Predictive Modeling for Chronic Kidney Disease Using Machine Learning

Hemangi Patil[1], Gaurav Acharya[2]

[1]*Independent Researcher, Mumbai, India*
[2]*Department of master's in computer science, IIT Chicago, Illinois*

Abstract - *Chronic Kidney Disease (CKD) is a global health problem since it starts asymptomatically and progresses in an irreversible manner. Therefore, the earlier and accurate diagnosis plays a significant role in improving patient outcomes. This paper presents a machine learning (ML) framework that handles data problems such as missing values and evaluated different ML-models for addressing the problem of CKD diagnosis. K-Nearest Neighbors (KNN) data imputation and classifier evaluation on six classifiers show Random Forest (RF) out of all trained classifiers performs best with the accuracy of 99.75%. A new hybrid method with Logistic Regression (LOG) and RF gives an additional improvement in accuracy (99.83%). This approach is scalable and adaptable in a clinical environment.*

Key Words: **Chronic Kidney Disease, Machine Learning, Data Imputation, Random Forest, Logistic Regression, Predictive Modeling**

1. INTRODUCTION

Chronic Kidney Disease (CKD) is a progressive condition characterized by the gradual decline of kidney function, often remaining undetected until it reaches advanced stages. Early detection is critical for preventing further complications and improving patient outcomes. However, traditional diagnostic methods are time-consuming and may not always be practical in clinical settings. In recent years, machine learning (ML) has emerged as a promising solution for automating CKD diagnosis, offering faster and more accurate predictions. Despite its potential, existing ML models face challenges such as incomplete datasets and limited generalization. This paper proposes a novel ML-based framework for CKD diagnosis that addresses these challenges. Specifically, it utilizes K-Nearest Neighbors (KNN) imputation to handle missing data, making the model applicable even when diagnostic categories are unknown or incomplete. The study evaluates several ML algorithms, including Logistic Regression (LOG), Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Naive Bayes (NB), and Feedforward Neural Networks (FNN) to establish CKD diagnostic models. Furthermore, a hybrid model combining Logistic Regression and Random Forest is introduced to improve the accuracy of the predictions. This hybrid model achieves an impressive accuracy of 99.83%, highlighting its effectiveness and potential for clinical adoption in CKD diagnosis.

1.2. OBJECTIVES

The main objective of this study is to develop an early diagnostic model for Chronic Kidney Disease (CKD) that minimizes testing and cost while achieving high accuracy. The study's objectives are to use K-Nearest Neighbors (KNN) imputation to manage missing data in the CKD dataset and feature selection with information gain to determine which features are most crucial for CKD identification. To find the best accurate model, a variety of machine learning methods will be used and compared, such as Feedforward Neural Networks, Random Forest, Support Vector Machine, K-Nearest Neighbor, Naive Bayes, and Logistic Regression. The objective is to use 24 important predictors to predict the presence of CKD and evaluate the accuracy and other pertinent metrics of various algorithms.

2. LITERATURE SURVEY

Different papers and articles have been reviewed for this project. Also, their conclusions are summarized in this section. The section present documents that were studied prior and post project development. The mentioned articles provide with a better understanding about structure of the system and how various algorithms could be combined together so as to build a system with higher efficiency.

Table -1: Publications Cited:

Title	Year	Author	Summary
Diagnosis of patients with chronic kidney disease by using two fuzzy classifiers	2016 (Chemometr. Intell. Lab.)	Z. Chen et al.	Used fuzzy classifiers to diagnose CKD, handling incomplete datasets for better accuracy.
Diagnosis of chronic kidney disease by using random forest	2017 (Int. Conf. Medical and Biological Engineering)	A. Subasi, E. Alickovic, J. Kevric	Applied Random Forest for CKD prediction, with emphasis on feature selection.

Prevalence of chronic kidney disease in China: A cross-sectional survey	2012 (Lancet)	L. Zhang et al.	Explored CKD prevalence in China, underlining the need for early detection.
Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration	2015 (J. Biomed. Inform.)	A. Singh et al.	Integrated temporal EHR data to improve CKD risk prediction.
Prevalence of chronic kidney disease in an adult population	2014 (Arch. Med. Res.)	A. M. Cueto-Manzano et al.	Focused on CKD prevalence, stressing the importance of early diagnosis.
Diagnosis of chronic kidney disease based on support vector machine by feature selection methods	2017 (J. Med. Syst.)	H. Polat, H.D. Mehr, A. Cetin	Used SVM with feature selection for improved CKD diagnosis.
A new machine learning approach for predicting the response to anemia treatment in end-stage renal disease patients	2015 (Comput. Biol. Med.)	C. Barbieri et al.	Applied ML to predict anemia treatment responses in end-stage renal disease.

3. TECHNICAL DEFINATION

3.1 LOGISTIC REGRESSION

Logistic regression is also used to estimate the relationship between a dependent variable and one or more independent variables, but it is used to make a prediction about a categorical variable versus a continuous one. A categorical variable can be true or false, yes or no, 1 or 0, et cetera. The unit of measure also differs from linear regression as it produces a probability, but the logit function transforms the S-curve into straight line.

