ANALYSIS OF HEALTH INSURANCE PREMIUMS USING DATA VISUALIZATION TECHNIQUES IN PYTHON

By Kalyan Pediredla
d.pediredla@liverpool.ac.uk

Declaration

I hereby declare that all the work presented in this project report and associated code is entirely my own. I have not received unauthorized assistance, and I have followed the university norms as set in the students' handbook regarding academic integrity.

We acknowledge that any external sources used for reference or inspiration are properly cited within the project report. Additionally, we understand the consequences of academic dishonesty and are committed to upholding the principles of honesty and integrity.

Contents

List of Figures

Analysis of Health Insurance Premiums:

Introduction

From over hundreds of years insurance played a crucial role in providing the financial protection against unforeseen circumstances, mitigating risks in various aspects of life, from health to property. On the other hand, data is a key as it allows businesses and individuals to analyse trends, to make informed decisions, to improve efficiency and to make strategic choices. Insurance data helps insurers to access risks more accurately and tailor policies accordingly. The report aims to closely examine the insurance data by finding the descriptive statistics of the insurance data and walks us through some good data visualization and concludes with recommendations and observations based on data analysis & visualization.

Deciphering the data:

Before comprehending the data, let me walk you through how we can import our large health insurance data into python repository or kernel for further analysis. Here I used Pandas library to import csv file. Once the file is imported, I validated the data by checking its dynamics. Through dataframe.isna().sum() syntax.

Figure:1 Importing CSV file into python repository using pandas.

```
import pandas as pd
import csv as read_csv

#importing csv file into python kernal
df= pd.read_csv("/Users/kalyanpediredla/Downloads/insurance_dataset.csv")
```

Figure 2: Checking the dynamics of the dataframe.

```
The insurence data contains 1000000 rows and 12 columns.
age                           0
gender                        0
bmi                           0
children                      0
smoker                        0
region                        0
medical_history          250762
family_medical_history   250404
exercise_frequency            0
occupation                    0
coverage_level                0
charges                       0
dtype: int64
```

I found some missing data which is represented by 'NaN' in the health insurance data file and to proceed with our analysis on the report it is essential that we have complete dataset without any missing values in it. So, the missing data in the data frame is replaced by the text 'No Medical History', 'No Family Medical History' considering the nature of the data (Assuming that these health insurance policy holders do not have any health issues by the time they enrolled for the health insurance).

Now, I have checked the top 8 rows of the data frame to grasp the content of every column and reviewed the dimensions of the data frame which confirmed it comprises 1,000,000 rows and 12 columns.

Figure 3: Shape of the dataset.

```
The insurence data contains 1000000 rows and 12 columns.
```

Figure 4: Top 8 rows of the data frame. (Before replacing the missing values)

	age	gender	bmi	children	smoker	region	medical_history	family_medical_history	exercise_frequency	occupation	coverage_level	charges
0	46	male	21.45	5	yes	southeast	Diabetes	NaN	Never	Blue collar	Premium	20460.307669
1	25	female	25.38	2	yes	northwest	Diabetes	High blood pressure	Occasionally	White collar	Premium	20390.899218
2	38	male	44.88	2	yes	southwest	NaN	High blood pressure	Occasionally	Blue collar	Premium	20204.476302
3	25	male	19.89	0	no	northwest	NaN	Diabetes	Rarely	White collar	Standard	11789.029843
4	49	male	38.21	3	yes	northwest	Diabetes	High blood pressure	Rarely	White collar	Standard	19268.309838
5	55	female	36.41	0	yes	northeast	NaN	NaN	Never	Student	Basic	11896.836613
6	64	female	20.12	2	no	northeast	High blood pressure	High blood pressure	Never	Blue collar	Basic	9563.655011
7	53	male	30.51	4	no	southeast	Heart disease	High blood pressure	Rarely	Student	Standard	15845.293730

Figure 5: Top 8 rows of the data frame. (After replacing the missing values)

	age	gender	bmi	children	smoker	region	medical_history	family_medical_history	exercise_frequency	occupation	coverage_level	charges
0	46	male	21.45	5	yes	southeast	Diabetes	No Family Medical History	Never	Blue collar	Premium	20460.307669
1	25	female	25.38	2	yes	northwest	Diabetes	High blood pressure	Occasionally	White collar	Premium	20390.899218
2	38	male	44.88	2	yes	southwest	No Medical History	High blood pressure	Occasionally	Blue collar	Premium	20204.476302
3	25	male	19.89	0	no	northwest	No Medical History	Diabetes	Rarely	White collar	Standard	11789.029843
4	49	male	38.21	3	yes	northwest	Diabetes	High blood pressure	Rarely	White collar	Standard	19268.309838
5	55	female	36.41	0	yes	northeast	No Medical History	No Family Medical History	Never	Student	Basic	11896.836613
6	64	female	20.12	2	no	northeast	High blood pressure	High blood pressure	Never	Blue collar	Basic	9563.655011
7	53	male	30.51	4	no	southeast	Heart disease	High blood pressure	Rarely	Student	Standard	15845.293730

Now to analyse the statistical summary of the data frame. I used two different syntaxes 'df.describe()', 'df.describe(include=["object", "bool"])' as the data frame consists of both numerical and categorical data in the columns. Here the first syntax calculates the mean, std, min, max and percentile values of the numerical data and the second syntax calculates the count, unique, top and frequency of the categorical data. These statistics helps us in understanding characteristics and distribution of health insurance data.

Figure 6: Statistical analysis of all the numerical data from the data frame.

	age	bmi	children	charges
count	1000000.000000	1000000.000000	1000000.000000	1000000.000000
mean	41.495282	34.001839	2.499886	16735.117481
std	13.855189	9.231680	1.707679	4415.808211
min	18.000000	18.000000	0.000000	3445.011643
25%	29.000000	26.020000	1.000000	13600.372379
50%	41.000000	34.000000	2.000000	16622.127973
75%	53.000000	41.990000	4.000000	19781.465410
max	65.000000	50.000000	5.000000	32561.560374

Figure 7: Statistical analysis of all the categorical data from the data frame.

	gender	smoker	region	medical_history	\
count	1000000	1000000	1000000	1000000	
unique	2	2	4	4	
top	male	yes	northeast	No Medical History	
freq	500107	500129	250343	250762	

	family_medical_history	exercise_frequency	occupation	\
count	1000000	1000000	1000000	
unique	4	4	4	
top	No Family Medical History	Rarely	Unemployed	
freq	250404	250538	250571	

	coverage_level
count	1000000
unique	3
top	Basic
freq	333515

Exploratory Data Analysis:

Our Exploration of the dataset involves visualizing the data comprehensively. To understand the distribution of health insurance policy holders across various regions we plotted the pie chart using matplotlib and seaborn libraries in python to represent the distribution visually.

Figure 8: Health insurance policyholders' distribution across regions through pie chart.

