Tool and Techniques Name

1) Python Libraries

- Numpy
- Pandas
- Matplotlib
- Seaborn
- Sklearn.metrics
- Sklearn.preprocessing

2) Dataset Link

- https://www.kaggle.com/datasets/bidemiayinde/thyroid-sickness-determination

3) Preprocessing Techniques

- Label encoded the categorical values
- Replaced the question marks with np.nan values.
- Filled these nan values with mean of the particular column.
- Applied Random Undersampler for undersampling.

4) Feature Extraction/ Selection Technique

- Applied CMIM technique for Feature Selection.
- Since, all these features were not present in our base paper that's why I have dropped them and left with features like sex, goitre, tumor, pregnant, etc.

6) Classification Technique

- We have 2 classes in our base paper which are whether the person is suffering from the thyroid disease or not.
- Classes are Yes or No

7) Data Split Ratio

- We splitted the data into training and testing in the ratio of 70:30.

8) Base Model

- K Neighbors Classifier
- Random Forest Classifier
- Ann Classifier

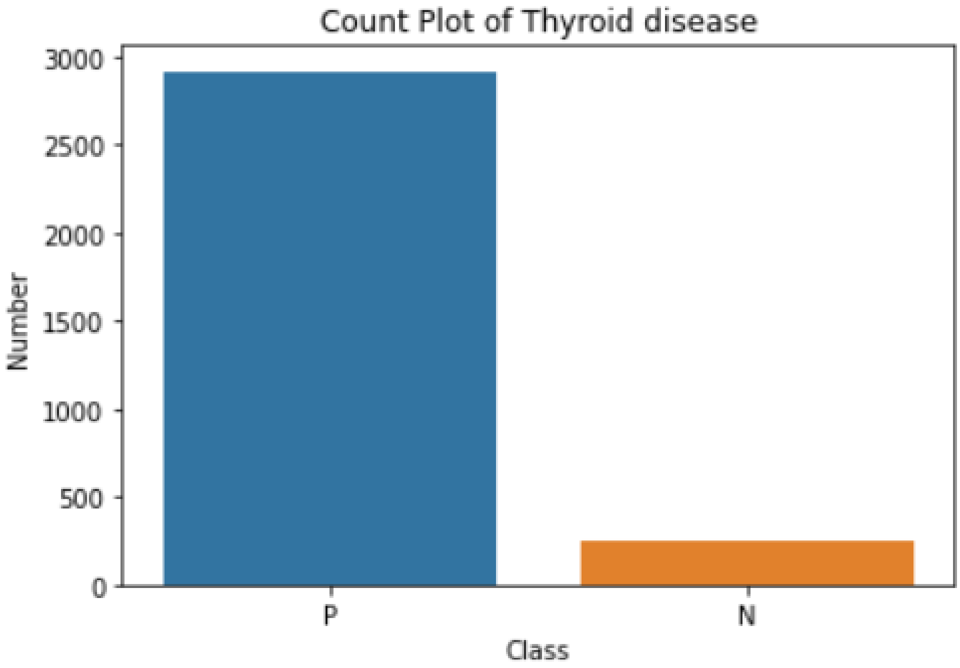
if any another tool and technique used so pls include it and remove it to above

points

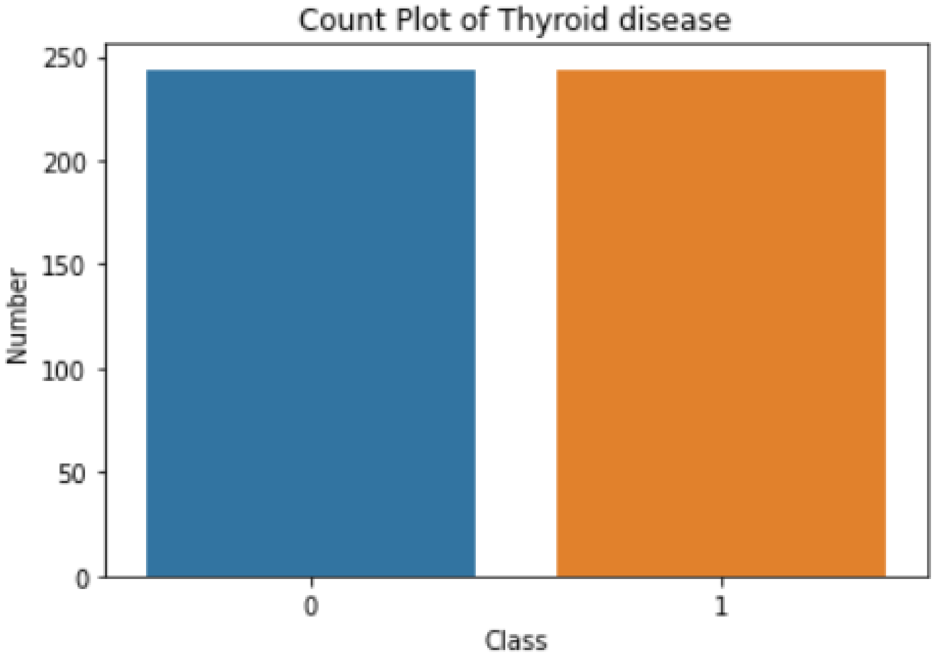
Screenshots of base and proposed results

