

The Department of Computer Science

**Finding Radio Pulsars**

CIS3140 CW2 – Project Report

Student: Emanuel Marques Lourenco

Supervisor: Dr Robert Lyon

Date:29/04/2020

*This Report is submitted in partial fulfilment of the requirements for the BSc Honours Computer Science Degree at Edge Hill University.*

Abstract

Radio pulsar surveys currently output too many pulsar candidates for human inspection, therefore machine learning approaches have been attempted and implemented in order to reduce the number of candidates by very significant amounts. In this project, nine features were extracted from a radio telescope dataset and compared how six different machine learning algorithms performed with said features across different settings using a supervised approach, as a binary problem (classifying candidates as pulsars or non-pulsars). The results indicate that some of the features extra are suitable whereas others are not when applied to the algorithms. Furthermore, it is also identified that either SVM or Random Forest are the most suitable algorithms for the given problem, depending on which measurement is considered most important.

Contents

[Abstract 1](#_Toc39220905)

[Chapter 1: introduction 5](#_Toc39220906)

[1.1 Introduction 5](#_Toc39220907)

[1.2 Aim and Objectives 7](#_Toc39220908)

[1.3 Scope 8](#_Toc39220909)

[1.4 Report Structure 8](#_Toc39220910)

[Chapter 2: Related Work 9](#_Toc39220911)

[Chapter 3: Methods and Dataset 12](#_Toc39220912)

[3.1 Machine Learning Methodologies 12](#_Toc39220913)

[3.2 Classifiers 13](#_Toc39220914)

[3.3 Dataset 14](#_Toc39220915)

[3.4 Attributes from Dataset 15](#_Toc39220916)

[Chapter 4: Features 17](#_Toc39220917)

[4.1 Feature Extraction 17](#_Toc39220918)

[4.2 Feature Selection 19](#_Toc39220919)

[4.3 Imbalance Treatment 21](#_Toc39220920)

[4.3 Feature Scaling 22](#_Toc39220921)

[Chapter 5: Classification Performance 23](#_Toc39220922)

[5.1 Evaluation 23](#_Toc39220923)

[5.2 Cross-Validation 24](#_Toc39220924)

[5.3 Results 25](#_Toc39220925)

[**Baseline** 25](#_Toc39220926)

[**Implementation of algorithms** 26](#_Toc39220927)

[Chapter 6: Conclusions 29](#_Toc39220928)

[Summary of the Project 29](#_Toc39220929)

[Critical evaluation 31](#_Toc39220930)

[Future work 32](#_Toc39220931)

[Conclusion 33](#_Toc39220932)

[References 34](#_Toc39220933)

List of Tables

[Table 4.1 - Features Ranked by Information Gain 19](#_Toc39220934)

[Table 4.2 - Features Ranked by Gain Ratio 20](#_Toc39220935)

[Table 4.3 - Comparison Between Normalised and Standardised Data 22](#_Toc39220936)

[Table 5.1 - Baseline results across 6 different algorithms 25](#_Toc39220937)

[Table 5.2 - Performance results across 4 different settings for 6 different algorithms (best values displayed in bold) 26](#_Toc39220938)

[Table 5.3 - Table 5.2 - Performance results across four different kernels of the SVM classifier (best values displayed in bold) 28](#_Toc39220939)

List of Figures

[Figure 1.1 - Model Sketch of a Radio Pulsar. Figure taken from Levin (2012) 5](#_Toc39220940)

[Figure 3.1-Centred Profile of a Pulsar at a) and Non-Pulsar at b) 15](#_Toc39220941)

[Figure 3.2 - DM Curve of a Pulsar at a) and Non-Pulsar 16](#_Toc39220942)

[Figure 5.1 - Confusion Matrix. Figure Taken from Raschka (2015) 23](#_Toc39220943)

# introduction

## 1.1 Introduction

Radio Pulsars are a special type of Neutron Star that rapidly rotate, with spin periods that are known to range from 1.396 milliseconds to 8.51 seconds (Lyon, 2016). Their emission of radio waves can be detected from earth as ‘pulses’ (Devine, Goseva-Popstojanova and McLaughlin, 2016) with regular periods. They have been described as extraordinary natural laboratories (Ng et al., 2015; Lyon, 2016) as they serve as probes of their environment and their study leads to a better understanding of a wide variety of fundamental physics and astronomy questions (Keith et al., 2010; Ng et al., 2015). Thus, even though more than 2500 Pulsars have been found, their discovery and study is still extremely important (Lyon, 2016). A sketch of a Radio Pulsar and its magnetosphere can be seen in figure 1.1.

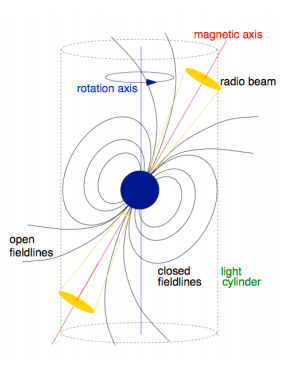


Figure 1.1 - Model Sketch of a Radio Pulsar. Figure taken from Levin (2012)

Eatough et al. (2010) describes how the discovery of Pulsars can be separated into two distinct stages: The first stage involves the signal processing of survey data acquired by radio telescopes which output can produce several million candidates (Morello et al., 2014). The vast majority of candidates however, originate from radio-frequency interference (RFI) produced by humans, or other various forms of noise, hiding real potential Pulsars (Lyon, 2016). The second stage requires manual visual inspection of the Pulsar candidates which is both time consuming (Eatough et al., 2010; Lyon, 2016) and inefficient given the millions of Pulsar candidates produced by surveys. As next generation instruments produce significantly larger data sets (Devine et al. 2016; Lyon, 2016) it makes it a necessity to automate these processes and filter out the majority of candidates in order to reduce the dependence on human input.

Some have turned to machine learning. As a subfield of artificial intelligence, machine learning offers a more efficient approach to the development of predictive models, by allowing self-learning algorithms to gain knowledge from data in order to make predictions (Raschka, 2015) and has been suggested by Devine, Goseva-Popstojanova and McLaughlin (2016) that it is becoming the new standard for pulsar classification. Having already addressed a variety of problems in the past for a wide range of fields such as e-mail spam filtering, text and voice recognition and search engines (Raschka, 2015) it can then be applied to the candidate problem as a binary approach (classifying signals as either Pulsars or non-pulsars) or as a multiclass classification approach by dividing Pulsars into different classes: Pulsars, very bright Pulsars, Rotating Radio Transients (RRAT) like Pulsars, Fast Radio Bursts (FRBs) and non-pulsars (Devine, Goseva-Popstojanova and McLaughlin, 2016). Machine learning approaches are being used to tackle the growing rate of Pulsar candidates that are gathered from radio telescope surveys, such as these Pulsar candidates can be correctly be predicted by the system as Pulsars or non-pulsars.

