**Classification Assignment- Chronic Kidney Disease Prediction (CKD)**

**Problem Statement or Requirement:**

A requirement from the Hospital, Management asked us to create a predictive model which will predict the Chronic Kidney Disease (CKD) based on the several parameters. The Client has provided the dataset of the same.

**1.) Identify your problem statement**

**Stage1: Domain selection:**

Input is a csv data in table format. So, the domain is machine learning.

**Stage2: Learning selection:**

* Requirement is very clear. Prediction of CKD
* It has both input and output is present.

**Stage3: Regression/Classification selection:**

The problem statement tells to predict whether the person has the chronic kidney disease or not. If yes CKD, if not NoCKD,

Output is a categorical data. So, it is a **Classification**.

**2.) Tell basic info about the dataset (Total number of rows, columns)**

No of Rows: 399

No of Columns: 25

**3.) Mention the pre-processing method if you’re doing any (like converting string to number – nominal data)**

We are converting the categorical columns into numerical columns. since it is a categories values like yes /no, normal/abnormal, Present/not present. We choose one-hot encoding and dropping the first column to avoid redundancy of values. Also we are checking if there are any Nan values removing it. We are converting datatype into integers to convert the true / false into numbers 1/0.

**4.) Develop a good model with good evaluation metric. You can use any machine learning algorithm; you can create many models. Finally, you must come up with final model.**

Since it is classification problem we perform evaluation metrics as CONFUSION MATRIX. We predict all the evaluation metrics using classification report.

**SUPPORT VECTOR MACHINE- CLASSIFICATION (SVC)**

Roc auc score : 0.9997037037037036

{'C': 10, 'gamma': 'scale', 'kernel': 'sigmoid'}

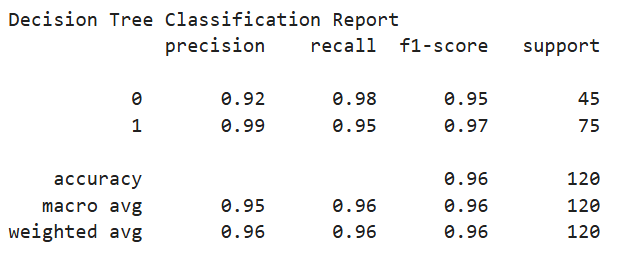
A screenshot of a graph

AI-generated content may be incorrect.

**DECISION TREE**

Roc auc socre; 0.9622222222222222

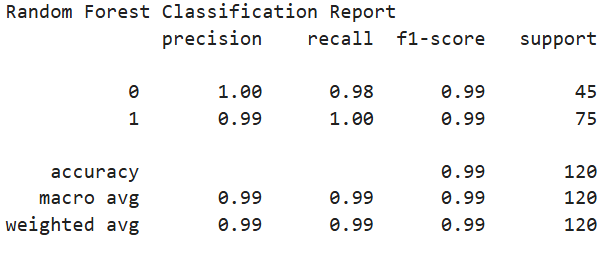
{'criterion': 'entropy', 'max\_features': 'sqrt', 'splitter': 'random'}



**RANDOM FOREST**

{'criterion': 'log\_loss', 'max\_features': 'log2', 'n\_estimators': 100}

Roc auc score 0.9997037037037038



**LOGISTIC REGRESSION**

Roc auc score 0.9976296296296296

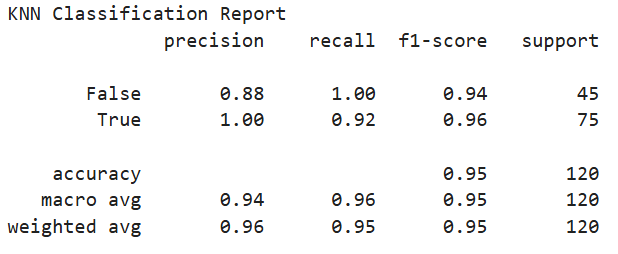
{'penalty': 'l2', 'solver': 'liblinear'}

A screenshot of a computer

AI-generated content may be incorrect.

**KNN**

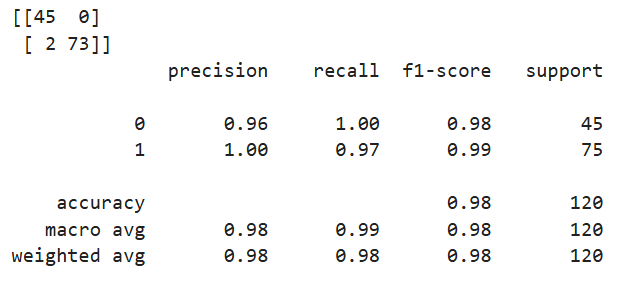
Roc auc score : 0.9995555555555555



**NAÏVE BAYES**

1. **Gaussian**

Roc auc score : 0.9997858099062918



1. **Multinomial**

Roc auc score : 0.934912985274431

A screenshot of a computer screen

AI-generated content may be incorrect.

1. **BernoulliNB**

Roc auc score: 0.994016064257028

A screenshot of a computer screen

AI-generated content may be incorrect.

1. **ComplementNB**

Roc auc score: 0.934912985274431

A screenshot of a computer screen

AI-generated content may be incorrect.

1. **CategoricalNB**

Roc auc score: 0.9999196787148594

A screenshot of a computer screen

AI-generated content may be incorrect.

5.) All the research values of each algorithm should be documented. (You can make tabulation or screenshot of the results.)

|  |  |  |  |
| --- | --- | --- | --- |
| S.no | Algorithm Name | Accuracy | ROC AUC score |
| 1 | SVM | 0.98 | 0.999 |
| 2 | Decision Tree | 0.96 | 0.962 |
| 3 | Random Forest | 0.99 | 0.999 |
| 4 | Logistic Regression | 0.98 | 0.997 |
| 5 | K-Nearest Neighbour | 0.95 | 0.999 |
| 6 | Naïve Bayes  GaussianNB  BernoulliNB  MultinomialNB  ComplementNB  CategoricalNB | 0.98  0.93  0.81  0.81  1.00 | 0.999  0.994  0.934  0.934  0.999 |

6.) Mention your final model, justify why u have chosen the same.

The final model is Categorical Naïve bayes which is 100% accuracy and roc score also 99% and second importance to Random Forest the accuracy and roc score is 99 %