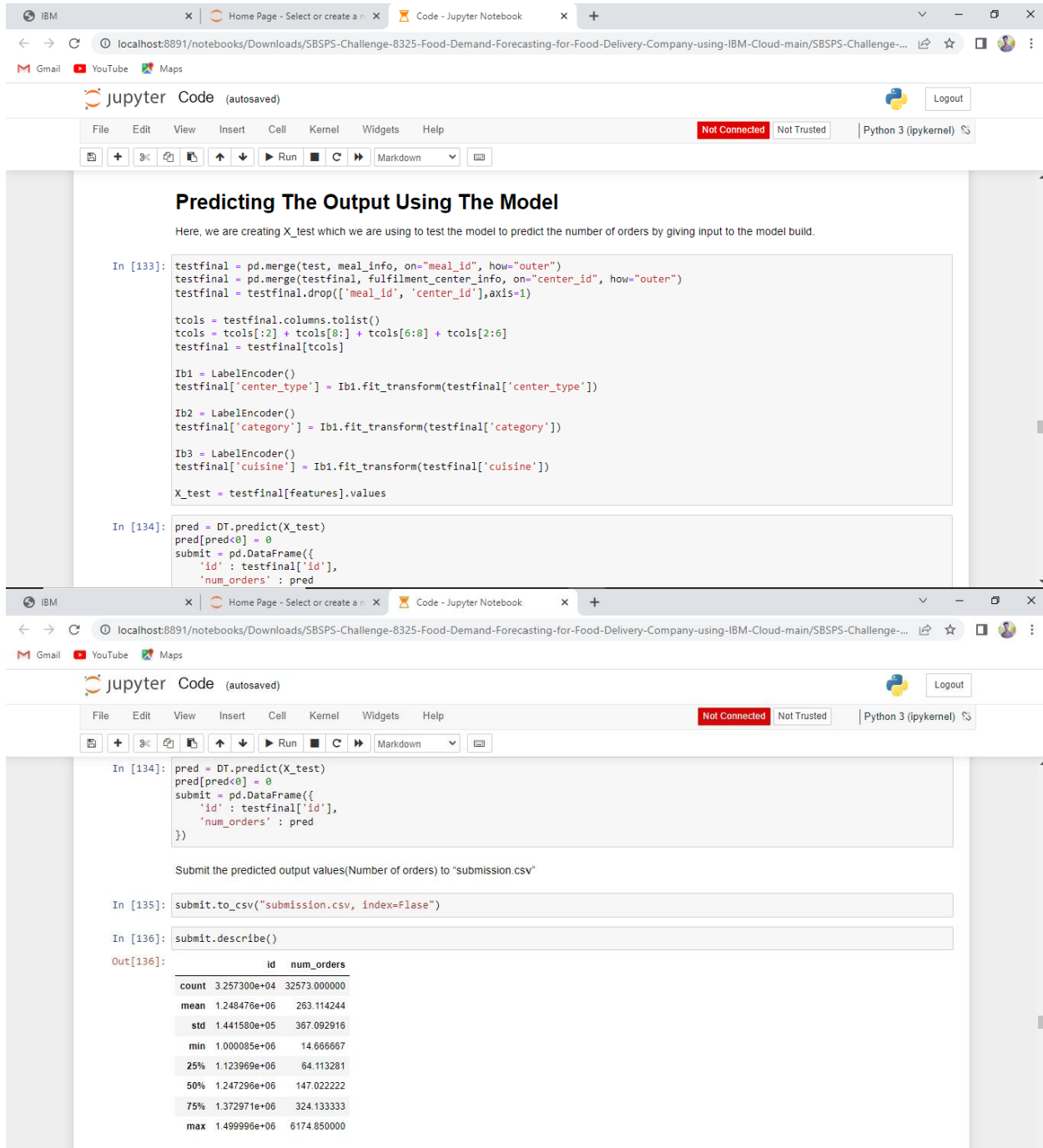


TEAM ID: PNT2022TMID52731

## PROJECT NAME: DemandEst - AI powered Food Demand Forecaster

### Team Leader



The screenshot displays a Jupyter Notebook interface with two visible code cells. The first cell, titled "Predicting The Output Using The Model", contains Python code for data preprocessing and model prediction. The second cell contains code for submitting the predicted values to a CSV file and displaying a summary of the submission data.

