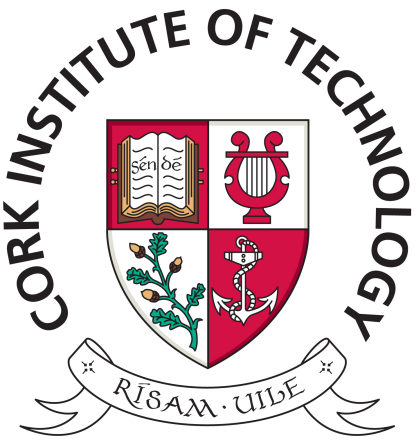

Comparative Analysis of various Machine Learning Algorithms for Predicting Chronic Kidney Disease

by
Sidharth Mehra

This proposal has been submitted in partial fulfillment for the module Research Practice and Ethics

in the
Faculty of Engineering and Science
Department of Computer Science

January 5, 2020

Declaration of Authorship

I, Sidharth Mehra, declare that this thesis titled, Comparative Analysis of various Machine Learning Algorithms for Predicting Chronic Kidney Disease and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for an masters degree at Cork Institute of Technology.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at Cork Institiute of Technology or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this project report is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.
- I understand that my project documentation may be stored in the library at CIT, and may be referenced by others in the future.

Signed: _____

Date: _____

Abstract

Chronic kidney disease has been a serious issue since long time in the medical science. The scope of our work lies in predicting the status of the person having chronic kidney disease or not; given the certain medical features of the person. We would be utilizing various machine learning algorithms like Support Vector Machines, K-nearest neighbour, Artificial Neural Networks and other ensemble techniques to develop our predictive machine learning model. The main motivation lies in the fact that our model will facilitate the timely diagnosis of the patients that are predicted with the chronic kidney disease and will in-turn cure the kidney failures preventing the situation from getting it worse. We would be leveraging the chronic kidney disease dataset from the UCI machine learning repository and performing the comparative analysis of various machine learning algorithms. Main focus is to handle the imbalance in our dataset which tends to weaken the model; through various sampling techniques. We would at the same time selecting the most appropriate features that contribute the most in predicting the target class through employing various feature selection mechanisms. Moreover the top performing machine learning models will undergo hyper-parameter optimization to further escalate the model detection performance. Finally a real-time system could be deployed in hospitals as a handy tool for the doctors for the prevention of the development of the chronic kidney disease and will also save the time and costs of the tests that are needed for the detection of this kidney disease.

Acknowledgements

Contents

Chapter 1

Introduction

This introduction chapter mainly focuses on three things. Firstly we will discuss about the **research question** along with its objectives. An in-depth detail on what we are going to do is clearly stated in this chapter. Secondly we will discuss the **context** of the research question, explaining about the **motivation** about the idea along with the identification of the important challenges. Third aspect of this chapter deals with an overview of the **basic methodology** on how we will proceed with this research project. We would also be discussing how the idea behind this project would be **validated**.

The kidney plays a very important role in the human body. The main role of the kidney is to filter out the waste products and toxins from the blood. It also maintains an electrolyte balance and controls the blood pressure. The kidneys are also responsible for the production of red blood cells and its normal functioning has the significant impact on the health of an individual. Chronic Kidney disease is the very serious concern in the field of medical science where the kidneys of a person don't function normally and eventually loses the filtration ability (less than 10% of the normal capacity) due to which there is the accumulation of the fluid and waste in the body. This condition is very common in the old age and is caused mainly due to the high blood pressure and diabetes. If this kind of situation persists in a person, then it may lead to the kidney failure and the person has to take the support of artificial purification of the waste products and other toxins from the blood through the dialysis or may need to take the support of kidney transplant. With the advent of AI and **Machine learning** along with its

increasing use-cases in the medical science, one can able to predict whether the person is going to have the chronic kidney disease or not.

The **research objective** of this work is to develop an accurate machine learning system that will predict the status whether an individual would be having chronic kidney disease or not. To carry out this research project we would be following the end to end machine learning pipeline. The scope of the work lies in comparing various machine learning algorithms for the prediction of the kidney disease. We also aim to tackle the problem of **imbalance** in the dataset and selecting the appropriate set of features **(feature selection)** followed by **hyper-parameter optimization** to obtain the best performing machine learning model.

Imbalance in the medical dataset is the main challenge that tends to reduce the performance of the machine learning model. We will handle this issue with the help of various under-sampling and over-sampling techniques such as **Tomek Links** and **SMOTE** (Synthetic Minority Oversampling Technique). Next challenge associated with this work is the about selecting those features which are most relevant to the target feature which we will handle using various **Univariate, Tree-based and Greedy-based** feature selection mechanisms. For the hyper-parameter optimization, we would be exploring the techniques such as **Grid-Search CV and Random-Search CV** to improve the model performance. This project aims to tackle with all these challenges in order to come up with an accurate system for predicting the chronic disease. An accurate machine learning system will enable the timely diagnosis of the patients that are predicted with the higher chances of getting into this severe situation. The early detection of the disease with machine learning will prevent the further damage (kidney failure) through the means of proper medication and lifestyle changes. It will also reduce the chances of the heart stroke among the patients.

