Predictive Analysis of Insurance Charges using Multiple Linear Regression Analysis

SP JAIN School of Global Management
BDS21 – Statiscal Data Analysis

By Thao Chu – BS21DON022

CONTENTS:

1. Introduction

It is observed that health costs have been sharply increasing for years, remaining a significant point of concern for many families. This leads to insurance charges as a solution to help individuals be able to cover their medical bills.

The objective of this paper is to predict the insurance charges based on each individual's health and economic conditions. By using statistical analysis, this paper gives insights on how individual information affects one's charges; besides, with multiple linear regression models, it is possible to predict individual medical costs billed by health insurance.

The data set consists of following attributes:

1. age: the age of primary beneficiary.
2. sex: insurance contractor gender (female of male).
3. bmi: BMI - Body mass index which gives understanding about body weights related to an individual's height. (kg/m^2)
4. numChild: number of children covered by health insurance.
5. smoker: whether the individual is a smoker or not. (yes, no)
6. region: the individual's living area in the US (northeast, northwest, southeast or southwest)
7. charges: the individual medical costs billed by health insurance. (USD)

Charges is the dependent variable to be studied based on the input/independent variables which are age, sex, bmi, numChild, smoker and region.

2. Data Understanding

Note: All the figures henceforth shown in the report will be an output from Python code.

Sample Data:

The sample data is as shown below:

	age	sex	bmi	numChild	smoker	region	charges
0	19	female	27.90	0	yes	southwest	16884.92
1	18	male	33.77	1	no	southeast	1725.55
2	28	male	33.00	3	no	southeast	4449.46
3	33	male	22.71	0	no	northwest	21984.47
4	32	male	28.88	0	no	northwest	3866.86