**Predicting The Output Using The Model**

Here, we are creating `X_test` which we are using to test the model to predict the number of orders by giving input to the model build.

```
In [133]: testfinal = pd.merge(test, meal_info, on="meal_id", how="outer")
testfinal = pd.merge(testfinal, fulfilment_center_info, on="center_id", how="outer")
testfinal = testfinal.drop(['meal_id', 'center_id'],axis=1)

tcols = testfinal.columns.tolist()
tcols = tcols[:2] + tcols[8:] + tcols[6:8] + tcols[2:6]
testfinal = testfinal[tcols]

Ib1 = LabelEncoder()
testfinal['center_type'] = Ib1.fit_transform(testfinal['center_type'])

Ib2 = LabelEncoder()
testfinal['category'] = Ib1.fit_transform(testfinal['category'])

Ib3 = LabelEncoder()
testfinal['cuisine'] = Ib1.fit_transform(testfinal['cuisine'])

X_test = testfinal[features].values

In [134]: pred = DT.predict(X_test)
pred[pred<0] = 0
submit = pd.DataFrame({
    'id' : testfinal['id'],
    'num_orders' : pred
})
```

Submit the predicted output values(Number of orders) to "submission.csv"

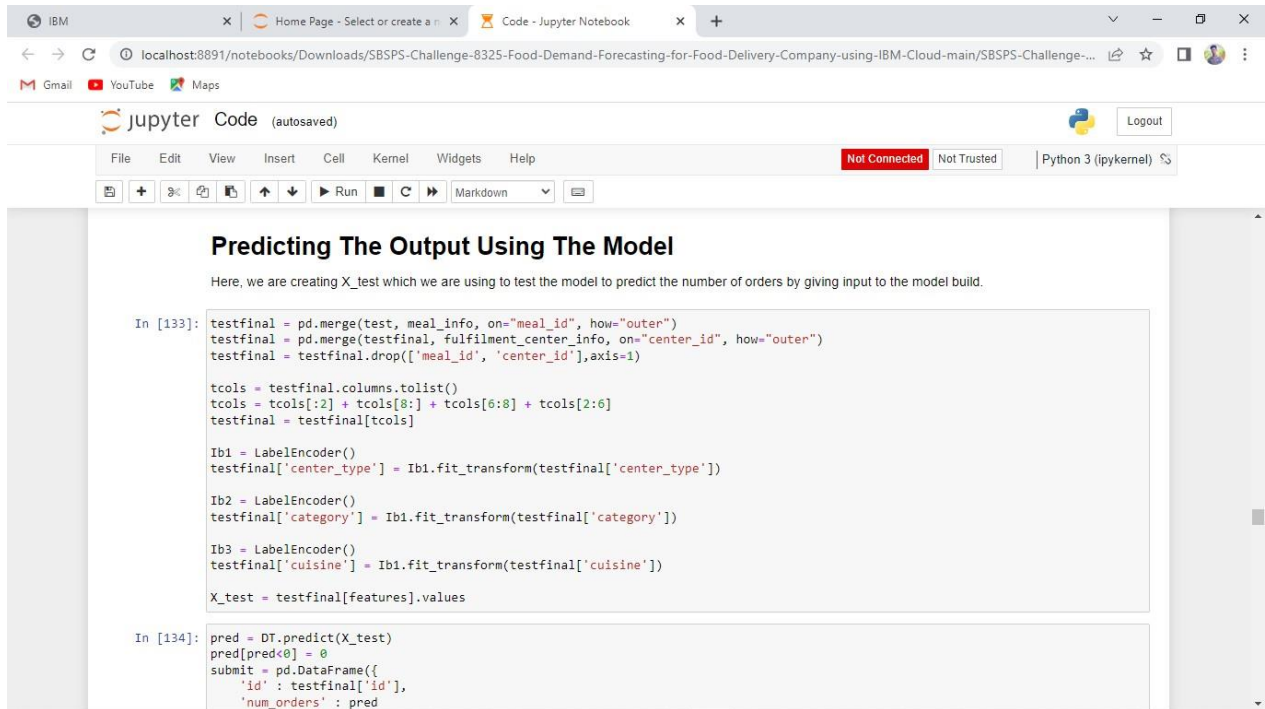
```
In [135]: submit.to_csv("submission.csv", index=False)
```

```
In [136]: submit.describe()
```

