\sim 50\%$ we can trust the result of our model. For example, above $\sim 50\%$ we have 105 correct predicted sources and 20 incorrect. Thus, more than 80% accurate at this domain. In other words, if we predict a star to be a spectral type of B0 with a probability of 70% we can trust this prediction with safety. In contrast, if the probability of the prediction is 30% we cannot tell if the source is classified correctly or not given the almost equal probability of those two option at this probability range.

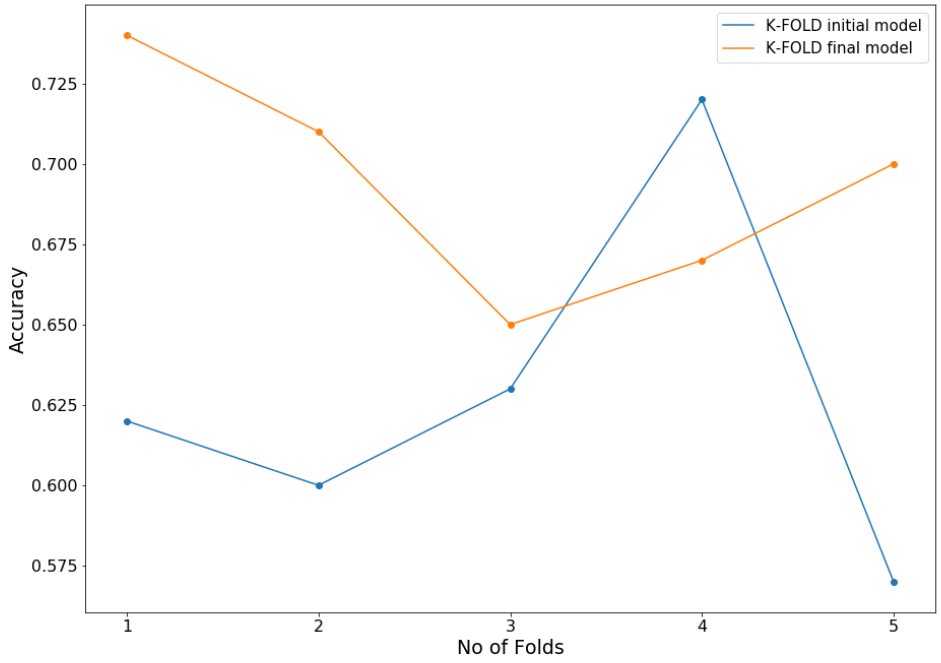

Figure 4.1: K-Fold comparison between initial and final model - Comparison of the accuracy's values between the initial(blue) and the final (orange) model based on the results of the K-Fold CV test.

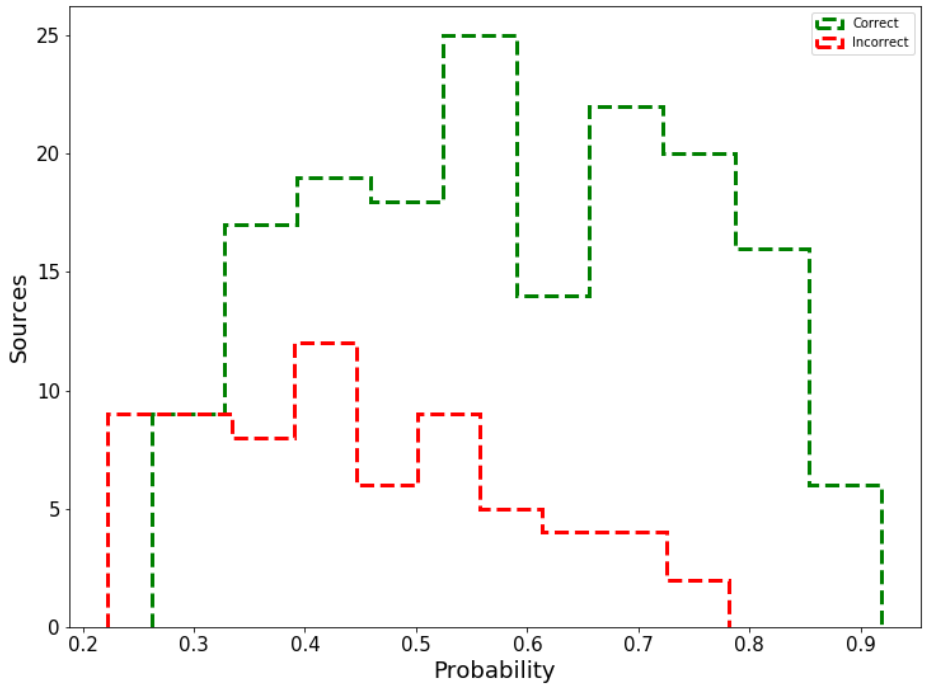

Figure 4.2: Probability distribution for the correct and incorrect sources - Comparison of the probability distribution between the sources that have been predicted correctly (green line) and incorrectly (red line). For each source we take into account the class corresponding to max proability with respect to its probability distribution plot.