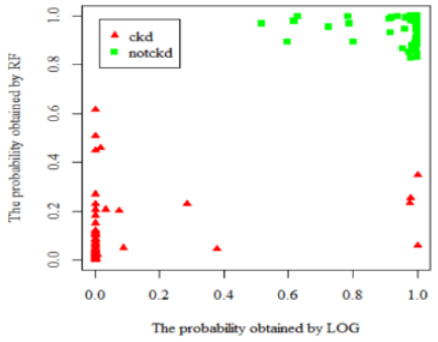


Fig. 3.1- Logistic Regress

3.2 K-NEAREST NEIGHBOR:

K-Nearest Neighbor is one of the simplest and most used supervised machine learning algorithms. Technically it does not train any dataset; instead, an observation is predicted to fall under those classes which have the largest proportion of k-nearest neighbors around it. Distance is considered to be a metric to determine similarity. For instance, the closest data point around the point under observation can be considered most similar to the data point. There are a large variety of distance-metrics like Euclidean distance (d).

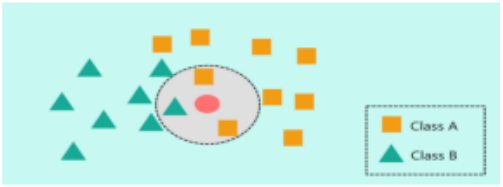


Fig. 3.2 K-Nearest Neighbor.

3.3 SUPPORT VECTOR MACHINE (SVM):

Support Vector Machine is one of the most robust algorithms based on the statistical learning framework which offers solutions for both regression and classification problems. Using the kernel trick, SVM can classify both linear and non-linear datasets. The datasets are separated by a (n-1) hyper plane, where every data point is considered to be an n-dimensional vector. For a two-dimensional space, the hyperplane is a line separating a plane in two parts.

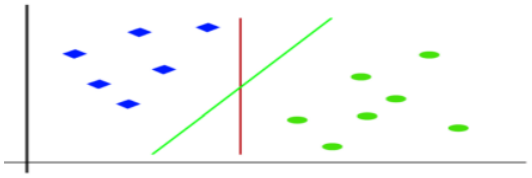


Fig. 3.3 - Support Vector Machine.

3.4 RANDOM FOREST (RF):

Random Forest is a learning algorithm which operates by creating multiple decision trees at training time and providing an output class of individual trees. It is applicable for both regression and classification. This model does a small tweak that utilizes the de-correlated tree by building a multitude of decision trees on bootstrapped samples from training data; this process is known as bagging. During bootstrapping, it filters a few numbers of feature columns out of all feature columns.

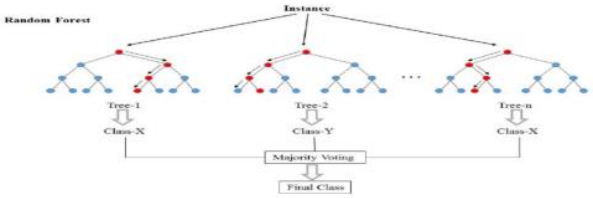


Fig. 3.4 – Decision Tree

3.5 NAÏVE BAYES (NB):

Naïve Bayes is a supervised algorithm which imposes independence of features while classifying data. This model is an effective tool for datasets which have a high number of input features. It considers all the features available including some of the features that have weak effects on the final prediction.

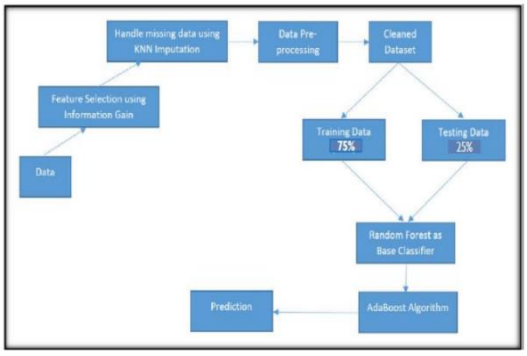


Fig. 3.5 - Naïve Bayes classifier

3.6 MULTILAYER PERCEPTRON:

A multilayer perceptron (MLP) is a feed-forward artificial neural network made up of several layers of perceptron. It contains nodes of at least three layers named input node, hidden layer and the output node. This network uses a non-linear activation function which maps weighted inputs to each neuron output. In this paper, we used sigmoid functions as the activation functions.

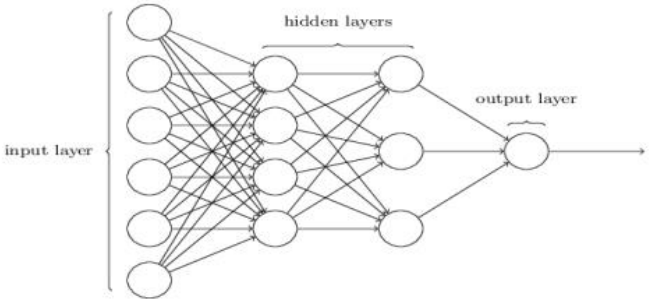


Fig. 3.6 - Multilayer Perceptron

4 PROPOSED SOLUTION

The proposed solution is intended to demonstrate the effectiveness of the Hybrid Machine Learning Models over the individual Machine Learning Models providing the remarkable efficiency and accuracy in the prediction of the Chronic Kidney Disease with the highest achieved accuracy of the 99.83%.

