

EXPERIMENT 4(B)

IMPLEMENTATION OF MULTIVARIATE LINEAR REGRESSION

NAME : SREENIDHI GANACHARI

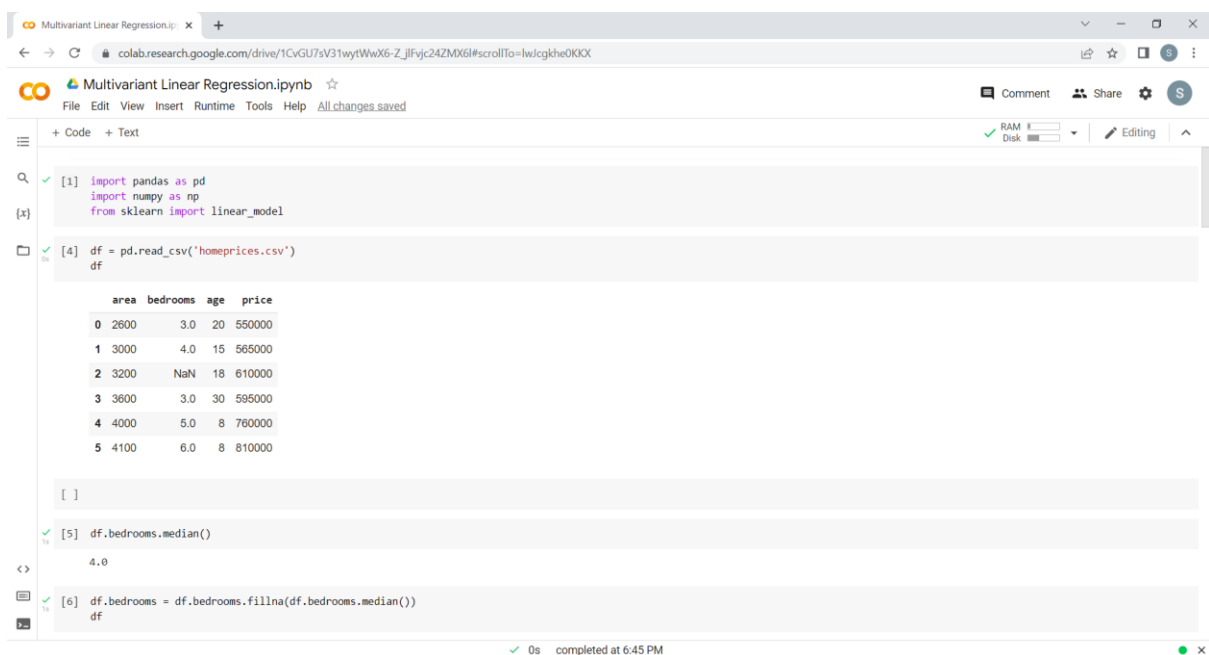
REGISTRATION NUMBER : 19BCE7230

SLOT NO : L-23+24

CODE –

```
import pandas as pd
import numpy as np
from sklearn import linear_model
df = pd.read_csv('homeprices.csv')
df
df.bedrooms.median()
df.bedrooms = df.bedrooms.fillna(df.bedrooms.median())
df
reg = linear_model.LinearRegression()
reg.fit(df.drop('price',axis='columns'),df.price)
reg.coef_
reg.intercept_
reg.predict([[3000, 3, 40]])
112.06244194*3000 + 23388.88007794*3 + -
3231.71790863*40 + 221323.00186540384
reg.predict([[2500, 4, 5]])
```

OUTPUT –



The screenshot shows a Google Colab notebook titled "Multivariate Linear Regression.ipynb". The notebook contains the following code cells:

```
[1] import pandas as pd
import numpy as np
from sklearn import linear_model
```

```
[4] df = pd.read_csv('homeprices.csv')
df
```

	area	bedrooms	age	price
0	2600	3.0	20	550000
1	3000	4.0	15	565000
2	3200	NaN	18	610000
3	3600	3.0	30	595000
4	4000	5.0	8	760000
5	4100	6.0	8	810000

```
[ ]
```

```
[5] df.bedrooms.median()
```

```
4.0
```

```
[6] df.bedrooms = df.bedrooms.fillna(df.bedrooms.median())
df
```

