Data Analysis of Claim of Health Insurance

Life of an Individual is generally calm and peaceful until any health issues arises without any indication since some of the health issues cannot be predicted earlier. These health issues can take up most of the saving in an instant, which is probably saved for some needs such as owning a house, or a motor car or a bank loan, which can postponed if the family is not financially strong enough. But these problems can be sorted out later but medical obligations which need immediate cash flows can affect the financial goals of the family such as the education, marriage of children and retirement plans. The only solution to overcome all these problems is the health insurance which will help in maintenance of good health of an individual and avoids financial crisis. Health Insurance is an insurance that includes the expenses of medication, surgery and other health related problems of an individual, family or a group of people. In health Insurance, an individual purchase health care coverage by paying fees in advance referred to as premium so that he/she won't have to face financial crisis when an incident happens suddenly.

In this project, we have been given a dataset ,which has been taken from Kaggle, of health of individuals of different age groups and we will analyze the key factors affecting people's claim such as age, smoking behaviour, diabetes, regular exercises, etc. We will also use exploratory data analysis and visualization to throw light on the key factors.

Downloading the Dataset

Firstly, we will download the dataset from the kaggle.

```
!pip install jovian opendatasets --upgrade --quiet
```

Let's begin by downloading the data, and listing the files within the dataset.

```
dataset_url = 'https://www.kaggle.com/datasets/sureshgupta/health-insurance-data-set'
```

```
import opendatasets as od
od.download(dataset_url)
```

```
Please provide your Kaggle credentials to download this dataset. Learn more:
http://bit.ly/kaggle-creds
Your Kaggle username: elijahxx7
Your Kaggle Key: ········
Downloading health-insurance-data-set.zip to ./health-insurance-data-set

100%|██████████| 213k/213k [00:00<00:00, 76.9MB/s]
```

The dataset has been downloaded and extracted.

```
data_dir = './health-insurance-data-set'
```

```
import os
os.listdir(data_dir)
```

```
['1651277648862_healthinsurance.csv']
```

Let us save and upload our work to Jovian before continuing.

```
project_name = "data-analysis-of-claim-of-health-insurance"
```

```
!pip install jovian --upgrade -q
```

```
import jovian
```

```
jovian.commit(project=project_name)
```

```
[jovian] Updating notebook "saxena-arpit2001/data-analysis-of-claim-of-health-
insurance" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/saxena-arpit2001/data-analysis-of-
claim-of-health-insurance
```

```
'https://jovian.ai/saxena-arpit2001/data-analysis-of-claim-of-health-insurance'
```

Data Preparation and Cleaning

Firstly, we will download the dataset from Kaggle.

```
#Import Libraries

import pandas as pd
import numpy as np
```

```
#Upload Dataset and rename all columns

df = pd.read_csv(data_dir+ '/1651277648862_healthinsurance.csv')
df.rename(columns={'age':'Age', 'sex':'Sex','weight':'Weight', 'bmi':'BMI','hereditary_
                   'smoker':'Smokers', 'city':'City','bloodpressure':'Blood Pressure',
df.drop(columns=['Blood Pressure'], inplace=True)
df
```

	Age	Sex	Weight	BMI	Hereditary Diseases	Number of Dependents	Smokers	City	Diabetes	Regular Exercises	Job Pr
0	60.0	male	64	24.3	NoDisease	1	0	NewYork	0	0	A
1	49.0	female	75	22.6	NoDisease	1	0	Boston	1	1	Engi
2	32.0	female	64	17.8	Epilepsy	2	1	Phildelphia	1	1	Academi
3	61.0	female	53	36.4	NoDisease	1	1	Pittsburg	1	0	(
4	19.0	female	50	20.6	NoDisease	0	0	Buffalo	1	0	HomeMa

	Age	Sex	Weight	BMI	Hereditary Diseases	Number of Dependents	Smokers	City	Diabetes	Regular Exercises	Job Pr
...	...	...	...	...	...	...	...	...	...	...	
14995	39.0	male	49	28.3	NoDisease	1	1	Florence	1	0	FilmM
14996	39.0	male	74	29.6	NoDisease	4	0	Miami	1	0	Stu
14997	20.0	male	62	33.3	NoDisease	0	0	Tampa	1	0	FashionDesi
14998	52.0	male	88	36.7	NoDisease	0	0	PanamaCity	1	0	Fa
14999	52.0	male	57	26.4	NoDisease	3	0	Kingsport	1	0	Man

15000 rows × 12 columns

```python
#Access and Replace Boolean Values

df.loc[df["Smokers"] == 0, "Smokers"] = "Non-Smoker"
df.loc[df["Smokers"] == 1, "Smokers"] = "Smoker"
df.loc[df["Diabetes"] == 0, "Diabetes"] = "Non-Diabetic"
df.loc[df["Diabetes"] == 1,"Diabetes"] = "Diabetic"
df.loc[df["Regular Exercises"] == 0,"Regular Exercises"] = "Non-Regular"
df.loc[df["Regular Exercises"] == 1,"Regular Exercises"] = "Regular"
df
```

	Age	Sex	Weight	BMI	Hereditary Diseases	Number of Dependents	Smokers	City	Diabetes	Regular Exercises	Job Pr
0	60.0	male	64	24.3	NoDisease	1	Non-Smoker	NewYork	Non-Diabetic	Non-Regular	A
1	49.0	female	75	22.6	NoDisease	1	Non-Smoker	Boston	Diabetic	Regular	Engi
2	32.0	female	64	17.8	Epilepsy	2	Smoker	Phildelphia	Diabetic	Regular	Academi
3	61.0	female	53	36.4	NoDisease	1	Smoker	Pittsburg	Diabetic	Non-Regular	(
4	19.0	female	50	20.6	NoDisease	0	Non-Smoker	Buffalo	Diabetic	Non-Regular	HomeMa
...	...	...	...	...	...	...	...	...	...	...	
14995	39.0	male	49	28.3	NoDisease	1	Smoker	Florence	Diabetic	Non-Regular	FilmM
14996	39.0	male	74	29.6	NoDisease	4	Non-Smoker	Miami	Diabetic	Non-Regular	Stu
14997	20.0	male	62	33.3	NoDisease	0	Non-Smoker	Tampa	Diabetic	Non-Regular	FashionDesi
14998	52.0	male	88	36.7	NoDisease	0	Non-Smoker	PanamaCity	Diabetic	Non-Regular	Fa
14999	52.0	male	57	26.4	NoDisease	3	Non-Smoker	Kingsport	Diabetic	Non-Regular	Man

15000 rows × 12 columns

```python
#Check Missing Rows

df.isna().sum()
```

```
Age                   396
Sex                     0
Weight                  0
BMI                   956
Hereditary Diseases     0
Number of Dependents    0
Smokers                 0
City                    0
Diabetes                0
Regular Exercises       0
Job Profile             0
Claim                   0
dtype: int64
```

```
#Drop Missing Rows

df = df.dropna()
df
```