As per the NOIR classification (Nominal, Ordinal, Interval and Ratio classification) the data in the dataset can be classified into Interval data of continuous type. This figure below show the data type of each attribute.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1338 non-null   int64
 1   sex       1338 non-null   object
 2   bmi       1338 non-null   float64
 3   numChild  1338 non-null   int64
 4   smoker    1338 non-null   object
 5   region    1338 non-null   object
 6   charges   1338 non-null   float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.3+ KB
```

Key Statistics for Data:

Before we proceeded let us find the key parameters of the data attribute, particularly, the characteristics of numerical categories.

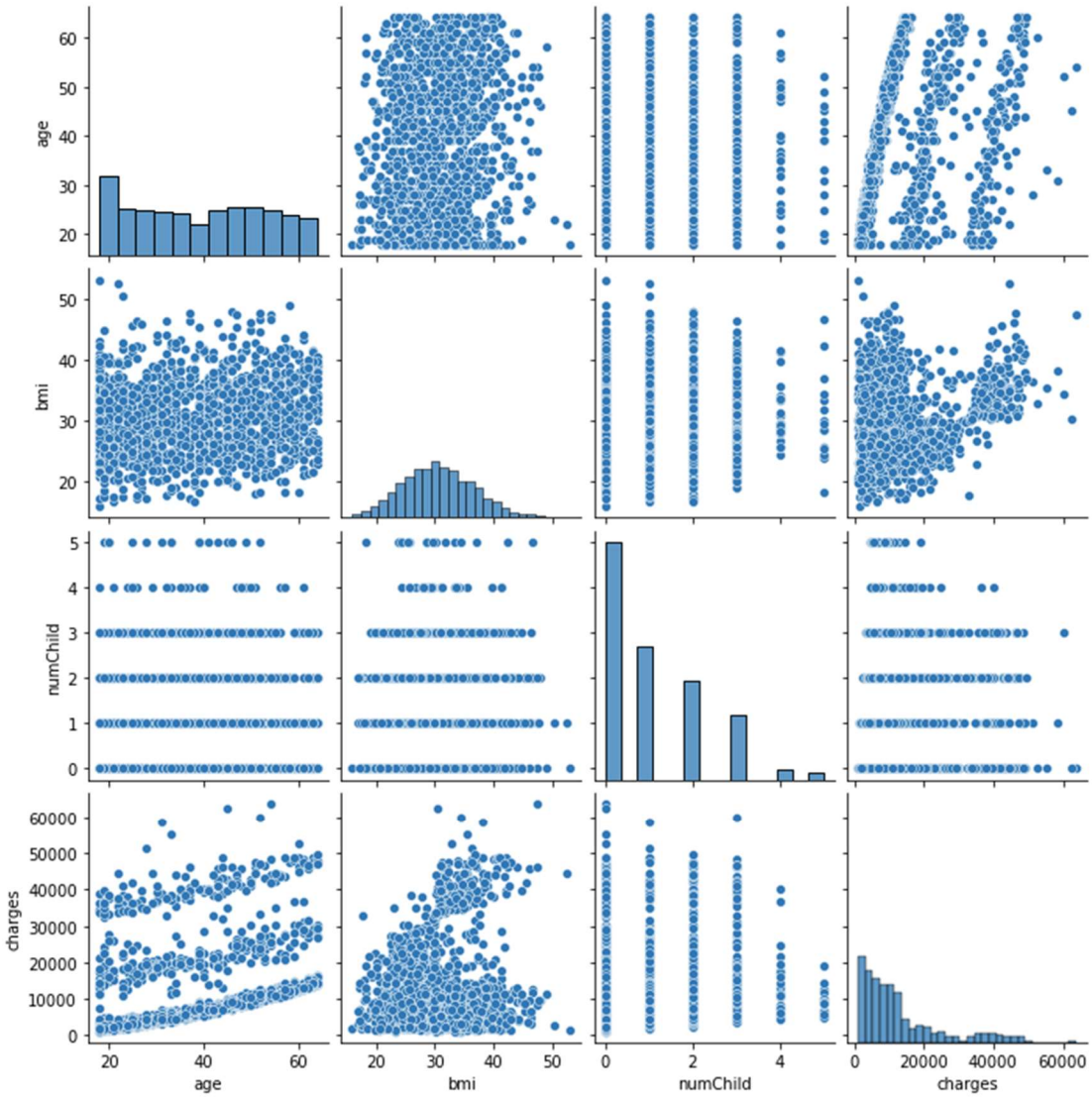
	age	bmi	numChild	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.664581	1.094918	13270.422414
std	14.049960	6.097922	1.205493	12110.011240
min	18.000000	15.960000	0.000000	1121.870000
25%	27.000000	26.297500	0.000000	4740.287500
50%	39.000000	30.400000	1.000000	9382.030000
75%	51.000000	34.695000	2.000000	16639.915000
max	64.000000	53.130000	5.000000	63770.430000

- The dataset has total of 1338 entries and 7 columns
