11. Data Science – Machine Learning – Multiple Linear Regression

Contents

11. Data Science – Machine Learning – Multiple Linear Regression

1. Multiple Linear Regression

- ✓ Multiple Linear Regression explains the relationship between a single dependent continuous variable and more than one independent variable.

2. Problem statement

- ✓ Assuming that we are planning to buy a new house and need to predict the price of a house.
- ✓ Here price depends on area (square feet), bed rooms and age of the home (in years).
- ✓ Given these prices we have to predict prices of new homes based on area, bed rooms and age
- ✓ Given these home prices find out price of a home that has,
    - ○ 3000 sqr ft area, 3 bedrooms, 40 year old
    - ○ 2500 sqr ft area, 4 bedrooms, 5 year old

3. Dataset

- ✓ homeprices1.csv is dataset we are using in this example
- ✓ This dataset contains columns as,
  - Area
  - Bedrooms
  - Age
  - Price

Area	Bedrooms	Age	price
2600	3	20	550000
3000	4	15	565000
3200		18	610000
3600	3	30	595000
4000	5	8	760000
4100	6	8	810000

4. Machine learning Terminology

4.1. Features and label

- ✓ Here area, bedrooms, age are called independent variables or features whereas price is a dependant variable

4.2. Models

- ✓ A machine learning model is simply a rule, or a formula, which predicts a label from the features.
- ✓ In this case, the model is the equation we found for the price.

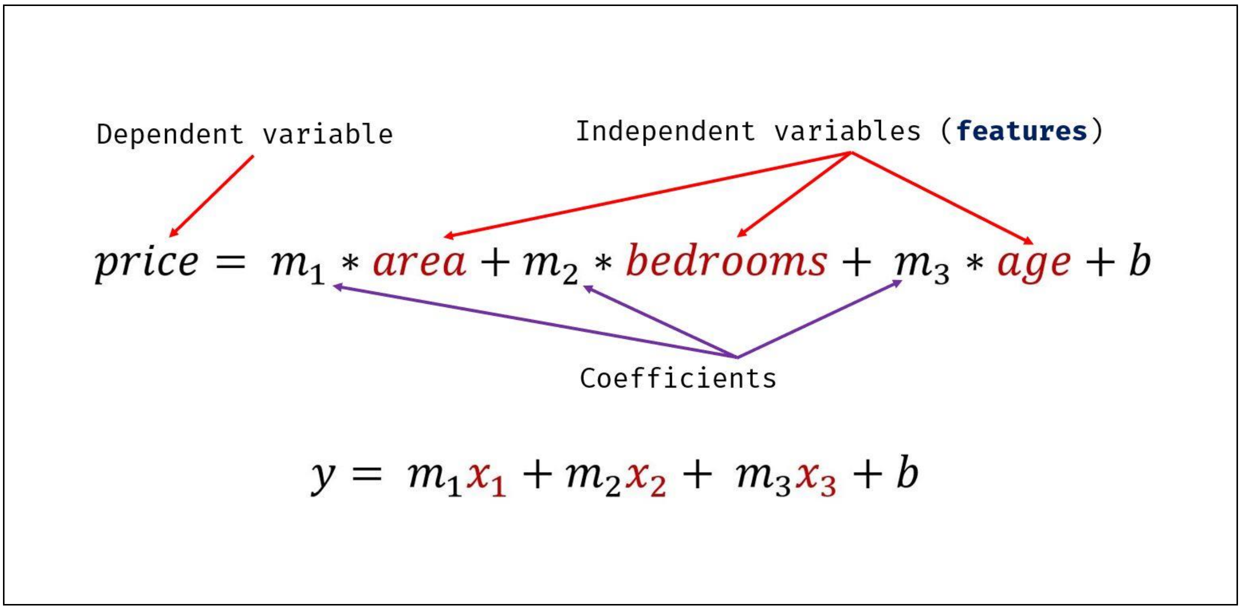
4.3. Prediction

- ✓ The prediction is simply the output of the model.

4.4. Formula

$$y = m_1 x_1 + m_2 x_2 + m_3 x_3 + b$$

$$price = m_1 * area + m_2 * bedrooms + m_3 * age + b$$



Dependent variable

Independent variables (**features**)

$$price = m_1 * area + m_2 * bedrooms + m_3 * age + b$$

Coefficients

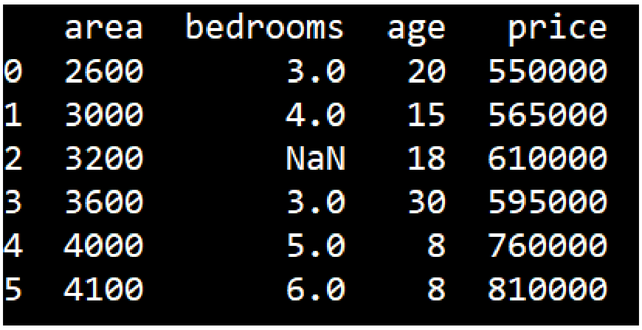
$$y = m_1 x_1 + m_2 x_2 + m_3 x_3 + b$$

Program Name

Loading house prices dataset
demo1.py

```
import pandas as pd

# Loading the dataset

df = pd.read_csv("homeprices1.csv")

print(df)
```

