INDIAN INSTITUTE OF SPACE SCIENCE AND TECHNOLOGY

DEPARTMENT OF EARTH AND SPACE SCIENCES

Did you use IIST style file for thesis? Looks different.

Classification of Globular cluster X-ray sources Using Machine Learning methods

MS Astronomy and Astrophysics- Thesis Phase-I

Shivam Kumaran
SC17B122
IIST

Guide
Dr. Samir Mandal , IIST
Dr Sudip Bhattacharya , TIFR
Dr Deepak Mishra IIST
Dr Anjali Rao , IIT Indore

No gap before coma and a gap after.
Follow this through...

December 12, 2021

Acknowledgement

Abstract

Contents

Chapter 1

Introduction

You have used ACIS data.
X-ray Chandra Check the angular resolution!
Advanced x-ray telescope like chandra with its subarcsec angular resolution and high sensitivity can detect and resolve very faint sources. The catalogue generated by all sky survey of such telescope contains wealth of untapped information. Using the ACIS observation data from Chandra, ~~catalogue of X-ray sources~~ Source Catalogue (CSC) was created and published which contains a mammoth population of 317, 167 sources. Identifying each source individually with manual methods are humanly not possible. However quite a few examples sources exist which are manually identified. Given sufficient amount of labelled data , machine learning models are proven to be very efficient in such classification task.

No gap
One such avenue where automatic classification could be put to practical use is for identifying different class population in globular clusters.(GC)

1.1 Motivation

1.1.1 Why CHANDRA ?

use correct symbol
With on-axis resolution of 0.5" chandra can resolve ~~point sources within~~ globular cluster. not in math m in 0.4 - 6 keV with 100 sec inte Due to its high sensitivity ($4 \times 10^{-15} ergs/cm^2/s$ in 104 s (0.4 to 6.0 keV)) it can detect faint x-ray sources, which are found to be abundant in globular clusters.[1]. With Chandra such studies on faint x-ray sources are already done on some globular cluster, for instance in 47 TUC 370 x-ray sources are found within a radius of 2.79 arcmin.[2] , 140 faint x-ray sources in GC ω Centauri.[3] Other than 47 TUC , chandra source catalogue 2.0 has observations corresponding to 1500 x-ray sources sources associated (not confirmed) with $\sim$ 80 globular clusters.

What stud
stop after citati
(CSC 2.

Figure 1.1: Images of the core of 47 Tuc made from 8 ks of Einstein data (Left), 77 ks of ROSAT data (Center), and 240 ks of Chandra data (Right) . Image and caption credits [4]

1.1.2 Why globular cluster ?

Due to high stellar population density , dynamic interactions, close encounters are quiet frequent in globular clusters than galactic place.[1]. Due to these encounters GC are efficient in formation of binary systems like (LMXB, CVs , MSP , AB). Simulation of dynamic evolution of GC without any x-ray binary system resulted in collapse with time scale smaller than mean age of galactic globular clusters($11.5 \pm 2.6 Gy$) [5]. Interactions of such binary systems plays a major role in maintaining the stability of GC against gravitational collapse and subsequent disruptions. The encounter frequency is well correlated with the population density of x-ray binaries.[1]. Hence the identification of x-ray source in GC could help in understanding the dynamic evolution and stability of GC.

1.1.3 GC X-ray binaries

GC X-ray binary systems and their current identification methods are discussed here briefly. Most bright sources in globular cluster with luminosity $L_X > 10^{34} ergs/s$ are LMXBs containing Neutron Star , (13 in 152 Globular clusters) ongoing type-I outburst. Faint sources with $L_X < 10^{34} ergs/s$ are composed of quiescence LMXB, cataclysmic variables (CV) , active binaries and milli second pulsars.[3]

Cataclysmic variables

CVs are x-ray binaries with a White dwarf accreting matter from late main sequence star. Low mass transfer rate result in outburst and object is identified as Dwarf Nova. [6] High mass transfer rates result in variability in magnitude on time scales of weeks. CVs can be identified by their blue variable counterpart.[3]

Millisecond Pulsars

Millisecond pulsars are rapidly rotating neutron stars with rotation period , $10ms$ with NS LMXB as their likely progenitor. They are spun up because of mass and angular momentum transfer from binary companion[7] 150 MSP in 28 globular clusters. Around one third of total population of MSP are found in globular clusters.[8] of which more

than half are in binary system. Main method of identification for MSPs by their radio counterpart.

Low Mass X-ray binaries

Low mass xray binaries are x-ray emitting binary systems where the compact object is either Neutron stars or Back holes, and the counterpart mass is $\leq 1.5 M_{\odot}$ Highly luminous xray sources $L_X > 10^{34} ergs/s$ in GC are LMXB in transient/outburst state.

1.2 ML for X-ray source classification

With the ever increasing number of detected sources the identification using manual methods have becoming challenging. Although not much work has been done using machine learning in x-ray astronomy for classification of sources. One such major work is done for XMM-Newton variable x-ray sources by Ferrel et al.[9]. They have considered all sky survey and not particularly globular cluster , and have used timing properties and trained Random Forest[10] algorithm for classification. However no work has been done for any such automated classification for Chandra source catalogue.

1.3 Problem Statement

We need to develop an automated classification algorithm using machine learning methods for classification of X-ray sources detected by chandra associated with globular cluster using the properties available in chandra source catalogue.

1.4 Workflow

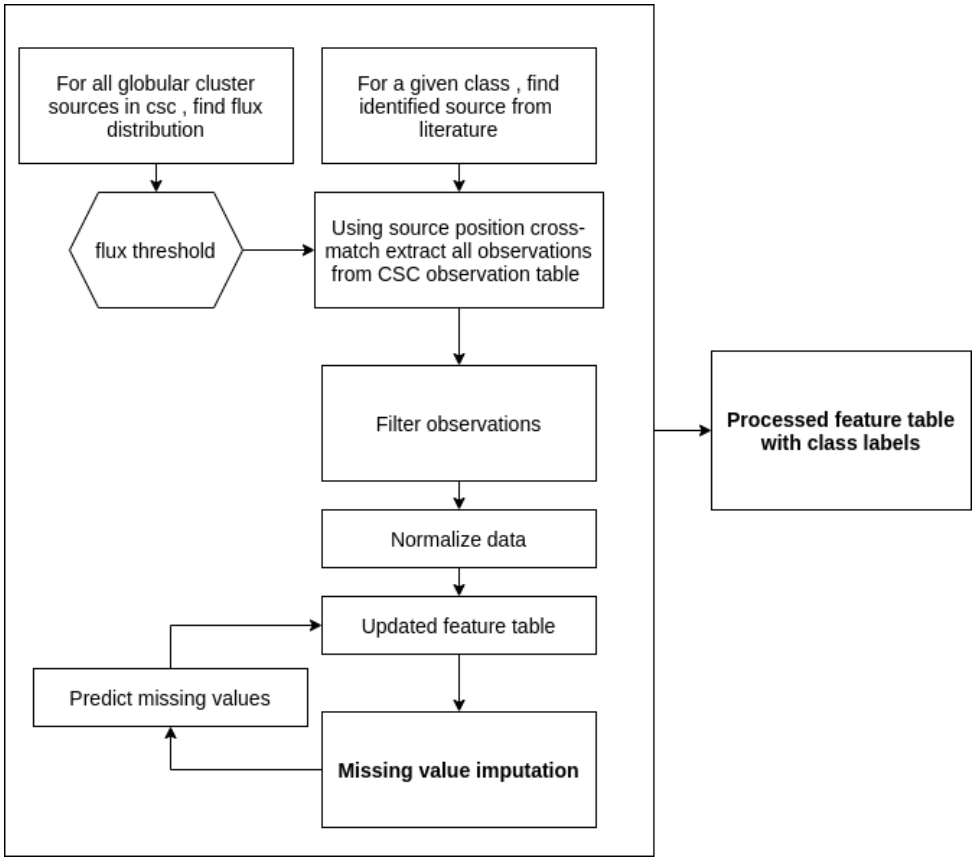


Figure 1.2: Flowchart showing the workflow for getting and preparing data for training.

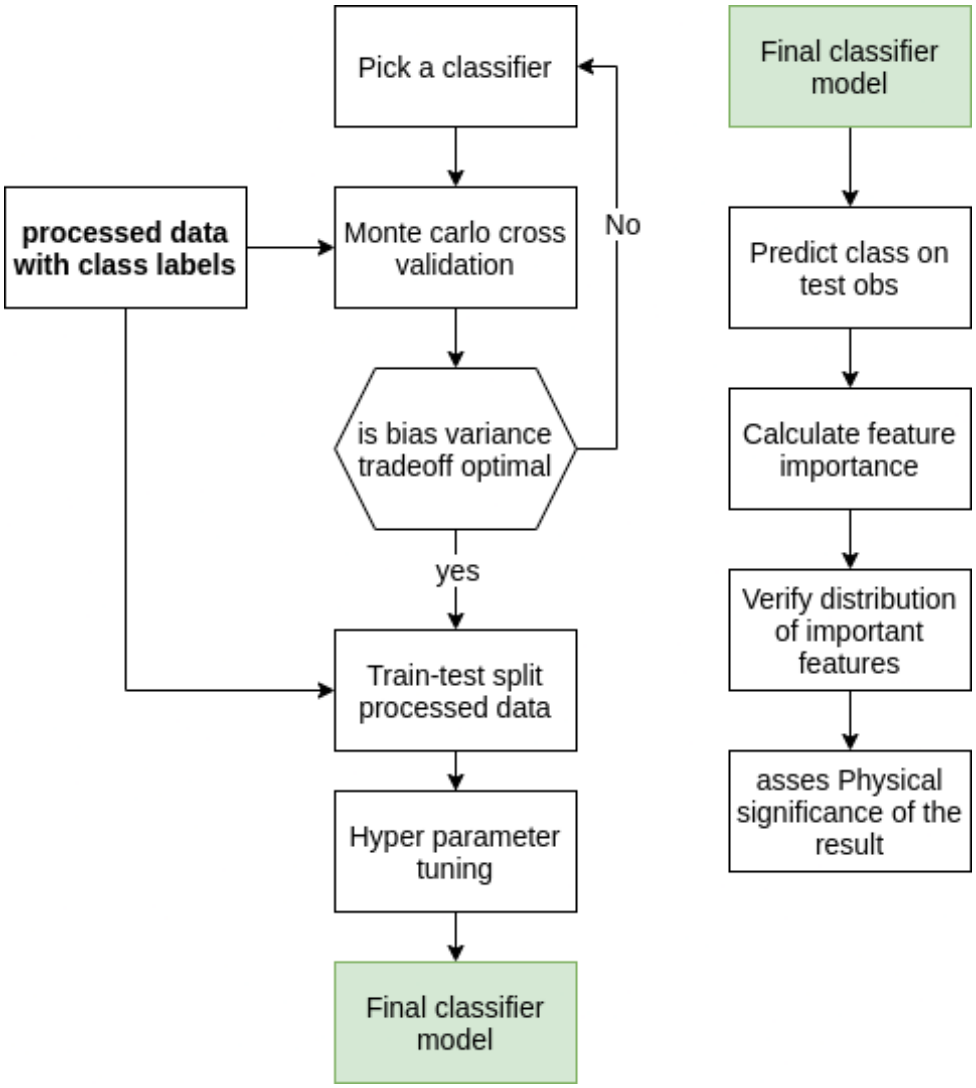


Figure 1.3: Flowchart showing the workflow to identify and train the ideal classifier starting from processed data

