Comparative study of chronic kidney disease prediction using machine learning approaches

P Saran Pandian
Akash Gupta
Bhavya Jain
Sharvari Gokhale

***Abstract*—Chronic Kidney Disease prediction is one of the most important issues in medical decision making. The discovery of CKD prediction is an important task because it depends on experts of doctor knowledge. Construct effective CKD prediction in time is essential to prevent healthy patients. Chronic kidney disease is one of the leading causes of death and early prediction of chronic kidney disease is important. Prediction is one of the most interesting and challenging tasks in day to life. Data mining plays an essential role in the prediction of the medical dataset. Classification algorithms in machine learning can be used to predict the disease in the early stage with the data related to symptoms such as diabetes, blood pressure, RBC count, etc. In this paper, we will look into algorithms such as SVM, KNN, XGBoost, Random Forest.**

Keywords— Data Mining, Machine Learning, Chronic Kidney Disease, Classification, Support Vector Machine, K- Nearest Neighbours, Random Forest, XGBoost, MICE, KNN Imputer, One Hot Encoding

I. Introduction

Machine learning algorithms have been widely applied in many domains like medicine, finance, etc Machine Learning (ML) is already lending a hand in diverse situations in healthcare. ML in healthcare helps to analyze thousands of different data points and suggest outcomes, provides timely risk scores, precise resource allocation, and has many other applications. In this paper, we will look into one of its applications in the medical domain of predicting the risk of chronic kidney disease(CKD). a total of 24 features such as blood pressure, red blood cells count, blood sugar level, etc is used to predict the risk of CKD. this becomes a classification problem where statistical methods are applied to predict the risk. the methods to be applied are K- Nearest Neighbours, Support Vector Machine and tree-based algorithms such as Random Forest and XGBoost.

Some machine learning algorithms can be used to pick up the most significant features among all. this enables medical researchers to identify the symptoms which contribute to chronic kidney disease. Anemia is a common complication of chronic kidney disease (CKD). CKD means your kidneys are damaged and can’t filter blood the way they should. This damage can cause wastes and fluid to build up in your body. CKD can also cause other health problems.

Thus hemoglobin level can contribute to CKD. we will look into the importance of hemoglobin level in predicting CKD and also other features which are important.

II. Methods

A. *Dataset and preprocessing:*

Dataset:

We selected the dataset "Chronic_Kidney_Disease DataSet " from UCI Machine Learning Repository [a]
The dataset is gathered from many medical labs, pharmacies,community health centers and hospitals. In our dataset we have 400 rows and 25 columns . There are 13 categorical columns and 11 numerical columns and 1 column for class, there are only two classes in our dataset which are:

a.Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

CKD - having Chronic Kidney Disease
notCKD - not having Chronic Kidney Disease

The features in dataset are : age , blood pressure , specific gravity ,albumin ,sugar, red blood cells, pus cells, pus cell clumps, bacteria, blood glucose random, blood urea, serum creatinine , sodium, potassium, haemoglobin, packed cell volume, white blood cell count , red blood cell count , hypertension, diabetes mellitus , coronary artery disease, appetite, pedal edema ,anemia ,class.

Feature engineering:

Data Preprocessing is an integral step in Machine Learning as the quality of data and the useful information that can be derived from it directly affects the ability of our model to learn ;therefore ,it is extremely important that we preprocess our data before feeding it into our model. Data Preprocessing aims to reduce the data size , find relations between data , normalize data, remove outliers and extract features for data.

Missing data are often encountered for various reasons in biomedical research and present challenges for data analysis. It is well known that inadequate handling of missing data may lead to biased estimation and inference. In our dataset we had many missing values, numerical columns with missing values were handled using MICE imputation technique, and categorical columns having missing values were handled using KNN Imputer.