## 1.2 Aim and Objectives

**Aim**

The aim of this project is to investigate and improve the performance of machine learning algorithms to classify Pulsar candidates using real data collected by a radio telescope.

**Objectives**

1. Understand the basic astrophysics knowledge

* Identify how Pulsars are currently discovered via a literature review of recent academic papers/journals and books.
* Learn how Pulsar searches with radio telescopes are conducted.

1. Learn the classification challenges that relate to the Pulsar candidate selection problem

* Identify Pulsar candidate features and how they differ (or do not) from other signal sources such as RFI and noise.
* Find relevant Pulsar features suited for machine learning applications via a literature review of recent academic papers/journals and books and/or feature extraction techniques.
* Understand how to tackle the imbalance problem of the dataset.

1. Apply machine learning algorithms to a dataset

* Use relevant pre-processing techniques to prepare the dataset such as feature extraction, scaling and selection.
* Train an algorithm on the data.

1. Evaluate algorithmic performance

* Find which evaluation metrics are best suited for the problem.
* Apply evaluation metrics to the algorithms and analyse performance using cross-validation methods.

1. Improve algorithm performance

* Apply hyper parameter optimisation techniques.

## 1.3 Scope

The focus of this project is in the identification of features that can be extracted from the dataset and in applying said features to a different range of classifiers. Comparing these results, it will be possible to conclude whether the features were relevant and which classifier is producing the best results for the given problem. It will also focus on attempting to improve these performances by testing different settings/combinations across the classifiers

Some issues or problems that were identified in literature were not covered as it was beyond the scope of the project such as the fact that supervised and unsupervised learning methods cannot adapt to RFI and environment once trained.

## 1.4 Report Structure

The remainder of this report is structured as follows.

Chapter 2 covers the different machine learning approaches that have been found in literature to attempt to solve the candidate selection problem.

Chapter 3 describes the different methods that exist within machine learning and which methods will be used in this report. It also covers what dataset and specific attributes that will be used.

Chapter 4 is focused on the features to be used with the machine learning algorithms. It covers which features have been extracted using the dataset, feature selection using specific feature relevance metrics, a possible solution for the imbalance nature of the dataset and feature scaling.

Chapter 5 underlines what evaluation metrics will be used to test the performance of the classifiers and provides the results and discussion of their performance.

Chapter 6 concludes the report by giving a summary of the project, a critical evaluation and possible future work based on what was achieved by the project, followed by a final conclusion.

# Related Work

Several machine learning approaches can already be found in literature:

Eatough et al. (2010) used 2 different Artificial Neural Network (ANN) models, a model which used a set of 8 features and another with 4 more features, 12 in total including the pulse profile SNR and pulse profile width. When applied to the test data, the ANN model that used 8 features successfully labelled 92% of the pulsars, this increased to 93% for the ANN model that used 12 features instead. Their ANN model was successful in discovering a new Pulsar (Devine, Goseva-Popstojanova and McLaughlin, 2016).

More recently, an ANN was again used by Morello et al. (2014) to create a new binary classification model called SPINN. Here 6 features were used instead, to train the model, as it was identified that an excessive number of features would result in a decline in performance. The dataset used to train and evaluate the classifier contained all known Pulsars (1196 at the time) from the HTRU all-sky survey and also 89,996 non-pulsars, when cross-validated, it was able to reduce the number of candidates requiring human inspection by a factor of approximately 150 while still identifying all potential Pulsars. It is possible that Morello et al. (2014) approach performed significantly better compared to Eatough et al. (2010) due to a reduction in features and their nature or possibly due to the implementation of a much larger dataset to train the classifier since it contained more than 90,000 compared against nearly 2,000.

Zhu et al. (2014) created a slightly different approach as they implemented a two-layer image recognition algorithm called Pulsar Image-based Classification System (PICS), that was designed to emulate a human expert’s visual identification process. The first layer was comprised of a combination of ANN’s, CNN’s (convolutional neural networks) and SVM’s (support vector machines), the results from the first layer were then passed on to the second one which used a logistic regression model to output a final classification. To train their algorithm they used 4 different features: the summed profile, time versus phase plot, frequency versus phase plot and the DM curve which were chosen based on the fact these are the four main subplots that human experts look at when classifying candidates. Their training dataset consisted of 3756 labelled PALFA candidates which was then tested on a complete new different dataset from a different survey to determine whether their approach could be generalised to other surveys. They achieved promising results being able to reduce the speed of the classification process by a human expert by a factor of 100 which the authors claim could be improved by gathering more training data. It was therefore integrated to the PALFA survey pipeline, where it has discovered six new pulsars.

Devine, Goseva-Popstojanova and McLaughlin (2016) used a supervised learning approach and tested six different types of machine learning algorithms: An Artificial Neural Network (ANN), a support vector machine (SVM), a standard tree learner (J48), hybrid rule-and-tree learner (PART), and random forest as an ensemble tree learner and applied those for both binary and multiple class classification. They concluded from their results that random forest could be a potential good classifier for this problem as it provided the highest F-measure and good recall values when imbalance treatments were used (See Devine, Goseva-Popstojanova and McLaughlin, 2016, table 3.). Even though a much smaller dataset was used (10,000 instances), a good point of their approach was that 6 different classifiers were used, over four different settings: one with no imbalance treatment on the dataset, and 3 other tries each one with a different imbalance treatment. This goes accordingly to the no-free lunch theorem (Wolpert, 2002) implying that no classifier has the better performance across every problem, therefore it is useful to try different classifiers and approaches to see what is best suited for the problem at hand.