This report presents the first phase of the work. Chandra source catalogue and the features available for observations are discussed in details. Next section gives an overview of the available training data and discuss problems associated with them and their solution. In next section we try to identify best possible classifier using cross validation. Best identified classifier is trained and tested after hyper parameter tuning which is explained in selected.

Chapter 2

Chandra Source catalogue

2.1 Chandra observatory

CHANDRA NASA's X-ray observatory having unparalleled on-axis spatial resolution of 0.5 arcsec (almost 10 times of ROSAT or 40 times of XMM-Newton). With this superior resolution Chandra can better resolve closely spaced sources as that in the globular cluster. Basic info on instrum

2.1.1 Instruments

2.2 CSC 2.0 overview

Chandra source catalogue is the catalogue of x-ray sources detected by Chandra . With high sensitivity limit and subarcsec on-axis spatial resolution this catalogue contain a large number of uniquely identified and uniformly calibrated sources.

- Total Number of unique sources - 317,167
- Total number of unique observation detections - 928,280
- Number of sources associated with globular cluster -

The catalogue is split into three tables :

- **Master Sources Table** :The Master Sources Table contains 'best estimate' sources properties for each unique X-ray source in the catalog. These properties are typically determined by analyzing simultaneously the detections from the set of individual observations.
- **Stacked Observation Detection Table** contains detection properties based on observational data extracted independently from each stack of Chandra pointing observations in which the source is detected; why'
- **Per-Observation Detection Table** contains detection properties based on observational data extracted independently from each individual observation in which

Detection table per ovserv

the corresponding stacked observation detection is located.

2.3 Features

Features available corresponding to each source observations in the catalogue

2.3.1 source Information

- Position RA-DEC
- Galactic coordinates position
- Exposure timings
- Instrument used
- Flux significance - ratio of the single-observation detection photon flux to the estimated error in the photon flux, for each band

Likelihood

Source likelihood is given by

$$L = -2ln(P) \tag{2.1}$$

Where P shows the probability that the source count in the aperture is possible to obtain via a model Poisson distribution with mean aperture background counts. Higher P values mean higher chance that the count in the aperture could simply come from background and hence lower likelihood.

Flux significance

Ratio of flux measured to its average error

$$\sigma = \frac{(Photflux_uplim) - (photflux_lolim)}{2} \tag{2.2}$$

2.3.2 Aperture Photometry

Corresponding to different bands , from the counts photonflux is derived with background counts from nearby region subtracted. From photon flux ($counts/s/cm^2$) energy flux in units ($erg/s/cm^2$) is calculated by taking total number of source counts and scaling by the local value of ARF at the location of incident photon. values are reported with upper and lower confidence limit based on background-marginalized posterior probability distribution.

- photon flux in different bands
- energy flux in different bands

2.3.3 Spectral Properties

Hardness ratio

Hardness ratio from the fluxes in three different energy bands are given :

- hard(h, 2.0-7.0 keV),
- medium (m, 1.2-2.0 keV)
- soft (s, 0.5-1.2 keV)

And the hardnes ratio is given by :

$$hard_{xy} = \frac{F(x) - F(y)}{F(x) + F(y)} \tag{2.3}$$

where x,y represent different bands , (by convention x is always higher band) and F is the flux in the corresponding band.

Hardness ratio between two bands (x,y : where x,y may be h,m,s,u bands) are given as

$$hard_xy = \frac{F(x) - F(y)}{F(x) + F(y)} \tag{2.4}$$

where F(x) and F(y) are aperture source photon flux in x and y band , where x,y may be h, m, s, u.

Model fit

Table 2.1: Model fit parameters

Power-law	**Black Body**	**bremsstrahlung**
photon index $F_E \propto E^{-\gamma}$	BB Temperature	Temperature
Model Flux	Model Flux	Model Flux
NH Column density	NH Column density	NH Column density
Amplitude	Amplitude	Normalisation

2.3.4 Source Variability

Based on source region counts within observation source variability is calculated by the following methods.

- Kolmogorov-Smirnov (K-S) test
- the Kuiper's test
- computation of the Gregory-Loredo variability probability

Both inter-observation and intra-observation source variability properties are given in the table. We are using only intra-observation properties.

Chapter 3

Data Preparation

Chandra source catalogue is huge with 317,000 unique identified sources. Limiting our search to around 30 arcsec of radius around globular cluster results in $\sim$ 1500 sources. From the literature review of globular cluster we have identified the major population of x-ray sources in globular cluster are of x-ray binaries viz. Low-mass xray binaries (LMXB) , cataclysmic variables (CV) , Pulsars.[1]. As our concern is GC sources, We will try to do classification for these three classes. Using **CIAO** tool we can download the features corresponding to different observations for these sources. The major challenge is that in CSC there is no classification of any source/observation given. But for training supervised machine learning algorithm we need some already classified dataset , hence we need a sample of sources for which are available in CSC and for which we know the class.

Define source c

3.1 Data Collection

For each class :

- From literature survey or other catalogue, find out sources belong to that class
- Pick out RA - DEC of those sources
- Using this RA-DEC, cross match with chandra source catalogue (cross-match radius 10 arcsec)
- If there are more than one cross matched source (very likely due to higher resolution of chandra than any other x-ray observatory) , keep the source with highest cross match.
- Download all the observation data corresponding tho this cross-matched source.

BH LMXRB catalogues

Table 3.1: Published Catalogues used to get sources identified as black hole lmxrb , catalogue codes are given as per HEASARC

Catalogue CODE	Cat Name
INTREFCAT	INTEGRAL Reference Catalog
NGC3115CXO	NGC 3115 Chandra X-Ray Point Source Catalog
RITTERLMXB	Ritter Low-Mass X-Ray Binaries Catalog
SAXWFCCAT	BeppoSAX Wide Field Camera X-Ray Source Catalog
SAXWFCCAT2	BeppoSAX Wide Field Camera Unbiased X-Ray Source Catalog
WGACAT	ROSAT PSPC White, Giommi, and Angelini 'Good' Source Catalog

NS LMXRB catalogues

Table 3.2: Published Catalogues used to get sources identified as black hole lmxrb , catalogue codes are given as per HEASARC

CAT CODE	**CAT name**
IBISCAT	IBIS/ISGRI Soft Gamma-Ray Survey Catalog
INTREFCAT	INTEGRAL Reference Catalog
RASS2MASS	ROSAT All-Sky Survey BSC/2MASS PSC Cross-Associations XID II Catalog
RITTERLMXB	Ritter Low-Mass X-Ray Binaries Catalog
SAXWFCCAT	BeppoSAX Wide Field Camera X-Ray Source Catalog
SMCWINGCXO	Small Magellanic Cloud Wing Survey Chandra X-Ray Point Source Catalog
WGACAT	ROSAT PSPC White, Giommi, and Angelini 'Good' Source Catalog
XMMSSCLWBS	XMM-Newton 2XMMi-DR3 Selected Source Classifications Catalog
XRBCAT	X-Ray Binaries Catalog

Cataclysmic variable catalogue

For Cataclysmic variable sources , **The Open Cataclysmic Variable Catalog** is used. This catalogue was published on December - 2020 , and contains information about $\sim 12,000$ CVs and candidate CVs and is available through university of Washington website.[6]

Pulsar Catalogue

For Pulsars , Australian telescope National Facility **ATNF** catalogue is used, which contains information about $\sim$ 3100 rotation powered pulsars. It gives classification for pulsaras as

- AXP Anomalous X-ray Pulsar or Soft Gamma-ray Repeater with detected pulsations
- BINARY Pulsar has one or more stellar companion(s)

- HE Spin-powered pulsar with pulsed emission from radio to infrared or higher frequencies
- NRAD Spin-powered pulsar with pulsed emission only at infrared or higher frequencies
- RADIO Pulsars with pulsed emission in the radio band
- RAT Pulsars with intermittently pulsed radio emission
- XINS Isolated neutron stars with pulsed thermal X-ray emission but no detectable radio emission

More information about number of sources available in catalogues and number of cross matches found with Chandra source catalogue belonging to these classes are give in table ??

3.1.1 Problems with Dataset

- **Missing values** Data table is very sparse , 50% values are missing.
- Large difference in number of observation for different sources , refer figure 5.1
- **Small dataset** Since the number of classified sources are small and limited , number of sample for training is very small
- **Class Imbalance** LMXB dataset is very small compared to other classes.
- Order of magnitude difference in different features
- In the data table Datatype is not uniform across samples for some of the columns

3.2 Data Filtering

- Flux filtering
- Pileup-filter
- streak source filter
- saturation filter
- significance filter

Flux filter

In maximum cases the identified sources are classified only when they goes into outburst (for the case of Xray binaries). But our aim is to classify sources in CSC which are in quiescent state , which have quiet different properties even for the same source. So we need to make sure that the data our classifier is trained on comes from the sources in quiescent state so we need to take observations corresponding to the source when they were in quiescent state.