We briefly introduce the two imputation techniques that we used :

MICE : Multivariate Imputation by Chained Equations or sometimes called "sequential regression multiple imputation " has emerged in the statistical literature as one principled method for addressing missing data. It works on assumption of missingness at random (MAR), i.e., missingness only depends on observed values. Creating multiple imputations , as opposed to single imputations,account for the uncertainty in the imputations. In addition, the chained equations approach is very flexible and can handle variables of varying types(e.g. continuous or binary). [1]

KNN imputation : KNN finds the k most relevant nearest neighbors for a instance with missing data from complete instances using Euclidean distance, and the contribution of each instance is weighted to replace the missing data.[2s]
Since our dataset has 13 categorical columns , we need to convert them into numerical type so we can perform data analysis techniques on them. For this we have used two methods , one is Label Encoding and other is One Hot Encoding.

Few additional Feature engineering techniques were applied on the dataset in use to make the analysis more viable and conducive to the algorithms which will be further used:

1. Unscaled data with Label encoding

2. Scaled data with Label Encoding

3. Unscaled and One-Hot-Encoded data

4. Scaled and One-Hot-Encoded data

B. *Support Vector Machine*

Support Vector Machine (SVM) is a machine learning algorithm used for classification and regression. It is widely known for delivering higher performance on classification problems like the one in context, efficiently. SVM achieves this by outputting a hyperplane in n-dimensional space that segregates and classifies the data as per the unique classes. The hyperplanes created by the algorithm can be many and out of them the one which maximizes the distance between the hyperplane and the data points is selected to act as a classification plane. The deciding hyperplane is selected by using data points closer to the hyperplanes or at the boundaries of the data cluster.[4]These data points are called support vectors. (Fig.1.)

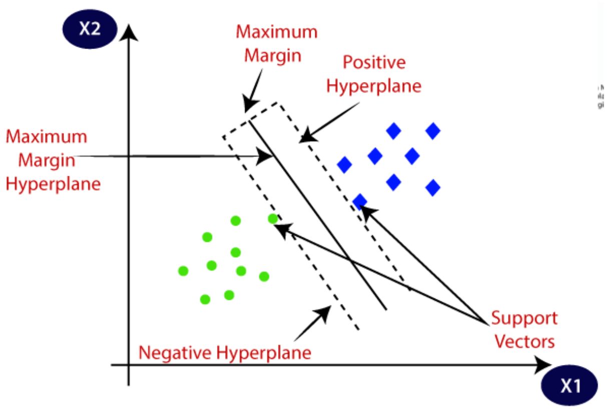

Fig.1. HyperPlanes for SVM

SVM uses a method known as the kernel trick. The kernels responsible for compounding and expanding and transforming the data, help to compute optimal boundaries around the data for efficient classification. The kernels used for this experiment is *Linear* kernel and non-linear kernels like – *Gaussian or Radial Basis Function*(rbf), *Polynomial*(poly), and *Sigmoid.*

All of the four kernels were applied to obtain results of the above-mentioned data transformations. The hyperparameters help in approaching and building the classification model with a more robust approach. Tuning the hyperparameters reduces the chance of overfitting or underfitting the training model. The parameters used in the SVM model were C and gamma. C parameter, or the Cost parameter, decides the acceptable misclassification of the data points by SVM. Gamma decides the curvature weight to be given to the decision boundary or the optimal hyperplane.

After choosing the optimal C and gamma values, the most favorable results were obtained for Gaussian or the Radial Bias Function method or the rbf kernel, with the scaled and One-Hot-Encoded data.

C. *K Nearest Neighbour:*

The k-Nearest-Neighbours (kNN) is a non-parametric classification method, which is simple but effective in many cases.[3][4] For a data record t to be classified, its k nearest neighbours are retrieved, and this forms a neighbourhood of t. Majority voting among the data records in the neighbourhood is usually used to decide the classification for data records with or without consideration of distance-based weighting.

However, to apply kNN we need to choose an appropriate value for k, and the success of classification is very much dependent on this value. In a sense, the kNN method is dependent on the value of k. In order to choose k different k values were used and the k having most accuracy was considered.

The major drawbacks with respect to kNN are its low efficiency ,works with less number of inputs and its dependency on the selection of a “good value” for k.

