Iyad Ashraf Khamis

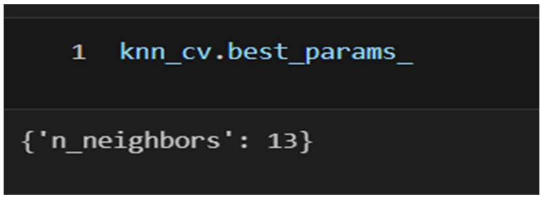
7392

Introduction to machine learning

Assignment 1

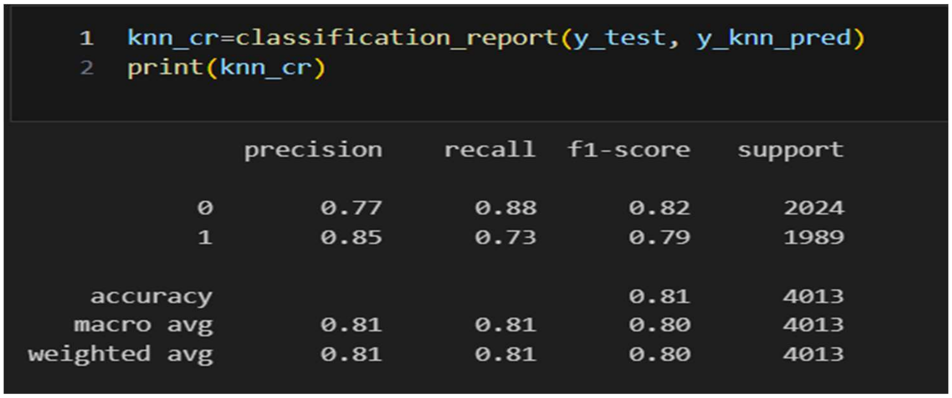
1.KNN

Best performance was when ‘k’ was = 13 with accuracy of 0.81



```
knn_cv.best_params_
```

```
{'n_neighbors': 13}
```

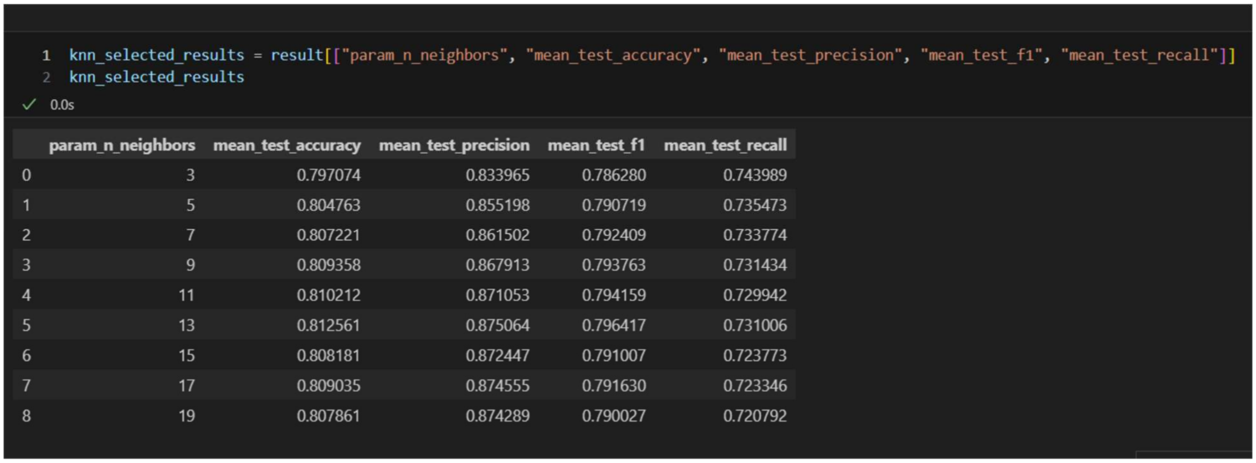


```
knn_cr=classification_report(y_test, y_knn_pred)
print(knn_cr)
```

```
              precision    recall  f1-score   support

           0       0.77      0.88      0.82      2024
           1       0.85      0.73      0.79      1989

    accuracy                           0.81      4013
   macro avg       0.81      0.81      0.80      4013
weighted avg       0.81      0.81      0.80      4013
```



```
knn_selected_results = result[["param_n_neighbors", "mean_test_accuracy", "mean_test_precision", "mean_test_f1", "mean_test_recall"]]
knn_selected_results
```