Lyon et al. (2016) have also created a new classifier (GH-VFDT) designed for online operation using a new set of 8 features including: The mean, standard deviation, excess kurtosis and skewness of both the integrated pulse profile (See Lyon et. al, 2016, Figure 1.c.) and DM-SNR curve (See Lyon et. al, 2016, Figure 1.d.). These features were created not only to maximise the separation between noise and actual Pulsars, but also to be survey-independent in order to solve a problem identified by the same authors, who highlight that many existing features used for candidate classification are “implementation dependent”, thus, difficult to build upon the implementation of others. This can be evidenced by the related works discussed thus far, as all of them use a different set of features each time and it is even stated by Morello et al. (2014: 1655) “Note that the usefulness of this feature is dependent on the RFI landscape at the place and even time of observation, and its portability to other surveys is unknown”. Zhu et al. (2014) mentioned the aim of a system that has the ability to adaptively train an algorithm as more data is available, since it would allow the algorithm to adapt as the RFI environments change over time. As supervised and unsupervised learning methods are unable to adapt once trained (Lyon, 2016), this new approach aimed to develop a solution that fills that requirement by using an incremental stream classifier. After testing their new features extracted across 5 different classifiers (4 standard classifiers, and their own GH-VFDT), the results not only suggest that these are suitable survey independent features for the candidate selection problem, but also that even though it was getting lower recall rates on two of the datasets the GH-VFDT was able to outperform the standard static classifiers on relevant evaluation metrics specifically on the largest dataset which is of importance since the performance of stream algorithms improves as more examples are fed into the classifier (Lyon et al., 2016). Using this approach more than 20 new Pulsars have been discovered.

It has been established that machine learning algorithms have already been tested and even implemented leading to the discovery of new Pulsars and will most likely be the way forward as the size of pulsar candidates increases over time. Whilst most approaches covered in this literature review have used supervised learning, Lyon (2016) identifies how these methods cannot adapt to change automatically which will be a requirement for the new radio telescopes such as the SKA, and other approaches must also be explored to deal with this issue.

# Methods and Dataset

## 3.1 Machine Learning Methodologies

Han, Kamber and Pei (2012) identify how machine learning has different methodologies which will differ depending on what type of dataset is available to tackle the problem: Supervised learning is used when the dataset is fully labelled. In other words, for this project, a dataset where every single instance has been correctly labelled as pulsars or non-pulsars can be used for supervised learning methods. However, this is not always the case. For problems where there are no available labelled datasets an unsupervised methodology would have to be used. Instead of classifying the data (predicting them as pulsars or non-pulsars for example), this methodology tries to cluster it into different groups. A potential use of the unsupervised approach for the candidate problem was identified by Eatough et al. (2010) as it could be used to group different types of pulsars, and even different forms of non-pulsars. Semi-supervised learning uses both labelled and unlabelled data when training a model. The labelled data is used to define the different classes and the unlabelled data can be used to refine the boundaries between them (Han, Kamber and Pei, 2012). Lyon (2016) states how this approach is mostly used when there is a lot of unlabelled data that could be used to possibly improve the model or when limited labelled data is available. Lyon (2016) identifies several other machine learning methodologies which are suitable for the candidate selection problem depending on different requirements (See Lyon, 2016, Table 4.3). This project will apply supervised learning since the datasets that were used to train the classifiers, were already be labelled as a binary problem (classifying candidates as either pulsars or non-pulsars).

Within supervised learning there are a wide range of different classifiers available, some of which have been used more often than others for the candidate problem. Artificial Neural Networks have been used more than once to tackle the pulsar candidate selection problem (Eatough et al. 2010; Morello et al. 2014; Devine, Goseva- Popstojanova andMcLaughlin, 2016). It is however, unclear which classifier will have the best performance for any given problem as the no-free lunch theorem (Wolpert, 2002) indicates that supervised classifiers perform equally well across all possible problems. This means that no classifier will be the most suitable for every problem, therefore, several classifiers were evaluated and compared against each other to ascertain which was the most suitable for this problem.

## 3.2 Classifiers

Six different classifiers/algorithms were used to predict whether candidates were pulsars or non-pulsars: Support Vector Machine (SVM), Naïve Bayes, Decision Trees, Multilayer Perceptron (MLP), Random Forest and Logistic Regression.

Han, Kamber and Pei (2012) define support vector machines as an algorithm that creates a hyperplane (decision boundary) using training tuples (support vectors) that separate the classes of the instances. This type of algorithm which can be used to create accurate models by creating complex non-linear decision boundaries and have been used for handwritten digit recognition.

Naïve Bayes is a probabilistic algorithm based on the Bayes theorem which calculates the posterior probability using the prior probability of some evidence (Raschka, 2014), the naïve part comes from the fact that this algorithm assumes that the features do not influence each other and each one has an equal impact on the classification outcome. Its limitation is that it can have poor performance when strong violations of its naïve assumption occur (Han, Kamber and Pei, 2012).

Multilayer perceptron is a type of artificial neural network (ANN) algorithm which has been shown in the previous chapter as a type of algorithm used many times to tackle the candidate classification problem. The term multilayer comes from the fact that each layer of the algorithm serves as input for the next one. It iteratively learns a set of weights used to make class predictions (Han, Kamber and Pei, 2012).

Logistic Regression is a discriminative probabilistic model that combines a linear decision boundary with logistic calibration which can output decision boundaries considerably different from generative classifiers (Flach, 2012). Even though this classifier has not been seen in literature applied for this problem, the decision to still use it came from the fact that it works differently from the other classifiers covered in this project, therefore it could lead to interesting results. This a novel contribution made by this work.

Decision tree algorithms use a ‘divide-and-conquer’ strategy that try to determine what criteria best divide the dataset into separate classes. Many different decision trees have been created, the one implemented here is called CART which uses the Gini index as impurity measure (Flach, 2012).

The last classifier used was Random forest. An ensemble algorithm that uses a set number of different decision trees: each individual tree is trained on a randomly selected number of features from the original dataset, and to classify new instances each tree individually predicts its class, then through majority vote the prevalent class is selected (Breiman, 2001; Raschka, 2015; Gutiérrez-Esparza, Vallejo-Allende and Hernández-Torruco, 2019).

## 3.3 Dataset

Eatough et al. (2010) highlights how important it is to use datasets that include pulsars with a wide variety of range and properties, as training the classifiers on a specific set of pulsars will limit the discovery of atypical or unusual phenomena.

The dataset that was used to train the classifiers was the HTRU Medium Latitude training data which was used by Morello et al. (2014) to train their SPINN classifier. It can be found at: <https://astronomy.swin.edu.au/~vmorello/>. It includes 1,196 known pulsar candidates and 89,996 non-pulsar candidates. To parse the data into arrays that can be used in python, a candidate class written in python was provided by the same authors along with the dataset, which was imported into this implementation.