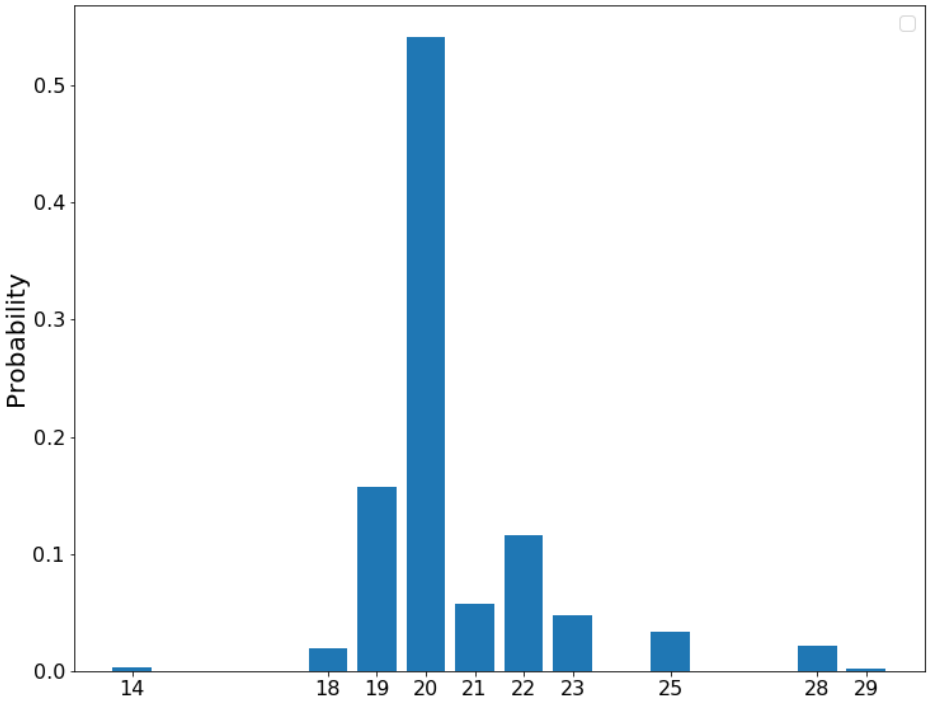

Figure 4.3: Probability distribution of a source to belong in each class - In this probability distribution plot we can see the probability of the source to belong in each class. The final class corresponds to the spectral type with the maximum probability. Thus for this source the final prediction is B0. For the spectral types we follow the convention: "O" corresponds to "1" and "B" corresponds to "2".(e.g "14" is "O4")

Application of the method on unseen data

The ultimate goal of this project is the application of the method developed, on unseen data and more specifically in BeXRBs sources, which as we saw in the introduction one of their components is an OB star. Thus, we applied the best model on a total sample of 23 stars (33 spectra; for a few sources we have more than one spectra) that have been previously classified by visual insepction. So, 18 out of 23 objects are from the SMC and have been classified in the work of (Maravelias et al., 2014) and the rest 5 objects are galactic spectra taken from Skinakas observatory[1](P.Reig,private communication) and their classification is derived from the `SIMBAD` astronomical database[2]. Thus, the full sample consists of stars from different metallicity enviroments (SMC and Milky Way).

In order to decide which object has been predicted correctly or incorrectly we investigated the probability distribution of each object to belong in each class. However, for a few objects in both samples(SMC and Galactic) we saw that in their probability distribution plots there was not a clear suggested class. These objects are CH3_18,CH4_8 and CH3_7 for the SMC sample and LSI61235 and SAX2103 for the Galactic sample. For example, the star CH3_7 has $\sim$ 38% probability to belong in B2 spectral type and 28% probability to belong in B0 spectral type. Thus, we can not say with safety that the object has a spectral type B2. Consequently, the most realistic choice is to assign a range of spectral types B0-B2. The rest 34% is distributing on the other spectral types.