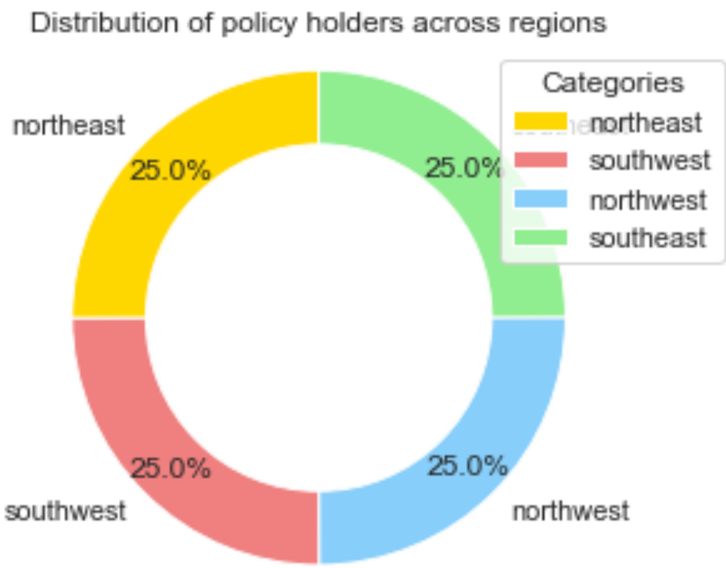
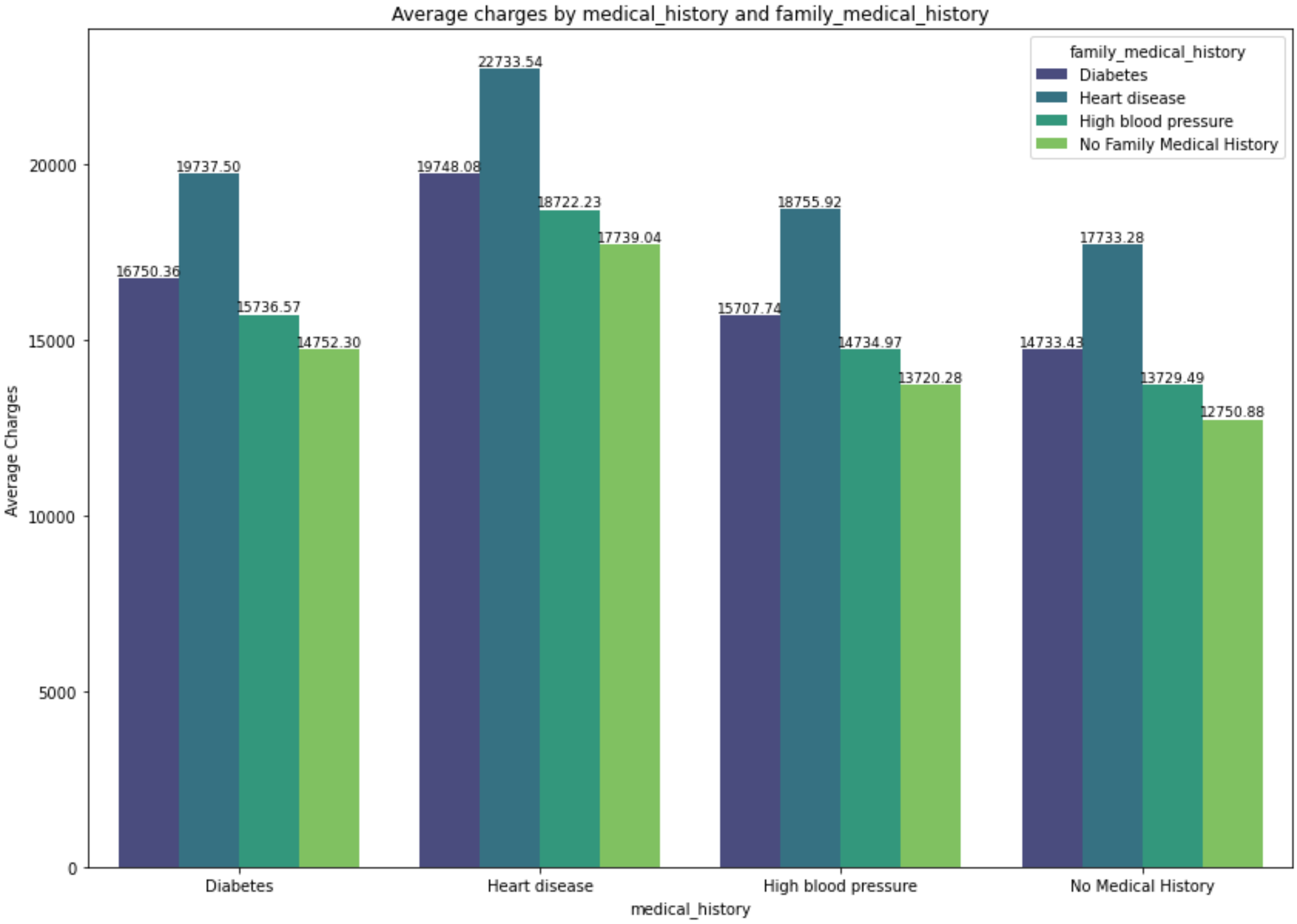


Figure1 pie chart concludes that we have equal distribution of health insurance policy holders across northeast, southwest, northwest and southeast regions. It means we have around 2.5 lakhs insurance policy holders for each region.

Now, let us explore the how medical history and family medical history impact the premium charges.

Figure 9: Average charges by medical history and family medical history



The above plot describes policy holder's medical history data on x-axis and consider each unique category with all the unique categories of family medical history column and shows us the average charges for each unique combination of both 'medical history' and 'family medical history' column data. It gives us the clear picture of how the policy holders medical history impacts the premium charges.

