Chronic Kidney Disease Detection using Ensemble Classifiers and Feature Set Reduction

by

Naveed Rahman Shawan
13201027
Syed Samiul Alam Mehrab
16101175
Fardeen Ahmed
15101073
Mohammad Sharatul Hasmi
15101133

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
Spring 2019

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at BRAC University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Naveed Rahman Shawan 13201027	Syed Samiul Alam Mehrab 16101175
Fardeen Ahmed 15101073	Mohammad Sharatul Hasmi 15101133

Approval

The thesis titled "Chronic Kidney Disease Detection using Ensemble Classifiers and Feature Set Reduction" submitted by

1. Naveed Rahman Shawan (13201027)
2. Syed Samiul Alam Mehrab (16101175)
3. Fardeen Ahmed (15101073)
4. Mohammad Sharatul Hasmi (15101133)

Of Spring, 2019 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on April 25, 2019.

Examining Committee:

Supervisor:

Mr. Hossain Arif
Assistant Professor
Department of Computer Science and Engineering
BRAC University

Co-Supervisor:

Dr. Md. Iftekharul Mobin
Assistant Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:

Md. Abdul Motallib, PhD
Professor and Chairperson
Department of Computer Science and Engineering
BRAC University

By the grace of Almighty Allah, we have compiled our thesis with the help of our Supervisor Mr. Hossain Arif and Co Supervisor Dr. Md. Iftekharul Mobin We would like to dedicate this thesis to our loving parents, teachers and all our well wishers

Acknowledgements

We would like to thank Mr. Hossain Arif and Dr. Md. Iftekharul Mobin for agreeing to supervise us with our thesis. Their patience and confidence in us have been a source of encouragement and this thesis would not have been possible without their constant support, inspiration and guidance. We believe their dedication to this paper deserves to be reciprocated with great gratitude. In addition, we want to thank our parents for the financial assistance and motivating us during this entire paper. Finally, we want to thank our team members, fellow class mates and friends for their valuable opinions and ideas. All in all, this token of gratitude is for everyone who helped us directly and indirectly in this paper. A special thanks to the Thesis committee for taking the time to review and evaluate our thesis as part of our undergraduate program.

Abstract

Chronic kidney disease (CKD) is the gradual loss of kidney function over a duration of months or years. One in ten people are affected by it at some stage. Some ethnicities such as African Americans and South Asians are predisposed to having the disease. Globally the number of people affected has been growing through the years, with 752.7 million having the disease in 2016 The disease has no cure, so early detection is key to better manage the disease and control other risk factors such as diabetes and blood pressure. Although CKD has no early symptoms and requires medical tests on blood and/or urine samples, medical tests conducted for other diseases hold clues to whether someone has CKD . The datasets that are available have a multitude of features and are also incomplete and imbalanced. We want to overcome this problems through feature engineering to reduce the number of features. A comparative study of various classifiers needs to be done to find those that hold promise and are robust enough to handle currently available datasets, which are both incomplete and unbalanced. Our study is to bring down the number of attributes/features using recursive feature elimination method and use Ensemble classifier to predict the existence of CKD from the reduced features.

Keywords : Ensemble Learning, Imbalanced Dataset, Supervised Learning, Chronic Kidney Disease, Machine Learning

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

CKD Chronic Kidney Disease

CSP Common Spatial Pattern

KNN K-Nearest Neighbor Model

LDA Linear Discriminant Analysis

NaN Not a Number

REF Recursive Feature Elimination

RFECV Recursive Feature Elimination with Cross Validation

RF Random Forest

SMOTE Synthetic Minority Over–sampling Technique

SVM Support Vector Machine

Chapter 1

INTRODUCTION

CHRONIC kidney disease (CKD) also called chronic renal disease is a condition in which kidneys gradually lose their function. If the kidney does not function properly, this could cause waste and excess fluid accumulation in the body, affecting its functionality, and potentially leading to complications. The disease can progress to end-stage renal disease which is complete kidney failure. This occurs when kidney function is worsened to a point where dialysis or kidney transplantation is required for survival. People with CKD also have an increased risk of developing cardiovascular diseases (CVD) [18] Currently, 4 out of every 1000 person in the United Kingdom are suffering from kidney failure and more than 300,000 American patients are in the last-stage of kidney disease who are surviving with the help of dialysis [1] . Moreover, according to the National Health Service, kidney diseases are more prevalent in South Asia, Africa compared to other countries around the globe. As detection the chronic kidney failure is not possible until the kidney failure is at its end stage, identifying the kidney failure in the first stage is extremely crucial. [18] Through early diagnosis, the condition of each kidney can be taken under control, which decreases the risk of incurable consequences. For this reason, routine check-up and early diagnosis are mandatory for patients for preventing vital risks of renal failure and other related diseases [1] . Blood test is one of the most significant steps to detect CKD. It be easily distinguished by measuring factors, and physicians and doctors can identify treatment processes for reducing the rate of deterioration of the kidney [8] . Kidney imaging can be process which can be used to affirm the possibilities of the existence of the disease, although due to very small numbers, it is not feasible for everyone to undergo the test, but only those who have higher chances to have CKD may be strongly recommended [16]

1.1 Motivation

Chronic kidney disease has grown significantly and now become a global problem. It has become alarming for us. There are some specific attributes to which chronic kidney disease appears and harms us. Here we felt the necessity to work on it. Also the most of the available dataset has imbalanced values. If we can overcome this problems through our research to reduce the number of features as well as doing a comparative study of various classifiers would be helpful for medical specialist in working with chronic kidney disease. With the recent development in Machine Learning field, the scope of performing in different sectors and concluding with better accuracy and optimized performance has increased. Medical science has also improved over time. Considering all these factors, we have decided to do our thesis on data science and machine learning technique to contribute something in modern medical science which ultimately defines the betterment of mankind.

1.2 Objective

In this paper our objective is to reduce the number of features from our dataset consisting of 24 features which determines the possibilities of CKD. To do such feature engineering we will use multiple techniques like Recursive Feature Elimination which is a feature selection method that recursively removes features of low importance and creates a model with the remaining ones. This is done to increase accuracy and reduces the complexity of the model. To predict CKD, Ensemble classification methods will also be used to make our model more robust and reliable as we will be judging this model based on accuracy and the number of false positives.

1.3 Thesis Outline

The rest of the discussions is organized as follows:

Chapter 2 contain the literature review is divided into separate paragraphs addressing all the parts of the study and the papers we have seen in each paragraphs. The first para talks about the feature selection process and about the papers whose workings we have used for feature selection. Para 2 talks about the imbalance part and how other relevant papers have worked to address this problem. Para 3 and para 4 talks about the different algorithms used and how they have worked with imbalance or missing data.

Chapter 3 includes implementation part which has 8 sub chapters. 3.1 includes algorithm part which has 7 subsections which are basically the algorithms used in our study. 3.2 talks

about the dataset which shows the number of data and brief description of all the attributes used in the dataset. 3.3 talks about eh methods of data preprocessing methods used in the study. 3.4 helps us visualize the data through Violin plot graph. 3.5 shows the correlation heatmap which shows the relationship between the attributes. 3.6 Frame work shows our implementation workflow through description and flow chart

Chapter 4 shows the results of the study and analysis of the results. 4.1 Feature selection process described in this chapter where we show the different methods used to do it. 4.2 describes the results of the algorithms used and show their diagram and tables. 4.3 is result analysis where we compare the results of the algorithms and make the statement about our study

Chapter 5 is conclusion where we speak about our results and our challenges faced to achieve it. 5.2 talks about our future plans with the thesis and about implementations of it in the future

Chapter 2

LITERATURE REVIEW

Feature selection is a vital part in discovery of knowledge, recognition of patterns. The main purpose of feature selection is to remove feature subsets from inputs which are not important for the study or do not cause major change to the results of the study. A significant problem for feature reduction is identifying the best subset of features to get the best results for classifications . Feature selection can decrease overfitting problems and the volume of data storage, and also it can bring down the cost of training to obtain higher accuracy [3]. Sedighi in his paper examined the effect of feature selection in chronic kidney disease classification. Some filter and wrapper based feature selection techniques are compared in terms of classification accuracy with a special classification approach. Selecting a subset of features in some applications not only reduces the number of features, but also removes features that make noise or have low correlation with other features. This study compares some common feature selection methods and shows that genetic algorithm is a narrative way to select a subset of features using an ensemble classification. The classification accuracy with features subset obtained good results in comparison with the original features [2]. Similarly Polat in his paper has researched on the accuracy rate by data reduction using different methods of feature selection. Wrapper and filter methods based on Best First and Greedy stepwise search were constructed to test the feature selection methods and the accuracy of classification algorithms. SVM classification algorithm was applied on the data set for the diagnosis of CKD and then two methods of wrapper approach and two methods of filter approach were implemented for feature selection. These methods were used to reduce the dimensions of dataset through which higher accuracy of classification can be obtained in a shorter time. After performing feature selection methods and reducing dimensions of dataset, SVM algorithm was used for classification and diagnosis CKD. The accuracy rate of SVM on the lowest dimension of CKD dataset by 7 attributes by ClassifierSubsetEval with Greedy stepwise search engine is not the highest accuracy rate (98%), however the accuracy rate of

SVM classifier on 13 attributes of CKD dataset, by using FilterSubsetEval with Best First feature selection method, has got the most accuracy rate (98.5%) in CKD diagnosis [12].