4.1 REQUIREMENTS

4.1.1 HARDWARE REQUIREMENT

- Laptop (RAM 8GB)
- Storage (500 GB)
- Monitor (15” LED)
- Processor (Pentium i3 / i5 Processor)

4.1.2 SOFTWARE REQUIREMENT

- Windows 10 Operating System.
- Jupyter Notebook
- Python Libraries (Numpy, Pandas, Matplotlib, SKLearn, SeaBorn)

5. SYSTEM DESIGN

SYSTEM ARCHITECTURE

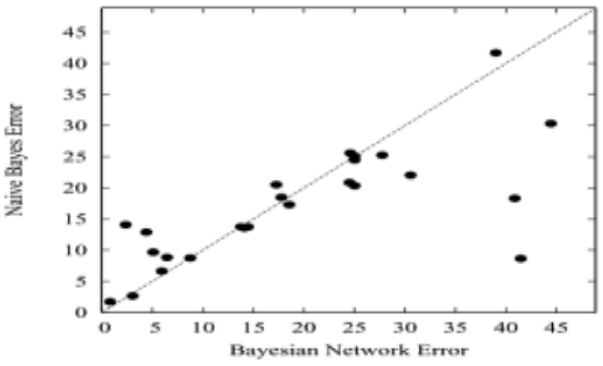


Fig. 5.1 System Architecture

5.1. ACQUIRE THE DATASET:

The dataset for predicting chronic kidney disease was taken from the UCI machine learning repository, a well-known source of all machine learning datasets. The CKD dataset contains 400 patient records and 25 characteristics.

These are either symptoms or other disease-related characteristics such as hypertension, blood pressure, specific gravity, albumin, and so on. Among the 400 patient records, 250 have the disease, whereas the remaining 150 do not.

Variables	Explain	Class	Scale	Missing Rate
age	Age	Numerical	age in years	2.25%
bp	Blood Pressure	Numerical	in mm/Hg	3%
sg	Specific Gravity	Nominal	(1.005,1.010,1.015,1.020,1.025)	11.75%
al	Albumin	Nominal	(0,1,2,3,4,5)	11.5%
su	Sugar	Nominal	(0,1,2,3,4,5)	12.25%
rbc	Red Blood Cells	Nominal	(normal,abnormal)	38%
pc	Pus Cell	Nominal	(normal,abnormal)	16.25%
pcc	Pus Cell clumps	Nominal	(present,notpresent)	1%
ba	Bacteria	Nominal	(present,notpresent)	1%
bgr	Blood Glucose Random	Numerical	in mgs/dl	11%
bu	Blood Urea	Numerical	in mgs/dl	4.75%
sc	Serum Creatinine	Numerical	in mgs/dl	4.25%
sod	Sodium	Numerical	in mEq/L	21.75%
pot	Potassium	Numerical	in mEq/L	22%
hemo	Hemoglobin	Numerical	in gms	13%
pcv	Packed Cell Volume	Numerical	-	17.75%
wbcc	White Blood Cell Count	Numerical	in cells/cumm	26.5%
rbcc	Red Blood Cell Count	Numerical	in millions/cmm	32.75%
htn	Hypertension	Nominal	(yes,no)	0.5%
dm	Diabetes Mellitus	Nominal	(yes,no)	0.5%
cad	Coronary Artery Disease	Nominal	(yes,no)	0.5%
appet	appet	Nominal	(good,poor)	0.25%
pe	Pedal Edema	Nominal	(yes,no)	0.25%
ane	Anemia	Nominal	(yes,no)	0.25%
class	Class	Nominal	(ckd,notckd)	0%

Fig 5.2 Dataset Attributes and Missing value Percentage

5.2. IMPORT ALL THE CRUCIAL LIBRARIES:

To effectively execute data preprocessing in Python, it is imperative to import a range of predefined libraries, each tailored to fulfill specific roles within the realm of machine learning. In the present study, a selection of libraries, including NumPy, pandas, matplotlib, and scikit-learn (sklearn), has been utilized. Each library is designated for particular tasks that collectively contribute to the data processing and analytical phases of the research.

5.3 IDENTIFY AND MANAGE MISSING VALUES:

Next, it is very essential to manage the missing values in a dataset as it will cause a huge issue later while applying the machine learning algorithms. Hence it is essential to handle missing values that are there in the dataset. The CKD dataset has various missing values as some of the patients might forget or miss to fill in certain values in real life. Hence, KNN imputation is going to be used to handle missing values as it has proven to be effective in experiments. Every missing value can then be handled by replacing them with the mean value of the k-nearest neighbors, to do so in general Euclidean distance metric is used by default.

