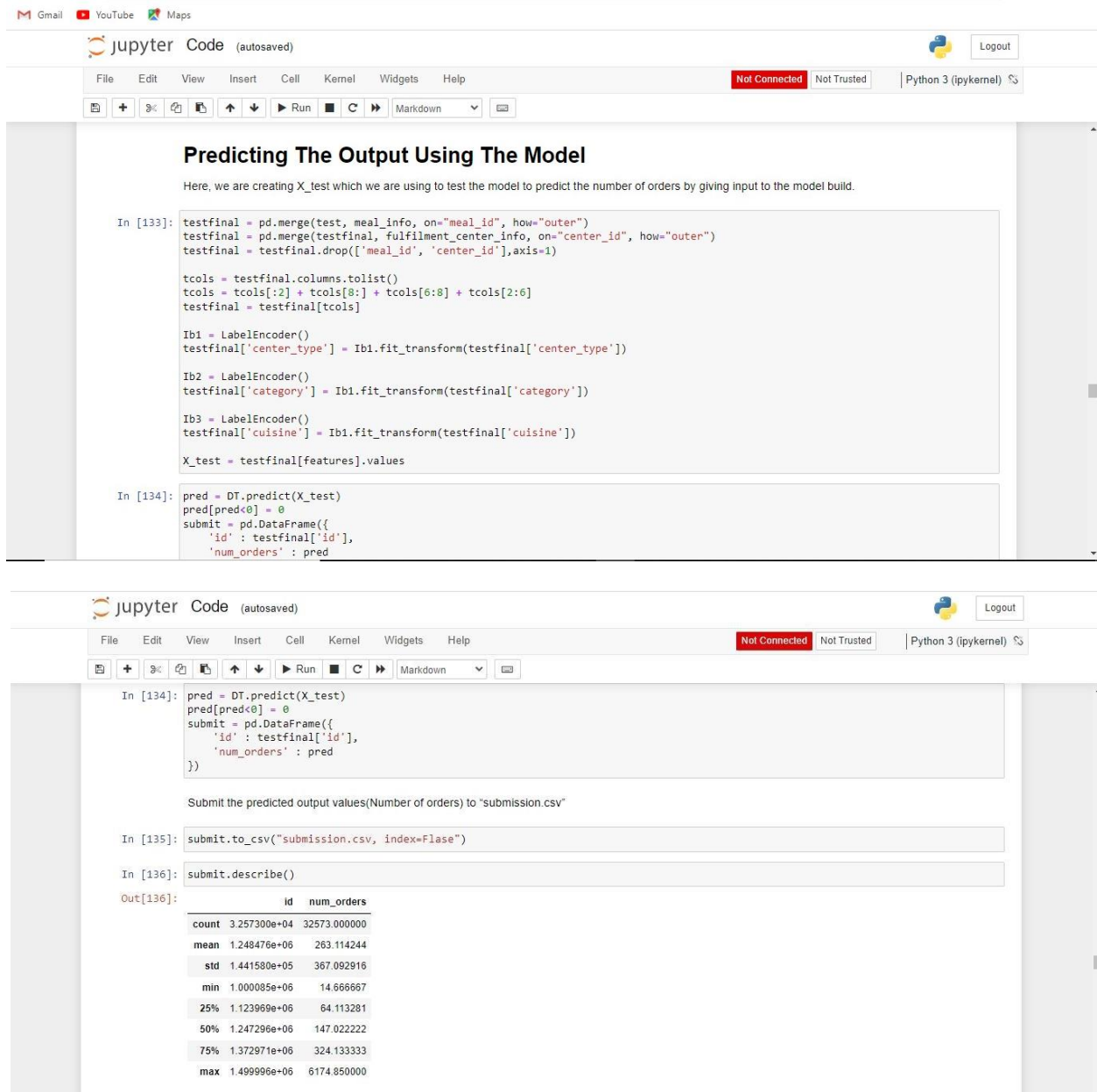


TEAM ID: PNT2022TMID32006

PROJECT NAME: DemandEst - AI powered Food DemandForecaster

Team Leader – SAI SIVA SANJAY R



The screenshot displays a Jupyter Notebook interface with a code cell containing the following Python code:

```
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
})
```