The dataset we would be utilizing is the **chronic kidney disease dataset** obtained from the UCI machine learning repository. This is a **supervised** machine learning problem as the dataset is labelled which means for every patient record we have the associated class which is a person either have CKD or not having CKD. The **input** to the system is all the medical feature records of the persons and the **output** of the system is the predicted status of the person having kidney

disease or not which makes our problem a **binary classification** problem. For our prediction task we would be employing various machine learning algorithms like Support Vector Machines, Naïve Bayes, K-nearest neighbour etc. We would also be utilizing the **ensemble methods** such as random forest or gradient boosting which are considered to be the powerful algorithms.

Whole of our work implementation could be broken down into **4 individual steps** that would be the main **process flow** for the prediction of the kidney disease using machine learning. First step of our pipeline is the **Exploratory Data Analysis** which will help us summarize the main characteristics of our kidney disease dataset with the help of visual methods. Next step is the **Data Pre-processing** phase where we will process our kidney disease dataset so that it can be a valid input for most of the machine learning algorithms. Here we will divide the whole dataset into train and test split. The next step of the lifecycle is the **Model Building and Evaluation**. Here we would be training different machine learning models and chose the top performing models based on the various evaluation metrics. Next step is related with the **Fine Tuning and Hyper-parameter Optimization** where the best models will undergo the parameter tuning. Next phase is the **Model Validation** where we would be testing the generalization capability of the model on the unseen test data. All these phases would be covered in great details in the Chapter 3 of this research proposal.

This document is **structured** as follows: **Chapter 2** presents the existing solutions as well as the background information on kidney disease prediction. **Chapter 3** presents the proposed work towards predicting chronic kidney disease based on certain medical features of the person; the algorithms to be employed and the process used for training and tuning the best machine learning models. **Chapter 4** details the process taken, the tools used, with the special emphasis on evaluation. **Chapter 5** breaks down the work into work packages and develops a project plan and Gantt chart; the work plan will include the expected challenges and ways to deal with them (mitigation). **Chapter 6** closes the document with a summary.

Chapter 2

Background

In this section we would be critically analysing the 10 academic papers that revolve around the chronic kidney disease prediction using machine learning. We would be extracting the summaries of these papers along with their main characteristics which are defined with the proper structure described in the later part of this chapter. We will also determine the common and different points between these papers which will help us to extract the structure and the context of the problem solution. After analysing these 10 papers that are formulated according to the below structure, we will develop a similar structure to illustrate the context of our problem solution. We will also compare the existing methods and try to identify that is there any clear best solution or are there any trade-offs. The main characteristics of the ten papers are structured as the individual sections of this chapter following a chronological order.

2.1 Existing Methods

The work [1] published in 2013 was the first one that initiated the research topic of the predicting the various stages of the disease progression ranging from stage 1 to 5; if a person is detected with the Chronic Renal Failure based on certain medical features; so that the appropriate medication could be started in accordance with the predicted stage. It utilized various machine learning algorithms like Artificial Neural Network, Naïve Bayes and Decision Tree. This work took the support of WEKA; an open-source data mining tool for implementing various machine learning algorithms. The authors in [5] utilized radial basis function network, multilayer perceptron and logistic regression for the prediction of the

kidney disease with the help of the WEKA tool. They followed all the phases of the data mining from knowledge gathering to the evaluation. The work presented in [10] and [9] are by the same authors for the prediction of the kidney disease. In [10] they compared the two algorithms SVM and ANN while in [9] there is the comparison between Naïve Bayes and SVM. They used the MATLAB software to train and evaluate the model performance in both the works. Authors in [8] made use of Rapid-miner tool to implement algorithms like Naïve Bayes and ANN. Authors compared and implemented the algorithms- KNN, SVM, LR and DT using WEKA and MATLAB softwares using 5-fold and 10-fold cross validation in [2] and [7] respectively. In [4], the authors implemented Ensemble techniques such as bagging, boosting, voting and stacking while in [3] the Random forest was the best classifier with the 10-fold cross validation. In [6], the authors compared the performance of KNN, Random Forest and Neural Networks using 10-fold cross validation.

2.2 Challenges and their mitigation

The main challenge associated in [1] is the knowledge acquisition and analysis. Authors took the help of doctors to prepare the dataset. The extracted knowledge is then fed to build a predictive system after all the pre-processing steps. Since the work [5], [10], [9] were the starting point for the kidney disease prediction on the different datasets; the main challenge was only the prediction and the evaluation which then set the ground base for the identification of the other challenges in the future to further improve the performance. In order to reduce the training time and improve the accuracy, the authors in [2] tried to reduce the feature set to the subset by employing the BestFirst search greedy hill-climbing technique while the authors in [7] employed the ranking algorithm in their work. Authors in [4] used the SMOTE technique to handle the imbalance in the UCI dataset. The challenge of feature selection in [6] is handled by the two methods which are LASSO regularization and the wrapper method of feature importance.