Figure 10: Average charges by medical history

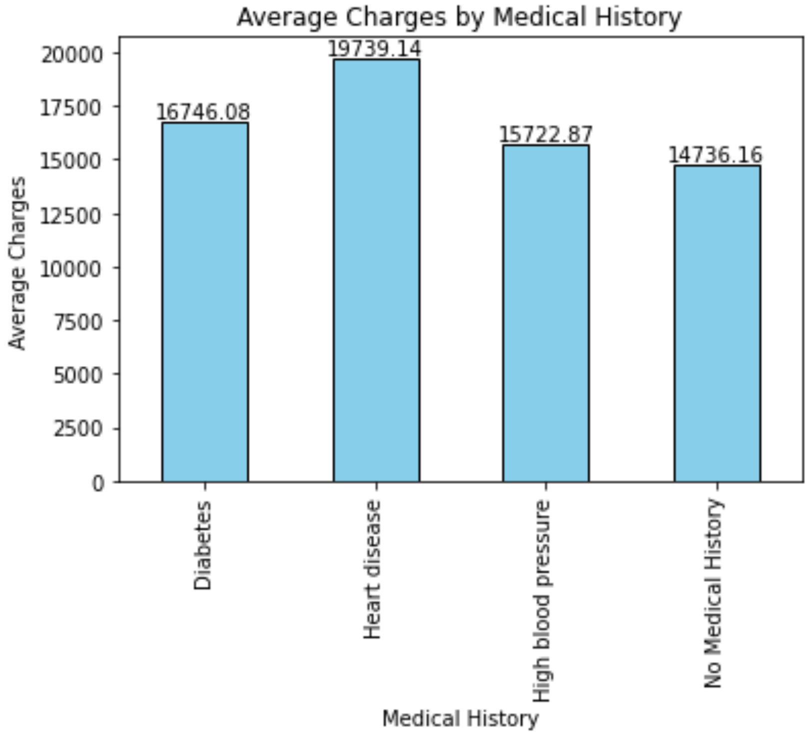
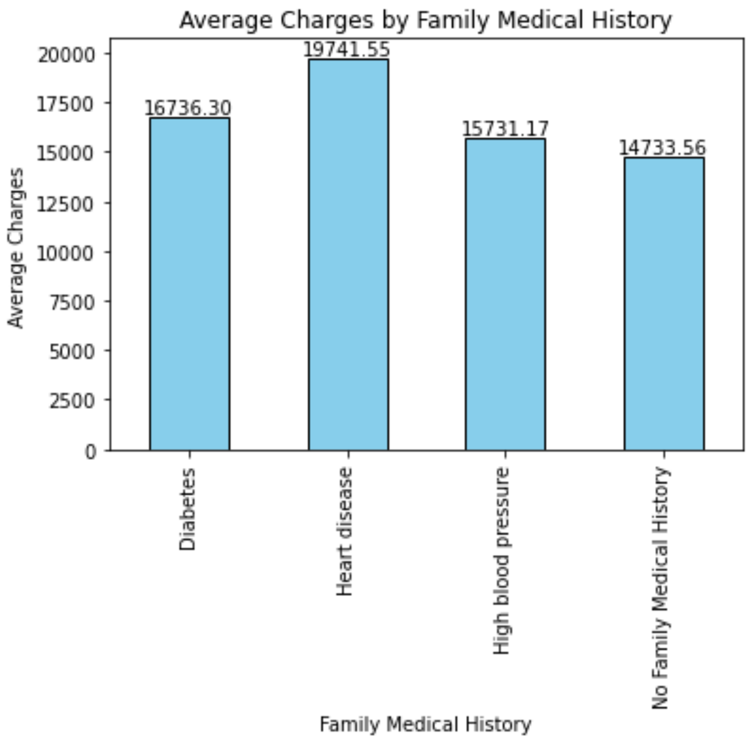
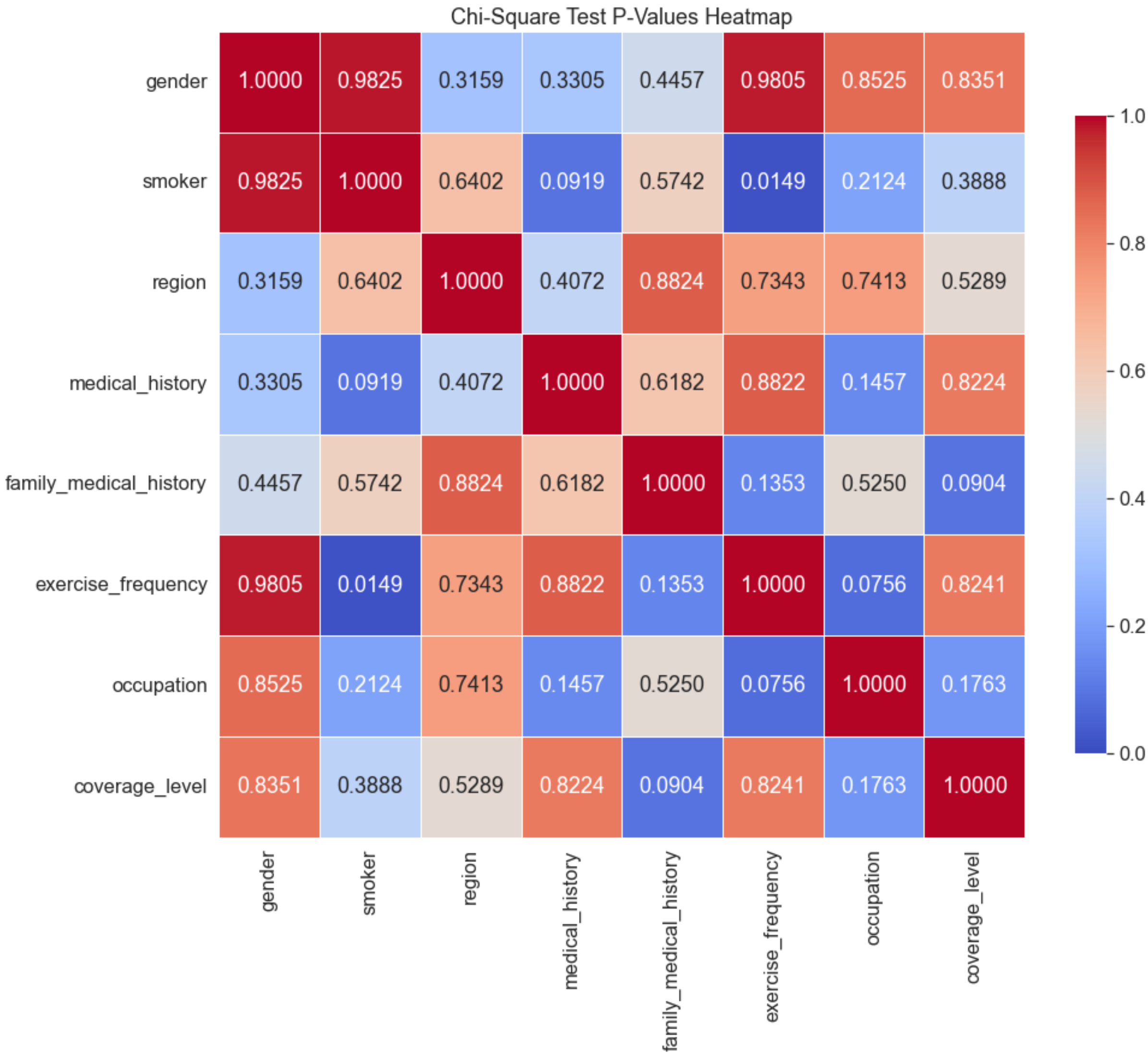


Figure 11: Average charges by Family medical history.



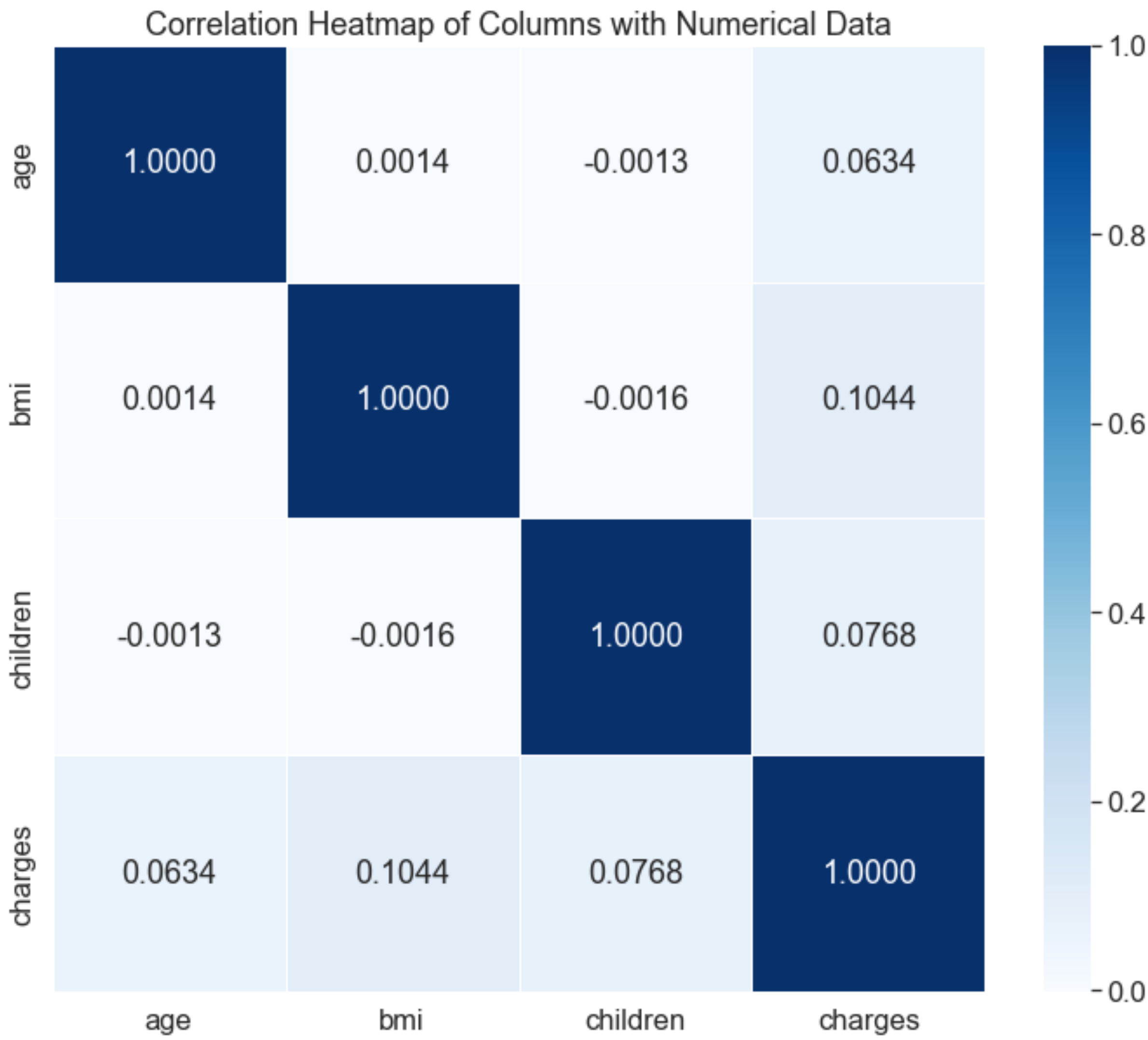
The insurance data is further analysed by finding correlation between all the categorical data and numerical data separately as it opens new insights into the data.