- NULL value test was performed on the dataset. No NULL values present in the dataset.
- The data set has 4 numerical and 3 object categories.
- As the main concern is charges, according to to the description:
    - The charges ranges from about 1121.87 USD to 63770.43 USD
    - The mean value is 13270.422414 with the median is 9382.03. Hence, the dataset has a right-skewed distribution.

Statistical Analysis based on Distributions and Plots:

a. Relationship between only NUMERICAL CATEGORIES:

The figures summarize all the mutual correlations between different categories and distributions of data for each category separately, for all numerical categories.

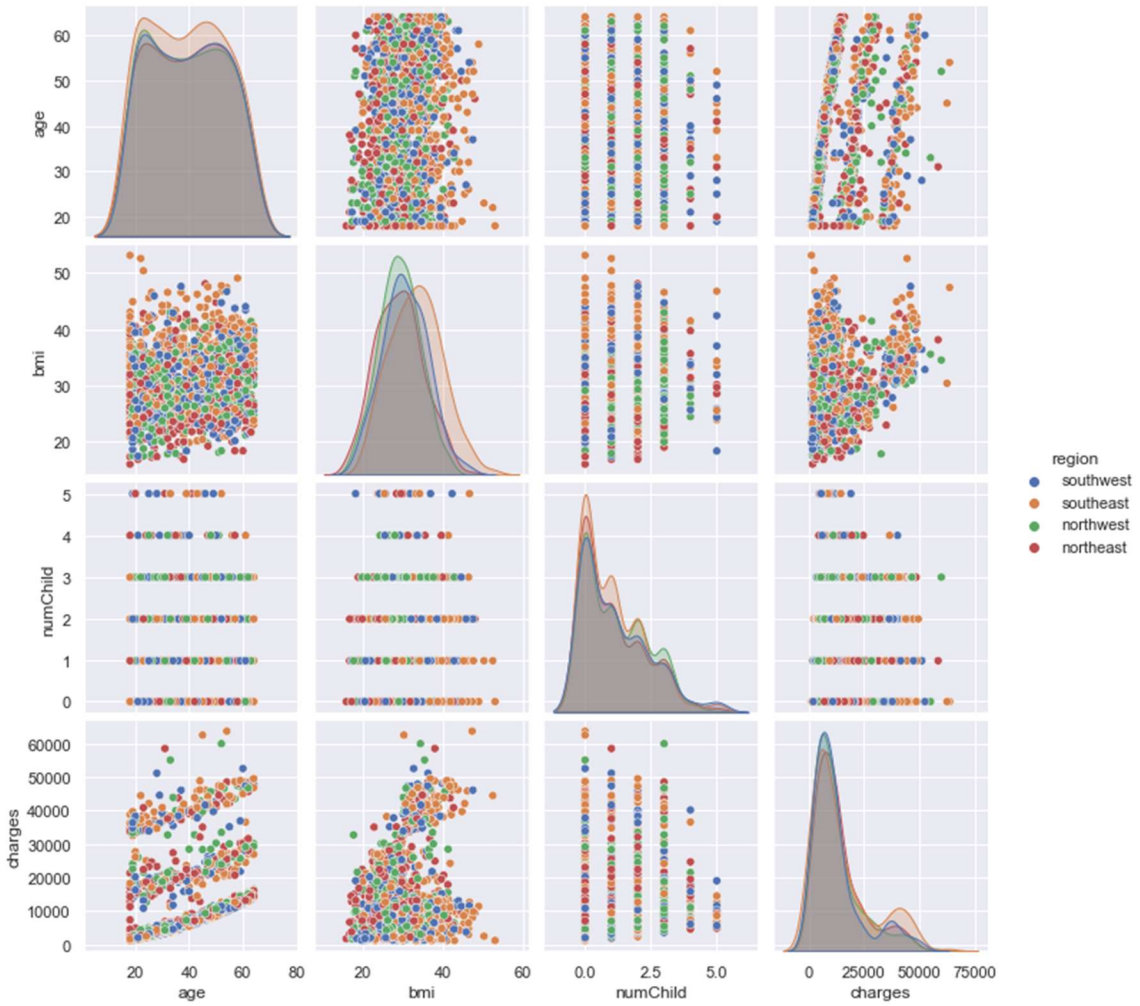


Short conclusion:

- All numerical categories have a positive relationship with 'charges' variable.
- Distribution: 'age' is uniform, 'bmi' is unimodal; while 'charges' tends to be skewed-right.

b. Relationship between only NUMERICAL CATEGORIES grouped by OBJECT CATEGORIES:

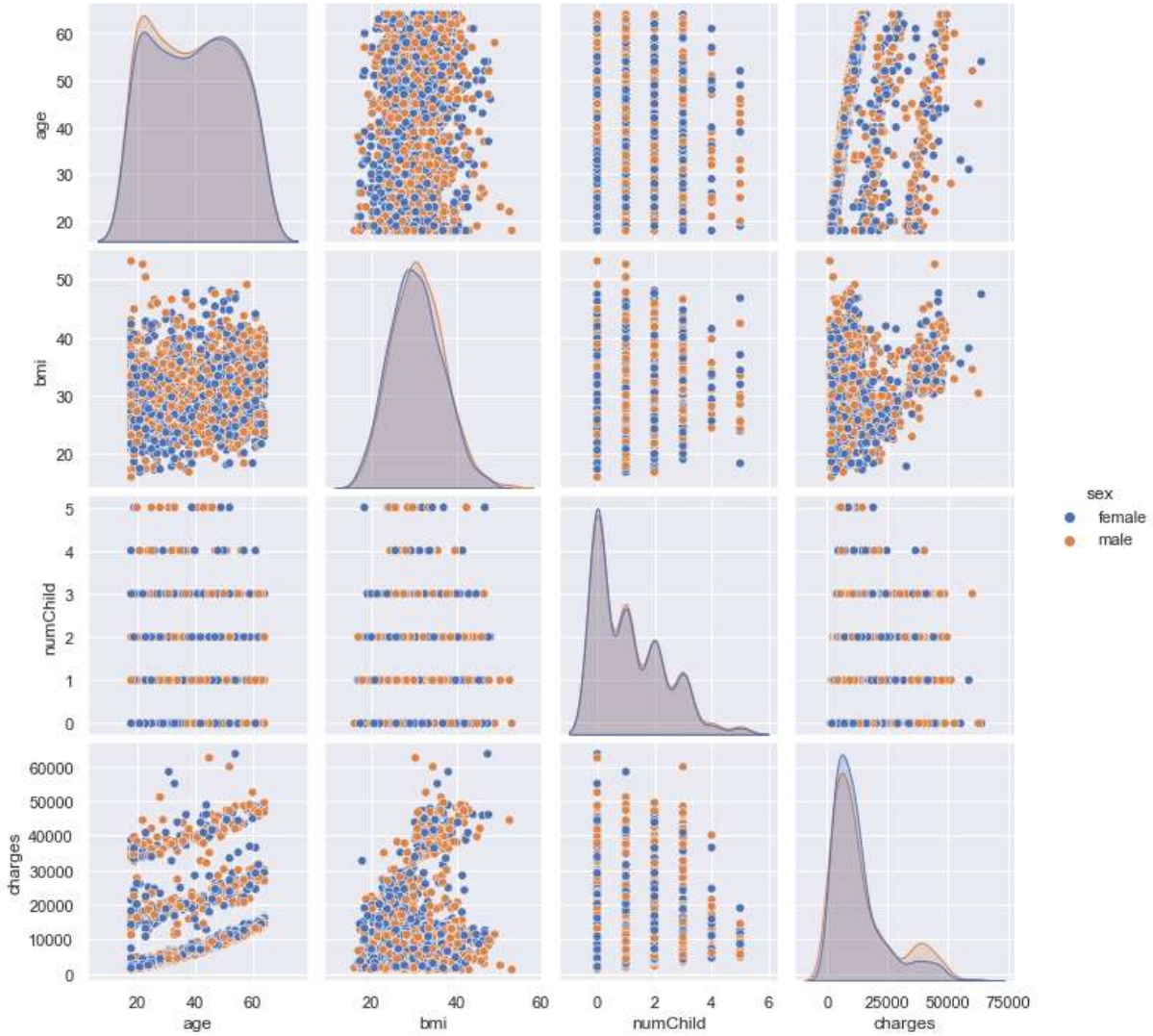
i. Grouped by 'region' variable:



Short conclusion:

- The distributions of four numerical variables grouped by four elements in region (southwest, southeast, northwest, northeast) are quite the same.