Anandanadarajah Nishanth had a different approach on his paper. The importance of features is determined in two ways. First using Common Spatial Pattern (CSP) and Linear Discriminant Analysis (LDA) and later using LDA and KNN on reduced subsets of features. CSP and LDA were sequentially applied to created a weighting vector to get the mean weighted difference to find features with the largest separating between classes. For the classification process, 2 method were used to break the feature set. One-Omit method, where iteratively 1 of the 18 features was eliminated and then the remaining 17 were used to classify using KNN and LDA. The eliminated features with most loss of accuracy we deemed more important. Similarly, Four-attribute combination was used. Subsets of 4 of the 18 features were created, and used to classify using KNN and LDA. Top 5% of the combinations with most accuracy were chosen.1. [20] The features were ranked in terms of the number of representations in the top 5% of the combinations. The overall rank of an attribute is the mean of its rank in all tests. Hemoglobin, Hypertension, Albumin, Serum Creatinine and Anemia were the top 5 attributes. For combinations of 4 features, Specific Gravity, Albumin, Diabetes Mellitus and Hemoglobin were most accurate for LDA, and for KNN Specific Gravity, Sugar, Serum Creatinine and Hemoglobin were most accurate [10]. In another paper consisting of datasets of 400 people (with a lot of noisy data) having 24 attributes over-fitting was used and to do feature reduction, the wrapper and LASSO regularization method was used which reduced it down to 10 attributes. Using random forest classifier gave an accuracy of 0.993 according to the F1-measure with a 0.1084 root mean square error. About 60% and 56% RMSE reduction compared to the Modification of Diet in Renal Disease (MDRD) equation [10] and the Chronic Kidney Disease Epidemiology Collaboration (CKD-EPI) equation was achieved. A cost effective classifier was identified using 5 attributes (specific gravity, albumin, diabetes mellitus, hypertension and hemoglobin) and achieved 0.98 F1-measure and 0.11 RMS [1].

Medical datasets are usually imbalanced with positive cases being the minority, so the overall accuracy of a classifier may be very high but when identifying the positive instances its accuracy may still be poor [19]. Min Zhu has seen that traditional practices to eliminate data imbalances are unsuitable as they can lead to loss of data or generating synthetic samples becomes complex and inaccurate. Random forest classifiers are better suited for imbalanced data and minimize classification errors. The general framework of CWsRF is a collection of multiple classifiers with weights for each individual class. The ensemble of classifiers are combined together in a voting classifier. While CWsRF may in some cases be less accurate then other classifiers in identifying majority class, it has always identified minority

classes more accurately in all test instances [19]. P.Yildirim performed different sampling algorithms and methods to predict chronic kidney disease. Sampling methods work well in dealing with imbalanced data and the ones used here are SMOTE, Resample and Spread Sub Sample. SMOTE creates artificial data based on the feature space similarities between existing minority examples . Re sample produces a random subsample of a dataset doing the sampling with replacement. Spread Sub Sample produces a random subsample with a given spread between class frequencies and samples with replacement [17]. Pinar Yildirim however has dealt with the imbalanced data in a different way in his paper. His aim was to help to identify the challenges in imbalanced data problems in health science and highlight the effects of learning rate parameter on multilayer perceptron model using back propagation algorithm. Initially he thought Sampling can be a solution. He tried Under-Sampling and Over-Sampling techniques but they could not came up with a satisfactory result. He then Multilayer Perceptron which is a class of feed forward artificial neural network. It consists of, at least, three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. Back propagation is the most widely applied learning algorithm for multilayer perceptron in neural networks. Back propagation employs gradient descent to minimize the squared error between the network output values and desired values for those outputs. A problem with back propagation networks is that its convergence time is usually very long. The experimental result they used Chronic Kidney Disease dataset to compare different sampling methods for the prediction of disease. After the research work they had selected multilayer perceptron to evaluate classification accuracy [18].

Using data mining techniques were performed, one trained sub-set is tested out of 10 classifier and cross validation is arrived. Attributes for evaluating through NB were age, sex, smoking, alcohol, cholesterol HDL etc. Performance of the model is evaluated by True Positive and True Negative classifications. NB gave accuracy of 86%. Decision Tree provided an accuracy of 91% using information gain as split parameter

As can be seen, there are a variety of challenges to predicting CKD and the implications are crucial to the well being of a patient. Firstly, the importance of each feature will be determine followed by a feature study to reduce the number of features. For our prediction model, we will start using Random Forest Classifier, as it is better suited to handle imbalanced dataset. We will also be comparing against other classifiers to determine their merits. The key metric for us will be accuracy, but the number of False Negatives will be crucial as well.

Chapter 3

IMPLEMENTATIONS

3.1 Algorithms

3.1.1 Random Forest

The random forest approach [17] is an ensemble approach which can also be thought of as a method of predictor of nearest neighbor. These ensembles [11] are a kind of divide-and-conquer methodology which is generally used to enhance the performance. The main principle behind ensemble methods is that a group of 'weak learners' can come together to form a 'strong learner'. The random forest starts with a standard machine learning technique called a 'decision tree' which, in ensemble terms, corresponds to our weak learner. The random forest takes this concept to the next level by combining trees with the concept of an ensemble. Thus, in this terms, the trees are assumed as 'weak learners' and on the other hand the random forest is a 'strong learner'. The advantages of a random forest classifier are that it is very fast in processing whereas weaknesses of this algorithm are that when used for regression it cannot predict outside the range in the training data as well as it might be over fit and noisy.

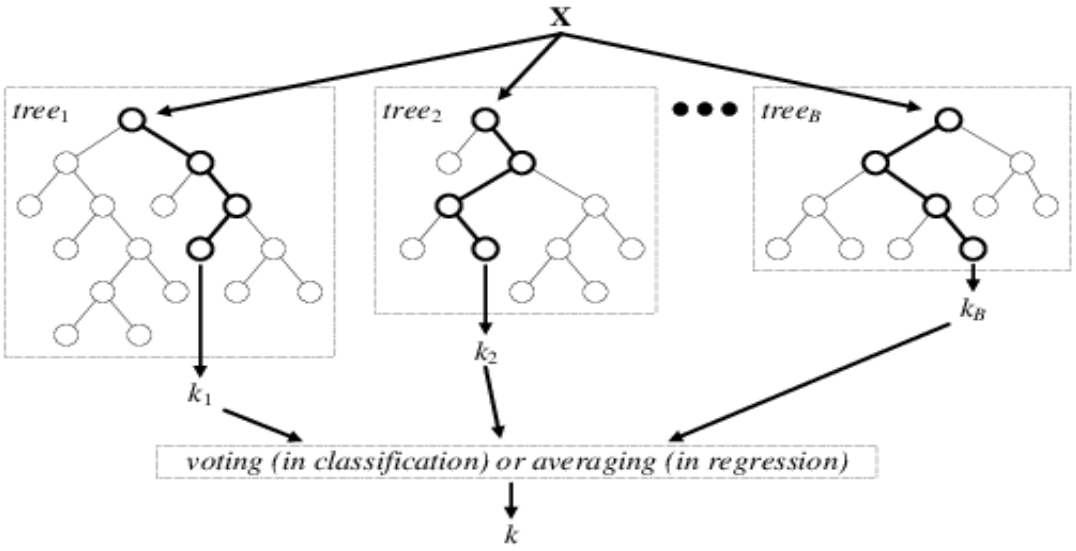

Fig. 3.1 Architecture of Random Forest Model

3.1.2 Naïve Bayes

A Naive Bayes classifier is a modest probabilistic classifier constructed on applying Bayes' theorem (from Bayesian statistics) with strong (naive) independence conventions. In order to elaborate the term it can be said that for the underlying probability model it would be "independent feature model". This controlled impartiality assumption occasionally holds true in applications in real word, here the classification as Naive yet the algorithm rises to perform well as well as learn rapidly in various classification problems which are supervised [10]. A benefit of this Naive Bayes classifier is that it only needs training data in small amount to estimate the constraints necessary for classification. Since autonomous variables are anticipated, only the variances of the variables for each class need to be determined and not the whole covariance matrix [14].
Here we can see the Bayes theorem:

1. P (C|X) = P (X|C) ·P(C) / P(X).
2. P(X) is constant for all classes.
3. P(C) = relative freq of class C samples c such that p is increased=c Such that P (X|C) P(C) is increased
4. Problem: computing P (X|C) is unfeasible!.

3.1.3 RECURSIVE FEATURE ELIMINATION

Recursive feature elimination is a type of feature selection method where it fits a model and deletes the weakest feature(s) until a certain desired number of features is reached.There is an external estimator that sets weights to features that are to be removed. [4] At first the estimator is trained on the initial set of features which are ranked in the order of their importance by the model's "coef_" or "feature_importances_" attributes.By recursively removing a small number of features every loop, RFE removes dependencies and collinearity that might be present in the model.

3.1.4 RFECV : RECURSIVE FEATURE ELIMINATION WITH CROSS VALIDATION

In any set of data it is a bad practice to reuse the same data for training and testing. Its is wiser to use the first 75% of the data for training and the last 25% of data for testing. Since we do not know which block of data to use for training and which one for testing cross-validation uses all the data blocks. Cross-validation uses one block of data at a time and summarizes the result.

3.1.5 UNIVARIATE FEATURE SELECTION

Univariate feature selection is a very powerful technique used to lower computational cost and to improve the a model's performance. Statistical tests are used to judge the relationship between each input feature and output feature. Input features that have a very good statistical tie with the output feature are selected while the others leftover features are rejected. [15]

3.1.6 CORRELATION

The statistical relationship between two variables is referred to as their correlation. A correlation could be positive, meaning both variables move in the same direction, or negative, meaning that when one variable's value increases, the other variables' values decrease. Correlation can be neutral when there is no relation between the variables[16].

3.1.7 SVM

Support vector machine is a form of a machine learning technique which is on the basis of theory of statistical learning. It has advantage of using 'kernel trick' which refers to the distance between a particle and the hyper plane can be calculated in a nonlinear feature space,

lacking of the unambiguous transformation of the original descriptors [5]. The radial basis function kernel which is the most commonly used was functional to this study.
The kernel function is expressed as follows:

$$K(\bar{x},\bar{xi}) = exp(-\frac{\left\|\vec{x}-\vec{xi}\right\|}{2a^2})$$

In the above equation (a), the kernel width parameters control the amplitude of the Gaussian function reflecting the simplification capability of SVM. The regularization parameter C is censurable for inhibiting transaction among maximizing the margin and minimizing the training error.

3.1.8 KNN

K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure. KNN has been used in statistical estimation and pattern recognition technique. A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. [6] If K = 1, then the case is simply assigned to the class of its nearest neighbor.