1) Dataset visualization screenshots

- Count plot of thyroid disease before undersampler

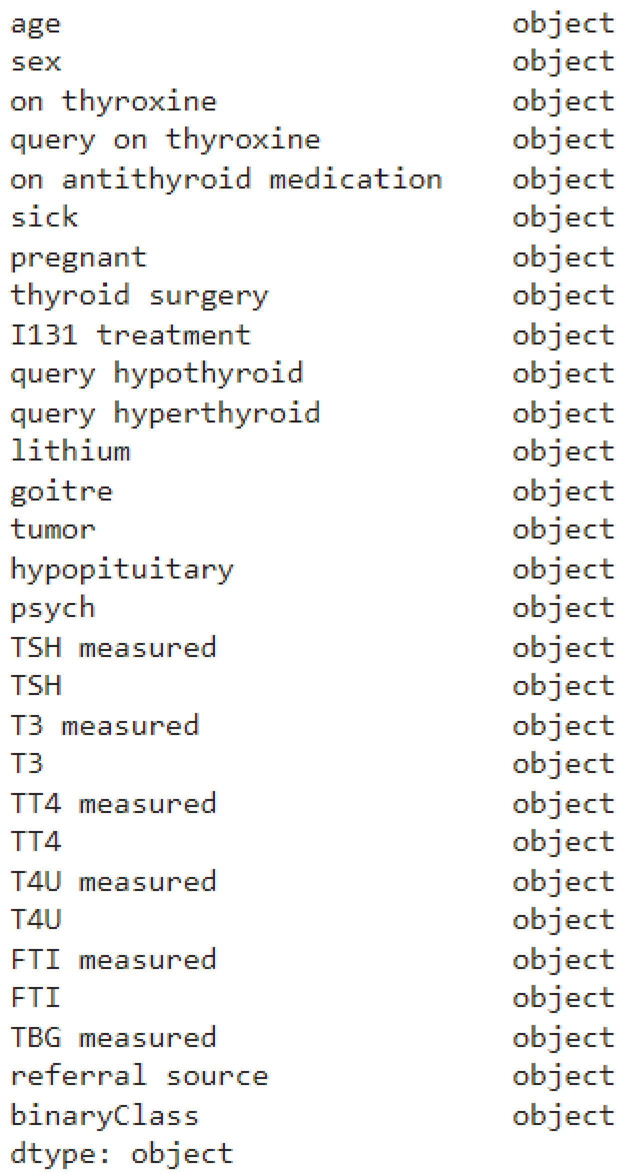


- Count plot after Undersampler



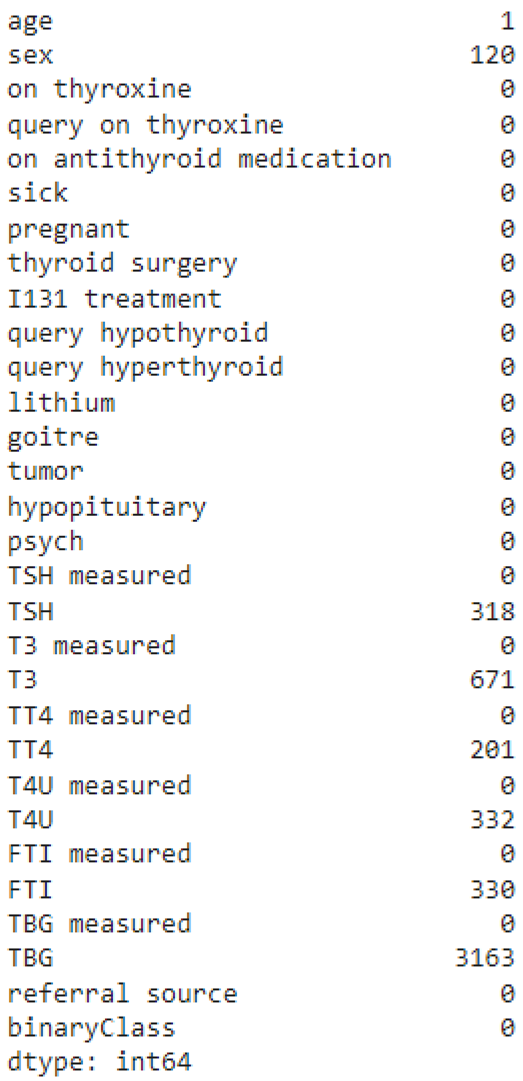
2) Preprocessing results

- Datatypes of features



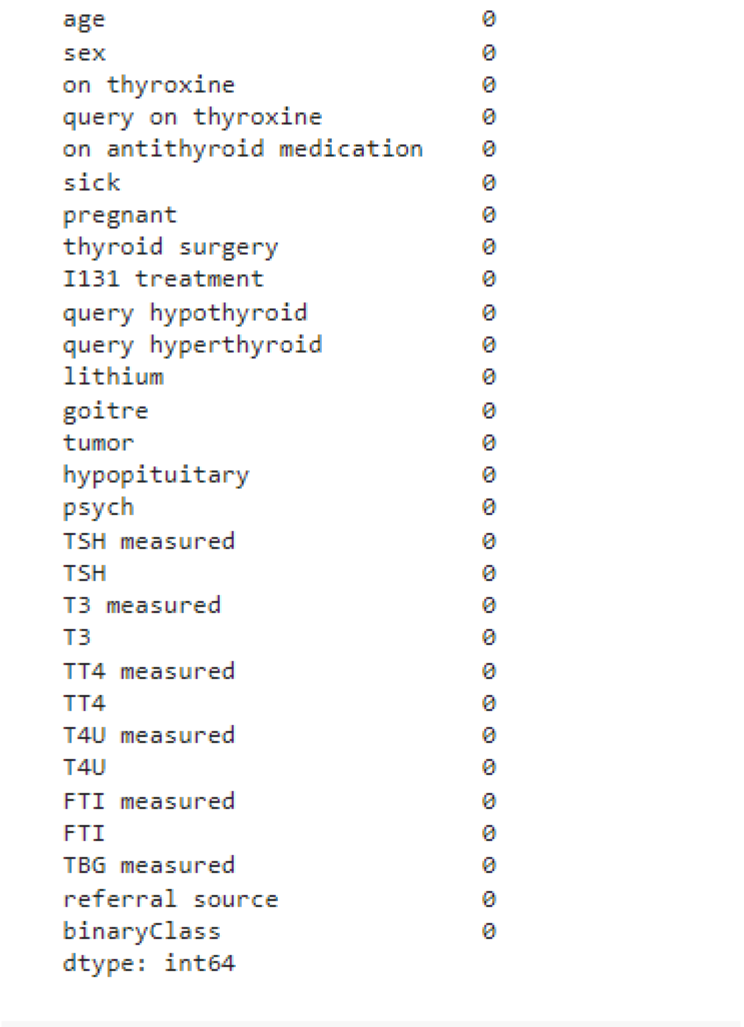
```
age                            object
sex                            object
on thyroxine                   object
query on thyroxine             object
on antithyroid medication      object
sick                           object
pregnant                       object
thyroid surgery                object
I131 treatment                 object
query hypothyroid              object
query hyperthyroid             object
lithium                        object
goitre                         object
tumor                          object
hypopituitary                  object
psych                          object
TSH measured                   object
TSH                            object
T3 measured                    object
T3                             object
TT4 measured                   object
TT4                            object
T4U measured                   object
T4U                            object
FTI measured                   object
FTI                            object
TBG measured                   object
referral source                object
binaryClass                    object
dtype: object
```

- Checking the null values

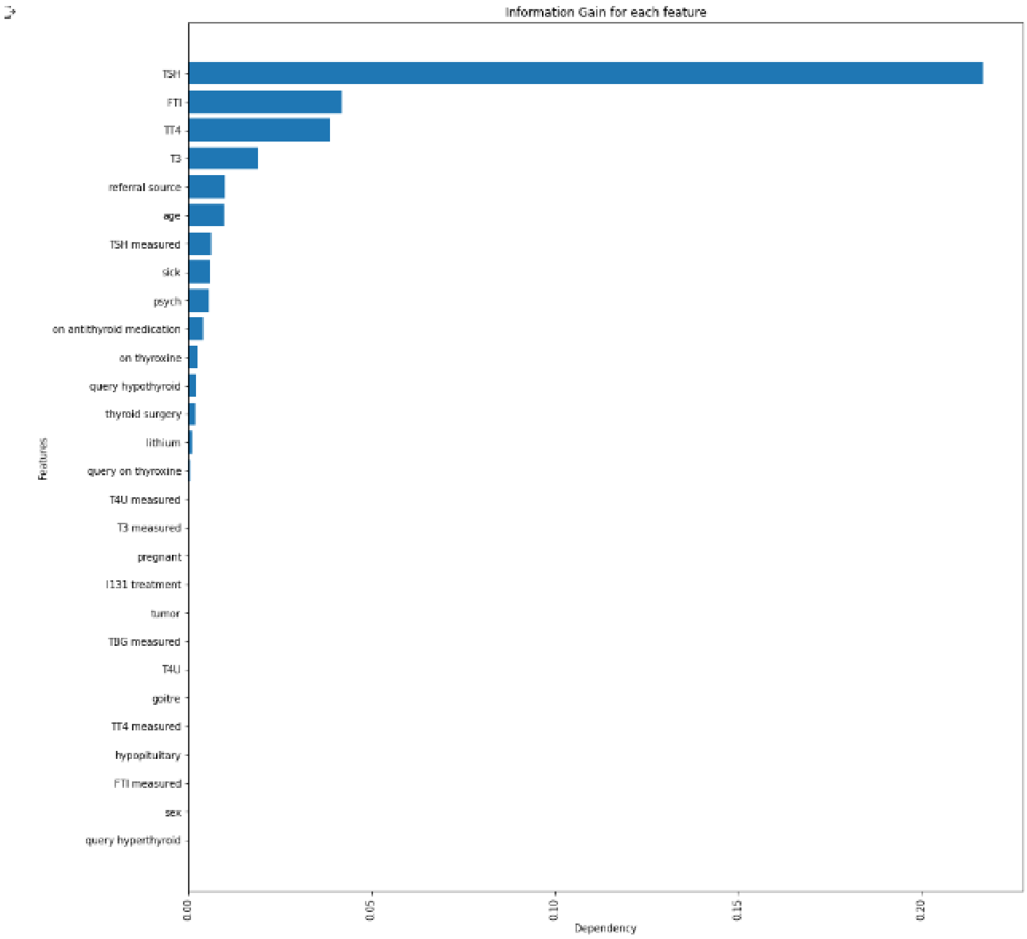


```
age                            1
sex                          120
on thyroxine                   0
query on thyroxine             0
on antithyroid medication      0
sick                           0
pregnant                       0
thyroid surgery                0
I131 treatment                 0
query hypothyroid              0
query hyperthyroid             0
lithium                        0
goitre                         0
tumor                          0
hypopituitary                  0
psych                          0
TSH measured                   0
TSH                          318
T3 measured                    0
T3                           671
TT4 measured                   0
TT4                          201
T4U measured                   0
T4U                          332
FTI measured                   0
FTI                          330
TBG measured                   0
TBG                         3163
referral source                0
binaryClass                    0
dtype: int64
```

