BHANU PRATAP REDDY

CAPSTONE PROJECT – FINAL REPORT

11/06/2023

TABLE OF CONTENTS

PROBLEM STATEMENT 4
DATA DICTIONARY 4
NEED FOR THE STUDY 6
BUSINESS OPPORTUNITY 6
DATA SUMMARY 7
EXPLORATORY DATA ANALYSIS 8
MISSING VALUES 8
OUTLIERS 11
VARIABLE TRANSFORMATION 13
UNIVARIATE ANALYSIS: CATEGORICAL VARIABLES 14
UNIVARIATE ANALYSIS: NUMERICAL VARIABLES 20
ADDITION OF VARIABLES 24
BIVARIATE ANALYSIS: 26
CLUSTERING 34
INSIGHTS FROM CLUSTERING 38
MODEL BUILDING PRE-REQUISITES 39
PRE-PROCESSING 40
ONE HOT ENCODING, SCALING & TRAIN TEST SPLIT 42
MODEL BUILDING 43
LINEAR REGRESSION 43
POLYNOMIAL REGRESSION 44
DECISION TREE REGRESSION 44
XGBOOST 45
ENSEMBLE TECHNIQUES 46
MODEL METRICS 49
MODEL INTERPRETEATION 51
BUISNESS RECCOMENDATION 53

TABLE OF FIGURES

Figure 1: First 5 values of the dataframe 7
Figure 2: Data given as is info........ 7
Figure 3: Null Values in a column represented as a percentage of total values in column. 8
Figure 4: Null values in each column........ 8
Figure 5: data Info after preliminary treatment........ 9
Figure 6: Null values after Imputation. 10
Figure 7: Boxplot of all numerical variables........ 11
Figure 8: Outliers as a percentage of total values in numerical column. 11
Figure 9: Total no. of outliers in each numerical column........ 11
Figure 10: Boxplot of numerical variables after outlier treatment........ 12
Figure 11: Value count of entries in categorical variable - I........ 13
Figure 12: Value count of entries in categorical variable - II........ 13
Figure 13: Distribution of categorical variables - I........ 14
Figure 14: Distribution of categorical variables - II........ 15
Figure 15: Distribution of categorical variables - III........ 16
Figure 16: Distribution of Numerical Variables - I........ 20
Figure 17: Distribution of Numerical Variables - II........ 21
Figure 18: Distribution of Numerical Variables - III........ 22
Figure 19: 8-point data summary of numerical variables. 22
Figure 20: Distribution of 'Obesitty_classsification' column. 25
Figure 21: Years of Insurance with us vs. Last year Health Checkups. 26
Figure 22: BMI vs. Cholesterol levels. 27
Figure 23: Smoking status vs. Doctor Visits in past year........ 28
Figure 24: Age vs. Heart disease HIstory........ 29
Figure 25: Insurance with us vs. Covered by another company. 30
Figure 26: Alcohol vs. Gender 31
Figure 27: Heatmap of all numerical variables. 32
Figure 28: WSS Plot........ 34
Figure 29: Silhouette Score 34
Figure 30: Silhouette Plot 35
Figure 31: Cluster Profile........ 35
Figure 32: VIF of all Numerical Variables........ 40
Figure 33: The columns that would be removed if we only considered VIF > 5 to strictly remove multicollinearity. 40
Figure 34: Data Info after scaling and one hot encoding. 42
Figure 35: Coeff Visualization of Linear regression, Decision Tree, XG Boost, ADA Boost, Random Forest, Boosting. 50

LIST OF TABLES

Table 1: Data Dictionary 4
Table 2: Categorical Variable Clustering Summary........ 36
Table 3: Numerical Variables Clustering Summary........ 37
Table 4: Performance Metrics of all Models (including ensemble). 49

PROBLEM STATEMENT

The objective of this study is to develop a model that can provide the optimum insurance cost for an individual based on their health and habit-related parameters. In the healthcare industry, insurance plays a vital role in ensuring that individuals can receive necessary medical care without facing significant financial burdens. However, the high cost of medical treatment and procedures can cause considerable financial strain, particularly for those who are not covered by insurance.

To mitigate this risk, insurance companies are constantly seeking to optimize insurance costs while still providing comprehensive coverage to their customers. By leveraging data related to an individual's health and habits, this study aims to develop a model that can accurately estimate insurance costs for an individual.

DATA DICTIONARY

***Table 1**: Data Dictionary*

Variable	Business Definition
applicant_id	Applicant unique ID
years_of_insurance_with_us	Since how many years customer is taking policy from the same company only
regular_checkup_lasy_year	Number of times customers has done the regular health check-up in last one year
adventure_sports	Customer is involved with adventure sports like climbing, diving etc.
Occupation	Occupation of the customer
visited_doctor_last_1_year	Number of times customer has visited doctor in last one year

cholesterol_level	Cholesterol level of the customers while applying for insurance
daily_avg_steps	Average daily steps walked by customers
age	Age of the customer
heart_decs_history	Any past heart diseases
other_major_decs_history	Any past major diseases apart from heart like any operation
Gender	Gender of the customer
avg_glucose_level	Average glucose level of the customer while applying the insurance
bmi	BMI of the customer while applying the insurance
smoking_status	Smoking status of the customer
Year_last_admitted	When customer have been admitted in the hospital last time
Location	Location of the hospital
weight	Weight of the customer
covered_by_any_other_co mpany	Customer is covered from any other insurance company
Alcohol	Alcohol consumption status of the customer
exercise	Regular exercise status of the customer
weight_change_in_last_one _year	How much variation has been seen in the weight of the customer in last year
fat_percentage	Fat percentage of the customer while applying the insurance
insurance_cost	Total Insurance cost

NEED FOR THE STUDY

The healthcare industry is critical to the well-being of individuals and society as a whole. The rising cost of medical care and procedures has made it difficult for individuals to access necessary medical care, particularly those who are not covered by insurance. Insurance companies play a vital role in mitigating this risk, but it is essential to optimize insurance costs to ensure that individuals can access the medical care they need without incurring significant financial burdens.

By developing a model that can accurately estimate insurance costs based on an individual's health and habits, insurance companies can provide comprehensive coverage while minimizing the cost to both the company and the individual. This study can help insurance companies better understand the factors that contribute to insurance costs and develop strategies to optimize insurance costs while still providing quality coverage to their customers. Additionally, the model developed in this study can help individuals better understand how their health and habits can impact their insurance costs, motivating them to make healthier choices and reduce their insurance costs.

BUSINESS OPPORTUNITY

The opportunity for this project lies in the potential to optimize insurance costs for individuals in the health care industry. By building a model that accurately estimates insurance costs based on health and habit related parameters, insurance companies can better assess risk and adjust premiums accordingly. This can lead to cost savings for individuals who maintain healthy habits and lifestyles, as well as for insurance companies who can minimize their risk exposure. Additionally, this project has the potential to promote healthy behavior by incentivizing individuals to take care of their health in order to reduce insurance costs. Overall, the successful implementation of this project can benefit both individuals and insurance companies while also promoting a healthier society.

DATA SUMMARY

Before we proceed with anything let us have a look at the data as is in the form given to us.

First, we shall load the data and see if it can be done so properly:

	applicant_id	years_of_insurance_with_us	regular_checkup_lasy_year	adventure_sports	Occupation	visited_doctor_last_1_year	cholesterol_level	daily_avg_steps	age	heart_decs_history	...	smoking_status	Year_l
0	5000	3	1	1	Salried	2	125 to 150	4866	28	1	...	Unknown	
1	5001	0	0	0	Student	4	150 to 175	6411	50	0	...	formerly smoked	
2	5002	1	0	0	Business	4	200 to 225	4509	68	0	...	formerly smoked	
3	5003	7	4	0	Business	2	175 to 200	6214	51	0	...	Unknown	
4	5004	3	1	0	Student	2	150 to 175	4938	44	0	...	never smoked	

Figure 1: *First 5 values of the dataframe*

Let us look at the data info for the given data set to better understand it:

```
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 24 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   applicant_id                  25000 non-null  int64
 1   years_of_insurance_with_us    25000 non-null  int64
 2   regular_checkup_lasy_year     25000 non-null  int64
 3   adventure_sports              25000 non-null  int64
 4   Occupation                    25000 non-null  object
 5   visited_doctor_last_1_year    25000 non-null  int64
 6   cholesterol_level             25000 non-null  object
 7   daily_avg_steps               25000 non-null  int64
 8   age                           25000 non-null  int64
 9   heart_decs_history            25000 non-null  int64
 10  other_major_decs_history      25000 non-null  int64
 11  Gender                        25000 non-null  object
 12  avg_glucose_level             25000 non-null  int64
 13  bmi                           24010 non-null  float64
 14  smoking_status                25000 non-null  object
 15  Year_last_admitted            13119 non-null  float64
 16  Location                      25000 non-null  object
 17  weight                        25000 non-null  int64
 18  covered_by_any_other_company  25000 non-null  object
 19  Alcohol                       25000 non-null  object
 20  exercise                      25000 non-null  object
 21  weight_change_in_last_one_year  25000 non-null  int64
 22  fat_percentage                25000 non-null  int64
 23  insurance_cost                25000 non-null  int64
dtypes: float64(2), int64(14), object(8)
```

Figure 2: *Data given as is info*

The dataset consists of 25,000 rows of insurance policyholders. The data includes personal and medical information, such as age, gender, occupation, cholesterol level, BMI, smoking status, alcohol consumption, exercise habits, and insurance cost. The data was collected through an unspecified methodology and the frequency of data collection is not provided.

The applicant ID column is not necessary for the analysis and will be removed. Additionally, the column 'regular_checkup_lasy_year' has a typo in the name and will be corrected. The BMI column has missing values, with only 24,010 non-null values. The column Year_last_admitted also has missing values, with only 13,119 non-null values.

Overall, the dataset contains relevant information for analyzing the factors that affect insurance cost. However, further data cleaning and analysis may be necessary to address missing values and inconsistencies in the data.

EXPLORATORY DATA ANALYSIS

MISSING VALUES

Before anything we must look into treating the data. Let us first have a look the missing values:

```
applicant_id                        0
years_of_insurance_with_us          0
regular_checkup_last_year           0
adventure_sports                    0
Occupation                          0
visited_doctor_last_1_year          0
cholesterol_level                   0
daily_avg_steps                     0
age                                 0
heart_decs_history                  0
other_major_decs_history            0
Gender                              0
avg_glucose_level                   0
bmi                               990
smoking_status                      0
Year_last_admitted              11881
Location                            0
weight                              0
covered_by_any_other_company        0
Alcohol                             0
exercise                            0
weight_change_in_last_one_year      0
fat_percentage                      0
insurance_cost                      0
```

***Figure 4**: Null values in each column*

```
applicant_id                       0.00
years_of_insurance_with_us         0.00
regular_checkup_last_year          0.00
adventure_sports                   0.00
Occupation                         0.00
visited_doctor_last_1_year         0.00
cholesterol_level                  0.00
daily_avg_steps                    0.00
age                                0.00
heart_decs_history                 0.00
other_major_decs_history           0.00
Gender                             0.00
avg_glucose_level                  0.00
bmi                                3.96
smoking_status                     0.00
Year_last_admitted                47.52
Location                           0.00
weight                             0.00
covered_by_any_other_company       0.00
Alcohol                            0.00
exercise                           0.00
weight_change_in_last_one_year     0.00
fat_percentage                     0.00
insurance_cost                     0.00
```

***Figure 3**: Null Values in a column represented as a percentage of total values in column.*

Based on the .data summary, we have observed that there are missing values in the data set, with "bmi" and "Year_last_admitted" columns having 3.96% and 47.52% missing values respectively.We are to drop the "applicant_id" column as it is not needed in the analysis, and to rename the "regular_checkup_lasy_year" column to "regular_checkup_last_year" for consistency.

The 'Year_last_admitted' column has a high percentage of missing values (47.52%), which makes it difficult to accurately impute the missing values. Additionally, this column may or may not be relevant for our analysis, as it pertains to the year in which the applicant was last admitted to a hospital, and may not have a significant impact on their current health status. Hence, we are dropping this column from our analysis.

By dropping the 'Year_last_admitted' column and imputing the missing values in the 'bmi' column, we can improve the quality of our data and make more accurate predictions based on the remaining features.

Therefore, the data summary after having dropped said columns and correcting the typo:

```
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 22 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   years_of_insurance_with_us      25000 non-null  int64
 1   regular_checkup_last_year       25000 non-null  int64
 2   adventure_sports                25000 non-null  int64
 3   Occupation                      25000 non-null  object
 4   visited_doctor_last_1_year      25000 non-null  int64
 5   cholesterol_level               25000 non-null  object
 6   daily_avg_steps                 25000 non-null  int64
 7   age                             25000 non-null  int64
 8   heart_decs_history              25000 non-null  int64
 9   other_major_decs_history        25000 non-null  int64
 10  Gender                          25000 non-null  object
 11  avg_glucose_level               25000 non-null  int64
 12  bmi                             24010 non-null  float64
 13  smoking_status                  25000 non-null  object
 14  Location                        25000 non-null  object
 15  weight                          25000 non-null  int64
 16  covered_by_any_other_company    25000 non-null  object
 17  Alcohol                         25000 non-null  object
 18  exercise                        25000 non-null  object
 19  weight_change_in_last_one_year  25000 non-null  int64
 20  fat_percentage                  25000 non-null  int64
 21  insurance_cost                  25000 non-null  int64
dtypes: float64(1), int64(13), object(8)
```

Figure 5*: data Info after preliminary treatment*

Now with the help of a KNN imputer module we shall impute the 'bmi' column. The result:

```
years_of_insurance_with_us        0
regular_checkup_last_year         0
adventure_sports                  0
Occupation                        0
visited_doctor_last_1_year        0
cholesterol_level                 0
daily_avg_steps                   0
age                               0
heart_decs_history                0
other_major_decs_history          0
Gender                            0
avg_glucose_level                 0
bmi                               0
smoking_status                    0
Location                          0
weight                            0
covered_by_any_other_company      0
Alcohol                           0
exercise                          0
weight_change_in_last_one_year    0
fat_percentage                    0
insurance_cost                    0
```

***Figure 6**: Null values after Imputation.*

After successfully handling all the missing values in our dataset, our next step is to identify and address any outliers that may exist in the data. This is particularly important because we are working with a continuous variable, the 'insurance_cost' which is the target variable for our problem. As we know, linear regression and polynomial regression are among the commonly used techniques for solving regression problems. However, it is important to note that these techniques are sensitive to outliers. Therefore, it is essential that we carefully examine the data for any potential outliers and take appropriate steps to address them.

Although decision tree and random forest classifiers are less sensitive to outliers, it would be prudent for us to conduct an outlier analysis before proceeding with any model building. This will ensure that we are working with a clean and reliable dataset, which is crucial for the success of our project. By addressing outliers, we can ensure that our models are more accurate and robust, which will ultimately lead to better insights and outcomes.

