Model Development Phase Template

Date	24 April 2024
Team ID	739877
Project Title	Crystal Ball Analysis: Projecting Share Prices Of The Leading Gpu Titans
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):

Paste the screenshot of the model training code

Model Validation and Evaluation Report (5 marks):

Model	Summary	Training and Validation Performance Metrics
| Model 1 | For a Linear Regression model, performance metrics include Mean Absolute Error (MAE) which measures average absolute errors, Mean Squared Error (MSE) which squares the errors to penalize larger deviations, and R-squared which indicates the proportion of variance in the dependent variable explained by the model. | ```python
[ ] x_train.fillna(x_train.mean(), inplace=True)
    x_test.fillna(x_test.mean(), inplace=True)

[ ] y_train.fillna(y_train.mean(), inplace=True)
    y_test.fillna(y_test.mean(), inplace=True)
```<br>using linear regression<br>```python
[ ] lr=LinearRegression()
    lr.fit(x_train,y_train)
```<br>▾ LinearRegression<br>LinearRegression() |
| Model 2 | Decision tree classifier model commonly include accuracy, precision, recall, F1 score which help assess the model's prediction accuracy and generalizability | using decision tree<br>```python
from sklearn.tree import DecisionTreeRegressor
det=DecisionTreeRegressor()
det.fit(x_train,y_train)
```<br>▾ DecisionTreeRegressor<br>DecisionTreeRegressor()<br>```python
[ ] print('test score:',det.score(x_train,y_train))
    print('train score:',det.score(x_test,y_test))
```<br>test score: 1.0<br>train score: 0.9978457284488803<br>```python
[ ] y_pred=det.predict(x_test)
    print('r2 score:',r2_score(y_test,y_pred))
    print('mean absolute error:',mean_absolute_error(y_test,y_pred))
```<br>r2 score: 0.9978457284488803<br>mean absolute error: 2.140286973858129 |