Multiple Linear Regression

Multiple Linear Regression

- Multiple Linear Regression is a type of Regression that models the relationship between a dependent variable(y) and two or more independent variables(x1,x2,x3,…,xn).
- It can be mathematically represented as:

$$\mathbf{y= a_0+a_1x_1+a_2x_2+a_3x_3+…+a_nx_n + \varepsilon}$$

- Where,

  a_0= It is the intercept of the Regression line
  a1,a2,a3,…,an= Coefficients of the model,

  x1,x2,x3,…,xn = Different independent/feature variables

Implementation of Multiple Linear Regression using Python

- Step-1: Read the data:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
df= pd.read_csv('insurance.csv')
print(df)
```

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520
5	31	female	25.740	0	no	southeast	3756.62160
6	46	female	33.440	1	no	southeast	8240.58960
7	37	female	27.740	3	no	northwest	7281.50560
8	37	male	29.830	2	no	northeast	6406.41070
9	60	female	25.840	0	no	northwest	28923.13692
10	25	male	26.220	0	no	northeast	2721.32080
11	62	female	26.290	0	yes	southeast	27808.72510
12	23	male	34.400	0	no	southwest	1826.84300
13	56	female	39.820	0	no	southeast	11090.71780
14	27	male	42.130	0	yes	southeast	39611.75770
15	19	male	24.600	1	no	southwest	1837.23700

Implementation of Multiple Linear Regression using Python

- Step-2: Pre-processing of data:

➢ Conversion of categorical data into numerical data or dummy variables:

```
def map_sex(column):
    ls=[]
    for x in column:
        if x=='female':
            ls.append(1)
        else:
            ls.append(0)
    return ls
df['sex_norm']=map_sex(df['sex'])
```

Note: Similarly we can convert ‘smoker’ column into (yes=1) and (no=0)

	age	sex	bmi	...	region	charges	sex_norm
0	19	female	27.900	...	southwest	16884.92400	1
1	18	male	33.770	...	southeast	1725.55230	0
2	28	male	33.000	...	southeast	4449.46200	0
3	33	male	22.705	...	northwest	21984.47061	0
4	32	male	28.880	...	northwest	3866.85520	0
5	31	female	25.740	...	southeast	3756.62160	1
6	46	female	33.440	...	southeast	8240.58960	1
7	37	female	27.740	...	northwest	7281.50560	1
8	37	male	29.830	...	northeast	6406.41070	0
9	60	female	25.840	...	northwest	28923.13692	1
10	25	male	26.220	...	northeast	2721.32080	0
11	62	female	26.290	...	southeast	27808.72510	1
12	23	male	34.400	...	southwest	1826.84300	0
13	56	female	39.820	...	southeast	11090.71780	1
14	27	male	42.130	...	southeast	39611.75770	0

Implementation of Multiple Linear Regression using Python

- Step-2: Pre-processing of data:

➢ Conversion of categorical data into numerical data or dummy variables:

Hot encoding is a procedure of transforming categorical variables as binary vectors.

```
region_dummy=pd.get_dummies
(df['region'])

df=pd.concat([df,region_dummy],
axis=1)
df.drop(['sex','smoker','region'],
inplace=True,axis=1)