OUTLIERS

Visualization of the outliers in the numerical data:

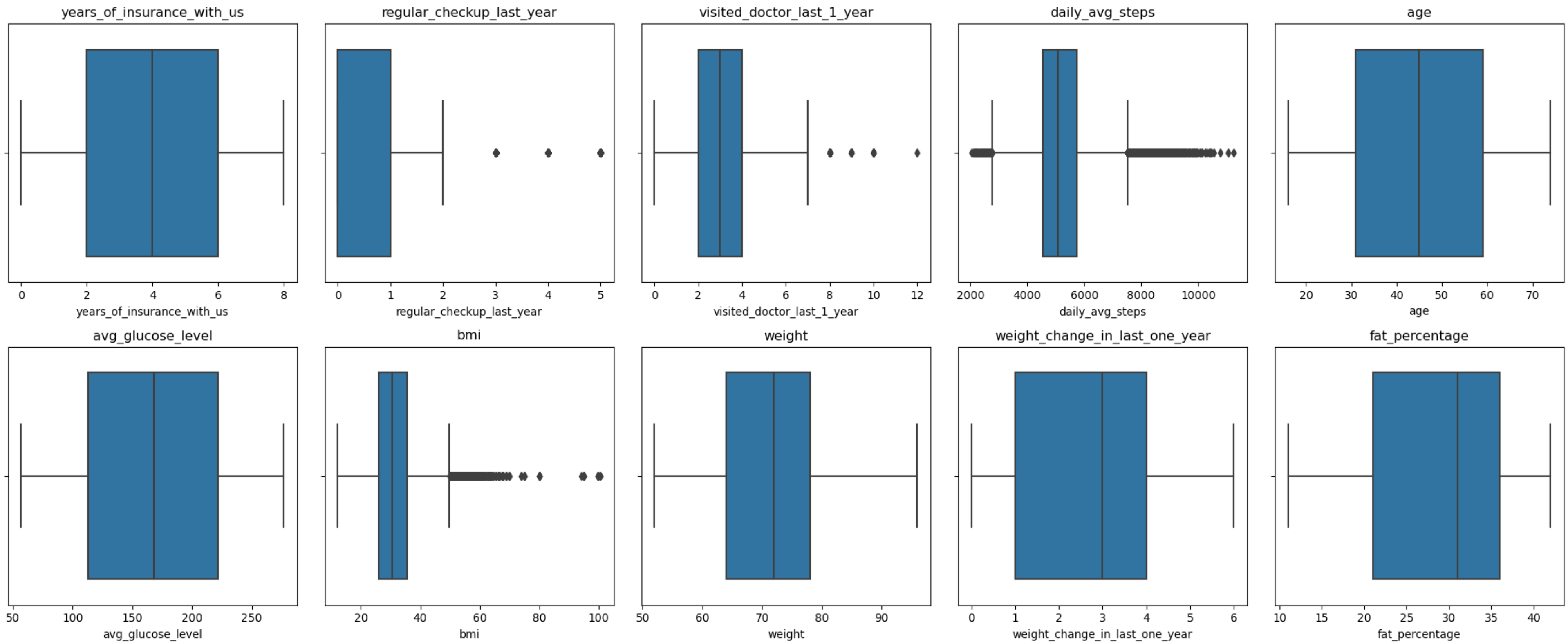

Figure 7: Boxplot of all numerical variables

Numerically this is given as:

years_of_insurance_with_us	0
regular_checkup_last_year	2943
visited_doctor_last_1_year	96
daily_avg_steps	952
age	0
avg_glucose_level	0
bmi	549
weight	0
weight_change_in_last_one_year	0
fat_percentage	0

***Figure 9**: Total no. of outliers in each numerical column.*

years_of_insurance_with_us	0.00
regular_checkup_last_year	11.77
visited_doctor_last_1_year	0.38
daily_avg_steps	3.81
age	0.00
avg_glucose_level	0.00
bmi	2.20
weight	0.00
weight_change_in_last_one_year	0.00
fat_percentage	0.00

***Figure 8**: Outliers as a percentage of total values in numerical column.*

As we can see that whatever outliers present it is of no real significance if we treat it since its treatment will not skew the data to the point at which it becomes a problem.

Hence the visualization after treatment of outlier by clamping them to the Upper and Lower limits:

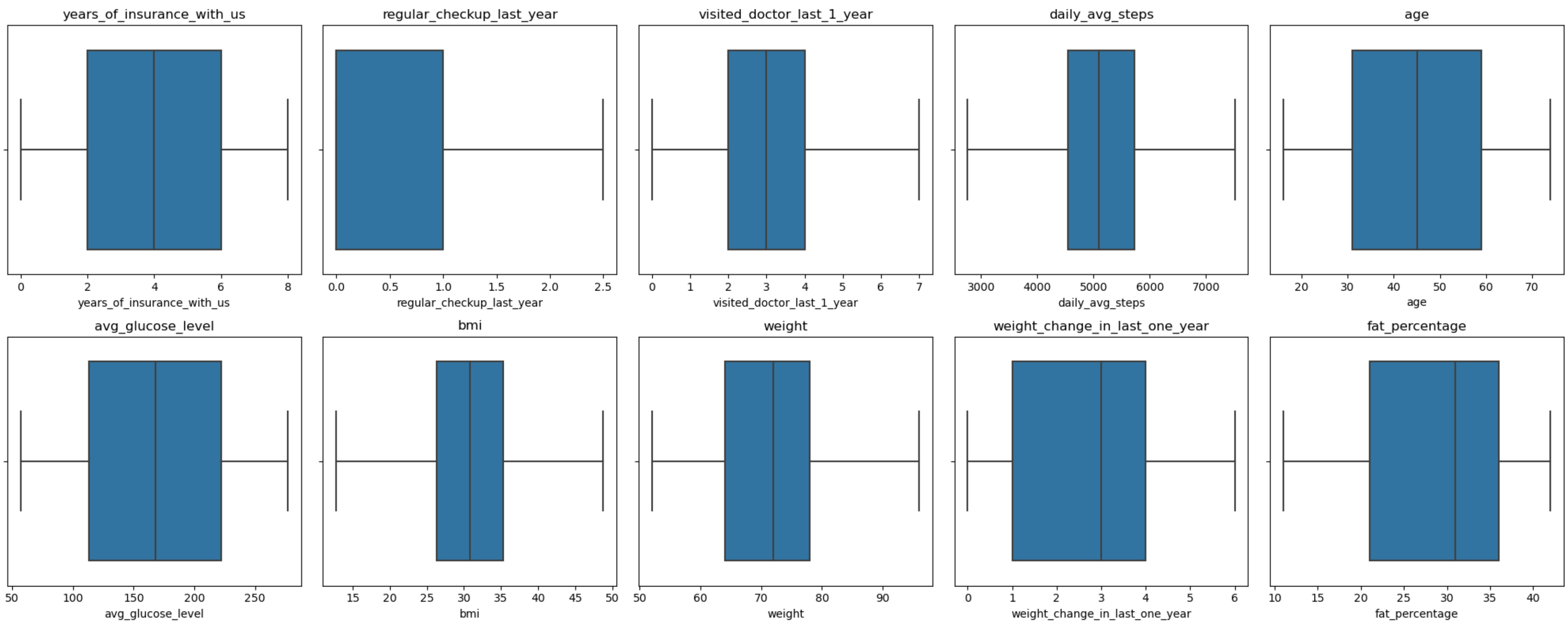

***Figure 10**: Boxplot of numerical variables after outlier treatment.*

VARIABLE TRANSFORMATION

Before we proceed to the univariate analysis of the data some of the categorical variables have the Reponses as '0' and '1' where 1 is affirmative we shall rename the entries of the column appropriately. This is mainly done for 'adventure_sports', 'adventure_sports' and 'other_major_decs_history' column.

The entries of the categorical variables and their value counts are given as:

```
No     22957
Yes     2043
Name: adventure_sports, dtype: int64


Student     10169
Business    10020
Salried      4811
Name: Occupation, dtype: int64


150 to 175    8763
125 to 150    8339
200 to 225    2963
175 to 200    2881
225 to 250    2054
Name: cholesterol_level, dtype: int64


No     23634
Yes     1366
Name: heart_decs_history, dtype: int64


No     22546
Yes     2454
Name: other_major_decs_history, dtype: int64


Male      16422
Female     8578
Name: Gender, dtype: int64
```

***Figure 11**: Value count of entries in categorical variable - I*

```
never smoked       9249
Unknown            7555
formerly smoked    4329
smokes             3867
Name: smoking_status, dtype: int64


Bangalore      1742
Jaipur         1706
Bhubaneswar    1704
Mangalore      1697
Delhi          1680
Ahmedabad      1677
Guwahati       1672
Chennai        1669
Kanpur         1664
Nagpur         1663
Mumbai         1658
Lucknow        1637
Pune           1622
Kolkata        1620
Surat          1589
Name: Location, dtype: int64


N    17418
Y     7582
Name: covered_by_any_other_company


Rare     13752
No        8541
Daily     2707
Name: Alcohol, dtype: int64


Moderate    14638
Extreme      5248
No           5114
Name: exercise, dtype: int64
```

***Figure 12**: Value count of entries in categorical variable - II*

UNIVARIATE ANALYSIS: CATEGORICAL VARIABLES

The univariate analysis of the categorical variables can be visualized as:

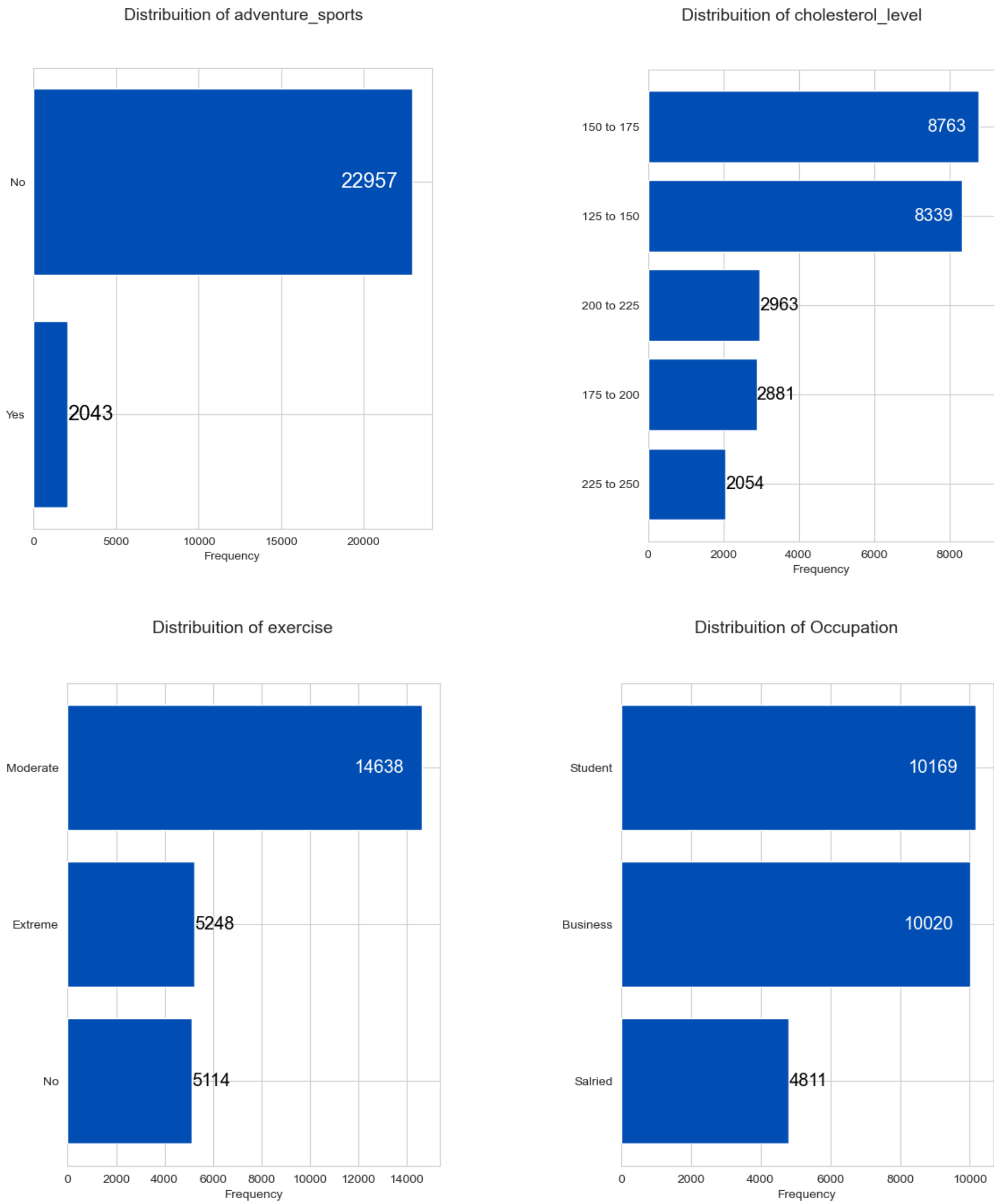

***Figure 13**: Distribution of categorical variables - I*

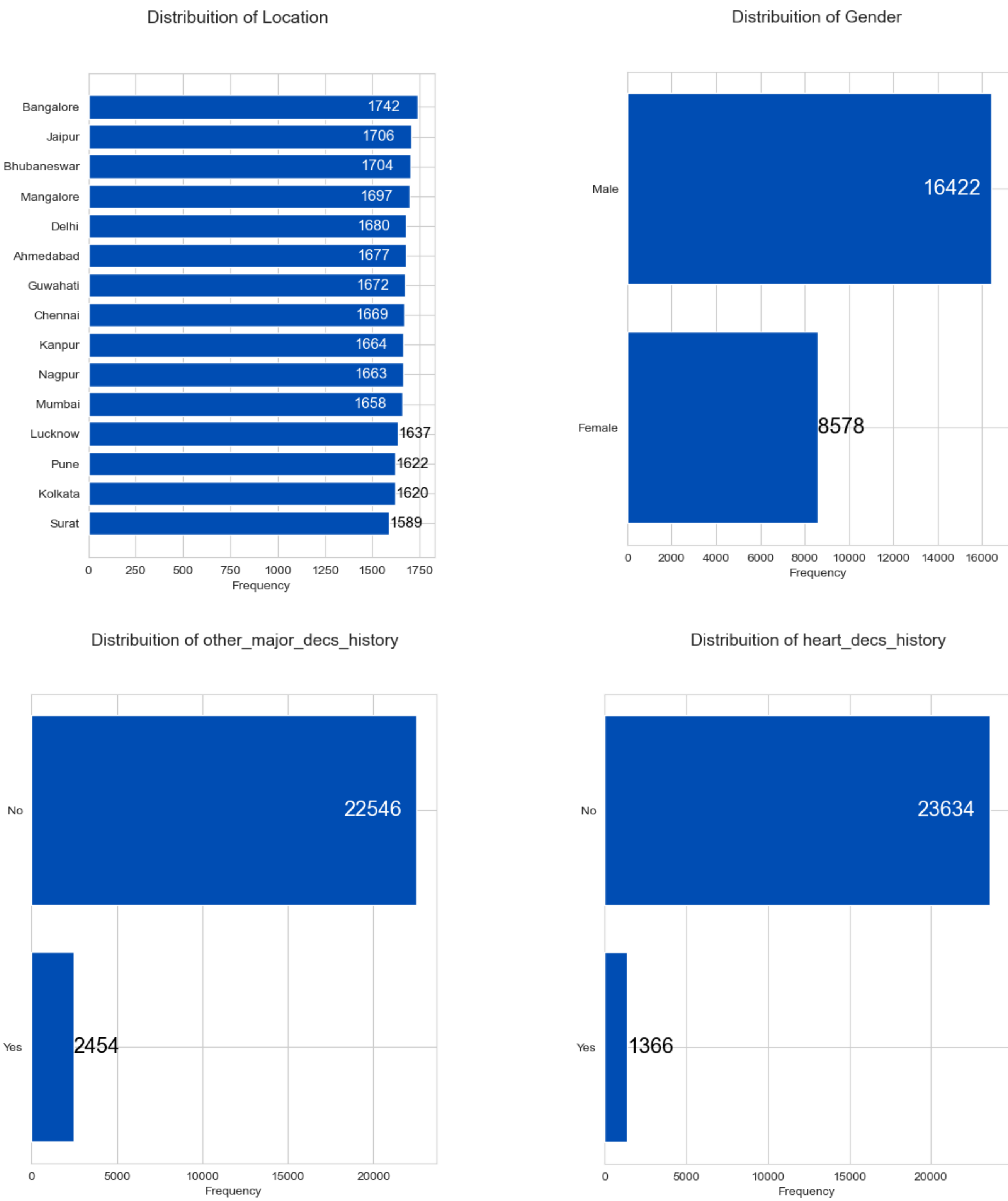

***Figure 14**: Distribution of categorical variables - II*

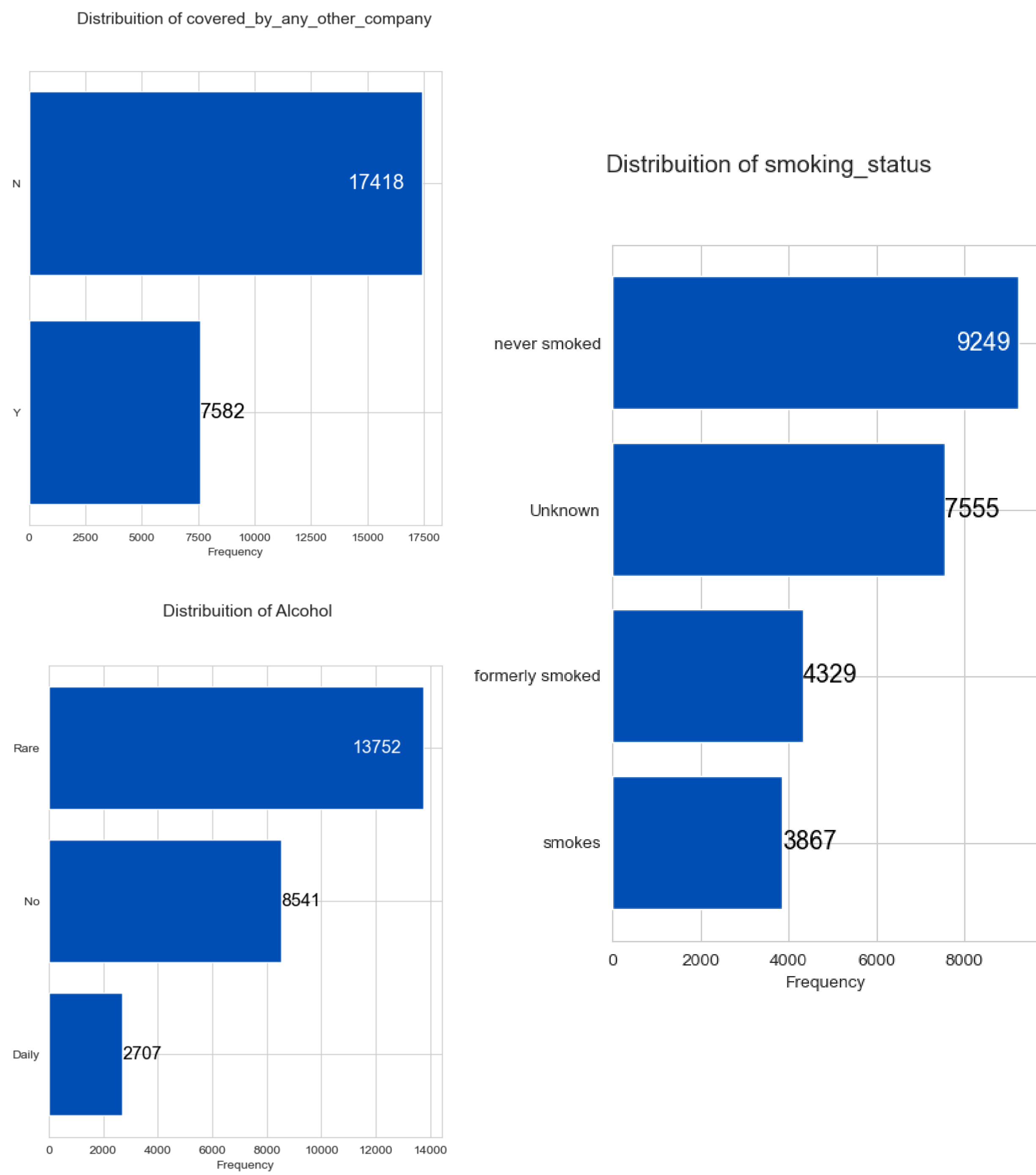

***Figure 15**: Distribution of categorical variables - III*

The Summary of all univariate analysis can be given as:

- Adventure sports:
    - 91.83% of respondents do not participate in adventure sports.
    - 8.17% of respondents participate in adventure sports.
- Occupation:
    - 40.68% of respondents are students.
    - 40.08% of respondents are in business.
    - 19.24% of respondents are salaried.
- Cholesterol level:
    - 35.05% of respondents have a cholesterol level of 150 to 175.
    - 33.36% of respondents have a cholesterol level of 125 to 150.
    - 11.85% of respondents have a cholesterol level of 200 to 225.
    - 11.52% of respondents have a cholesterol level of 175 to 200.
    - 8.22% of respondents have a cholesterol level of 225 to 250.
- Heart disease history:
    - 94.54% of respondents do not have a history of heart disease.
    - 5.46% of respondents have a history of heart disease.
- Major disease history:
    - 90.18% of respondents do not have a history of major diseases.
    - 9.82% of respondents have a history of major diseases.
- Gender:
    - 65.69% of respondents are male.
    - 34.31% of respondents are female.
- Smoking status:
    - 36.99% of respondents have never smoked.
    - 30.22% of respondents have an unknown smoking status.
    - 17.32% of respondents are formerly smokers.
    - 15.47% of respondents are current smokers.

- Location:
    - Respondents are from various locations, with the highest percentages being:
        - Bangalore (6.97%)
        - Jaipur (6.82%)
        - Bhubaneswar (6.81%)
        - Mangalore (6.79%)
        - Delhi (6.72%)
        - Ahmedabad (6.71%)
        - Guwahati (6.69%)
        - Chennai (6.68%)
        - Kanpur (6.66%)
        - Nagpur (6.65%)
        - Mumbai (6.63%)
        - Lucknow (6.55%)
        - Pune (6.49%)
        - Kolkata (6.48%)
        - Surat (6.36%)

- Covered by any other company:
    - 69.67% of respondents are not covered by any other company.
    - 30.33% of respondents are covered by another company.

- Alcohol consumption:
    - 55.01% of respondents consume alcohol rarely.
    - 34.16% of respondents do not consume alcohol.
    - 10.83% of respondents consume alcohol daily.

- Exercise:
    - 58.55% of respondents exercise moderately.
    - 20.99% of respondents exercise extremely.
    - 20.46% of respondents do not exercise.

The inference of the above summary can be given as:

- The univariate analysis includes the distribution of various variables such as Adventure Sports, Occupation, Cholesterol Level, Heart Disease History, Other Major Disease History, Gender, Smoking Status, Location, Covered by Any Other Company, Alcohol and Exercise.
- **The imbalance in some variables such as 'Adventure Sports'**, where 91.8% of the respondents did not participate in adventure sports, may affect the regression model as it may lead to biased predictions.
- **The same applies to 'Heart Disease History'**, where 94.5% of respondents did not have a history of heart disease, and Other Major Disease History, where 90.2% did not have a history of other major diseases.
- The variable '**Covered by Any Other Company' also shows an imbalance**, with 69.7% of respondents not covered by any other company, which may also affect the regression model.
- **Overall, it is important to take into account the imbalances in the variables during regression analysis, as they may result in inaccurate predictions**.

UNIVARIATE ANALYSIS: NUMERICAL VARIABLES

The univariate analysis of the numerical variables can be visualized as:

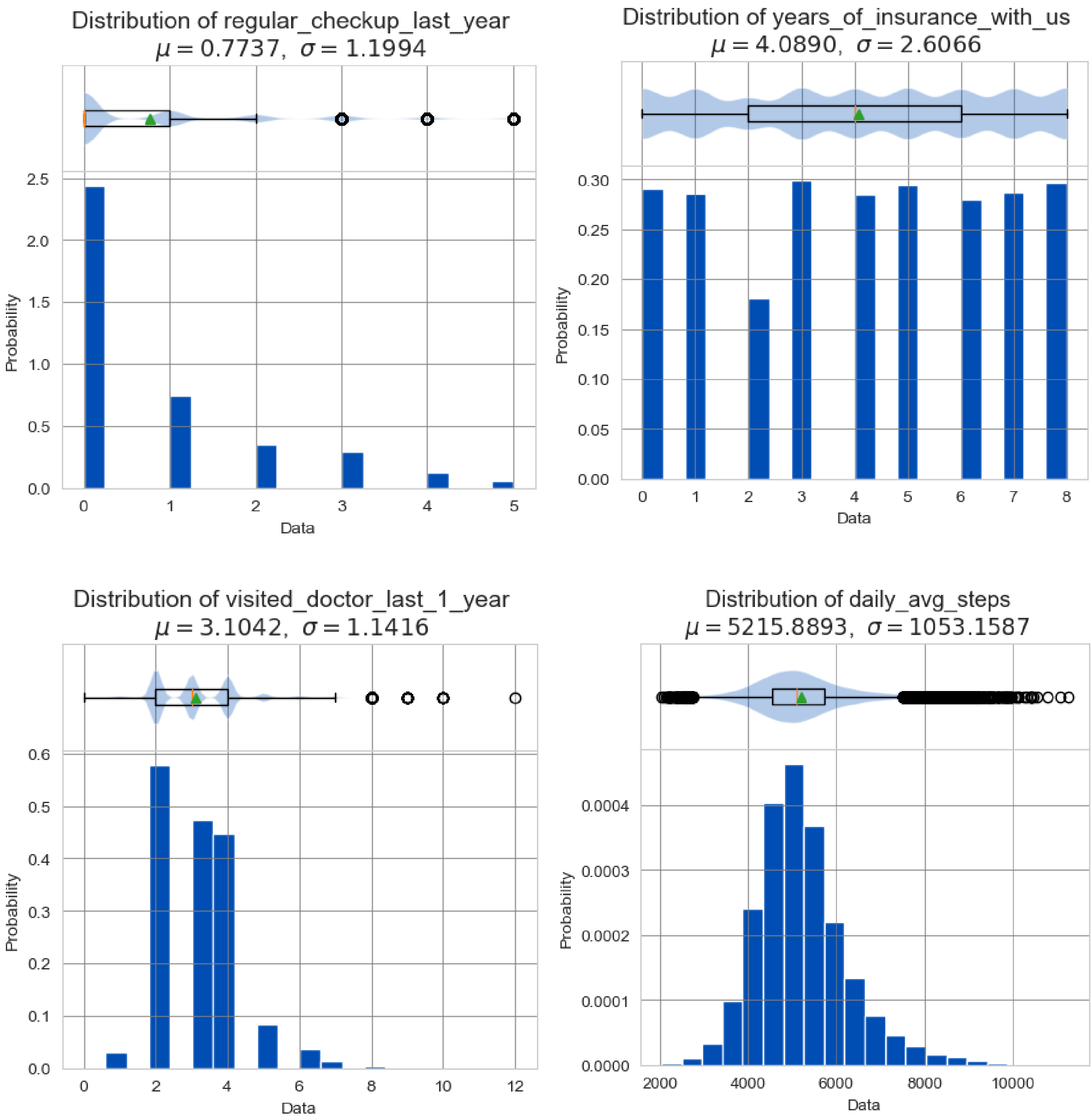

Figure 16: *Distribution of Numerical Variables - I*

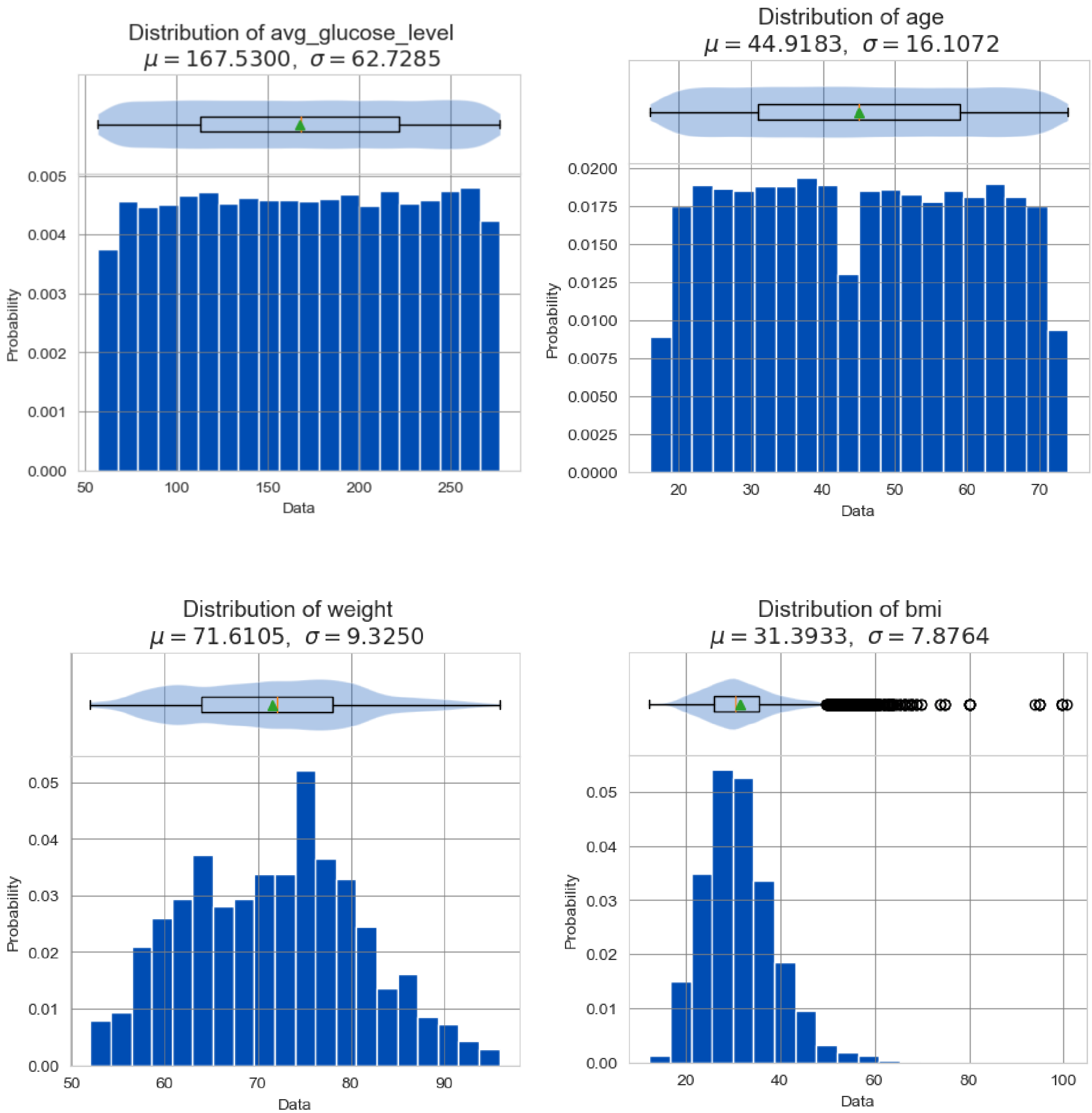

***Figure 17**: Distribution of Numerical Variables - II*

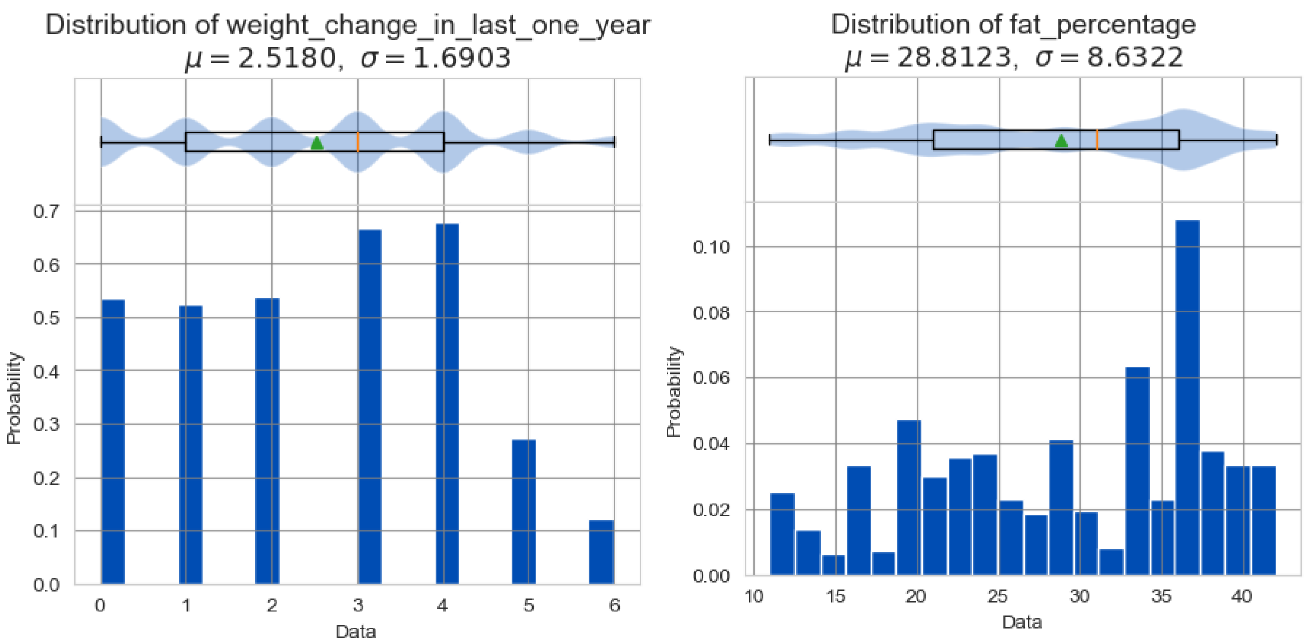

Figure 18*: Distribution of Numerical Variables - III*

The numerical summary of which can be given as:

	count	mean	std	min	25%	50%	75%	max	Skewness	Kurtosis	Shapiro-Wilk Test
years_of_insurance_with_us	25000.0	4.08904	2.606612	0.0	2.0	4.0	6.0	8.0	-0.075212	-1.220693	Non-normal
regular_checkup_last_year	25000.0	0.77368	1.199449	0.0	0.0	0.0	1.0	5.0	1.61081	1.837831	Non-normal
visited_doctor_last_1_year	25000.0	3.1042	1.141663	0.0	2.0	3.0	4.0	12.0	0.978397	1.785771	Non-normal
daily_avg_steps	25000.0	5215.88932	1053.179748	2034.0	4543.0	5089.0	5730.0	11255.0	0.908812	1.853775	Non-normal
age	25000.0	44.91832	16.107492	16.0	31.0	45.0	59.0	74.0	0.013859	-1.176539	Non-normal
avg_glucose_level	25000.0	167.53	62.729712	57.0	113.0	168.0	222.0	277.0	-0.006389	-1.199167	Non-normal
bmi	24010.0	31.393328	7.876535	12.3	26.1	30.5	35.6	100.6	NaN	NaN	Normal
weight	25000.0	71.61048	9.325183	52.0	64.0	72.0	78.0	96.0	0.10907	-0.63815	Non-normal
weight_change_in_last_one_year	25000.0	2.51796	1.690335	0.0	1.0	3.0	4.0	6.0	0.068022	-0.952198	Non-normal
fat_percentage	25000.0	28.81228	8.632382	11.0	21.0	31.0	36.0	42.0	-0.36324	-1.05737	Non-normal

Figure 19*: 8-point data summary of numerical variables.*

Univariate analysis summary:

- The variable 'years_of_insurance_with_us' has a mean of 4.09, standard deviation of 2.61, and ranges from 0 to 8.
- The variable 'regular_checkup_last_year' has a mean of 0.77, standard deviation of 1.20, and ranges from 0 to 5.

- The variable 'visited_doctor_last_1_year' has a mean of 3.10, standard deviation of 1.14, and ranges from 0 to 12.
- The variable 'daily_avg_steps' has a mean of 5,215.89, standard deviation of 1,053.18, and ranges from 2,034 to 11,255.
- The variable 'age' has a mean of 44.92, standard deviation of 16.11, and ranges from 16 to 74.
- The variable 'avg_glucose_level' has a mean of 167.53, standard deviation of 62.73, and ranges from 57 to 277.
- The variable 'bmi' has a mean of 31.39, standard deviation of 7.88, and ranges from 12.3 to 100.6.
- The variable 'weight' has a mean of 71.61, standard deviation of 9.33, and ranges from 52 to 96.

  The variable 'fat_percentage' has a mean of 28.81, standard deviation of 8.63, and ranges from 11 to 42.

Inference:

- All variables have a non-normal distribution except for 'bmi', which has a normal distribution.
- The variable 'daily_avg_steps' has the highest mean among all the variables and ranges from 2,034 to 11,255, indicating that the individuals in the dataset are physically active.
- The variable 'avg_glucose_level' has a mean of 167.53 and ranges from 57 to 277, indicating that the individuals in the dataset may have varying degrees of risk for diabetes.
- The variable 'weight' has a mean of 71.61 and ranges from 52 to 96, indicating that the individuals in the dataset have a healthy weight range.
- The variable 'age' has a mean of 44.92 and ranges from 16 to 74, indicating that the individuals in the dataset have a wide age range.
- The variable 'bmi' has a mean of 31.39 and ranges from 12.3 to 100.6, indicating that the individuals in the dataset have a wide range of body mass index values.

ADDITION OF VARIABLES

Adding variables to the dataset can potentially help in several ways:

1. Improve model performance: Additional variables may provide more information to the model, which may lead to better performance in predicting the outcome variable.

2. Better account for confounding variables: Confounding variables are variables that affect both the independent and dependent variable, and can lead to biased estimates. By including additional variables in the model, we may be better able to account for confounding variables and obtain more accurate estimates.

3. Balance the dataset: The dataset may be unbalanced. Adding additional variables can potentially help balance the dataset and improve the model's ability to make accurate predictions for both classes.

4. Identify interactions: By including interaction terms between variables, we may be able to identify non-linear relationships between the independent and dependent variable that may have been missed otherwise.

Overall, adding variables to the dataset can potentially lead to a better understanding of the relationship between the independent and dependent variable, and may improve the model's ability to make accurate predictions. However, it is important to carefully consider the selection of additional variables and ensure that they are relevant to the research question and not introducing bias.

In this case we will be adding a categorical variable ‘obesity_classification’ where we will utilise ‘fat_percentage’ variable to classify the entry to either ‘Obese’ Or ‘Not Obese’. Why fat percentage and not BMI?

Fat percentage is a better measure for obesity classification than BMI because it provides a more accurate reflection of body fat content. BMI only takes into account height and weight, and can lead to misclassification of individuals with high muscle mass or low body fat. Fat percentage, on the other hand, directly measures the amount of fat in the body and can better identify individuals with excess body fat.

And hence the Univariate analysis after classification:

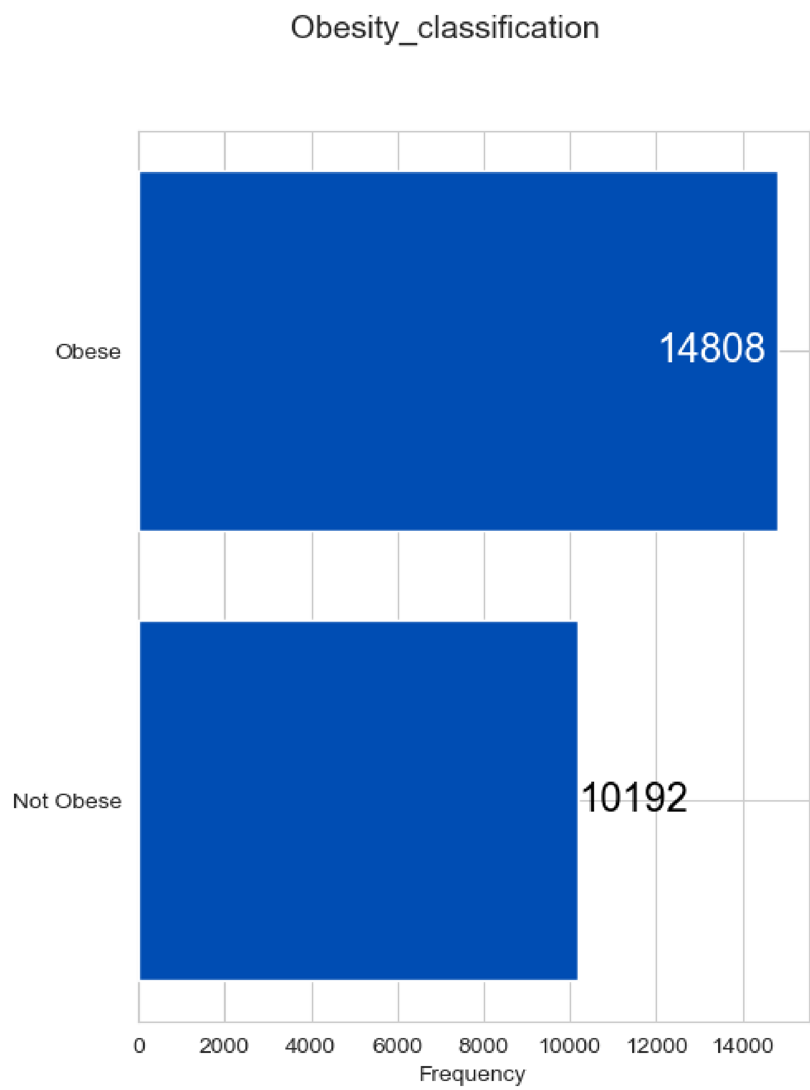

Figure 20*: Distribution of 'Obesitty_classsification' column.*

Similar new variable creation could have been done with 'avg_glucose_level' but in this dataset, the variable "avg_glucose_level" cannot be used to create a new categorical classification for diabetic or not**. This is because the dataset does not state which type of test was used to record the glucose level**. As per the American Diabetes Association, the different types of tests for measuring glucose levels include fasting plasma glucose test, oral glucose tolerance test, and random plasma glucose test. These tests have different diagnostic criteria for diabetes and pre-diabetes. Hence, using the average glucose level as a proxy for diabetes classification without knowing the type of test used would be inaccurate and inappropriate.

BIVARIATE ANALYSIS:

There are many different possible combinations of Bivariate analysis that would yield valuable insights. However, we will only be able analyze a few of them:

- Years of insurance and regular checkup.