- Simarly to the short conlusion above, after being grouped, the distribution of age is uniform, bmi is unimodal; while numChild and charges are right-skewed.

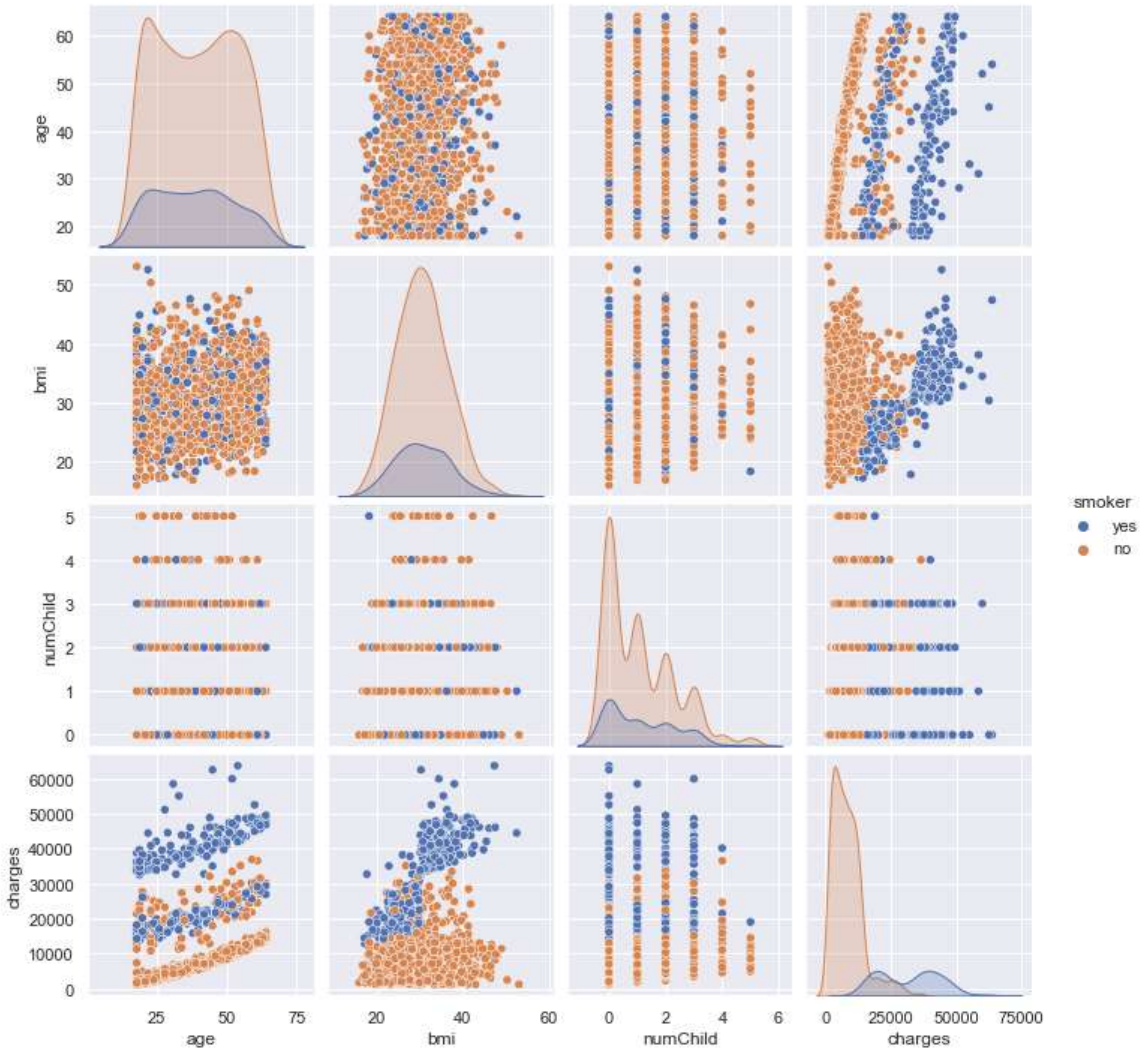
ii. *Grouped by 'sex' variable:*



Short conclusion:

- The distributions of four numerical variables grouped by two elements in sex (female, male) are quite the same.
- Simarly to the short conlusion in region.

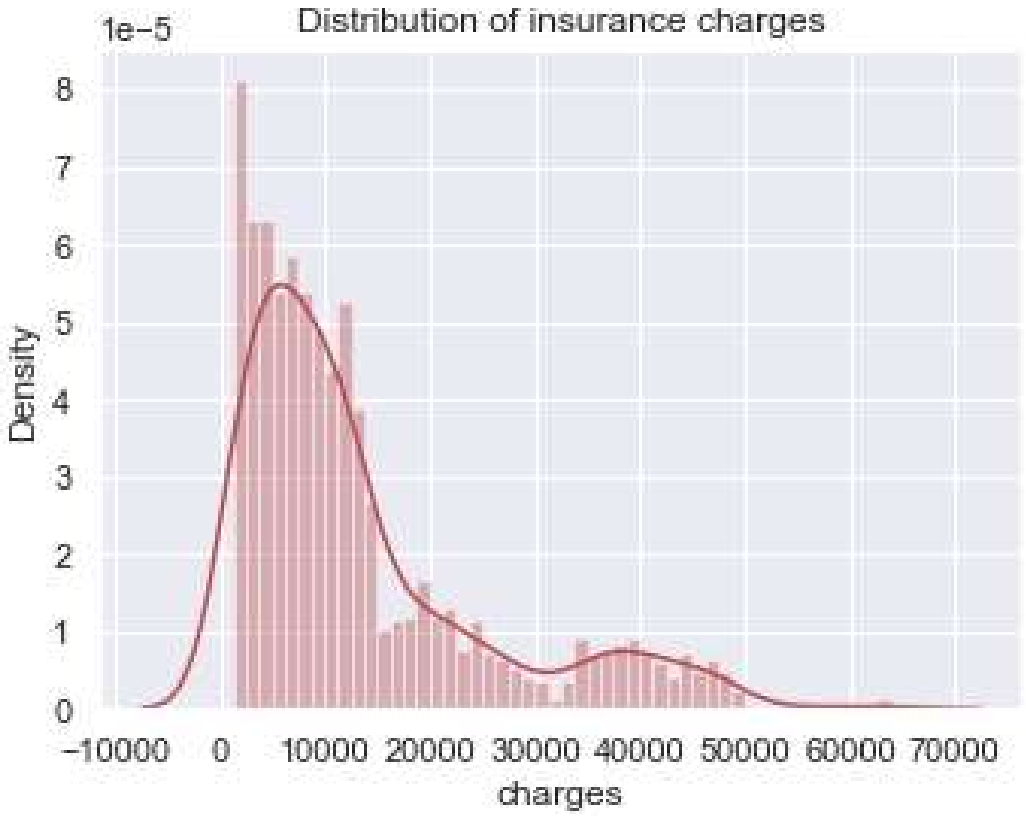
iii. Grouped by 'smoker' variable:



Short conclusion:

- The distributions of four numerical variables grouped by two elements in smoker (yes, no) are widely separated; all the distributions grouped by 'no' element are bigger than the ones grouped by 'yes' elements.
- Despite the difference in the size of distributions, after being grouped, similar to the two short conclusion above, it is observed that the distribution of age is uniform, bmi is unimodal; while numChild and charges are right-skewed.

c. Distribution of dependent variable 'charges':



Thanks to this figure, it is clearer to see the distribution of 'charges' variable is not really right-skewed but rather a mixture distribution, with the values varies from 1120 to 63500.