Figure12: Correlation heatmap of all the categorical features from the health insurance data.



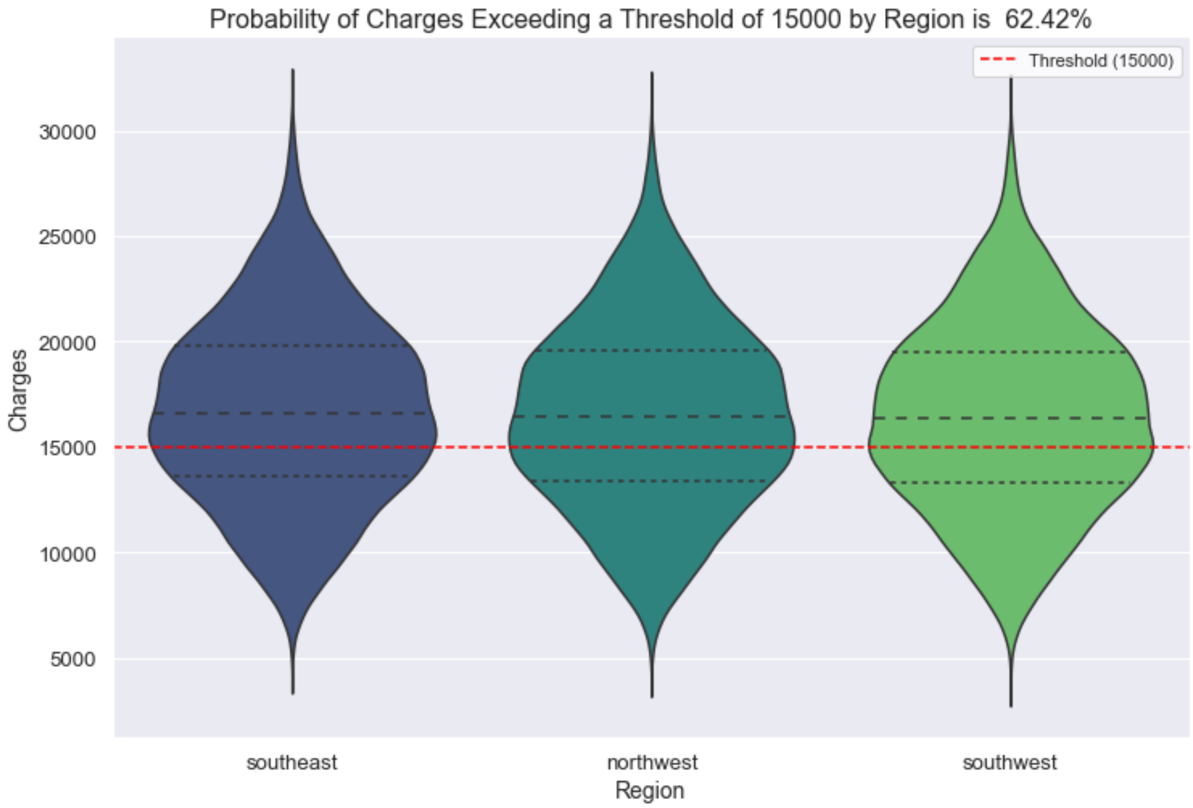
For finding correlation between categorical data, here we used Chi-Square test and plotted the heat map. There is hight correlation between gender and smoker, gender and exercise frequency and we have very low correlation between smokers and exercise frequency.

Figure13: Correlation heatmap between all the numerical data from data frame.



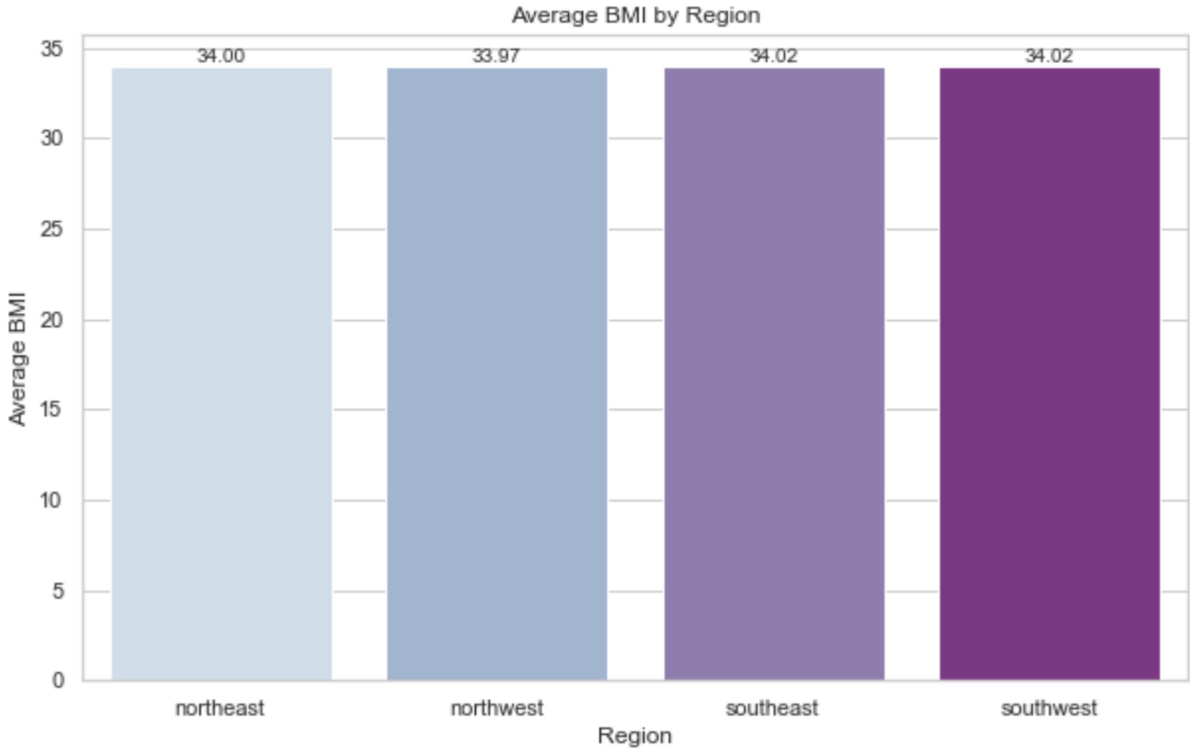
The above correlation heatmap shows that there is 10% positive correlation between BMI and Charges and 6% positive correlation between age and the premium charges.

Figure:14 Violin plot with probability of charges exceeding a threshold level.