In Figures 4.4, 4.5 ,4.6 and 4.7 we present these probability distributions of the SMC sample and the Galactic sample respectively

Driven by these probability distribution plots, we considered as correctly identified sources the ones with a predicted class or range of classes that was consistent within 0.5 or 1 sub-spectral type with the real one (e.g O9.5 and the predicted class of a form e.g O9).

In Table 4.1 and Table 4.2 we present the classification results of our model as well as the comparison with previous works.

In conclusion, for the SMC objects we see that our model predicts correctly **11** out of **18** objects ,i.e a success rate of $\sim$ 60%. For the galactic sample the result is **3** out of **5** corresponding again to $\sim$ 60%. Given that our model accuracy is $\sim$ 70% , this

[1] `http://skinakas.physics.uoc.gr`

[2] `http://simbad.u-strasbg.fr/simbad/`

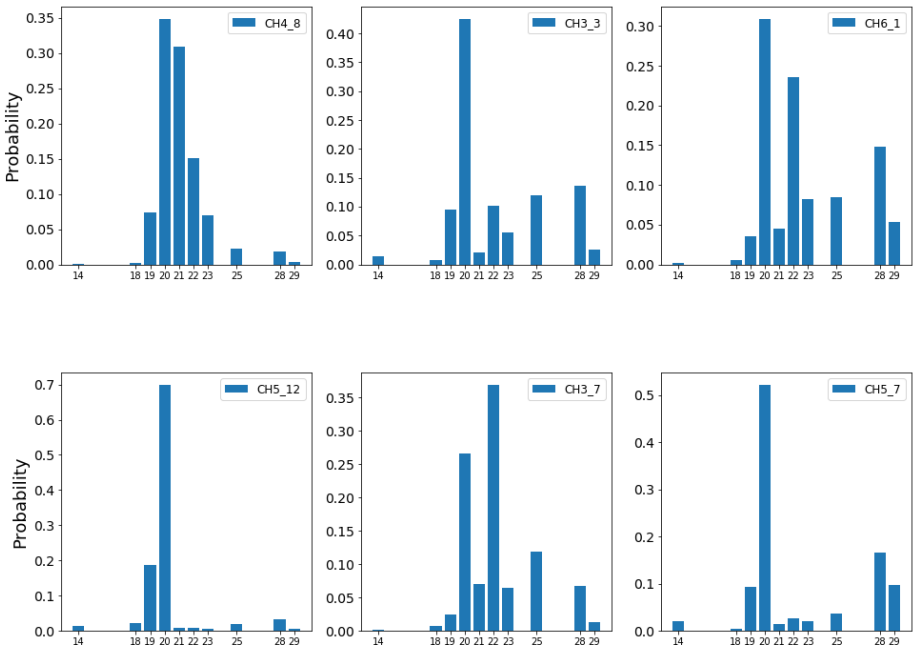

Figure 4.4: Probability distribution for the objects from SMC sample to belong in each class (1) - The probability distribution for all classes in our model (18 corresponds to SMC sample and 5 to Galactic sample. In most cases we see that a specific spectral type can be derived as a clear maximum probabilty. However, in some cases (e.g CH4_8, CH3_7 etc.) we cannot clearly set a spectral type given the distribution of the probability to nearby classes. Thus, for these sources we provide a range of types (see text for more details).