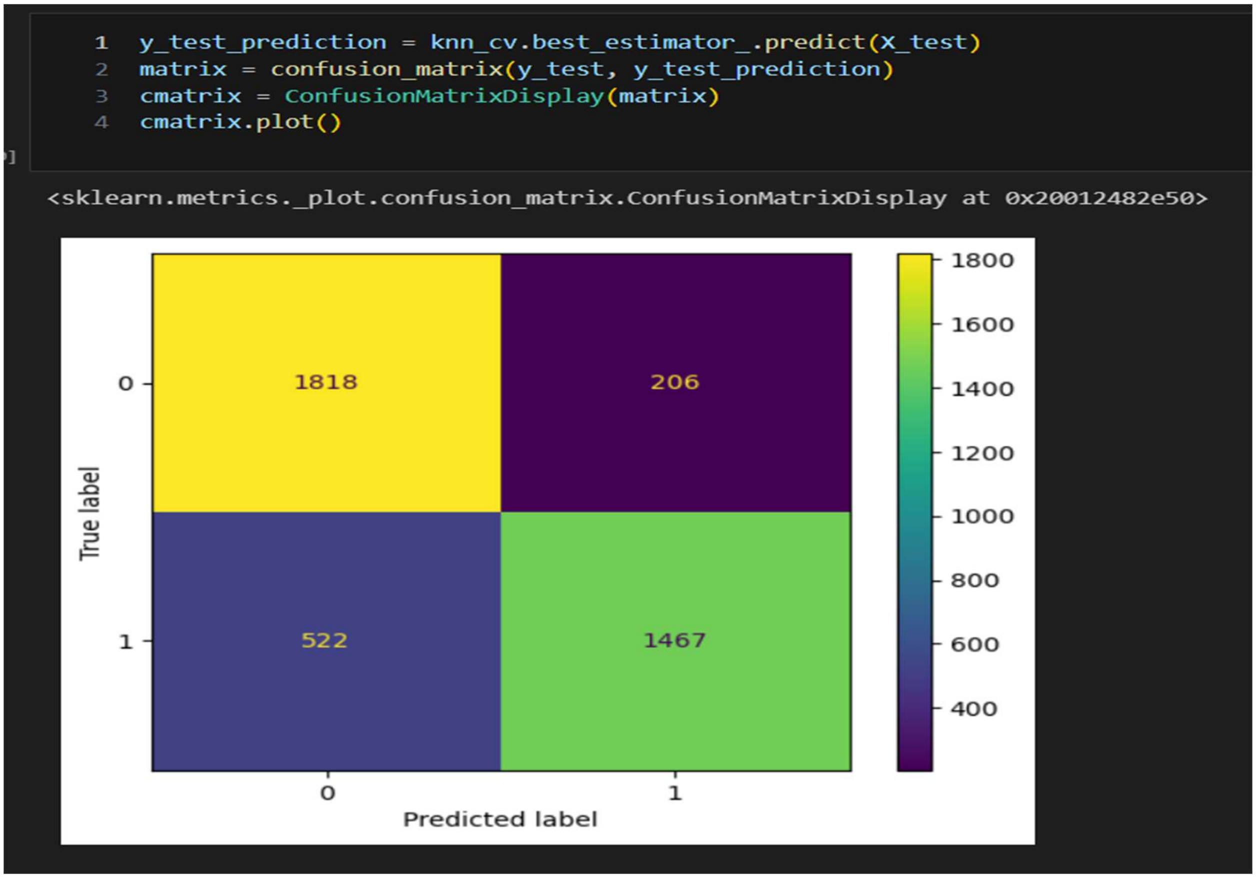
✓ 0.0s

	param_n_neighbors	mean_test_accuracy	mean_test_precision	mean_test_f1	mean_test_recall
0	3	0.797074	0.833965	0.786280	0.743989
1	5	0.804763	0.855198	0.790719	0.735473
2	7	0.807221	0.861502	0.792409	0.733774
3	9	0.809358	0.867913	0.793763	0.731434
4	11	0.810212	0.871053	0.794159	0.729942
5	13	0.812561	0.875064	0.796417	0.731006
6	15	0.808181	0.872447	0.791007	0.723773
7	17	0.809035	0.874555	0.791630	0.723346
8	19	0.807861	0.874289	0.790027	0.720792