2.3 Datasets involved

The dataset involved in [1] only had 102 instances of the clinical records of the persons along with the categorization of each individual with and without the

chronic renal failure. The authors in [5] were the creators and donors of the most widely used open source dataset on the UCI repository for the kidney disease prediction and utilized the same in their work. This dataset has 25 features and 400 instances of clinical tests of the individuals. The target class is the output label (CKD or NOT-CKD) making it a binary classification problem. This dataset is imbalanced with the class distribution of the 63% (CKD) and 37% (NOT-CKD). The dataset that was employed in [10] and [9] contained 584 instances of the medical records and 6 attributes/features. The target class contained the 5 classes depicting the various stages of the kidney disease. Work [8] to [6] utilized the UCI kidney disease dataset.

2.4 Performance Evaluation Metrics

In [1], [2] the performance of the model is measured using the accuracy, specificity and sensitivity. It was then evaluated by the physicians. Work [5] utilized the performance measures like F-score, type-1 and type-2 error rates and most importantly the kappa value which represents the measure of the agreement between the prediction made by the experts and the data mining classifiers. Performance in [10] is evaluated in terms of accuracy and the execution time. The main evaluation metrics chosen in [9], [4] were the accuracy, precision, recall and F-measure. Accuracy, Confusion matrix and Kappa score were used as the evaluation criteria in [8]. The work [7] utilizes the metrics such as Kappa Statistics, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Receiver Operating Characteristics (ROC) Area and Normal Accuracy to evaluate the model performance. The work in [3] used the ROC Area, Accuracy and Matthew's Correlation Coefficient (MCC) to evaluate the model performance on the test set. Paper [6] considers the F1-score, RMSE and Accuracy for the evaluation.

2.5 Results

The work in [1] achieved the 82% accuracy by the decision tree classifier. In [5] the authors were able to get the best accuracy of the 85% with the Multilayer Perceptron Neural Network. In [10], ANN achieved the higher classification accuracy (84.70%) than SVM (72.96%) but at the same time SVM got the less execution time (3.22 sec) than ANN (7.26 sec). In [9], SVM performed better

(76%) than Naïve Bayes (70%) in terms of classification accuracy but took more time than the other one. Work [8] showed that the Naïve Bayes (82%) classifier outperformed the ANN model (71%) in terms of classification accuracy. In [2], SVM proved to be the best model in terms of accuracy (83%) and sensitivity than any other model. In [7] the best (84%) algorithm proved to be the decision tree classifier and the ranking algorithm increased the performance when the number of attributes were 15. The results from [4] depicted that OneR algorithm performance is increased after ensembling with Jrip and RidoR in case of imbalanced and balanced data. Random Forest algorithm with the reduced set of attributes achieved the high accuracy of the 86% using the f1 measure and the 0.107 root mean square error in the work [6].

2.6 Conclusion Analysis of the 10 papers

The important findings from the analysis of the above papers are as follows:

- The most common dataset that was employed for the kidney disease prediction was the chronic kidney disease dataset from the UCI machine learning repository
- Most of the papers have utilized the basic machine learning algorithms like SVM, ANN, KNN, Naïve Bayes, Decision Tree but only few have used the Ensemble techniques like Random Forest, Ada Boost or Gradient Boosting.
- There were very less papers that did the exhaustive exploration of the techniques in the data pre-processing phase and evaluated the impact of each of them on the performance. Steps like Handling the Outliers and Missing values, Feature Encoding of the categorical variables and Feature Scaling should be taken into consideration to obtain the model with the good accuracy.
- Few papers explored either the techniques to handle the imbalance or feature selection mechanisms but there should be consideration of tackling both the issues using various techniques in this research project. This can significantly impact the performance of the model.
- Hyper-parameter optimization is the important part in machine learning that was not dealt in any paper so this part would be explored in detail in

this project.

- The best performance on the UCI dataset was achieved by the random forest in [6] with the Lasso Regularization mechanism of the feature selection.

Chapter 3

Proposal

The **conclusion analysis** from the background chapter has formed a basis about the existing work and gives us the insight about the best performing methods. This section represents the direction that we will undertake for this research project. We have already defined the research objectives in the previous chapters; here we will discuss the **techniques** to achieve these objectives. Therefore we would be presenting our detailed **contribution** in terms of design decisions, algorithms and methods, towards the accurate prediction of the chronic kidney disease. This chapter describes the **dataset** that we will use along with the features it contain. It will also describe the exhaustive **machine learning lifecycle** that we would be following explaining the each step of the lifecycle along with the **methods** we would be employing for carrying out each of these steps. This chapter will also list the **challenges** we will encounter in our project along with a plan and the appropriate techniques to mitigate them.

3.1 Dataset Description

We would be utilizing the open-source chronic kidney disease dataset from the UCI machine learning repository. The dataset was prepared under the supervision of Dr.P.Soundarapandian who is the Senior Consultant Nephrologist at the Apollo Hospitals, Tamil Nadu, India. The dataset was created by L.Jerlin Rubini who is the Research Scholar at Alagappa University and guided by the Dr.P.Eswaran who is the Assistant Professor in the same university. The dataset contains the **25 medical features** including the target feature out of which 11 are numeric and 14 are categorical. Each feature represents the type of clinical test where each row