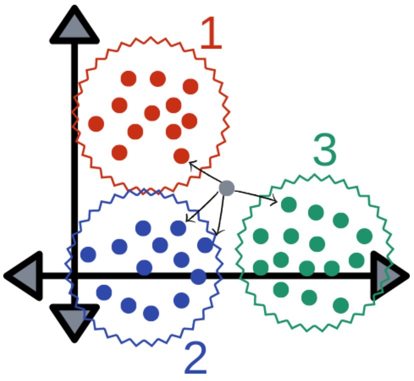

Fig.2. Classification in K-NN using distance measures

As far as results are concerned one hot encoding with a standard scaler gave the best accuracy.

D. *Random Forest:*

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble.[3][5] Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model’s prediction.

The fundamental concept behind random forest is a simple but powerful one — the wisdom of crowds. the reason that the random forest model works so well is: A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.

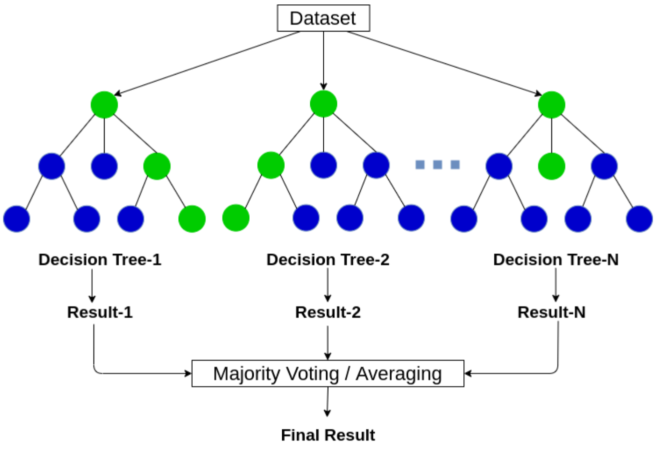

Fig.3. Ensembled Model of Decision Trees (Random Forest)

Bagging (Bootstrap Aggregation) — Decisions trees are very sensitive to the data they are trained on — small changes to the training set can result in significantly different tree structures. Random forest takes advantage of this by allowing each individual tree to randomly sample from the dataset with replacement, resulting in different trees. This process is known as bagging. Applying random forest on the dataset by tuning the number of estimators parameter we found that 30 and 50 estimators for label encoded and one hot encoded data respectively were able to provide very good comparable results. 'Gini' impurity was used to choose the root nodes of the tree.

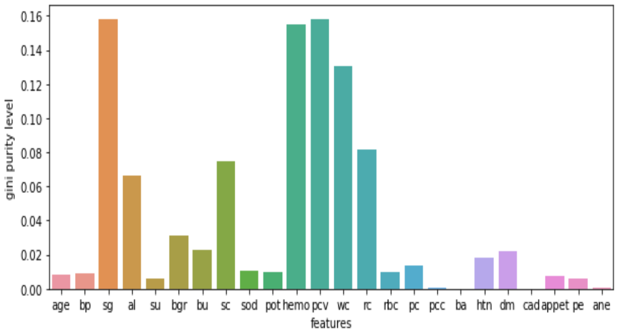

Fig.4. GINI Purity Plot

To know which features contributed to the final result we plotted the purity plot which showed the importance of each feature. (Fig.4.) We found that Hemoglobin level, Packed Cell Volume, Albumin, Serum Creatinine, White Blood Cells count, Red Blood Cells count contributed more to the prediction. Some features were so impure that they did not contribute to the prediction. These features will be neglected or pushed down of the Decision Tree by the algorithm. Hence Random Forest is able to pick the most significant features and give better results. Better results were given by one hot encoded dataset.

E. XGBoost(Extreme Gradient Boosting):

The term ‘Boosting’ refers to a family of algorithms that converts weak learners to strong learners. Boosting is an ensemble method for improving the model predictions of any given learning algorithm. One such boosting algorithm is XGBoost. eXtreme Gradient Boosting (XGBoost) is a scalable and improved version of the gradient boosting algorithm designed for efficiency, computational speed, and model performance. In addition to the traditional Boosting algorithm, it includes Pruning which is a machine learning technique to reduce the size of regression trees by replacing nodes that don’t contribute to improving classification on leaves. The idea of pruning a regression tree is to prevent overfitting of the training data. The most efficient method to do pruning is Cost Complexity or Weakest Link Pruning which internally uses mean square error, k-fold cross-validation, and learning rate. XGBoost creates nodes (also called splits) up to the maximum depth specified and starts pruning from backward until the loss is below a threshold.