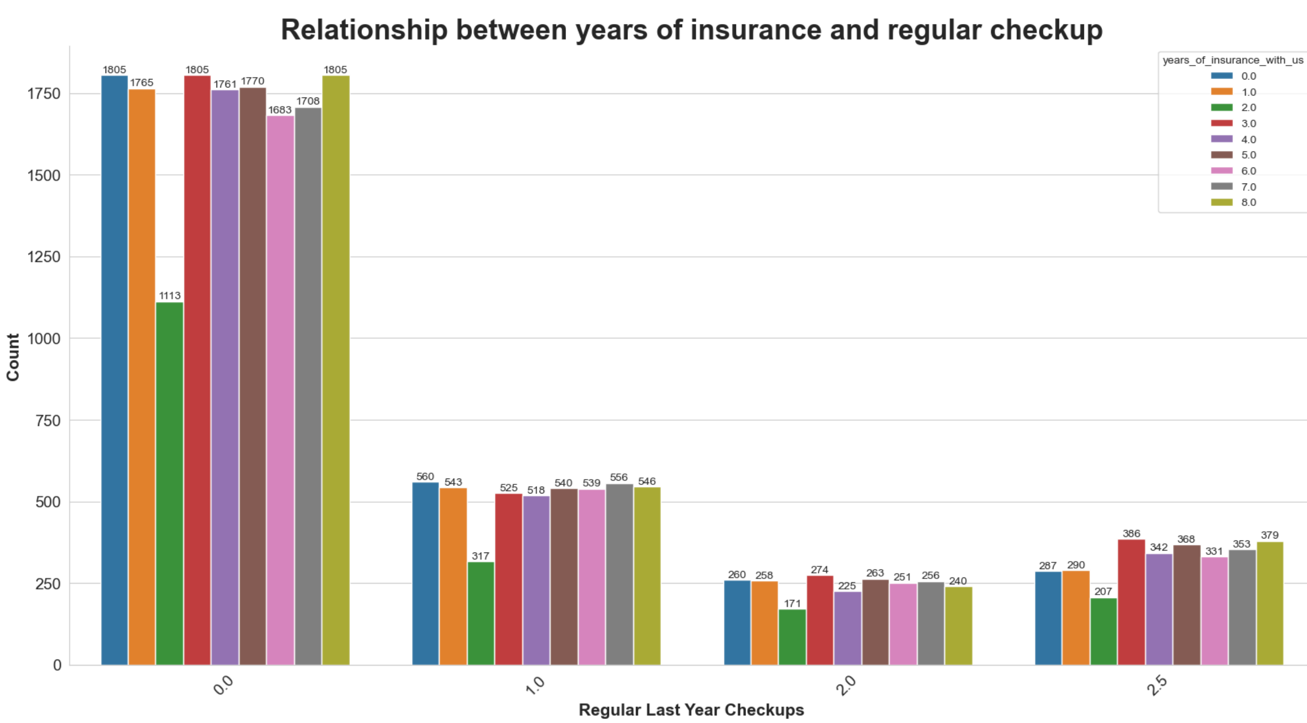

Figure 21*: Years of Insurance with us vs. Last year Health Checkups.*

- The first noticeable trend is the customers with 2 years of insurance are consistently the lowest for the number of regular last year checkups. This is an **indication of pattern** OR **an indication of imbalance** in the data that might have occurred because of data gathering, either way no clear inference can be drawn.

- The second trend noticed is that no matter how many years the customer has had the insurance an overwhelming majority of them did not opt for a single health checkup in the last year.

- Cholesterol levels with respect to BMI.

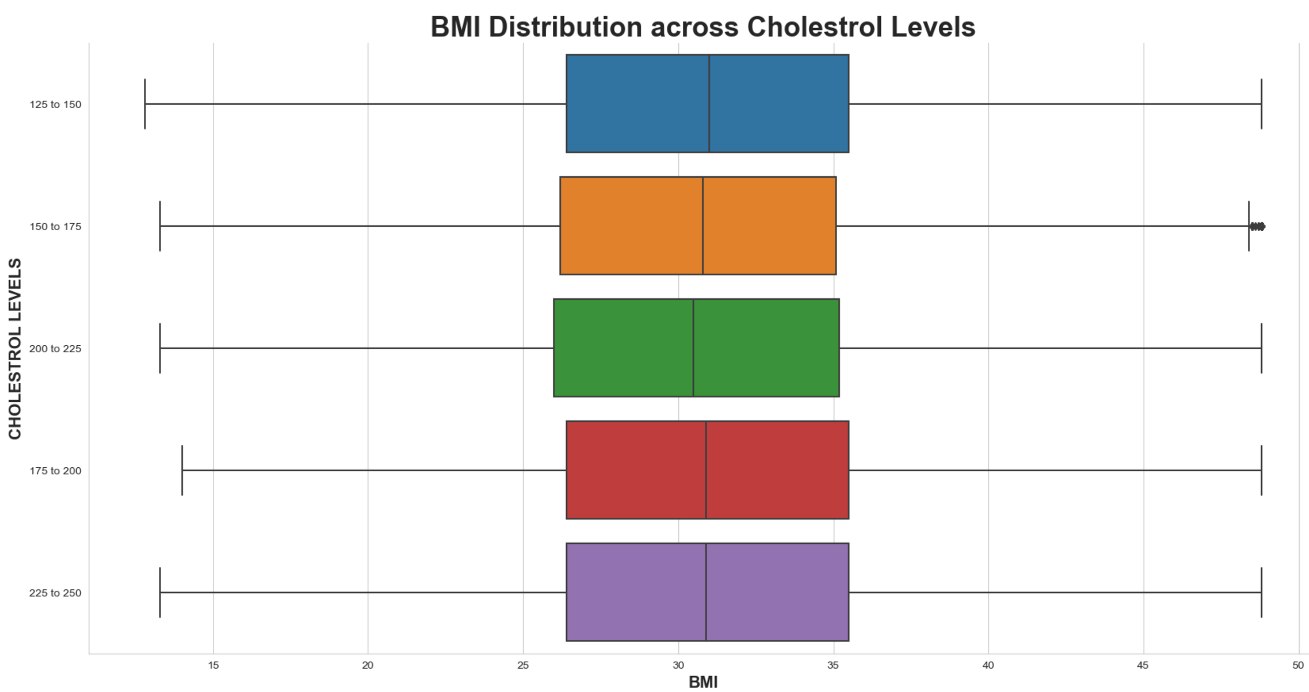

Figure 22: *BMI vs. Cholesterol levels.*

- It appears BMI is not in any way indicative of cholesterol levels and vice versa as we can see the BMI distribution is the same across all levels.

- And that should very much be true as cholesterol levels is an indication of a persons diet hence it will be a key factor in the regression models because of its orthogonality.

- Smoking status vs visitation to the doctor in the past year.

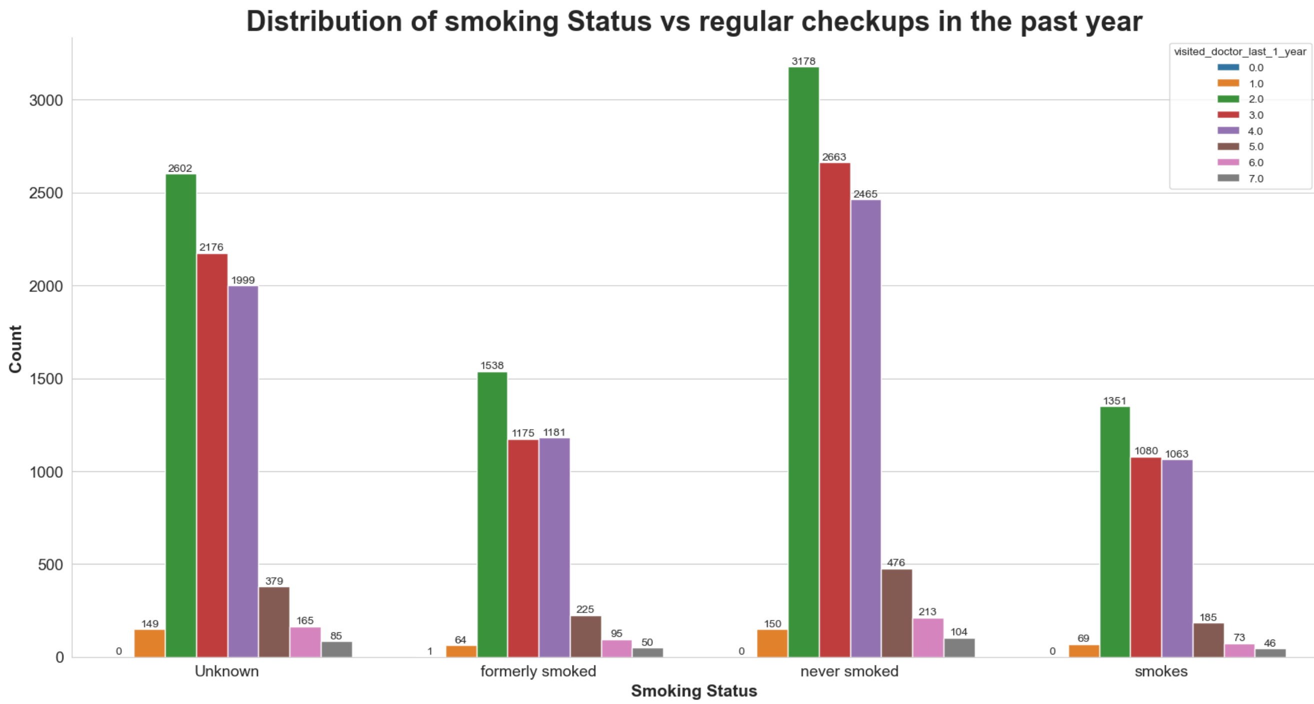

Figure 23*: Smoking status vs. Doctor Visits in past year*

  - Whether smoking or not the trend appears to be the same across all categories

  - There are almost no policy holders who do not go for a doctor.

  - The vast majority of them have either 2, 3 or 4 visits in the past year.

- Heart disease and age.

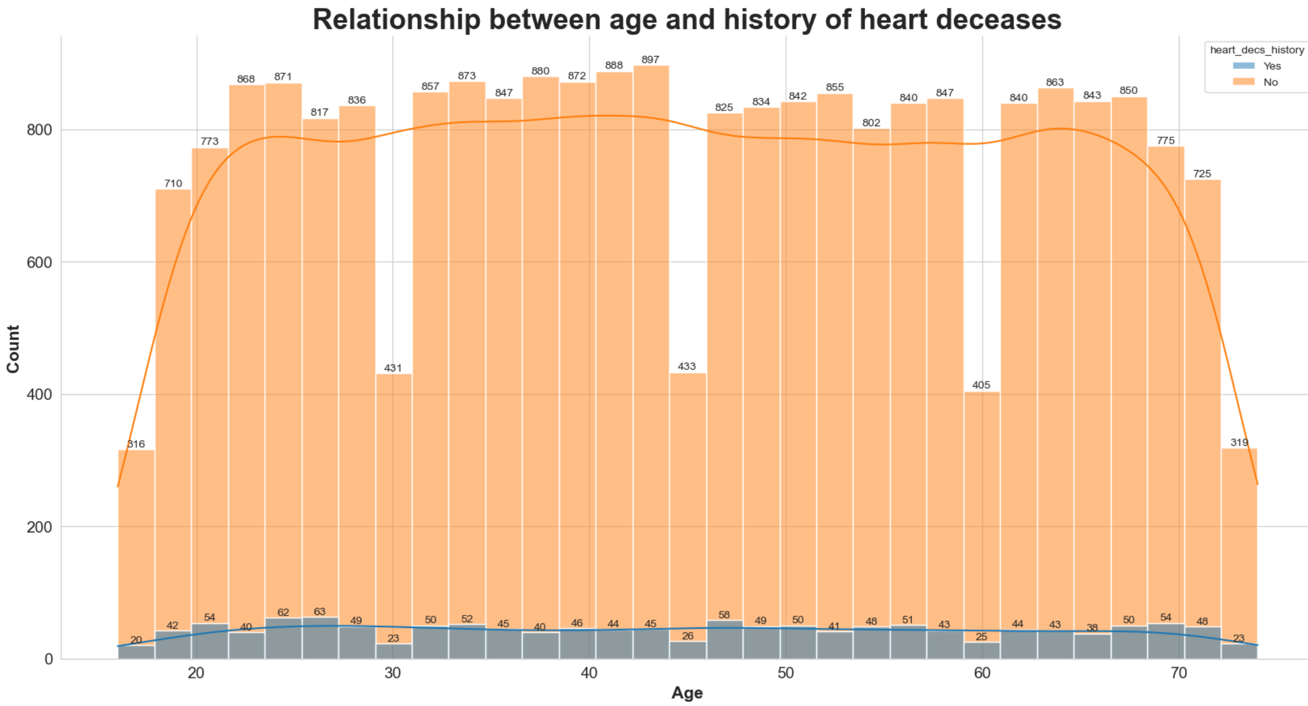

Figure 24: Age vs. Heart disease HIstory

  - There appears to be no trend apart from the sudden drops for particular ages again it my or may not **indicate a pattern** OR **indicate of imbalance** that has occurred because of data gathering.

  - We can also clearly see the unbalanced ratio of No heart diseases with respect to having a history of heart disease.

  - **The imbalances in the variables during regression analysis result in inaccurate predictions**.

- Coverage of Insurance by another company and no. of years with us.

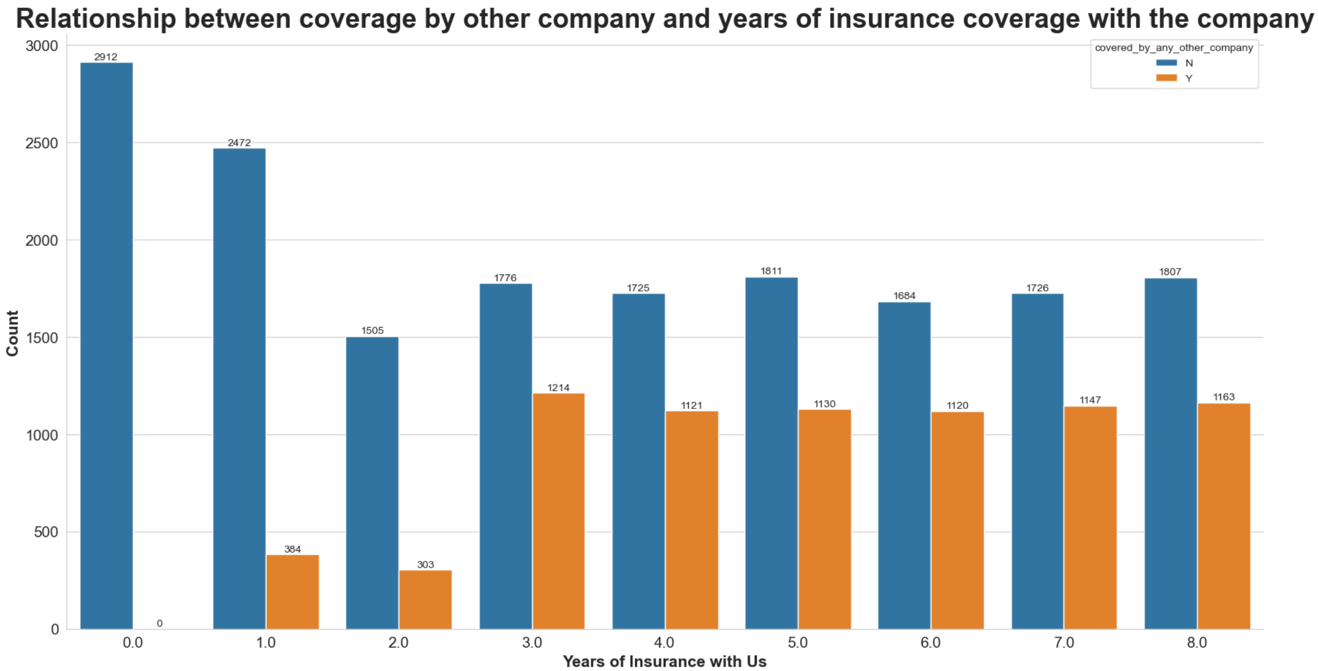

Figure 25: *Insurance with us vs. Covered by another company.*

- The graph shows that the company's performance has been drastically good in the recent years.