	Age	Sex	Weight	BMI	Hereditary Diseases	Number of Dependents	Smokers	City	Diabetes	Regular Exercises	Job Pr
0	60.0	male	64	24.3	NoDisease	1	Non-Smoker	NewYork	Non-Diabetic	Non-Regular	A
1	49.0	female	75	22.6	NoDisease	1	Non-Smoker	Boston	Diabetic	Regular	Engi
2	32.0	female	64	17.8	Epilepsy	2	Smoker	Phildelphia	Diabetic	Regular	Academi
3	61.0	female	53	36.4	NoDisease	1	Smoker	Pittsburg	Diabetic	Non-Regular	(
4	19.0	female	50	20.6	NoDisease	0	Non-Smoker	Buffalo	Diabetic	Non-Regular	HomeMa
...	...	...	...	...	...	...	...	...	...	...	
14995	39.0	male	49	28.3	NoDisease	1	Smoker	Florence	Diabetic	Non-Regular	FilmM
14996	39.0	male	74	29.6	NoDisease	4	Non-Smoker	Miami	Diabetic	Non-Regular	Stu
14997	20.0	male	62	33.3	NoDisease	0	Non-Smoker	Tampa	Diabetic	Non-Regular	FashionDesi
14998	52.0	male	88	36.7	NoDisease	0	Non-Smoker	PanamaCity	Diabetic	Non-Regular	Fa
14999	52.0	male	57	26.4	NoDisease	3	Non-Smoker	Kingsport	Diabetic	Non-Regular	Man

13648 rows × 12 columns

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 13648 entries, 0 to 14999
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype
```

```
---  ------                --------------  -----
 0   Age                   13648 non-null  float64
 1   Sex                   13648 non-null  object
 2   Weight                13648 non-null  int64
 3   BMI                   13648 non-null  float64
 4   Hereditary Diseases   13648 non-null  object
 5   Number of Dependents  13648 non-null  int64
 6   Smokers               13648 non-null  object
 7   City                  13648 non-null  object
 8   Diabetes              13648 non-null  object
 9   Regular Exercises     13648 non-null  object
 10  Job Profile           13648 non-null  object
 11  Claim                 13648 non-null  float64
dtypes: float64(3), int64(2), object(7)
memory usage: 1.4+ MB
```

```
df.shape
```

```
(13648, 12)
```

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Updating notebook "saxena-arpit2001/data-analysis-of-claim-of-health-insurance" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/saxena-arpit2001/data-analysis-of-claim-of-health-insurance
```

```
'https://jovian.ai/saxena-arpit2001/data-analysis-of-claim-of-health-insurance'
```

Exploratory Analysis and Visualization

Now, we will explore the dataset using basic statistics and then visualize them.

1) Explore Dataframes using Descriptive Statistics

```
df.describe()
```

	Age	Weight	BMI	Number of Dependents	Claim
count	13648.000000	13648.000000	13648.000000	13648.000000	13648.000000
mean	39.586533	64.689478	30.287295	1.106462	13416.465050
std	14.040870	13.655520	6.133622	1.209568	12080.022325
min	18.000000	34.000000	16.000000	0.000000	1121.900000
25%	27.000000	54.000000	25.700000	0.000000	4889.000000

	Age	Weight	BMI	Number of Dependents	Claim
50%	40.000000	63.000000	29.400000	1.000000	9715.800000
75%	52.000000	75.000000	34.400000	2.000000	16450.900000
max	64.000000	95.000000	53.100000	5.000000	63770.400000

Here, it can be observed that :

- 13648 people come for the insurance having age between 18 to 64.
- Some People have less health related problems which may be due to less age, normal since they have less claim.
- Some People have more health related problems which may be due to more age, obese, since they have high claim.
- The people who have insurance plans ranges from 0 to 5.

Also, it can also be observed the value of mean, standard deviation, Interquartile range and median of each feature.

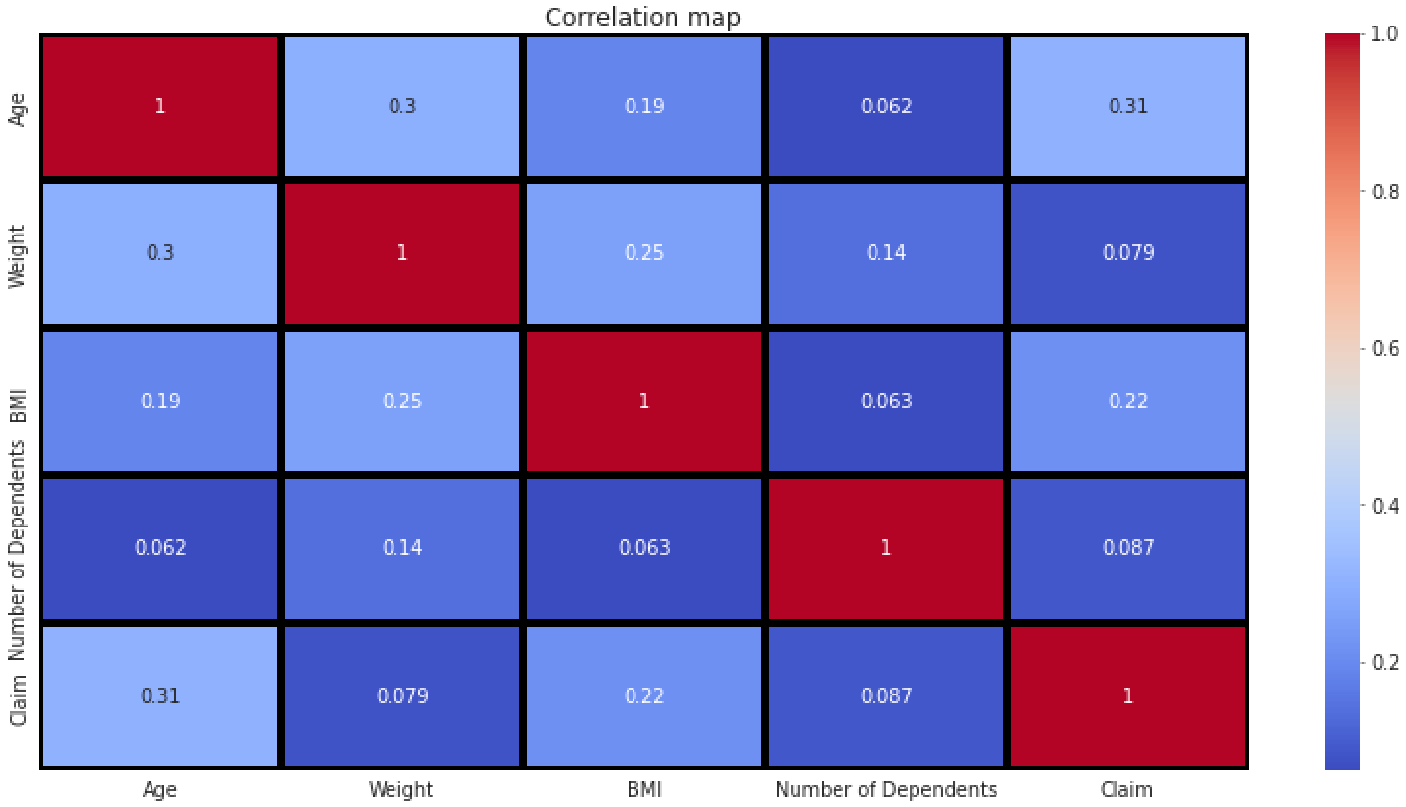
Let's import `matplotlib.pyplot` and `seaborn` .

```
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

sns.set_style('darkgrid')
matplotlib.rcParams['font.size'] = 14
matplotlib.rcParams['figure.figsize'] = (9, 5)
matplotlib.rcParams['figure.facecolor'] = '#00000000'
```