It should also be noted that all three distance measures are only valid for continuous variables. In the instance of categorical variables the Hamming distance must be used. It also brings up the issue of standardization of the numerical variables between 0 and 1 when there

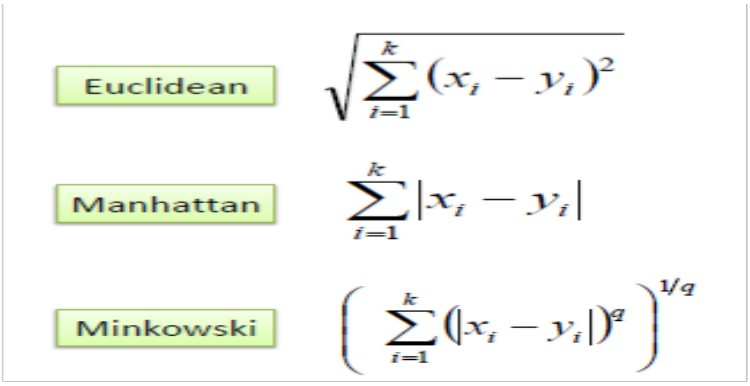

Fig. 3.2 Distance Functions

is a mixture of numerical and categorical variables in the dataset.

3.1.9 Logistic regression

Logistic regression is one of the best form of regression analysis to perform especially when the dependent variable is in binary form. We know regression analysis is a kind of predictive analysis, the logistic regression is as well a sort of predictive analysis. The most common uses of Logistic regression is to define data and to explicate the relationship among one binary variable (might be dependent) and one or more nominal, ordinal, interval or ratio-level independent variables. Logistic regressions are not easy to understand, the Intellects Statistics tool easily allows us to conduct the analysis, then in plain English interprets the output. Logistic regression uses an equation as the representation, very much like linear regression. [9] Input values (x) are combined linearly using weights or coefficient value to predict an output value (y). A key difference from linear regression is that the output value being modeled is a binary values (0 or 1) rather than a numeric value.

Here is an example of logistic regression equation:

$$y = \frac{e^{(b0 + b1*x)}}{1 + e^{(b0 + b1*x)}} \tag{3.1}$$

Where y is the predicted output, b0 is the bias or intercept term and b1 is the coefficient for the single input value (x). Each column in our input data has an associated b coefficient

that must be learned from our training data.

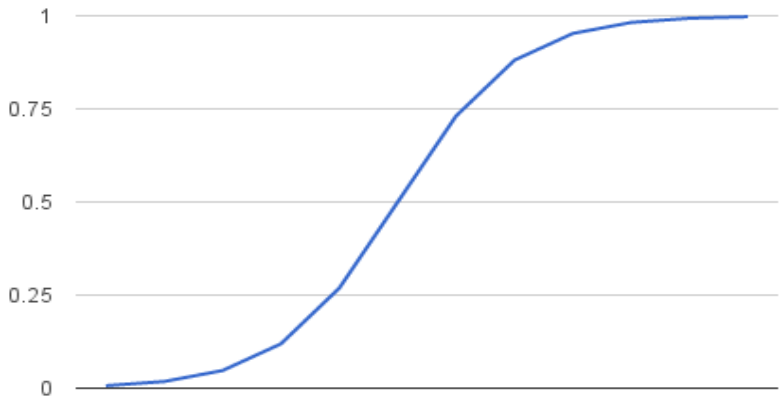

Fig. 3.3 1Logistic Function

3.2 The Dataset

The dataset we are using, was created in 2015 by Dr.P.Soundarapandian of Apollo Hospitals (Tamil Nadu, India). There are 400 instances with 25 attributes, of which 1 attribute is the class attribute. There 150 instances not having CKD, and 250 instances having CKD. 11 attributes are numerical and 14 attributes are nominal. Most attributes also have various amounts of missing entries. Of the 158 instances with complete entries, 115 instances do not having CKD and 43 have CKD. As we can see, not only does the dataset contain missing entries, it is unbalanced as well. Figure 01 show us how correlated each of the features are. Table 01 lists all features and the amount of missing entries in each.

Table 3.1 Attributes of the dataset

No.	**Attributes**	**Data Type**	**Value**	**Missing Entries**
01	Age (age)	Numeric	Continuous	9 (2%)
02	Blood Pressure (bp)	Numeric	Continuous	12 (3%)
03	Specific Gravity (sg)	Nominal	1.005, 1.01, 1.015, 1.02, 1.025	47 (12%)
04	Albumin (al)	Nominal	0,1,2,3,4,5	46 (12%)
05	Sugar (su)	Nominal	0,1,2,3,4,5	49 (12%)
06	Red Blood Cells (rbc)	Nominal	normal, abnormal	152 (38%)
07	Pus Cell (pc)	Nominal	normal, abnormal	65 (16%)
08	Pus Cell Clumps (pcc)	Nominal	present, not present	4 (1%)
09	Bacteria (ba)	Numeric	present, not present	4 (1%)
10	Blood Glucose Random (bgr)	Numeric	Continuous	44 (11%)
11	Blood Urea (bu)	Numeric	Continuous	19 (5%)
12	Serum Creatinine (sc)	Numeric	Continuous	17 (4%)
13	Sodium (sod)	Numeric	Continuous	87 (22%)
14	Potassium (pot)	Numeric	Continuous	88 (22%)
15	Haemoglobin (haemo)	Numeric	Continuous	52 (13%)
16	Packed Cell Volume (pcv)	Numeric	Continuous	71 (18%)
17	White Blood Cell Count (wc)	Numeric	Continuous	106 (27%)
18	Red Blood Cell Count (rc)	Numeric	Continuous	131 (33%)
19	Hypertension (htn)	Nominal	yes, no	2 (1%)
20	Diabetes Mellitus (dm)	Nominal	yes, no	2 (1%)
21	Coronary Artery Disease (cad)	Nominal	yes, no	2 (1%)
22	Appetite (appet)	Nominal	good, poor	1 (0%)
23	Pedal Edema(pd)	Nominal	yes, no	1 (0%)
24	Anemia(ane)	Nominal	yes, no	1 (0%)

3.3 Data Preprocessing

The dataset had multiple challenges for us. Apart from the small size of sample available, the dataset contains features of different data types, both numeric and categorical. Most features also have various amounts of missing entries. Using data preprocessing techniques, we first converted categorical data to discrete numbers using One-hot encoding and then dropped all patient entries missing any of the 24 features. This left us with 158 patient entries, with 115 negative cases and 43 positive cases, which was a very small number of samples and was also imbalanced. [7] The remaining data was normalized to the range of -1 to +1 using MinMax scaling.

3.4 Data Visualization

The following diagrams contain the boxplots of all the features in the dataset. The feature values have been normalized using Min-Max Scaling. Each feature has two boxplots, one with respect to each class. The classes have been coded as such, '0' representing patients without CKD and '1' representing patients with CKD.

