Project Development Phase
Model Performance Test

Date	19 November 2022
Team ID	PNT2022TMID29329
Project Name	DemandEst-AI powered food demand forecaster
Maximum Marks	10 Marks

Model Performance Testing:

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** MAE –105.93055528844432 MSE –50625.66862688001 RMSE –225.00148583260514 R2 score -0.6645666634919214	Attached below
2.	Tune the Model	Hyperparameter Tuning - GridSearchCV Validation Method – KFold cross Validation	Attached below

Metrics:

In [32]:
```
DT=DecisionTreeRegressor()
DT.fit(x_train,y_train)
y_pred=DT.predict(x_val)
y_pred[y_pred<0]=0
from sklearn import metrics
print('RMSLE:',100*np.sqrt(metrics.mean_squared_log_error(y_val,y_pred)))
```

```
RMSLE: 62.76267907544625
```

In [33]:
```
from sklearn.metrics import r2_score
test_score=r2_score(y_val,y_pred)
test_score
```

Out[33]:
```
0.6645666634919214
```

In [34]:
```
train_score=DT.score(x_train,y_train)
train_score
```

Out[34]:
```
0.6792448145375074
```

In [35]:
```
from sklearn import metrics

print('MAE:', metrics.mean_absolute_error(y_val,y_pred))
print('MSE:', metrics.mean_squared_error(y_val,y_pred))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_val,y_pred)))
```

```
MAE: 105.93055528844432
MSE: 50625.66862688001
RMSE: 225.00148583260514
```

Data Validation:

In [41]:
```
from sklearn.model_selection import GridSearchCV
```

In [42]:
```
param_grid = {
    "max_depth":[3,5,10,15,20,None],
    "min_samples_split":[2,5,7,10],
    "min_samples_leaf":[1,2,5]
             }
```

In [45]:
```
grid_cv= GridSearchCV(DT,param_grid,cv=5,n_jobs=-1)
```

In [48]:
```
grid_cv.fit(x_train,y_train)
print('Best Parametrs:',grid_cv.best_params_)
```

```
Best Parametrs: {'max_depth': None, 'min_samples_leaf': 5, 'min_samples_split': 2}
```

In [49]:
```
from sklearn.model_selection import KFold,cross_val_score
cv=KFold(n_splits=10)

print('Trian Data Validation')
print('Decision Tree :'+str(cross_val_score(DT,x_train,y_train,cv=cv).mean()))

print('\n\n')
print('Test Data Validation')
print('Decision Tree :'+str(cross_val_score(DT,x_val,y_val,cv=cv).mean()))
```

```
Trian Data Validation
Decision Tree :0.6542250948553616



Test Data Validation
Decision Tree :0.6067937915420428
```