- **After Removing the null values from the dataset**

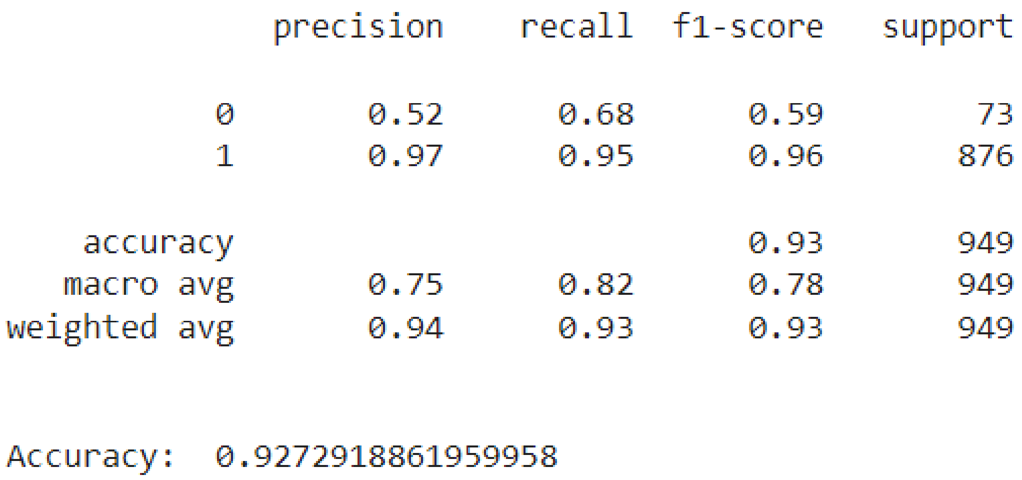


```
age                           0
sex                           0
on thyroxine                  0
query on thyroxine            0
on antithyroid medication     0
sick                          0
pregnant                      0
thyroid surgery               0
I131 treatment                0
query hypothyroid             0
query hyperthyroid            0
lithium                       0
goitre                        0
tumor                         0
hypopituitary                 0
psych                         0
TSH measured                  0
TSH                           0
T3 measured                   0
T3                            0
TT4 measured                  0
TT4                           0
T4U measured                  0
T4U                           0
FTI measured                  0
FTI                           0
TBG measured                  0
referral source               0
binaryClass                   0
dtype: int64
```



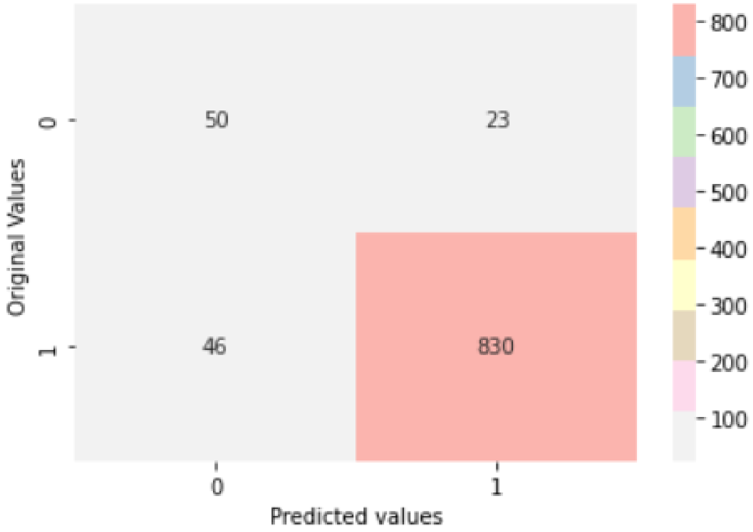
5) Base model results---like confusion matrix, ROC curve, classification report etc.

KNN (K = 2)



```
              precision    recall  f1-score   support

           0       0.52      0.68      0.59        73
           1       0.97      0.95      0.96       876

    accuracy                           0.93       949
   macro avg       0.75      0.82      0.78       949
weighted avg       0.94      0.93      0.93       949


Accuracy:  0.9272918861959958
```

```
Text(33.0, 0.5, 'Original Values')
```



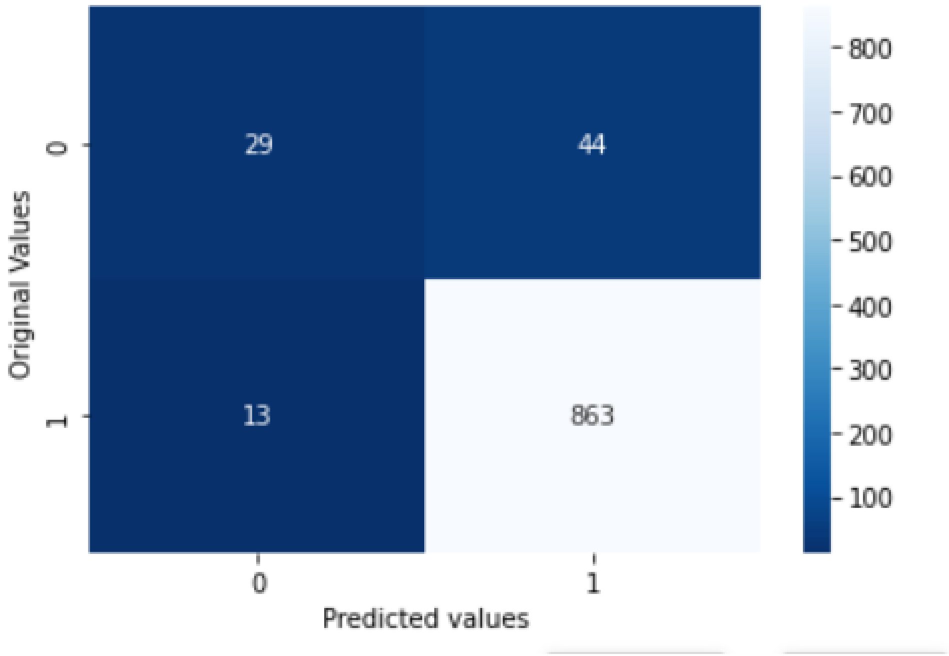
```
Sensitivity :  0.9474885844748858
Specificity :  0.684931506849315
```

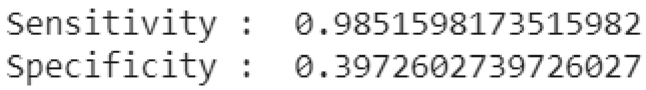
KNN (K = 10)

```
              precision    recall  f1-score   support

           0       0.69      0.40      0.50        73
           1       0.95      0.99      0.97       876

    accuracy                           0.94       949
   macro avg       0.82      0.69      0.74       949
weighted avg       0.93      0.94      0.93       949
```

```
Accuracy:  0.9399367755532139
```

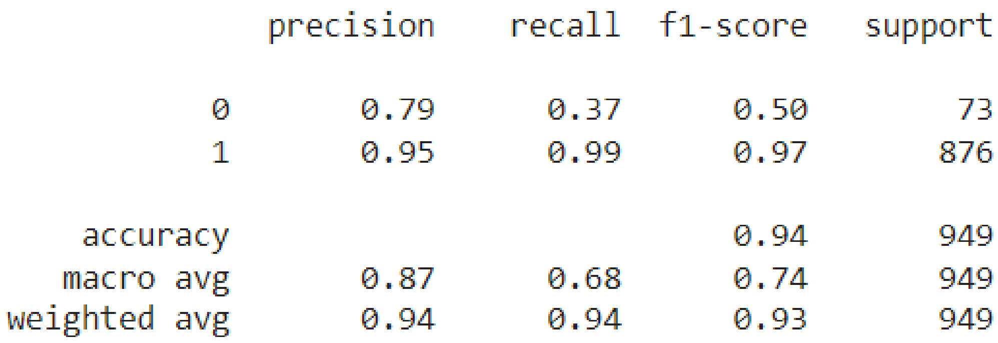
```
Text(33.0, 0.5, 'Original Values')
```