One of the challenges faced was that the entire dataset was quite big, being approximately 2GB in size and was also encoded in hexadecimal strings, therefore, to reduce the loading and converting time, the different attributes of the candidates that were used to create the features for the classifier, were converted and stored into csv files instead, massively reducing the time it takes to have the attributes readily available to manipulate and test.

## 3.4 Attributes from Dataset

In order to extract features to train the classifiers, two different attributes from the original dataset were used: The Integrated pulse profile, and the DM Curve. Initially, the two-dimensional array of sub-integrations had been averaged to form a one-dimensional array to also extract features from, however it was found that this is the equivalent of the candidate profile, so in the end its only use was to validate the dataset.

**Profiles**

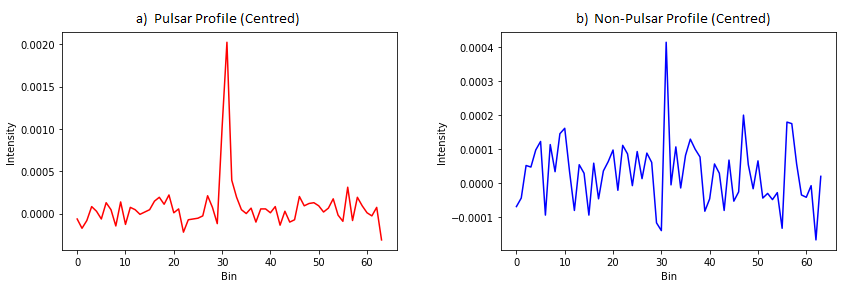


Figure 3.1-Centred Profile of a Pulsar at a) and Non-Pulsar at b)

The integrated pulse profile can be defined as the signal averaged across all observed frequencies and time (Lyon, 2016) which describes the pulse intensity across a number of phase bins. Figure 3.1 shows plots for a pulsar integrated profile in a) that has had its peak centred, and in b) a plot of a non-pulsar integrated profile with its peak also centred.

**DM Curve**

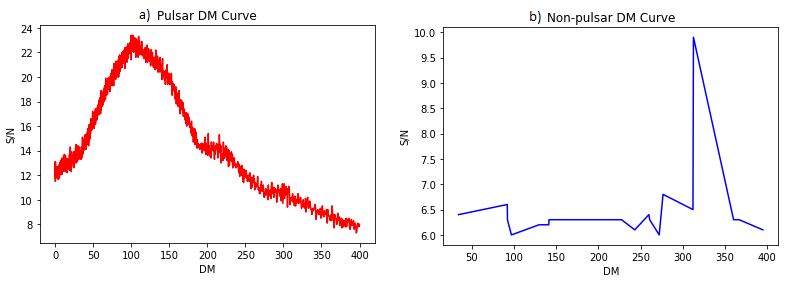


Figure 3.2 - DM Curve of a Pulsar at a) and Non-Pulsar

Pulsars signals are dispersed since they are affected by travel through the interstellar medium. Whereas RFI originating from earth does not experience such dispersive effects. This means that when checking signals for signs of dispersion as a means to determine if they originated from deep space, we compute something called the Dispersion Measure (DM). RFI and local Earth-based signals should have a DM peaking at zero when measured, while pulsars on the other hand should peak at a DM value bigger than zero (Lyon, 2016). To find the DM we must search for dispersive effects by correcting for potential dispersion via a series of trials that produce a DM curve (Morello et al., 2014). The DM curve corresponds to the horizontal flattened matrix of a period-DM plane (Lyon, 2016) which aims to find the optimal period and DM values, where the S/N peaks. Figure 3.2 shows plots for the DM curve of a pulsar in a), and for the DM curve of a non-pulsar candidate in b).

# Features

Flach (2012:13) states that “machine learning is all about using the right features to build the right models that achieve the right tasks”. Classifiers take the features of each candidate and based on this, decide which class they are most likely to belong to (Morello et al., 2014). Therefore, it is important to extract relevant features from the given attributes of the dataset with which we can use statistical distributions to better separate the pulsars from non-pulsar candidates. It is also identified by several authors that an excessive number of features results in a decline in performance due to a problem known as the ‘curse of dimensionality’ (Bishop, 2006; Morello et al., 2014; Raschka, 2015; Lyon et al., 2016) hence it is important to limit the set of features to the most relevant ones.

## 4.1 Feature Extraction

This section covers what features have been extracted from the different dataset attributes and the thought behind those decisions – i.e. why they could possibly be relevant or suited to separate pulsars from non-pulsars

**Peak and Lowest Value from Profile**

When observing the sub-integrations of a real Pulsar, it is possible to discern a line throughout its observation index, unlike the sub-integrations of the non-pulsar example, as this seems to be more dispersed and scattered. Therefore, a reasonable assumption to make is that, when we average the observation index across the same bins (profile), the pulsar values should give a significant difference between the peak and the lowest value whereas perhaps a good amount of non-pulsar candidates will not have such a clear discrepancy.

**Product of S/N peak with its DM value**

The maximum S/N of a signal originating from earth will be achieved at a DM value of zero (Lyon, 2016), therefore by multiplying the peak of the S/N to its DM value from the DM curve array, it could potentially lead to a good feature to be used with the classifiers as it will at least identify some clear RFI candidates.

**Product of S/N lowest value with its DM value**

There is no evidence that suggests that multiplying the lowest value of signal to noise against its ratio could give an indicator of being a pulsar or a non-pulsar, therefore this is just experimental.

**Geometric Mean of DM curve**

According to (Chen and Chen, 2005), the geometric mean of positive numbers can be defined by:

Computing the geometric mean of the DM curve might give the classifiers some important information about the candidates to better differentiate the pulsar from non-pulsar candidates, as it has not been found in literature, it could be of interest to test how relevant this feature might be. This is another novel contribution made in this work.

**Mean, Standard Deviation, Skewness and Kurtosis of the Profile**

It has been found from the literature that the mean, standard deviation, skewness and kurtosis of the integrated profile are suitable features to be used for the candidate problem (Lyon et al. 2016), therefore these will be added to the new untested features.

In total, 9 different features have been extracted: Maximum value (peak), lowest value, mean, standard deviation, skewness and kurtosis of the profile, product of S/N peak with its DM value, product of S/N lowest value with its DM value and the geometric mean of the DM curve.