Out[136]:

	id	num_orders
count	3.257300e+04	32573.000000
mean	1.248476e+06	263.114244
std	1.441580e+05	367.092916
min	1.000085e+06	14.666667
25%	1.123969e+06	64.113281
50%	1.247296e+06	147.022222
75%	1.372971e+06	324.133333
max	1.499996e+06	6174.850000

# Team Member 1



The screenshot shows a Jupyter Notebook interface with a browser window at the top. The notebook is titled "Predicting The Output Using The Model". Below the title, there is a text block stating: "Here, we are creating X\_test which we are using to test the model to predict the number of orders by giving input to the model build." The notebook contains two code cells. The first cell, labeled "In [133]:", contains code for merging dataframes, dropping columns, and creating a new dataframe 'testfinal' with specific columns. It also includes code for creating LabelEncoders for 'center\_type', 'category', and 'cuisine', and transforming the data. The second cell, labeled "In [134]:", contains code for predicting the output using the model and creating a DataFrame for submission.

```
In [133]: testfinal = pd.merge(test, meal_info, on="meal_id", how="outer")
testfinal = pd.merge(testfinal, fulfilment_center_info, on="center_id", how="outer")
testfinal = testfinal.drop(['meal_id', 'center_id'], axis=1)

tcols = testfinal.columns.tolist()
tcols = tcols[:2] + tcols[8:] + tcols[6:8] + tcols[2:6]
testfinal = testfinal[tcols]

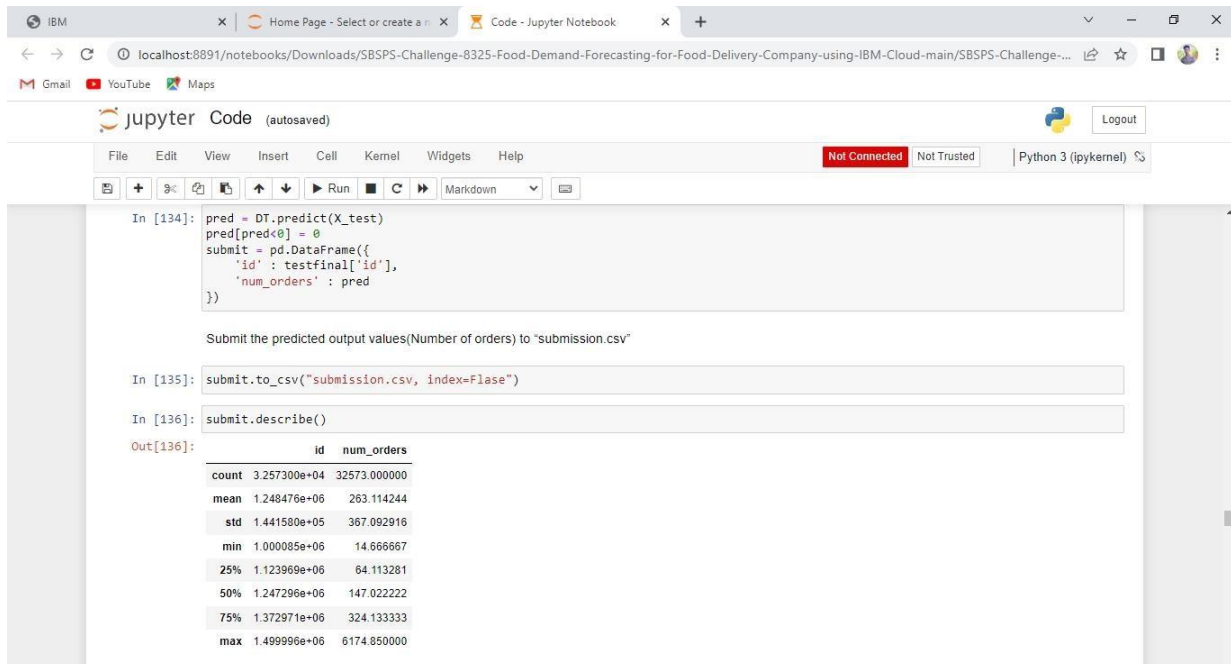
Ib1 = LabelEncoder()
testfinal['center_type'] = Ib1.fit_transform(testfinal['center_type'])

Ib2 = LabelEncoder()
testfinal['category'] = Ib1.fit_transform(testfinal['category'])

Ib3 = LabelEncoder()
testfinal['cuisine'] = Ib1.fit_transform(testfinal['cuisine'])

X_test = testfinal[features].values

In [134]: pred = DT.predict(X_test)
pred[pred<0] = 0
submit = pd.DataFrame({
    'id': testfinal['id'],
    'num_orders': pred
```



