Classification Assignment

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**

Machine learning---->Supervised learning---->Classification

- Domain-dataset inputs are numbers, table format in excel-sheet(Machine Learning)
- Requirement is very clear & input and output are present in the dataset so, it comes under (Supervised learning)
- Output columns are Categorical Value, so it comes under(Classification)

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

Total number or Rows-399

Columns-25

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

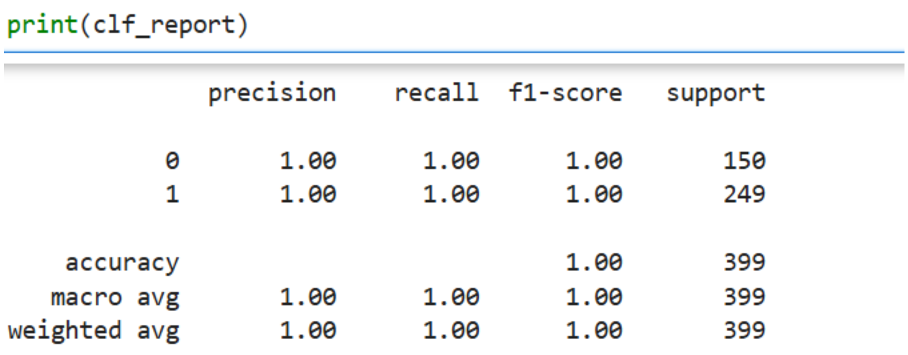
Categorial column convert into Nominal data using--->0ne Hot Encoding

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

Final model-"LOGISTIC-Regression-Algorithm

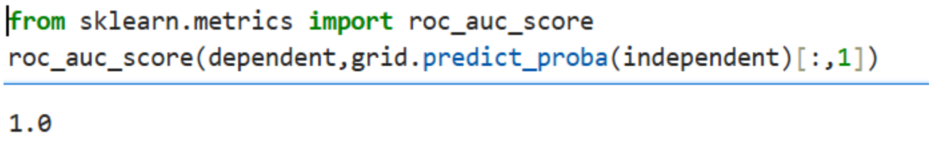
5. All the research values of each algorithm should be documented. (You can make tabulation or screenshot of the results.)

1.Decision tree-classification report



```
print(clf_report)
```

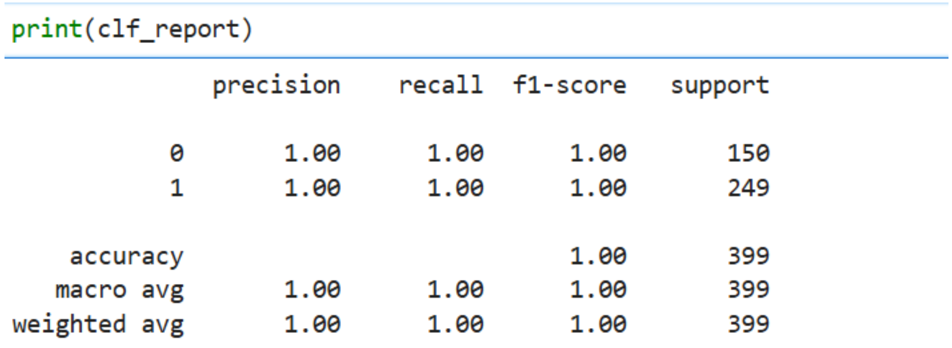
```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       150
           1       1.00      1.00      1.00       249

    accuracy                           1.00       399
   macro avg       1.00      1.00      1.00       399
weighted avg       1.00      1.00      1.00       399
```



```
from sklearn.metrics import roc_auc_score
roc_auc_score(dependent,grid.predict_proba(independent)[:,1])
```

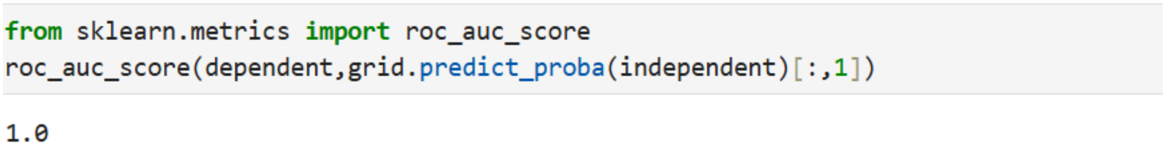
```
1.0
```

2. Support Vector Machine(SVM)-classification report



```
print(clf_report)
```

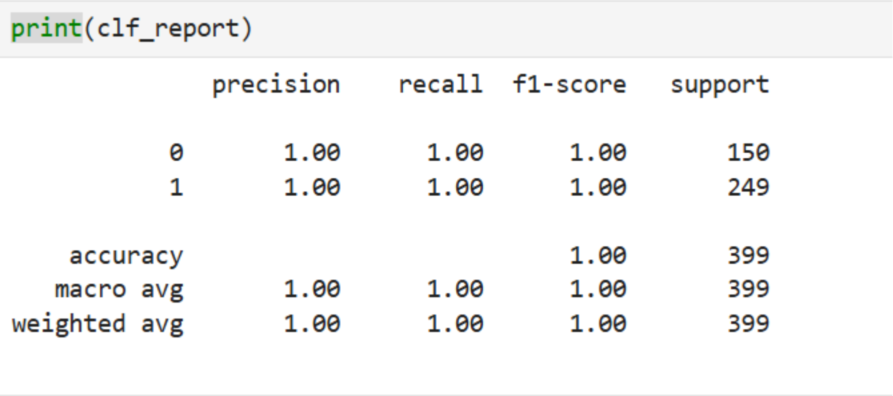
```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       150
           1       1.00      1.00      1.00       249

    accuracy                           1.00       399
   macro avg       1.00      1.00      1.00       399
weighted avg       1.00      1.00      1.00       399
```



```
from sklearn.metrics import roc_auc_score
roc_auc_score(dependent,grid.predict_proba(independent)[:,1])
```