2) Correlation of the Variables among each other

```
#Plot the Heatmap

plt.figure(figsize=(20,10), dpi=50)
sns.heatmap(df.corr(),cmap= 'coolwarm', linewidths=3, linecolor='black', annot=True)
plt.title('Correlation map',fontsize=18)
```

```
Text(0.5, 1.0, 'Correlation map')
```

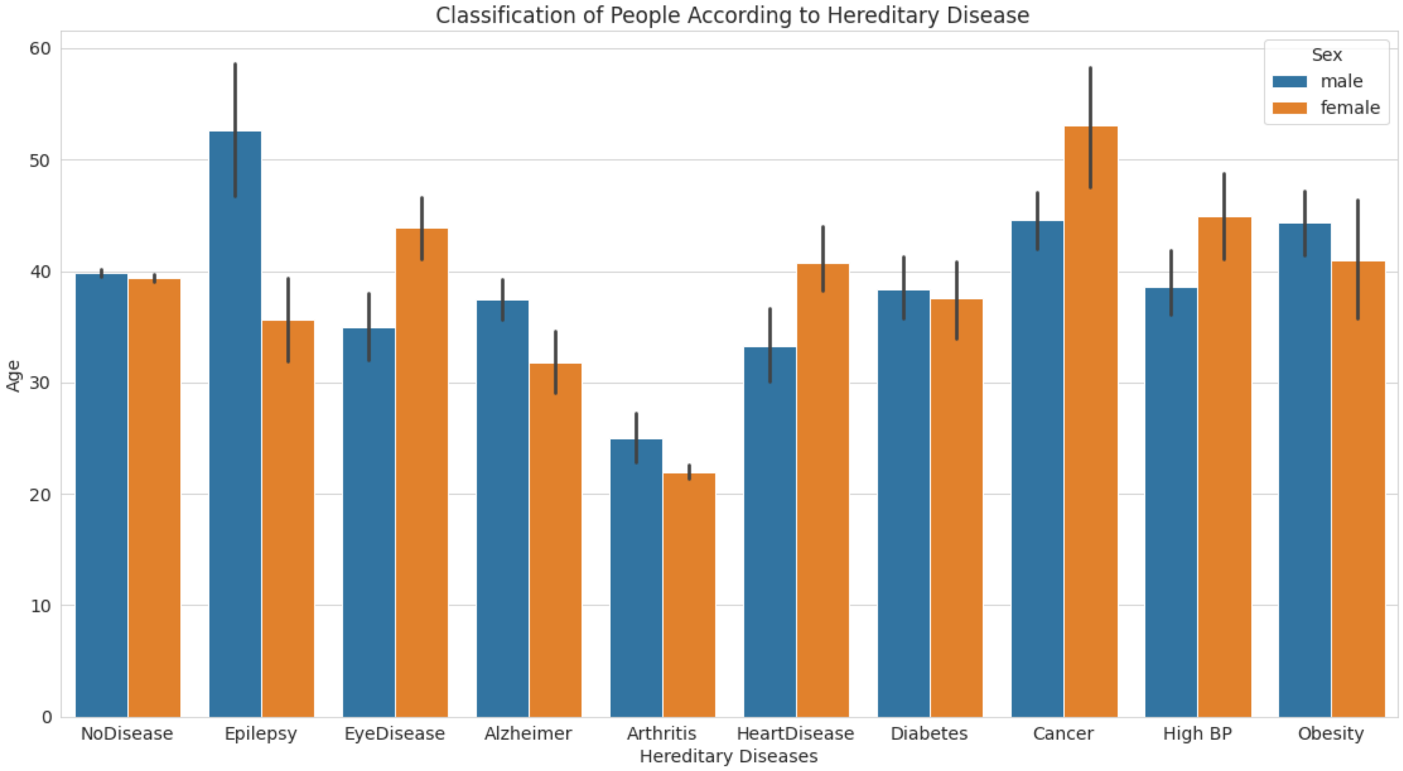


Here, it can be observed that

- The diagonals have a correlation of 1 indicating that they are correlating to each
- There is no strong positive and negative correlations.
- Only Age and Weight are more correlated with each other as compared to others.
- Rest of the columns have no linear relationship or a very weak linear relationship

3) Classification of People According to Hereditary Disease

```
#Plot a Graph
plt.figure(figsize=(19,10))
sns.set_style("whitegrid")
plt.title(" Classification of People According to Hereditary Disease")
sns.barplot(x='Hereditary Diseases', y='Age',hue='Sex', data=df);
```



Here, it can be observed that

```
• There are some males and females of age around 40 who have no hereditary disease.
• Males and females of age more than 50 have suffered only from Epilepsy and Cancer r
• Males and females have suffered from Arthritis at a young age of around 20-25 years
• Maximum males and females have suffered from a disease at the age of 30-40 years ol
```

4) Smokers and Non-Smokers

```
#Total Smokers
len(df[df['Smokers'] == 'Smoker'])
```

```
2704
```

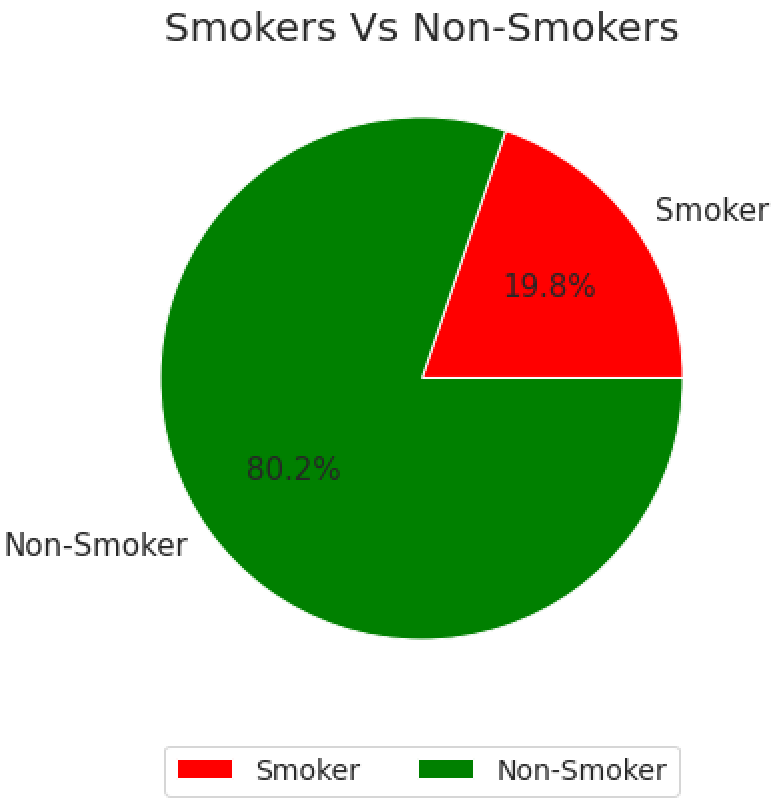
```
#Total Non-Smokers
len(df[df['Smokers'] == 'Non-Smoker'])
```

```
10944
```

```
#Define Data
data = [2704,10944]
labels = ['Smoker','Non-Smoker']
plt.tight_layout()

#Select Colour
colors = ("red","green")

#Select Label Font
textprops = {"fontsize":15}
```

```
#Select Pie Chart Font
plt.figure(figsize=(12,6))

#Create Pie Chart
plt.pie(data, colors=colors,labels=labels, textprops =textprops, autopct='%.1f%%')

#Plot Legend
plt.legend(labels=labels, loc='upper center', bbox_to_anchor=(0.5, -0.04), ncol=2)

#Select Title
plt.title("Smokers Vs Non-Smokers", fontsize=20)

#Display Pie Chart
plt.show()
```

```
<Figure size 648x360 with 0 Axes>
```