The screenshot shows a Jupyter Notebook interface with a browser window at the top. The notebook contains two code cells. The first cell, labeled "In [134]:", contains code for predicting the output using the model and creating a DataFrame for submission. The second cell, labeled "In [135]:", contains code for submitting the predicted output values to a CSV file. The third cell, labeled "In [136]:", contains code for describing the submission DataFrame. The output of the third cell is displayed below the code cell.

```
In [134]: pred = DT.predict(X_test)
pred[pred<0] = 0
submit = pd.DataFrame({
    'id': testfinal['id'],
    'num_orders': pred
})

Submit the predicted output values(Number of orders) to "submission.csv"

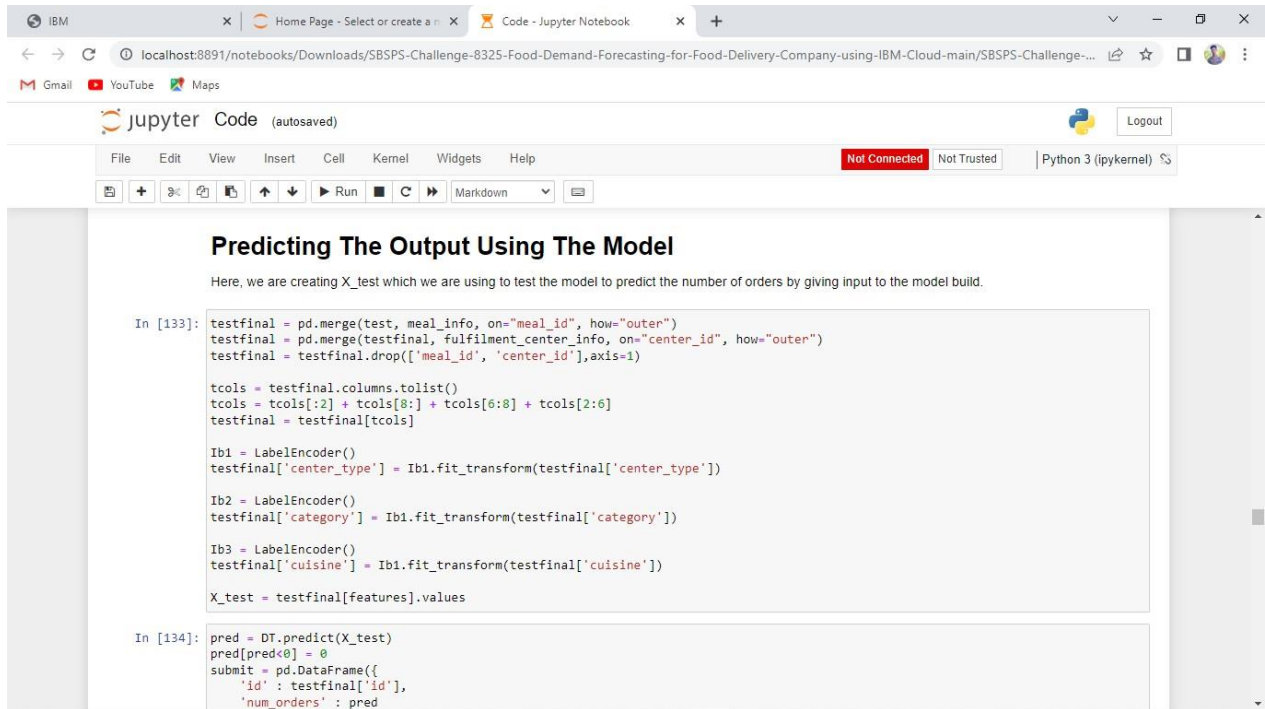
In [135]: submit.to_csv("submission.csv", index=False)

In [136]: submit.describe()

Out[136]:
```

	id	num_orders
count	3.257300e+04	32573.000000
mean	1.248476e+06	263.114244
std	1.441580e+05	367.092916
min	1.000085e+06	14.666667
25%	1.123969e+06	64.113281
50%	1.247296e+06	147.022222
75%	1.372971e+06	324.133333
max	1.499996e+06	6174.850000