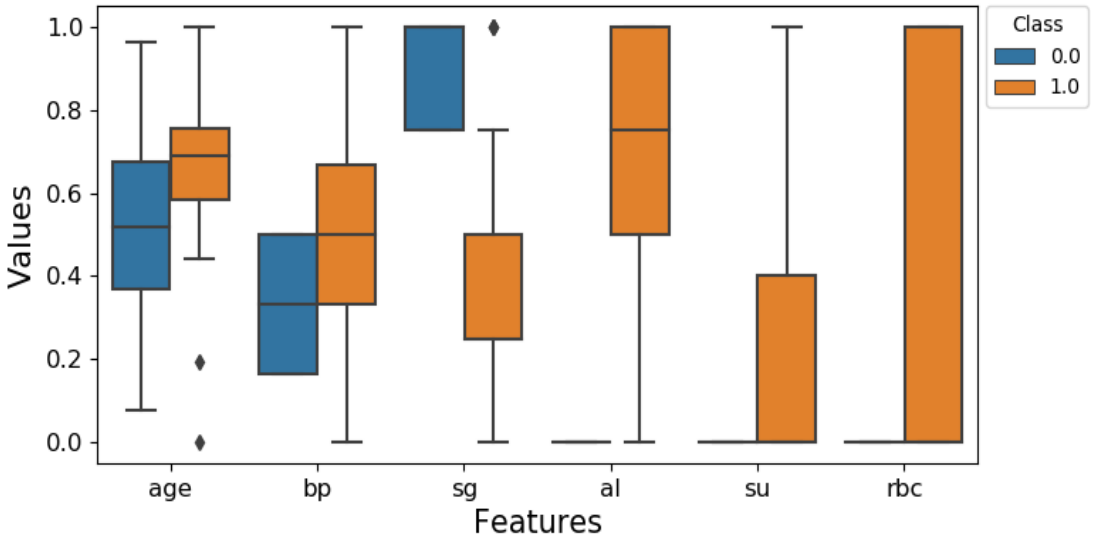

Fig. 3.4 Boxplots of features 1 through 6

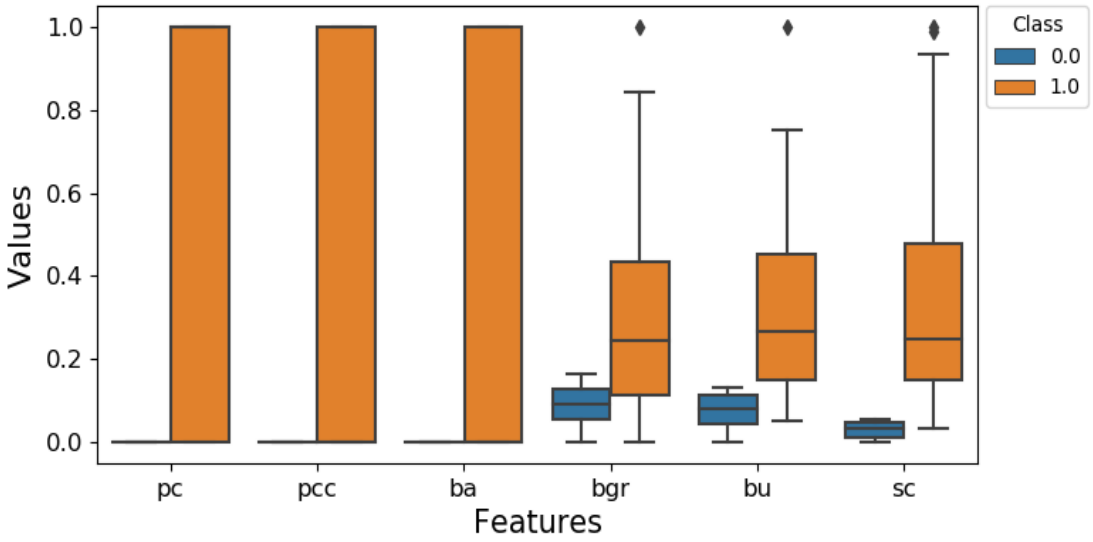

Fig. 3.5 Boxplots of features 7 through 12

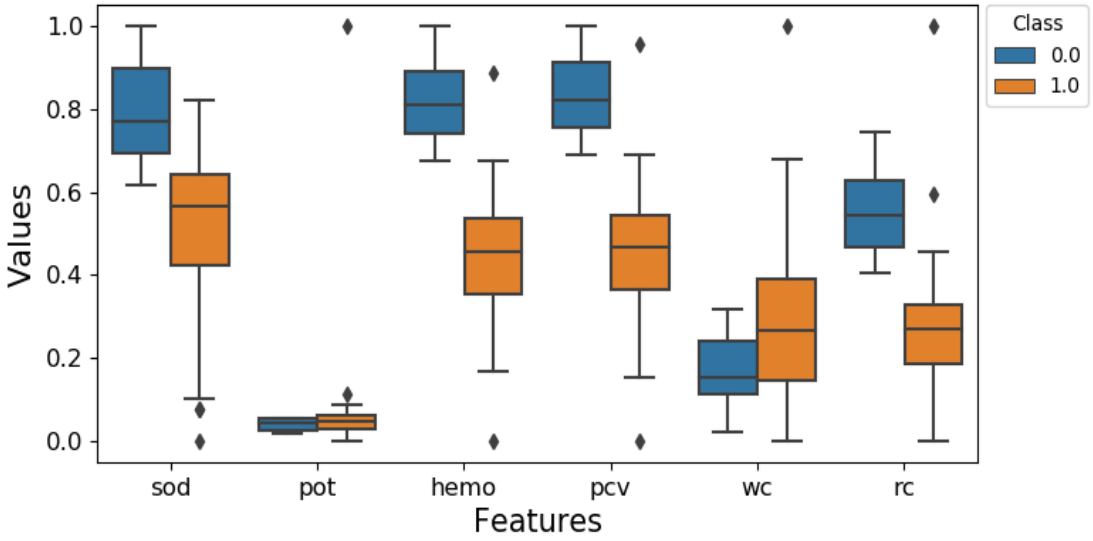

Fig. 3.6 Boxplots of features 13 through 18

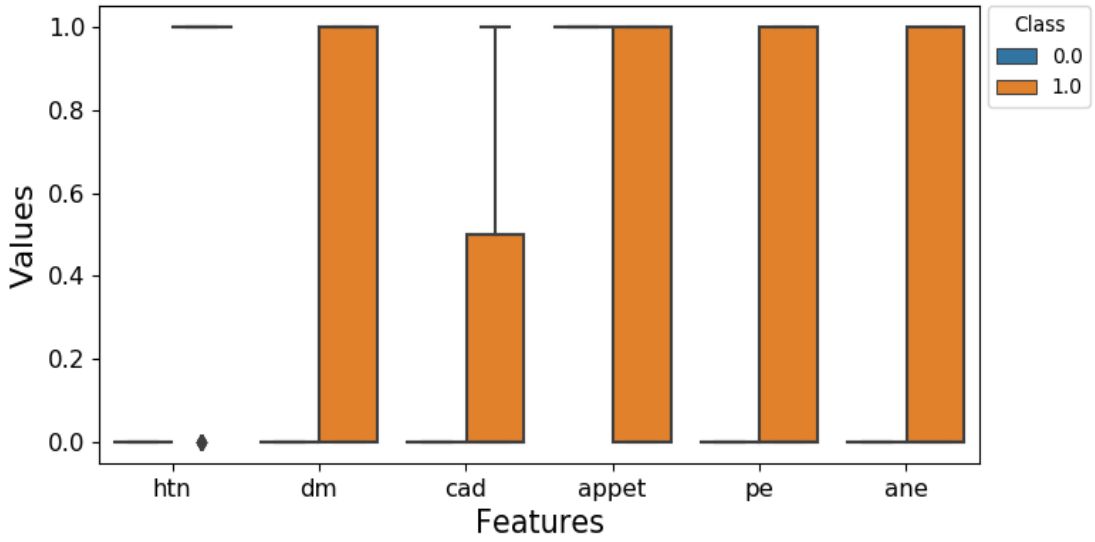

Fig. 3.7 Boxplots of features 19 through 24

The boxplots help us to visualize the distribution within each feature. The values in each feature have been normalized to the range of [0,1] using Min-Max scaling. Two separate boxplots for have been generated, one for each class. This helps us to see the distribution of the values with regards to class with in one feature. Higher degrees of separation suggest the feature can be a good candidate to be used for prediction. Also, is some of the nominal features we can see the values for class '0' converge to a single point, while the values for class '1' are distributed over a range. This means that feature is more suited to predict NotCKD cases.

3.5 Correlation HeatMap

In Fig 3.7, correlation heatmap shows us how connected the features are within themselves. Highly connected or correlated features can all be dropped except one with no loss in prediction accuracy. [13] This will help us in reducing the number of features that we will use for classification. As we see in the correlation heatmap, Specific Gravity (sg) and Hemoglobin (hemo) have a strong negative correlation with class. Albumin (al), Red Blood Cells (rbc), Hypertension (htn) and Diabetes Mellitus (dm) have moderate positive correlation to class. It can be expected that these features might be useful in classification purposes.

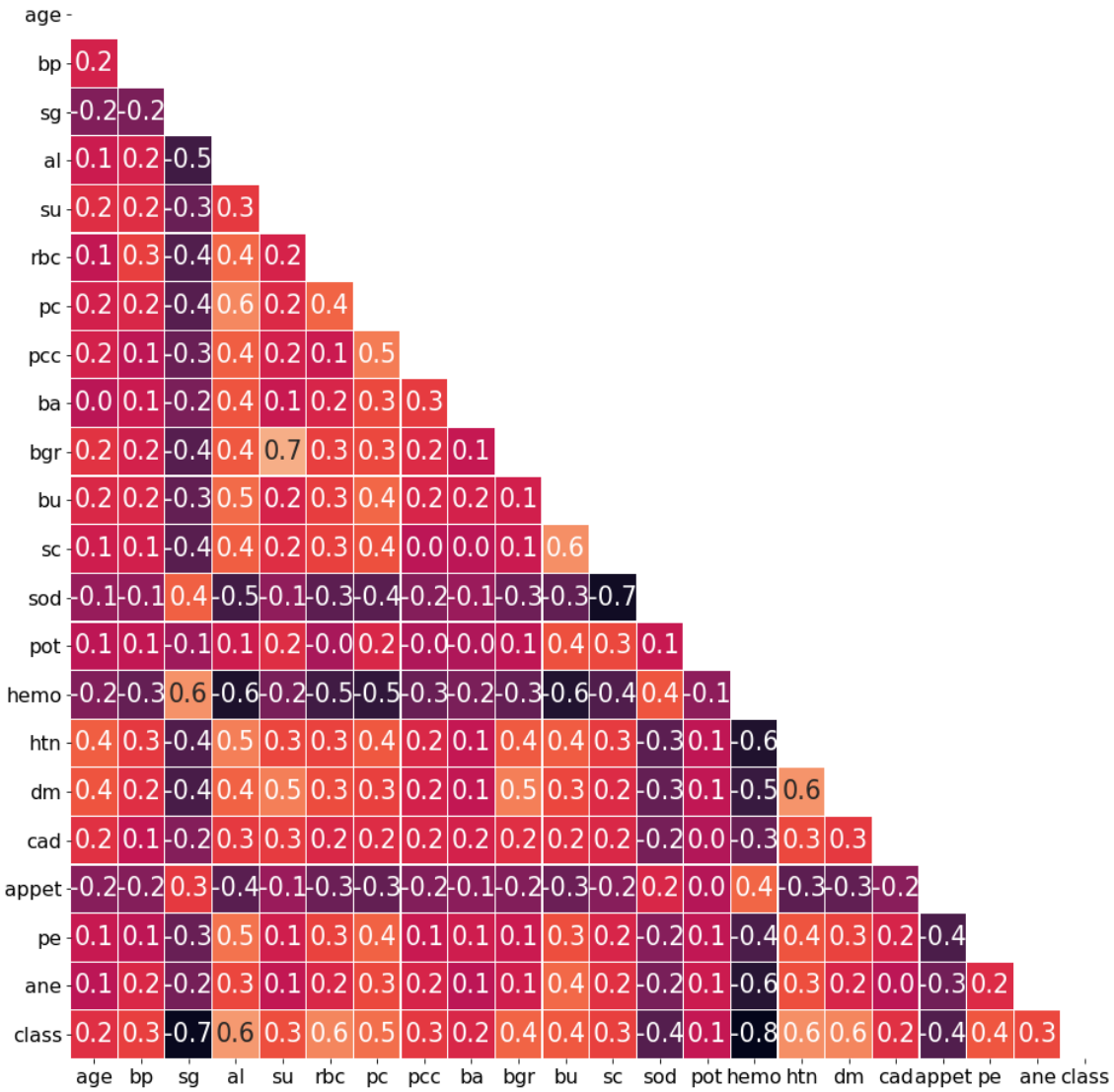

Fig. 3.8 Correlation Heatmap

3.6 Framework

For our model we have two key objectives. First, we want to reduce the number of features to be used in the classification process. Second, we want to build an Ensemble Classifier, such that it is able to handle small or imbalanced datasets with minimal loss of prediction capabilities.To reduce the number of features, we first need to determine the importance of each feature.For that purpose, we will be using Feature Selection Algorithms.The 2 algorithms we chose to use are Recursive Feature Elimination and Univariate Feature Selection.Recursive Feature Elimination iteratively generates feature subsets starting from a single feature and progressing to the maximum number of features available, as test the prediction capability of each subset to determine the subset with highest accuracy.We have used a Random Forest Classifier with 5-Fold Cross Validation for our model with accuracy being the differentiating metric.As the number of features is quite large, we will run 50 iterations of Recursive Feature Elimination to get an aggregate rank for each feature, with features with higher average rank being more important for correct prediction. The data used here will be from only the complete entries in our dataset.Univariate Feature Selection helps us to understand how much the separation of values within an individual feature correspond to the separation between classes.We will be using Chi-Squared Test for our model, and selected the same number of features as chosen in Recursive Feature Elimination.Using the two sets of features from the algorithms above we will reduce our dataset twice. Once keep only the features ranking highest in Recursive Feature Elimination and this new dataset will be referred to as Dataset A. Next, our dataset will be reduced again, this time keeping features scoring highest in Univariate Features selection and this dataset will be referred to as Dataset B.To build our ensemble classifier we have chosen 5 classification algorithms, Random Forest, Naïve Bayes, Support Vector Machine, K Nearest Neighbor and Logistic Regression.First, we will determine how suitable each classifier is for our dataset. Each classifier will be tested on both datasets generated through feature selection techniques. Each dataset will be split into a Train Set (70%) and a Test Set (30%). The classifiers will be put through 5-Fold Cross Validation and trained on the Train Set.The Test Set will be used to determine the accuracy of each classifier.The 3 most accurate classifiers and the corresponding dataset will be used to build our Ensemble Classifier.The purpose of the ensemble classifier is to combine the voted of each of its 3 inner classifiers.The “votes” or rather the class of an instance determined by each classifier is weighted equally and the majority class among the predictions is chosen as the class for that instance.The dataset will split into Train and Test Sets as before with a 70:30 split.The Train Set is used to 5-Fold Cross Validate and to train the ensemble classifier.The Test Set is used to determine the prediction accuracy of our Ensemble Classifier.