Attribute	Data-type	Attribute Unit
Age	numerical	age in years
Blood Pressure	numerical	bp in mm/Hg
Specific Gravity	nominal	sg - (1.005,1.010,1.015,1.020,1.025)
Albumin	nominal	al - (0,1,2,3,4,5)
Sugar	nominal	su - (0,1,2,3,4,5)
Red Blood Cells	nominal	rbc - (normal, abnormal)
Pus Cell	nominal	pc - (normal, abnormal)
Pus Cell clumps	nominal	pcc - (present, notpresent)
Bacteria	nominal	ba - (present, notpresent)
Blood Glucose	Random numerical	bgr in mgs/dl
Blood Urea	numerical	bu in mgs/dl
Serum Creatinine	numerical	sc in mgs/dl
Sodium	numerical	sod in mEq/L
Potassium	numerical	pot in mEq/L
Haemoglobin	numerical	hemo in gms
Packed Cell Volume	numerical	-
White Blood Cell Count	numerical	wc in cells/cumm
Red Blood Cell Count	numerical	rc in millions/cmm
Hypertension	nominal	htn - (yes, no)
Diabetes Mellitus	nominal	dm - (yes, no)
Coronary Artery Disease	nominal	cad - (yes, no)
Appetite	nominal	appet - (good, poor)
Pedal Edema	nominal	pe - (yes, no)
Anaemia	nominal	ane - (yes, no)
Class	nominal	class - (ckd, notckd)

Table 3.1: Feature Information of the Chronic Kidney Disease Dataset

or instance is the representation of all the clinical tests that the individual has undergone in order to check whether he/she is having the chronic kidney disease or not. So there are a total of 24 feature attributes and the last column (25th) is the target attribute which have 2 classes – having the chronic kidney disease **(CKD)** and not having the chronic kidney disease **(Not-CKD).**

3.2 Kidney Disease Prediction Lifecycle

We would be following an exhaustive machine learning lifecycle to carry out the kidney disease prediction. This entire machine learning pipeline would be broken down into the individual steps. We would be looking at each of these steps in much more detail in this section. Below is the schematic representation of the

process flow which we will follow in this research project.

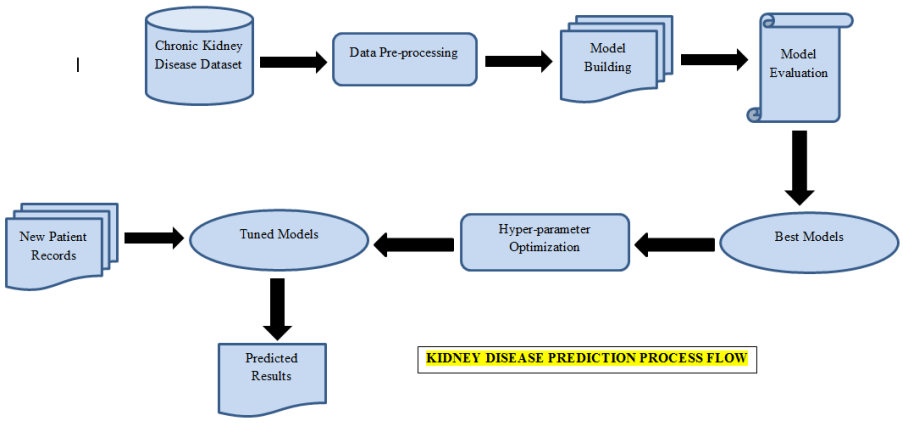

Figure 3.1: Proposed Process Flow

Steps followed in the process flow are as below:

1) **Chronic Kidney Disease Dataset:** We obtained the raw dataset from the UCI machine learning repository which needs to be prepared.

2) **Exploratory Data Analysis:** This step will help us summarize the main characteristics of our kidney disease dataset with the help of visual methods. EDA is for seeing what data can tell us beyond formal modelling or hypothesis testing task. We will understand the feature data in terms of missing values, outliers and the data-types. We will look into some visual plots like box-plots, scatter-plots and correlation matrix that will help us to make more sense out of the data.

3) **Data Pre-processing:** This is the most essential step in machine learning which prepares our data so that it can be a valid input for most of the machine learning and optimization algorithms. Dealing with missing values and the outliers along with feature encoding are the main sub-phases that would be undertaken in this step. We would also be doing feature scaling so that the range of all the features in the dataset is same. Here we will divide the whole dataset into train and test split. The most important steps in this pre-processing phase is to handle the imbalance (only on the training set)

and to select the most appropriate features through the feature selection mechanisms that would be discussed in detail in later chapter.

4) Model Building: Here we would be employing the best performing machine learning algorithms obtained from the background research such as ANN, SVM and KNN. Also we would be applying the unexplored ensemble techniques on the kidney disease dataset like Ada-boost, Random Forest and Gradient Boosting. These techniques combine the results from the number of learning algorithms to obtain a better predictive performance than any single machine learning algorithm. We would be training these models on the standard parameters on the training set using the 10-fold cross validation.

5) Model Evaluation: Most widely used evaluation metrics for the binary classification problem like ours are confusion metrics, f1-score, precision, recall and ROC area. We would be then choosing the best performing models based on these metrics.

6) Hyper-parameter Optimization: We will then tune the parameters of the best models with the help of the techniques like Grid Search CV and Randomized Search CV. We will compare each of these techniques based on the performance.