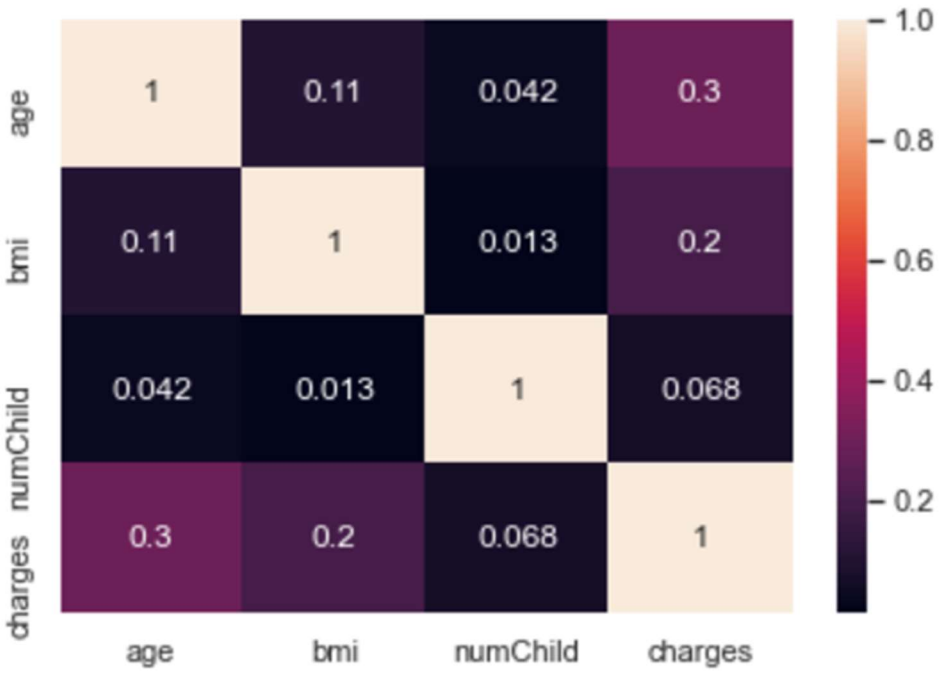
d. Correlation:

- Correlation coefficient between two random variables X and Y, usually denoted by $r(X, Y)$ or r_{XY} is a numerical measure of linear relationship between them and is defined as:

$$rXY = \frac{Cov(X,Y)}{\sigma X \sigma Y}$$

- r_{XY} provided a measure of linear relationship between X and Y.
- It is a measure of degree of relationship.

The heatmap figure below represents the mutual mutual correlation of numerical categories from our dataset.



Short conclusion:

- All the variables (both dependent and independent) are positive correlated with each others.
- The numChild category (Number of children covered by health insurance) has the lowest correlation with "charges"; meanwhile the one has highest correlation with our dependent variable is age.

3. Regression Analysis

a. Data preparation

Since there are three object variables, in orrder to build regression model, it is necessary to transform these categorical variables into numerical variables.

The below figure is the sample data after being transformed:

	age	bmi	numChild	charges	female	male	non-smoker	nicotian	northeast	northwest	southeast	southwest