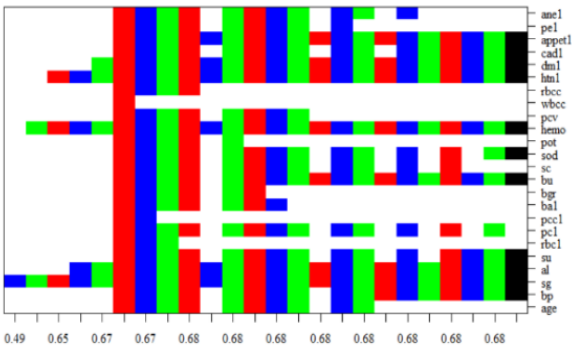


Fig. 5.3 - Finding the Missing values and introducing impostor values.

5.4. FEATURE SELECTION:

A Machine Learning model can suffer from the issue of overfitting if the number of features becomes identical or huge. To prevent this from happening it is important to reduce the number of features in the dataset. Feature selection refers to reducing the number of features in a dataset when building a machine learning model. Another advantage of decreasing the number of input attributes is the decrease in the cost of building the ML model and in a few situations, it might even improve the accuracy of the model.

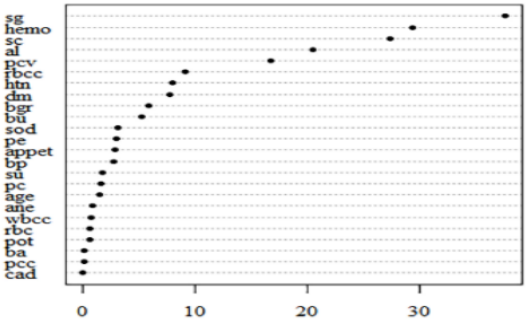


Fig. 5.4 - Results of Important Feature Extraction.

5.5. IDENTIFY AND MANAGE MISSING VALUES:

Next, it is very essential to manage the missing values in a dataset as it will cause a huge issue later while applying the machine learning algorithms. Hence it is essential to handle missing values that are there in the dataset. The CKD dataset has various missing values as some of the patients might forget or miss to fill in certain values in real life. Hence, KNN imputation is going to be used to handle missing values as it has proven to be effective in experiments. Every missing value can then be handled by replacing them with the mean value of the k-nearest neighbors, to do so in general Euclidean distance metric is used by default.

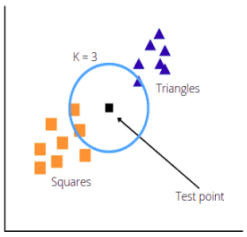


Fig. 5.5 - Calculating Missing Value

5.6. ENCODING THE CATEGORICAL DATA:

Categorical data in a dataset are the ones that have certain categories like in this dataset there are categorical variables like packed cell volume, hypertension and classification. These categorical data might cause a huge challenge while building the model as machine learning only deals with numbers and mathematics. So, it is essential to convert these categorical variables into numbers. There are various techniques for this purpose and in this paper, Label Encoding of the SKLearn library is going to be used.

Attributes	Information
age- Age	Discrete Integer Values
bp- Blood pressure	Discrete Integer Values
al- Albumin	Nominal Values (0,1,2,3,4,5)
su- Sugar	Nominal Values (0,1,2,3,4,5)
rbc- Red blood cells	Nominal Values (Normal, Abnormal)
pc- Pus cell	Nominal Values (Normal, Abnormal)
pcc- Pus cells clumps	Nominal Values (Present, Not-Present)
ba-Bacteria	Nominal Values (Present, Not-Present)
bgr- Blood Glucose Random	Numerical Values in mgs/dl
bu- Blood Urea	Numerical Values in mgs/dl
sc- Serum creatinine	Numerical Values
sod- Sodium	Numerical Values in mEq/L
pot- Potassium	Numerical Values in mEq/L
hemo- Hemoglobin	Numerical Values in gms
pcv- Packed Cell Volume	Numerical Values
wc- White blood cell count	Discrete Integer Values
rc- Red blood cell count	Numeric Values
htn- Hypertension	Nominal Values (Yes, No)
dm- Diabetes Mellitus	Nominal Values (Yes, No)
cad- Coronary Artery Disease	Nominal Values (Yes, No)
appet- Appetite	Nominal Values (Good, Poor)
pe- Pedal Edema	Nominal Values (Yes, No)
ane- Anemia	Nominal Values (Yes, No)
class- Classification	Nominal Values (CKD, Not CKD)

Fig. 5.6 Attributes and Classes in each Attribute

5.7. SPLITTING THE DATASET:

All the datasets must be split into the training subset and the testing subset to proceed with the prediction. The training subset is used to perform training whereas the testing subset is used to perform testing. One is aware of the results in the training subset but one is not 13 aware of the prediction result or output of the testing subset. The CKD dataset is going to be split in the ratio of 75:25.