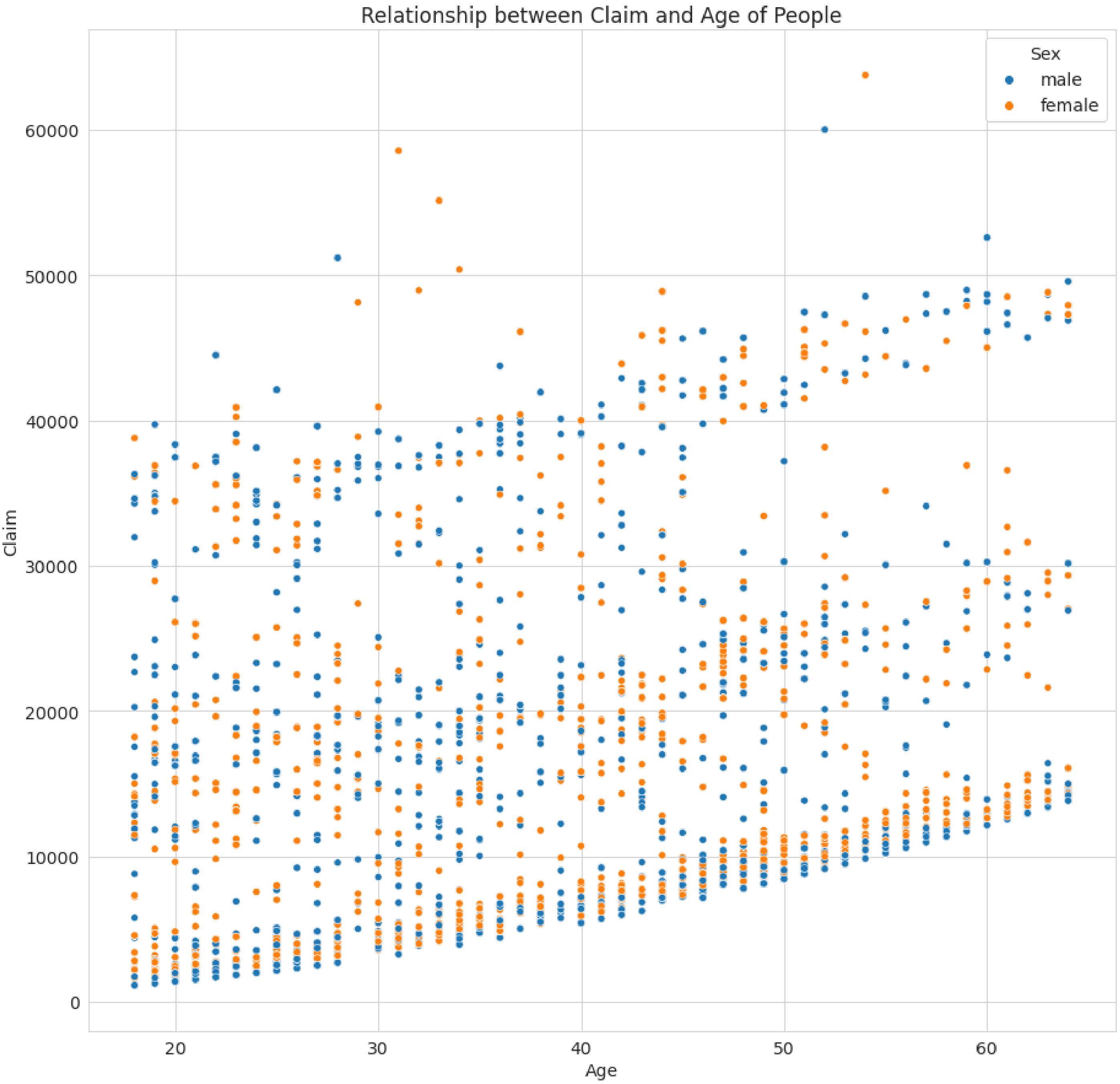


Here, it can be observed that the maximum people who came for insurance are non-smokers while few people are smokers.

5) Relationship between Claim and Age of People

```
#Plot the Graph


plt.figure(figsize=(15,15))
plt.title('Relationship between Claim and Age of People')
sns.scatterplot(x=df.Age,y=df.Claim,hue=df.Sex)
```

```
<AxesSubplot:title={'center':'Relationship between Claim and Age of People'},
xlabel='Age', ylabel='Claim'>
```