0	19	27.90	0	16884.92	1	0	0	1	0	0	0	1
1	18	33.77	1	1725.55	0	1	1	0	0	0	1	0
2	28	33.00	3	4449.46	0	1	1	0	0	0	1	0
3	33	22.71	0	21984.47	0	1	1	0	0	1	0	0
4	32	28.88	0	3866.86	0	1	1	0	0	1	0	0

For easier analysis, the four elements in 'region' variables have been separated into four variables/columns (northeast, northwest, southeast, southwest); the two elements in 'sex' variable turn to

two varibles/columns (female and male); the ‘smoker’ varialbe has also been splited intro ‘non-smoker’ and ‘nicotian’ variable.
Note: the number 0 represents ‘no’ and 1 represents ‘yes’.

More information about the present data:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   age         1338 non-null   int64  
 1   bmi         1338 non-null   float64
 2   numChild    1338 non-null   int64  
 3   charges     1338 non-null   float64
 4   female      1338 non-null   uint8  
 5   male        1338 non-null   uint8  
 6   non-smoker  1338 non-null   uint8  
 7   nicotian    1338 non-null   uint8  
 8   northeast   1338 non-null   uint8  
 9   northwest   1338 non-null   uint8  
 10  southeast   1338 non-null   uint8  
 11  southwest   1338 non-null   uint8  
dtypes: float64(2), int64(2), uint8(8)
memory usage: 52.4 KB
```