K value	Model	The numbers of misjudgments
K=3	LOG	189, 225, 42, 193, 212
	RF	166
	SVM	225, 118
	FNN (23)	**90, 189, 225, 210**
	KNN	193, 106, 103, 210
K=5	LOG	189, 225, 193, 212, 267
	RF	166
	SVM	225, 118
	FNN (27)	**90, 225, 229, 210**
	KNN	183, 193, 229, 132, 103, 210
K=7	LOG	90, 92, 189, 225, 193, 212, 267, 340
	RF	90, 166
	SVM	225, 118
	FNN (27)	**90, 225, 229, 210**
	KNN	183, 193, 229, 132, 103, 210
K=9	LOG	92, 189, 225, 193, 212
	RF	166
	SVM	225, 118
	FNN (23)	**90, 225**, 229, **210**
	KNN	183, 132, 103, 210
K=11	LOG	189, 225, 193, 212
	RF	90, 166
	SVM	225, 118
	FNN (5)	**225**, 229
	KNN	183, 132, 103

Fig. 5.7 Splitting Dataset

5.8. MODEL DEVELOPMENT AND EVALUATION SCHEME:

Developing individual ML models and evaluating them the various machine learning algorithms applied for prediction are:

- Logistic Regression
- Naïve Bayes
- Artificial Neural Network
- Random Forest

The performance of these classifiers is analyzed based on various metrics such as Accuracy, Precision, Recall, F-measure and loss and the best classifier for diagnosing chronic kidney disease is found.

6. SYSTEM FLOW DIAGRAM:

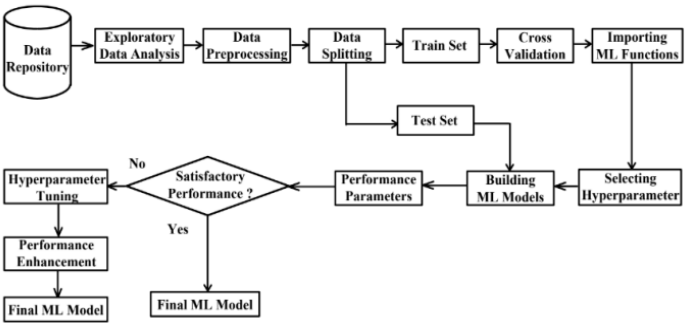


Fig. 6 - System Flow Diagram

7. METHODOLOGY:

7.1. DATASET BEFORE FEATURE SELECTION

The dataset after cleaning and preprocessing does not contain any null values thus before feature selection on the dataset the dataset is consistent and ready to be used.

1 .PERFORMANCE INDICATORS for Logistic Regression

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	0.996667	0.98	38	0	2	60
knn_imputed_5	1.000000	0.97	36	0	3	61
knn_imputed_7	1.000000	0.98	33	0	2	65
knn_imputed_9	0.996667	0.99	42	0	1	57
knn_imputed_11	0.996667	0.98	43	2	0	55
mean_imputed	1.000000	0.99	38	0	1	61

2 .PERFORMANCE INDICATORS for KNN

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	0.973333	0.99	40	0	1	59
knn_imputed_5	0.970000	0.96	35	1	3	61
knn_imputed_7	0.983333	0.98	38	0	2	60
knn_imputed_9	0.983333	1.00	46	0	0	54
knn_imputed_11	0.980000	0.98	36	0	2	62
mean_imputed	0.983333	0.94	28	0	0	66

3 .PERFORMANCE INDICATORS for Random Forest

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	1.0	0.99	37	1	0	62
knn_imputed_5	1.0	0.99	41	1	0	58
knn_imputed_7	1.0	1.00	36	0	0	64
knn_imputed_9	1.0	0.99	45	1	0	54
knn_imputed_11	1.0	0.96	36	3	1	60
mean_imputed	1.0	1.00	35	0	0	65

4 .PERFORMANCE INDICATORS for SVM

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	0.993333	0.98	33	1	1	65
knn_imputed_5	0.986667	0.99	34	1	0	65
knn_imputed_7	0.993333	1.00	40	0	0	60
knn_imputed_9	0.996667	0.99	38	0	1	61
knn_imputed_11	0.990000	0.99	38	1	0	61
mean_imputed	1.000000	1.00	39	0	0	61

5 .PERFORMANCE INDICATORS for Naive Bayes

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	0.930000	0.93	37	1	6	56
knn_imputed_5	0.953333	0.92	42	4	4	50
knn_imputed_7	0.953333	0.94	40	1	5	54
knn_imputed_9	0.943333	0.92	33	1	7	59
knn_imputed_11	0.950000	0.93	35	1	6	58
mean_imputed	0.943333	0.94	28	1	5	66

6 .PERFORMANCE INDICATORS for MultiLayer-Perceptron

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	1.0	0.98	36	1	1	62
knn_imputed_5	1.0	0.96	45	1	3	51
knn_imputed_7	1.0	0.99	41	0	1	58
knn_imputed_9	1.0	0.99	28	1	0	71
knn_imputed_11	1.0	0.99	34	0	1	65
mean_imputed	1.0	1.00	32	0	0	68

Fig. 7.1.1 - Dataset's Before feature selection.

