Project Notes
FinalSubmissionREPORT

By: Parthasarathi Swain

Contents

1. Introduction

We all know that Health care is a very important domain in the market. It is directly linked with the life of the individual; hence we have to always be proactive in this particular domain. Money plays a major role in this domain because sometimes treatment becomes super costly and if any individual is not covered under the insurance, then it will become a pretty tough financial situation for that individual.

The companies in medical insurance also want to reduce their risk by optimizing the insurance cost, because we all know a healthy body is in the hand of the individual only. If individuals eat healthy and do proper exercise the chance of getting ill is drastically reduced.

Defining problem statement

The objective of this exercise is to build a model, using data that provides the optimum insurance cost for an individual by using the health and habit-related parameters for the estimated cost of insurance.

Need of the study/project

The need to study the project is required because,

1. Analyze relevant data: To build a predictive model, we need to analyze the data. In this case, we need to analyze data on health and habit-related parameters and their relationship with insurance costs.

2. Define the target variable: In this case, the target variable is the insurance cost.

3. Select appropriate modeling techniques: We need to select appropriate modeling techniques that will help us build an accurate predictive model. We may need to consider techniques such as linear regression, decision trees, or neural networks.

4. Feature engineering: We need to identify the most relevant features or variables that affect insurance costs. We may need to do feature engineering to create new features or transform existing features to make them more relevant for our model.

5. Model evaluation: Once we have built our model, we need to evaluate its performance using appropriate metrics. We need to ensure that our model is accurate and generalizes well to new data.

6. Interpretation: Finally, we need to interpret the results of our model. We need to identify the key factors that affect insurance costs and provide insights that can help insurance companies make better decisions.

Understanding business/social opportunity

1. Improved risk assessment: By using a predictive model, insurance companies can better assess the risk associated with insuring an individual. This can

help them determine appropriate premiums and reduce the likelihood of losses.

2. More personalized insurance plans: A predictive model can help insurance companies create more personalized insurance plans that are tailored to an individual's health and lifestyle. This can lead to higher customer satisfaction and retention.
3. Improved health outcomes: Insurance companies can encourage individuals to adopt healthier habits by providing incentives for healthier lifestyles. This can lead to improved health outcomes and reduced healthcare costs.
4. Competitive advantage: Insurance companies can gain a competitive advantage in the marketplace by leveraging data and analytics to make better decisions. They can differentiate themselves by offering more personalized and affordable insurance plans.
5. Social impact: By providing more affordable and accessible insurance plans, insurance companies can contribute to broader social goals of improving healthcare access and reducing healthcare disparities.

2. EDA and Business Implication

Uni-variate / Bi-variate / multi-variate analysis to understand relationship b/w variables. How your analysis is impacting the business?

Non-visual inspection of data

Viewing the first five records of the dataset.

	applicant_id	years_of_insurance_with_us	regular_checkup_lasy_year	adventure_sports	Occupation	visited_doctor_last_1_year	cholesterol_level	daily_avg_st
0	5000	3	1	1	Salried	2	125 to 150	4
1	5001	0	0	0	Student	4	150 to 175	6
2	5002	1	0	0	Business	4	200 to 225	4
3	5003	7	4	0	Business	2	175 to 200	6
4	5004	3	1	0	Student	2	150 to 175	4

5 rows × 24 columns

	applicant_id	years_of_insurance_with_us	regular_checkup_lasy_year	adventure_sports	visited_doctor_last_1_year	daily_avg_steps	age	heart_
count	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000	25000.000000	
mean	17499.500000	4.089040	0.773680	0.081720	3.104200	5215.889320	44.918320	
std	7217.022701	2.606612	1.199449	0.273943	1.141663	1053.179748	16.107492	
min	5000.000000	0.000000	0.000000	0.000000	0.000000	2034.000000	16.000000	
25%	11249.750000	2.000000	0.000000	0.000000	2.000000	4543.000000	31.000000	
50%	17499.500000	4.000000	0.000000	0.000000	3.000000	5089.000000	45.000000	
75%	23749.250000	6.000000	1.000000	0.000000	4.000000	5730.000000	59.000000	
max	29999.000000	8.000000	5.000000	1.000000	12.000000	11255.000000	74.000000	

By using the describe () function we can get the5-numberr summary of the data, which includes, 25%, 50%, 75%, min, and max also we get the count of the data and mean of the data variables.

From the summary of the data, we can understand a few points like,

- Years_of_insuarance_with_us column has a mean of 4 years, which means the mean duration a customer has been with the company is 4 years, of which 8 years is the maximum record for the association while 0 years which means there are some people who have been associated for less than a year.
- The average age of customer that comes for insurance is 44, while 74 is the maximum and the minimum is 16.
- The mean BMI for an individual is 31, maximum is 100 which sounds suspicious, data has to be checked.
- The average weight is 71, the maximum is 96 and the minimum is 52.
- The average Fat percentage in an individual is 28%, the Maximum is 42% and the minimum is 11%.
- The average insurance cost is 27,147 INR which can go up to the maximum amount of 67,870 INR and a minimum of 2,468 INR.