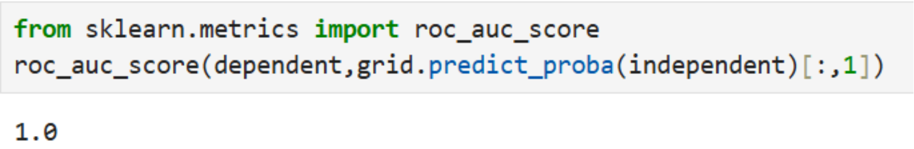
```
1.0
```

3. Random Forest



```
print(clf_report)
```

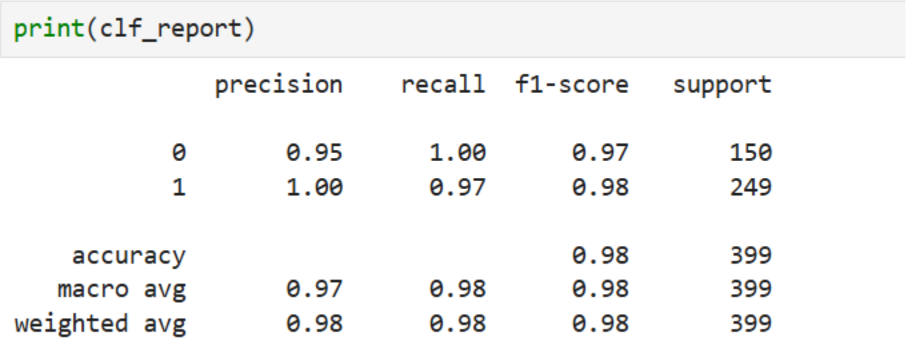
```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       150
           1       1.00      1.00      1.00       249

    accuracy                           1.00       399
   macro avg       1.00      1.00      1.00       399
weighted avg       1.00      1.00      1.00       399
```



```
from sklearn.metrics import roc_auc_score
roc_auc_score(dependent,grid.predict_proba(independent)[:,1])
```

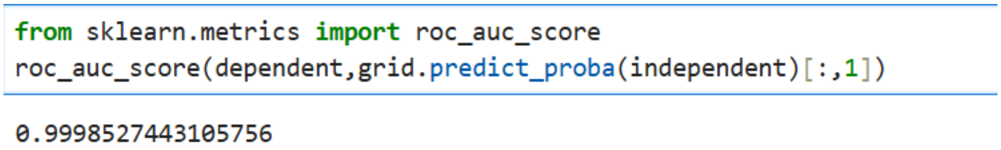
```
1.0
```

4. KNN-k Nearest Neighbor



```
print(clf_report)
```

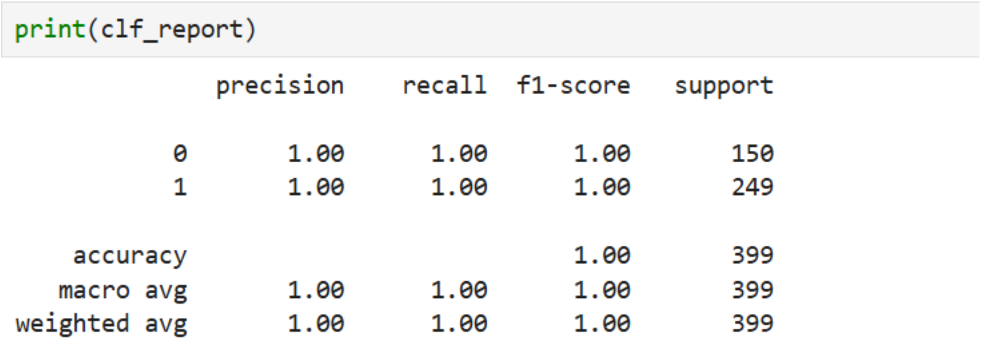
```
              precision    recall  f1-score   support

           0       0.95      1.00      0.97       150
           1       1.00      0.97      0.98       249

    accuracy                           0.98       399
   macro avg       0.97      0.98      0.98       399
weighted avg       0.98      0.98      0.98       399
```



```
from sklearn.metrics import roc_auc_score
roc_auc_score(dependent,grid.predict_proba(independent)[:,1])
```

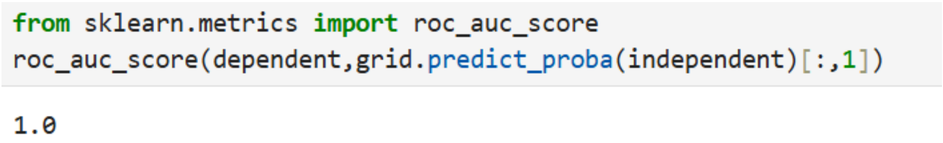
```
0.9998527443105756
```

5. Logistic Regression



```
print(clf_report)
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       150
           1       1.00      1.00      1.00       249

    accuracy                           1.00       399
   macro avg       1.00      1.00      1.00       399
weighted avg       1.00      1.00      1.00       399
```



```
from sklearn.metrics import roc_auc_score
roc_auc_score(dependent,grid.predict_proba(independent)[:,1])
```

```
1.0
```

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

The above research, all the models performed well, and they learned effectively,(Accuracy=1.0 & roc_score=1.0) for the final model, I chose the logistic Regression model.