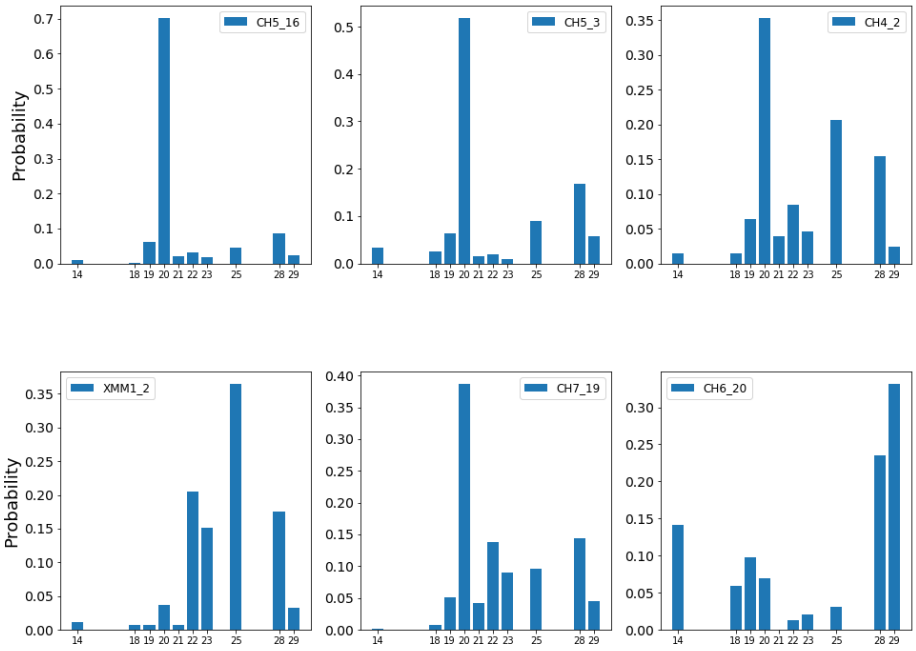

Figure 4.5: Probability distribution for the objects from SMC sample to belong in each class (2) - Similar to Figure 4.4

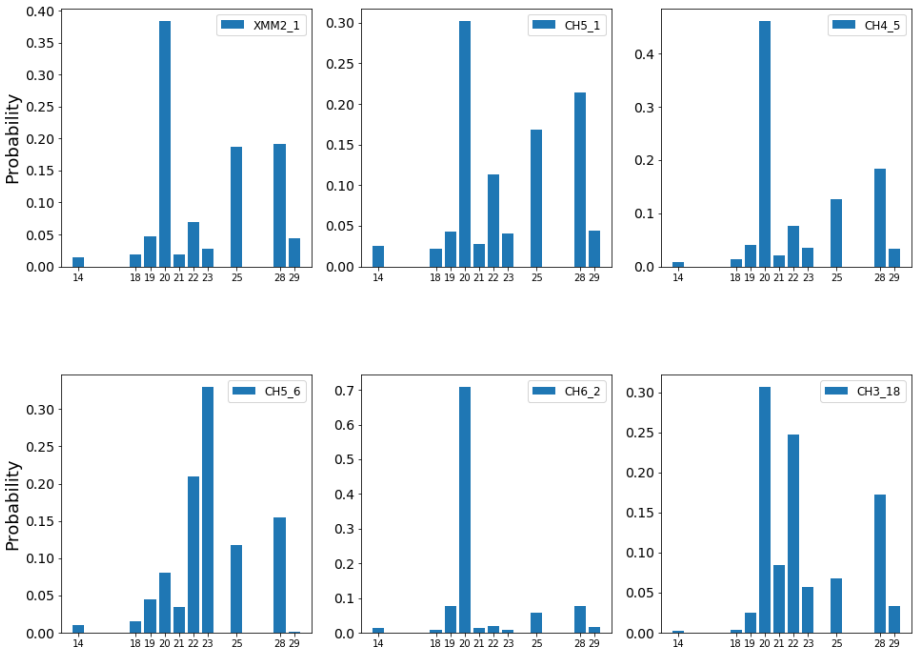

Figure 4.6: Probability distribution for the objects from SMC sample to belong in each class (3) - Similar to Figure 4.4