print(df)
```

	age	bmi	children	...	northwest	southeast	southwest
0	19	27.900	0	...	0	0	1
1	18	33.770	1	...	0	1	0
2	28	33.000	3	...	0	1	0
3	33	22.705	0	...	1	0	0
4	32	28.880	0	...	1	0	0
5	31	25.740	0	...	0	1	0
6	46	33.440	1	...	0	1	0
7	37	27.740	3	...	1	0	0
8	37	29.830	2	...	0	0	0
9	60	25.840	0	...	1	0	0
10	25	26.220	0	...	0	0	0
11	62	26.290	0	...	0	1	0
12	23	34.400	0	...	0	0	1
13	56	39.820	0	...	0	1	0
14	27	42.130	0	...	0	1	0
15	19	24.600	1	...	0	0	1

Implementation of Multiple Linear Regression using Python

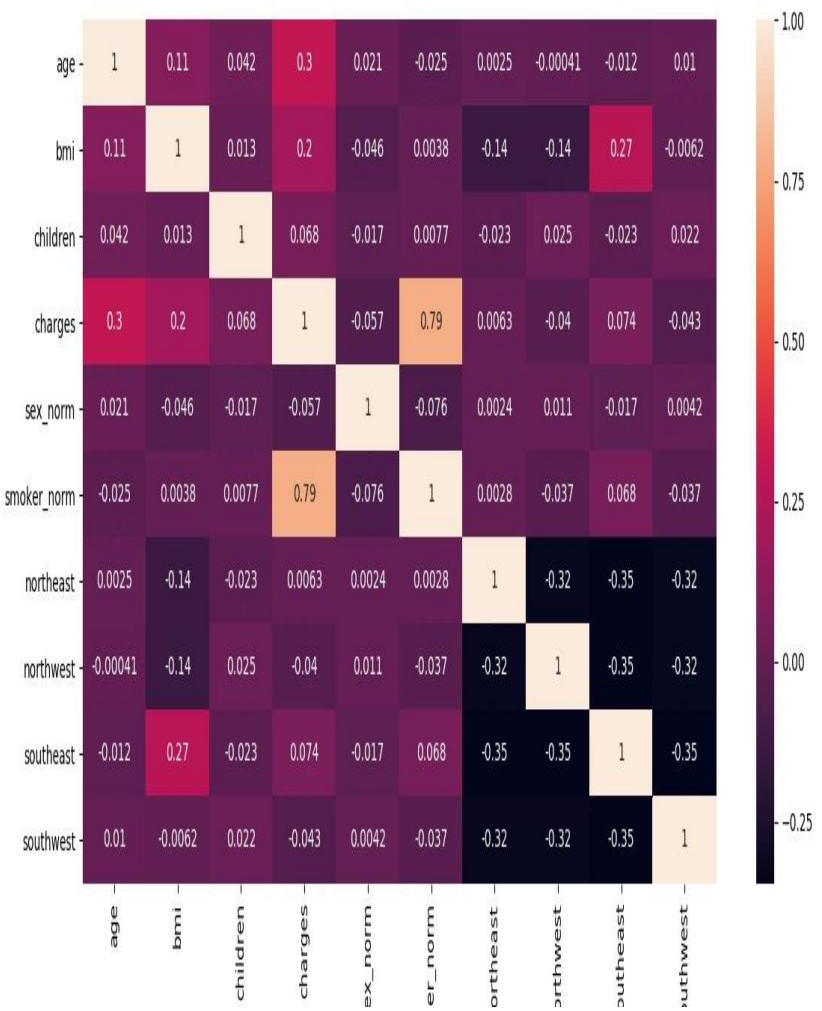
- Step-2: Pre-processing of data:

➢ Create a correlation matrix that measures the linear relationships between the variables:

```
cor_matrix = df.corr()

sns.heatmap(data=cor_matrix,
annot=True)
```

Note: An important point in selecting features for a linear regression model is to check for multi-co-linearity. If any feature pairs are strongly correlated to each other, we should not select both these features together for training the model.



Implementation of Multiple Linear Regression using Python

- Step-3: Extracting independent and dependent variables :

```
x=df[['age','bmi','children','sex_norm','smoker_norm','northwest','southeast','
   southwest']]
y=df['charges']
```

- Step-4: Splitting the dataset into training and test set:

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test= train_test_split(x, y, test_size= 1/3, rando
                                                                  m_state=0)
```

test_size=1/3: Here we will split our dataset (30 observations) into 2 parts (training set, test set) and the ratio of test set compare to dataset is 1/3 (10 observations will be put into the test set).

random_state=0: This is the seed for the random number generator. If we leave it blank or 0, then np.random is used for randomly choosing the elements for training set and test set.

Implementation of Simple Linear Regression using Python

- Step-5: Build the Regression Model:

from sklearn.linear_model import LinearRegression # Here we are importing the LinearRegression class of the linear_model library from the scikit learn.

linreg= LinearRegression()

linreg.fit(x_train, y_train)

- Step-6: Prediction of Test set Result:

test_pred= linreg.predict(x_test)

train_pred= linreg.predict(x_train)

Here we create a prediction vector test_pred, and train_pred, which contains predictions of test dataset, and training dataset respectively.

Implementation of Simple Linear Regression using Python

Step-7: Model Evaluation using R2-Score:

Model evaluation for training set:

```
r2 = r2_score(y_train, train_pred)
print("The model performance for training set:")
print("--------------------------------")
print('R2 score is {}'.format(r2))
```

The model performance for training Set:

R2 score is 0.728575273459003

Model evaluation for testing set:

```
r2 = r2_score(y_test, test_pred)
print("The model performance for testing set:")
print("--------------------------------")
print('R2 score is {}'.format(r2))
```

The model performance for testing set

R2 score is 0.787711930301308