

## Model Development Phase Template

Date	15 JULY 2024
Team ID	740075
Project Title	Detection Of Autistic Spectrum Disorder: Classification
Maximum Marks	4 Marks

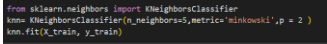
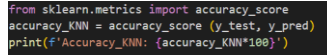
### Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

#### Initial Model Training Code:

Paste the screenshot of the model training code

#### Model Validation and Evaluation Report:

Model	Classification Report	Accuracy	Confusion Matrix
1. K Nearest Neighbors Model		55.18867	

			<pre> precision    recall  f1-score   support   apple      1.00    1.00    1.00        23  banana     1.00    1.00    1.00        20  blackgram   0.91    1.00    0.95        21  chickpea    1.00    1.00    1.00        22  coconut     1.00    1.00    1.00        20  coffee      1.00    1.00    1.00        24  cotton      0.95    1.00    0.98        20  grapes      1.00    1.00    1.00        20  jute        0.74    0.94    0.83        18  kidneybeans 0.90    1.00    0.95        19  lentil      1.00    0.96    0.98        25  maize       1.00    0.90    0.95        20  mango       1.00    1.00    1.00        17  mothbeans   0.93    0.93    0.93        14  mungbean    1.00    1.00    1.00        20  muskmelon   1.00    1.00    1.00        18  orange      1.00    1.00    1.00        24  papaya      1.00    0.93    0.97        15  pigeonpeas  1.00    0.90    0.95        21  pomegranate 1.00    1.00    1.00        23  rice        0.93    0.74    0.82        15  watermelon  1.00    1.00    1.00        17   accuracy   0.97    0.97    0.97       440  macro avg  0.97    0.97    0.97       440  weighted avg 0.97    0.97    0.97       440 </pre>
2. SVM Model	<pre> from sklearn.svm import SVC svm=SVC(kernel='rbf', random_state=0) svm.fit(X_train,y_train) </pre>	9.433	<pre> y_pred_svm=svm.predict(X_test) print('Training Set:',svm.score(X_train,y_train)) print('Training Set:',svm.score(X_test,y_test))  Training Set: 0.13211382113821138 Training Set: 0.09433962264150944 </pre>
3. Decision Tree Model	<pre> dt=DecisionTreeClassifier() dt.fit(X_train,y_train)  DecisionTreeClassifier DecisionTreeClassifier() </pre>	97.166	<pre> ... y_pred_dt=dt.predict(X_test) print('Training Set:',dt.score(X_train,y_train)) print('Training Set:',dt.score(X_test,y_test))  Training Set: 1.0 Training Set: 0.9716981132075472 </pre>
4. Random Forest Model	<pre> rand_forest=RandomForestClassifier(random_state=42) rand_forest.fit(X_train,y_train)  RandomForestClassifier RandomForestClassifier(random_state=42) </pre>	97.1669	<pre> predictionF=rand_forest.predict(X_test) print('Training Set:',rand_forest.score(X_train,y_train)) print('Training Set:',rand_forest.score(X_test,y_test))  Training Set: 1.0 Training Set: 0.37264150943396224 </pre>