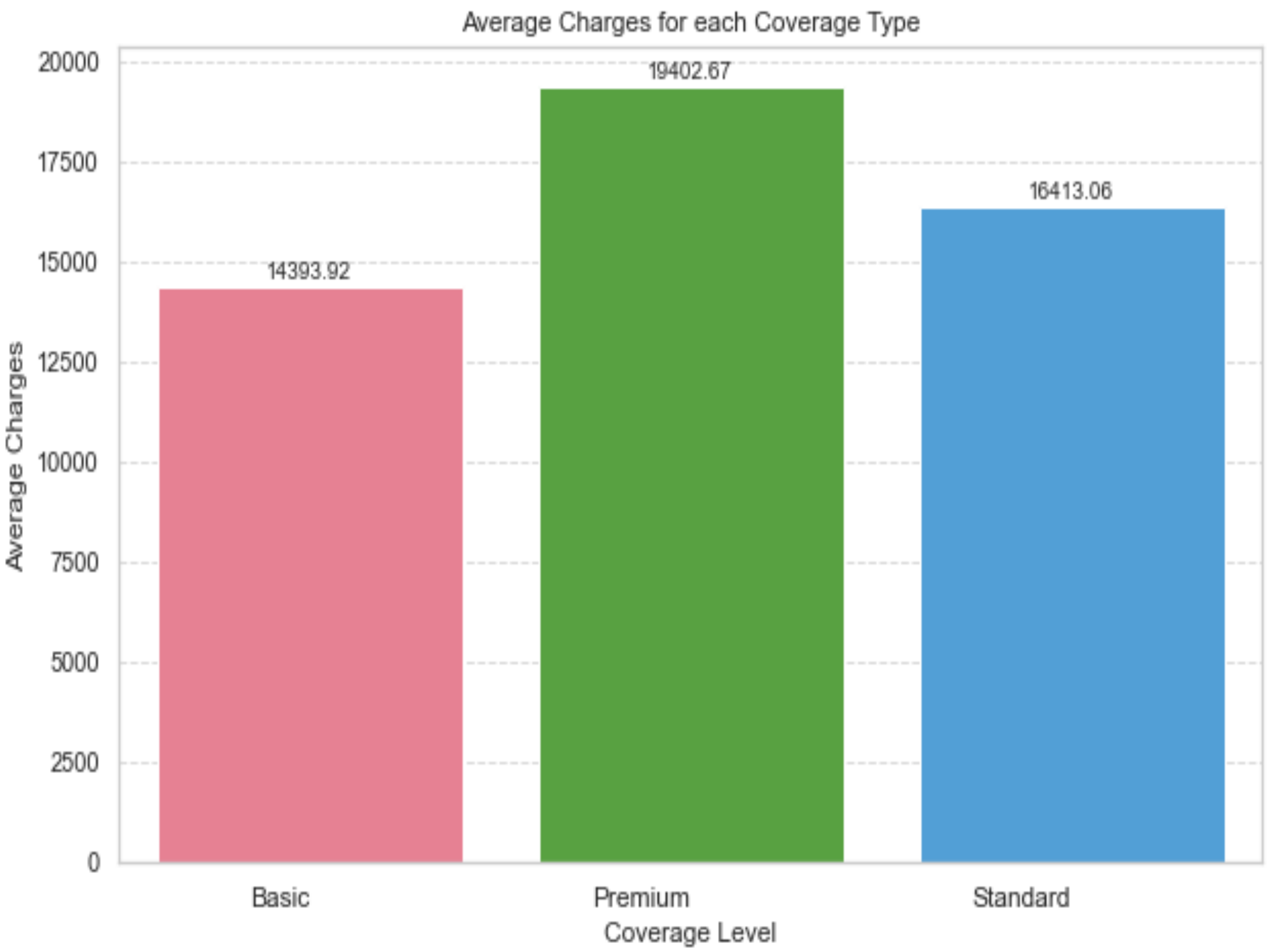
From the figure 14 the probability of charges exceeding the threshold level of $15,000 is 62.42%. The total number of health insurance policy holders in these three regions are equal. Hence, we have the similar probability. In the above plot red line shows the threshold level and the highlighted part shows the distribution of policy holders above and below the threshold level.

Figure 15: Bar plot of Average BMI for each region.



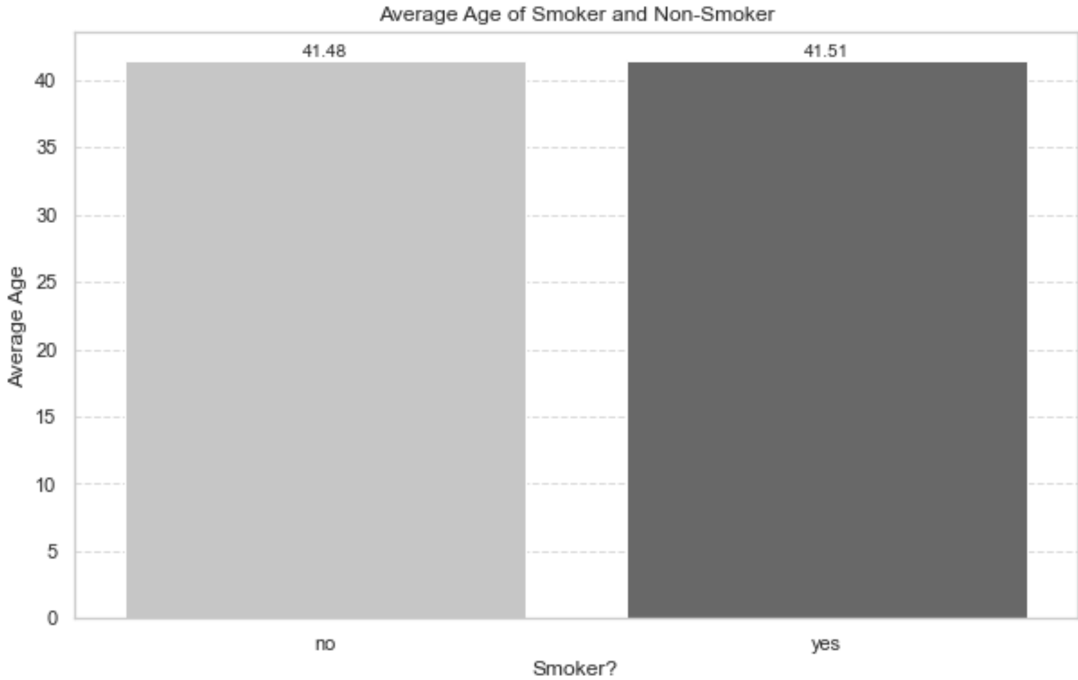
The average BMI for a specific region is calculated by dividing total BMI of all the health insurance policy holders in that specific region by total number of policy holders. From the figure 15 the average BMI of policy holders from northeast, northwest, southeast and southwest is 34.00, 33.97, 34.02 and 34.02 respectively. It shows that there is no major difference in the average BMIs of policy holders from all the regions.

Figure 16: Bar plot of average charges for each coverage level.