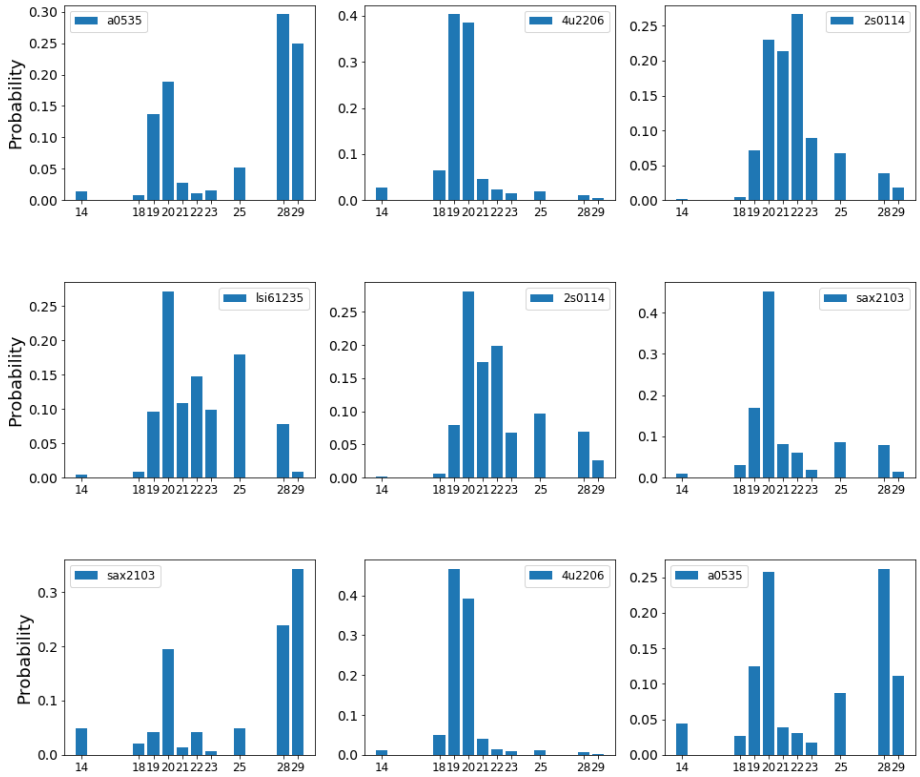

Figure 4.7: Probability distribution for the objects from Galactic sample to belong in each class - Similar to Figure 4.4s

Table 4.1: Classification results for SMC

ID	Classification		Result
	Maravelias	This work	
(1)	(2)	(3)	(4)
CH3_3	B1-B5	B0	Incorrect
CH3_7	B2	B0-B2	Correct
CH3_18	B2	B0-B2	Correct
CH4_2	B3-B5	B0	Incorrect
CH4_5	B1-B5	B0	Incorrect
CH4_8	B1.5	B0-B2	Correct
CH5_1	B3-B5	B0	Incorrect
CH5_3	B0.5	B0	Correct
CH5_6	B1-B5	B3	Correct
CH5_7	B0.5	B0	Correct
CH5_12	B0	B0	Correct
CH5_16	B0	B0	Correct
CH6_1	B1-B5	B0-B2	Correct
CH6_2	B1.5-B3	B0	Incorrect
CH6_20	B0	B9	Incorrect
CH7_19	B1-B5	B0	Incorrect
XMM1_2	B3-B5	B5	Correct
XMM2_1	B0-B1 or B1-B3	B0	Correct

Table 4.2: Classification results for galactic objects

ID	Classification		Result
	SIMBAD	This work	
(1)	(2)	(3)	(4)
LSI61235	B1	B0-B5	Correct
4U2206	O9.5	O9	Correct
SAX2103	B0	B0-B2	Correct
2S0114	B1	B0	Incorrect
A0535	O9-B0	B8	Incorrect

sample accuracy is more or less consistent. Furthermore, although our model does not classify correctly a few sources these cases can be explained.

In one particular case, CH6_20 ,from the SMC sample, the predicted spectral type is B9 instead of B0 which is the correct classification. This result is unfortunately not even close to the real spectral type. It is as B[e] supegiant star with a complicated circumstellar enviroment (Clark et al., 2013; Maravelias et al., 2018). More specifically CH6_20 has many emission lines within the continuum regions used for the EW, so the EW determination is not correct and resulting in an incorrect outcome also.

The quality of the spectrum is another factor that can affect the classification result of our method. For instance, in Figure 4.8 we present the spectrum range (i.e 4180-4700 Å) of two different Galactic sources, 4U2206 and A0535. For the first source the classification result from our model is O9 and is coherrent with the result of pevious works (prev. works O9.5 ; this work B0) which classified this source by visual inspection. As we can see in the spectrum the quality of characteristics spectral lines (e.g HeI/4200 , HeII/4541 and HeII/4686) ,that are indicators of a star with spectral type earlier than B0 , is high with respect to S/N ratio. In contrast, in source A0535 where the classification result is not even close with previous works (prev. works O9-B0 ; this work B8) we see that where we expect to see these lines we can not recognize them due to the low S/N ratio. In other words, that means that the EWs of these lines cannot be measured correctly due the high noise and an incorrect classification result is not unexpected. Thorough investigation with respect to the S/N sensitivity is needed to further understand the limitation of our model.

In general, we see that the developed model is working and even though we miss some cases these can be explained due to their uniqueness as sources or due to the poorer spectra. Thus, this is a promising method to further develop in a full automated spectral classifier.