During outburst variation of luminosity of LMXRB in quiescent state is $L_X \approx 10^{32} erg/s$. During outburst luminosity goes as high as $L_X \approx 10^{36} - 1p^{38} erg/s$ [11] Now if we

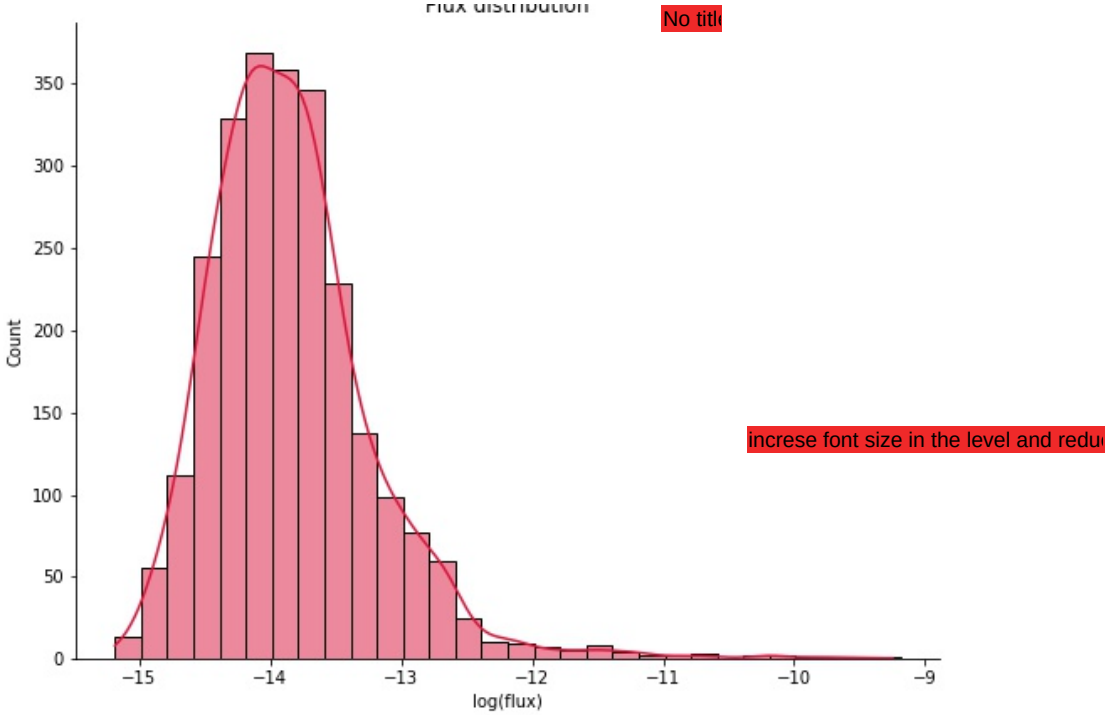
are considering only galactic sources, using distance limits (distance to the center of the galaxy - $\approx 8kpc$, diameter of the galaxy $\approx 15Kpc$) we can come up with a boundary value of minimum flux for outburst sources.

Dist | L_x	10^{36}	10^{38}
1kPc	8.4×10^{-9}	8.4×10^{-7}
8kPc	1.3×10^{-10}	8.4×10^{-8}
15kPc	3.7×10^{-11}	3.7×10^{-9}

centre

Put table num

Hence from the table above, we can conclude that if we take flux less than $10^(-12) erg/s/cm^2$, we can make sure that the obs is not during outburst. We also look at the flux distribution of all chandra x-ray sources belonging to 157 galactic globular cluster [12]



increse font size in the level and redu

Figure 3.1: Globular cluster x-ray sources flux distribution. Flux values considered are in ACIS Broad-band filter

The figure 3.1 shows the broad band filter energy flux distribution for GC X-ray sources. It is clear that most of the sources are well below flux level $10^{-12} erg/s/cm^2$.

	CV		LMXB		Pulsar	
	Source	Obs	Source	Obs	src	obs
Total Number of sources			99			
Unique Cross matched with Chandra	112		99		138	
Unique id , ACIS obs available	87	1079	99	735	135	545
Source filters	72	1044	79	662	127	475
Flux filter	65	1007	74	602	110	325
Significance filter	60	994	58	521	92	297

3.3 Prepossessing

Before passing the data to any ML classifier , prepossessing on data is very crucial have a very high impact on the classification performance. First of all data is sanitized, that is made sure that in every column the data format is consistent, that is data-type should be uniform across the sample for each column. Based on the classifier of choice further processing is required.

3.3.1 Flux normalisation

In the source observation table flux values are given in the units of $erg/s/cm^2$ which result in the values of the order of $10^-12 - 10^{-16}$. Log value of all the flux columns are taken.

3.3.2 Re scaling

several algorithms like Neural network[13] , CNN are highly sensitive to the scale of data. Classifier tends to assign more weights to the features with higher magnitudes and can result in feature importance which may be inherently wrong. Hence values for each column is standardized to have zero mean and unit variance.

- Normalisation for each column of features :

$$xi = (xi - max)/(max - min) \tag{3.1}$$

  Such that for each observations , feature all the feature values are within 0-1

- Standardisation

$$xi = (xi - mean)/var \tag{3.2}$$

  such that the distribution of each feature becomes standard , with '0' mean and unit variance

3.4 Missing value Imputation

Missing values in the data is a major challenge for our work.

3.4.1 Reason for missing values

- All observations are not taken in all bands
- Some source are too faint for certain bands , associated with ow likelihood
- For source flux values not available in m ,s, h band hardness values are not given.
- For some observation duration of obs is not sufficient for calculation of variability parameters
- For sources obs in less than 3 bands , model-fit parameters are not given.

—— plots for missing values should come here ——- —— show the amount of missing values for different features here —— For the features with more than 50 % missing values are removed from subsequent processing and not used. One general starting point is to

drop rows or columns with any missing value. But in our dataset all the columns have certain amount of sparsity and there are hardly a few observations (—-give actual number here—) for which all feature values are available , so we can not drop such columns, and we must find a way to guess those missing values.

Following different techniques are explored for predicting missing values :

3.4.2 Statistical imputation

- Simplest approach is to fill in the missing values with 0's , but since our dataset have 0's as true values , imputation with zero would result in distortion of dataset.
- Impute with column mean value
- Impute with column mode value

3.4.3 Correlation Imputation

This is a *novel technique* in this work, where we leverage the correlation between the features to make a prediction for missing value.

Procedure

1. correlation coefficient for each pair of feature is computed (feature-feature correlation matrix)
2. For each observation (say x_i) , for missing feature , say f_i , highest correlated feature , say f_j is selected.
3. for x_i , using value of feature f_j , feature f_j id filled with linear regression output from feature (f_i , f_j).

— make one such demonstration —-

3.4.4 Regression Imputation

The idea is to use a regerssor for predicting missing value.

Procedure

- Column with highest missing value is selected called candidate column
- other columns' missing values are imputed using column mode
- For candidate column , all the examples (rows) with Cn values available are treated as training dataset
- With all other non-candidate columns , regression model is trained
- With this trained regressor missing values for candidate columns are predicted.
- This is done iteratively with next most sparse column as candidate column.

The regressor can be any regressor say , Neural network , RF , etc. For faster performance RF is chosen.

3.4.5 Similarity Imputation

Here the idea is to use the similarity (or correlation) between examples to fill in the missing values

Procedure

- Initially all the missing values are filled with column modes
- For a selected column, for selected example missing value is filled with weighted average of values available in that column from with weighting factor being the similarity between examples.
- Once all the missing values filled as above, again similarity matrix (may be correlation matrix) is calculates and missing values are again predicted as above step
- Repeat this process until missing values prediction converges.

Chapter 4

Classifiers Selection

For identifying the best classifier a number of supervised machine learning models , starting form very simple models like KNN to more complex model like XGBoost was tried.

4.1 Classifier models

4.1.1 K-Nearest Neighbour

It is a supervised deterministic method for classification. KNN basically works under the assumption that in feature-space, same class instances are closely spaced together than those belonging to different class. The closeness is determined by the euclidean distance in feature-space.

How it works

For a given K-value and for unidentified sample in feature-space it find K-Nearest sample assigns the class which is most frequent in those k-nearest sample. The only hyper-parameter to learn in this is K-value.

caveats

This method fails if there exist a more complex decision boundary in feature space or in case of no class-wise clustering.

4.1.2 Fully connected network

Fully connected Neural network is a sub class of feed forward artificial neural network. It is composed of layers of nodes with number of nodes in output layer as number of classes and the number of nodes in input layer as number of features.

How it works

At the input layer , each feature is multiplied with a weight values and goes to node of next layers where the weighted sum is taken as input of activation function. These notes

are treated as the input to the subsequent layers. Finally the output is compared to the true output , and the difference can be expressed in terms of loss function.

For a given dataset and network architecture , the loss function is a function of weights and biases in the network. The aim of training is to minimise this loss value , which is done by gradient descent algorithms.

Caveats

- Easily prone to overfitting , especially on small dataset
-

4.1.3 Convolution Neural Network

In Deep Learning CNN is one of the advanced tools , which proves very useful for feature extraction as well.. It is also layered structure as the FCN , but at each layer instead of simple weights, if performs a convolution between input layer and a kernel (filter) to give output for the next layer.

CNN can extract feature from the input data. It tries to extract localised relation between neighbouring features to act as the feature in subsequent layers.