Output



```
   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
```

Program Name

Data pre-processing – Finding mean of bedrooms column
demo2.py

```
import pandas as pd

df = pd.read_csv("homeprices1.csv")

# Mean of the bedrooms
print("Mean of the bedrooms")
print(df.bedrooms.median())
```

Output

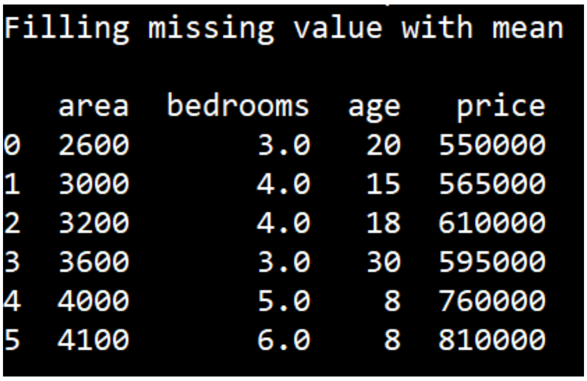
```
Mean of the bedrooms
4.0
```

Program Data pre-processing - Fill NA values with median value of a column

Name demo3.py

```
import pandas as pd

df = pd.read_csv("homeprices1.csv")

# Data loading
print("Filling missing value with mean\n")

# Data preprocessing

m = df.bedrooms.median()
df.bedrooms = df.bedrooms.fillna(m)
print(df)
```

Output



```
Filling missing value with mean

   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
```

Program Name

Model training
demo4.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

# Data loading
df = pd.read_csv("homeprices1.csv")

# Data preprocessing
m = df.bedrooms.median()
df.bedrooms = df.bedrooms.fillna(m)
a = df.drop('price', axis = 'columns')

# Model training
reg = LinearRegression()
reg.fit(a.values, df.price)
print("Model trained")
```

Output

```
Model trained
```

Program Name

Finding intercept
demo5.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

# Data loading
df=pd.read_csv("homeprices1.csv")

# Data preprocessing
m = df.bedrooms.median()
df.bedrooms = df.bedrooms.fillna(m)
a = df.drop('price', axis = 'columns')

# Model training
reg = LinearRegression()
reg.fit(a.values, df.price)

print("Intercept is:")
print(reg.intercept_)
```

Output

```
Intercept is:
221323.00186540408
```

Program Name

Finding coefficients
demo6.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

# Data loading
df = pd.read_csv("homeprices1.csv")

# Data preprocessing
m = df.bedrooms.median()
df.bedrooms = df.bedrooms.fillna(m)
a = df.drop('price', axis = 'columns')

# Model training
reg =LinearRegression()
reg.fit(a.vlaues, df.price)

print("Coefficients are:")
print(reg.coef_)
```

Output

```
Coefficients are:
 [  112.06244194 23388.88007794 -3231.71790863]
```

Program Name

price of home with 3000 sqr ft area, 3 bedrooms, 40 year old
demo7.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

# Data loading
df = pd.read_csv("homeprices1.csv")

# Data preprocessing
m = df.bedrooms.median()
df.bedrooms = df.bedrooms.fillna(m)
a = df.drop('price', axis='columns')

# Model training
reg = LinearRegression()
reg.fit(a.values, df.price)

# Prediction
print("price of home with 3000 sqr ft area, 3 bedrooms, 40 year
old")
print(reg.predict([[3000, 3, 40]]))
```

Output

```
price of home with 3000 sqr ft area, 3 bedrooms, 40 year old
[498408.25158031]
```

Program Name

price of home with 3000 sqr ft area, 3 bedrooms, 40 year old
demo8.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

# Data loading
df = pd.read_csv("homeprices1.csv")

# Data preprocessing
m = df.bedrooms.median()
df.bedrooms = df.bedrooms.fillna(m)
a = df.drop('price', axis = 'columns')

# Model training
reg = LinearRegression()
reg.fit(a.values, df.price)

# Prediction
print("price of home with 3000 sqr ft area, 3 bedrooms, 40 year old")

b = 112.06244194*3000 + 23388.88007794*3 + -3231.71790863*40 + 221323.00186540384
print(b)
```

Output

```
price of home with 3000 sqr ft area, 3 bedrooms, 40 year old
[498408.25158031]
```

Program Name

price of home with 2500 sqr ft area, 4 bedrooms, 5 year old
demo9.py

```
import pandas as pd
from sklearn.linear_model import LinearRegression

# Data loading
df = pd.read_csv("homeprices1.csv")

# Data preprocessing
m = df.bedrooms.median()
df.bedrooms = df.bedrooms.fillna(m)
a = df.drop('price', axis = 'columns')

# Model training
reg = LinearRegression()
reg.fit(a.values, df.price)

# Prediction
print("price of home with 2500 sqr ft area, 4 bedrooms, 5 year old")

print(reg.predict([[2500, 4, 5]]))
```

Output

```
price of home with 2500 sqr ft area, 4 bedrooms, 5 year old
[578876.03748933]
```