b. Train and Test Split

Now the data is split into training and testing data with the following commands:

```
#train test split
X = insurance.drop(['charges'],axis=1)
y = insurance['charges']
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=101)
```

Data is slit in 80:20 ratio, 80% data is considered for training and remaining 20% data is considered for testing.

c. Fit the model

At this time, the data has already splitted into training and testing part. The training data will be taken for building regression model.

In Linear Regression, coefficients are estimated using the least squares criterion, in which we try to minimize the sum of squared residuals. Multiple Linear Regression simply includes multiple features. It takes the following form:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n \quad (1)$$

In this case, the linearity of the model (1) is defined with respect to the regression coefficients.

X variables β_1, β_2, etc, are as follows:

1. age
2. female
3. male
4. bmi
5. numChild
6. non-smoker
7. nicotian
8. northeast
9. northwest
10. southeast
11. southwest

Y variable for the model is:

- charges

```
#train model
from sklearn.linear_model import LinearRegression
lm = LinearRegression()
lm.fit(X_train, y_train)
```

```
LinearRegression()
```

```
print(lm.intercept_)
```

```
-1006.5708329582994
```

```
coeff_insurance = pd.DataFrame(lm.coef_,X.columns,columns=['Coefficient'])
coeff_insurance
```

	Coefficient
age	244.507004
bmi	364.945133
numChild	412.986434
female	-54.354987
male	54.354987
non-smoker	-11827.949437
nicotian	11827.949437
northeast	423.928053
northwest	524.813531
southeast	-489.059952
southwest	-459.681632

The form of the model is:

$$y = -1006.57 + 244.51x1 + 364.95x2 + 412.99x3 + (-54.35)x4 + 54.35x5 + (-11827.95)x6 + 11827.95x7 + 423.93x8 + 524.81x9 + (-489.06)x10 + (-459.68)x11$$

For more information about the model:

```
#fit linear regression model
model = sm.OLS(y_train, X_train).fit()

#view model summary
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                charges   R-squared:                       0.748
Model:                            OLS   Adj. R-squared:                  0.746
Method:                 Least Squares   F-statistic:                     392.7
Date:                Mon, 16 May 2022   Prob (F-statistic):          7.36e-311
Time:                        02:22:58   Log-Likelihood:                -10846.
No. Observations:                1070   AIC:                         2.171e+04
Df Residuals:                    1061   BIC:                         2.175e+04
Df Model:                           8
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
age          244.5070     13.556     18.037      0.000     217.908     271.106
bmi          364.9451     32.286     11.304      0.000     301.594     428.296
numChild     412.9864    156.390      2.641      0.008     106.117     719.856
female      -456.9833    477.128     -0.958      0.338   -1393.205     479.238
male        -348.2733    481.583     -0.723      0.470   -1293.237     596.690
non-smoker -1.223e+04    475.345    -25.730      0.000   -1.32e+04   -1.13e+04
nicotian    1.143e+04    518.903     22.018      0.000    1.04e+04    1.24e+04
northeast    222.6139    377.699      0.589      0.556    -518.507     963.735
northwest    323.4994    378.263      0.855      0.393    -418.730    1065.728
southeast   -690.3741    431.335     -1.601      0.110   -1536.741     155.993
southwest   -660.9958    397.253     -1.664      0.096   -1440.486     118.494
==============================================================================
Omnibus:                      244.119   Durbin-Watson:                   1.987
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              568.618
Skew:                           1.232   Prob(JB):                    3.36e-124
Kurtosis:                       5.585   Cond. No.                     9.29e+17
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
[2] The smallest eigenvalue is 3.25e-30. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

4. Prediction

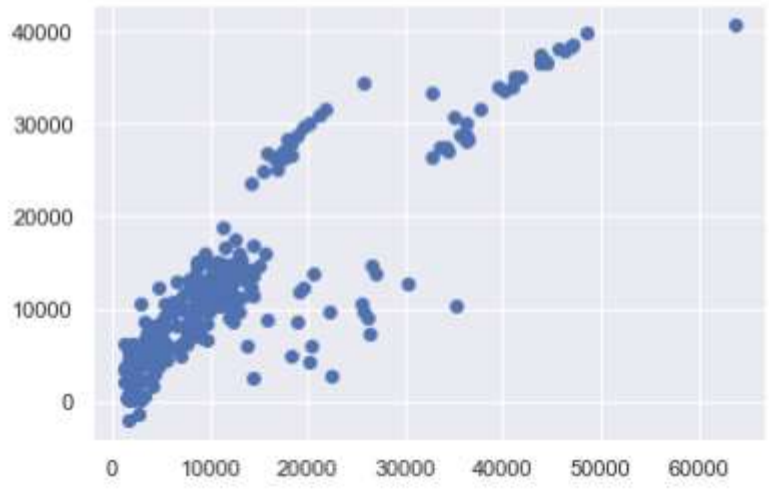
Final part of this short case study is to test the trained model for predicting a new “charges” values (by using prepared X_test data which is obtained by *train_test_split* function)