7) Model Validation: We will then check the generalization capability of the best tuned models on the unseen patient data and measure the performance based on the above mentioned evaluation metrics. And then come up with the most accurate machine learning system to predict whether the person is having the kidney disease or not based on certain clinical tests.

3.3 Challenges Identified and their mitigation

Based on the background research in the previous chapter we identified the below important challenges in the prediction of kidney disease whose mitigation would certainly improve the model performance. We would be seeing various techniques to mitigate each of these challenges and finally would be opting the one with the better suitability and the performance.

1) Handling the outliers: Outliers in the dataset tend to skew the distribution and could be responsible for the under-performance of the certain algorithms. We would be looking at two ways to handle these outliers. First we would be looking at the box-plots and delete the outliers. Secondly we would be applying the multivariate outlier detection technique – DBScan (clustering based technique) to deal with outliers.

2) Handling the missing values: There are missing values in our dataset which cannot be fed into the training algorithms like this. Therefore we would be looking at various imputation techniques like replacing the missing values with mean, mode or median of that feature column.

3) Feature Encoding: There are 14 categorical features in our dataset that needs to be represented into the numeric format to be as the valid input for the machine learning framework. All these 14 features are nominal meaning their values don't have any inherent ordering present in them. We would be employing both the techniques one-hot encoding and label encoding to encode these categorical features.

4) Feature Scaling: It is process of transforming the features in the dataset such that all the features are on the same scale. It is a good practice to either normalize (range of values is between the 0 and 1) or standardize (values are transformed such that the mean=0 and standard deviation=1) the feature columns so that the machine learning algorithms like KNN, SVM and Neural Networks could perform well.

5) Handling the Imbalance: Imbalance is the significant difference in the proportion of the classes present in the target feature. This dataset is imbalanced with the class distribution of the 63% (CKD) and 37% (NOT-CKD) which tends to reduce the performance of the ML models. We will handle this issue with the help of various under-sampling and over-sampling techniques such as Tomek Links and SMOTE (Synthetic Minority Oversampling Technique).

6) Feature Selection: Feature Selection is the process of ranking or quantifying the contribution of each feature in a dataset on the basis of the relationship of that feature column with the target feature column. To extract the most

important features (that contribute the most in predicting the target class) from the kidney dataset, we would be employing the Univariate feature selection, Tree based feature selection, Correlation heat-map and Greedy based feature selection.

7) Hyper-parameter Optimization: The main focus is to tune the parameters of the ensemble models like gradient boosting or random forest as the number of parameters of these models are quite large and it takes much time to find the best set of the parameters. Therefore it would be requiring the multiple runs to try out the different combinations. We would be exploring the optimization techniques such as Grid Search CV and Randomized Search CV for finding the optimal parameters.

Chapter 4

Methodology

We have already seen in detail in the previous chapter, the steps undertaken in our approach. Here in this chapter we will see how these steps would be implemented. We will discuss the tools, frameworks and programming language that we plan to consider while executing each step of our kidney disease prediction lifecycle. We would also be discussing the evaluation step of our machine learning lifecycle in great detail.

4.1 Tools Used

Programming Language that we would be using

To carry out each of the step in our machine learning pipeline we would be using the **Python 3** as the choice of our programming language.

Integrated Development Environment

We would be using the Jupyter Notebook and Spyder IDE of the Anaconda framework to execute our python code. For the debugging purposes in case of handling the errors we would be using the PyCharm IDE.

4.2 Implementation Details

Implementation of the Exploratory Data Analysis

Here we would be analysing our kidney disease data with the help of visual methods. We would be plotting the Violin plots, Box plots, Scatter plots, Histograms, Probability Density Functions to make the more sense out of the data. We would

be utilizing the functions of **Matplotlib and Seaborn** visualization libraries of python to plot these visual graphs.

Implementation of the Data pre-processing steps

We would be exhaustively employing the classes and the functions of **Pandas and Scikit Learn** libraries of python to prepare our kidney disease dataset so that it can be a valid input for the machine learning and the optimization algorithms.

- **Outlier Detection:** A simple box-plot method of **Univariate outlier detection** either through the Pandas library or Seaborn would help us identify the outliers in our dataset so that we can get rid of them. Another method is the **Multivariate outlier detection** which can be implemented through the DBScan method based on clustering. We can implement this technique by importing the **DBSCAN class** from the sklearn.cluster python module.

- **Handing Missing Values:** Missing values can be substituted by the imputation techniques. **SimpleImputer class** of sklearn.impute python module gives us the option to replace the missing values with mean, mode and median as the strategy argument of the class.

- **Feature Encoding:** We would be employing the two techniques for the encoding of the categorical variables. First one is label encoding where each value within a categorical variable is assigned an integer representation. We can achieve this by importing the **OrdinalEncoder class** from the sklearn.preprocessing python module. Another method is One Hot Encoding which can be achieved by importing the **OneHotEncoder class** from the same python module.

- **Feature Scaling: MinMaxScaler** function of the sklearn.preprocessing module normalizes each feature where as the **StandardScaler** function of the same module could standardize the feature of the dataset.