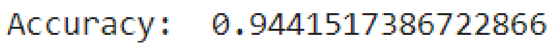


```
Sensitivity :  0.9851598173515982
Specificity :  0.3972602739726027
```

KNN (K = 20)

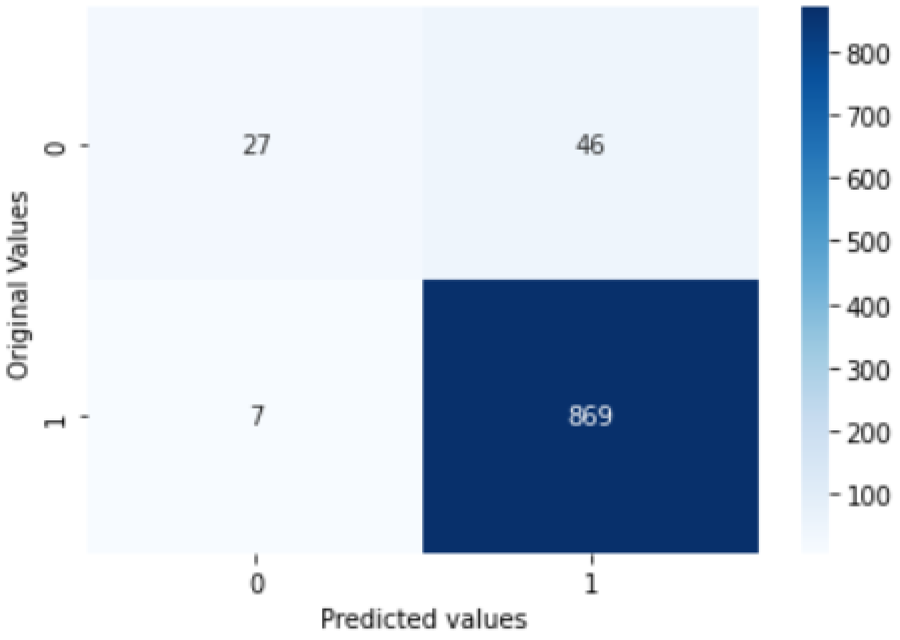


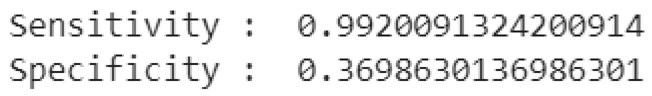
```
              precision    recall  f1-score   support

           0       0.79      0.37      0.50        73
           1       0.95      0.99      0.97       876

    accuracy                           0.94       949
   macro avg       0.87      0.68      0.74       949
weighted avg       0.94      0.94      0.93       949
```



```
Accuracy:  0.9441517386722866
```

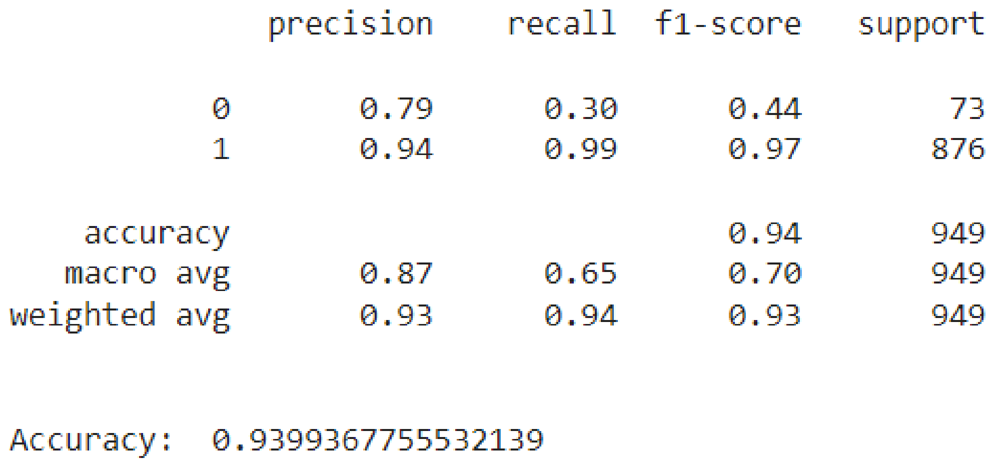
```
Text(33.0, 0.5, 'Original Values')
```



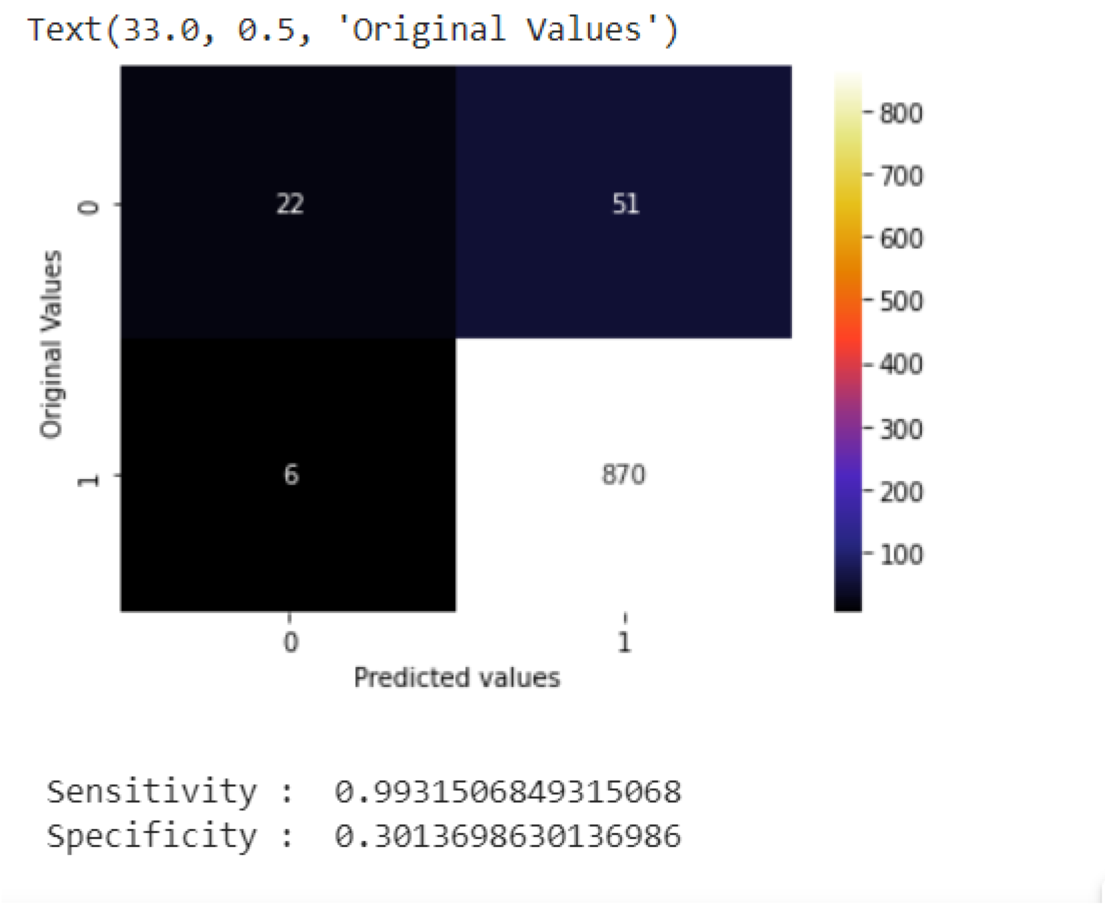


```
Sensitivity :  0.9920091324200914
Specificity :  0.3698630136986301
```

