Model Optimization and Tuning Phase Template

Date	March 2024
Team ID	xxxxxx
Project Title	Detection of Autistic Spectrum Disorder: Classification
Maximum Marks	10 Marks

Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

Hyperparameter Tuning Documentation (6 Marks):

Model	Tuned Hyperparameters

Decision Tree	**Decision Tree**
	
	`dt = DecisionTreeClassifier()`
	
	`dt.fit(X_train,y_train)`
	
	DecisionTreeClassifier
	DecisionTreeClassifier()
	
	`y_pred_dt=dt.predict(X_test)`
	
	`print('Training Set: ',dt.score(X_train,y_train))`
	`print('Test Set: ',dt.score(X_test,y_test))`
	
	Training Set: 1.0
	Test Set: 1.0
	
	`print("Accuracy:", metrics.accuracy_score(y_test, y_pred_dt)*100)`
	
	Accuracy: 100.0

Random Forest	**Random Forest**
	
	`rand_forest = RandomForestClassifier(random_state=42)`
	
	`rand_forest.fit(X_train, y_train)`
	
	RandomForestClassifier
	RandomForestClassifier(random_state=42)
	
	`predictionRF = rand_forest.predict(X_test)`
	
	`print('Training set: ',rand_forest.score(X_train, y_train))`
	`print('Testing set: ',rand_forest.score(X_test, y_test))`
	
	Training set: 1.0
	Testing set: 1.0
	
	`accuracy_RF=rand_forest.score(X_test, y_test)`
	`print ("Accuracy_RF:",accuracy_RF*100)`
	
	Accuracy_RF: 100.0

KNN	KNN
	`from sklearn.neighbors import KNeighborsClassifier`
	`knn= KNeighborsClassifier(n_neighbors=5, metric='minkowski`
	`knn.fit(X_train, y_train)`
	KNeighborsClassifier
	KNeighborsClassifier()
	`y_pred = knn.predict(X_test)`
	`#Calculate accuracy of the model`
	`from sklearn.metrics import accuracy_score`
	`accuracy_KNN = accuracy_score(y_test, y_pred)`
	`print(f'Accuracy_KNN: {accuracy_KNN*100}')`
	Accuracy_KNN: 96.17486338797814

Performance Metrics Comparison Report (2 Marks):

Model	Beline Metric

SVM	SVC
	`from sklearn.svm import SVC`
	`svm=SVC(kernel='rbf', random_state=0)`
	`svm.fit(X_train, y_train)`
	`SVC(random_state=0)`
	`y_pred_svc=svm.predict(X_test)`
	`print('Training Set: ', svm.score (X_train,y_train))`
	`print('Testing Set:',svm.score(X_test,y_test))`
	Training Set: 0.9530516431924883
	Testing Set: 0.9453551912568307
	`accuracy_SVC=svm.score(X_test,y_test)`
	`print('Accuracy_SVM:', accuracy_SVC*100)`
	Accuracy_SVM: 94.53551912568307

Logistic Regression	`from sklearn.model_selection import train_test_split`
	`X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)`
	`from sklearn.linear_model import LogisticRegression`
	`lgr=LogisticRegression()`
	`lgr.fit(X_train,y_train)`
	`LogisticRegression()`
	`pred=lgr.predict(X_test)`

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
Detection of Autistic Spectrum Disorder: Classification	Final Model" in machine learning is the culmination of rigorous evaluation, selection, and validation processes, aiming to identify the model that best meets performance criteria and practical considerations for deployment.