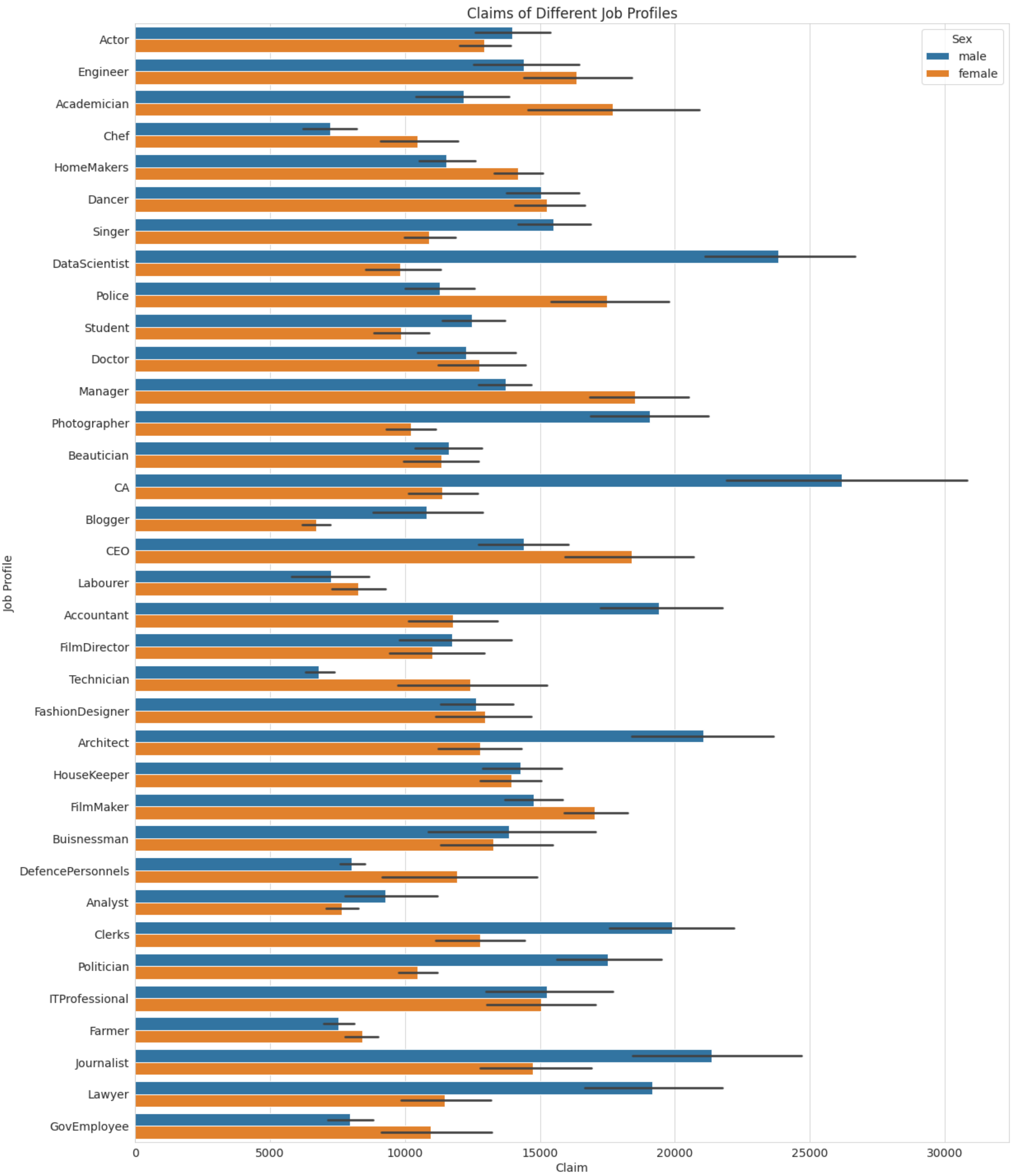


Here, it can be observed that

- There is an upward trend as the age increases, claim also increases.
- Males have a maximum claim of around Rs. 60,000 at the age nearly 50 years.
- Females have a maximum claim of more than Rs. 60,000 at the age nearly 50 years.
- Males have the lowest claim at every age as compared to females.
- Females have more claim as compared to males.

6) Claims of Different Job Profiles

```
#Plot the Graph

plt.figure(figsize=(19,25))
plt.title('Claims of Different Job Profiles')
sns.barplot(x='Claim', y='Job Profile',hue='Sex', data=df);
sns.set_style("darkgrid")
```



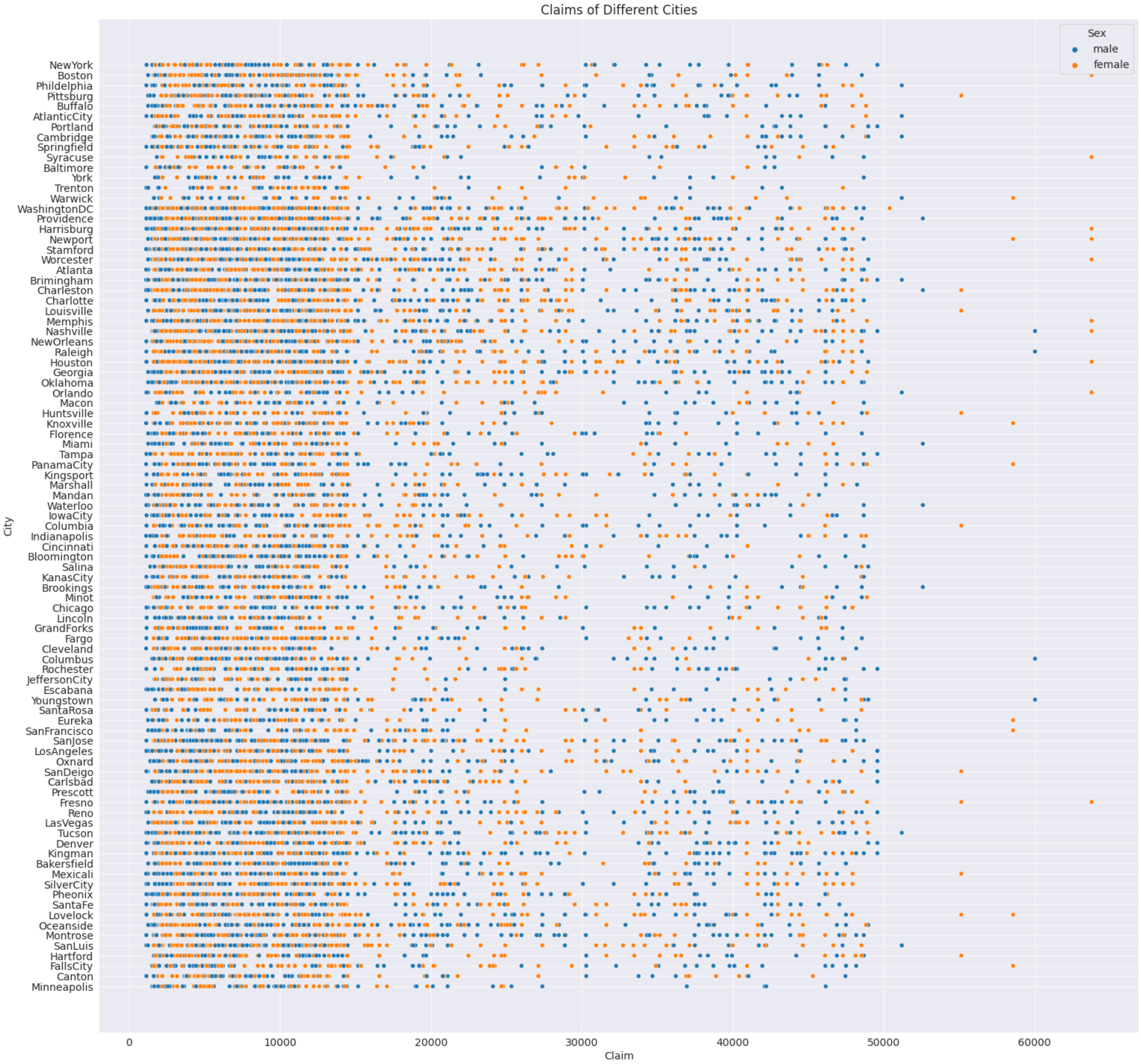
Here, it can be observed that

- Males who are CA have highest claim while in females, Manager have highest claim am
- Males who are Technician have least claim while in females, Blogger have least clai
- Females who are Beautician and CA have almost same claim.

7) Claims of Different Cities

```
plt.figure(figsize=(25,25))
plt.title('Claims of Different Cities')
sns.scatterplot(x='Claim',y='City',hue='Sex',data=df)
```

```
<AxesSubplot:title={'center':'Claims of Different Cities'}, xlabel='Claim',
ylabel='City'>
```



Here, it can be observed that

- 10 females have highest claim of more than Rs. 60,000 from Boston,Syracuse,Harrisbu
- 4 Males have highest claim of around Rs. 55,000 from Nashville, Raleigh, Columbus
- All the cities have maximum claim in range of around Rs.1000 to Rs.15000.

Let us save and upload our work to Jovian before continuing

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Updating notebook "saxena-arpit2001/data-analysis-of-claim-of-health-
insurance" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/saxena-arpit2001/data-analysis-of-
claim-of-health-insurance
```

```
'https://jovian.ai/saxena-arpit2001/data-analysis-of-claim-of-health-insurance'
```

Asking and Answering Questions

Let's learn more about this dataset.

Q1: How many People have diabetes? Visualize it.