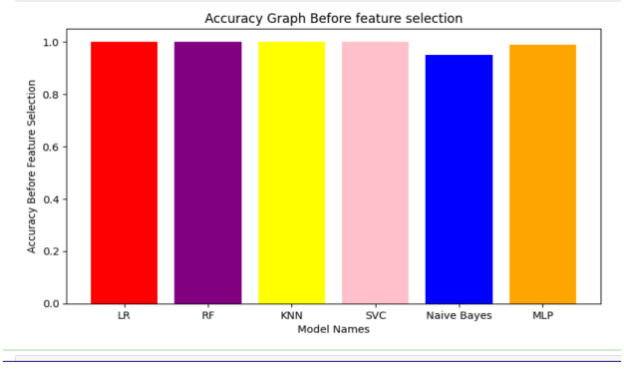


Fig. 7.1.2 - Models efficiency before Feature Selection

7.2 FEATURE SELECTION AND CORRELATION BETWEEN MODELS

After feature selection and finding the correlation between various prediction models the Heat-map is generated and is as follows

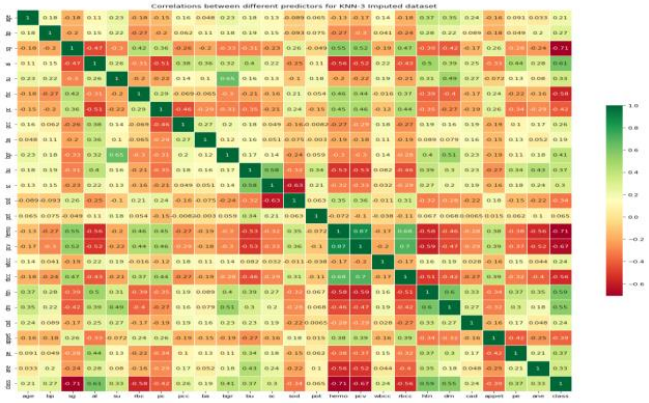


Fig. 7.2 - Heat-Map Correlation between Predictors

7.3 AFTER FEATURE SELECTION

The dataset after cleaning and preprocessing does not contain any null values thus after feature selection on the dataset, the dataset is consistent and ready to use.

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	0.996667	0.98	36	2	0	62
knn_imputed_5	0.996667	0.97	35	3	0	62
knn_imputed_7	1.000000	0.98	36	2	0	62
knn_imputed_9	1.000000	0.97	35	3	0	62
knn_imputed_11	0.996667	0.97	35	3	0	62
mean_imputed	1.000000	0.99	38	0	1	61

For Logistic Regression, we have false_pos + false_neg is 3,1,2,1,1,0 for respective imputed datasets.

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	1.0	0.99	34	1	0	65
knn_imputed_5	1.0	0.98	33	2	0	65
knn_imputed_7	1.0	0.98	33	2	0	65
knn_imputed_9	1.0	0.98	33	2	0	65
knn_imputed_11	1.0	0.98	33	2	0	65
mean_imputed	1.0	1.00	35	0	0	65

1 .PERFORMANCE INDICATORS for KNN

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	0.976667	0.97	37	1	2	60
knn_imputed_5	0.993333	0.95	37	1	4	58
knn_imputed_7	0.983333	0.98	38	0	2	60
knn_imputed_9	0.986667	0.99	38	0	1	61
knn_imputed_11	0.996667	0.98	38	0	2	60
mean_imputed	0.986667	0.99	38	0	1	61

2 .PERFORMANCE INDICATORS for SVC

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	0.993333	0.98	36	2	0	62
knn_imputed_5	0.996667	0.98	36	2	0	62
knn_imputed_7	1.000000	0.98	36	2	0	62
knn_imputed_9	0.996667	0.97	35	3	0	62
knn_imputed_11	0.996667	0.97	35	3	0	62
mean_imputed	0.996667	1.00	38	0	0	62

3 .PERFORMANCE INDICATORS for Naive Bayes

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	0.936667	0.91	33	5	4	58
knn_imputed_5	0.943333	0.92	33	5	3	59
knn_imputed_7	0.960000	0.93	34	4	3	59
knn_imputed_9	0.950000	0.92	33	5	3	59
knn_imputed_11	0.956667	0.95	34	4	1	61
mean_imputed	0.953333	0.94	33	5	1	61

4 .PERFORMANCE INDICATORS for MultiLeyer-perceptron

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	1.000000	0.98	36	2	0	62
knn_imputed_5	0.996667	0.97	36	2	1	61
knn_imputed_7	1.000000	0.97	35	3	0	62
knn_imputed_9	1.000000	0.97	35	3	0	62
knn_imputed_11	0.996667	0.97	35	3	0	62
mean_imputed	0.996667	0.99	38	0	1	61

Fig 7.3.1 - Datasets after feature selection

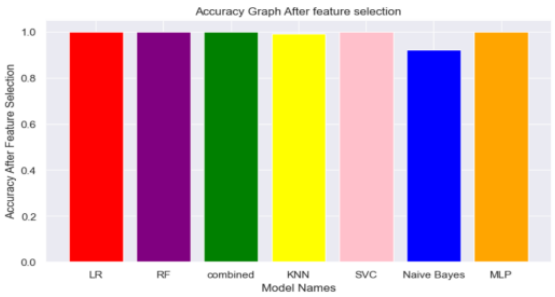


Fig 7.3.2 - Models efficiency after feature selection

7.4. ATTRIBUTE IMPORTANCE CALCULATION

As the dataset has 24 attributes for each record and some of them are more dominant and some have negligible impact on the results so we keep calculating the importance of each attribute and discard the attribute having the least importance. We have the least important attribute having the calculated importance weight of 0.01 or less(age, su, pc, sod, pot, wbcc, pcc, ba, cad, ane).

