

## CLASSIFICATION ASSIGNMENT:

DATASET TAKEN IS CKD:

### STEPS TO SOLVE THE ASSIGNMENT:

#### 1) PROBLEM STATEMENT:

DOMAIN : MACHINE LEARNING – UN SUPERVISED LEARNING – CLASSIFICATION.

#### 2) BASIC INFO ABOUT DATASET:

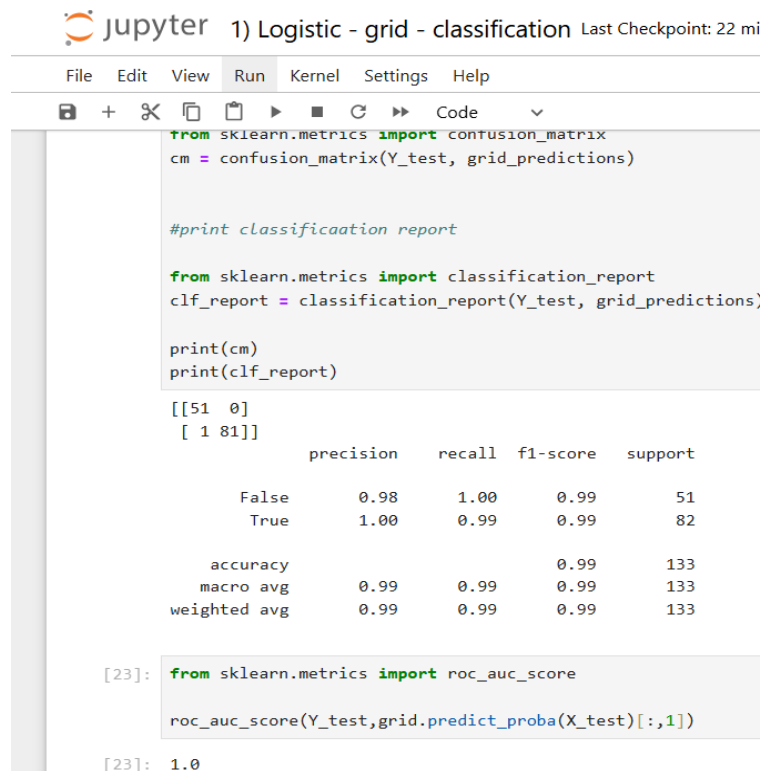
\*THERE IS 399 ROWS AND 25 COLUMNS.

\*THIS DATASET IS BASICALLY ALL ABOUT CHRONIC KIDNEY DISEASE PATIENT DETAILS WITH AGE AND ALL THINGS.

#### 3) PREPROCESSING THINGS IS DONE IN THIS DATASET LIKE **GET\_DIMMIES** WHICH IS CONVERTING THE NUMERICAL VALUES INTO CATEGORICAL VALUES AND STANDARD SCALER WHICH IS ACTUALLY CONVERTING THE NUMERICAL VALUES INTO PARTICULAR RANGE LIKE -3 TO +3 VALUES.

#### 4) I HAVE DEVELOPED 6 MODELS WITH HANDS ON AND ALSO USED DIFFERENT KIND OF PARAMETERS.

#### 5) HERE I HAVE MENTIONED BELOW THE ALGORITHMS WITH IMAGE FORMAT:



```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(Y_test, grid_predictions)

# print classification report

from sklearn.metrics import classification_report
clf_report = classification_report(Y_test, grid_predictions)

print(cm)
print(clf_report)
```

```
[[51  0]
 [ 1 81]]
```

	precision	recall	f1-score	support
False	0.98	1.00	0.99	51
True	1.00	0.99	0.99	82
accuracy			0.99	133
macro avg	0.99	0.99	0.99	133
weighted avg	0.99	0.99	0.99	133

```
[23]: from sklearn.metrics import roc_auc_score

roc_auc_score(Y_test, grid.predict_proba(X_test)[:,:1])
```

```
[23]: 1.0
```

### jupyter 3) DT - GRID - CLASSIFICATION Last Checkpoint:

```
File Edit View Run Kernel Settings Help
```

```

clf_report = classification_report(Y_test, grid_predict
print(cm)
print(clf_report)

[[49  2]
 [ 6 76]]

```

	precision	recall	f1-score	support
False	0.89	0.96	0.92	51
True	0.97	0.93	0.95	82
accuracy			0.94	133
macro avg	0.93	0.94	0.94	133
weighted avg	0.94	0.94	0.94	133

```

[17]: from sklearn.metrics import roc_auc_score

roc_auc_score(Y_test,grid.predict_proba(X_test)[:,:1])

[17]: 0.9438067910090866

```

### jupyter 2)SVM - GRID - CLASSIFICATION Last Checkpoint:

```
File Edit View Run Kernel Settings Help
```

```

clf_report = classification_report(Y_test, grid_predict
print(cm)
print(clf_report)

[[51  0]
 [ 1 81]]

```

	precision	recall	f1-score	support
False	0.98	1.00	0.99	51
True	1.00	0.99	0.99	82
accuracy			0.99	133
macro avg	0.99	0.99	0.99	133
weighted avg	0.99	0.99	0.99	133

```

[22]: from sklearn.metrics import roc_auc_score

roc_auc_score(Y_test,grid.predict_proba(X_test)[:,:1])

[22]: 1.0

```

```
print(cm)
print(clf_report)
```

```
[[51  0]
 [ 1 81]]
```

	precision	recall	f1-score	support
False	0.98	1.00	0.99	51
True	1.00	0.99	0.99	82
accuracy			0.99	133
macro avg	0.99	0.99	0.99	133
weighted avg	0.99	0.99	0.99	133

```
[17]: from sklearn.metrics import roc_auc_score

roc_auc_score(Y_test,grid.predict_proba(X_test)[:,:1])
```

[17]: 1.0

```
print(cm)
print(clf_report)
```

```
[[51  0]
 [ 5 77]]
```

	precision	recall	f1-score	support
False	0.91	1.00	0.95	51
True	1.00	0.94	0.97	82
accuracy			0.96	133
macro avg	0.96	0.97	0.96	133
weighted avg	0.97	0.96	0.96	133

```
[17]: from sklearn.metrics import roc_auc_score

roc_auc_score(Y_test,grid.predict_proba(X_test)[:,:1])
```

[17]: 0.9995217599234816

6) I HAVE TAKEN 3 MODELS AS BEST MODEL FOR THIS DATASET  
THERE ARE:

- 1) LOGISTIC CLASS GRID
- 2) SVM GRID

### 3) RANDOM FOREST CLASSIFICATION GRID