KNN (K = 25)



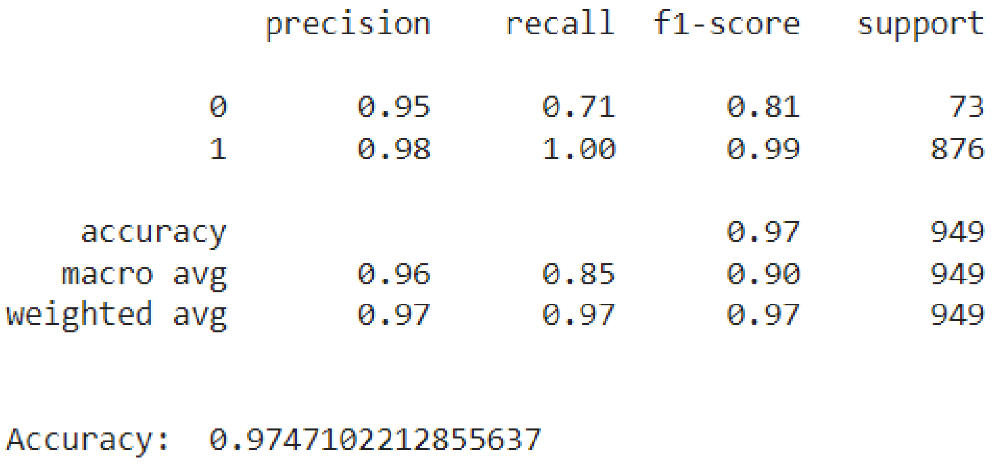
```
              precision    recall  f1-score   support

           0       0.79      0.30      0.44        73
           1       0.94      0.99      0.97       876

    accuracy                           0.94       949
   macro avg       0.87      0.65      0.70       949
weighted avg       0.93      0.94      0.93       949


Accuracy:  0.9399367755532139
```



```
Text(33.0, 0.5, 'Original Values')
```

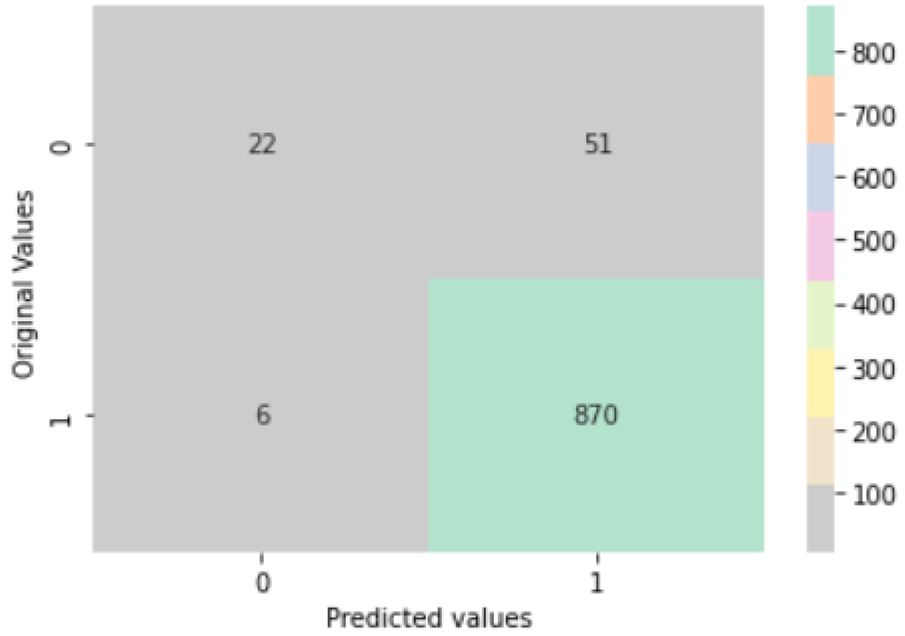
```
Sensitivity :  0.9931506849315068
Specificity :  0.3013698630136986
```

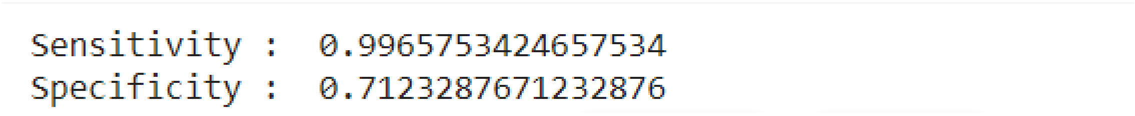
Random Forest Classifier



```
              precision    recall  f1-score   support

           0       0.95      0.71      0.81        73
           1       0.98      1.00      0.99       876

    accuracy                           0.97       949
   macro avg       0.96      0.85      0.90       949
weighted avg       0.97      0.97      0.97       949


Accuracy:  0.9747102212855637
```

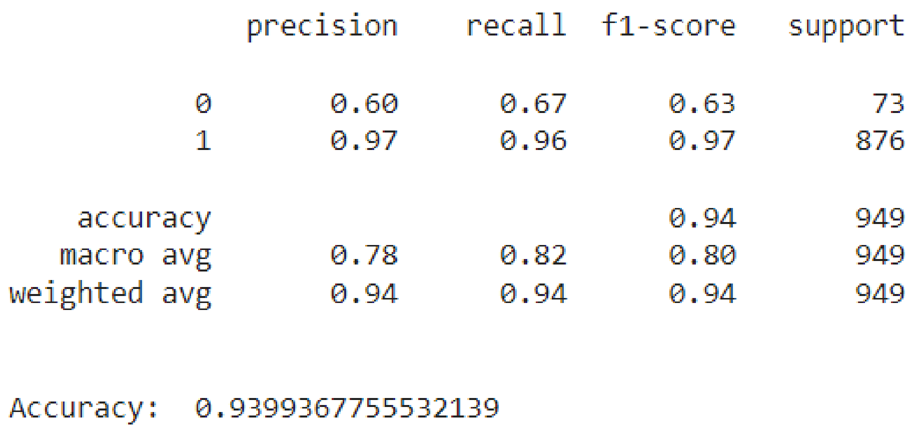
```
Text(33.0, 0.5, 'Original Values')
```



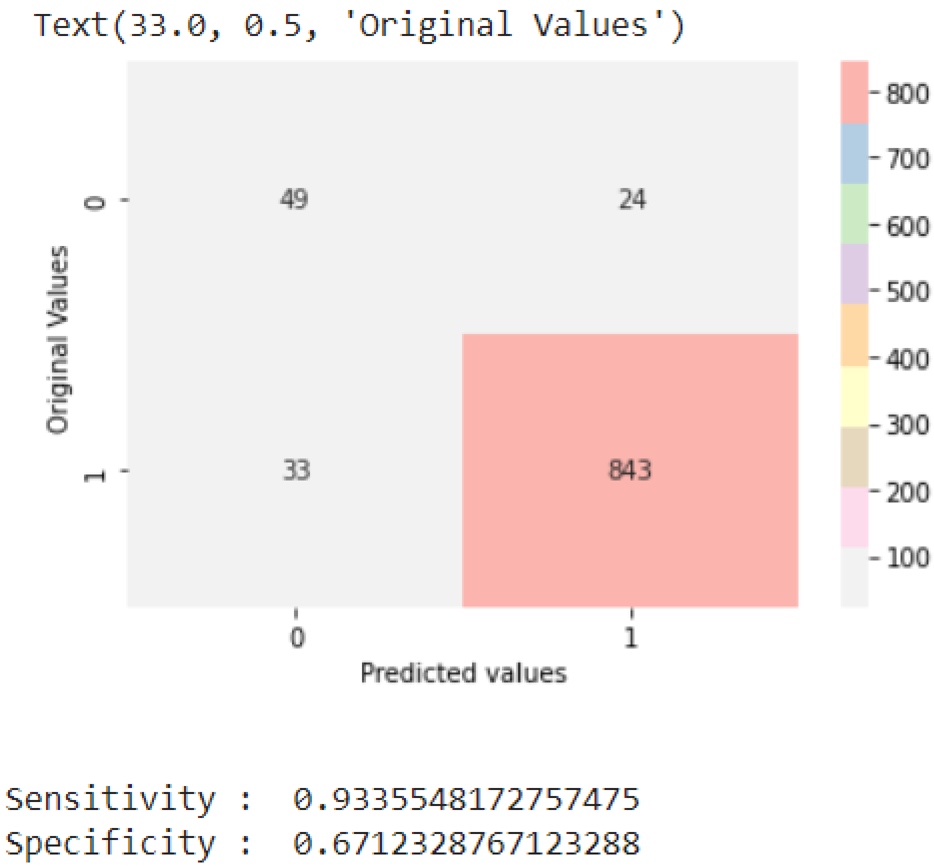


```
Sensitivity :  0.9965753424657534
Specificity :  0.7123287671232876
```