## Team Member 2



The screenshot shows a Jupyter Notebook interface with a browser window at the top. The notebook is titled "jupyter Code (autosaved)" and has a "Python 3 (ipykernel)" kernel. The code is as follows:

```
In [133]: testfinal = pd.merge(test, meal_info, on="meal_id", how="outer")
testfinal = pd.merge(testfinal, fulfilment_center_info, on="center_id", how="outer")
testfinal = testfinal.drop(['meal_id', 'center_id'], axis=1)

tcols = testfinal.columns.tolist()
tcols = tcols[:2] + tcols[8:] + tcols[6:8] + tcols[2:6]
testfinal = testfinal[tcols]

Ib1 = LabelEncoder()
testfinal['center_type'] = Ib1.fit_transform(testfinal['center_type'])

Ib2 = LabelEncoder()
testfinal['category'] = Ib1.fit_transform(testfinal['category'])

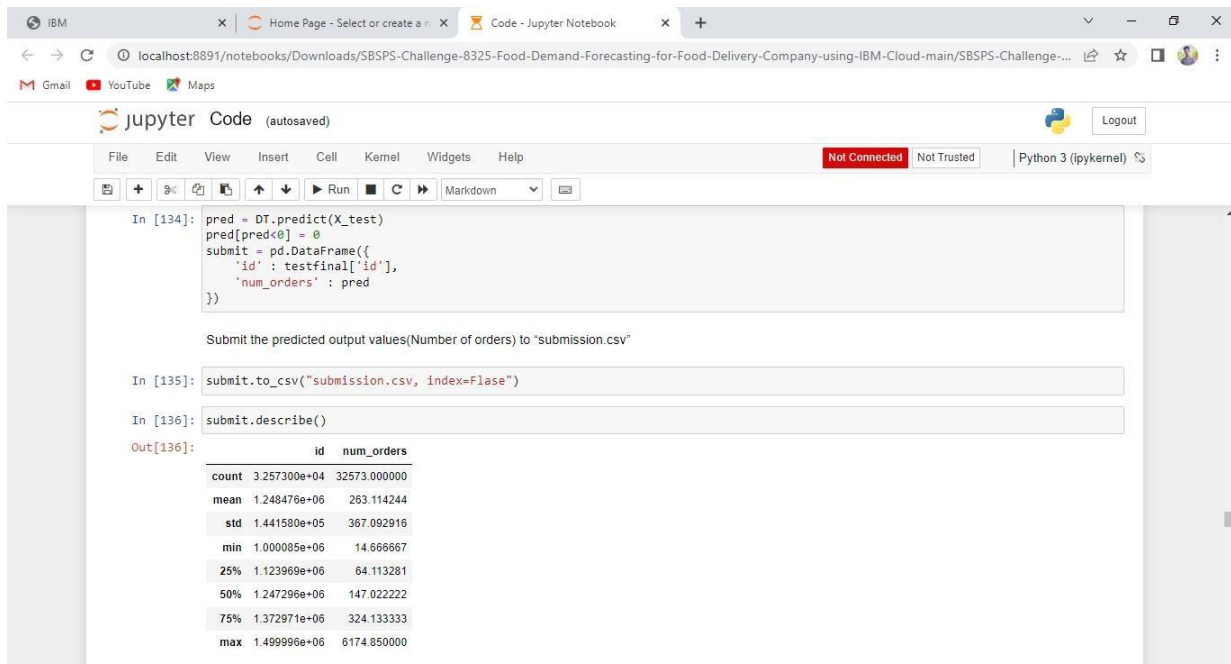
Ib3 = LabelEncoder()
testfinal['cuisine'] = Ib1.fit_transform(testfinal['cuisine'])

X_test = testfinal[features].values

In [134]: pred = DT.predict(X_test)
pred[pred<0] = 0
submit = pd.DataFrame({
    'id': testfinal['id'],
    'num_orders': pred
})
```

Predicting The Output Using The Model

Here, we are creating X\_test which we are using to test the model to predict the number of orders by giving input to the model build.