Below the code cell, there is a text instruction: "Submit the predicted output values(Number of orders) to "submission.csv"

The next code cell contains the following code:

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

The following code cell contains the following code:

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

The output of the last code cell is displayed as follows:

```
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 - GANESH M

Gmail YouTube Maps

jupyter Code (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Connected Not Trusted Python 3 (ipykernel)

Run

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
})
```

jupyter Code (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Connected Not Trusted Python 3 (ipykernel)

Run

```
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"
```

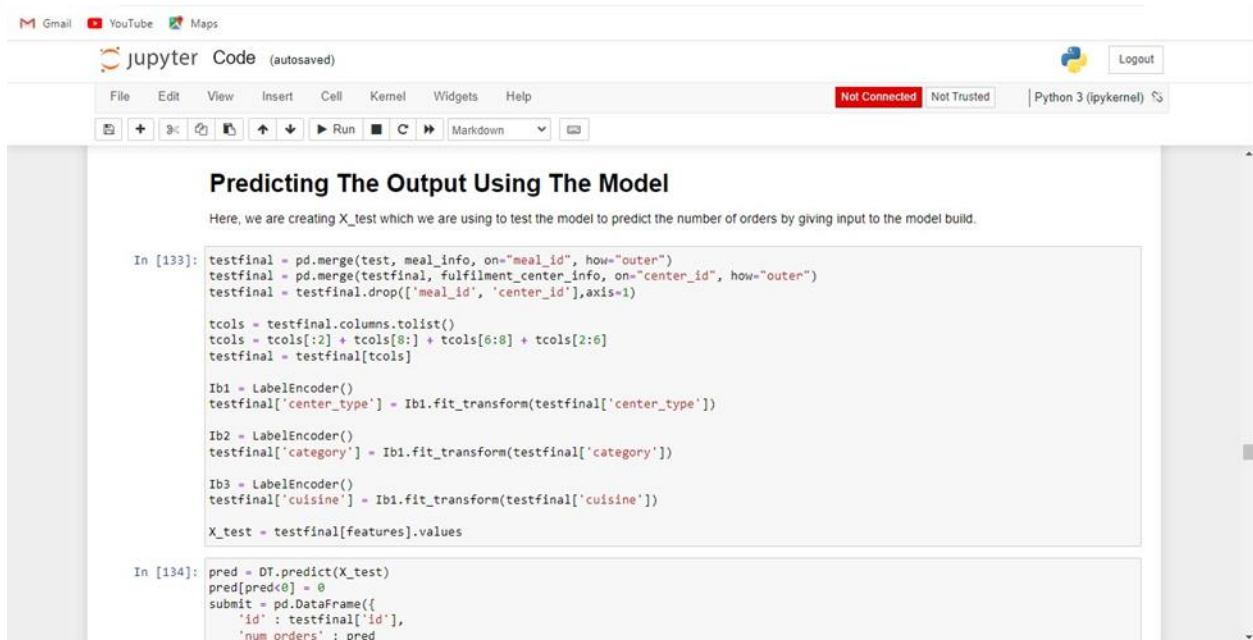
```
In [135]: submit.to_csv("submission.csv", index=False)
```

```
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 – MATHIVANAN R



The screenshot shows a Jupyter Code environment with the following code in cell [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'])

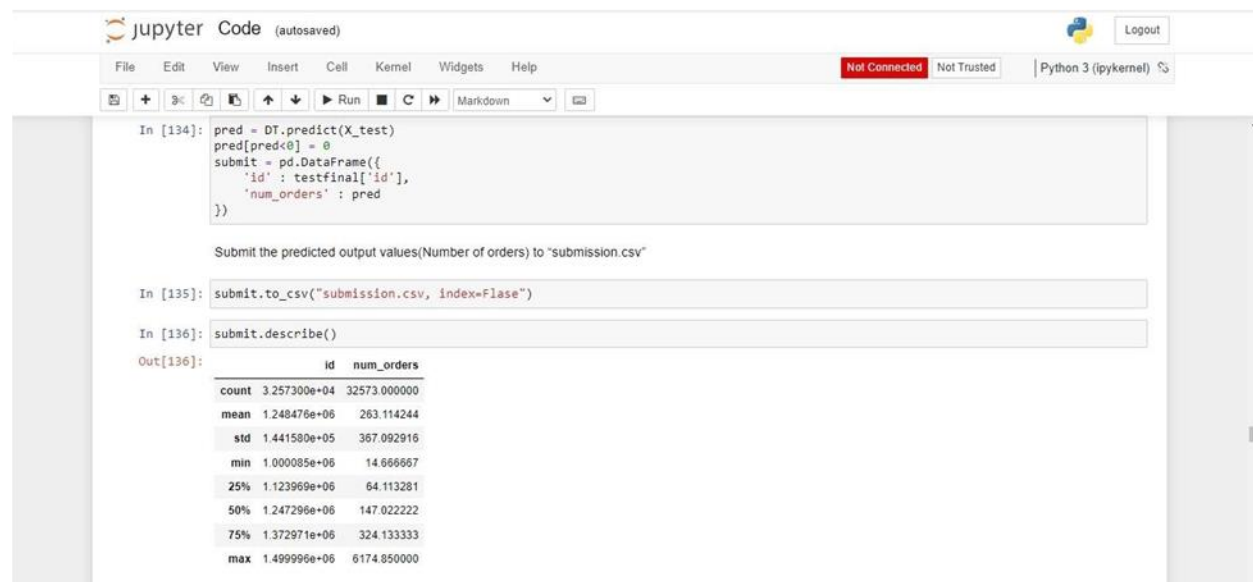
Ib2 = LabelEncoder()
testfinal['category'] = Ib1.fit_transform(testfinal['category'])

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 interface includes a top bar with links to Gmail, YouTube, and Maps, and a Jupyter Code header with a 'Logout' button. The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The status bar shows 'Not Connected', 'Not Trusted', and 'Python 3 (ipykernel)'.



The screenshot shows a Jupyter Code environment with the following code in cell [134]:

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

Below the code, a text prompt reads: "Submit the predicted output values (Number of orders) to 'submission.csv'"

Cell [135] contains the following code:

```
submit.to_csv("submission.csv", index=False)
```

Cell [136] contains the following code:

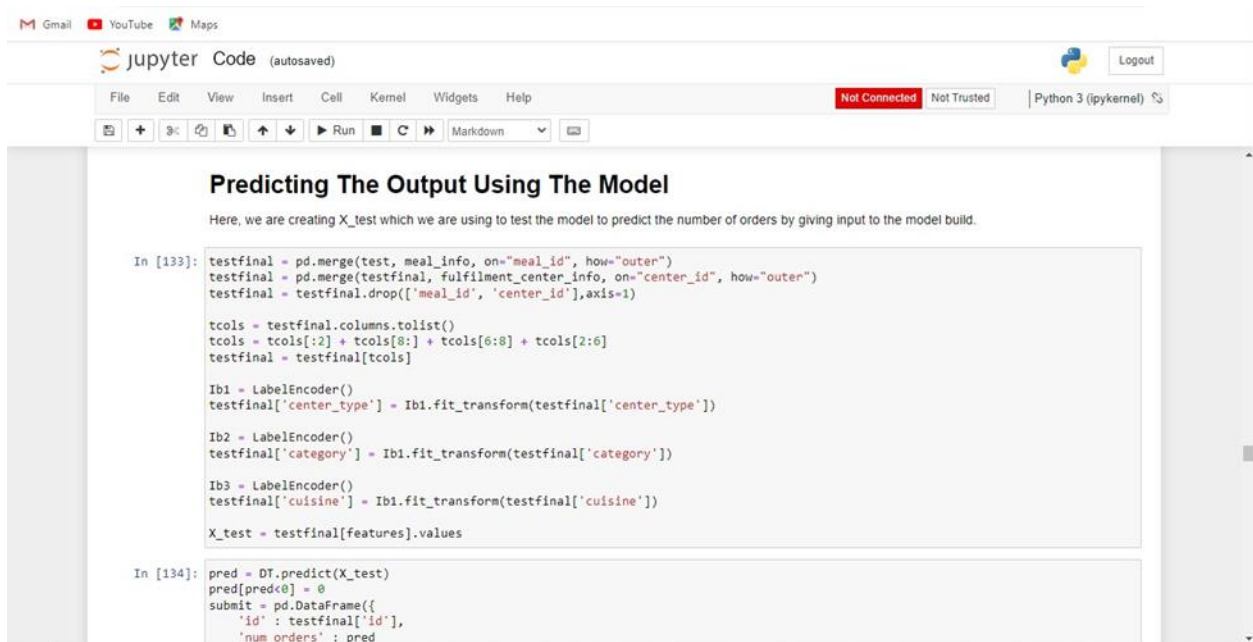
```
submit.describe()
```

The output of cell [136] is displayed as follows:

	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

The interface includes a top bar with links to Gmail, YouTube, and Maps, and a Jupyter Code header with a 'Logout' button. The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The status bar shows 'Not Connected', 'Not Trusted', and 'Python 3 (ipykernel)'.

TEAM MEMBER 3 – GAYATHRI K



The Jupyter Code interface shows a notebook titled "Predicting The Output Using The Model". The text above the code cell states: "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 code cell contains the following Python code:

```
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]

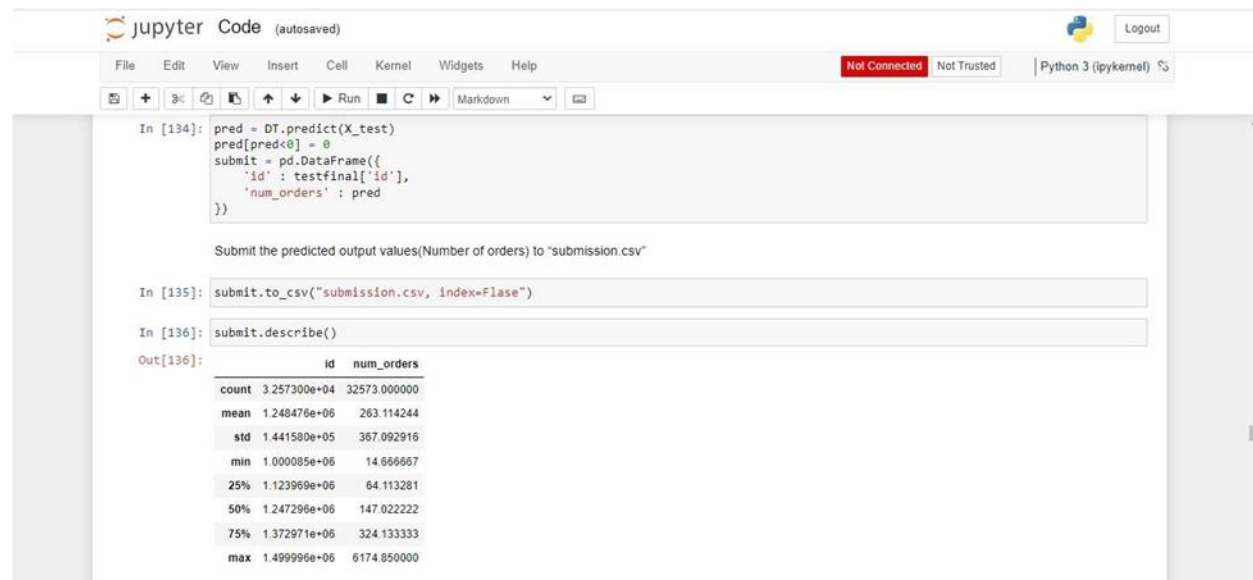
Ib1 = LabelEncoder()
testfinal['center_type'] = Ib1.fit_transform(testfinal['center_type'])

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 Jupyter Code interface shows the continuation of the notebook. The code cell contains the following Python code:

```
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"
```

The next code cell contains the following Python code:

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

The next code cell contains the following Python code:

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

The output of the last code cell is displayed as follows:

```
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	1.499999e+06	6174.850000