ANN Classifier



```
              precision    recall  f1-score   support

           0       0.60      0.67      0.63        73
           1       0.97      0.96      0.97       876

    accuracy                           0.94       949
   macro avg       0.78      0.82      0.80       949
weighted avg       0.94      0.94      0.94       949


Accuracy:  0.9399367755532139
```

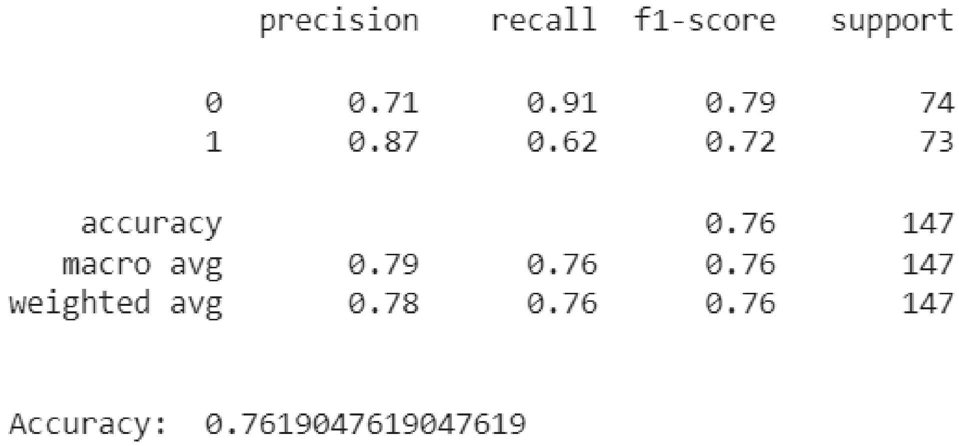


```
Text(33.0, 0.5, 'Original Values')
```

```
Sensitivity :  0.9335548172757475
Specificity :  0.6712328767123288
```

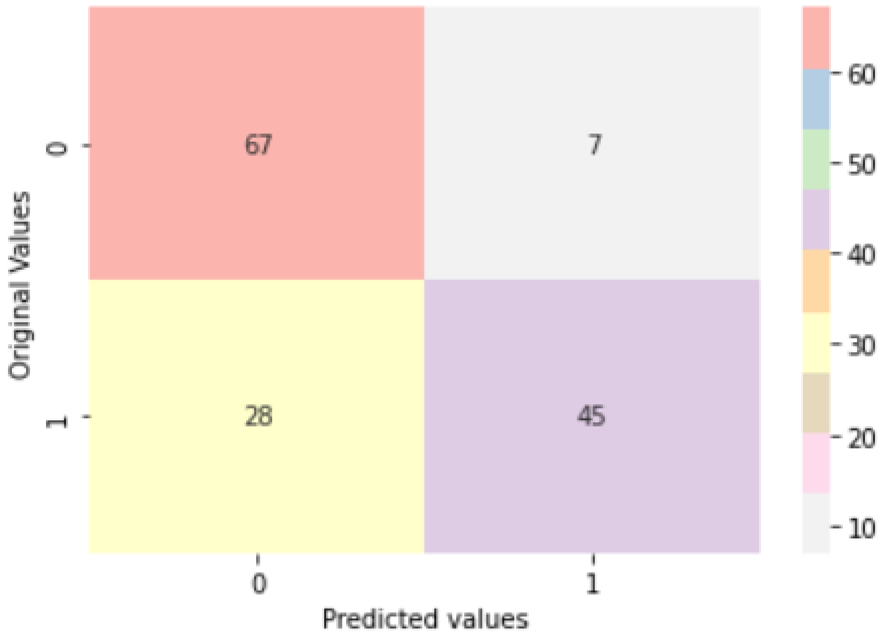
After UnderSampling

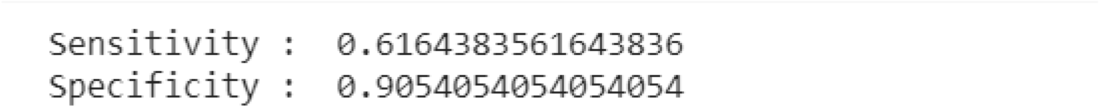
KNN (K = 2)



```
              precision    recall  f1-score   support

           0       0.71      0.91      0.79        74
           1       0.87      0.62      0.72        73

    accuracy                           0.76       147
   macro avg       0.79      0.76      0.76       147
weighted avg       0.78      0.76      0.76       147


Accuracy:  0.7619047619047619
```

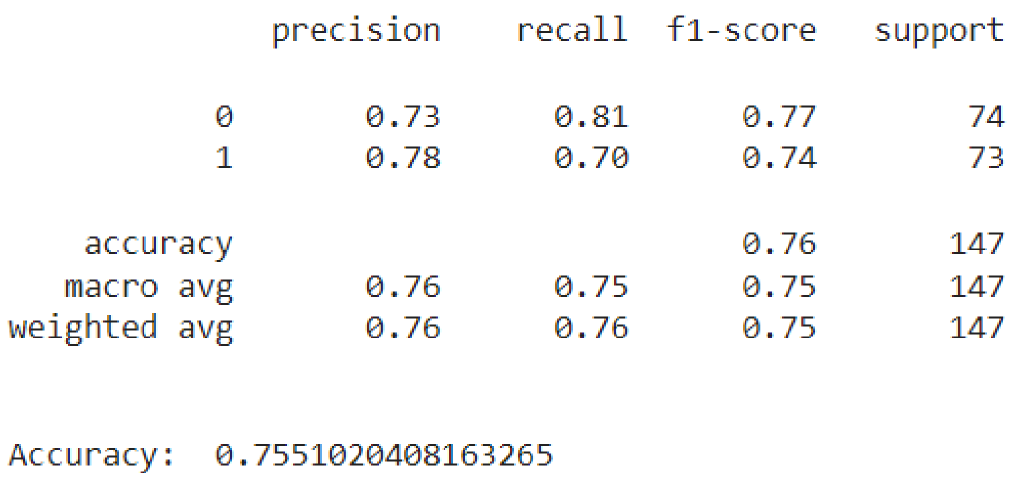
```
Text(33.0, 0.5, 'Original Values')
```





