

## Model Development Phase Template

Date	16 July 2024
Team ID	739942
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):

```
1 # Calculate accuracy of the model
2
3 from sklearn.metrics import accuracy_score
4 accuracy = accuracy_score(y_test, y_pred)
5 print(f'Accuracy: {accuracy*100}')
```

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

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

Accuracy of Naive Bayes: 96.65871121718376

### Model Validation and Evaluation Report (5 marks):

Model	Summary	Training and Validation Performance Metrics																																			
Model 1	<pre>1 from sklearn.metrics import classification_report 2 print("Report : ", classification_report(y_test, y_pred))</pre> <table><tr><td>Report :</td><td>precision</td><td>recall</td><td>f1-score</td><td>support</td></tr><tr><td>F</td><td>1.00</td><td>1.00</td><td>1.00</td><td>179</td></tr><tr><td>NF</td><td>0.99</td><td>1.00</td><td>1.00</td><td>108</td></tr><tr><td>PF</td><td>1.00</td><td>0.99</td><td>1.00</td><td>132</td></tr><tr><td>accuracy</td><td></td><td></td><td>1.00</td><td>419</td></tr><tr><td>macro avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>419</td></tr><tr><td>weighted avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>419</td></tr></table>	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	<pre>1 from sklearn.neighbors import KNeighborsClassifier 2 knn= KNeighborsClassifier(n_neighbors=5, metric='minkowski', p=2 ) 3 knn.fit(X_train, y_train)  KNeighborsClassifier()  1 #Predicting the test set result 2 y_pred= knn.predict(X_test) 3 y_pred</pre>
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																																	
Model 2	<pre>1 from sklearn.metrics import classification_report 2 print("Report : ", classification_report(y_test, y_pred_2))</pre> <table><tr><td>Report :</td><td>precision</td><td>recall</td><td>f1-score</td><td>support</td></tr><tr><td>F</td><td>0.98</td><td>1.00</td><td>0.99</td><td>179</td></tr><tr><td>NF</td><td>0.95</td><td>0.99</td><td>0.97</td><td>108</td></tr><tr><td>PF</td><td>0.99</td><td>0.93</td><td>0.96</td><td>132</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.98</td><td>419</td></tr><tr><td>macro avg</td><td>0.97</td><td>0.97</td><td>0.97</td><td>419</td></tr><tr><td>weighted avg</td><td>0.98</td><td>0.98</td><td>0.98</td><td>419</td></tr></table>	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	<pre>] : 1 from sklearn.svm import SVC 2 svm=SVC(kernel='rbf',random_state=0) 3 svm.fit(X_train,y_train)  ]: 1 y_pred_2=svm.predict(X_test)</pre>
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																																	