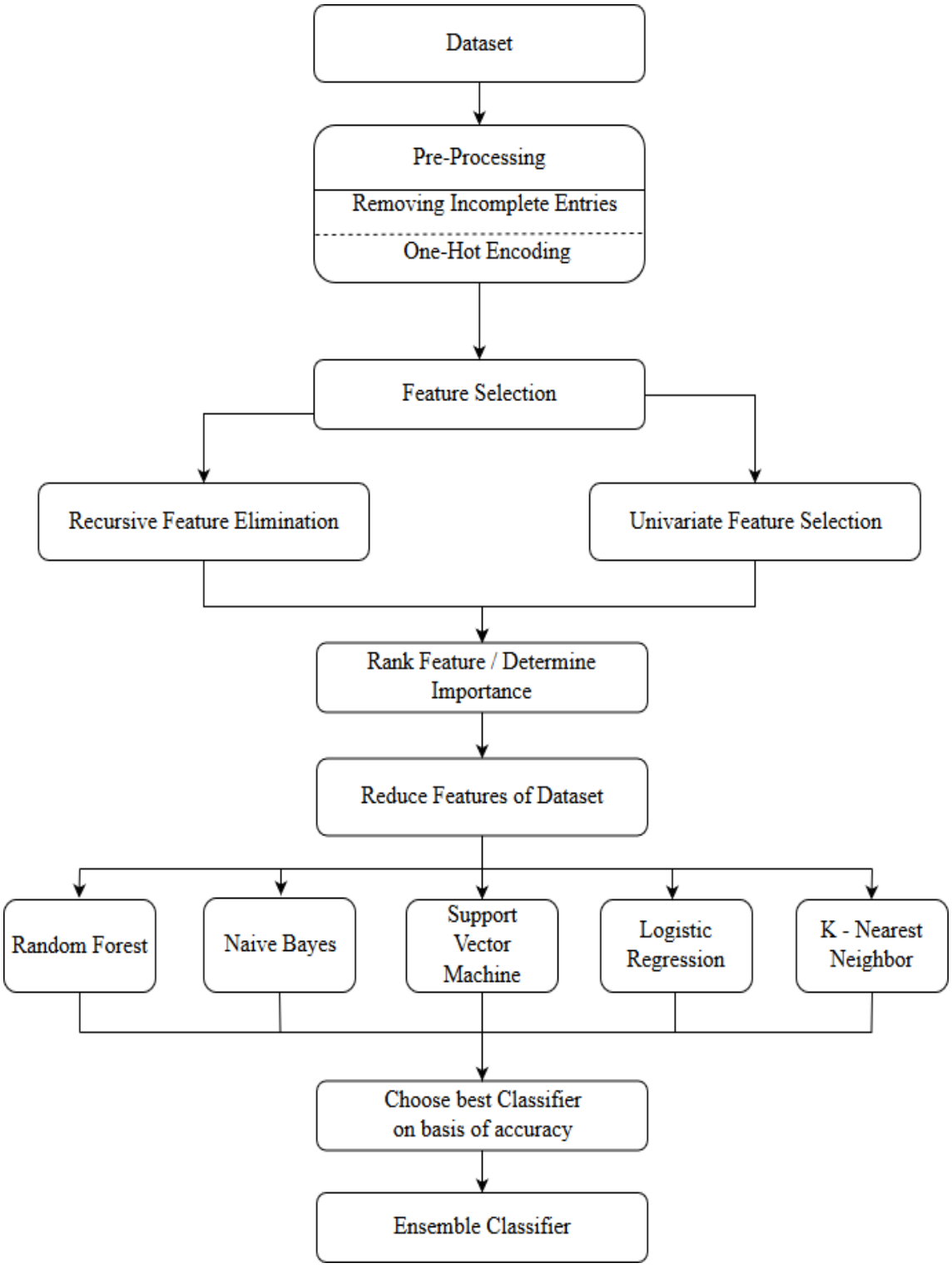

Fig. 3.9 Workflow Diagram

Chapter 4

RESULTS

4.1 Feature selection

Data Visualization and Correlation heat map shows the dependencies between the 26 attributes. As we are trying to find the best set of features which could predict the possibility of CKD, we will use:

1. Recursive feature Elimination
2. Univariate Feature Selection

4.1.1 Recursive feature Elimination

Recursive Feature elimination works by initially determining each features importance and the recursively prunes lower ranking features in each step. We Recursive Feature Elimination with 5-Fold Cross Validation with accuracy being the differentiating metric. We repeated 50 iterations of Recursive Feature Elimination, and recorded the aggregate rank of each feature. Features with the highest 10 ranks were chosen.

Table 4.1 Feature Ranking table

No.	Feature	Mean Rank
01	PCV	1.00
02	haemo	1.02
03	SC,WC	1.10
05	rc	1.12
06	ntn	1.22
07	al	1.24
08	bgr,bu	1.26
10	ba	1.32
11	dm	1.38
12	rbc,pcc,pot	1.40
15	pc	1.52
16	su	1.58
17	appet	1.78
18	dm	1.82
19	sg,sod	1.92
21	pe	1.96
22	bp	2.76
23	ane	2.88
24	age	3.78

The optimal features subset size at each iteration was on average 20, which was far higher than we wanted. So, we chose to keep only the 10 highest ranking features. The features were used to reduce our dataset. All features except those above were dropped with any remaining incomplete instances dropped as well. We were left with 102 instances of patients with CKD and 124 instances of patients not having CKD.

4.1.2 Univariate Feature Selection

Here we are evaluating each feature individually, and determine their importance towards classification. The features are scored using the Chi Squared test, which will help us to find out features that are independent of the class. Only the 10 highest scoring features will later be used for classification using Random Forest to validate our selection. The table below shows the 10 highest scoring features:

Table 4.2 Ten best scoring features from univariate feature selection

No.	Feature	Score
01	wc	30084
02	bu	3079.9
03	bgr	1762.0
04	al	336.98
05	sc	335.03
06	pcv	212.87
07	su	106.98
08	htn	90.930
09	pc	77.558
10	dm	74.884

We chose to determine only 10 highest scoring features to maintain consistency between the two feature selection algorithms. A new dataset generated by dropping all features except those above. After dropping any remaining incomplete instances in the dataset, we were left with 228 instances in the dataset. Of which 108 were CKD positive cases and 123 were CKD negative cases.

4.2 Results of different Algorithms

(replace this)We have used 5 algorithms as mentioned above as classifiers. The usage of these classifiers were due to our small dataset. The results of the 5 classifiers varies and there is where the ensemble classifier comes into act as the voting classifier to help us find the most accurate classifier. The results of all the classifiers are mentioned below:

4.2.1 Explanation for measuring metrics

Precision : all correctly identified members of a predicted class by the total number of members predicted to be in that class.

$$Precision = \frac{TP}{TP+FP} \tag{4.1}$$

Recall : all correctly identified members of a class divided by the total number of members in that class.

$$Recall = \frac{TP}{TP+FN} \tag{4.2}$$

F1-score: the harmonic mean of precision and recall. It is a measure of how robust the classifier is.

$$F1Score = 2 * \frac{Precision_x * Recall_x}{Precision_x + Recall_x} \tag{4.3}$$

Support : the number of member or instances in a particular class.

$$Support = FN + TP \tag{4.4}$$

Accuracy: the rate of correctly identified members or instances from all members or instances of all classes.

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{4.5}$$

Confusion matrix : It is an n by n matrix where n is the number of class. It shows the breakdown of the predicted classes of all members of a specific class.

4.2.2 Naive Bayes

Naive Bayes is usually well suited to overcome overfitting problems on small dataset and still be robust. Here Naive Bayes has upwards of 80% accuracy on both dataset, with a slightly higher mean accuracy for Dataset A. Although this accuracy is slightly skewed due to more correct prediction of class 0 instances. For class 1 the accuracy was around 66%. Naive Bayes can still be used in the ensemble classifier, as we can overcome this loss of prediction accuracy.