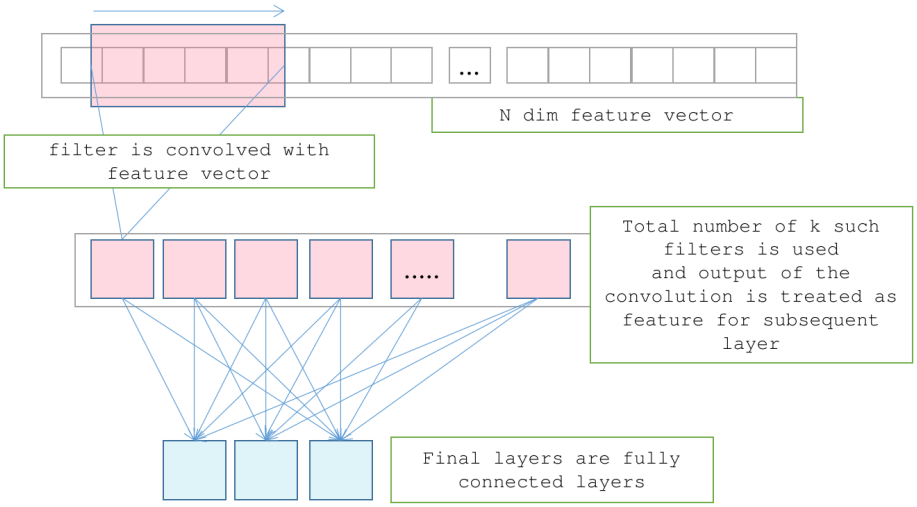


Figure 4.1: CNN Network illustration

In our dataset , if the features are arranges in a vector form with context-wise features (correlated features , say all the flux values in different bands) put together , CNN filters can extract some localised relations better than fully connected networks.

4.1.4 Random Forest[10]

Random forest[1] is a ensemble classifier , with **Decision trees** as the fundamental units. Each decision tree is composed of layers of decision nodes, where for each data sample

[1]Random Forests(tm) is a trademark of Leo Breiman and Adele Cutler

, at each node one of the feature value is compared against a threshold value based on which this node splits into different branches.

How it works

For training of decision tree ,all the training examples are passed and based on this feature values each of them end up in some of the leaf nodes. At the end ,each leaf node is assigned a class based on the majority class of the examples ending up in that leaf node. Now for any unlabelled example , we pass through the decision tree and assign a class , based on which leaf node it ends up.

given example is passed through each decision tree and class probability value is assigned as the fraction of trees predicting it as a given class.

Caveats

4.1.5 AdaBoost

AdaBoost[14] is also a ensemble classifier , but each ensemble component is build sequentially. First a weak learner is trained with equal weight to each training sample. Then in subsequent models the sample weights for wrongly classified examples are increased while the weights for correctly classified samples is reduced. Hence subsequent classifiers concentrate more on learning difficult examples. Finally the output is combined through a weighted majority vote for predictions.

4.1.6 Xtreme Gradient Boost

4.2 Model Selection Scheme

Now we have different schemes for **Data imputation** and then different models to build a classification pipeline. Any good classifier should have the following requirements

- Should have higher accuracy
- Classification performance should be less sensitive to change in training sample.
- Should have low P-score in permutation test , model must understand *Class-feature relationship.*
- classifier has to be confident in predicting Output class-membership probabilities should be high ,

4.2.1 Cross validation

K-fold cross validation

Entire training dataset is split into K-folds , and each time network is trained on K-1 folds and validated on the remaining set. This method makes sure that each example gets a chance to be in the training and validation set.

However, given that our dataset is very small, number of samples in validation set will be too small for any statistical interpretation. For significant population in validation set, we need small K value, but then the 'result' statistic will be poor.

Monte-Carlo cross validation

In MC cross validation in each iteration , examples for validation set is randomly selected. Number of iterations can be arbitrarily large to increase the chances of each sample coming in validation set at least. once.

Procedure: Selection Methodology

- Select a data imputation scheme
- Select classifier
- Train the classifier
- Random search hyper parameter tune classifier
- Perform Monte Carlo cross validation

For identifying the best pipeline , Classification within Low mass x-ray binary class is considered and classification is done under the class Neutron star LMXB or Black Hole LMXB.

4.2.2 MC validation Result

From the figure 4.2 , shows that except Random forest , data set without normalisation have a very high variance in test accuracy and also a lower mean accuracy. In most of the cases , standardisation works better. For RF case , data scaling does not have much impact.

From the plot 4.3 , we see that for KNN , FC and CNN Correlation imputation works best with accuracy distribution skewed towards higher side. For random forest , correlation imputation has resulted in lower accuracy, whilst median imputation works best for RF.

From these two figures(4.3 4.2) , it is also evident that Random Forest has least variance and higher test accuracy. *Random Forest is our classifier of choice* For comparison Random Forest imputation is tested with RF classifier. From plot 4.4 , we see that RF imputation has the least variance and highest mean test accuracy. Also mean train accuracy is lower for RF , however this shows least over fitting case. *RF imputation is our choice of data imputation* and since we have selected RF classifier , data scaling does not have much effect (this behaviour is expected since RF works based on decision trees), so to avoid floating point calculation accuracy we choose to use *Normalisation*

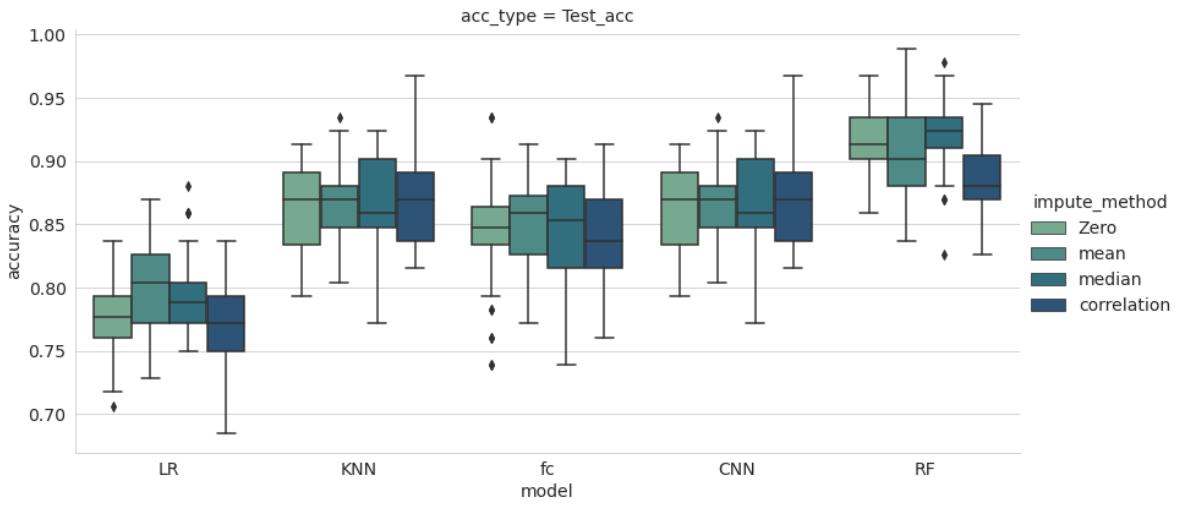


Figure 4.3: Variations of test accuracy for 32 times training random train-test split of dataset , with normalised dataset but with various imputation techniques for different classifiers

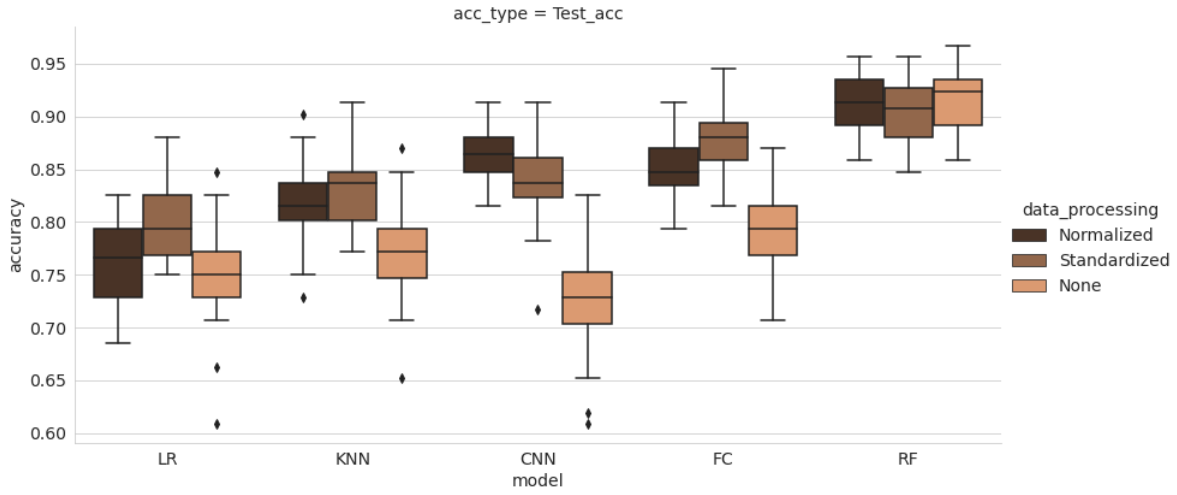


Figure 4.2: Variations of test accuracy for 32 times random train-test split of dataset , with zero imputation and indicated data normalisation for different classifiers

[H]