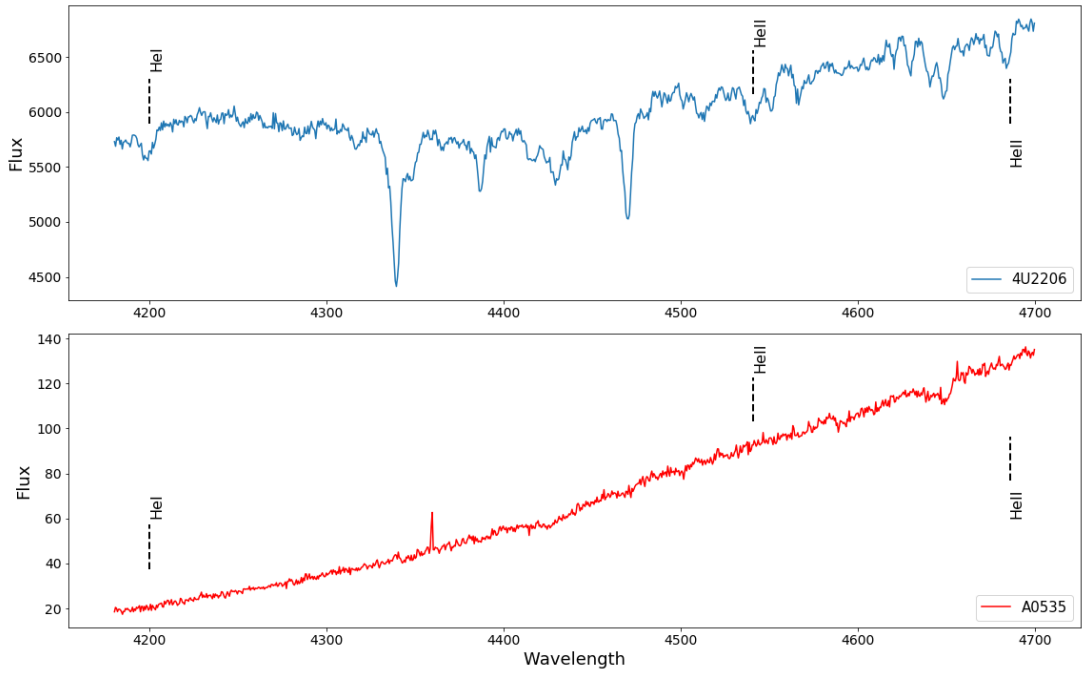

Figure 4.8: Spectra from two different Galactic sources - In this plot, we see the spectra regions of the Galactic sources 4U2206 (blue) and A0535 (red). In the first source the characteristics lines are clearly seen and the classification result is same with previous works. On the other hand, the source A0535 has a low quality spectrum that results in a incorrect determination of the EWs of these lines and thus in a wrong outcome.

5 Conclusions and Future plans

In this project we developed an new automated method for spectral classification of early type stars. We used the popular supervised machine learning alorithm Random Forests. For the input features of this classifier we used measurements of the EW for a set of spectral lines that characterize the classification of OB stars. We compiled a sample of 1375 stars from various publicly available surveys and we used finally 777 sources to build our model. After optimization we identified the optimal hyperparameter values and the number of features that provide the best score $\sim$ **70 %**. We applied our method on a sample of BeXRBs (previous classified by visual inspection) from the Galaxy and the SMC and we achieved an accuracy of $\sim$ 60 %. Misclassifid sources are due to the quality of the new spectra (i.e low S/N ratio) and the complicated enviroments of specific sources that results in a incorrect determination of the EW of characteristics lines. This is a preliminary work and there are a lot of plans for future work in order to increase the accuracy score of our model. More specifically the future plans can be summarized as follows:

- Try to increase the sample when larger samples of classified stellar spectra become available. In general, there is a lack of available public data that makes the collection of training sample a exacting job.

- Try to take into account the errors in the EWs measurements in order to have a more robust estimation of the error factor in the final predictions of our model.

- Making a number of sensitivity tests such as the limit in the S/N ratio of the spectra and investigate which spectral lines affect more in the prediction of each class
- Application of this new method in big databases (e.g SDSS) in order to classify early type stars in their sub-spectral types. A large number of these stars has a very general classification as OB spectral type stars.