## 4.2 Feature Selection

It is possible that the model created using the set of features does not generalise well to real data by fitting the parameters too closely to the training dataset, which is known as overfitting (Raschka, 2015). To avoid this problem, reduction of dimensionality of the model can be applied through feature selection. There are many methods to evaluate how relevant each feature is (Han, Kamber and Pei, 2012) which would help with the decision of what features to be selected. The two different methods that were considered to assess feature relevance were information gain and gain ratio using the WEKA data mining tool (can be found at: http://www.cs.waikato.ac.nz/ml/weka). The feature with the most information gain in a decision tree, would be the one that when split into tuples, minimises the information needed to classify the resulting tuples in the following partitions. This produces the least ‘impurity’ in these partitions (Han, Kamber and Pei, 2012), however the same author indicates that this measure can be biased towards many outcomes. The gain ratio tries to overcome this by applying a kind of normalisation to the information gain. Table 4.1 presents the ranking of the features by their information gain from most important to least, and the same can be found in table 4.2 but using the gain ratio measurement instead.

Table 4.1 - Features Ranked by Information Gain

|  |  |
| --- | --- |
| Feature | Information Gain |
| Profile Skewness | 0.07833 |
| Profile Kurtosis | 0.06596 |
| Profile Peak | 0.06288 |
| Profile Standard Deviation | 0.0542 |
| Geometric Mean of DM Curve | 0.04831 |
| Product of S/N peak with its DM value | 0.03534 |
| Profile Mean | 0.03135 |
| Product of S/N lowest value with its DM value | 0.00975 |
| Profile Lowest Value | 0.00161 |

Table 4.2 - Features Ranked by Gain Ratio

|  |  |
| --- | --- |
| Feature | Gain Ratio |
| Profile Standard Deviation | 0.08989 |
| Profile Skewness | 0.06313 |
| Geometric Mean of DM Curve | 0.04382 |
| Profile Kurtosis | 0.04186 |
| Profile Peak Value | 0.04149 |
| Product of S/N peak with its DM value | 0.02171 |
| Profile Mean | 0.01505 |
| Product of S/N lowest value with its DM value | 0.0054 |
| Profile Lowest Value | 0.00112 |

It can be seen that both the product of S/N lowest value with its DM value and the profile lowest value scored the lowest for both measurements by a very significant amount, therefore it is reasonable to assume that by removing these two features will result in an increase in overall performance. The results further suggest that at least 3 of the extracted features that were not taken from literature, are suitable for this problem: The geometric mean of the DM curve, the profile peak value and the product of S/N peak with its DM value.

## 4.3 Imbalance Treatment

It is identified by several authors (Morello et al., 2014; Devine, Goseva-Popstojanova and McLaughlin, 2016; Lyon, 2016) that one of the challenges of implementing machine learning algorithms to the Pulsar candidate problem, is that most of the data will originate from noise, therefore there can be a tendency for some algorithms to always predict candidates as non-pulsars. This is because there are a great deal more non-pulsar candidates than pulsars and, in this case, algorithms can achieve accurate performance just by always identifying candidates as non-pulsars

To deal with this imbalance of the dataset, SMOTE (Synthetic Minority Oversampling Technique) was applied to the datasets in order to try and increase the performance of the algorithms. SMOTE oversamples the minority class (in this case the pulsar candidates) but instead of simply replicating the candidates, it creates “synthetic” examples that have small random changes to their features (Chawla et al., 2002). SMOTE forces the decision region of the minority class to become more general (Chawla et al., 2002) and therefore, avoid the problem of overfitting by oversampling the minority class.

## 4.3 Feature Scaling

Feature scaling is an important step in pre-processing the data as most machine learning algorithms perform better with features on the same scale (Raschka, 2015). The author identifies two different approaches to feature scaling: normalisation and standardisation, and states that normalisation often refers to the rescaling of the features to the range of [0,1] whereas standardisation refers to scaling the data so it has a mean of zero and standard deviation of 1, resulting in a normal distribution. Table 4.3 shows a comparison between the two, given a simple dataset of the numbers between 1 and 5:

Table 4.3 - Comparison Between Normalised and Standardised Data

|  |  |  |
| --- | --- | --- |
| **Input** | **Normalised** | **Standardised** |
| 1 | 0 | -1.414214 |
| 2 | 0.25 | -0.707107 |
| 3 | 0.5 | 0 |
| 4 | 0.75 | 0.707107 |
| 5 | 1 | 1.414214 |

When comparing both, Raschka (2015) argues that standardisation can be more practical as it maintains useful information about outliers and makes the algorithm less sensitive to them compared to normalisation therefore, this will be the approach used for this implementation as it was also the feature scaling implemented by Morello et al. (2014).

The procedure to standardise a dataset is expressed by the following equation:

Where *μ* and *σ* are the mean and standard deviation of the dataset respectively.

# Classification Performance

## 5.1 Evaluation

To calculate the different metrics used for evaluation of the classifiers, it is necessary to first report the counts of true positives, true negatives, false positives and false negative predictions output by the classifiers. This can be reported using a confusion matrix which is in figure 5.1:

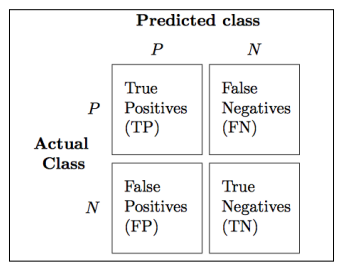


Figure 5.1 - Confusion Matrix. Figure Taken from Raschka (2015)

For this problem, True Positives (TP) correspond to candidates correctly predicted or classified as pulsars, True Negatives (TN) to those correctly classified as not pulsars, False Positives (FP) to incorrectly classified as pulsars and False Negatives (FN) to incorrectly classified as not pulsars. Using these values, it is then possible to calculate the relevant metrics to evaluate the classifiers performance which are defined by Raschka (2015) and Lyon (2016) as followed:

**Accuracy** - Provides general information about how many samples are misclassified. Defined by:

As it has been mentioned before, this problem deals with imbalanced data hence the measure of accuracy can be misleading. For example, if only 1 in 1000 candidates is an actual pulsar, algorithms can achieve 99.9% accuracy simply by always classifying candidates as non-pulsars, which on paper seems like an incredibly good performance but is meaningless as it is missing every pulsar, which is the sole aim of implementing algorithms.