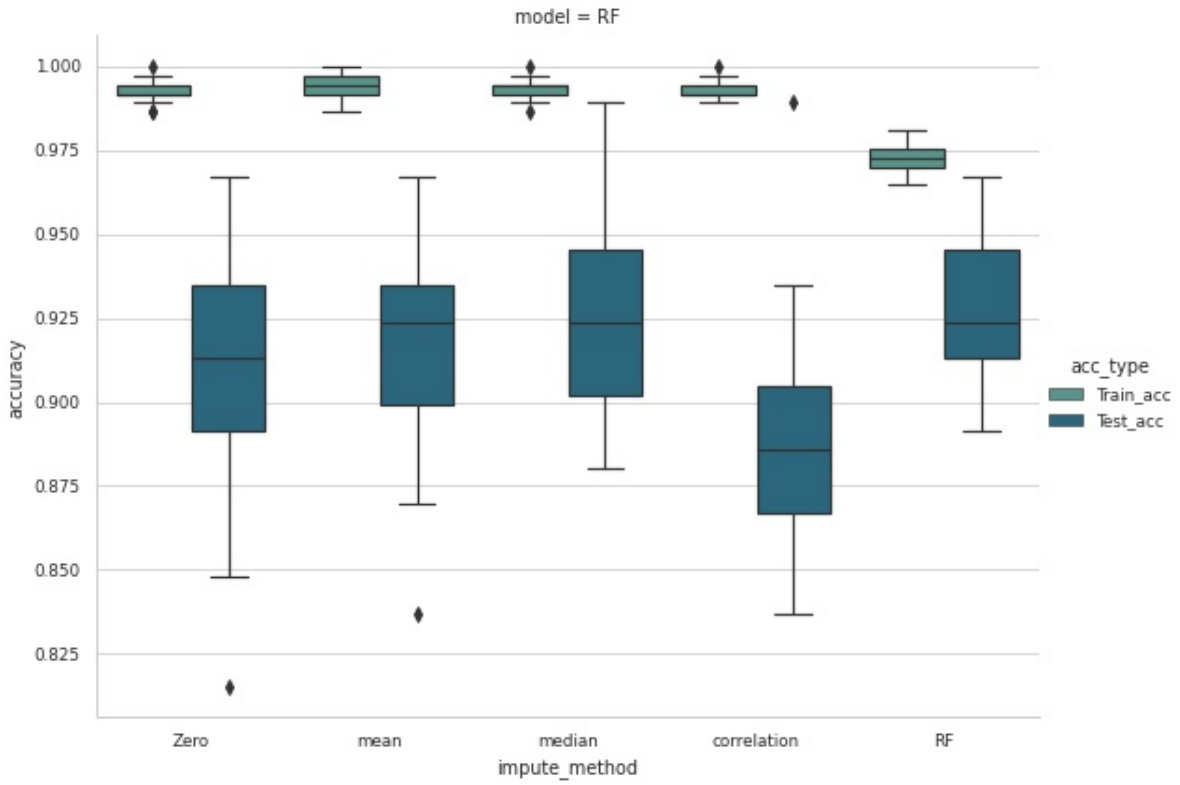


Figure 4.4: Variation of test accuracy for Random Forest classifier , for normalised data , but with different imputation methods , with Imputation using radom forest included.

: Schematic selected

- Data scaling - Normalisation
- Data Imputation - Random Forest Imputation
- Classifier - Random Forest

Chapter 5

Classification

By the Monte-Carlo cross validation for NS-BH LMXB classification , we have found out the best pipeline for classification is , to first normalise the data , then do imputation using random forest iterative method. For classification staged design is proposed , where in the first stage we classify sources in broad classification as LMXB , CV , Pulsars. Then further classification of LMXB in terms of Neutron star LMXB and Black hole LMXB , is done in the next stage. In this section we discuss the classification details of stage - I classifier.

5.0.1 Training - validation data overview

5.0.2 Source vs Obs classification

Since in the source catalogue, for each source multiple observations are available , and the number of observations per source is not uniform and have a large variation.

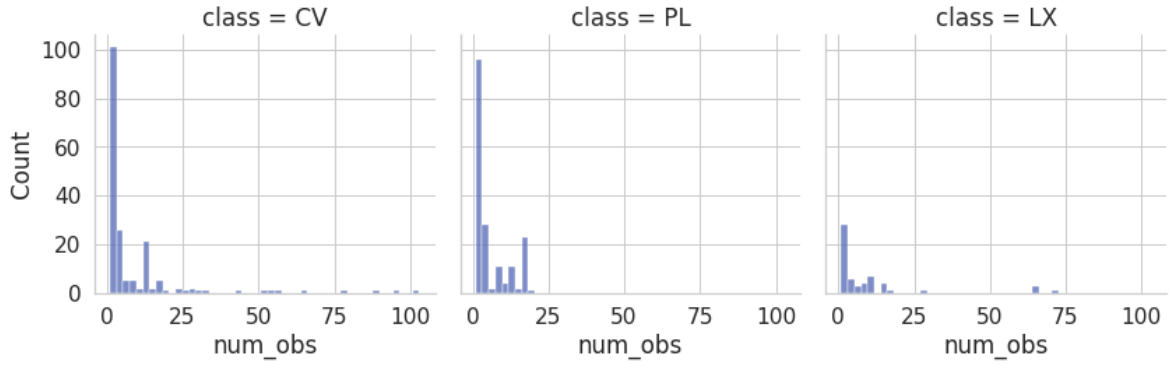


Figure 5.1: Histogram showing the number of observation (hence number of rows in data table) available corresponding to each source. Y axis shows total number such sources having number of observation in a given bin on x-axis Plots from left to right shows histogram for CV , PULSAR , LMXB.

From the figure 5.1 it shows that number of observation for individual sources is not uniform and for maximum number of sources, number of obs is 1 , whereas for some sources , number ob observations are as high as 100.

It gives us an option to use observation wise classification, we can either choose to train our classifier for identification of individual observations or we can combine the observation for individual sources and then train the classifier on combined observation set. The problem with observation-wise classification is that due to non-uniform number of observations classifier training becomes much sensitive to change in training validation dataset , giving much higher accuracy variance. While in the latter case , by combining observations improves the observations statistics and makes number of obs available for each source uniform with a trade-off that it reduces the number of training samples. It is also evident from the cross validation result shown in the table **??**. For observation-wise classification , the variation in accuracy is 4.3 and goes to as low as 51 % and as maximum validation accuracy 68%. Whereas for source-wise classification , mean accuracy is $\sim$ 10% higher and also the result is more consistent with standard deviation only 2 % .

Table 5.1: Cross validation accuracy for obs-wise classification and source-wise classification , compared on baseline RF model with 100 decision trees.

	Mean	Std	min	max
Obs-wise classification	62	4.3	51.46	68.31
Source-wise classification	76.37	1.8	71.4	79.36

With the selected data processing pipeline and from the above result we are going with source-wise classification scheme. Classifier chosen is random forest classifier and with default values in SKLEARN [15] , number of decision trees - 200 , classification result are shown as confusion matrix in figure 5.2

Confusion Matrix

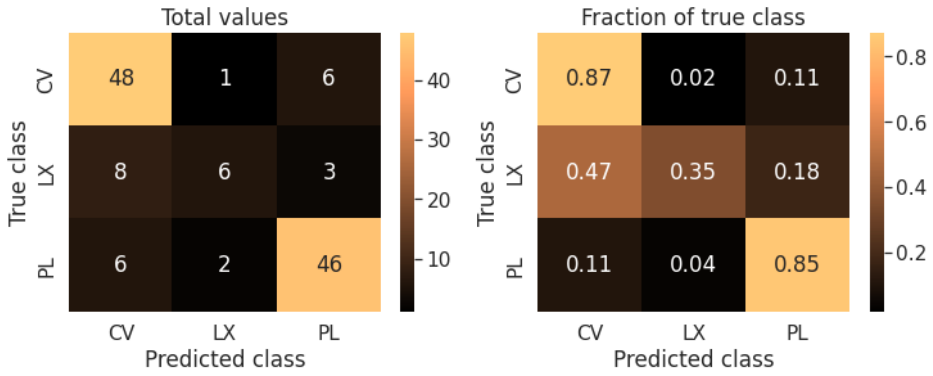


Figure 5.2: Confusion Matrix showing the result of source-wise classification. Y-axis shows the number of sources belonging to that class, x-axis show the number of sources predicted by our classifier. Right figure shows the fraction of sources truly belonging to class on x-axis predicted s that of the class on y-axis.

5.0.3 Classification score

Table 5.2: Class wise precision and recall score

	Precision	Recall
CV	0.77	0.87
LMXB	0.66	0.35
PULSAR	0.83	0.85

From the confusion matrix 5.2 it is shown that 87% of CVs and 85% of Pulsars are classified correctly while only 35 % of LMXB are classified correctly , while majority of LMXB are classified as CVs. This may be due to class imbalance, as number of LMXB is an order of magnitude smaller that other classes, and also the scoring statistics is poor for this class due to low number of examples in validation set. if we increase the number of examples in validation set , then training set will reduce and result in poor accuracy, hence there is a trade-off between training quality and validation statistics for small population class.

Training accuracy can be improved by using sampling examples from the distribution in feature space for minority class , one such up-sampling technique is explored and discussed in the next section.

5.1 Balancing Class: SMOTE

Synthetic Minority oversampling technique **SMOTE** , is a method to generate new samples of data synthetically for minority class.[16] In feature space, it uses linear interpolation between closely spaced data points belonging to the given class, and from between this linear interpolation data points are sampled. SMOTE algorithm works better if the number of data points available is enough to represent the parent distribution.

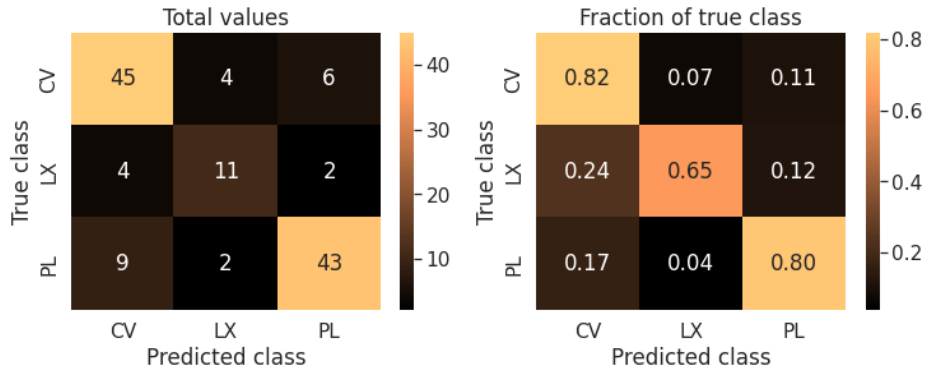
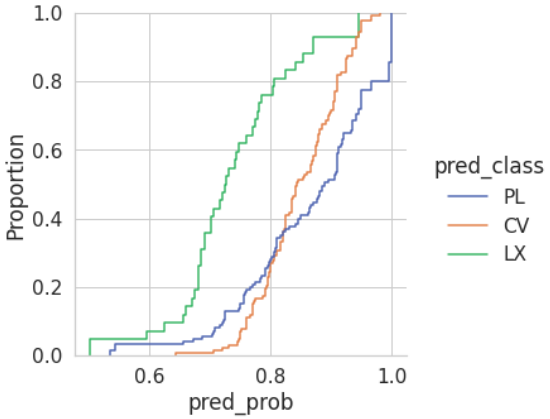
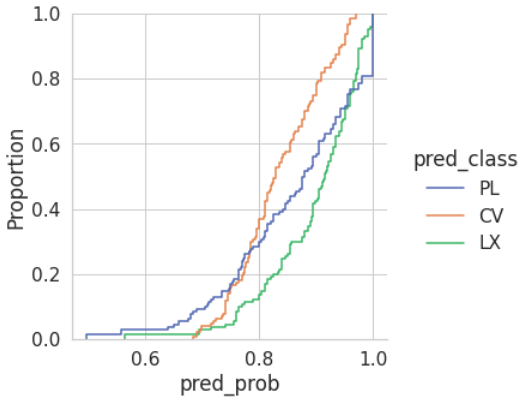


Figure 5.3: Confusion Matrix After SMOTE up-sampling for LMXB class



(a) ECDF before SMOTE Up-sampling



(b) ECDF After SMOTE up-sampling

Figure 5.4: Empirical cumulative distribution function (ECDF) Plot. On X-axis it shows the predicted probability , y-axis shows the proportion of cases in which the predicted probability is less than the probability value on x-axis.