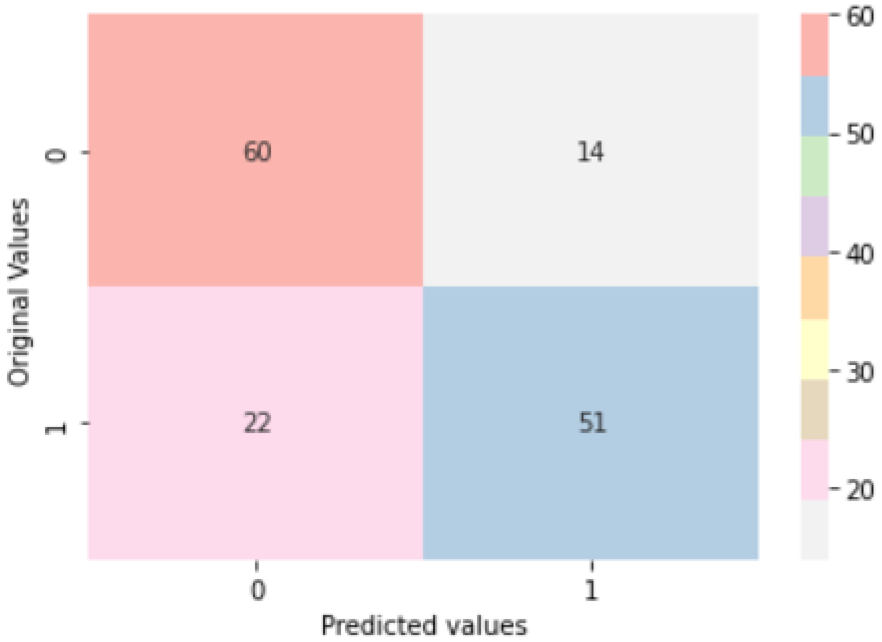
```
Sensitivity :  0.6164383561643836
Specificity :  0.9054054054054054
```

KNN (K = 10)



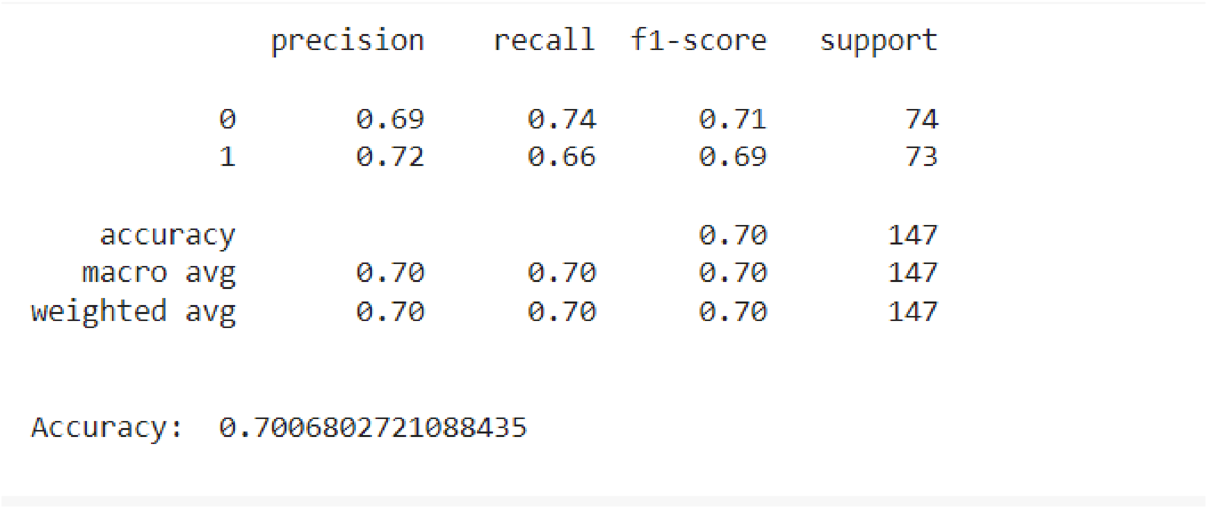
```
              precision    recall  f1-score   support

           0       0.73      0.81      0.77        74
           1       0.78      0.70      0.74        73

    accuracy                           0.76       147
   macro avg       0.76      0.75      0.75       147
weighted avg       0.76      0.76      0.75       147


Accuracy:  0.7551020408163265
```

```
Text(33.0, 0.5, 'Original Values')
```

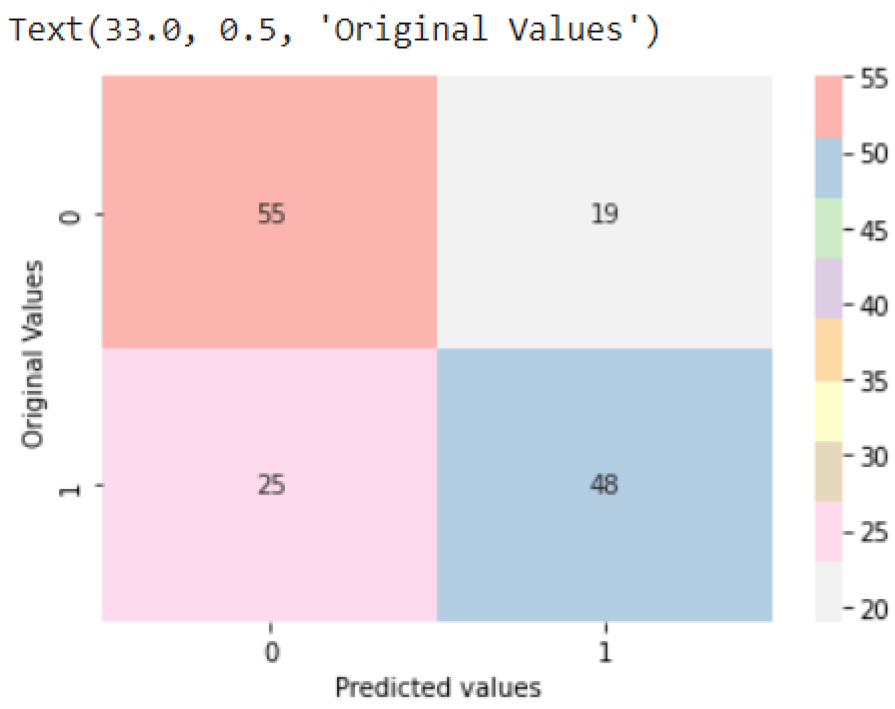


```
Sensitivity :  0.6986301369863014
Specificity :  0.8108108108108109
```

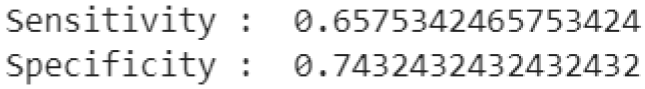
KNN (K = 20)



```
              precision    recall  f1-score   support

           0       0.69      0.74      0.71        74
           1       0.72      0.66      0.69        73

    accuracy                           0.70       147
   macro avg       0.70      0.70      0.70       147
weighted avg       0.70      0.70      0.70       147


Accuracy:  0.7006802721088435
```

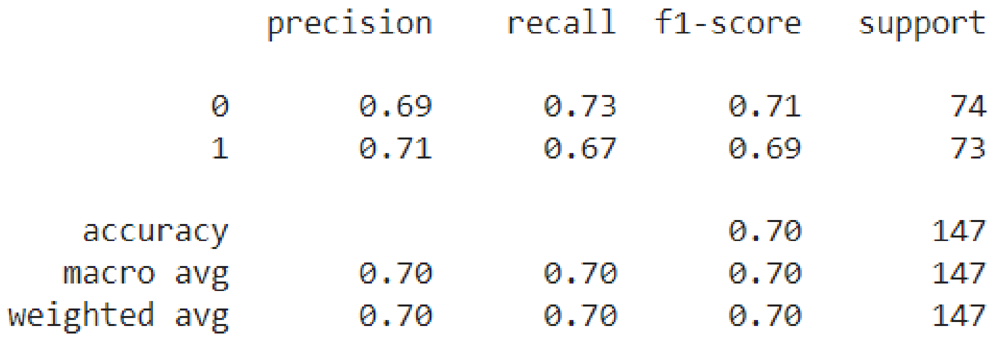


```
Text(33.0, 0.5, 'Original Values')
```

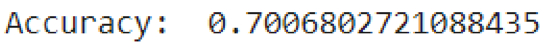


```
Sensitivity :  0.6575342465753424
Specificity :  0.7432432432432432
```

KNN (K = 25)