- Although there was a dip in the number of new insurers 2 years ago, with the most recent performance the company has effectively doubled the no. of new insurers while losing fewer/gaining more customers form competing companies.

- Gender with respect to alcohol consumption.

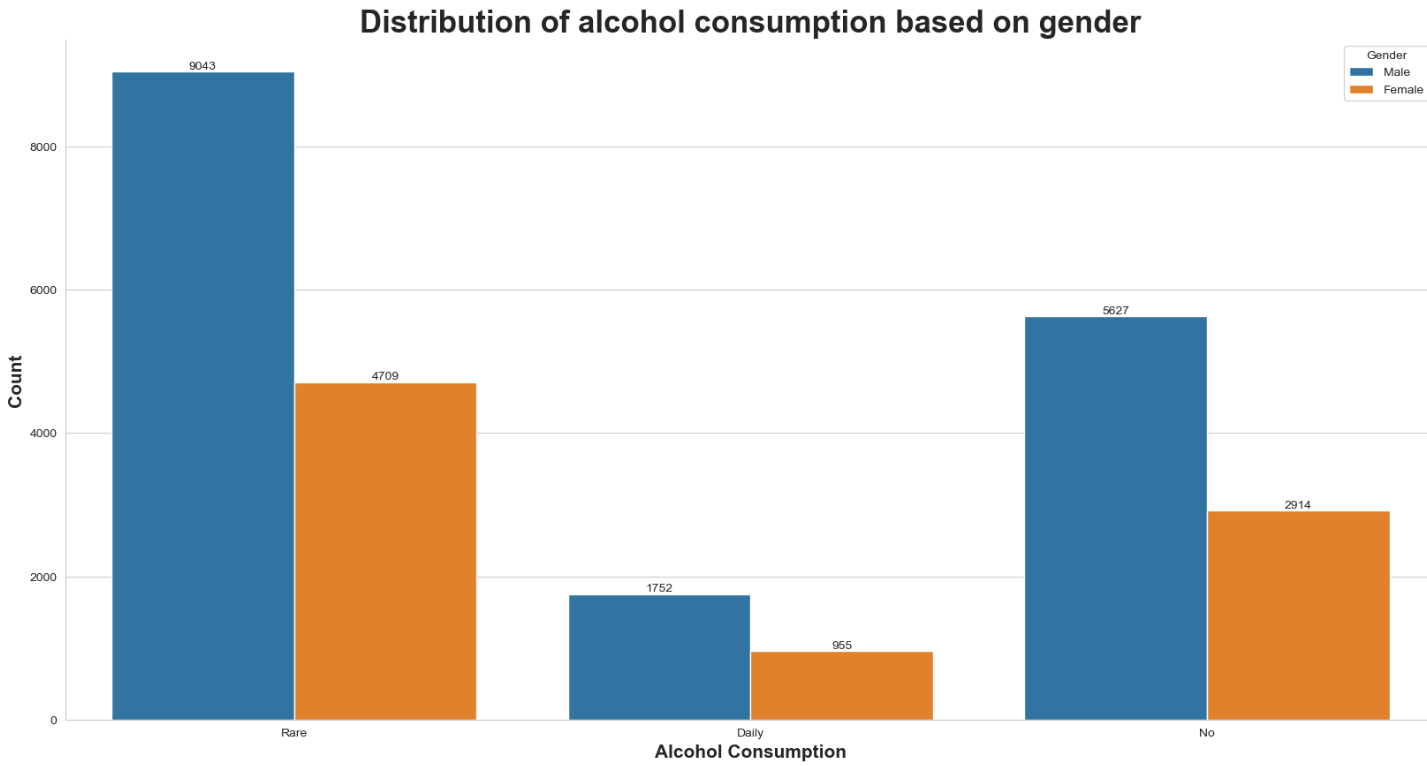

***Figure 26**: Alcohol vs. Gender*

  - A majority number of insurers consume alcohol at least on some occasions.

  - But because of the imbalance in the data for gender we can gain no insight on the trend of alcohol consumption across genders.

  - This is a very good example of how imbalanced data is wiping of any trend because in every category of alcohol consumption the male population is always higher by an order magnitude since data is imbalanced.

- Heatmap indicating the relationship between all numerical variables.

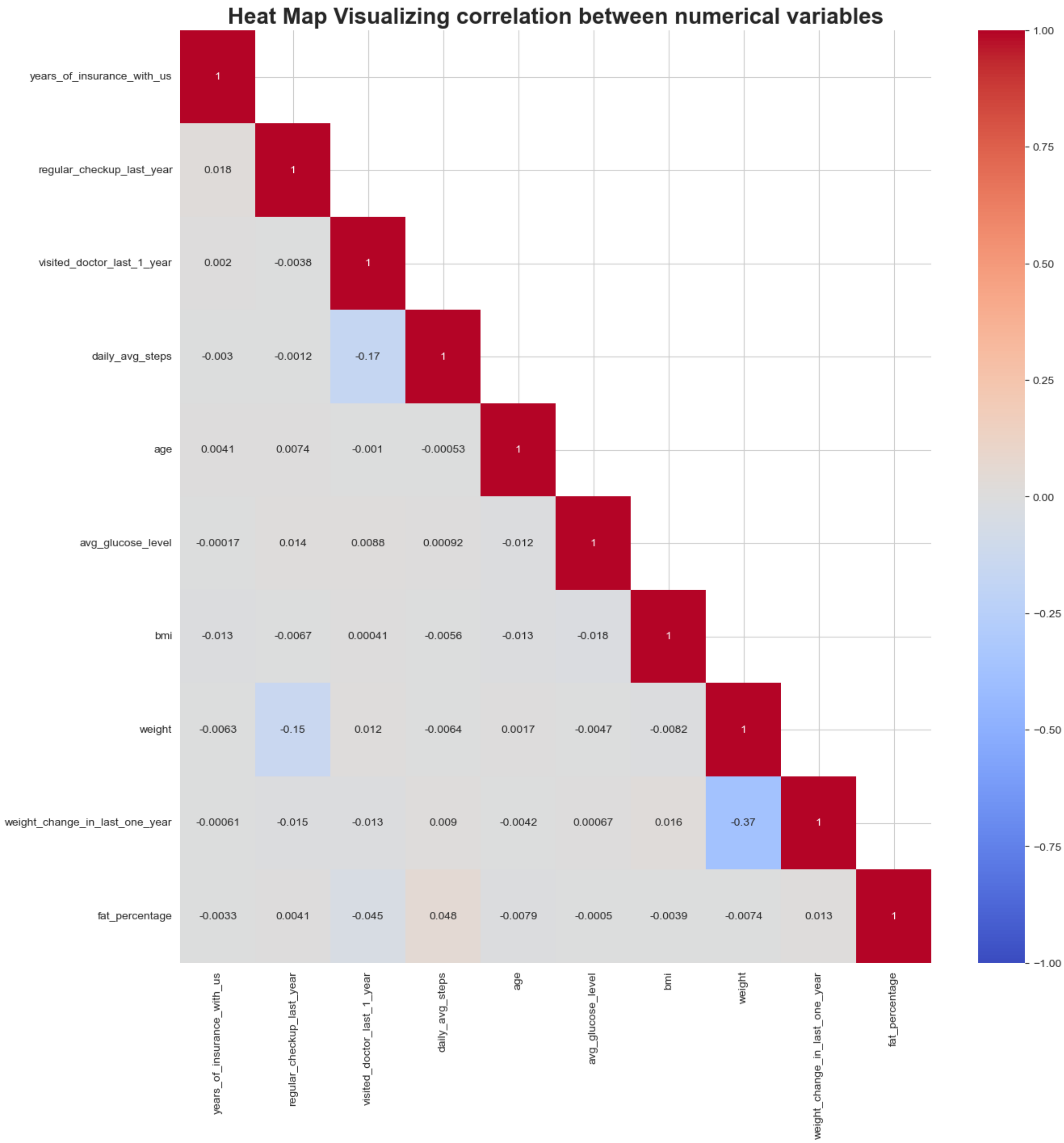

***Figure 27**: Heatmap of all numerical variables.*

- The correlation matrix shows the pairwise correlations between all the continuous independent variables in the dataset. None of the correlations are very strong, except correlation coefficient between weight and weight_change_in_last_one_year is -0.37, which is the strongest negative correlation in the table.

- In terms of interpreting the correlations, we can see that there are no very strong correlations (above 0.7 or below -0.7) between any two variables, indicating that there is no multicollinearity issue. However, there are some moderately strong correlations that are worth noting, such as the negative correlation between weight and weight_change_in_last_one_year, as well as the positive correlation between visited_doctor_last_1_year and daily_avg_steps. These correlations suggest that when one variable increases, the other variable tends to change in a predictable way.

- It's important to keep in mind that correlation does not imply causation, and that further analysis would be needed to determine whether there is a causal relationship between these variables. Additionally, it's important to note that correlation coefficients only capture linear relationships between variables, and that there may be non-linear relationships that are not captured by these coefficients.

CLUSTERING

Based on our initial findings and subsequent findings it's essential to note that an unbalanced categorical variable in the independent variable can lead to biased model performance.

But in order to get more out of the data before we think of any kind of model building let us consider clustering as it might help us put the customers in to the bins that would be suitable for us.

The WSS plot to decide the number of clusters is given as:

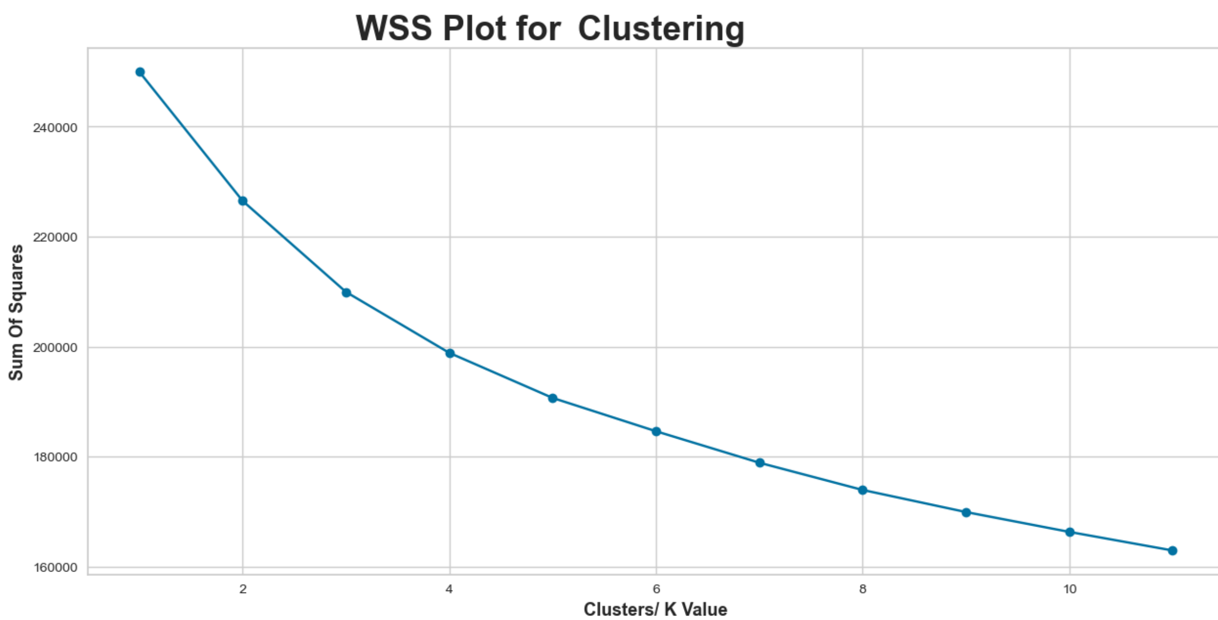

***Figure 28**: WSS Plot*

The corresponding Silhouette Score:

```
Silhouette Score for k =  2 is 0.09516847647539994
Silhouette Score for k =  3 is 0.09765170109148508
Silhouette Score for k =  4 is 0.08620484627044114
Silhouette Score for k =  5 is 0.0834612689238795
Silhouette Score for k =  6 is 0.0806433488593564
Silhouette Score for k =  7 is 0.08025141768802209
Silhouette Score for k =  8 is 0.0764804116378846
Silhouette Score for k =  9 is 0.07268118890838392
Silhouette Score for k =  10 is 0.07281455974182861
Silhouette Score for k =  11 is 0.07462718945655876
```

***Figure 29**: Silhouette Score*

From the Score and plot it would appear 4 or 5 Would be the Ideal cluster number. Let us look at the Silhouette plot (more specifically that of K=5 since it has a more even distribution compared to 4)

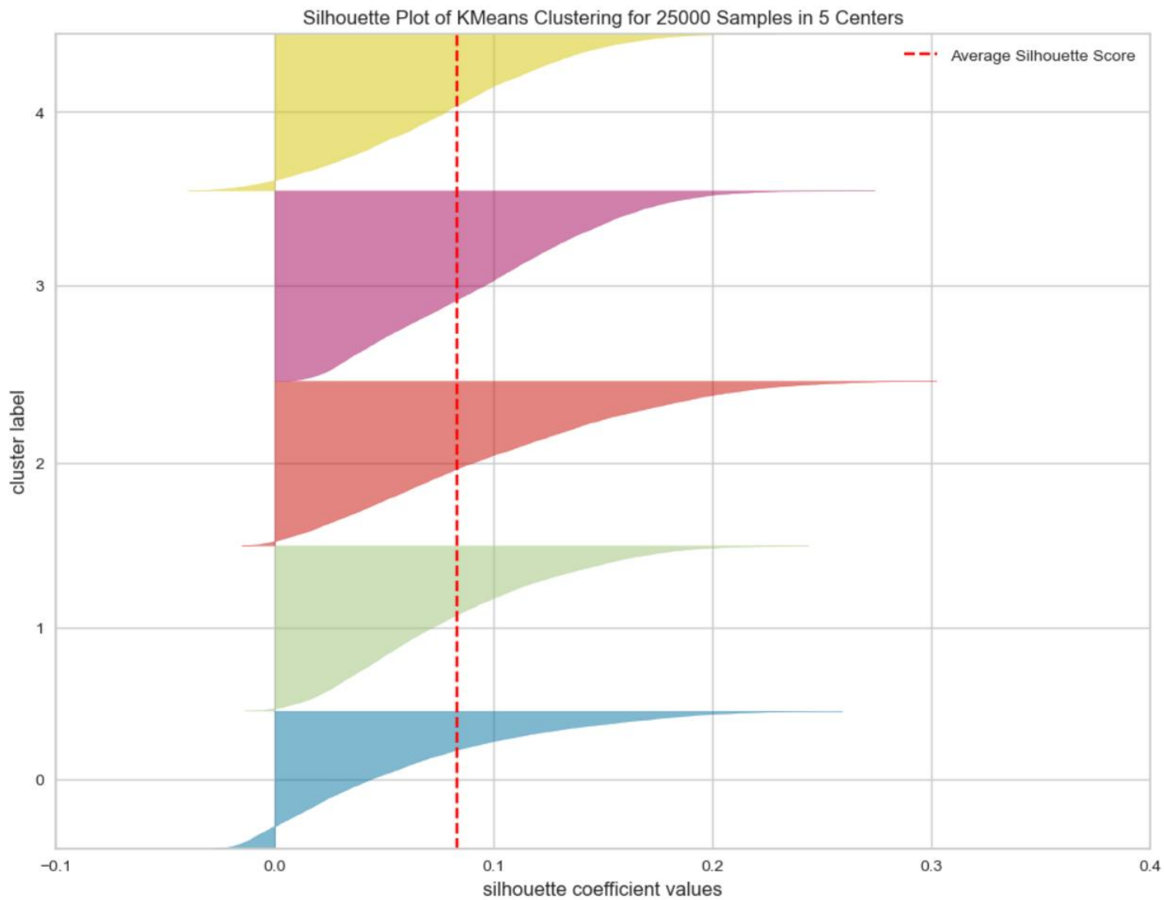

***Figure 30**: Silhouette Plot*

The Cluster profile for mean of the attributes for K = 5 is given as:

Cluster_Label	years_of_insurance_with_us	regular_checkup_last_year	visited_doctor_last_1_year	daily_avg_steps	age	avg_glucose_level	bmi	weight	weight_change_in_last_one_year	fat_percentage	freq
0	4.028416	0.310324	4.579683	4564.170732	43.740942	166.876391	31.148007	74.215960	2.108217	30.165049	4223
1	4.061490	0.242018	2.830508	5314.907075	45.541585	166.738668	31.082482	76.010248	1.882933	18.847261	5074
2	4.060768	0.298694	2.907363	5266.046912	44.695764	167.137767	31.482820	61.349367	4.413302	29.824228	5052
3	4.062639	0.237036	2.504878	5460.757659	45.074277	167.134691	31.270569	76.901078	1.804724	35.628102	5843
4	4.233153	2.280574	3.006240	5193.835379	45.339018	169.831739	31.087919	69.031198	2.423253	28.794093	4808

***Figure 31**: Cluster Profile*

The metrics Can be better summarized for categorical and numerical variables by the following tables:

__Table 2__: Categorical Variable Clustering Summary

Attribute	Athletic (count, top, freq%)	Highly Obese (count, top, freq%)	Lean (count, top,freq%)	Lethargic (count, top, freq%)	Hypochondriac (count, top, freq%)
adventure_sports	5052, No, 95.67%	5843, No, 90.73%	5074, No, 91.31%	4223, No, 89.95%	4808, No, 91.33%
Occupation	5052, Student, 41.77%	5843, Business, 49.26%	5074, Student, 40.16%	4223, Business, 44.88%	4808, Student, 40.59%
cholesterol_level	5052, 150 to 175, 35.90%	5843, 150 to 175, 31.70%	5074, 125 to 150, 42.50%	4223, 150 to 175, 38.11%	4808, 150 to 175, 35.86%
heart_decs_history	5052, No, 94.25%	5843, No, 94.61%	5074, No, 94.41%	4223, No, 94.91%	4808, No, 94.58%
other_major_decs_history	5052, No, 90.07%	5843, No, 90.15%	5074, No, 90.49%	4223, No, 89.25%	4808, No, 90.81%
Gender	5052, Male, 66.58%	5843, Male, 66.06%	5074, Male, 65.80%	4223, Male, 65.14%	4808, Male, 64.71%
smoking_status	5052, never smoked, 37.46%	5843, never smoked, 36.99%	5074, never smoked, 37.41%	4223, never smoked, 36.29%	4808, never smoked, 36.71%
Location	5052, Bhubaneswar, 7.62%	5843, Surat, 7.31%	5074, Jaipur, 7.45%	4223, Guwahati, 7.55%	4808, Bangalore, 10.21%
covered_by_any_other_company	5052, N, 75.24%	5843, N, 68.12%	5074, N, 68.47%	4223, N, 68.86%	4808, N, 67.67%
Alcohol	5052, Rare, 55.15%	5843, Rare, 55.93%	5074, Rare, 59.38%	4223, Rare, 47.80%	4808, Rare, 55.44%
exercise	5052, Moderate, 58.17%	5843, Moderate, 58.32%	5074, Moderate, 58.54%	4223, Moderate, 58.77%	4808, Moderate, 59.04%
obesity_classification	5052, Obese, 64.51%	5843, Obese, 96.48%	5074, Not Obese, 94.48%	4223, Obese, 66.33%	4808, Obese, 58.84%

***Table 3**: Numerical Variables Clustering Summary*

Cluster Label	Years Insured	Checkup	Doctor Visits	Daily Steps	Age	Glucose Level	BMI	Weight	Weight Change	Fat Percentage
Athletic	4.06 (4.00)	0.30 (0.00)	2.91 (3.00)	5266.05 (5158.00)	44.70 (45.00)	167.14 (167.00)	31.48 (31.10)	61.35 (61.00)	4.41 (4.00)	29.82 (32.00)
Highly Obese	4.06 (4.00)	0.24 (0.00)	2.50 (2.00)	5460.76 (5363.00)	45.07 (45.00)	167.13 (167.00)	31.27 (30.80)	76.90 (77.00)	1.80 (2.00)	35.63 (36.00)
Lean	4.06 (4.00)	0.24 (0.00)	2.83 (3.00)	5314.91 (5202.50)	45.54 (45.00)	166.74 (165.00)	31.08 (30.70)	76.01 (76.00)	1.88 (2.00)	18.85 (20.00)
Lethargic	4.03 (4.00)	0.31 (0.00)	4.58 (4.00)	4564.17 (4563.00)	43.74 (43.00)	166.88 (168.00)	31.15 (30.80)	74.22 (74.00)	2.11 (2.00)	30.17 (33.00)
Mitochondriac	4.23 (4.00)	2.28 (2.50)	3.01 (3.00)	5193.84 (5091.50)	45.34 (45.00)	169.83 (171.00)	31.09 (30.70)	69.03 (69.00)	2.42 (3.00)	28.79 (29.00)

The values are in the following format: mean(median) for Table 3.

Let us now gather insights from these summaries and clustering profiles.

INSIGHTS FROM CLUSTERING

Our analysis of the given data facilitated the identification of five distinct customer clusters based on their health and habit-related parameters. The clusters were differentiated on the basis of 'years_of_insurance_with_us', 'regular_checkup_last_year', 'visited_doctor_last_1_year', 'daily_avg_steps', 'avg_glucose_level', 'bmi', 'weight', 'weight_change_in_last_one_year', and 'fat_percentage'. These clusters were subsequently labelled as 'Lethargic', 'Lean', 'Athletic', 'Highly Obese', and 'Hypochondriac' for further analysis.

- Cluster 0 - '**Lethargic**': This group displayed the lowest daily average steps, suggesting a less active lifestyle. Furthermore, this cluster possessed higher 'bmi' and 'fat_percentage' readings, pointing towards a propensity for a sedentary lifestyle. However, the frequency of doctor visits was relatively low, implying a lack of proactive health management.
- Cluster 1 - '**Lean**': The members of this cluster exhibited higher daily average steps, lower weight, and lower 'fat_percentage', which is indicative of a more active lifestyle and healthier physique. This group therefore seems appropriately labeled as 'Lean'.
- Cluster 2 - '**Athletic**': The 'Athletic' cluster also maintained a high level of physical activity, similar to the 'Lean' cluster. Interestingly, this group experienced less weight loss in the past year, but held a comparatively higher 'fat_percentage', potentially indicating a larger muscle mass relative to the other clusters.
- Cluster 3 - **'Highly Obese'**: The members of this group showed the highest weight, 'bmi', and 'fat_percentage' of all the clusters. Alarmingly, despite their elevated health risk indicators, this group's frequency of doctor visits was the lowest amongst the clusters, highlighting a potential area of concern for insurers.
- Cluster 4 - '**Hypochondriac**': Finally, the 'Hypochondriac' cluster demonstrated a very high frequency of doctor visits, paired with moderate readings of weight, 'bmi', and 'fat_percentage'. Despite their intensive health monitoring, their daily physical activity did not significantly exceed the average.

These results offer valuable insights for insurance companies looking to offer personalized premiums and tailored packages based on an individual's health and lifestyle habits.

MODEL BUILDING PRE-REQUISITES

Based on the exploratory data analysis (EDA) results, we can see that the dataset contains a wealth of information, but also some challenges in terms of data imbalance and potential multicollinearity between certain features. As we transition from EDA to model building, it is essential to address these challenges in order to develop an effective predictive model.

- **Pre - Processing**: Before we do anything we need to make sure the data set is ready for model building i.e. separation of independent variables, scaling, train and test split, etc. as these are crucial steps without which whichever model we choose we might be unable to properly Evaluate it.

- **Model Selection**: After setting up our data, the subsequent step is model selection. This decision primarily depends on the nature of our data and the problem we are trying to solve. As we are dealing with a regression problem, we will consider models such as Linear Regression, Polynomial Regression, Decision Tree, Random Forest, Gradient Boosting, and XGBoost.

- **Model Training and Validation**: Once the appropriate model is chosen, it's time to train the model using our dataset. We also need to set aside a part of our data for validation. This validation set will help us assess our model's performance. Cross-validation techniques can be employed for a more robust evaluation. The key metrics for performance evaluation in our context are R^2, Adjusted R and Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE).

- **Model Optimization:** After the initial training, the models may require further tuning to optimize their performance. This could involve adjusting the model's hyperparameters using techniques like Grid Search or Randomized Search.

- **Model Evaluation:** After optimizing the models, we evaluate them on our test data. This gives us a good understanding of how well our model will perform on unseen data.

We will try to compensate for the data imbalance by trying to create better performing models through Ensemble techniques.

As we proceed, it is important to remember that model building is an iterative process. We may need to circle back to data preprocessing or feature engineering as we learn more about the data through the modelling process. It is through this iterative process of building, validating, and tuning models that we will arrive at the most effective solution for predicting insurance costs.

PRE-PROCESSING

So, we start off with VIF to check for Multicollinearity. Upon Calculating the VIF we get:

	column	vif
7	weight	37.896031
3	daily_avg_steps	24.775045
6	bmi	17.828776
9	fat_percentage	11.459090
4	age	8.367299
2	visited_doctor_last_1_year	8.197861
5	avg_glucose_level	7.789625
0	years_of_insurance_with_us	3.409032
8	weight_change_in_last_one_year	3.281529
1	regular_checkup_last_year	1.512341

***Figure 32**: VIF of all Numerical Variables*

And after Removal of columns with VIF > 5 We have:

```
Removed column weight with VIF 37.89603097815191
Removed column daily_avg_steps with VIF 19.75005608291131
Removed column bmi with VIF 13.538284172285266
Removed column fat_percentage with VIF 8.580195999615377
Removed column age with VIF 6.361244349599891
Removed column visited_doctor_last_1_year with VIF 5.492837348233674
```

	column	vif
2	avg_glucose_level	3.793536
0	years_of_insurance_with_us	2.828716
3	weight_change_in_last_one_year	2.674381
1	regular_checkup_last_year	1.459899

***Figure 33**: The columns that would be removed if we only considered VIF > 5 to strictly remove multicollinearity.*

Although this is in general isa good step to do it is not in our case because just removing a high VIF feature like 'weight' might not always be the best solution, especially if it's an important predictor for the target variable. **It may lead to a significant loss of information, which might be why you're seeing a drastic increase in RMSE and MAPE values**.

It's important to know that high **VIF is only a problem if you are interested in the coefficients of your variables**. If your goal is to only predict the target variable, a high VIF might not be an issue. High VIF mainly poses a problem when interpreting the coefficients because it becomes hard to explain the unique effect of a variable while holding other variables constant, which is often the premise of regression analysis. In the world of machine learning where prediction accuracy is more important than interpretation, **it's often acceptable to have multicollinearity.**

Therefore, in our case **WE DO NOT REMOVE ANY COLUMNS** and continue with the analysis.

ONE HOT ENCODING, SCALING & TRAIN TEST SPLIT

The data has been separated into dependent variables and independent variables and to the independent variables now we one hot encode it and scale it. The result:

```
RangeIndex: 25000 entries, 0 to 24999
Data columns (total 43 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   years_of_insurance_with_us      25000 non-null  float64
 1   regular_checkup_last_year       25000 non-null  float64
 2   visited_doctor_last_1_year      25000 non-null  float64
 3   daily_avg_steps                 25000 non-null  float64
 4   age                             25000 non-null  float64
 5   avg_glucose_level               25000 non-null  float64
 6   bmi                             25000 non-null  float64
 7   weight                          25000 non-null  float64
 8   weight_change_in_last_one_year  25000 non-null  float64
 9   fat_percentage                  25000 non-null  float64
 10  adventure_sports_Yes            25000 non-null  float64
 11  Occupation_Salried              25000 non-null  float64
 12  Occupation_Student              25000 non-null  float64
 13  cholesterol_level_150 to 175    25000 non-null  float64
 14  cholesterol_level_175 to 200    25000 non-null  float64
 15  cholesterol_level_200 to 225    25000 non-null  float64
 16  cholesterol_level_225 to 250    25000 non-null  float64
 17  heart_decs_history_Yes          25000 non-null  float64
 18  other_major_decs_history_Yes    25000 non-null  float64
 19  Gender_Male                     25000 non-null  float64
 20  smoking_status_formerly smoked  25000 non-null  float64
 21  smoking_status_never smoked     25000 non-null  float64
 22  smoking_status_smokes           25000 non-null  float64
 23  Location_Bangalore              25000 non-null  float64
 24  Location_Bhubaneswar            25000 non-null  float64
 25  Location_Chennai                25000 non-null  float64
 26  Location_Delhi                  25000 non-null  float64
 27  Location_Guwahati               25000 non-null  float64
 28  Location_Jaipur                 25000 non-null  float64
 29  Location_Kanpur                 25000 non-null  float64
 30  Location_Kolkata                25000 non-null  float64
 31  Location_Lucknow                25000 non-null  float64
 32  Location_Mangalore              25000 non-null  float64
 33  Location_Mumbai                 25000 non-null  float64
 34  Location_Nagpur                 25000 non-null  float64
 35  Location_Pune                   25000 non-null  float64
 36  Location_Surat                  25000 non-null  float64
 37  covered_by_any_other_company_Y  25000 non-null  float64
 38  Alcohol_No                      25000 non-null  float64
 39  Alcohol_Rare                    25000 non-null  float64
 40  exercise_Moderate               25000 non-null  float64
 41  exercise_No                     25000 non-null  float64
 42  obesity_classification_Obese    25000 non-null  float64
```

***Figure 34**: Data Info after scaling and one hot encoding.*

Since, we do not have any more data provided to us for the purposes of model testing and optimization we split our existing data into a train set and attest set.

