Model Development Phase

Date	11 July 2024
Team ID	SWTID1720359900
Project Title	Machine Learning Approach For Predicting The Price Of Natural Gas
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:

```
from sklearn.model_selection import train_test_split

X = df_2[['year', 'month', 'day']]
y = df_2['Price']

# Splitting the data into 80% training and 20% testing
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

print("Training set - Features:", X_train.shape, "Labels:", y_train.shape)
print("Testing set - Features:", X_test.shape, "Labels:", y_test.shape)
```

```
Training set - Features: (4749, 3) Labels: (4749,)
Testing set - Features: (1188, 3) Labels: (1188,)
```

```
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_squared_error
regressor = DecisionTreeRegressor(random_state=42)

regressor.fit(X_train, y_train)

y_pred = regressor.predict(X_test)

mse = mean_squared_error(y_test, y_pred)
print(f"Mean Squared Error (MSE): {mse}")
```

```
Mean Squared Error (MSE): 0.09586742424242424
```

```
from sklearn.metrics import r2_score

r2 = r2_score(y_test, y_pred)

print(f'R-squared (R2) Score: {r2:.4f}')
```

```
R-squared (R2) Score: 0.9814
```

```
from sklearn.linear_model import LinearRegression
```

```
lr = LinearRegression()
```

```
lr.fit(X_train,y_train)
```

```
▾ LinearRegression
LinearRegression()
```

```
ypred = lr.predict(X_test)
```

```
ypred
```

```
array([4.45282588, 4.05787683, 3.74130395, ..., 4.14557995, 4.11678668,
       3.98965119])
```

```
from sklearn.metrics import r2_score
acc = r2_score(y_pred,y_test)
acc
```

```
0.9808376083629289
```

Model Validation and Evaluation Report:

Model	Accuracy

Decision Tree	`from sklearn.metrics import r2_score`
	`r2 = r2_score(y_test, y_pred)`
	`print(f'R-squared (R2) Score: {r2:.4f}')`
	`R-squared (R2) Score: 0.9814`

Linear Regression	`from sklearn.metrics import r2_score`
	`acc = r2_score(y_pred,y_test)`
	`acc`
	`0.9808376083629289`