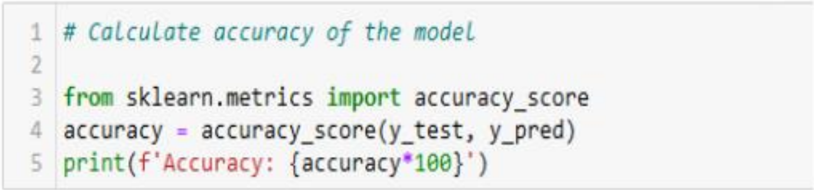
Model Development Phase Template

Date	16 July 2024
Team ID	739870
Project Title	Freedom Of The World Classification
Maximum Marks	10 Marks

Initial Model Training Code, Model Validation and Evaluation Report

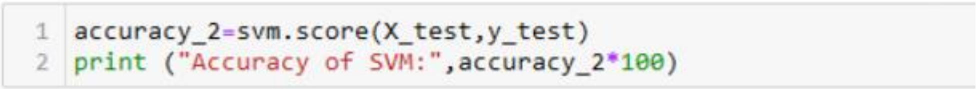
The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include a summary and training and validation performance metrics for multiple models, presented through respective screenshots.

Initial Model Training Code (5 marks):

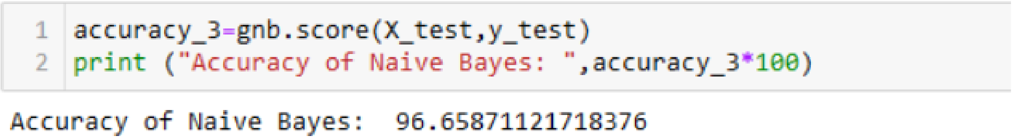


```python
# Calculate accuracy of the model

from sklearn.metrics import accuracy_score
accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy: {accuracy*100}')
```



```python
accuracy_2=svm.score(X_test,y_test)
print ("Accuracy of SVM:",accuracy_2*100)
```

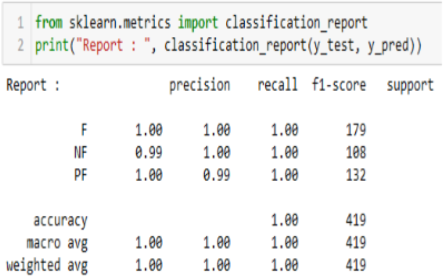


```python
accuracy_3=gnb.score(X_test,y_test)
print ("Accuracy of Naive Bayes: ",accuracy_3*100)
```

```
Accuracy of Naive Bayes:  96.65871121718376
```

Model Validation and Evaluation Report (5 marks):

<table>
<tr><th>Model</th><th>Classification Report</th><th>Training and Validation Performance Metrics</th></tr>
<tr><td>Model 1</td><td>



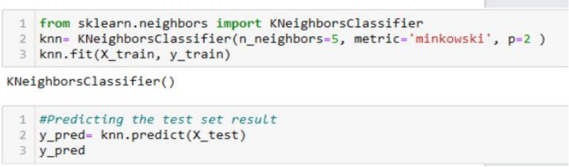
```
from sklearn.metrics import classification_report
print("Report : ", classification_report(y_test, y_pred))
```

```
Report :               precision    recall  f1-score   support

           F       1.00      1.00      1.00       179
          NF       0.99      1.00      1.00       108
          PF       1.00      0.99      1.00       132

    accuracy                           1.00       419
   macro avg       1.00      1.00      1.00       419
weighted avg       1.00      1.00      1.00       419
```

</td><td>

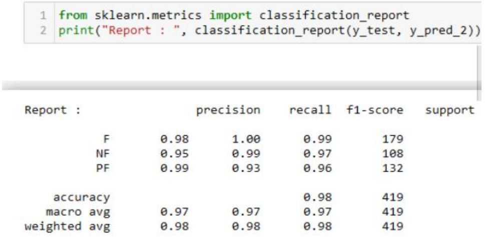


```
from sklearn.neighbors import KNeighborsClassifier
knn= KNeighborsClassifier(n_neighbors=5, metric='minkowski', p=2 )
knn.fit(X_train, y_train)
```

```
KNeighborsClassifier()
```

```
#Predicting the test set result
y_pred= knn.predict(X_test)
y_pred
```

</td></tr>
<tr><td>Model 2</td><td>



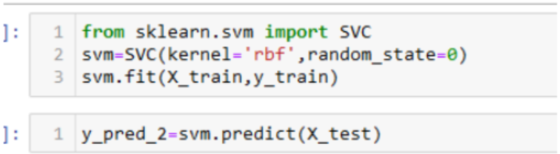
```
from sklearn.metrics import classification_report
print("Report : ", classification_report(y_test, y_pred_2))
```

```
Report :               precision    recall  f1-score   support

           F       0.98      1.00      0.99       179
          NF       0.95      0.99      0.97       108
          PF       0.99      0.93      0.96       132

    accuracy                           0.98       419
   macro avg       0.97      0.97      0.97       419
weighted avg       0.98      0.98      0.98       419
```

</td><td>



```
]: from sklearn.svm import SVC
   svm=SVC(kernel='rbf',random_state=0)
   svm.fit(X_train,y_train)
```

```
]: y_pred_2=svm.predict(X_test)
```

</td></tr>
</table>