The screenshot shows a Jupyter Notebook interface with a browser window at the top. The notebook is titled "jupyter Code (autosaved)" and has a "Python 3 (ipykernel)" kernel. The code is as follows:

```
In [134]: pred = DT.predict(X_test)
pred[pred<0] = 0
submit = pd.DataFrame({
    'id': testfinal['id'],
    'num_orders': pred
})

Submit the predicted output values(Number of orders) to "submission.csv"

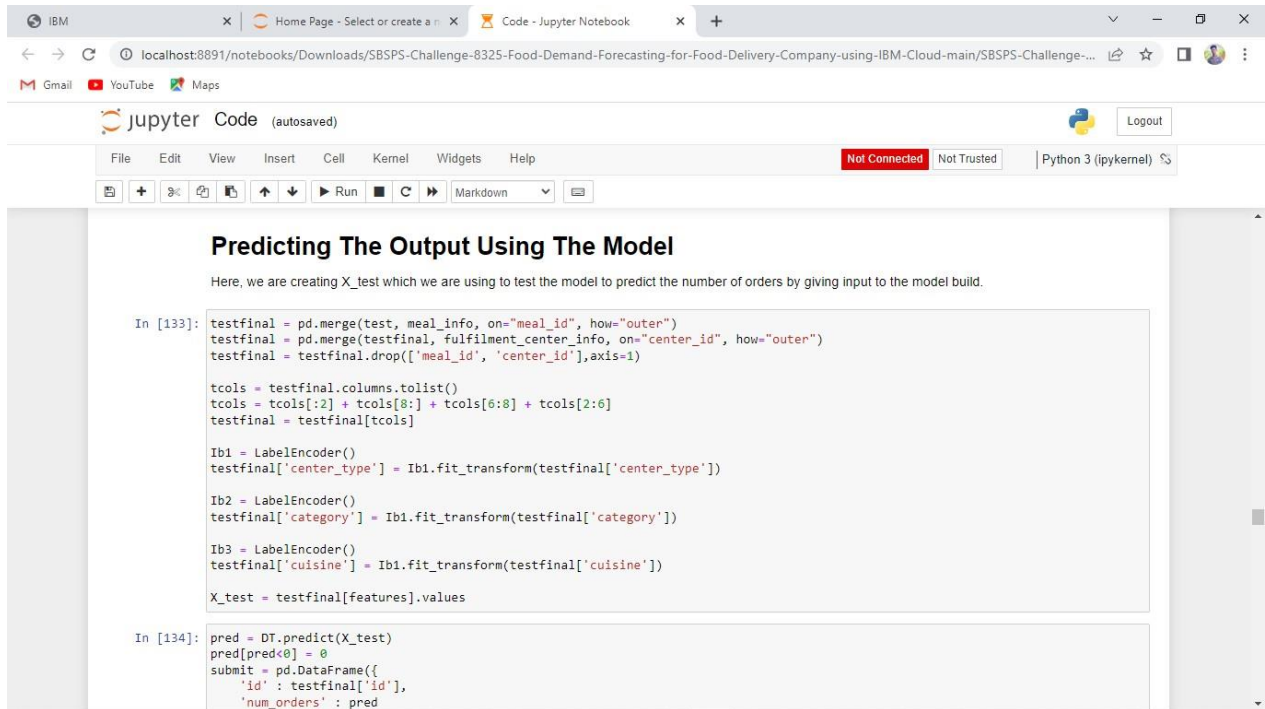
In [135]: submit.to_csv("submission.csv", index=False)

In [136]: submit.describe()

Out[136]:
```

	id	num_orders
count	3.257300e+04	32573.000000
mean	1.248476e+06	283.114244
std	1.441580e+05	367.092916
min	1.000085e+06	14.666667
25%	1.123969e+06	64.113281
50%	1.247296e+06	147.022222
75%	1.372971e+06	324.133333
max	1.499996e+06	6174.850000