```
#Total Diabetics
len(df[df['Diabetes'] == 'Diabetic'])
```

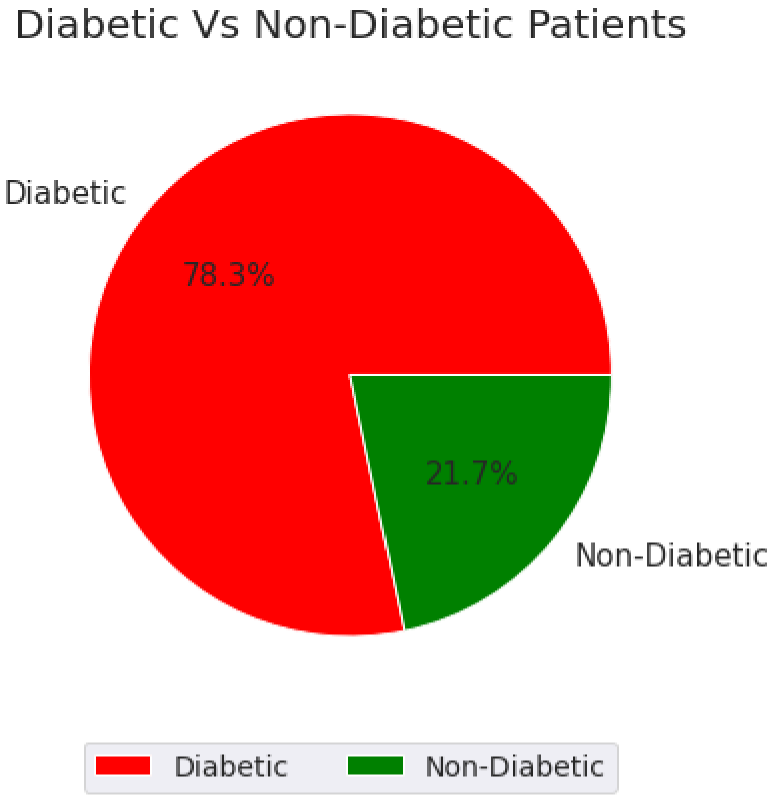
```
10688
```

```
#Total Non-Diabetics
len(df[df['Diabetes'] == 'Non-Diabetic'])
```

```
2960
```

```
#Define Data
data = [10688,2960]
labels = ['Diabetic','Non-Diabetic']
plt.tight_layout()

#Select Colour
colors = ("red","green")

#Select Label Font
textprops = {"fontsize":15}

#Select Pie Chart Font
plt.figure(figsize=(12,6))

#Create Pie Chart
plt.pie(data, colors=colors,labels=labels, textprops =textprops, autopct='%.1f%%')

#Plot Legend
plt.legend(labels=labels, loc='upper center', bbox_to_anchor=(0.5, -0.04), ncol=2)

#Select Title
plt.title("Diabetic Vs Non-Diabetic Patients", fontsize=20)

#Display Pie Chart
plt.show()
```

```
<Figure size 648x360 with 0 Axes>
```



There are 10,688 i.e. 78.3% people who have diabetes while 2960 i.e. 21.7% do not have diabetes.

Q2: What is the average BMI of the people? Interpret it.

```
mean_df=df['BMI'].mean()
mean_df
```

```
30.287294841734596
```

Since the average of BMI lies between 18.5 and 24.9 so it indicates that the average number of people have normal weight.

Q3: Create a column named 'BMI Results' and classify people as : Under weight, normal weight, overweight, Class I Obesity, Class II Obesity and Class III Obesity. Visualize it and explain what do you understand from it?

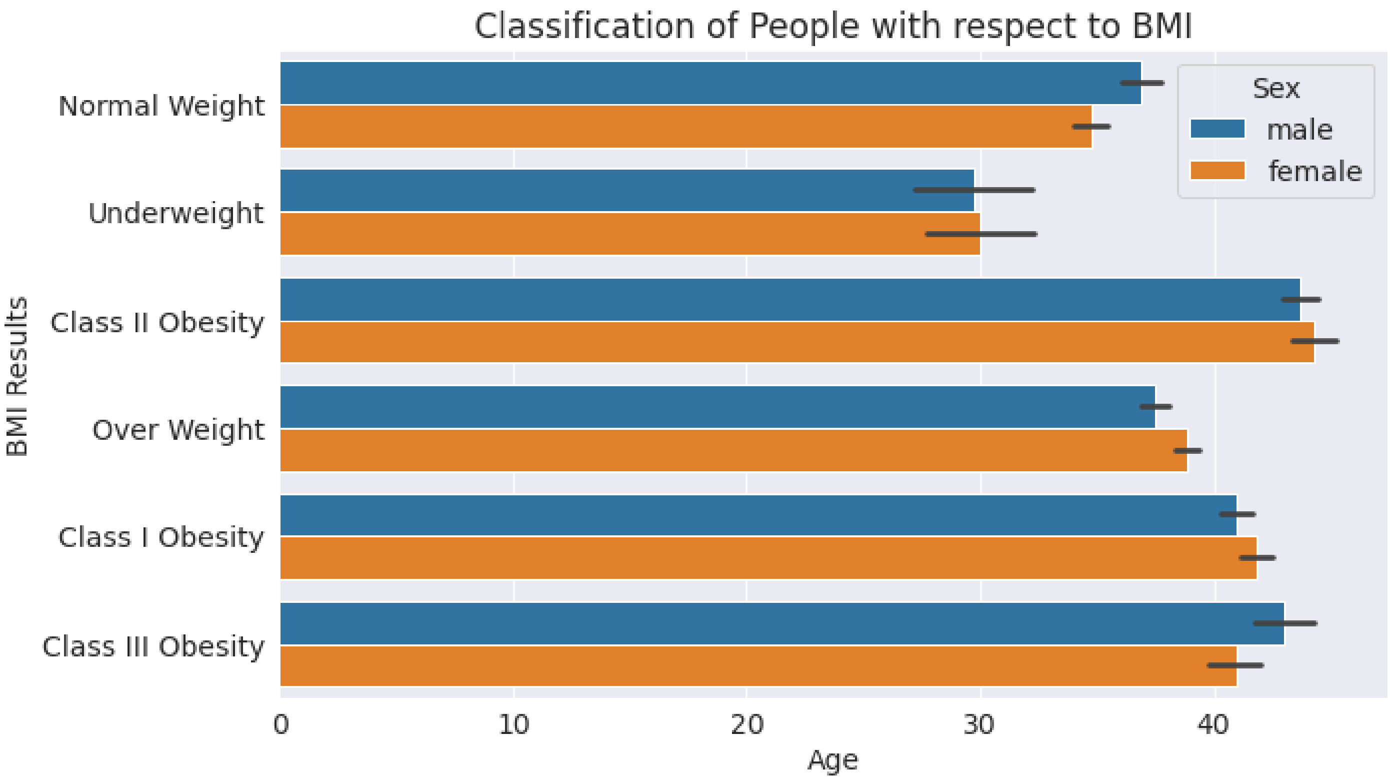
```
# Function to distinguish people.

def f(row):
    if row['BMI'] < 18.5:
        return 'Underweight'
    elif row['BMI'] >= 18.5 and row['BMI'] < 24.9:
        return 'Normal Weight'
    elif row['BMI'] >= 25.0 and row['BMI'] < 29.9:
        return 'Over Weight'
    elif row['BMI'] >= 30.0 and row['BMI'] < 34.9:
        return 'Class I Obesity'
    elif row['BMI'] >= 35.0 and row['BMI'] < 39.9:
        return 'Class II Obesity'
    elif row['BMI'] >= 40.0:
        return 'Class III Obesity'
```

```
# Create a New Column of BMI Results with the condition stated in the function.