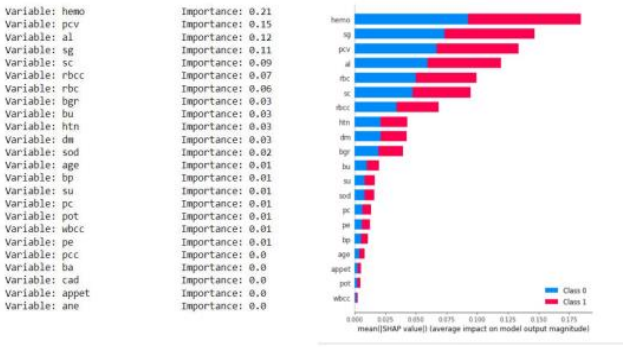
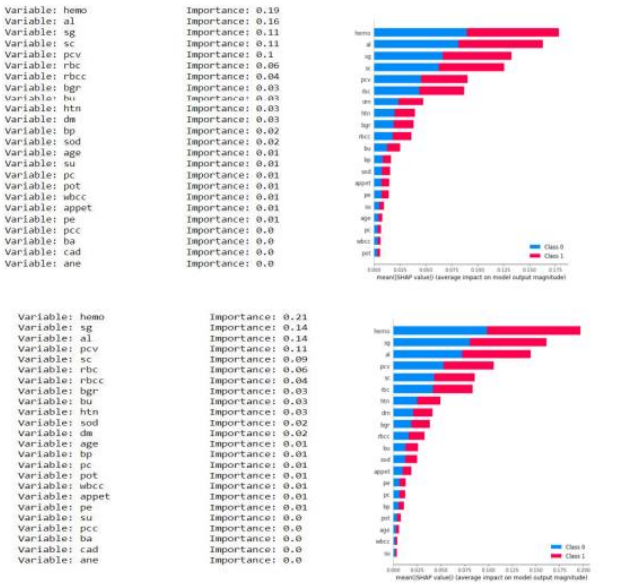


Fig 7.4.1 KNN imputation means Statistics and Graph.



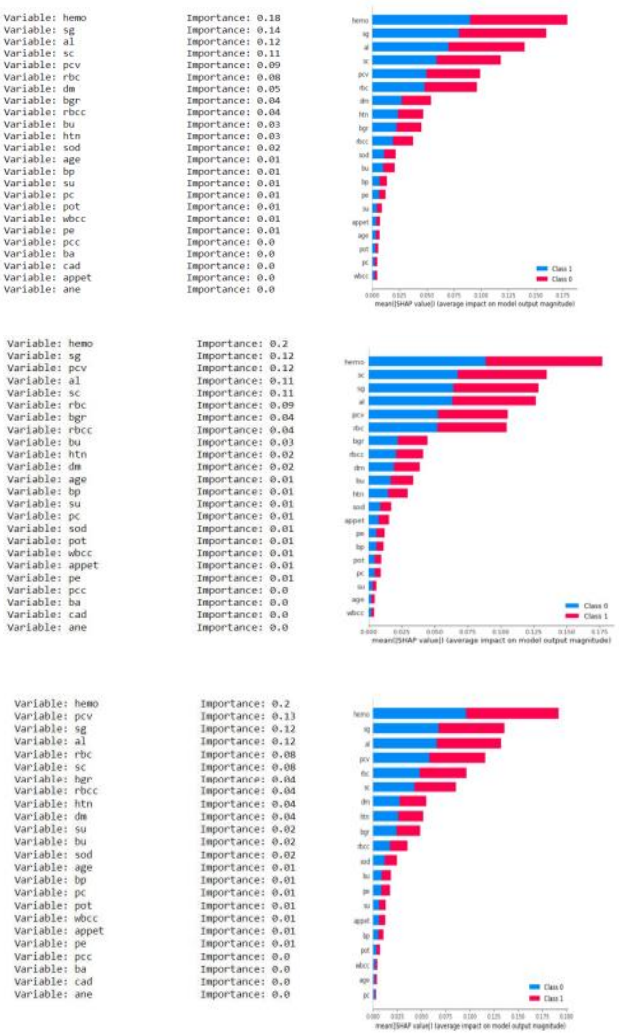


Fig. 7.4.2 - KNN Uniform Statistics and Grap

7.5. COMBINED MODELS LOGISTIC REGRESSION AND RANDOM FOREST

The combined approach of the Logistic Regression model and Random Forest model gives the result set with accuracy and precision 99.83%.