Table 4.3 Results of 5-Fold Cross Validation on Dataset A for Naïve Bayes

Accuracy of classifier on each fold				
0.844	0.937	0.812	0.903	0.839
The mean accuracy	0.867			
Standard Deviation	0.0460			

Table 4.4 Detailed Classification Report of Naïve Bayes Classification using Dataset A

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.80	0.90	0.84	39
1	0.83	0.69	0.75	29
Average/Total	0.81	0.79	0.79	68
Accuracy	0.80882			

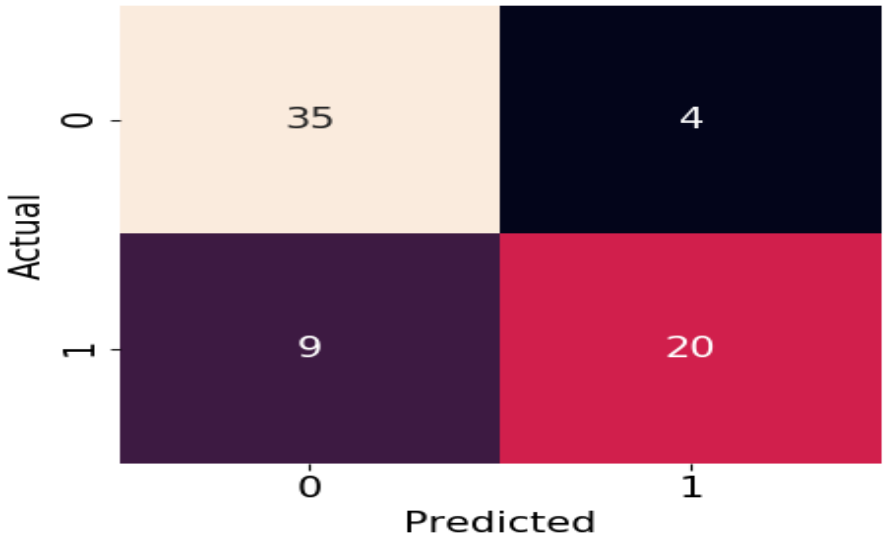

Fig. 4.1 Confusion Matrix of Naïve Bayes Classification on Dataset A

Table 4.5 Results of 5-Fold Cross Validation on Dataset B for Naïve Bayes

Accuracy of classifier on each fold				
0.848	0.906	0.750	0.839	0.871
The mean accuracy	0.843			
Standard Deviation	0.0519			

Table 4.6 Detailed Classification Report of Naïve Bayes Classification using Dataset B

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.79	0.93	0.85	40
1	0.86	0.66	0.75	29
Average/Total	0.82	0.81	0.81	69
Accuracy	0.81160			

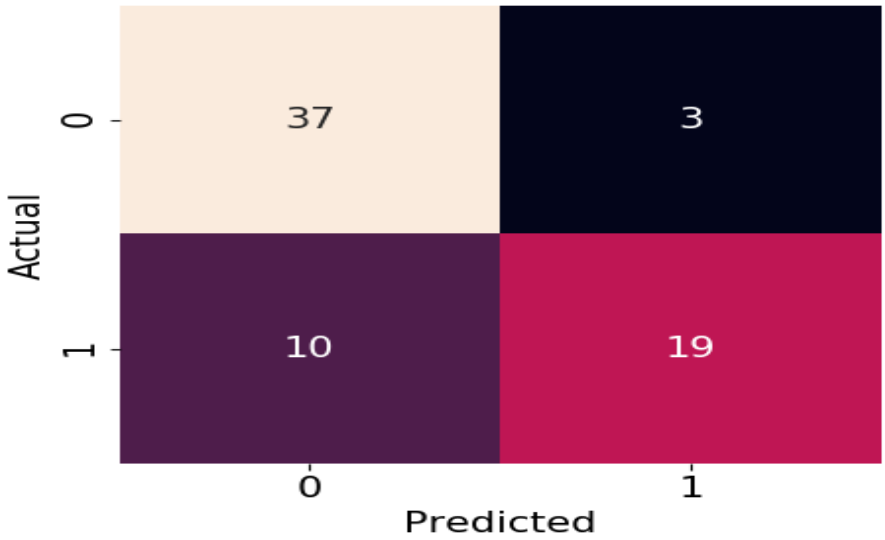

Fig. 4.2 Confusion Matrix of Naïve Bayes Classification on Dataset B

4.2.3 Random Forest Classifier

Random Forest show excellent accuracy in both datasets, with around 98% accuracy. The accuracy is consistent as standard deviation of accuracy was very low. Random Forest is a strong candidate for the ensemble classifier.

Table 4.7 Results of 5-Fold Cross Validation on Dataset A for Random Forest

Accuracy of classifier on each fold				
0.937	1.00	1.00	0.968	1.00
The mean accuracy	0.981			
Standard Deviation	0.0251			

Table 4.8 Detailed Classification Report of Random Forest Classification using Dataset A

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.95	1.00	0.97	39
1	1.00	0.93	0.96	29
Average/Total	0.98	0.97	0.97	68
Accuracy			0.97059	

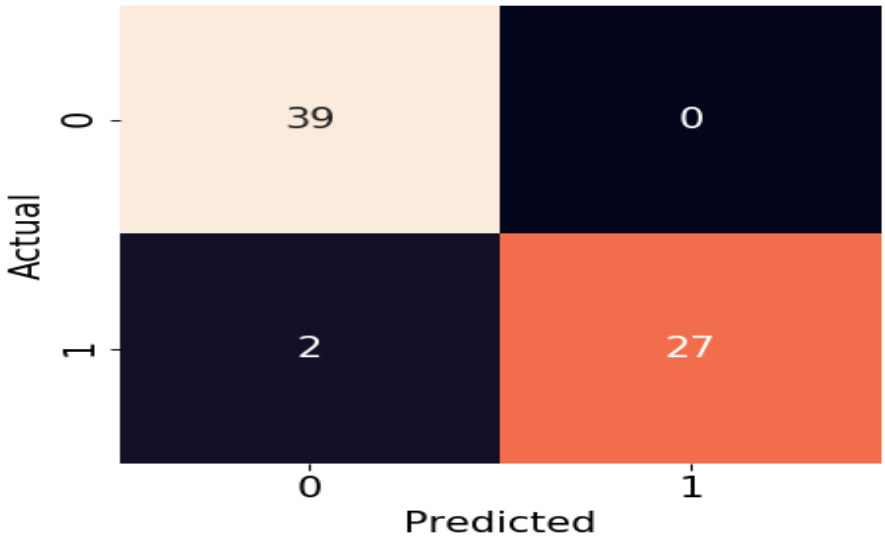

Fig. 4.3 Confusion Matrix of Random Forest Classification on Dataset A

Table 4.9 Results of 5-Fold Cross Validation on Dataset B for Random Forest

Accuracy of classifier on each fold				
0.970	1.00	1.00	1.00	1.00
The mean accuracy	0.994			
Standard Deviation	0.0121			

Table 4.10 Detailed Classification Report of Random Forest Classification using Dataset B

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.95	1.00	0.98	40
1	1.00	0.93	0.96	29
Average/Total	0.97	0.91	0.97	69
Accuracy	0.97101			

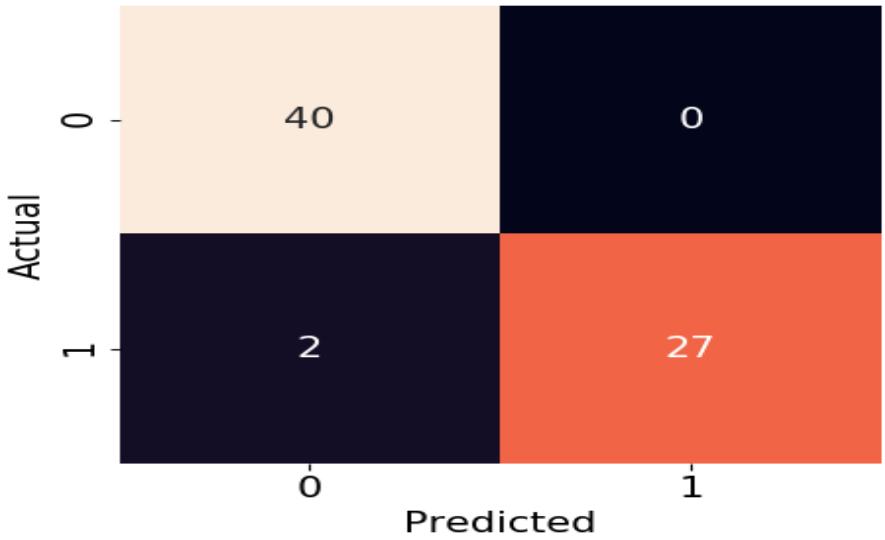

Fig. 4.4 Confusion Matrix of Random Forest Classification on Dataset B

4.2.4 SVM

Support Vector Performs poorly among the classifiers. It had an accuracy of 53% on Dataset A and an accuracy of 52% on Dataset B. In both cases it had extremely low Standard Deviation of around 0.007. This show that regardless of the orientation of the dataset, SVM has very prediction capability for our purposes. The accuracy is skewed as well, because Support Vector Machine predicted all test cases as NOTCKD, so the predictions for instances of that class might as well be lucky guesses.

Table 4.11 Results of 5-Fold Cross Validation on Dataset A for Support Vector Machine

Accuracy of classifier on each fold				
0.531	0.531	0.531	0.548	0.548
The mean accuracy	0.538			
Standard Deviation	0.00839			

Table 4.12 Detailed Classification Report of Support Vector Machine Classification using Dataset A

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.57	1.00	0.73	39
1	0.00	0.00	0.00	29
Average/Total	0.29	0.50	0.36	68
Accuracy	0.57353			

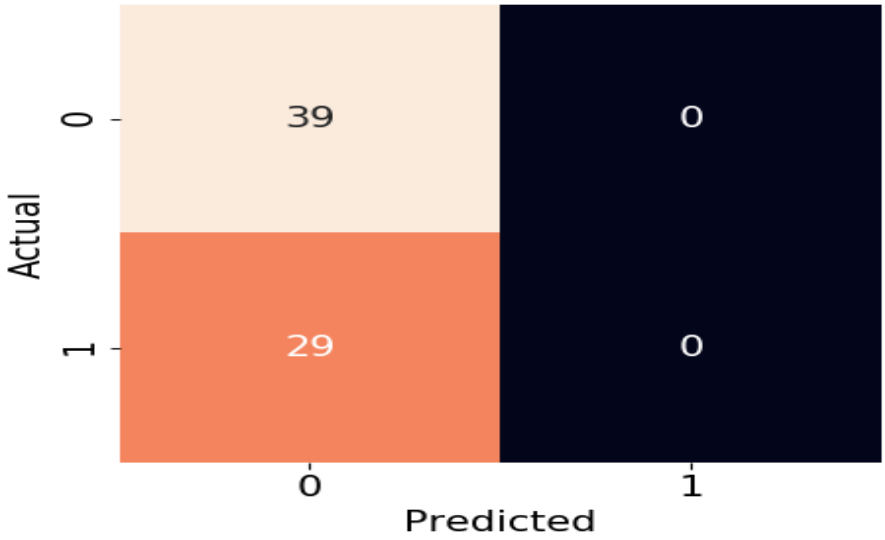

Fig. 4.5 Confusion Matrix of Support Vector Machine Classification on Dataset A

Table 4.13 Results of 5-Fold Cross Validation on Dataset B for Support Vector Machine