df = df.copy()
df['BMI Results'] = df.apply(f, axis=1)
df
```

	Age	Sex	Weight	BMI	Hereditary Diseases	Number of Dependents	Smokers	City	Diabetes	Regular Exercises	Job Pr
0	60.0	male	64	24.3	NoDisease	1	Non-Smoker	NewYork	Non-Diabetic	Non-Regular	A
1	49.0	female	75	22.6	NoDisease	1	Non-Smoker	Boston	Diabetic	Regular	Engi
2	32.0	female	64	17.8	Epilepsy	2	Smoker	Phildelphia	Diabetic	Regular	Academi
3	61.0	female	53	36.4	NoDisease	1	Smoker	Pittsburg	Diabetic	Non-Regular	(
4	19.0	female	50	20.6	NoDisease	0	Non-Smoker	Buffalo	Diabetic	Non-Regular	HomeMa
...	...	...	...	...	...	...	...	...	...	...	
14995	39.0	male	49	28.3	NoDisease	1	Smoker	Florence	Diabetic	Non-Regular	FilmM
14996	39.0	male	74	29.6	NoDisease	4	Non-Smoker	Miami	Diabetic	Non-Regular	Stu
14997	20.0	male	62	33.3	NoDisease	0	Non-Smoker	Tampa	Diabetic	Non-Regular	FashionDesi
14998	52.0	male	88	36.7	NoDisease	0	Non-Smoker	PanamaCity	Diabetic	Non-Regular	Fa
14999	52.0	male	57	26.4	NoDisease	3	Non-Smoker	Kingsport	Diabetic	Non-Regular	Man

13648 rows × 13 columns

```
# Plot the graph

plt.figure(figsize=(10,6))
plt.xlabel("Body Mass Index")
plt.ylabel("Age")
plt.title(" Classification of People with respect to BMI ")
sns.set_style("darkgrid")
sns.barplot(x='Age', y='BMI Results',hue='Sex', data=df);
```

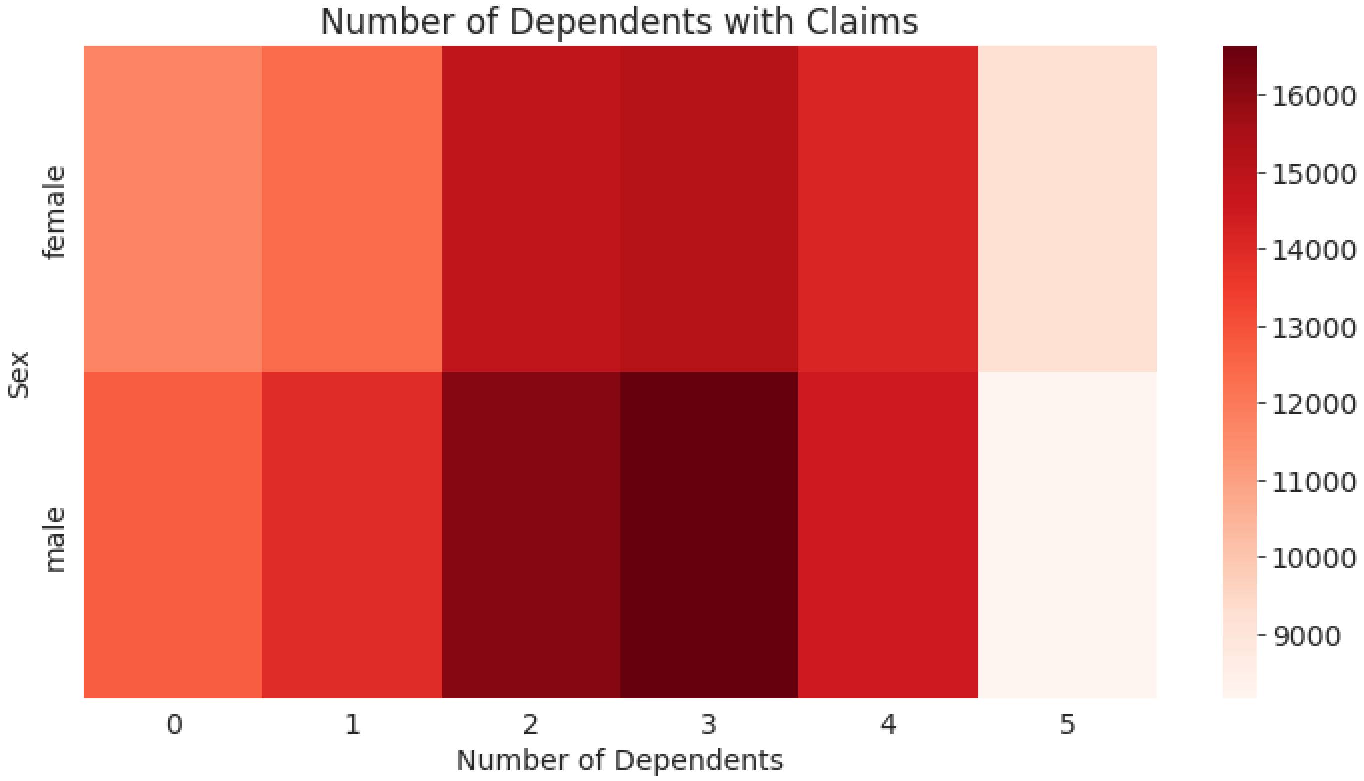


Here, it can be observed that

- Males and Females of age around 30 are underweight.
- Males and Females of age around 45 are Class II Obesity.
- Males and females of age 35-40 have normal weight.
- Maximum males and females have suffered in weights when they are around 40 years ol

Q4: Which dependent has the largest claim and least claim?

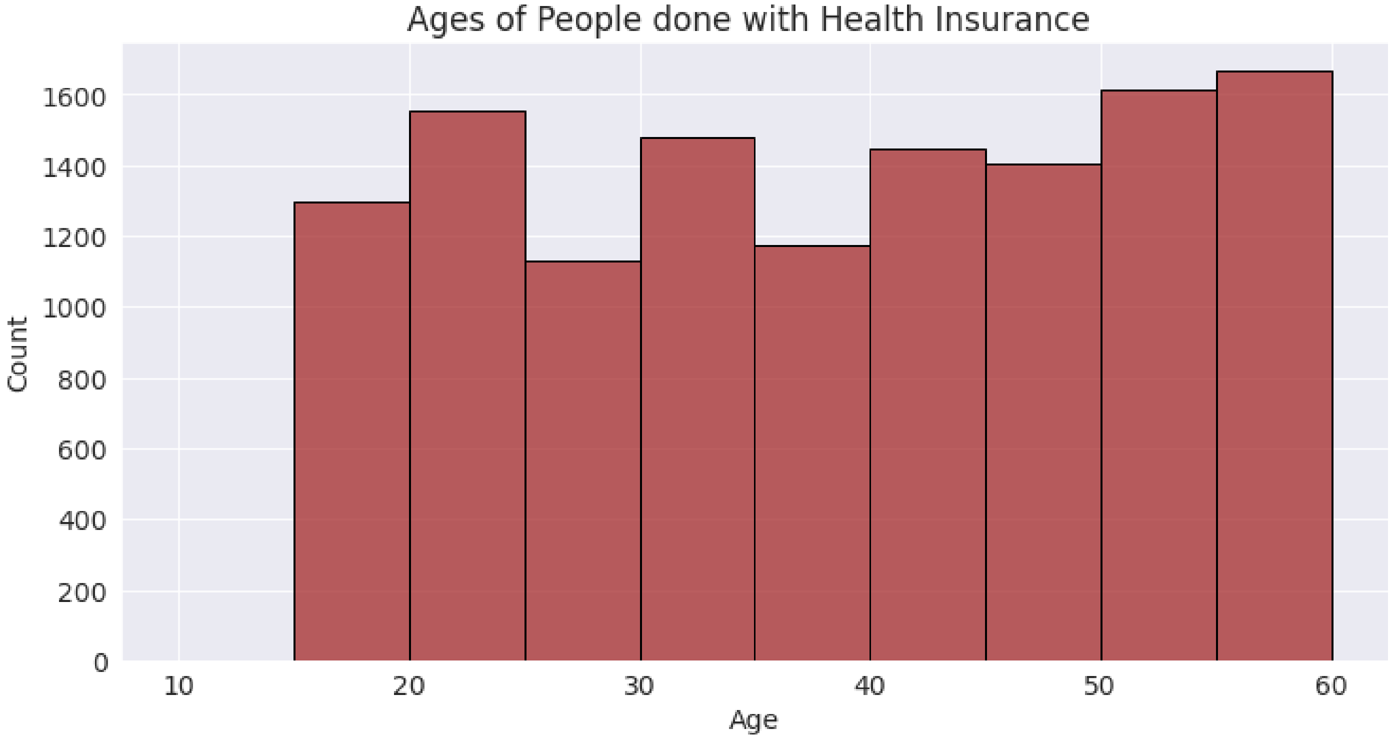
```
ndf=df[["Number of Dependents","Sex","Claim"]].groupby(["Number of Dependents","Sex"]).
ndf.reset_index(inplace=True)
ndf=ndf.pivot(index="Sex",columns="Number of Dependents",values="Claim")
fig, ax = plt.subplots(figsize=(12, 6))
ax.title.set_text('Number of Dependents with Claims')
fig.patch.set_facecolor('white')
s = sns.heatmap(ndf, cbar=True, cmap='Reds')
s.set(xlabel='Number of Dependents', ylabel='Sex');
```



Males having 3 dependents have largest claim while the males having 5 dependents have least claim.

Q5: Among which age group, the health insurance is done mostly?