	train_acc	test_acc	true_neg	false_pos	false_neg	true_pos
knn_imputed_3	99.666667	100.0	35	0	0	65
knn_imputed_5	99.666667	98.0	33	2	0	65
knn_imputed_7	100.000000	99.0	34	1	0	65
knn_imputed_9	100.000000	98.0	33	2	0	65
knn_imputed_11	99.666667	98.0	33	2	0	65
mean_imputed	99.666667	100.0	35	0	0	65

Fig. 7.5 - Combined Logistic Regression and Random Forest

8. CONCLUSIONS:

The suggested hybrid model for predicting chronic kidney disease (CKD) outperformed separate machine learning models in terms of efficiency and reliability. The hybrid technique, which combines two or more models,

considerably improves forecast accuracy, obtaining an astounding 99.83%. This beats older techniques, which typically attain a maximum accuracy of 95.60% in their most well-known applications. The findings imply that model combinations can produce better outcomes, encouraging the usage of hybrid implementations in future diagnostic systems. This approach has the potential to significantly improve the accuracy and efficiency of CKD prediction, hence contributing to the growth of machine learning applications in healthcare.

9. FUTURE SCOPE:

Future enhancements to the underlying algorithms have the potential to increase the models' usefulness and performance. With ongoing development, it is possible that forecast accuracy may reach 100%, making these models even more dependable.

Advances in algorithmic design and structure could significantly improve accuracy rates, ensuring that models are better prepared to handle real-world data. This would increase their applicability and readiness for implementation in actual healthcare settings, ultimately benefiting humanity by delivering highly accurate and efficient instruments for early disease diagnosis and management.

10. REFRENCES:

[1] Z. Chen et al., "Diagnosis of patients with chronic kidney disease by using two fuzzy classifiers," Chemometr. Intell. Lab., vol. 153, pp. 140-145, Apr. 2016.

[2] A. Subasi, E. Alickovic, J. Kevric, "Diagnosis of chronic kidney disease by using random forest," in Proc. Int. Conf. Medical and Biological Engineering, Mar. 2017, pp. 589-594.

[3] L. Zhang et al., "Prevalence of chronic kidney disease in china: a cross-sectional survey," Lancet, vol. 379, pp. 815-822, Aug. 2012.

[4] A. Singh et al., "Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration," J. Biomed. Inform., vol. 53, pp. 220-228, Feb. 2015.

[5] A. M. Cueto-Manzano et al., "Prevalence of chronic kidney disease in an adult population," Arch. Med. Res., vol. 45, no. 6, pp. 507-513, Aug. 2014.

[6] H. Polat, H.D. Mehr, A. Cetin, "Diagnosis of chronic kidney disease based on support vector machine by feature selection methods," J. Med. Syst., vol. 41, no. 4, Apr. 2017.

[7] C. Barbieri et al., "A new machine learning approach for predicting the response to anemia treatment in a large cohort of end stage renal disease patients undergoing dialysis," Comput. Biol. Med., vol. 61, pp. 56-61, Jun. 2015.

[8] V. Papademetriou et al., "Chronic kidney disease, basal insulin glargine, and health outcomes in people with dysglycemia: The origin study," Am. J. Med., vol. 130, no. 12, Dec. 2017.

[9] N. R. Hill et al., "Global prevalence of chronic kidney disease – A systematic review and meta-analysis," Plos One, vol. 11, no. 7, Jul. 2016.

[10] M. M. Hossain et al., "Mechanical anisotropy assessment in kidney cortex using ARFI peak displacement: Preclinical validation and pilot in vivo clinical results in kidney allografts," IEEE Trans. Ultrason. Ferr., vol. 66, no. 3, pp. 551-562, Mar. 2019.

[11] M. Alloghani et al., "Applications of machine learning techniques for software engineering learning and early prediction of students' performance," in Proc. Int. Conf. Soft Computing in Data Science, Dec. 2018, pp. 246-258.

[12] D. Gupta, S. Khare, A. Aggarwal, "A method to predict diagnostic codes for chronic diseases using machine learning techniques," in Proc. Int. Conf. Computing, Communication and Automation, Apr. 2016, pp. 281-287.

[13] L. Du et al., "A machine learning based approach to identify protected health information in Chinese clinical text," Int. J. Med. Inform., vol. 116, pp. 24-32, Aug. 2018.

[14] R. Abbas et al., "Classification of foetal distress and hypoxia using machine learning approaches," in Proc. Int. Conf. Intelligent Computing, Jul. 2018, pp. 767- 776.

[15] M. Mahyoub, M. Randles, T. Baker and P. Yang, "Comparison analysis of machine learning algorithms to rank alzheimer's disease risk factors by im-portance," in Proc. 11th Int. Conf. Developments in eSystems Engineering,Sep. 2018.

[16] E. Alickovic, A. Subasi, "Medical decision support system for diagnosis of heart arrhythmia using DWT and random forests classifier," J. Med. Syst., vol. 40, no. 4, Apr. 2016.

[17] Z. Masetic, A. Subasi, "Congestive heart failure detection using random forest classifier," Comput. Meth. Prog. Bio., vol. 130, pp. 56-64, Jul. 2016

[18] Q. Zou et al., "Predicting diabetes mellitus with machine learning tech-niques," Front. Genet., vol. 9, Nov. 2018.