- **Feature Selection:** We would be exploring the 3 techniques to select the most appropriate features from our dataset. The first one is **Univariate** feature selection which examine the relationship between each feature and the target class. This can be implemented by importing the **SelectKBest class** of the sklearn.feature-selection python module. Next method is the

Tree based feature selection which builds many trees and calculates the average reduction in uncertainty achieved by each feature across all the trees and uses this as the means of feature ranking. The features with the largest reduction have the highest impact (the most important features). We can use the **feature_importance** attribute of the tree based classifier to get the respective feature importance of each of the features in the dataset. The third approach is the **greedy based** feature selection which is **recursive feature elimination with cross validation**. This approach of feature selection using the recursive feature elimination mechanism automatically reduces the number of features involved in learning model based on their effective contribution to the overall accuracy performance of the algorithm. In this technique the external estimator assigns the weight to each of the feature by building the feature importance array and recursively consider the smaller and smaller set of features to select the most important features. This can be implemented using the **RFECV class** of the sklearn.feature_selection python module.

- **Handling Imbalance:** For dealing with imbalance in our dataset we would be employing the techniques such as **SMOTE**; an over-sampling technique and the **Tomek links**; an under-sampling technique. We would be using the classes of **Imbalanced learn** contribution package of the Scikit learn to balance the dataset through the sampling.

Sampling of the Kidney Dataset into train test split

We would be dividing our kidney disease dataset into the training set and the test set. For this we would be utilizing the **train_test_split function** of the sklearn.model_selection module of python. We would be dividing the whole dataset such that the **70%** of the instances goes to the training set and the remaining **30%** goes to the test set by passing the appropriate arguments into the function.

Implementation of the Model Building

We would be building the K-nearest neighbour, Support Vector Machine and Artificial Neural Networks machine learning models on the training set. For building the KNN model we would be importing the **KNeighborsClassifier class** of the sklearn.neighbors module of python and for the SVM the **SVC class** of the sklearn.svm module of python. For the ANN we would be importing the **MLP-**

Classifier class of sklearn.neural_network module of python. We would also be training the 3 ensemble techniques – **Random Forest, Gradient Boosting and Ada Boost** by importing the classes present in the **Scikit learn** library.

Implementation of Different Hyper-parameter Optimization Techniques

The top 2 models obtained will undergo the parameter tuning. There is a defined **search process** for finding out the optimal values of these hyper-parameters which is as follows:-

- A searching algorithm
- A parameter space (range of values for the parameters)
- A method for searching the values.
- Cross validation for finding the best hyper-parameters.

There are mainly 2 techniques for hyper-parameter tuning- Grid Search CV and Randomized Search CV. We would be importing the GridSearchCV and RandomizedSearchCV classes from **sklearn.model_selection module** of python.

1. **Grid Search CV** – In this approach, we try every combination of preset values given in the form of parameter grid which is a list of dictionaries. When the hyper-parameters are more, then the number of evaluation increases exponentially with each additional parameter.
2. **Randomized Search CV** – Random combinations of hyper-parameters are used to find the optimal parameters. It tries a random range of values given in the form of parameter distribution.

4.3 Evaluation Methods

Implementation of Different Evaluation Metrics

Based on the suitability, background and the challenges in our research problem, we would be employing the below evaluation metrics to compare the above 6 machine learning models. We would be implementing these metrics by importing the corresponding functions from the **sklearn.metrics** module of python. We would then be choosing the top 2 models based on these metrics for the tuning of their parameters

- **Accuracy:** It is the measure of the total number of test instances that are predicted correctly by our model. It does not give us insight about the accuracy for the individual classes present in the target feature. When dealing with the imbalance dataset, the model could overall give a high accuracy but the accuracy of the minority class could be very low as compared with the accuracy of the majority class. In our case the test set is imbalanced so we can't rely on the accuracy and need to look at the other metrics for evaluation.

- **Confusion Matrix:** It is the representation of the information about the actual and predicted classifications by the classification algorithm. It gives the insight into the accuracy of each of the classes that are present in the target feature meaning that we get the number of test instances for each class that are correctly as well incorrectly predicted by our model. This makes it a good metrics for the evaluation.

- **Recall:** It is the measure of sensitivity or the true positive rate. It tells how confident we can be that all instances belonging to a specific class have been correctly classified by the model. The formula for the recall is as follows

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{4.1}$$

- **Precision:** It is the measure of the positive predictive value. It tells how confident we can be that any instance predicted as belonging to a certain class actually belongs to that class. The formula for the precision is as follows

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{4.2}$$

- **F1 Score:** It is interpreted as the weighted average of precision and recall for a particular class. For F1 score to be high of any class, both the precision and recall values should be high as well. The best value for f1 score is 1 and worst value is 0. The formula for F1 score is

$$F1\ Square = 2\frac{Recall * Precision}{Recall + Precision} \tag{4.3}$$

- **Classification Report:** This report provides an insight into the precision, recall and f1 score broken down by the class. For our problem as the data

is imbalanced, so we will not see the normal accuracy, rather our evaluation for this biased classification would be on the basis of confusion matrix (indication of the accuracy of each of the class). Also we would be considering the f1 score across each of the 2 classes which tell about how good the balance is between the precision and recall values. Moreover this classification report also generates the mean and weighted average of precision, recall and f1 score for each of the individual classes in our target feature. So we would be considering the macro average and weighted average of f1 scores for both the classes as the one of the metric for evaluation.