-Confusion Matrix of KNN Model:

The number of true positive ‘1’ is 1467 while number of true negative ‘0’ is 1818.

Where ‘0’ is gamma and ‘1’ is hadrons.

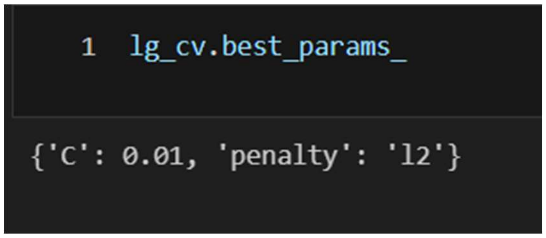


```
y_test_prediction = knn_cv.best_estimator_.predict(X_test)
matrix = confusion_matrix(y_test, y_test_prediction)
cmatrix = ConfusionMatrixDisplay(matrix)
cmatrix.plot()
```

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x20012482e50>
```

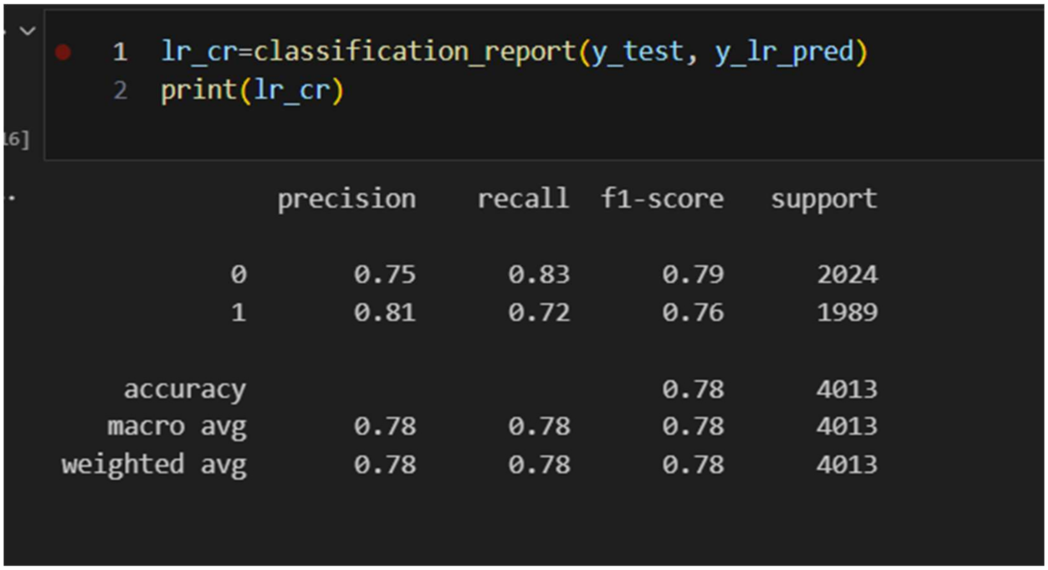
2.Logistic regression

Best performance was when ‘c’ was = 0.01 and penalty =’l2’ with accuracy of 0.78