MODEL BUILDING

Now we first talk about eh models that will be considered for building and what are their pros and cons in context of our data set.

LINEAR REGRESSION

Linear regression is a basic predictive analytics technique. It is used to predict a dependent variable based on the values of one or more independent variables. The predicted values of the dependent variable lie along a straight line when plotted against the independent variables. This straight line is called the regression line and represented by the equation Y= a*X + b, where Y is the dependent variable and X is the independent variable, and a and b being the parameters of the model that are learned.

Pros:

- Simple to implement and interpret.
- Coefficients provide a clear view of the relationship between each feature and the target variable.

Cons:

- Assumes a linear relationship between independent and dependent variables, which might not hold in all cases (e.g., variables like 'cholesterol level', 'age', 'bmi' might not have a linear relationship with the insurance cost).
- Can be sensitive to outliers.

POLYNOMIAL REGRESSION

Polynomial regression is a type of regression analysis in which the relationship between the independent variable and the dependent variable is modeled as an nth degree polynomial. Polynomial regression fits a curved line to your data. Polynomial Regression is a form of linear regression in which the relationship between the independent variable x and dependent variable y is modeled as an nth degree polynomial. Polynomial regression can model relationships between variables that aren't linear.

Pros:

- More flexible than linear regression, capable of modeling complex relationships.

Cons:

- Tends to overfit if the degree of the polynomial is large.
- The choice of the degree of the polynomial can be subjective and might require domain knowledge.

DECISION TREE REGRESSION

A decision tree regression fits a sine curve with additional noisy observation. These models are capable of fitting complex datasets. They segment the predictor space into a number of simple regions, within which the model is a constant. Decision Tree builds regression or classification models in the form of a tree structure. It breaks down a dataset into smaller subsets while at the same time an associated decision tree is incrementally developed.

Pros:

- Can capture complex relationships in the data.
- Doesn't require any assumption about the relationship between variables.

Cons:

- Can easily overfit or underfit the data. This can be controlled using parameters like the depth of the tree.
- Decision trees can become unstable because small variations in the data might result in a completely different tree being generated.

XGBOOST

XGBoost stands for Extreme Gradient Boosting. It is a decision-tree-based ensemble machine learning algorithm that uses a gradient boosting framework. Rather than training all the models in isolation of one another, boosting trains models in succession, with each new model being trained to correct the errors made by the previous ones. Models are added sequentially until no further improvements can be made. Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction.

Pros:

- Often provides one of the most effective machine learning algorithms for structured (tabular) data prediction.
- Handles missing values, and heterogeneous features (different columns require different preprocessing techniques), which is common in this problem.
- Can regularize data, hence is robust to overfitting.

Cons:

- Can be computationally intensive, particularly with large datasets and many iterations.
- May require careful tuning of the parameters.
- The model becomes a bit of a black box with less interpretability.

ENSEMBLE TECHNIQUES

If the results with regular models is not satisfactory, we can get better results with a few ensemble models which can be discussed as:

Random Forest

Random Forest is an ensemble learning method that operates by constructing multiple decision trees at training time and outputting the mean prediction of the individual trees for regression problems. It operates by creating a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Each decision tree in the forest considers a random subset of observations and a random subset of features to split on, which brings in randomness into the model and makes the model more robust and less prone to overfitting.

Pros:

- Generally, it provides a pretty good prediction accuracy due to its ensemble nature.
- Robust to overfitting due to the injection of randomness.
- Can handle categorical and numerical features.

Cons:

- Random forests have been observed to overfit for some datasets with noisy classification/regression tasks.
- They're not easily interpretable like decision trees.

Gradient Boosting

Gradient Boosting is an ensemble machine learning algorithm that's used for classification and regression problems. It produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion, and it generalizes them by allowing optimization of an arbitrary differentiable loss function. Each new tree added to the ensemble attempts to correct the prediction errors made by the trees already present in the ensemble.

Pros:

- Often provides one of the highest accuracies in structured (tabular) datasets.
- Can handle different types of predictor variables (numerical, categorical).

Cons:

- Can be sensitive to noisy data and outliers.
- Learning rate and the number of trees need to be carefully tuned, which can be computationally intensive.

AdaBoost (Adaptive Boosting)

Adaptive Boosting or AdaBoost is one of the simplest boosting algorithms. It creates a strong classifier from a number of weak classifiers by building a model from the training data, then creating a second model that attempts to correct the errors from the first model. Models are added until the training set is predicted perfectly or a maximum number of models are added. The core principle of AdaBoost is to fit a sequence of weak learners (models that are only slightly better than random guessing, such as small decision trees) on repeatedly modified versions of the data.

Pros:

- AdaBoost is easy to implement and does not require to tune a lot of parameters.
- It is resistant to overfitting when low noise is present.

Cons:

- AdaBoost can be sensitive to noisy data and outliers in data.
- It may not work well with a small number of observations.

Bagging

Bagging, or Bootstrap Aggregating, is a simple and very powerful ensemble method. It is the application of the Bootstrap procedure to a high-variance machine learning algorithm, typically decision trees. Bagging works by creating an ensemble of models where each model is trained on a different subset of data. The subsets are created by sampling the data with replacement, which means that some observations may be repeated in each subset. The final prediction is obtained by averaging the predictions of all models for regression problems or by voting for classification problems.

Pros:

- Helps to reduce overfitting by averaging the predictions of multiple models.
- Can handle a large amount of data and features without too much decrease in model performance.

Cons:

- Bagging models can be computationally expensive due to the number of models needed to be trained.
- As with other ensemble models, individual model interpretation is lost.

MODEL METRICS

The Models for all the above discussed were built and their resulting R^2 , Adjusted R, RMSE and MAPE was calculated, the resulting table:

***Table 4**: Performance Metrics of all Models (including ensemble).*

Model/Ensemble	Train Set R^2	Test Set R^2	Train Set R-Adj	Test Set R-Adj	Train Set RMSE	Test Set RMSE	Train Set MAPE	Test Set MAPE
Linear Regression	0.94	0.94	0.94	0.94	3361.96	3369.64	15.14	15.51
Polynomial Regression	0.95	0.94	0.95	0.94	3133.37	3269.04	13.61	14.36
Decision Tree Regression	1	0.90	1	0.90	0	4331.97	0	16.16
XG Boost Regression	0.97	0.95	0.97	0.95	2168.33	3168.60	8.57	12.66
Random Forest Regression	0.99	0.95	0.99	0.95	1175.98	3105.89	4.5825	12.20
Gradient Boosting Regression	0.95	0.95	0.95	0.95	2990.69	3013.70	11.9569	12.10
AdaBoost Regression	0.94	0.94	0.94	0.94	3315.37	3294.67	15.89	15.90
Bagging Regression	0.99	0.94	0.99	0.94	1376.11	3253.85	4.9338	12.74

The Coefficients can also be visualized as:

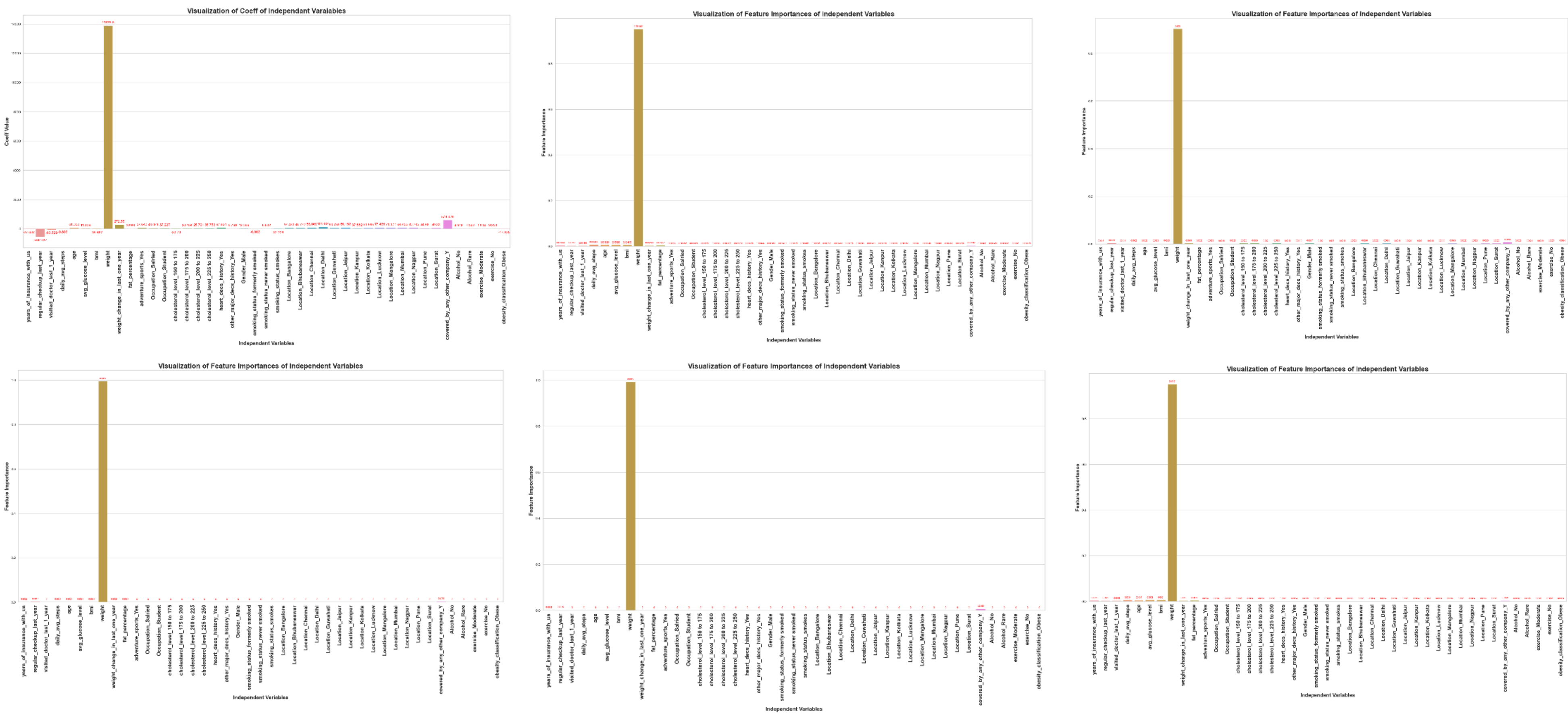

***Figure 35**: Coeff Visualization of Linear regression, Decision Tree, XG Boost, ADA Boost, Random Forest, Boosting.*

As we can see the Coeff of just one variable i.e., 'weight' is many orders of magnitude bigger than the other every other variable. As we discussed in the earlier part of the report this could very well be due to multicollinearity but just as we said in the earlier part of the report: "In the world of machine learning where prediction accuracy is more important than interpretation, **it's often acceptable to have multicollinearity.** "

MODEL INTERPRETEATION

The given Table 4 provides a comparative summary of the performance metrics of several regression models trained to estimate insurance premiums based on health and lifestyle parameters of individuals. These models are evaluated based on their R-Squared (R2), Adjusted R-Squared (R-Adj), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) on both the training and test sets.

1. **Linear Regression**: With a high R2 and R-Adj of 0.94 on both the training and test sets, the model exhibits a strong fit to the data. However, when compared to other models, it has a higher RMSE and MAPE, implying there's still some degree of error in the prediction.

2. **Polynomial Regression**: This model shows a slight improvement over the linear regression model with an R2 and R-Adj of 0.95. The RMSE and MAPE are also lower, indicating a better predictive capability with a lower error rate.

3. **Decision Tree Regression**: The decision tree model shows a perfect fit on the training set, with R2 and R-Adj of 1.00, and no error. However, its performance decreases on the test set (R2 of 0.90), highlighting a possible overfitting issue.

4. **XG Boost Regression**: This model demonstrates impressive predictive performance, with R2 and R-Adj of 0.97 on the training set and 0.95 on the test set. Its RMSE and MAPE are significantly lower than the previous models, making it a strong contender.

5. **Random Forest Regression**: The random forest model displays the best performance so far, with an almost perfect fit on the training set (R2 and R-Adj of 0.99) and very low RMSE and MAPE values. Despite a slight drop in R2 and R-Adj on the test set, it still maintains a solid performance.

6. **Gradient Boosting Regression**: This model showcases an excellent balance between training and test set performance, with R2 and R-Adj of 0.95 on both sets. It also has relatively low RMSE and MAPE values, underscoring its reliable predictive capability.

7. **AdaBoost Regression**: The AdaBoost model has a similar performance to the linear regression model, but with slightly higher RMSE and MAPE values, indicating room for improvement.

8. **Bagging Regression**: While this model exhibits a near-perfect fit on the training set (R2 and R-Adj of 0.99), its performance on the test set decreases (R2 and R-Adj of 0.94), suggesting a potential overfitting issue.

In conclusion, the Random Forest Regression and Gradient Boosting Regression models demonstrate the most promising results, balancing high performance on the training set with robust performance on the test set. However, the choice of the final model should also consider computational efficiency, the complexity of implementation, and the ability to interpret model outputs.

BUISNESS RECCOMENDATION

Finally, from everything we have gathered from this study the recommendations to the insurance company are:

1. **Enhancing Data Collection Practices**: One of the key observations from the analysis was the imbalance in the dataset and the presence of numerous null values. It is crucial for the company to enhance its data collection process to ensure the availability of more balanced and complete data. This would not only improve the accuracy of predictive models but also enable the use of a larger set of variables for predictions.

2. **Obesity as a Key Indicator**: The 'weight' variable proved to be a critical factor in all of the models, indicating the importance of obesity as an indicator of health risks. The company might want to consider focusing more on weight and fat percentage during health check-ups to identify potential high-risk customers. The company could develop programs or incentives to encourage policyholders to maintain a healthy weight, which could reduce potential claims related to obesity-linked health conditions.

3. **Customer Segmentation**: The clustering analysis helped categorize customers into specific bins such as Lethargic, Lean, Athletic, Highly Obese, and Hypochondriac. This classification provides a valuable opportunity for the company to tailor their insurance policies according to these categories. For instance, individuals labelled as 'Athletic' might be less prone to health risks than those labelled 'Highly Obese', and therefore may be offered different premium rates or specific policy benefits.

4. **Model Selection for Predictions**: Given that Random Forest and Gradient Boosting methods performed the best among all models, it is recommended to use these techniques for future predictions. These models provided a good balance between bias and variance, offering robust and accurate predictions.

5. **Continuous Model Refinement**: The development and refinement of prediction models is a continuous process. As more data becomes available and as customer profiles evolve over time, it is important for the company to regularly update their models to reflect these changes. Ongoing model evaluation is crucial to ensure that the models stay relevant and accurate.

These recommendations, if implemented effectively, can help the insurance company in better risk assessment, more accurate insurance cost estimations, improved customer segmentation, and overall, a more **data-driven decision-making process.**