```
predictions = lm.predict(X_test)
print("Predicted medical costs values:", predictions)
```

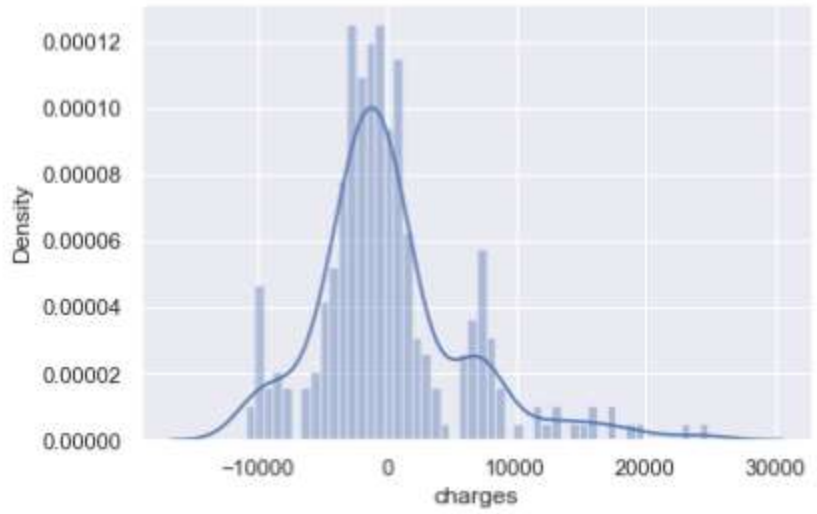
```
Predicted medical costs values: [10869.23256848  2931.96327822 10551.63054316 27006.93011892
  6217.38521797   434.97746879 15347.1473205   3968.92927222
 11161.89365725  6826.26206393  8229.39256378 10752.09550751
  6493.13536567 12751.61385514 30972.89816484 36900.79776344
 10489.55616139  2871.24426768 15159.83421346 13820.49180861
 30078.99439532  7351.43647149 12740.6948772   5914.03251769
   471.47198214 25134.94429351 12192.70350286 27390.28027849
 37879.68663708  4478.67563868  6058.7987007   7238.32218498
  3553.95296524  2545.00043836  5715.88194485  5964.42608933
   335.79840633 12338.96825697 13784.36937444 11014.48748977
 10292.84436758  9627.90806793 14866.02395458 10324.87637688
  6313.63068629  6168.39520549  8245.83184732  2131.34716781
 12718.91021743  2212.99019054 -1450.87803884  2464.22080572
  7271.00250697  8672.26730131  4310.76139103 14274.19615822
 31619.01297891 11104.23189992 10565.87806952  3799.43918253
   331.48743678 33559.38591051 36506.6445494  13274.28833851
 11290.53416161 16525.16523997 13119.65808778 28717.75546334
 17513.69914952 11733.46057744 28127.31791359 30060.86767926
  3562.53232274  3784.53615249  5413.65886718 10952.01058971
 13781.96043391 39799.21878341  5326.44961365  2827.30735081
 36915.43921344 16080.72264977 35085.86722865  9113.27173933
   647.5253038   6335.57103896 12827.39006778 -2130.58681943
 26449.63549623 38144.63909435 14704.01790522  9578.98513034
  6859.11959585  8935.28588081 12886.65602442  3696.96179036
  7615.08143446 36632.07419845  7403.09415284 26619.27106634
 12403.21645608 26711.89805409  6270.99870629 10481.84037486
 30659.72040125 38399.4630463  10778.62464901  9997.13849551
 24890.10618491  9719.14301038  4359.60327427 10900.37990692
 26750.75920422 28419.03210121 11529.23272701  2048.30994826
  2277.93168792  5015.25776873  4311.3917345  13371.30738551
```

Note: *the figure above has not included all the value.*

This figure below is a graphical comparison of expected values of our analysis (y_test) and predicted values (predictions) of our trained model:



Error distribution graph of our predictions:



There is also a need in calculating the mean absolute error (MAE) and mean squared error (MSE) for our predictions. These measures will represent quality parameters of our model (achieved prediction accuracy).

```
from sklearn import metrics
print('MAE = ',metrics.mean_absolute_error(y_test, predictions))
print('MSE = ',metrics.mean_squared_error(y_test, predictions))
```

```
MAE =  4036.865772278072
MSE =  33748606.755860314
```

5. Conclusion

This paper shows the ways of using statistical analysis and regression models can help us to achieve better prediction performances. In this paper, insurance charges is one of the popular real-life application of statistics. This case study demonstrates the power of using statistics understanding in solving real-life problems.

6. Reference

- https://www.kaggle.com/code/sudhirnl7/linear-regression-tutorial/notebook
- https://www.kaggle.com/code/sudhirnl7/linear-regression-tutorial/data