Therefore, another measure called G-mean is also used which is insensitive to the distribution of pulsars and non-pulsars (Lyon et al., 2016)

**G-Mean** – Metric used for imbalanced datasets that calculates the ratio of positive and negative accuracy. Defined by:

**Precision** - What fraction of predicted positives were correct. Defined by:

**Recall or true positive rate** – What fraction of positives was predicted as such. Defined by:

**False positive rate (FPR)**– Fraction of negative instances that were predicted as positive. Defined by:

**F-Score** - Combination of precision and recall. Defined by:

## 5.2 Cross-Validation

To evaluate the classification performance, a fivefold cross-validation procedure was implemented. This procedure is done by randomly partitioning a dataset into five subsets of equal size, four are used to train the algorithm and one is held out as the test set. This is repeated five times to ensure that the performance is always tested on each individual partition exactly once. The performance of each sub partition is then averaged to achieve the overall classification performance. By implementing cross-validation on the dataset it is possible to achieve low bias and variance estimate of the model performance (Han, Kamber and Pei, 2012).

## 5.3 Results

### **Baseline**

Multiple features were created based on the different attributes of the candidates, to evaluate how well these features work with the classifiers, first a baseline was established. This consisted of feeding the profiles of the candidates, which were resized from the original 64 to 32 bins to the classifiers serving as features.

Table 5.1 - Baseline results across 6 different algorithms

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Classifier | F-Score | G-Mean | Precision | Recall | Accuracy | FPR |
| SVM | 0 | 0 | 0 | 0 | 0.9869 | 0 |
| Naïve Bayes | 0.2267 | 0.4120 | 0.3412 | 0.1706 | 0.9848 | 0.0044 |
| Random Forest | 0.7854 | 0.8391 | 0.8783 | 0.7111 | 0.9949 | 0.0013 |
| MLP | 0 | 0 | 0 | 0 | 0.9869 | 0 |
| Decision Tree (Cart) | 0.6330 | 0.8141 | 0.6012 | 0.6689 | 0.9898 | 0.0059 |
| Logistic Regression | 0 | 0 | 0 | 0 | 0.9869 | 0 |

The results in table 5.1 show that using the raw data from the radio telescopes to train the machine learning classifiers are not suitable, three algorithms (SVM, Multilayer Perceptron and Logistic Regression) simply classify every candidate as non-pulsar, and Naïve Bayes presents very low measurements. The algorithms Random forest and Decision Tree were able to some degree identify pulsars as they have significantly higher scores for all measurements. The next section shows how by performing feature extraction, feature selection and imbalance treatments on the datasets, these results can be improved.

### **Implementation of algorithms**

The tests include the implementation of the 6 different algorithms across 4 different settings: Including all features extracted, using feature selection to remove features that have low information gain and gain ratio (features removed were the profile lowest value and the Product of S/N lowest value with its DM value) , using all features extracted with SMOTE imbalance treatment and using feature selection with SMOTE imbalance treatment. The results are showed in table 5.2

Table 5.2 - Performance results across 4 different settings for 6 different algorithms (best values displayed in bold)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Setting | Classifier | F-Score | G-Mean | Precision | Recall | Accuracy | FPR |
| All Features | SVM | 0.8616 | 0.8944 | 0.9231 | 0.8090 | 0.9966 | 0.0009 |
| Naïve Bayes | 0.7494 | 0.8690 | 0.7373 | 0.7632 | 0.9933 | 0.0036 |
| Random Forest | **0.9414** | 0.9510 | **0.9690** | 0.9155 | **0.9985** | **0.0004** |
| MLP | 0.9379 | 0.9586 | 0.9456 | 0.9304 | 0.9984 | 0.0007 |
| Decision Tree (Cart) | 0.9019 | 0.9471 | 0.8961 | 0.9079 | 0.9974 | 0.0014 |
| Logistic Regression | 0.8479 | 0.8808 | 0.9236 | 0.7842 | 0.9963 | 0.0009 |
|  |  |  |  |  |  |  |  |
| Feature Selection | SVM | 0.8658 | 0.9013 | 0.9159 | 0.8213 | 0.9967 | 0.0010 |
| Naïve Bayes | 0.6956 | 0.8681 | 0.6410 | 0.7616 | 0.9913 | 0.0057 |
| Random Forest | 0.9381 | 0.9490 | 0.9664 | 0.9115 | 0.9984 | **0.0004** |
| MLP | 0.9364 | 0.9562 | 0.9481 | 0.9256 | 0.9984 | 0.0007 |
| Decision Tree (Cart) | 0.9043 | 0.9416 | 0.9118 | 0.8972 | 0.9975 | 0.0011 |
| Logistic Regression | 0.8466 | 0.8818 | 0.9184 | 0.7859 | 0.9963 | 0.0009 |
|  |  |  |  |  |  |  |  |
| All Features and SMOTE | SVM | 0.7388 | 0.9859 | 0.5914 | 0.9848 | 0.9909 | 0.0090 |
| Naïve Bayes | 0.7538 | 0.8932 | 0.7079 | 0.8066 | 0.9931 | 0.0044 |
| Random Forest | 0.9221 | 0.9731 | 0.8881 | 0.9590 | 0.9979 | 0.0016 |
| MLP | 0.8975 | 0.9692 | 0.8500 | 0.9515 | 0.9971 | 0.0022 |
| Decision Tree (Cart) | 0.8955 | 0.9638 | 0.8549 | 0.9406 | 0.9971 | 0.0021 |
| Logistic Regression | 0.5591 | 0.9712 | 0.3952 | 0.9556 | 0.9802 | 0.0195 |
|  |  |  |  |  |  |  |  |
| Feature Selection and SMOTE | SVM | 0.7201 | **0.9863** | 0.5675 | **0.9856** | 0.9900 | 0.0100 |
| Naïve Bayes | 0.6807 | 0.8905 | 0.5919 | 0.8016 | 0.9901 | 0.0074 |
| Random Forest | 0.9179 | 0.9727 | 0.8810 | 0.9582 | 0.9978 | 0.0017 |
| MLP | 0.8750 | 0.9752 | 0.8020 | 0.9633 | 0.9964 | 0.0032 |
| Decision Tree (Cart) | 0.8939 | 0.9677 | 0.8455 | 0.9483 | 0.9970 | 0.0023 |
| Logistic Regression | 0.5442 | 0.9721 | 0.3802 | 0.9573 | 0.9790 | 0.0208 |

The difference between using raw data to feed the algorithms, and feature extraction is clear. By using relevant features, the performance of every single algorithm improved drastically, reinforcing the statement by Flach (2012) that highlights how using the right features is what makes machine learning achieve the right tasks.