Chapter 5

Work plan

From the Chapter 3 and Chapter 4 of Proposal and Methodology, we have identified the structure of our solution in terms of components. We have the breakdown of our approach towards the development of an accurate kidney disease prediction system at our disposal. We would be fitting the **individual components** of our work in the **work-packages** which are further split into the **tasks**. We will then allocate some amount of time (days or weeks) that each of the individual tasks in our work packages might take according to its rigor. With all our work broken down into tasks, we will determine the relations between the tasks i.e. dependencies. We would be representing our work-plan as the **Gantt Chart** depicting how much time each task would take to execute. Finally, we will then fit this work-plan into the allocated time frame to carry out the implementation and the thesis writing of this research project in the next semester.

5.1 Work packages and its Tasks

The components of our solution are represented as the work-packages which are further broken down into the individual tasks. Below is the concise representation of these work-packages and its tasks following a chronological order

5.2 Gantt Chart of the Work Plan

Now we would be representing our work plan as the Gantt chart. We need to assign a time frame to each of the work package. We will also provide reasoning for allocating a particular time to the corresponding component of our research

project. The **total duration** of this project is estimated to be of **4 months** (Feb 2020 to May 2020) totalling up to **16 weeks**. Below is the description of the estimated time frame that we would be giving to each of the work package.

1. **Exploratory Data Analysis:** This phase will take a maximum of **1 week** for plotting the graphs for the Univariate and Bivariate analysis of the respective features.

2. **Data Pre-processing:** This phase will take a maximum of **2 weeks**. Initially we will inculcate any widely used technique for the each step of the pre-processing so as to make a valid input for the machine learning and optimization algorithms.

3. **Model Training:** This phase will take a maximum of **3 weeks** for training the 6 different machine learning models on the standard parameters. Also we would be exploring the different combinations of the pre-processing steps and opting the best combination based on the best training performance.

4. **Model Evaluation:** This phase will take a maximum of **2 weeks** for evaluating and analysing the different models based on 10-fold cross validation score. Then we would select the best models.

5. **Hyper-parameter Optimization:** This phase will take a maximum of **2 weeks.** Here the best models will undergo the analysis of the two techniques for the parameter tuning in terms of execution time and different parameter combinations.

6. **Model Validation:** This phase will take a maximum of **2 weeks** for doing the comparative analysis of the tuned models on the unseen data. We will deep dive into the various metrics for the classification problems.

7. **Thesis Writing:** This phase will take a maximum of **4 weeks**. We will explain the each and every technique employed in our project in great detail with the strong basis formed from the literature review and the associated challenges. And in the end would be analysing the results of the proposed method for the detection of kidney disease along with the conclusion and the scope of future work.

WORK PACKAGE (WP)	INDIVIDUAL TASKS (IT)
WP1- Exploratory Data Analysis	• **IT1.1- Analysis of the Statistical Summary of the dataset.** • **IT1.2- Univariate analysis (PDF, CDF, Box-plots, Violin-plots)** • **IT1.3- Bivariate Analysis (Scatter plots and Pair plots)**
WP2- Data Pre-processing	• **IT2.1- Outlier Detection** • **IT2.2- Handling Missing Values** • **IT2.3- Feature Encoding** • **IT2.4- Feature Scaling** • **IT2.5- Handling Imbalance** • **IT2.6- Feature Selection**
WP3- Model Training	• **IT3.1- Building Various Models (on the standard hyper-parameters).** • **IT3.3- Trying out the various combinations of the pre-processing methods and choosing the one based on best training accuracy.**
WP4- Model Evaluation	• **IT4.1- Evaluating the performance of these models based on 10 fold cross validation score.** • **IT4.2- Comparative analysis of these models to get the best ones.**
WP5- Hyper-parameter Optimization	• **IT5.1-Evaluating the impact of GridSearchCV on the best models.** • **IT5.2-Evaluating the impact of RandomSearchCV on the best models.** • **IT5.3- Comparative analysis of the 2 techniques to obtain the optimal set of parameters.**
WP6- Model Validation	• **IT6.1- Testing the best tuned models on the unseen test data using various evaluation metrics.** • **IT6.2- Comparative analysis of these tuned models to get the most accurate prediction system.**
WP7- Thesis Writing	• **IT7.1- Introduction** • **IT7.2- Literature Review** • **IT7.3- Challenges Identified** • **IT7.4- Proposed Methodology** • **IT7.5- Results and Analysis** • **IT7.6- Conclusion and Future Work**