```
lg_cv.best_params_
```

```
{'C': 0.01, 'penalty': 'l2'}
```

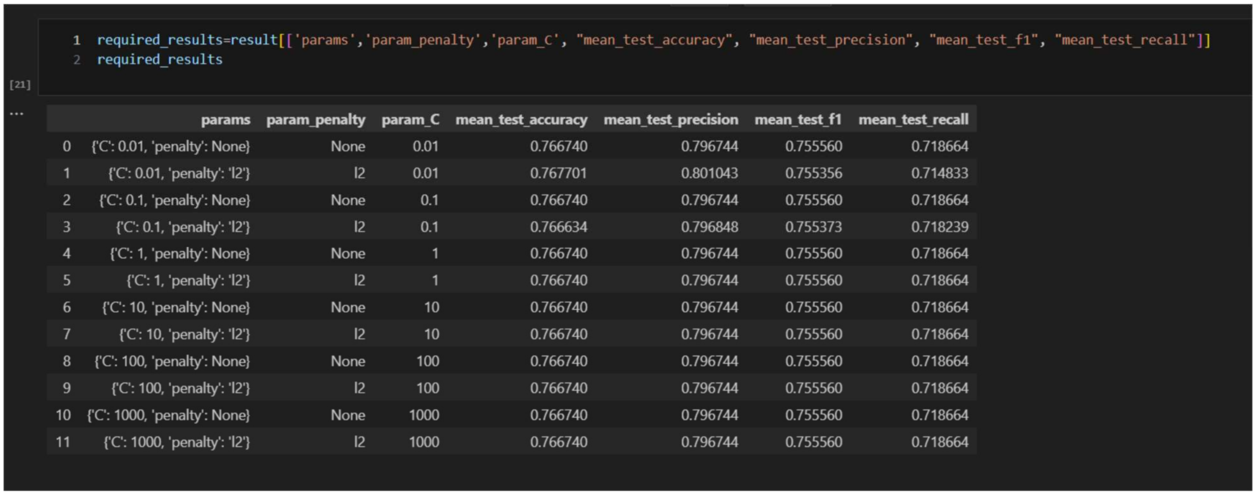


```
lr_cr=classification_report(y_test, y_lr_pred)
print(lr_cr)
```

```
              precision    recall  f1-score   support

           0       0.75      0.83      0.79      2024
           1       0.81      0.72      0.76      1989