After applying feature selection and reducing the number of features by two, there was a slight decrease in performance across most measurements. The differences are quite small it, however, this suggests that although their information gain and gain ratio are much lower than the other features, the “product of S/N lowest value with its DM value” and the “profile lowest value” still contributed to the separation of pulsars from non-pulsars by the algorithms. Despite this, feature selection with SMOTE imbalance treatment achieved the highest recall rate with the SVM Classifier.

It can be seen that there is no clear highest performing classifier across all measurements hence the best classifier choice depends on the most desirable performance measurement. If the aim is to get as many pulsars as possible, feature selection with SMOTE seems to be the best choice as it provides the highest G-mean and recall rate. However, the price is lower precision which means there will be a higher number of misclassified non-pulsars as pulsars, and will therefore increase the amount of manual inspection required. If the aim however, is the reduce the amount of manual inspection as much as possible, then the recommended choice would be Random forest using all features as this achieved the highest f-score, precision, accuracy and false positive rate, this setting output will make sure that when the classifier labels a candidate as a pulsar, there is a very high chance that it is in fact one since the false positive rate is extremely low. However, the recall is a great deal lower than the first discussed setting, and it means a lot more pulsars will be missed than SVM with feature selection and SMOTE.

Another observation is that the SMOTE imbalance treatment for both settings (including all features, or with feature selection) increased the recall rate for every single classifier by a significant amount at the cost of lower precision, so it must be taken into consideration what measurement is more important for the given problem. Morello et al. (2014) suggests that recall should be prioritised as pulsars remain rare objects, therefore, it can be argued that the application of the SMOTE treatment is in fact relevant for this given situation.

**Optimisation**

Due to its good performance and highest recall rate, SVM with feature selection and SMOTE seems to be the best suitable for this problem. As an attempt to further improve its performance, different parameters of this classifier were tested, more precisely, different kernels. The four different kernels were: linear, radial basis function (RBF), polynomial and sigmoid. The parameter that was used with SVM previously to be compared against the other algorithms, was the default RBF kernel. The results of the comparison between the four different kernels can be found below in table 5.3.

Table 5.3 - Table 5.2 - Performance results across four different kernels of the SVM classifier (best values displayed in bold)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Setting | SVM Kernel Parameter | F-Score | G-Mean | Precision | Recall | Accuracy | FPR |
| Feature Selection and SMOTE | Linear | 0.5745 | 0.733 | 0.4101 | 0.9597 | 0.9813 | 0.0184 |
| RBF (default) | 0.7201 | **0.9863** | 0.5675 | **0.9856** | 0.9900 | 0.0100 |
| Polynomial | **0.7566** | 0.9632 | **0.6336** | 0.9395 | **0.9921** | **0.0072** |
| Sigmoid | 0.0479 | 0.8063 | 0.0248 | 0.6589 | 0.6563 | 0.3438 |

The default RBF kernel still outputs the better G-mean and recall rates out of the four different settings. The polynomial kernel did have the best F-score, precision, accuracy and false positive rate out of the four. This is still not the best choice, as if the favoured metric is not the recall rate, then the random forest classifier still provides the better performances across the same four metrics (F-score, precision, accuracy and FPR).

# Conclusions

## Summary of the Project

The aim of the project was to investigate and improve the performance of machine learning algorithms to classify Pulsar candidates. To do this the project was broken down into steps or objectives, that each served a purpose or were a requirement of the next. Although not always in the planned order, the aim and objectives have been met at least to some extent.

The first objective included identifying how Pulsars are currently discovered and how radio telescope conduct Pulsar searches. This has been covered in section 1.1: Introduction and section 2.1: Related works.

The second objective included the identification of Pulsar features that are suitable for machine learning applications, and how to tackle the imbalance problem of the dataset. This objective was covered thoroughly in Chapter 4: starting with the extraction of features. An imbalance treatment called SMOTE was also applied to the dataset and its performance was compared against the dataset without the imbalance treatment applied.

The third objective aimed to implement the machine learning algorithms to the dataset. Its first step was covered in chapter 4 as feature selection was applied and they were then assessed as to how well they were suitable for machine learning applications by using metrics used in industry (information gain and gain ratio) and was found that at least some extracted features could potentially be very good for this problem. Feature scaling was also applied to the features with the method that has been suggested by literature to conclude the pre-processing stage. The results of the implementation of the algorithms to the dataset was then discussed in chapter 5.

The fourth objective was achieved in Chapter 5. As evaluation metrics were identified using literature, and then these were applied to the algorithms in order to analyse and compare their performance which can be found at section 5.3 Results.

The last objective which aimed to improve the performance of the algorithms was not met with much success as section 5.3 included the application of different parameters for the SVM classifier in the attempt of further increasing its performance, however this did not occur, as the default parameter still gave the best possible performance.

Some conclusions can be drawn from this project. The most suitable classifier for any given problem is not clear, it is up to the researchers to test a good variety of classifiers and understand which ones are performing as close as possible to the desired outcome. This can also mean that there might not be a clear “winner”, some classifiers might produce better recall rates while others produce better precision, and it should then be identified what is the most important metric for the given problem.

The imbalance nature of the dataset (having non-pulsar candidates outnumbering pulsars by a great factor) adds an extra challenge to the classifiers as their performance will tend to decrease.

Despite the efforts of achieving the best performance possible, no classifier has achieved 100% recall rate, which is a massive limitation of all the attempted models as it has been stated before how important it is not to miss real pulsars.

## Critical evaluation

As for every project, there are things that worked better than others, or sometimes the results do not meet the expectations previously set upon. Something that was found useful when implementing the algorithms was that saving the attributes into separate csv files instead of loading them directly from the raw hexadecimal files, increased the running time of the programs by a significant amount, in the end, saving a lot of hours considering a lot of iterations were done before the final implementation was achieved.

Another lesson that was reinforced by this project, was that both the right features, and the right algorithms can make a difference to how well something can be predicted by machine learning. Something that worked for this project was the extraction of new features as at least some of the features that were extracted are in fact relevant and suitable for this project, as it was shown by the feature importance measurements in section 4.2 and the results in section 5.3. Furthermore, it was shown how important it is to test different types of classifiers as no algorithm is best for every single problem; the results show that some classifiers outperformed others. For this particular project, random forest achieved the best scores for different measurements: f-score, precision, recall and accuracy; and SVM achieved the best scores for g-mean and recall when combined with the SMOTE imbalance treatment.