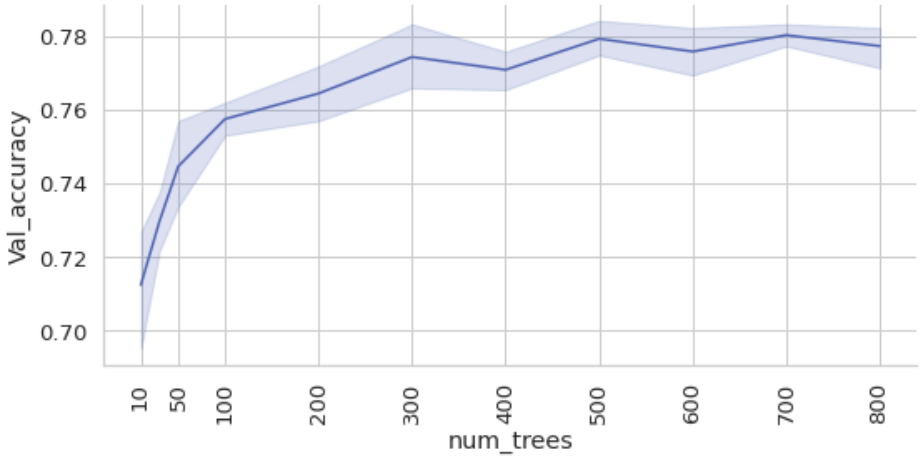
From the figure 5.3, we can see that the number of LMXB classified as LMXB has increased. Also from the predicted probabilities plot , we can see that the confidence of network towards LMXB has increased which earlier was $\sim$ 80% predictions were below 0.8 , has now come down to only 20% .

5.2 Hyper Parameter tuning

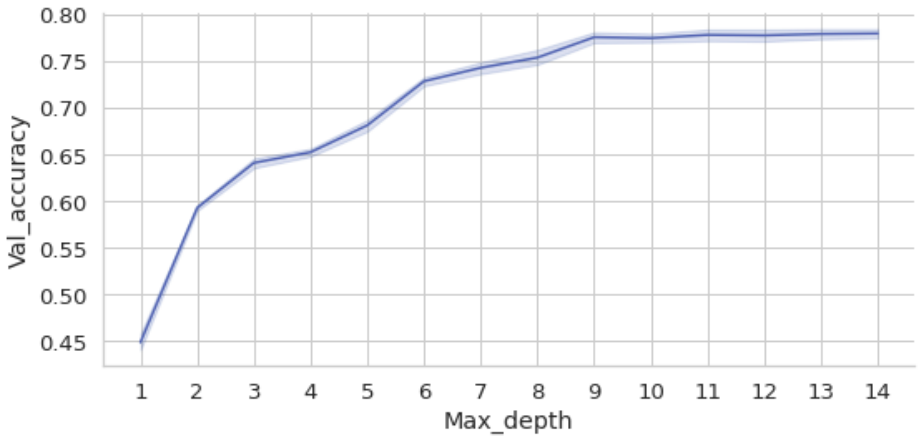
Random forest is a very robust algorithm with very few adjustable hyper-parameters. Hyper parameters to tune :

- Number of decision trees
- Maximum depth of decision tree
- Minimum sample split
- Minimum Sample Leaf

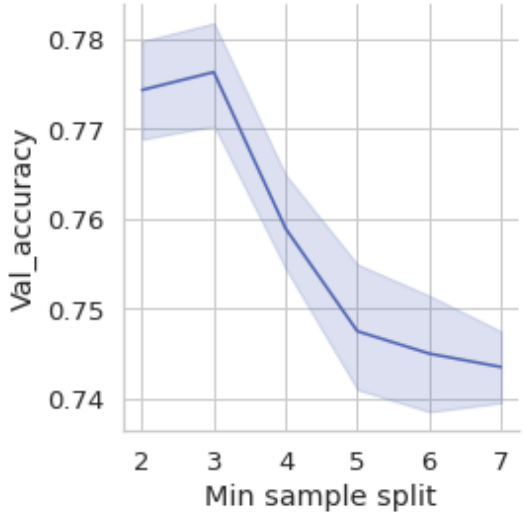
Monte-Carlo cross validation is run for each hyper-parameter by keeping all other parameters constant and varying one of the parameters. For each set of such hyper-parameters in each iteration, new RF is created and trained on randomly selected data from 70% of dataset, and rest 30% is used to calculate validation accuracy.



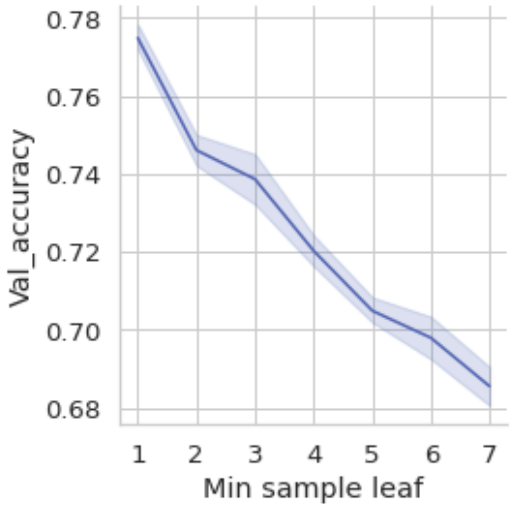
(a) Variation of validation accuracy with number of decision trees in random forest



(b) Variation of validation accuracy with maximum depth of decision trees



(c) validation accuracy variation with Minimum number of samples required to split at each node



(d) validation accuracy variation with Minimum number of samples required to be at a leaf node

Figure 5.5: Monte Carlo cross validation score for varying values of hyper-parameters form Random forest. Line and shaded region shows the validation accuracy and standard deviation in validation accuracy for each set of hyper-parameters as indicated in different sub-figures.

5.2.1 Best hyper parameters

From the figure 5.5 , following observations are made

- With increasing number of decision trees , validation accuracy increase and gets saturated at about 500 trees.
- At about 500 trees, the variance in accuracy is also low.
- till max depth of 9 accuracy increases , after which it saturates
- Max depth appears to be a very important hyper parameter , as with varying max-depth accuracy varies from 45 to 78%.
- Accuracy does not much depend on sample split , with varying only from 74 - 78 %
- With increasing number of min sample required at each leaf node , accuracy continues to drop , from 78 to 68 %.

- Number of decision trees - 500
- Max-depth 10
- Min sample split - 3
- min sample leaf - 7

5.3 Feature selection

Currently after removing feature s with sparsity more than 50% we have 49 features. Current results are for these features. More details of these features are given in appendix.

5.3.1 Remove correlated features

From the figure 5.8 , it is evident that several features are highly correlated. Which means that features are redundant and can be removed.