The notebook interface shows the code is executed successfully, with a status bar indicating "0s completed at 6:45 PM".

```
[6]
   area  bedrooms  age  price
0  2600         3.0   20  550000
1  3000         4.0   15  565000
2  3200         4.0   18  610000
3  3600         3.0   30  595000
4  4000         5.0   8   760000
5  4100         6.0   8   810000

[7] reg = linear_model.LinearRegression()
reg.fit(df.drop('price',axis='columns'),df.price)

LinearRegression()

[8] reg.coef_

array([ 112.06244194, 23388.88007794, -3231.71790863])

[9] reg.intercept_

221323.00186540396

[10] reg.predict([[3000, 3, 40]])

/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
"X does not have valid feature names, but"
array([498408.25158031])
```

```
array([ 112.06244194, 23388.88007794, -3231.71790863])

[9] reg.intercept_

221323.00186540396

[10] reg.predict([[3000, 3, 40]])

/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
"X does not have valid feature names, but"
array([498408.25158031])

[11] 112.06244194*3000 + 23388.88007794*3 + -3231.71790863*40 + 221323.00186540384

498408.25157402386

[12] reg.predict([[2500, 4, 5]])

/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
"X does not have valid feature names, but"
array([578876.03748933])
```

EXERCISE - HIRING DATASET –

CODE –

```
df = pd.read_csv('hiring.csv')
df
df.experience.median()
df.experience = df.experience.fillna(df.experience.median())
df
df.test.median()
df.test = df.test.fillna(df.test.median())
df
reg = linear_model.LinearRegression()
reg.fit(df.drop('salary',axis='columns'),df.salary)
reg.coef_
reg.intercept_
```

```
reg.predict([[2,9,6]])
2813.00813008*2 + 1333.33333333*9 + 2926.82926829*6 + 11869.91869918695
reg.predict([[12,10,10]])
2813.00813008*12 + 1333.33333333*10 + 2926.82926829*10 + 11869.91869918
695
```

OUTPUT -

The image displays two sequential screenshots of a Google Colab notebook titled "Multivariate Linear Regression.ipynb".

Top Screenshot:

- Code Cell [17]:** `df = pd.read_csv('hiring.csv')`
- Output [17]:** A DataFrame with columns: experience, test, interview, salary.

	experience	test	interview	salary
0	NaN	8.0	9	50000
1	NaN	8.0	6	45000
2	5.0	6.0	7	60000
3	2.0	10.0	10	65000
4	7.0	9.0	6	70000
5	3.0	7.0	10	62000
6	10.0	NaN	7	72000
7	11.0	7.0	8	80000
- Code Cell [18]:** `df.experience.median()`
- Output [18]:** 6.0
- Code Cell [19]:** `df.experience = df.experience.fillna(df.experience.median())`
- Output [19]:** A DataFrame where the 'experience' column for rows 0 and 1 is now filled with 6.0.

	experience	test	interview	salary
0	6.0	8.0	9	50000
1	6.0	8.0	6	45000

Bottom Screenshot:

- Code Cell [19]:** `df.experience = df.experience.fillna(df.experience.median())`
- Output [19]:** A DataFrame where the 'experience' column for rows 0 and 1 is now filled with 6.0.

	experience	test	interview	salary
0	6.0	8.0	9	50000
1	6.0	8.0	6	45000
- Code Cell [20]:** `df.test.median()`
- Output [20]:** 8.0
- Code Cell [21]:** `df.test = df.test.fillna(df.test.median())`
- Output [21]:** A DataFrame where the 'test' column for row 6 is now filled with 8.0.

	experience	test	interview	salary
0	6.0	8.0	9	50000
1	6.0	8.0	6	45000

Multivariate Linear Regression.ipynb

colab.research.google.com/drive/1CvGU7sV31wytWwX6-Z_jlFvjC24ZMX6l#scrollTo=lwlcgkhe0KXX

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM Disk

Editing

```
[21] df.test = df.test.fillna(df.test.median())
df

   experience  test  interview  salary
0         6.0    8.0         9  50000
1         6.0    8.0         6  45000
2         5.0    6.0         7  60000
3         2.0   10.0        10  65000
4         7.0    9.0         6  70000
5         3.0    7.0        10  62000
6        10.0    8.0         7  72000
7        11.0    7.0         8  80000

[22] reg = linear_model.LinearRegression()
reg.fit(df.drop('salary',axis='columns'),df.salary)

LinearRegression()

[23] reg.coef_

array([2813.00813008, 1333.33333333, 2926.82926829])

[24] reg.intercept_

0s completed at 6:45 PM
```

Multivariate Linear Regression.ipynb

colab.research.google.com/drive/1CvGU7sV31wytWwX6-Z_jlFvjC24ZMX6l#scrollTo=lwlcgkhe0KXX

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM Disk

Editing

```
[24] reg.intercept_

11869.91869918695

[25] reg.predict([[2,9,6]])

/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
"X does not have valid feature names, but"
array([47056.910569056956])

[27] 2813.00813008*2 + 1333.33333333*9 + 2926.82926829*6 + 11869.91869918695

47056.910569056956

[28] reg.predict([[12,10,10]])

/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
"X does not have valid feature names, but"
array([88227.64227642])

2813.00813008*12 + 1333.33333333*10 + 2926.82926829*10 + 11869.91869918695

88227.64227634695

0s completed at 6:45 PM
```