    accuracy                           0.78      4013
   macro avg       0.78      0.78      0.78      4013
weighted avg       0.78      0.78      0.78      4013
```



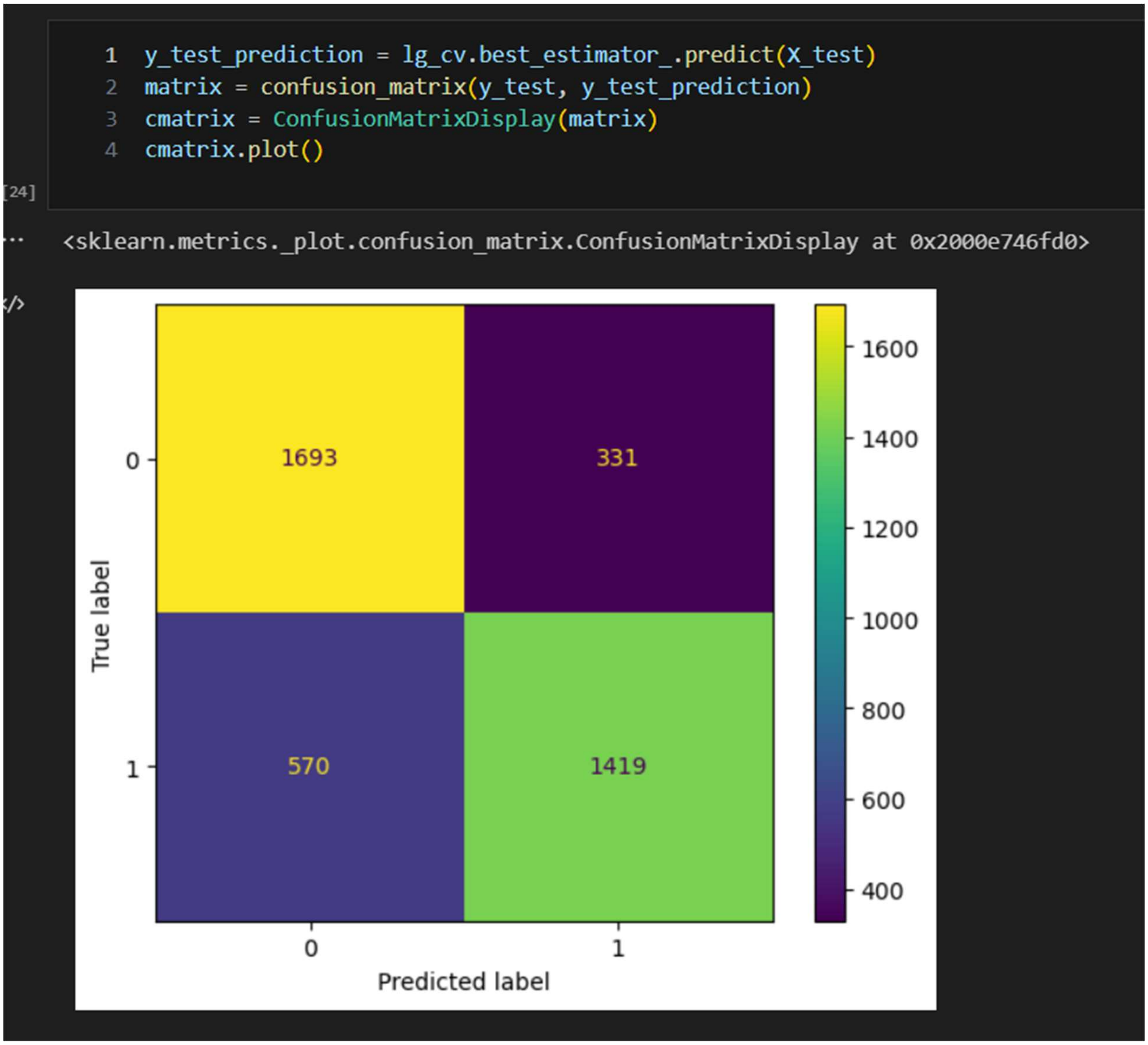
```
required_results=result[['params','param_penalty','param_C', "mean_test_accuracy", "mean_test_precision", "mean_test_f1", "mean_test_recall"]]
required_results
```

	params	param_penalty	param_C	mean_test_accuracy	mean_test_precision	mean_test_f1	mean_test_recall
0	{'C': 0.01, 'penalty': None}	None	0.01	0.766740	0.796744	0.755560	0.718664
1	{'C': 0.01, 'penalty': 'l2'}	l2	0.01	0.767701	0.801043	0.755356	0.714833
2	{'C': 0.1, 'penalty': None}	None	0.1	0.766740	0.796744	0.755560	0.718664
3	{'C': 0.1, 'penalty': 'l2'}	l2	0.1	0.766634	0.796848	0.755373	0.718239
4	{'C': 1, 'penalty': None}	None	1	0.766740	0.796744	0.755560	0.718664
5	{'C': 1, 'penalty': 'l2'}	l2	1	0.766740	0.796744	0.755560	0.718664
6	{'C': 10, 'penalty': None}	None	10	0.766740	0.796744	0.755560	0.718664
7	{'C': 10, 'penalty': 'l2'}	l2	10	0.766740	0.796744	0.755560	0.718664
8	{'C': 100, 'penalty': None}	None	100	0.766740	0.796744	0.755560	0.718664
9	{'C': 100, 'penalty': 'l2'}	l2	100	0.766740	0.796744	0.755560	0.718664
10	{'C': 1000, 'penalty': None}	None	1000	0.766740	0.796744	0.755560	0.718664
11	{'C': 1000, 'penalty': 'l2'}	l2	1000	0.766740	0.796744	0.755560	0.718664

-Confusion Matrix of KNN Model:

The number of true positive ‘1’ is 1419 while number of true negative ‘0’ is 1693.

Where ‘0’ is gamma and ‘1’ is hadrons.



```
y_test_prediction = lg_cv.best_estimator_.predict(X_test)
matrix = confusion_matrix(y_test, y_test_prediction)
cmatrix = ConfusionMatrixDisplay(matrix)
cmatrix.plot()
```

[24]

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x2000e746fd0>
```