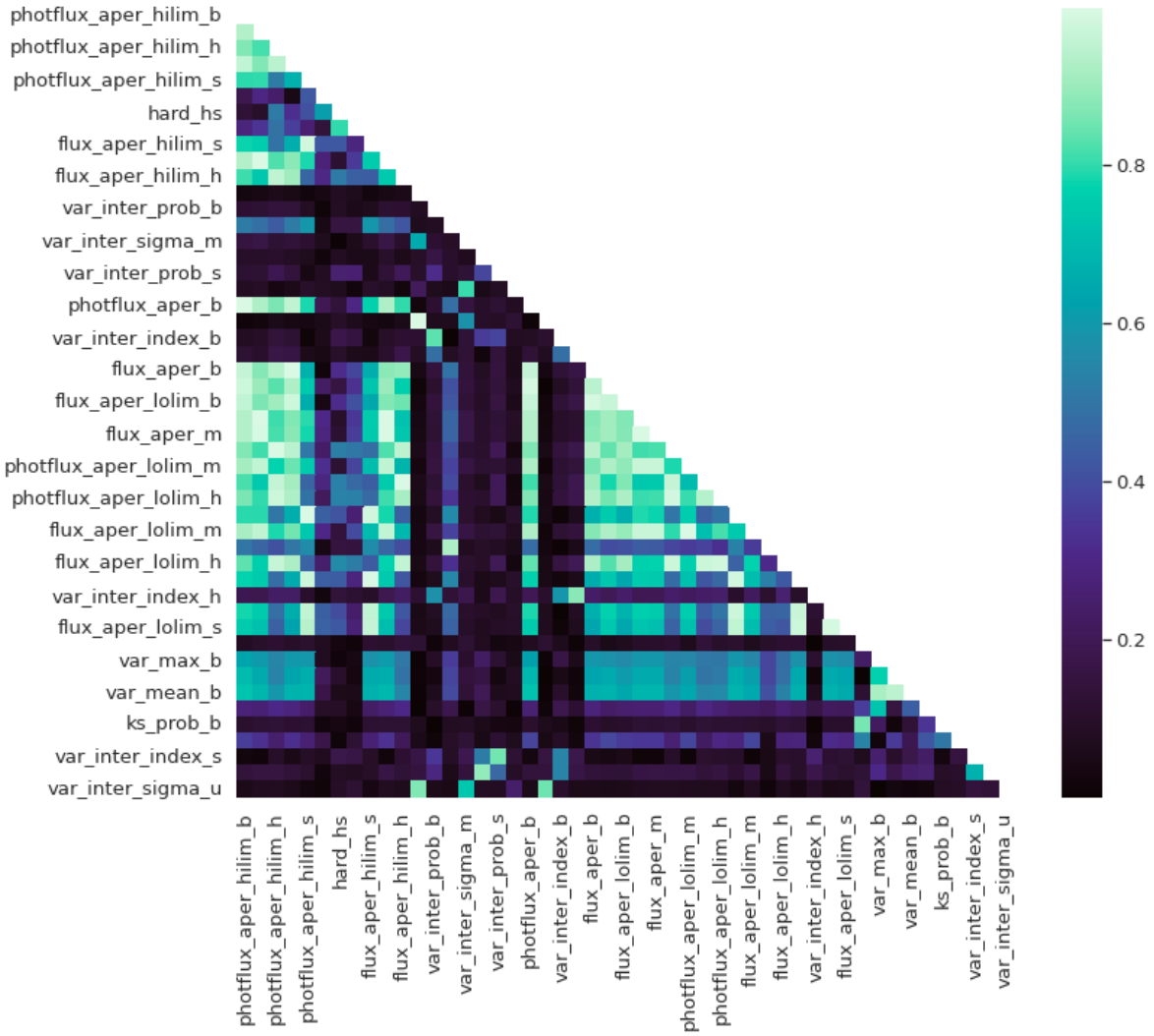


Figure 5.6: Feature feature correlation. Color map show the absolute value of correlation coefficient between pair of features. For clarity not all the features are labelled on x and y axis.

We removed features having correlation more than 0.85 . After removal total number of features we have is 19. Classification validation accuracy on these less correlated , small subset of features is examined and the results are tabulated in 5.3

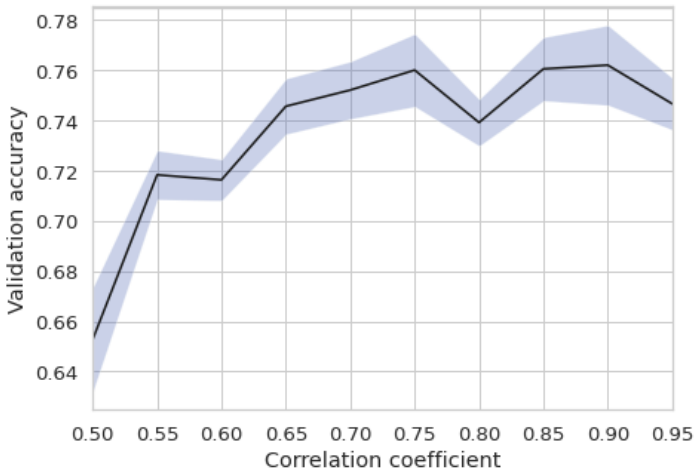


Figure 5.7: On X-axis shows the validation accuracy with subset of feature selected by removing features having correlation coefficient above the threshold given on x-axis

From the figure 5.7 , it is clear that after removing features with correlation coefficient above 0.85 , we get maximum accuracy . However even with lower thresholds (up-to - 0.75) also we get almost similar accuracy.

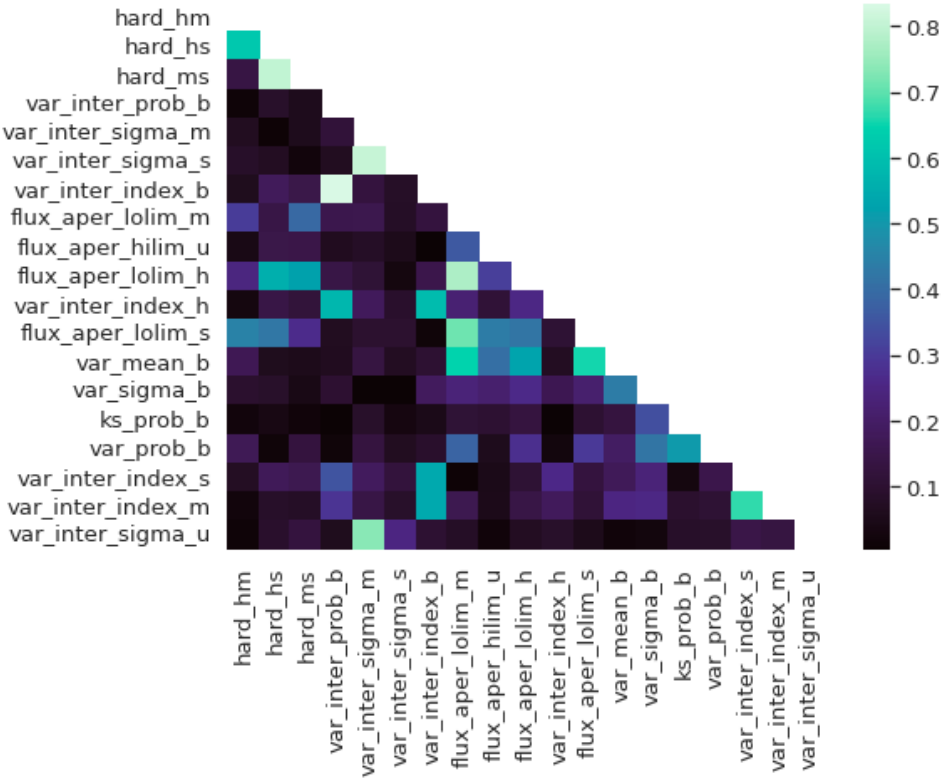


Figure 5.8: Feature feature correlation after removing correlated features. Color map show the absolute value of correlation coefficient between pair of features. For clarity not all the features are labelled on x and y axis.

Table 5.3: Validation accuracy before and after removing correlated features

	Number of features	Mean accuracy	Std accuracy
Before removing correlated features	49	76.2	1.2
After removing correlated features	19	77.3	1.1

From the table we can conclude that after removing correlated feature , even with a relatively small subset of feature (19) we are able to get similar accuracy within small variance.

5.3.2 Feature Importance

Here we calculate permutation based feature importance. One by one one of the feature value for entire training set is randomized and then classifier is fit and validation score is calculated. For feature which are important for classification , this randomisation of feature results in higher drop in accuracy and thus is reasoned as important feature. The relative drop in accuracy can be quantified as feature relative feature importance.

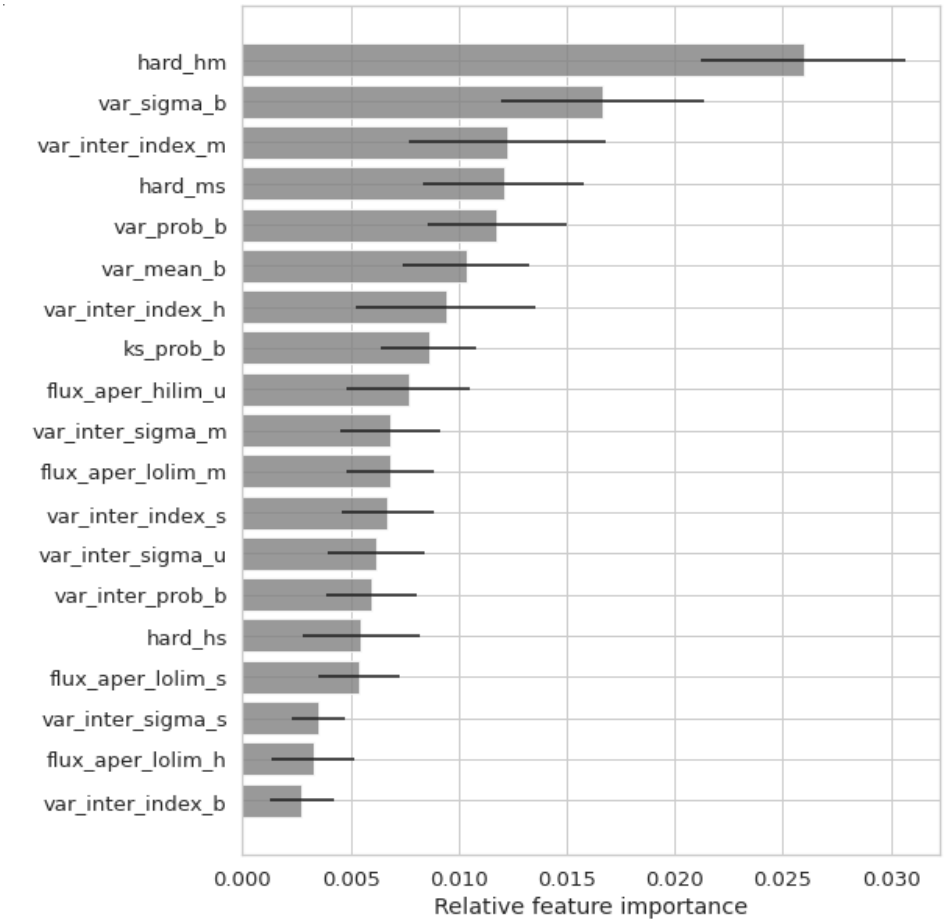


Figure 5.9: Relative permutation feature importance.

Class-wise feature importance

By Using classifier considering classification as **one vs rest** , feature importance can be computed for each class.