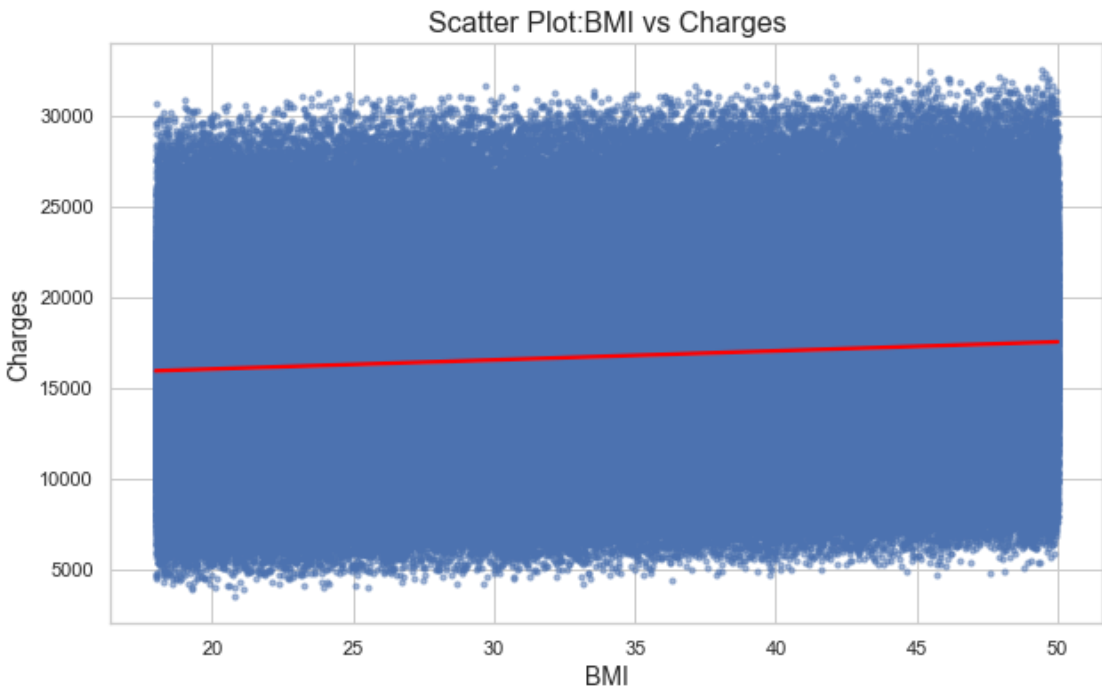
The average charge for basic coverage is 14393.92, standard coverage is 16413.06 and for the premium coverage is 19402.67. The average premium charge is increasing as we choose the higher coverage level as the benefits of higher coverage level are more when compared to the basic coverage level.

Figure 17: Bar plot of average age between smoker and non-smoker.



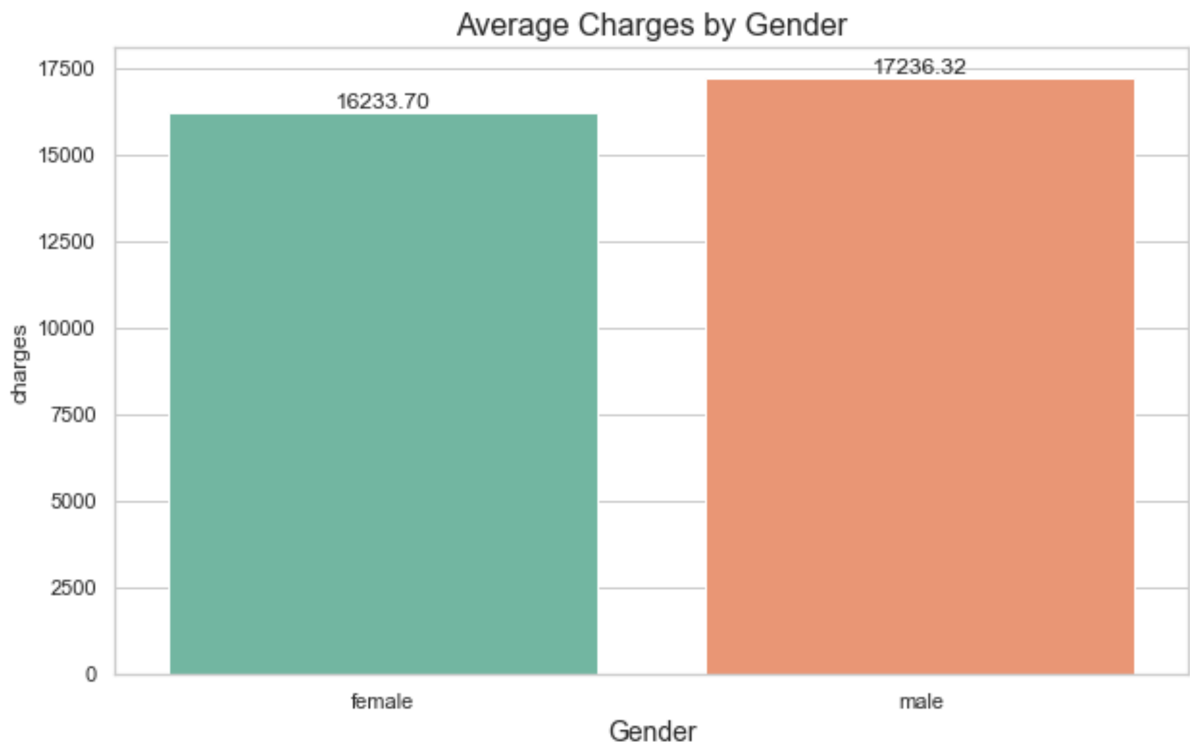
The average age of health insurance policy holders who smoke and doesn't smoke is 41.51, 41.48 respectively. There is minor difference between average age of a smoker which suggests that age is not a significant factor influencing smoking habits.

Figure18: Scatter plot visualizing the relationship between BMI and Charges.



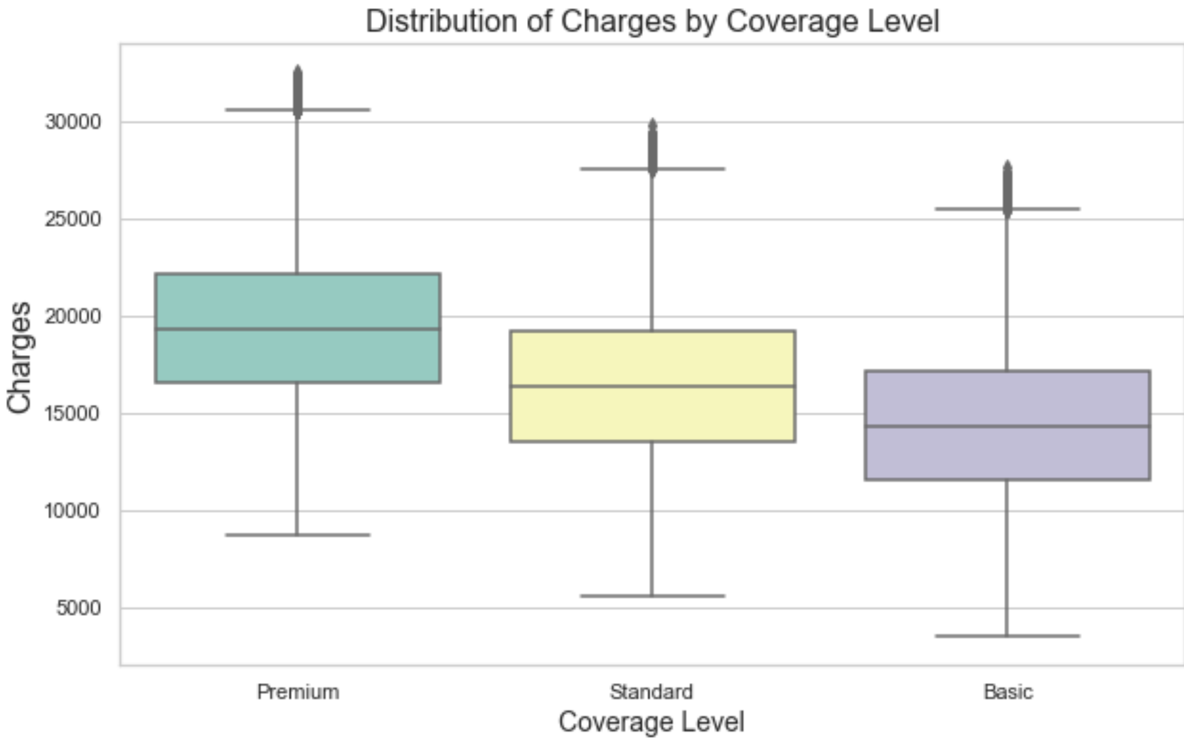
The Body Mass Index (BMI) is considered within a normal range when the weight and height of an individual are proportionate. The red line in the scatter plot represents the regression line that best fit with the available data. There is a slight positive trend in the distribution of data which is clearly visible from the above scatter plot.