Accuracy of classifier on each fold				
0.515	0.531	0.531	0.516	0.516
The mean accuracy	0.522			
Standard Deviation	0.00758			

Table 4.14 Detailed Classification Report of Support Vector Machine Classification using Dataset B

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.58	1.00	0.73	40
1	0.00	0.00	0.00	29
Average/Total	0.34	0.58	0.43	69
Accuracy	0.57971			

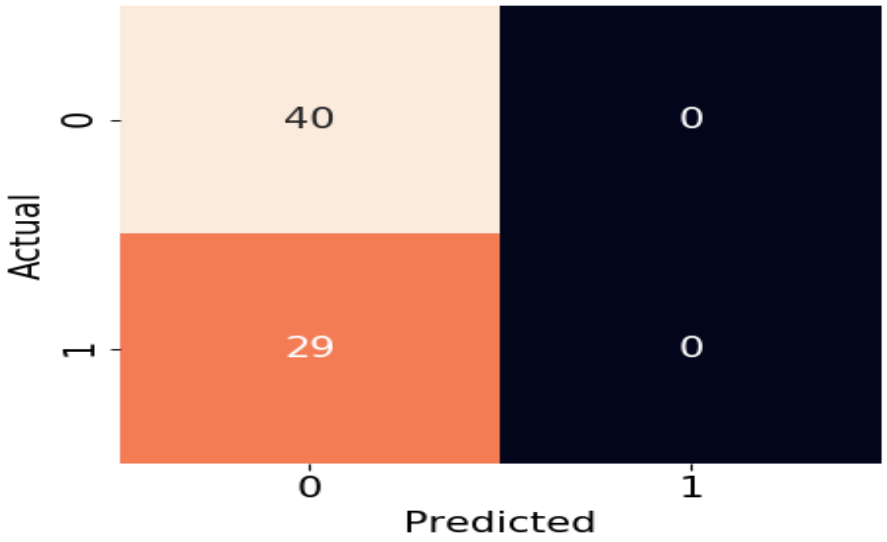

Fig. 4.6 Confusion Matrix of Support Vector Machine Classification on Dataset B

4.2.5 KNN

K Nearest Neighbor classifier had a relatively low accuracy, with 70% mean accuracy on Dataset A, and 61% accuracy on Dataset B. When predicting the CKD positive cases, accuracy was even lower, with 41% accuracy on Dataset A and 52% accuracy on Dataset B.

Table 4.15 Results of 5-Fold Cross Validation on Dataset A for K Nearest Neighbor

Accuracy of classifier on each fold				
0.719	0.688	0.625	0.839	0.645
The mean accuracy	0.703			
Standard Deviation	0.753			

Table 4.16 Detailed Classification Report of K Nearest Neighbor Classification using Dataset A

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.67	0.87	0.76	39
1	0.71	0.41	0.52	29
Average/Total	0.69	0.64	0.64	68
Accuracy	0.67647			

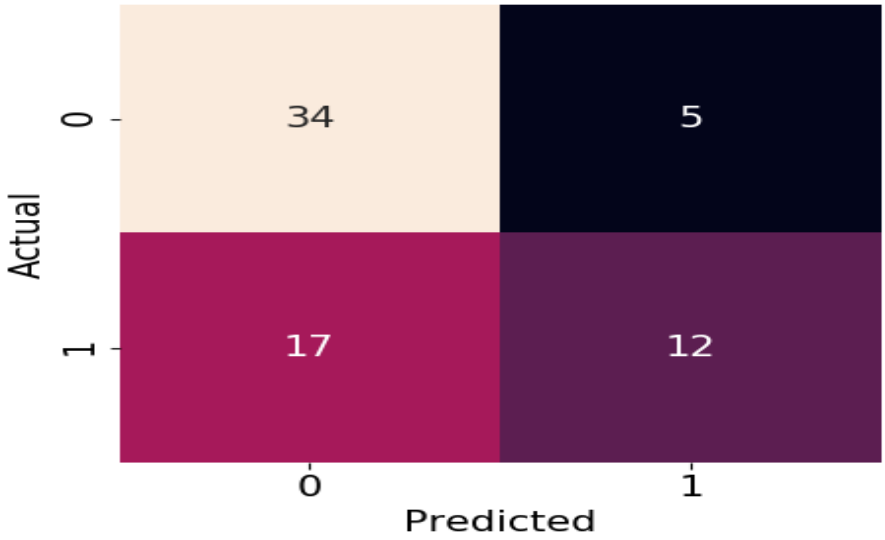

Fig. 4.7 Confusion Matrix of K Nearest Neighbor Classification on Dataset A

Table 4.17 Results of 5-Fold Cross Validation on Dataset B for K Nearest Neighbor

Accuracy of classifier on each fold				
0.515	0.594	0.688	0.774	0.516
The mean accuracy	0.617			
Standard Deviation	0.101			

Table 4.18 Detailed Classification Report of K Nearest Neighbor Classification using Dataset B

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.71	0.85	0.77	40
1	0.71	0.52	0.60	29
Average/Total	0.71	0.71	0.70	69
Accuracy	0.71014			

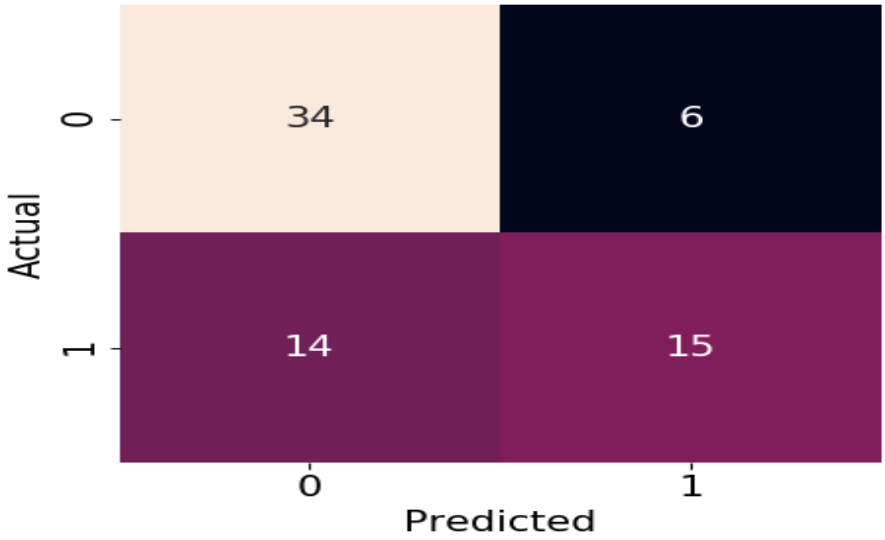

Fig. 4.8 Confusion Matrix of K Nearest Neighbor Classification on Dataset B

4.2.6 Logistic Regression

Logistic Regression Performs extremely well for both Dataset A and B. It had a mean accuracy of 98.1% with a standard deviation of 0.025 using 5 fold cross validation on Dataset A. For Dataset B the mean accuracy was 96.3% and standard deviation was 0.022. The small loss in prediction accuracy is coming from falsely predicted classes for positive CKD instances.

Table 4.19 Results of 5-Fold Cross Validation on Dataset A for Logistic Regression

Accuracy of classifier on each fold				
0.937	1.00	1.00	1.00	0.968
The mean accuracy	0.981			
Standard Deviation	0.0251			

Table 4.20 Detailed Classification Report of Logistic Regression Classification using Dataset A

Class	Precision	Recall	F1-Score	Support
0	0.95	1.00	0.97	39
1	1.00	0.93	0.96	29
Average/Total	0.98	0.97	0.97	68
Accuracy	0.97059			

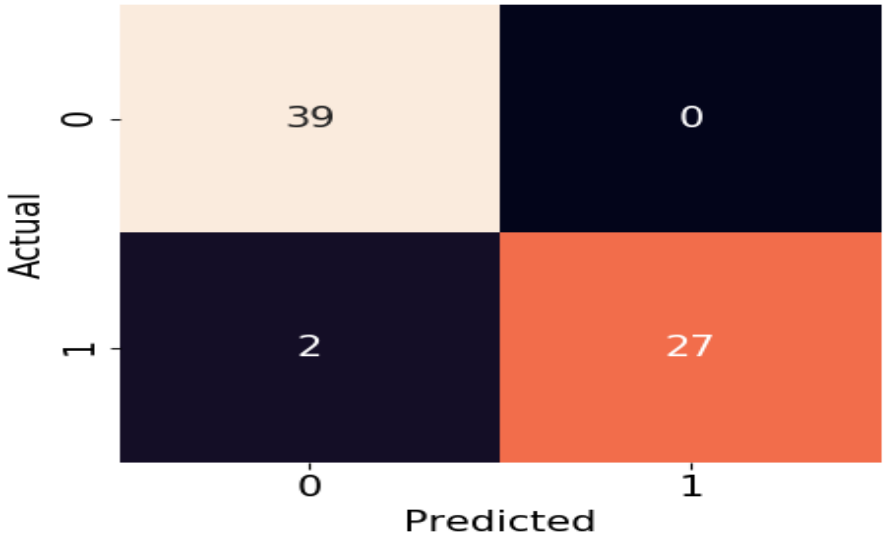

Fig. 4.9 Confusion Matrix of Logistic Regression Classification on Dataset A

Table 4.21 Results of 5-Fold Cross Validation on Dataset B for Logistic Regression

Accuracy of classifier on each fold				
0.939	0.969	0.938	1.00	0.968
The mean accuracy	0.963			
Standard Deviation	0.0229			

Table 4.22 Detailed Classification Report of Logistic Regression Classification using Dataset B