```
plt.figure(figsize=(12,6))
sns.histplot(data=df,x="Age",bins=[10,15,20,25,30,35,40,45,50,55,60], color='Brown', ec
plt.title("Ages of People done with Health Insurance ")
plt.xlabel("Age")
sns.set_style("whitegrid")
```



Mostly, the Health Insurance of people aged 55-60 years old has been done while the Health Insurance of people aged 25-30 years old has been done least.

Q6: How many people do exercise regularly? Visualize it.

```python
#Total People who do regular Excercises

len(df[df['Regular Exercises'] == 'Regular'])
```

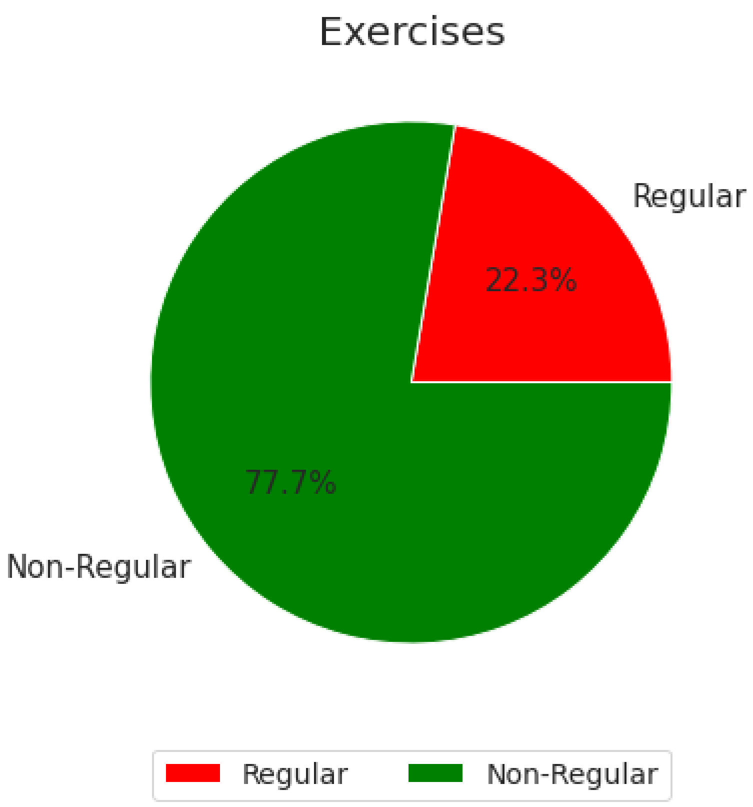
```
3045
```

```python
#Total People who don't do regular Excercises

len(df[df['Regular Exercises'] == 'Non-Regular'])
```

```
10603
```

```python
#Define Data
data = [3045,10603]
labels = ['Regular','Non-Regular']
plt.tight_layout()

#Select Colour
colors = ("red","green")

#Select Label Font
textprops = {"fontsize":15}

#Select Pie Chart Font
plt.figure(figsize=(12,6))

#Create Pie Chart
plt.pie(data, colors=colors,labels=labels, textprops =textprops, autopct='%.1f%%')

#Plot Legend
plt.legend(labels=labels, loc='upper center', bbox_to_anchor=(0.5, -0.04), ncol=2)

#Select Title
plt.title("Exercises", fontsize=20)

#Display Pie Chart
plt.show()
```

```
<Figure size 648x360 with 0 Axes>
```



There are 3045 i.e. 22.3% people who do regular exercises while 10,603 i.e. 77.7% people do not excercise regularly.

Let us save and upload our work to Jovian before continuing.

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Updating notebook "saxena-arpit2001/data-analysis-of-claim-of-health-insurance" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/saxena-arpit2001/data-analysis-of-claim-of-health-insurance
```

```
'https://jovian.ai/saxena-arpit2001/data-analysis-of-claim-of-health-insurance'
```

Inferences and Conclusion

From the above analysis, it can be concluded that

```
    1.  There is an upward trend as the age increases, claim also increases.
    2.  There is no strong positive or negative correlation among each column.
    3.  Different people have different BMI's depending on their height and weight.
    4.  BMI of andAverage Persons shows that the average number of people have normal wei
    5.  Maximum people who have purchased their health insurance plan are of age 55-60 ye
    6.  Some males and females of age around 40 have no hereditary disease.
    7.  Less people are smoker, but many have diabetes and doesn't do regular exercises.
    8.  Claim varies of different job profiles and different cities.
```

It shows how it can impact people's life on the basis of their habits like smoking behaviour, having diabetes and not doing regular exercises due to which instances happen simultaneiously without any indication. Therefore, people should purchase their health insurance plan which are made for their own benefits.

```python
import jovian
```

```python
jovian.commit()
```

References and Future Work

More Exploratory analysis and some more visualizations can be done in the same dataset which is taken from Kaggle. https://www.kaggle.com/datasets/sureshgupta/health-insurance-data-set

```python
import jovian
```

```python
jovian.commit()
```

```
[jovian] Attempting to save notebook..
[jovian] Updating notebook "aakashns/zerotopandas-course-project-starter" on https://jovian.ml/
[jovian] Uploading notebook..
[jovian] Capturing environment..
[jovian] Committed successfully! https://jovian.ml/aakashns/zerotopandas-course-project-starter
```

```
'https://jovian.ml/aakashns/zerotopandas-course-project-starter'
```