Figure:19 Bar graph comparing the average charges between male and female.



The average health insurance premium of male, female is 17236.32, 16233.70 respectively. It shows the average insurance premium among male policyholders is higher when compared to females. The possible reason could be male policyholders are less healthy when compared to all the female policyholders or there might be significant difference in the insurance coverage level opted by males and female policyholders. Several additional significant elements could clarify the variation in insurance premium costs for men and women.

Figure:20 Distribution charges for different coverage levels through boxplot.



In the above box plot the line inside the box shows the median premium charge for each coverage level accordingly and the entire box represents the inter quartile range or central 50% of premium charges distributions for the different health insurance coverage levels. The whiskers (the lines extending from both ends represents high or low premium charges compared to majority of policyholders premium charges. And the unusual high premium charges are shows with the outliers in the above box plot.

Findings and Recommendations

From the above visualization we found that we have almost equal number of policy holders across the regions with northeast having the highest policy holders with 250343 individuals. The descriptive stats give the insights into the minimum and maximum insurance premium charges i.e., £ 3445 and £ 32561 respectively. The highest number of people opted for basic coverage type; we have more male policy holders in total when compared to the other policy holders. Most of the policy holders are unemployed and the individuals with no medical history or family medical history are more when compared to other category of people.

And given the dynamics of the data we have replaced NAN with ‘No medical history’ and it opened clear insights which we can observe from figures 9,10 & 11. The medical history of the insurance policy holder is having the highest impact on the premiums charged. The policyholders with no medical history and no family medical history have the less premium charges. The average charges for this category of policy holders are £12750 which is significantly low when compared with all other possibilities and the policyholders with heart disease in both categories (‘medical history’, ‘family medical history’) are paying the highest premium charges of £ 22733.54 on an average. And from the correlation heatmap, we can observe 98% positive correlation between smoker and gender which concludes that majority of the unique gender (either male or female) has smoking habits. There is 98% positive correlation between gender and exercise frequency which again concludes that a unique gender is exercising frequently. There is 10% positive correlation between BMI and Charges. From the figure 16, the regional factors do not play substantial role in influencing the BMI.

Finally, I recommend the health insurance policy holders to do regular exercises and maintain the healthy work life balance as mental stress often led to health complications. While health insurance offers support during challenging circumstances, prioritizing personal well-being is crucial, contributing to an extended life span. Health issues, at times, can result in fatalities.

Appendix A- Further recommendations:

I recommend insurance companies to focus on communicating to policyholders about available benefits and preventive measures and to educate them on health choices can impact insurance costs. And I also recommend exploring partnerships with digital health platforms to enhance policyholder engagement.

Appendix B - Conclusion

In conclusion, I have imported and conducted preliminary checks on data formats and dynamics of the data, to ensure accurate implementation and analysis, substituted absent values with the appropriate data and continued by employing data visualization techniques to gain insights into dataset. Pie chart, bar plot, box plot, scatter plot, violin plot and heat map were created to find the trends and relationship within the data. Several python inbuilt libraries like, matplotlib, seaborn, pandas, label encoder, scikit-learn etc., were used to create a great visualization plot to identify and interpret the complex information. To improve the efficiency and accuracy of data interpretation and analysis. I segmented my data analysis process into distinct tasks and executed the analysis by developing compact python functions, ensuring efficient data management and the generation of meaningful visualization.

Appendix C– Python Code

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Mon Jan  8 12:14:07 2024

@author: kalyanpediredla
"""

import pandas as pd
import csv as read_csv

#importing csv file into python kernal
df= pd.read_csv("/Users/kalyanpediredla/Downloads/insurance_dataset.csv")
pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns',None)

#Verifying the presence of null values in the data frame to assess its readiness for analysis.
df.notnull().all()
df.shape #checking shape of the dataframe
print(f'The insurence data contains {df.shape[0]} rows and {df.shape[1]} columns.')
print(df.isna().sum())

## Replacing missing values with most frequent value in its column.
df['medical_history']=df['medical_history'].fillna('No Medical History')
df['family_medical_history']=df['family_medical_history'].fillna('No Family Medical History')

#Displaying the initial 8 rows of the data frame for insight into the data's charactersitics.
print(df.head(8))
print(df.isna().sum())

##Generating statistical summaries for the numerical data columns within the data frame.
desc_stat_of_numercialdata= df.describe()

##Generating statistical summaries for the categorical data columns within the data frame.
desc_stat_of_categoricaldata= df.describe(include=["object", "bool"])

print(desc_stat_of_numercialdata)
print(desc_stat_of_categoricaldata)

#Confirming distinct values in the 'region' column of the data frame.
print(df['region'].unique())

##importing matplotlib and seaborn libraries to facilitate data visualization.
import matplotlib.pyplot as plt
import seaborn as sns
```

```python
## Creating a pie chart depicting the distribution of policy holders among different regions.
category_counts=df['region'].value_counts()
explode=(0,0,0,0)
colors=['gold','lightcoral','lightskyblue','lightgreen']
plt.subplots()
plt.pie(category_counts, labels=category_counts.index,autopct='%1.1f%%',startangle=90,pctdistance=0.85, explode=explode,colors=colors)
centre_circle=plt.Circle((0,0),0.70,fc='white')
fig=plt.gcf()
fig.gca().add_artist(centre_circle)
plt.axis('equal')
plt.legend(category_counts.index,title='Categories',loc='upper right')
plt.title('Distribution of policy holders across regions')
plt.show()


print() ## creating space from the previous output in the console.


groupby_medical_history=  df.groupby('medical_history')['charges'].mean()

ax=groupby_medical_history.plot(kind='bar', color='skyblue',edgecolor='black')
plt.title('Average Charges by Medical History')
plt.xlabel('Medical History')
plt.ylabel('Average Charges')
for index, value in enumerate(groupby_medical_history):
    ax.text(index,value + 0.1, f'{value:.2f}', ha='center', va='bottom')
plt.show()

groupby_family_medical_history=  df.groupby('family_medical_history')['charges'].mean()

ax=groupby_family_medical_history.plot(kind='bar', color='skyblue',edgecolor='black')
plt.title('Average Charges by Family Medical History')
plt.xlabel('Family Medical History')
plt.ylabel('Average Charges')
for index, value in enumerate(groupby_family_medical_history):
    ax.text(index,value + 0.1, f'{value:.2f}', ha='center', va='bottom')
plt.show()

grouped_data= df.groupby(['medical_history','family_medical_history'])['charges'].mean().reset_index()
plt.figure(figsize=(14,10))
ax=sns.barplot(x='medical_history', y='charges',hue='family_medical_history', data=grouped_data,palette='viridis')
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}', (p.get_x()+p.get_width()/2., p.get_height()), ha='center', va='bottom', fontsize=9)

plt.title('Average charges by medical_history and family_medical_history')
plt.xlabel('medical_history')
plt.ylabel('Average Charges')
plt.show()
```