Class	Precision	Recall	F1-Score	Support
0	0.95	1.00	0.98	40
1	1.00	0.93	0.96	29
Average/Total	0.97	0.97	0.97	69
Accuracy	0.97101			

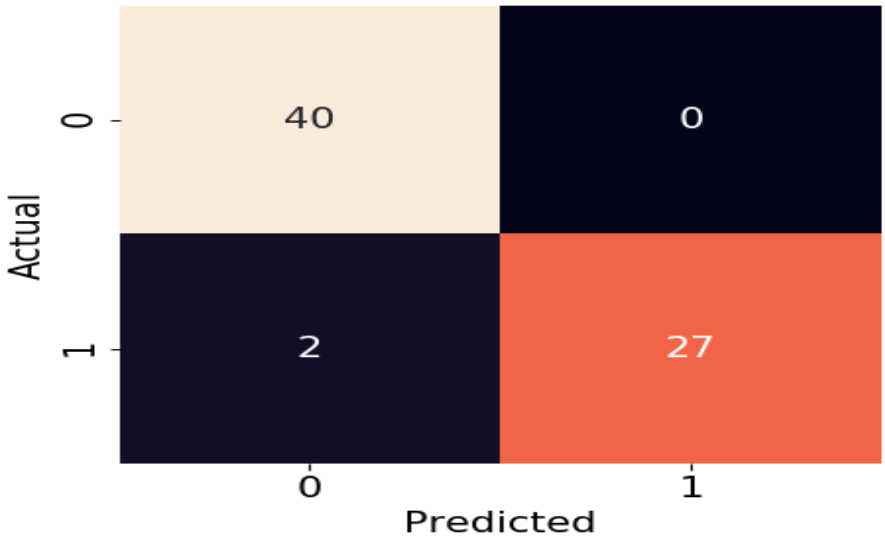

Fig. 4.10 Confusion Matrix of Logistic Regression Classification on Dataset B

4.3 Result Analysis

4.3.1 Ensemble Classifier

As determined from our analysis of the classifiers on the two datasets, 3 classifiers, namely Random Forest, Naive Bayes and Logistic Regression performed remarkably with particularly high degree of accuracy in predicting class 0 instances, i.e. when patients did not have CKD. K-Nearest Neighbor and Support Vector Machine, performed poorly. A possible cause might have been the poor distribution inherent in our dataset due to the mix of nominal and numeric features. So, Random Forest, Naive Bayes and Logistic Regression will be used to build our ensemble classifier. A voting model was used, so each vote or classification from each classifier is weighted equally and majority class is chosen as predicted class.

Table 4.23 Results of 5-Fold Cross Validation on Dataset A for Ensemble Classifier

Accuracy of classifier on each fold				
0.937	1.00	1.00	0.968	0.968
The mean accuracy	0.975			
Standard Deviation	0.0235			

Table 4.24 Detailed Classification Report of Ensemble Classifier Classification using Dataset A

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.93	1.00	0.96	39
1	1.00	0.90	0.95	29
Average/Total	0.96	0.96	0.96	68
Accuracy	0.95588			

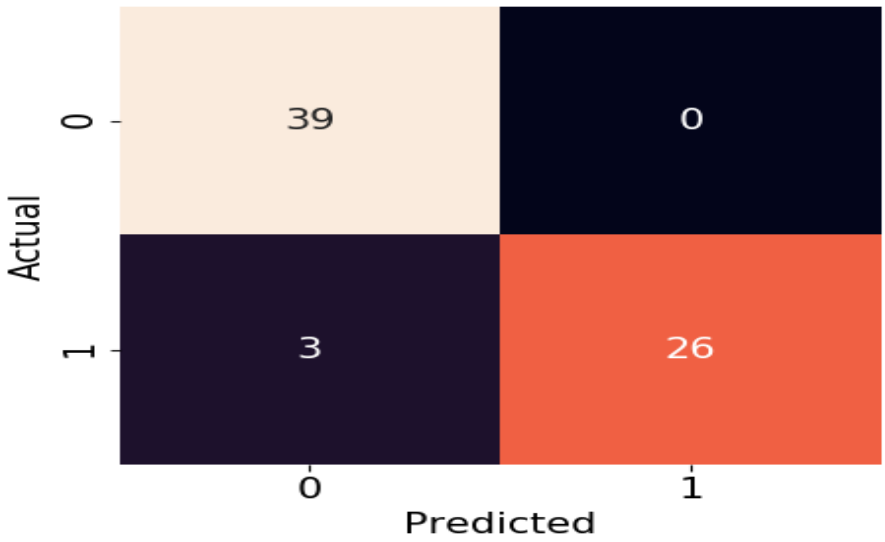

Fig. 4.11 Confusion Matrix of Ensemble Classifier Classification on Dataset A

The chosen classifiers performed equally on both Dataset A and B. But their performance was more consistent on Dataset A as the Standard Deviation for accuracy was lower. Therefore we chose to use Dataset A to train and test our Ensemble Classifier. The Dataset was split into Train and Test set in the ratio 70:30. The Train set was used to first 5-fold cross validate the Ensemble Classifier and then train the classifier. The Test set was later used to determine the prediction performance of the ensemble classifier. As can be seen from the detailed classification report, the classifier is very accurate with a few false negatives. This is something we would like to improve in the future.

Chapter 5

CONCLUSION AND FUTURE WORKS

5.1 Conclusion

The study was made to find the smallest group of attributes used to predict CKD. This is because CKD is such a disease that cannot be identified before last stages so by using simple attributes which are regularly tested by patients for different diseases. The use of ensemble classifier was to vote out the best classifier as medical datas are often imbalanced missing. We were able to find the best set of 10 attributes with highest ranking through Recursive feature elimination and univariate feature selection with random forest classifier. Ensemble classification was done using 5 algorithms out of which 3 gave 98% accuracy namely Logistic Regression, Naive Bayes and Random Forest. SVM and KNN didn't provide expected results probably due to small data set. The risk of overfitting in a small dataset is always there but the the 3 algorithm used to give 98% accuracy are well known for their ability to work with small data sets. Our main challenge was our small data set which we had to use due to lack of primary or secondary data. A bigger data set would have worked to make the study more robust.

5.2 Future Work

In future we want to collect medical datas around the country and compile them to make a big data set as it was our study's main challenge. We want to develop the paper for making publications and hope to make it into a journal. And then we want the study to help people so we want to use this to make a web application for the general public around our country who can easily check their possibilities of CKD using test results from other diseases. We

want also to develop the study to work into more complex data set and if possible make classifications for other kidney related diseases.

References

[1] Alasker, H., Alharkan, S., Alharkan, W., Zaki, A., and Riza, L. S. (2017). Detection of kidney disease using various intelligent classifiers. In *2017 3rd International Conference on Science in Information Technology (ICSITech)*, pages 681–684.

[2] Chiu, R. K., Chen, R. Y., and and (2012). Intelligent systems on the cloud for the early detection of chronic kidney disease. In *2012 International Conference on Machine Learning and Cybernetics*, volume 5, pages 1737–1742.

[3] Fricks, R. B., Bobbio, A., and Trivedi, K. S. (2016). Reliability models of chronic kidney disease. In *2016 Annual Reliability and Maintainability Symposium (RAMS)*, pages 1–6.

[4] Granitto, P. M., Furlanello, C., Biasioli, F., and Gasperi, F. (2006). Recursive feature elimination with random forest for ptr-ms analysis of agroindustrial products. *Chemometrics and Intelligent Laboratory Systems*, 83(2):83–90.

[5] Joachims, T. (1998). Making large-scale svm learning practical. Technical report, Technical report, SFB 475: Komplexitätsreduktion in Multivariaten ....

[6] Keller, J. M., Gray, M. R., and Givens, J. A. (1985). A fuzzy k-nearest neighbor algorithm. *IEEE transactions on systems, man, and cybernetics*, (4):580–585.

[7] Kotsiantis, S., Kanellopoulos, D., and Pintelas, P. (2006). Data preprocessing for supervised leaning. *International Journal of Computer Science*, 1(2):111–117.

[8] Levey, A. S., Eckardt, K.-U., Tsukamoto, Y., Levin, A., Coresh, J., Rossert, J., Zeeuw, D. D., Hostetter, T. H., Lameire, N., and Eknoyan, G. (2005). Definition and classification of chronic kidney disease: a position statement from kidney disease: Improving global outcomes (kdigo). *Kidney international*, 67(6):2089–2100.

[9] Menard, S. (2002). *Applied logistic regression analysis*, volume 106. Sage.

[10] Nishanth, A. and Thiruvaran, T. (2018). Identifying important attributes for early detection of chronic kidney disease. *IEEE Reviews in Biomedical Engineering*, 11:208–216.

[11] Padmanaban, K. A. and Parthiban, G. (2016). Applying machine learning techniques for predicting the risk of chronic kidney disease. *Indian Journal of Science and Technology*, 9(29):1–6.

[12] Polat, H., Danaei Mehr, H., and Cetin, A. (2017). Diagnosis of chronic kidney disease based on support vector machine by feature selection methods. *Journal of Medical Systems*, 41(4):55.

[13] Pryke, A., Mostaghim, S., and Nazemi, A. (2007). Heatmap visualization of population based multi objective algorithms. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 361–375. Springer.

[14] Rish, I. et al. (2001). An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46.

[15] Saeys, Y., Inza, I., and Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *bioinformatics*, 23(19):2507–2517.

[16] Salekin, A. and Stankovic, J. (2016). Detection of chronic kidney disease and selecting important predictive attributes. In *2016 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 262–270.

[17] Vijayarani, S. and Dhayanand, S. (2015). Data mining classification algorithms for kidney disease prediction. *International Journal on Cybernetics & Informatics (IJCI)*, 4(4):13–25.

[18] Yildirim, P. (2017). Chronic kidney disease prediction on imbalanced data by multilayer perceptron: Chronic kidney disease prediction. pages 193–198.

[19] Zhu, M., Xia, J., Jin, X., Yan, M., Cai, G., Yan, J., and Ning, G. (2018). Class weights random forest algorithm for processing class imbalanced medical data. *IEEE Access*, 6:4641–4652.

[20] Zięba, M. (2014). Service-oriented medical system for supporting decisions with missing and imbalanced data. *IEEE journal of biomedical and health informatics*, 18(5):1533–1540.