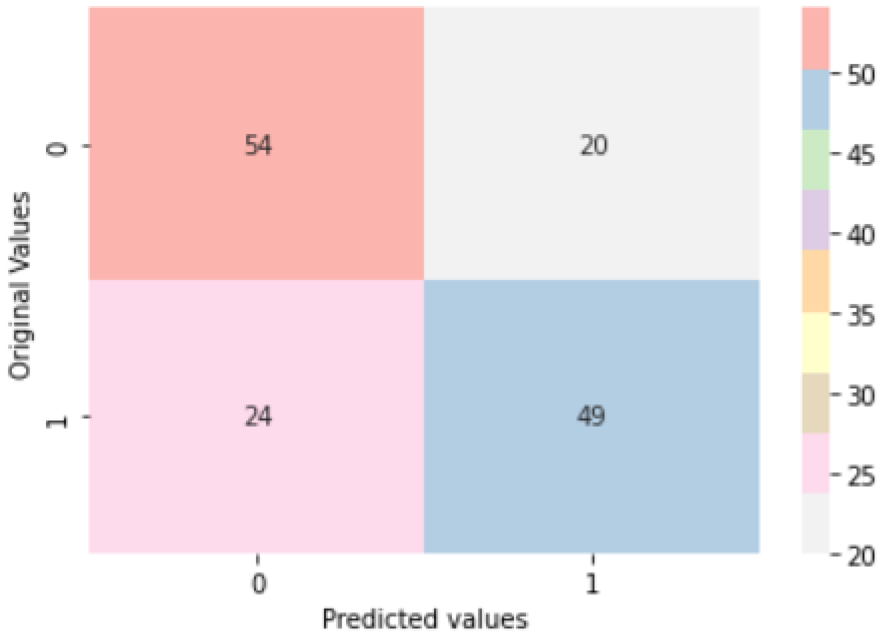


```
              precision    recall  f1-score   support

           0       0.69      0.73      0.71        74
           1       0.71      0.67      0.69        73

    accuracy                           0.70       147
   macro avg       0.70      0.70      0.70       147
weighted avg       0.70      0.70      0.70       147
```



```
Accuracy:  0.7006802721088435
```

```
Text(33.0, 0.5, 'Original Values')
```



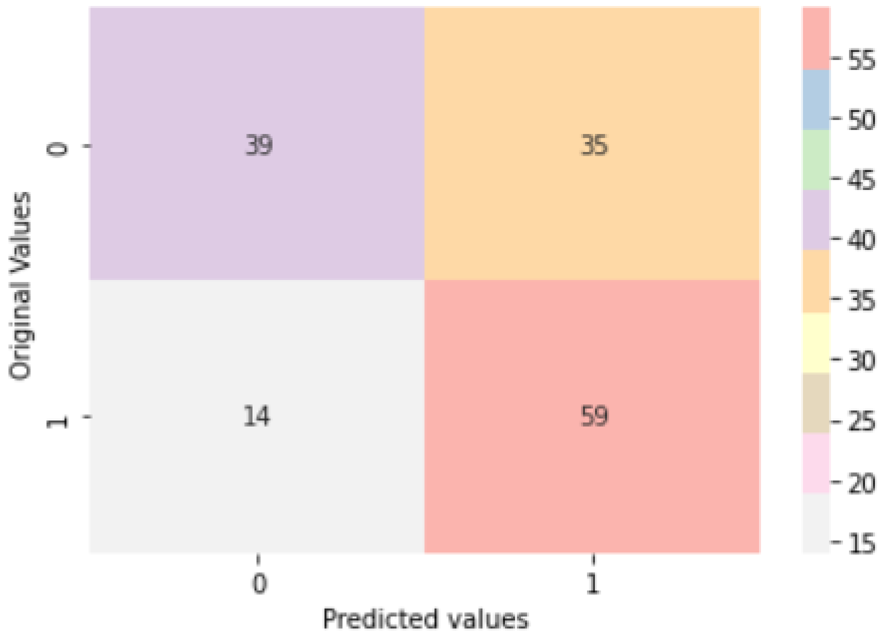
```
Sensitivity :  0.6712328767123288
Specificity :  0.7297297297297297
```

ANN Classifier

```
              precision    recall  f1-score   support

           0       0.74      0.53      0.61        74
           1       0.63      0.81      0.71        73

    accuracy                           0.67       147
   macro avg       0.68      0.67      0.66       147
weighted avg       0.68      0.67      0.66       147
```

```
Accuracy:  0.6666666666666666
```

```
Text(33.0, 0.5, 'Original Values')
```



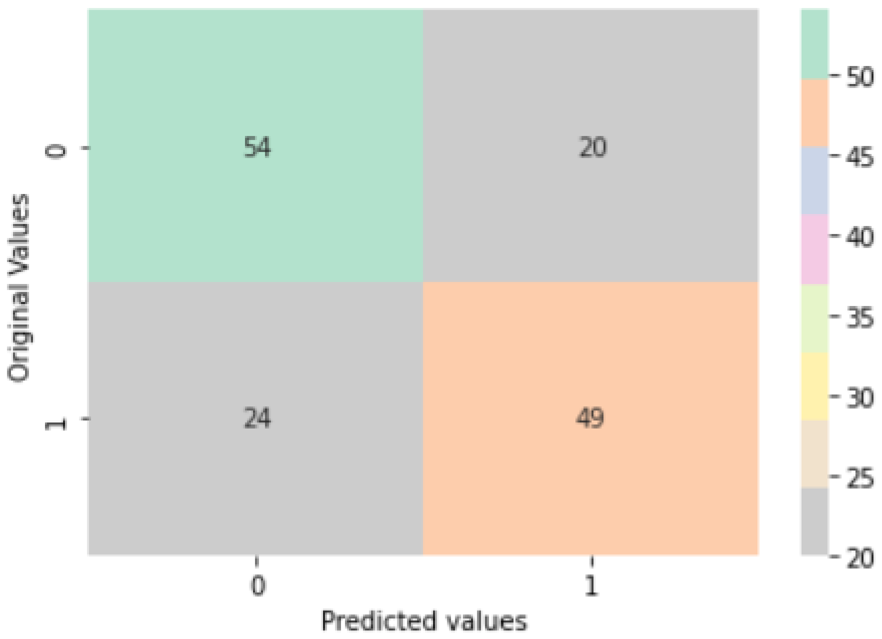
```
Sensitivity :  0.9365079365079365
Specificity :  0.527027027027027
```

Random Forest Classifier

```
              precision    recall  f1-score   support

           0       0.83      0.95      0.89        74
           1       0.94      0.81      0.87        73

    accuracy                           0.88       147
   macro avg       0.88      0.88      0.88       147
weighted avg       0.88      0.88      0.88       147
```

```
Accuracy:  0.8775510204081632
```

```
Text(33.0, 0.5, 'Original Values')
```



```
Sensitivity :  0.8082191780821918
Specificity :  0.9459459459459459
```

Base Result table before under sampling

Model	Accuracy	Precision	Recall	F1 Score
KNN (K =2)	**92.36%**	**0.94**	**0.93**	**0.93**
KNN (K=10)	**93%**	**0.93**	**0.94**	**0.93**
KNN (K=20)	**94%**	**0.94**	**0.94**	**0.93**
KNN (K=25)	**93%**	**0.93**	**0.94**	**0.93**
RFC	**97%**	**0.97**	**0.97**	**0.97**
ANN	**93%**	**0.94**	**0.94**	**0.94**

Base Result Table After Under sampling

Model	Accuracy	Precision	Recall	F1 Score
KNN (K =2)	**76%**	**0.76**	**0.76**	**0.76**
KNN (K=10)	**75%**	**0.75**	**0.75**	**0.75**
KNN (K=20)	**70%**	**0.70**	**0.70**	**0.70**
KNN (K=25)	**70%**	**0.70**	**0.70**	**0.70**
RFC	**87%**	**0.88**	**0.88**	**0.88**
ANN	**66%**	**0.68**	**0.67**	**0.66**

Base and proposed results comparison table

	Accuracy	Precision	Recall	F1 score
Base	**97.26%**	**0.97**	**0.97**	**0.97**
Propose	**99%**	**0.99**	**0.98**	**0.98**