Some aspects did not work as well as expected, such as the fact that even though several techniques were used to try improve the classifiers performance, a 100% recall rate of the pulsars was never achieved like some other implementations covered in the related work section. Furthermore, by applying the imbalance treatment, a trade off was attained, instead of a direct increase in performance by all metrics: a higher recall rate was achieved (meaning more Pulsar candidates are being identified) but at the cost of lower precision (meaning more non-pulsars are being labelled as such) increasing the amount of manual inspection required.

## Future work

Although by using different algorithms, settings and features better results were found, it was shown in section 5.3 how they were still far from perfect. Some potential future work, to achieve a better performance of the classifiers could be to find even more relevant features from the datasets as it was shown, the right features is what makes the classifiers perform better.

Given the problem of imbalance of the datasets, finding other imbalance treatments could also potentially increase the performance of at least some classifiers as well, furthermore, more classifiers could be tested to see if there is an even more suitable classifier for this problem.

Finally, the newly extracted feature “Geometric Mean of DM Curve” was shown to be suitable for the problem achieving the third best gain ratio. More tests should be conducted to conclude just how relevant it is and how it differs from the mean of the DM curve when applied to machine learning algorithms.

## Conclusion

The literature and results of the project show that the number of candidates left for human inspection can in fact be reduced by using machine learning approaches. By achieving high recall rates and low false positive rates it was shown that the number of candidates left for human inspection were reduced. These approaches however, do not come without its drawbacks. Unless a 100% recall rate is achieved, the output of the machine learning algorithms will miss real pulsars from the surveys’ datasets which is not an ideal solution as pulsars are extremely rare. So, although the project was able to identify solutions to reduce the amount of human inspection required, these solutions would miss some real pulsars since a 100% recall was not achieved. In order to solve this drawback, either more suitable features should be extracted, or more suitable machine learning algorithms and configurations should be identified.

References

Bishop, C.M., 2006. *Pattern recognition and machine learning*. springer.

Breiman, L., 2001. Random forests. *Machine learning*, *45*(1), pp.5-32.

Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, pp.321-357.

Chen, S.J. and Chen, S.M., 2005. Fuzzy information retrieval based on geometric-mean averaging operators. *Computers & Mathematics with Applications*, *49*(7-8), pp.1213-1231.

Devine, T.R., Goseva-Popstojanova, K. & McLaughlin, M. 2016, "Detection of dispersed radio pulses: a machine learning approach to candidate identification and classification", *Monthly Notices of the Royal Astronomical Society,*vol. 459, no. 2, pp. 1519-1532.

Eatough, R.P., Molkenthin, N., Kramer, M., Noutsos, A., Keith, M.J., Stappers, B.W. & Lyne, A.G. 2010, "Selection of radio pulsar candidates using artificial neural networks", *Monthly Notices of the Royal Astronomical Society,*vol. 407, no. 4, pp. 2443-2450.

Flach, P., 2012. *Machine learning: the art and science of algorithms that make sense of data*. Cambridge University Press.

Gutiérrez-Esparza, G.O., Vallejo-Allende, M. & Hernández-Torruco, J. 2019, "Classification of Cyber-Aggression Cases Applying Machine Learning", *Applied Sciences,*vol. 9, no. 9, pp. 1828.

Han, J., Kamber, M., Pei, J. & ProQuest (Firm) 2012, *Data mining: concepts and techniques,*3rd ed. Elsevier, Burlington, Mass.

Keith, M.J., Jameson, A., van Straten, W., Bailes, M., Johnston, S., Kramer, M., Possenti, A., Bates, S.D., Bhat, N.D.R., Burgay, M., Burke-Spolaor, S., D'Amico, N., Levin, L., McMahon, P.L., Milia, S. & Stappers, B.W. 2010, "The High Time Resolution Universe Pulsar Survey - I. System configuration and initial discoveries", *Monthly Notices of the Royal Astronomical Society,*vol. 409, no. 2, pp. 619-627.

Levin L., 2012, “A Search for Radio Pulsars: from Millisecond Pulsars to Magnetars.”, Ph.D. thesis, Swinburne University, Faculty of Information and Communication Technology.

Lyon R. J., 2016, *"Why Are Pulsars Hard To Find?"*, PhD thesis, University of Manchester.

Lyon, R.J., Stappers, B.W., Cooper, S., Brooke, J.M. & Knowles, J.D. 2016, "Fifty years of pulsar candidate selection: from simple filters to a new principled real-time classification approach", *Monthly Notices of the Royal Astronomical Society,*vol. 459, no. 1, pp. 1104-1123

Morello, V., Barr, E.D., Bailes, M., Flynn, C.M., Keane, E.F. & van Straten, W. 2014, "SPINN: a straightforward machine learning solution to the pulsar candidate selection problem", *Monthly Notices of the Royal Astronomical Society,*vol. 443, no. 2, pp. 1651-1662.

Ng, C., Champion, D.J., Bailes, M., Barr, E.D., Bates, S.D., Bhat, N.D.R., Burgay, M., Burke-Spolaor, S., Flynn, C.M.L., Jameson, A., Johnston, S., Keith, M.J., Kramer, M., Levin, L., Petroff, E., Possenti, A., Stappers, B.W., van Straten, W., Tiburzi, C., Eatough, R.P. & Lyne, A.G. 2015, "The High Time Resolution Universe Pulsar Survey – XII. Galactic plane acceleration search and the discovery of 60 pulsars", *Monthly Notices of the Royal Astronomical Society,*vol. 450, no. 3, pp. 2922-2947.

Raschka, S., 2014. Naive Bayes and Text Classification I - introduction and Theory. *Computing Research Repository (CoRR)*, 2014. Available from: http://arxiv.org/ pdf/1410.5329v3.pdf [Accessed 19 November 2019].

Raschka, S., 2015. *Python Machine Learning.* 1st ed. Birmingham: Packt Publishing Ltd.

Wolpert D.H. “*The Supervised Learning No-Free-Lunch Theorems*.” In Soft Computing and Industry, edited by R. Roy, M. K¨oppen, S. Ovaska, T. Furuhashi, and F. Hoffmann, pages 25–42. Springer London, 2002.

Zhu, W.W., Berndsen, A., Madsen, E.C., Tan, M., Stairs, I.H., Brazier, A., Lazarus, P., Lynch, R., Scholz, P., Stovall, K. and Ransom, S.M., 2014. Searching for pulsars using image pattern recognition. *The Astrophysical Journal*, 781(2), p.117.