Figure 5.1: Work Packages and Tasks

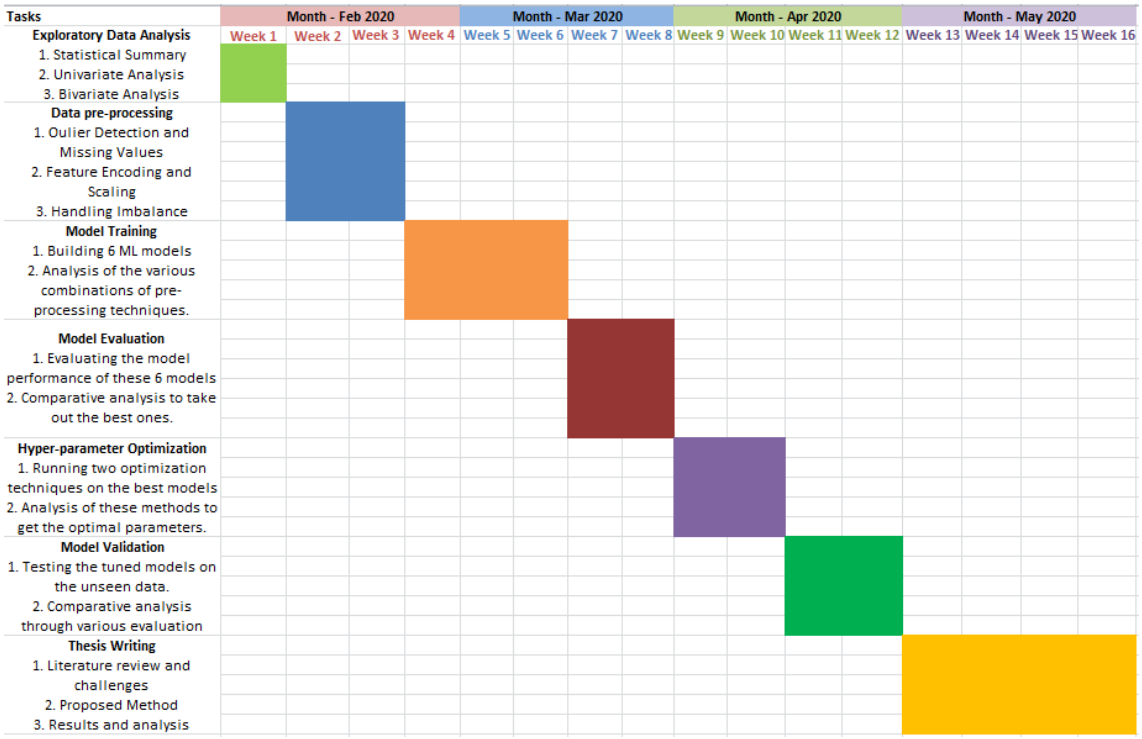

Figure 5.2: Gantt Chart of the Work Plan

Chapter 6

Summary

This research proposal contains the sequential step by step approach towards the development of an accurate machine learning system that can predict whether the person is having chronic kidney disease or not given the results of certain clinical tests of an individual. This problem is considered an important issue in the field of medical science. This system could facilitate the timely diagnosis of the patients that are predicted with the kidney disease which can prevent the further complications in the human body that could be created because of the inability of the kidney functions. We did the background research on the existing solutions and identified the important challenges in this prediction task. This formed a basis for our proposed work where the main focus is on handling the imbalance, selecting the most appropriate features via various feature selection mechanisms and the hyper-parameter optimization of the best models obtained from the comparative analysis of the six different machine learning models that would be trained on the standard parameters. We also discussed the implementation details of the each of the steps in our developed kidney disease prediction process flow. Next all the components (work packages) of our work plan is broken down into the individual tasks which are allocated an appropriate time frame with the development of the Gantt Chart that needs to be executed in the next semester. Finally an accurate kidney disease prediction system could be deployed in the hospitals as the handy tools for the doctors that could save lives and reduce the cost of tests that are associated with the disease.

Bibliography

[1] Abeer Y Al-Hyari, Ahmad M Al-Taee, and Majid A Al-Taee. Clinical decision support system for diagnosis and management of chronic renal failure. pages 1–6, 2013.

[2] Anusorn Charleonnan, Thipwan Fufaung, Tippawan Niyomwong, Wandee Chokchueypattanakit, Sathit Suwannawach, and Nitat Ninchawee. Predictive analytics for chronic kidney disease using machine learning techniques. In *2016 Management and Innovation Technology International Conference (MITicon)*, pages MIT–80. IEEE, 2016.

[3] Manish Kumar. Prediction of chronic kidney disease using random forest machine learning algorithm. *International Journal of Computer Science and Mobile Computing*, 5(2):24–33, 2016.

[4] Sai Prasad Potharaju and M Sreedevi. Ensembled rule based classification algorithms for predicting imbalanced kidney disease data. *J Eng Sci Technol Rev*, 9(5):201–207, 2016.

[5] L Jerlin Rubini and P Eswaran. Generating comparative analysis of early stage prediction of chronic kidney disease. *International Journal of Modern Engineering Research (IJMER)*, 5(7):49–55, 2015.

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

[7] Nusrat Tazin, Shahed Anzarus Sabab, and Muhammed Tawfiq Chowdhury. Diagnosis of chronic kidney disease using effective classification and feature selection technique. In *2016 International Conference on Medical Engineering, Health Informatics and Technology (MediTec)*, pages 1–6. IEEE, 2016.

[8] Kunwar Veenita, Khushboo Chandel, A Sai Sabitha, and Abhay Bansal. Chronic kidney disease analysis using data mining classification techniques. In *6th International ConferenceCloud System and Big Data Engineering (Confluence)*, 2016.

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

[10] S Vijayarani, S Dhayanand, and M Phil. Kidney disease prediction using svm and ann algorithms. *International Journal of Computing and Business Research (IJCBR)*, 6(2), 2015.