References

Abbott, B. P., Abbott, R., Abbott, T. D., & Acernese, F. 2017, GW170817: Observation of Gravitational Waves from a Binary Neutron Star Inspiral 2

Antoniou, V. & Zezas, A. 2016, Star formation history and X-ray binary populations: the case of the Large Magellanic Cloud 6

Antoniou, V., Zezas, A., Hatzidimitriou, D., & Kalogera, V. 2010, Star Formation History and X-ray Binary Populations: The Case of the Small Magellanic Cloud 6

Ball, N. M., Brunner, R. J., Myers, A. D., Strand, N. E., Alberts, S. L., & Tcheng, D. 2008, Robust Machine Learning Applied to Astronomical Data Sets. III. Probabilistic Photometric Redshifts for Galaxies and Quasars in the SDSS andGALEX 9

Ball, N. M., Brunner, R. J., Myers, A. D., & Tcheng, D. 2006, Robust Machine Learning Applied to Astronomical Data Sets. I. Star-Galaxy Classification of the Sloan Digital Sky Survey DR3 Using Decision Trees 9

Baron, D. 2019, Machine Learning in Astronomy: a practical overview 8, 10, 12, 19

Belgiu, M. & Drăguţ, L. 2016, Random forest in remote sensing: A review of applications and future directions 21

Bilicki, M., Hoekstra, H., Brown, M. J. I., Amaro, V., Blake, C., Cavuoti, S., de Jong, J. T. A., Georgiou, C., Hildebrandt, H., Wolf, C., Amon, A., Brescia, M., Brough, S., Costa-Duarte, M. V., Erben, T., Glazebrook, K., Grado, A., Heymans, C., Jarrett, T., Joudaki, S., Kuijken, K., Longo, G., Napolitano, N., Parkinson, D., Vellucci, C., Verdoes Kleijn, G. A., & Wang, L. 2018, Photometric redshifts for the Kilo-Degree Survey. Machine-learning analysis with artificial neural networks 11

Carliles, S., Budavári, T., Heinis, S., Priebe, C., & Szalay, A. S. 2010, Random Forests for Photometric Redshifts 15

Chaty, S. 2011, Nature, Formation, and Evolution of High Mass X-Ray Binaries 4

Clark, J. S., Bartlett, E. S., Coe, M. J., Dorda, R., Haberl, F., Lamb, J. B., Negueruela, I., & Udalski, A. 2013, The supergiant B[e] star LHA 115-S 18 - binary and/or luminous blue variable? 63

Cox, A. N. 2000, Allen's astrophysical quantities 6

Evans, C. J., Howarth, I. D., Irwin, M. J., Burnley, A. W., & Harries, T. J. 2004, A 2dF survey of the Small Magellanic Cloud 22, 23

Gaia Collaboration, Prusti, T., & de Bruijne, J. H. J. 2016, The Gaia mission 8

Gray, R. O. & Corbally, J., C. 2009, Stellar Spectral Classification 22

Hartley, P., Flamary, R., Jackson, N., Tagore, A. S., & Metcalf, R. B. 2017, Support vector machine classification of strong gravitational lenses [LINK] 9

Huertas-Company, M., Primack, J. R., Dekel, A., Koo, D. C., Lapiner, S., Ceverino, D., Simons, R. C., Snyder, G. F., Bernardi, M., Chen, Z., Dominguez-Sanchez, H., Lee, C. T., Margalef-Bentabol, B., & Tuccillo, D. 2018, Deep Learning Identifies High-z Galaxies in a Central Blue Nugget Phase in a Characteristic Mass Range 11

Hui, J., Aragon, M., Cui, X., & Flegal, J. M. 2018, A machine learning approach to galaxy-LSS classification - I. Imprints on halo merger trees 9

Kohavi, R. 1995, A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection 36

Krakowski, T., Małek, K., Bilicki, M., Pollo, A., Kurcz, A., & Krupa, M. 2016, Machine-learning identification of galaxies in the WISE SuperCOSMOS all-sky catalogue 9

Liu, Q. Z., van Paradijs, J., & van den Heuvel, E. P. J. 2006, Catalogue of high-mass X-ray binaries in the Galaxy (4th edition) 1

Mahabal, A., Sheth, K., Gieseke, F., Pai, A., Djorgovski, S. G., Drake, A., Graham, M., & the CSS/CRTS/PTF Collaboration. 2017, Deep-Learnt Classification of Light Curves 11, 33, 34

Maíz Apellániz, J., Sota, A., Morrell, N. I., Barbá, R. H., Walborn, N. R., Alfaro, E. J., Gamen, R. C., Arias, J. I., & Gallego Calvente, A. T. 2013, First whole-sky results from the Galactic O-Star Spectroscopic Survey 20, 22

Maíz Apellániz, J., Sota, A., Walborn, N. R., Alfaro, E. J., Barbá, R. H., Morrell, N. I., Gamen, R. C., & Arias, J. I. 2011, The Galactic O-star spectroscopic survey (GOSSS) 20, 22