Applying the XGBoost Algorithm, we got a very good result compared to the Random Forest algorithm. Since the algorithm uses pruning and stops creating nodes within the specified maximum depth it is able to pick the most important features from the dataset and avoid overfitting.

III. Results

A. *Experimental Results:*

With the appropriate selection of the evaluation parameters, a proper measure of model performance is gauged. The assessment parameters selected must be suitable metrics for the algorithms. The performance evaluation parameters selected in this paper are:

1) *Accuracy:* In general, the accuracy metric measures the ratio of correct predictions over the total number of instances evaluated.

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

2) *Precision:* Precision is used to measure the positive patterns that are correctly predicted from the total predicted patterns in a positive.

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

3) *Recall:* Recall is used to measure the fraction of positive patterns that are correctly classified.

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

4) *F1 score:* This metric represents the harmonic mean between recall(R) and precision(P) values.

$$F1\,Score = \frac{2 * P * R}{P + R}$$

The four methods used – SVM, K-NN, XGBoost, and Random Forest, performed to each of their best efficacy. Certain observations made about the model performance were that scaled OHE data is giving better results for the K-NN algorithm. Unscaled OHE data is giving better results for the Random Forest algorithm while XGBoost is capable of producing good results irrespective of any data transformations.

ML Classifiers Used	Evaluation Parameters			
	Accuracy	Precision	F1	Recall
SVM	99.37	98.75	99.24	100.00
K-NN	65.90	78.47	66.77	93.81
XGBoost	99.24	100.00	98.81	99.40
RandomForest	99.24	100.00	98.81	99.40

Table 1. Performance Evaluation Metrics

To summarize, from the table (Table. 1.), while the Recall measure for K-NN is elevated, comparatively, other measures aren’t performing well. XGBoost and RandomForest surpass all the other algorithms used for the analysis in terms of Accuracy, Precision, F1 score as well as Recall.

IV. Conclusion

For Feature Engineering, since the dataset was filled with a lot of null values some robust techniques of imputation were necessary to impute the null values. So MICE and KNNImputer algorithms were applied to fill numerical and categorical data respectively instead removing or imputing with mean, median values. As these were robust techniques they performed well when applied to the models.

For dealing with categorical data both label and one hot encoding techniques were applied which showed good results.

when different machine learning models such as SVM, KNN, Random forest and XGboost classifiers were used, it was found that the SVM, Random Forest, XGboost algorithms performed well with high accuracy and f1 score, but KNN performed worst as the level clusters were so impure (as seen in the plot). Due to the fact that KNN is not a robust method it showed poor results compared to other algorithms.

SVM showed better results as it uses different kernels for the model which is able to capture classify with better accuracy.

XGBoost and Random Forest are Tree-based algorithms. Hence they are able to capture the important features and neglect the effect of the least important features. We can conclude that this model works better with XGBoost and Random forest methods compared to other methods.

References

[1] Azur, M. J., Stuart, E. A., Frangakis, C., & Leaf, P. J. (2011). Multiple imputation by chained equations: what is it and how does it work?. *International journal of methods in psychiatric research*, *20*(1), 40-49.

[2] Pan, R., Yang, T., Cao, J., Lu, K., & Zhang, Z. (2015). Missing data imputation by K nearest neighbours based on grey relational structure and mutual information. *Applied Intelligence*, *43*(3), 614-632.

[3] Devika, R., Avilala, S. V., & Subramaniyaswamy, V. (2019, March). Comparative study of classifier for chronic kidney disease prediction using naive Bayes, KNN and random forest.

[4] Sinha, P., & Sinha, P. (2015). Comparative study of chronic kidney disease prediction using KNN and SVM. International Journal of Engineering Research and Technology, 4(12),

[5] Saha, A., Saha, A., & Mittra, T. (2019, July). Performance measurements of machine learning approaches for prediction and diagnosis of chronic kidney disease (CKD). In Proceedings of the 2019 7th international conference on computer and communications.