```python
##plotting violin plot that represents the probability of charges exceeding the threshold level.(eg.$15000)
def prob_charges_exceeding_thershold_by_region(threshold_charges):
    selected_regions=['northwest', 'southeast', 'southwest']
    selected_df = df[df['region'].isin(selected_regions)]
    total_records=len(selected_df)
    exceed_theresshold_records= len(selected_df[selected_df['charges']>threshold_charges])
    probability_exceeding_threshold= exceed_theresshold_records/total_records
    print(f"Probability of charges exceeding $15000 for selected regions: {probability_exceeding_threshold: .2%}")
    plt.figure(figsize=(12,8))
    colors=sns.color_palette('viridis', n_colors=len(selected_regions))
    ax=sns.violinplot(x='region' ,y='charges', data= selected_df, inner='quartile',palette=colors)
    sns.set_theme(style='whitegrid')
    ax.axhline(y=threshold_charges , color='red', linestyle='--', label= f'Threshold ({threshold_charges})')
    plt.xlabel('Region', fontsize=14)
    plt.ylabel('Charges', fontsize=14)
    plt.title(f'Probability of Charges Exceeding a Threshold of {threshold_charges} by Region is {probability_exceeding_threshold: .2%}',fontsize=16)
    ax.legend()
    plt.show()


prob_charges_exceeding_thershold_by_region(threshold_charges)

##plotting the barplot that represents the average BMI by region.
def plot_avg_bmi_by_region(df,bmi,region):
    average_bmi_by_region= df.groupby(region)[bmi].mean().reset_index()
    print(average_bmi_by_region)
    plt.figure(figsize=(10,6))
    colors=sns.color_palette("BuPu",n_colors=len(average_bmi_by_region))
    ax=sns.barplot(x=region,y=bmi, data=average_bmi_by_region,palette=colors)
    ax.set(xlabel='Region', ylabel='Average BMI', title='Average BMI by Region')
    for index, value in enumerate((average_bmi_by_region[bmi])):
        ax.text(index, value + .01, f'{value:.2f}',ha='center', va='bottom', fontsize=10)
    plt.show()
plot_avg_bmi_by_region(df, 'bmi', 'region')


##plotting the barplot that represents the average charges by coverage level.
def plot_avg_charges_by_coverage(df,charges,coverage_level):
    average_charges_by_coverage= df.groupby(coverage_level)[charges].mean().reset_index()
    print(average_charges_by_coverage)
    plt.figure(figsize=(10,6))
    colors=sns.color_palette("husl",n_colors=len(average_charges_by_coverage))
    ax=sns.barplot(x=coverage_level,y=charges, data=average_charges_by_coverage,palette=colors,)
    sns.set_theme(style='whitegrid')
    ax.grid(axis='y', linestyle='--', alpha=0.7)
    ax.set_axisbelow(True)
    ax.set(xlabel='Coverage Level', ylabel='Average Charges', title='Average Charges for each Coverage Type')
    ax.set_xticklabels(average_charges_by_coverage[coverage_level], ha='right', fontsize=12)
    for index, value in enumerate((average_charges_by_coverage[charges])):
        ax.text(index, value + 50, f'{value:.2f}',ha='center', va='bottom', fontsize=10)
    plt.show()
plot_avg_charges_by_coverage(df,'charges', 'coverage_level')
```

```python
##plotting the barplot that represents the average age by smoking status.
def plot_avg_age_by_smoking_status(df,age,smoker):
    average_age_by_smoking_status= df.groupby(smoker)[age].mean().reset_index()
    print(average_age_by_smoking_status)

    plt.figure(figsize=(10,6))

    colors=sns.color_palette('Greys',n_colors=len(average_age_by_smoking_status))
    ax=sns.barplot(x=smoker,y=age, data=average_age_by_smoking_status,palette=colors)
    sns.set_theme(style='whitegrid')
    ax.grid(axis='y', linestyle='--', alpha=0.7)
    ax.set_axisbelow(True)
    ax.set(xlabel='Smoker?', ylabel='Average Age', title='Average Age of Smoker and Non-Smoker')
    for index, value in enumerate((average_age_by_smoking_status[age])):
        ax.text(index, value + .01, f'{value:.2f}',ha='center', va='bottom', fontsize=10)
    plt.show()
plot_avg_age_by_smoking_status(df, 'age', 'smoker')

## plotting scatter plot with regression line that shows the relationship between BMI and Charges.
def bmi_vs_charges_plot(df, bmi, charges):
    plt.figure(figsize=(10,6))
    sns.regplot(x=bmi,y=charges,data=df,scatter_kws={'s':10,'alpha':0.5},line_kws={'color':'red'})
    sns.set_theme(style='whitegrid')
    plt.xlabel('BMI', fontsize=14)
    plt.ylabel('Charges', fontsize=14)
    plt.title('Scatter Plot:BMI vs Charges',fontsize=16)
    plt.show()
bmi_vs_charges_plot(df, 'bmi', 'charges')


### plotting barplot that represents average charges by gender.
def compare_charges_by_gender(df, charges, gender):
    average_charges_by_gender=df.groupby(gender)[charges].mean().reset_index()
    print(average_charges_by_gender)

    plt.figure(figsize=(10,6))
    colors=sns.color_palette("Set2",n_colors=len(average_charges_by_gender))
    sns.barplot(x=gender,y=charges, data=average_charges_by_gender,palette=colors)
    sns.set_theme(style='whitegrid')
    plt.xlabel('Gender', fontsize=14)
    plt.title('Average Charges by Gender', fontsize=16)
    for index, value in enumerate (average_charges_by_gender[charges]):
        plt.text(index, value + .01 , f'{value:.2f}', ha= 'center', va='bottom', fontsize=12)
    plt.show()
compare_charges_by_gender(df, 'charges', 'gender')
```

```python
from sklearn.preprocessing import LabelEncoder
from scipy.stats import chi2_contingency
## plotting correlation heatmap to visually show the correlation between all the categorical columns data
def chi_square_heatmap(df):

    categorical_cols= df.select_dtypes(include=['object']).columns
    label_encoder= LabelEncoder()
    df[categorical_cols]= df[categorical_cols].apply(label_encoder.fit_transform)
    chi2_matrix= pd.DataFrame(index=categorical_cols,columns=categorical_cols,dtype=float)

    for row in categorical_cols:
        for col in categorical_cols:
            if row== col:
                chi2_matrix.loc[row,col]=1.0
            else:
                contingency_table=pd.crosstab(df[row],df[col])
                _, p_value,_, _ = chi2_contingency(contingency_table)
                chi2_matrix.loc[row,col]=p_value
    plt.figure(figsize=(15,13))
    sns.set(font_scale=1.5)
    sns.heatmap(chi2_matrix, annot=True, cmap='coolwarm', fmt=".4f", linewidths=0.5,  vmin=0, vmax=1,square=True,
                cbar_kws={"shrink":0.75}, xticklabels=chi2_matrix.columns)
    plt.title("Chi-Square Test P-Values Heatmap")
    plt.show()

print(chi_square_heatmap(df))

## plotting correlation heatmap to visually show the correlation between all the numerical columns data.
def numerical_data_heatmap(df, numerical_columns):
    new_df=df[numerical_columns]
    correlation_matrix= new_df.corr()

    plt.figure(figsize=(12,10))
    sns.heatmap(correlation_matrix, annot=True, cmap='viridis', fmt='.4f', linewidths=.5)
    plt.title('Correlation Heatmap of Columns with Numerical Data')
    plt.show()

print(numerical_data_heatmap(df,['age','bmi','children','charges']))


### plotting boxplot that represents the distribution of charges by coverage level.
def distribution_of_charges_by_coverage_level(df, coverage_level,charges):
    plt.figure(figsize=(10,6))
    colors= sns.color_palette("Set3",n_colors=len(df[coverage_level].unique()))
    sns.boxplot(x=coverage_level,y=charges, data=df, palette=colors)
    sns.set_theme(style='whitegrid')
    plt.xlabel('Coverage Level', fontsize=14)
    plt.ylabel('Charges',fontsize=16)
    plt.title('Distribution of Charges by Coverage Level', fontsize=16)
    plt.show()

distribution_of_charges_by_coverage_level(df, 'coverage_level', 'charges')
```