Maravelias, G. 2014, Investigation of the High-Mass X-ray Binary populations in the Small Magellanic Cloud (PhD Thesis) xiiixiii, 26, 28, 30

Maravelias, G., Kraus, M., Cidale, L. S., Borges Fernand es, M., Arias, M. L., Curé, M., & Vasilopoulos, G. 2018, Resolving the kinematics of the discs around Galactic B[e] supergiants 63

Maravelias, G., Zezas, A., Antoniou, V., & Hatzidimitriou, D. 2014, Optical spectra of five new Be/X-ray binaries in the Small Magellanic Cloud and the link of the supergiant B[e] star LHA 115-S 18 with an X-ray source xiiixiii, 23, 24, 57

Naul, B., Bloom, J. S., Pérez, F., & van der Walt, S. 2018, A recurrent neural network for classification of unevenly sampled variable stars 11

Navarro, S. G., Corradi, R. L. M., & Mampaso, A. 2012a, Automatic spectral classification of stellar spectra with low signal-to-noise ratio using artificial neural networks 7

—. 2012b, Automatic spectral classification of stellar spectra with low signal-to-noise ratio using artificial neural networks 9

Okazaki, A. T. 1997, On the confinement of one-armed oscillations in discs of Be stars. 6

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. 2011, Scikit-learn: Machine Learning in Python 29

Pesenson, M. Z., Pesenson, I. Z., & McCollum, B. 2010, The Data Big Bang and the Expanding Digital Universe: High-Dimensional, Complex and Massive Data Sets in an Inflationary Epoch 8

Porter, J. M. & Rivinius, T. 2003, Classical Be Stars 4, 23

Pudil, P., Novovičová, J., & Kittler, J. 1994, Floating search methods in feature selection [LINK] 43

Reig, P. 2011, Be/X-ray binaries 1, 4

Reig, P. & Zezas, A. 2014, Disc-loss episode in the Be shell optical counterpart to the high-mass X-ray binary IGR J21343+4738 23

Reis, I. & Baron, D. 2019, PRF: Probabilistic Random Forest, Astrophysics Source Code Library 16, 19

Rivinius, T., Baade, D., Townsend, R. H. D., Carciofi, A. C., & Štefl, S. 2013, Variable rotational line broadening in the Be star Achernar 4

Simón-Díaz, S., Castro, N., Garcia, M., Herrero, A., & Markova, N. 2011, The IACOB spectroscopic database of Northern Galactic OB stars 20, 22

Simón-Díaz, S., Negueruela, I., Maíz Apellániz, J., Castro, N., Herrero, A., Garcia, M., Pérez-Prieto, J. A., Caon, N., Alacid, J. M., Camacho, I., Dorda, R., Godart, M., González-Fernández, C., Holgado, G., & Rübke, K. 2015, The IACOB spectroscopic database:

recent updates and first data release 20, 22

Tauris, T. M. & van den Heuvel, E. P. J. 2006, Formation and evolution of compact stellar X-ray sources 3, 5

Townsend, R. H. D., Owocki, S. P., & Howarth, I. D. 2004, Be-star rotation: how close to critical? 4

Vapnik, V. N. 1979, Estimation of Dependences Based on Empirical Data [in Russian] 9

Vasconcellos, E. C., de Carvalho, R. R., Gal, R. R., LaBarbera, F. L., Capelato, H. V., Frago Campos Velho, H., Trevisan, M., & Ruiz, R. S. R. 2011, Decision Tree Classifiers for Star/Galaxy Separation 15, 17, 18

Wadadekar, Y. 2005, Estimating Photometric Redshifts Using Support Vector Machines 9

Walborn, N. R. & Fitzpatrick, E. L. 1990, Contemporary optical spectral classification of the OB stars - A digital atlas 20, 23

York, D. G. & Adelman. 2000, The Sloan Digital Sky Survey: Technical Summary 8

Zhao, G., Zhao, Y., Chu, Y., Jing, Y., & Deng, L. 2012, LAMOST Spectral Survey 8

Declaration

I herewith declare that I have produced this paper without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This paper has not previously been presented in identical or similar form to any examination board.

The thesis work was conducted from 09/2018 to 09/2019 under the supervision of Prof. Andrea Zezas and Dr. Grigori Maravelia at the University of Crete-Department of Physics.

Heraklio, 27/09/2019