## Team Member 3



The screenshot shows a Jupyter Notebook interface with a browser window at the top. The notebook is titled "Predicting The Output Using The Model". Below the title, there is a text block stating: "Here, we are creating X\_test which we are using to test the model to predict the number of orders by giving input to the model build." The notebook contains two code cells. The first cell, labeled "In [133]:", contains code for merging dataframes, dropping columns, and creating X\_test. The second cell, labeled "In [134]:", contains code for predicting the output using the model and creating a submit dataframe.

```
In [133]: testfinal = pd.merge(test, meal_info, on="meal_id", how="outer")
testfinal = pd.merge(testfinal, fulfilment_center_info, on="center_id", how="outer")
testfinal = testfinal.drop(['meal_id', 'center_id'], axis=1)

tcols = testfinal.columns.tolist()
tcols = tcols[:2] + tcols[8:] + tcols[6:8] + tcols[2:6]
testfinal = testfinal[tcols]

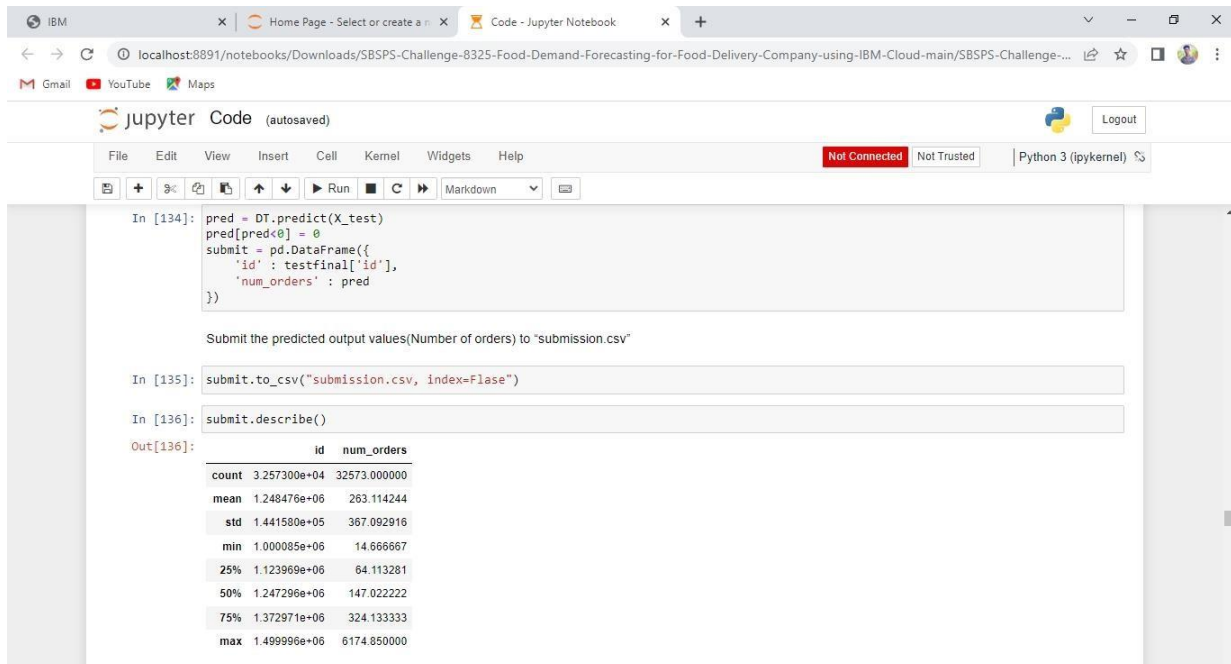
Ib1 = LabelEncoder()
testfinal['center_type'] = Ib1.fit_transform(testfinal['center_type'])

Ib2 = LabelEncoder()
testfinal['category'] = Ib1.fit_transform(testfinal['category'])

Ib3 = LabelEncoder()
testfinal['cuisine'] = Ib1.fit_transform(testfinal['cuisine'])

X_test = testfinal[features].values

In [134]: pred = DT.predict(X_test)
pred[pred<0] = 0
submit = pd.DataFrame({
    'id': testfinal['id'],
    'num_orders': pred
})
```



The screenshot shows a Jupyter Notebook interface with a browser window at the top. The notebook contains two code cells. The first cell, labeled "In [134]:", contains code for predicting the output using the model and creating a submit dataframe. The second cell, labeled "In [135]:", contains code for submitting the predicted output values to a CSV file. Below the code cells, there is a text block stating: "Submit the predicted output values(Number of orders) to 'submission.csv'". The notebook also shows the output of the submit.describe() function, which is a summary of the submit dataframe.

```
In [134]: pred = DT.predict(X_test)
pred[pred<0] = 0
submit = pd.DataFrame({
    'id': testfinal['id'],
    'num_orders': pred
})

In [135]: submit.to_csv("submission.csv", index=False)

In [136]: submit.describe()

Out[136]:
```

	id	num_orders
count	3.257300e+04	32573.000000
mean	1.248476e+06	263.114244
std	1.441580e+05	367.092916
min	1.000085e+06	14.666667
25%	1.123969e+06	64.113281
50%	1.247296e+06	147.022222
75%	1.372971e+06	324.133333
max	1.499996e+06	6174.850000