Model Development Phase Template

Date	5th July 2024
Team ID	739794
Project Title	Cost Prediction of Acquiring a Customer.
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:

```python
RandomForest Regressor:
from sklearn.ensemble import RandomForestRegressor
rf=RandomForestRegressor()
rf.fit(X1_train,Y_train)
y_pred=rf.predict(X1_test)
x_pred=rf.predict(X1_train)
score=r2_score(Y_test,y_pred)
rmse=np.sqrt(mean_squared_error(Y_test,y_pred))
score1=r2_score(Y_train,x_pred)
rmse1=np.sqrt(mean_squared_error(Y_train,x_pred))

DecisionTree Regressor:
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import
r2_score,mean_squared_error,mean_absolute_error
Dtr=DecisionTreeRegressor()
Dtr.fit(X1_train,Y_train)
y_pred=Dtr.predict(X1_test)
x_pred=Dtr.predict(X1_train)
score=r2_score(Y_test,y_pred)
rmse=np.sqrt(mean_squared_error(Y_test,y_pred))
score1=r2_score(Y_train,x_pred)
rmse1=np.sqrt(mean_squared_error(Y_train,x_pred))
mae=mean_absolute_error(Y_test,y_pred)
mse=mean_squared_error(Y_test,y_pred)
```

Gradient BoostingRegressor:

```python
from sklearn.ensemble import GradientBoostingRegressor
gb=GradientBoostingRegressor()
gb.fit(X1_train,Y_train)
y_pred=gb.predict(X1_test)
x_pred=rf.predict(X1_train)
score=r2_score(Y_test,y_pred)
rmse=np.sqrt(mean_squared_error(Y_test,y_pred))
score1=r2_score(Y_train,x_pred)
rmse1=np.sqrt(mean_squared_error(Y_train,x_pred))
```

Model Validation and Evaluation Report:

Model	Classification Report	Accuracy	Confusion Matrix
Random Forest Regressor	```from sklearn.ensemble import RandomForestRegressor rf=RandomForestRegressor() rf.fit(X1_train,Y_train) y_pred=rf.predict(X1_test) x_pred=rf.predict(X1_train) score=r2_score(Y_test,y_pred) rmse=np.sqrt(mean_squared_error(Y_test,y_pred)) score1=r2_score(Y_train,x_pred) rmse1=np.sqrt(mean_squared_error(Y_train,x_pred))``` ---------------------------------- Model performance for Test set R2_Score = 0.9999992223770472 RMSE = 0.026495659012272195 ---------------------------------- Model performance for Train set R2_Score = 0.9999999370963878 RMSE = 0.007523146092761905	99%	-
Decision Tree Regressor	```print('---****---Decision Tree Regressor model---****---') print('Model performance for test set') print('R2 Score = ',score) print('RMSE = ',rmse) print('--------------------------------') print('Model performance for Test set') print('R2_Score = ',score1) print('RMSE = ',rmse1) print("mean_absolute_error: ",mae) print("mean_squared_error: ",mse)``` ---****---Decision Tree Regressor model---****--- Model performance for test set R2 Score = 0.999999475396002 RMSE = 0.021762359911033424 -------------------------------- Model performance for Test set R2_Score = 1.0 RMSE = 2.7555815953433333e-13 mean_absolute_error: 0.0002487726848418434 mean_squared_error: 0.00047360030889735465	99%	-

Gradient Boosting Regressor		99%	-

```
print('---****---Gradient Boosting Regressor model---****---')
print('Model performance for test set')
print('R2 Score = ',score)
print('RMSE = ',rmse)

print('-------------------------------')
print('Model performance for Test set')
print('R2_Score = ',score1)
print('RMSE = ',rmse1)
```

```
---****---Gradient Boosting Regressor model---****---
Model performance for test set
R2 Score =  0.9998527192407322
RMSE =  0.36463906787159356
-------------------------------
Model performance for Test set
R2_Score =  0.9999999370963878
RMSE =  0.007523146092761905
```