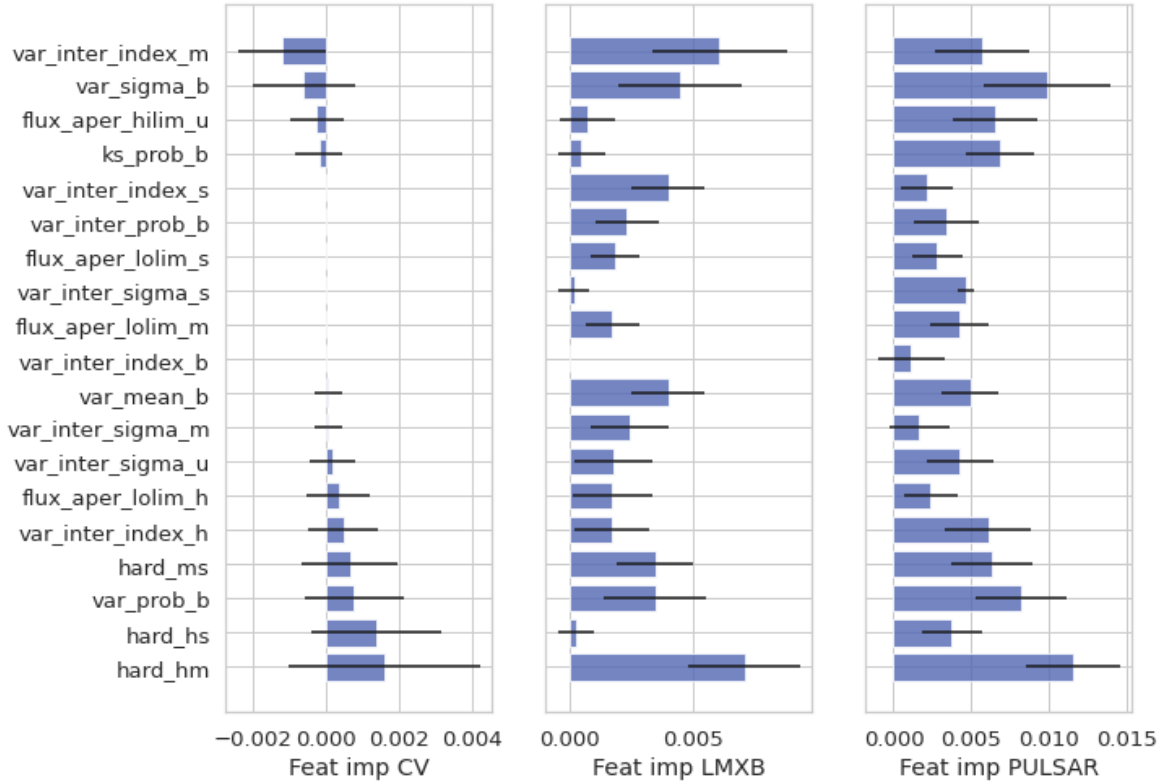


Figure 5.10: Relative permutation feature importance. Three panels shows relative feature importance for classification as one vs rest for each class, CV , LMXRB , PULSAR from left to right

5.3.3 Classification Significance

We need to verify whether the classifier has learnt class-feature relation or the accuracy obtained is just by chance. The idea is for the entire dataset , class label is reshuffled row-wise and then K-fold cross validation is performed. N Null distribution of accuracies are generated with features remaining the same but class labels are permuted. 1000 such permutations are generated. Null hypothesis is that there exist no dependence on class labels and feature dataset. This null hypothesis is quantified by a p-value which is the probability that the for a given class label permutation the accuracy will be higher than or equal to that for original data.[15][17]

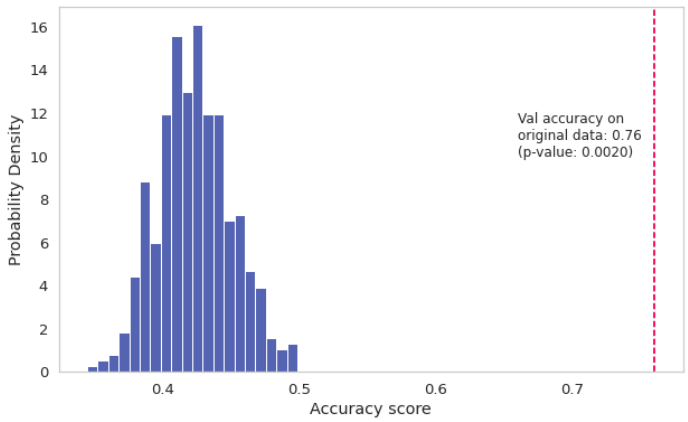


Figure 5.11: Permutation significance plot. Probability density plot for cross validation accuracies for 500 random permutations of class labels. Accuracy on original data , with non-permuted label is shown by red vertical line.

From the Histogram plot permutation accuracies have a distribution around 0.4 , and p-score for our null hypothesis is 0.002. Hence it is highly unlikely that the accuracy we are getting is just by chance. And we can conclude that there exist a relation between selected features and class labels and our classifier is indeed learning it.

Chapter 6

Conclusion

APPENDIX

Bibliography

[1] D. Pooley, Pooley, and D. "Globular cluster X-ray sources". In: *MmSAI* 87 (2016), p. 547. ISSN: 0037-8720. URL: https://ui.adsabs.harvard.edu/abs/2016MmSAI..87..547P/abstract (pages 1, 2, 9).

[2] Souradeep Bhattacharya, Craig O. Heinke, Andrey I. Chugunov, Paulo C.C. Freire, Alessandro Ridolfi, and Slavko Bogdanov. "Chandra studies of the globular cluster 47 Tucanae: A deeper X-ray source catalogue, five new X-ray counterparts to millisecond radio pulsars and new constraints to r-mode instability window". In: *Monthly Notices of the Royal Astronomical Society* 472 (3 2017), pp. 3706–3721. ISSN: 13652966. DOI: 10.1093/MNRAS/STX2241 (page 1).

[3] B. Gendre, D. Barret, and N. A. Webb. "An XMM-Newton observation of the globular cluster Omega Centauri". In: *Astronomy and Astrophysics* 400 (2 2003), pp. 521–531. ISSN: 00046361. DOI: 10.1051/0004-6361:20021845. URL: https://ui.adsabs.harvard.edu/abs/2003A&A...400..521G/abstract (pages 1, 2).

[4] David Pooley. "Globular cluster x-ray sources". In: *PNAS* (2009). DOI: 10.1073/pnas.0913903107. URL: www.pnas.org/cgi/doi/10.1073/pnas.0913903107 (page 2).

[5] Eugenio Carretta, Raffaele G Gratton, Osservatorio Astronomico, and Di Bologna. "DISTANCES, AGES, AND EPOCH OF FORMATION OF GLOBULAR CLUSTERS1 GISELLA CLEMENTINI AND FLAVIO FUSI PECCI2". In: *THE ASTROPHYSICAL JOURNAL* 533 (2000), pp. 215–235 (page 2).

[6] Ryan Jackim, Paula Szkody, Bryna Hazelton, and Noah C. Benson. "The Open Cataclysmic Variable Catalog". In: *Research Notes of the AAS* 4 (12 Dec. 2020), p. 219. ISSN: 2515-5172. DOI: 10.3847/2515-5172/ABD104. URL: https://iopscience.iop.org/article/10.3847/2515-5172/abd104%20https://iopscience.iop.org/article/10.3847/2515-5172/abd104/meta (pages 2, 10).

[7] "Formation and evolution of binary and millisecond radio pulsars". In: *PhR* 203 (1-2 1991), pp. 1–124. ISSN: 0370-1573. DOI: 10.1016/0370-1573(91)90064-S. URL: https://ui.adsabs.harvard.edu/abs/1991PhR...203....1B/abstract (page 2).

[8] Claire S Ye, Kyle Kremer, Sourav Chatterjee, Carl L Rodriguez, and Frederic A Rasio. "Millisecond Pulsars and Black Holes in Globular Clusters". In: *The Astrophysical Journal* 877 (2019), p. 122. DOI: 10.3847/1538-4357/ab1b21. URL: https://doi.org/10.3847/1538-4357/ab1b21 (page 2).

[9] Sean A. Farrell, Tara Murphy, and Kitty K. Lo. "AUTOCLASSIFICATION OF THE VARIABLE 3XMM SOURCES USING THE RANDOM FOREST MACHINE LEARNING ALGORITHM". In: *The Astrophysical Journal* 813 (1 Oct. 2015), p. 28. ISSN: 0004-637X. DOI: 10.1088/0004-637X/813/1/28. URL: https://iopscience.iop.org/article/10.1088/0004-637X/813/1/28%20https://iopscience.iop.org/article/10.1088/0004-637X/813/1/28/meta (page 3).

[10] Leo Breiman. "randomforest2001". In: *Statistics Department University of California Berkeley, CA 94720* (2001). URL: https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf (pages 3, 17).

[11] M. A. Tucker, B. J. Shappee, T. W.-S. Holoien, K. Auchettl, J. Strader, K. Z. Stanek, C. S. Kochanek, A. Bahramian, Subo Dong, J. L. Prieto, J. Shields, Todd A. Thompson, John F. Beacom, L. Chomiuk, L. Denneau, H. Flewelling, A. N. Heinze, K. W. Smith, B. Stalder, J. L. Tonry, H. Weiland, A. Rest, M. E. Huber, D. M. Rowan, K. Dage, D. M. Rowan, and K. Dage. "ASASSN-18ey: The Rise of a New Black Hole X-Ray Binary". In: *The Astrophysical Journal Letters* 867 (1 Oct. 2018), p. L9. ISSN: 2041-8205. DOI: 10.3847/2041-8213/AAE88A. URL: https://iopscience.iop.org/article/10.3847/2041-8213/aae88a%20https://iopscience.iop.org/article/10.3847/2041-8213/aae88a/meta (page 11).

[12] William E. Harris, Harris, and William E. "A Catalog of Parameters for Globular Clusters in the Milky Way". In: *AJ* 112 (4 Oct. 1996), p. 1487. ISSN: 0004-6256. DOI: 10.1086/118116. URL: https://ui.adsabs.harvard.edu/abs/1996AJ....112.1487H/abstract (page 12).

[13] C. Menéndez, J. B. Ordieres, and F. Ortega. "Importance of information preprocessing in the improvement of neural network results". In: *Expert Systems* 13 (2 1996), pp. 95–103. ISSN: 02664720. DOI: 10.1111/J.1468-0394.1996.TB00182.X (page 13).

[14] Ji Zhu, Hui Zou, Saharon Rosset, and Trevor Hastie. "Multi-class AdaBoost *". In: *Statistics and Its Interface* 2 (2009), pp. 349–360 (page 18).

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (pages 23, 31).

[16] Jason Brownlee. "SMOTE for Imbalanced Classification with Python". In: (2020). URL: https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/ (page 24).

[17] Markus Ojala@tkk Fi and Gemma C Garriga. "Permutation Tests for Studying Classifier Performance Markus Ojala". In: *Journal of Machine Learning Research* 11 (2010), pp. 1833–1863 (page 31).