Checking the shape of the data

```
(25000, 24)
```

- By checking the shape () of the data it gives the output of no. of rows and columns in the dataset. We have 25000 data points and 24 columns.

Checking the info () of the data

```
 #   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)
memory usage: 4.6+ MB
```

From the info () command we can see that:

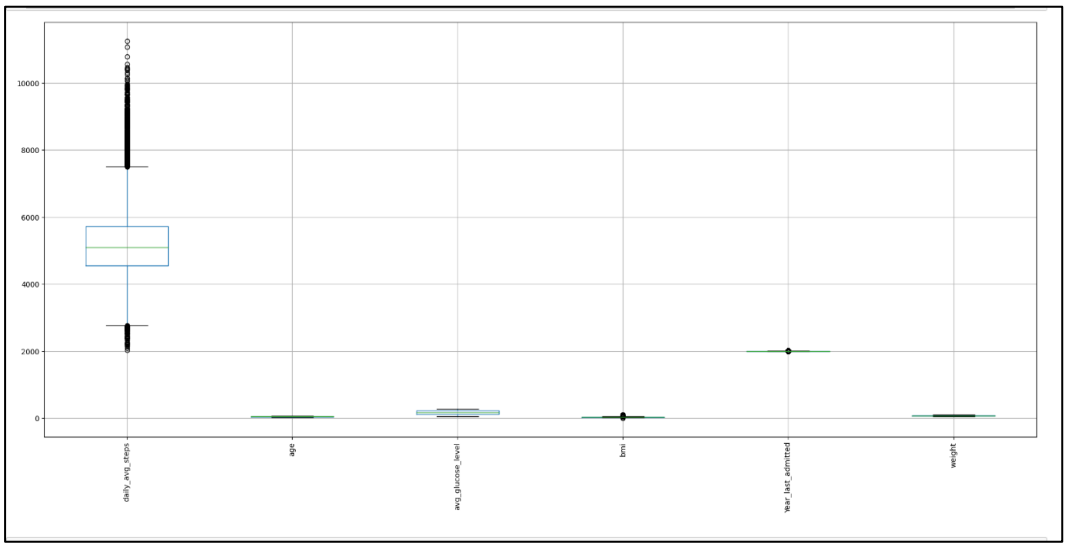
- We have 2 columns with float datatype,14 columns with integer datatype, and 8 columns with object datatype.
- Columns such as 'Bmi', and 'Year_last_admitted' have some null values.

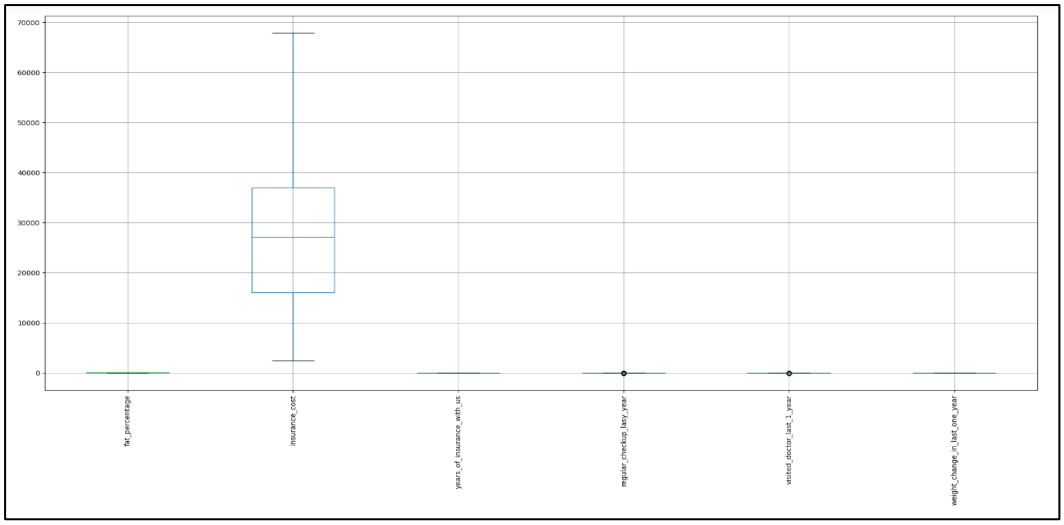
Checking the Null and duplicate values in the data

```
applicant_id                        0
years_of_insurance_with_us          0
regular_checkup_lasy_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
dtype: int64
```

- We have 990 in the BMI column and 11881 missing values in the Year_last_submitted column in the dataset.
- 39% of data in the BMI column is missing and 48% of data from the year_last_admitted column is missing.
- We have 0 duplicate values in the dataset.

UnivariateAnalysis



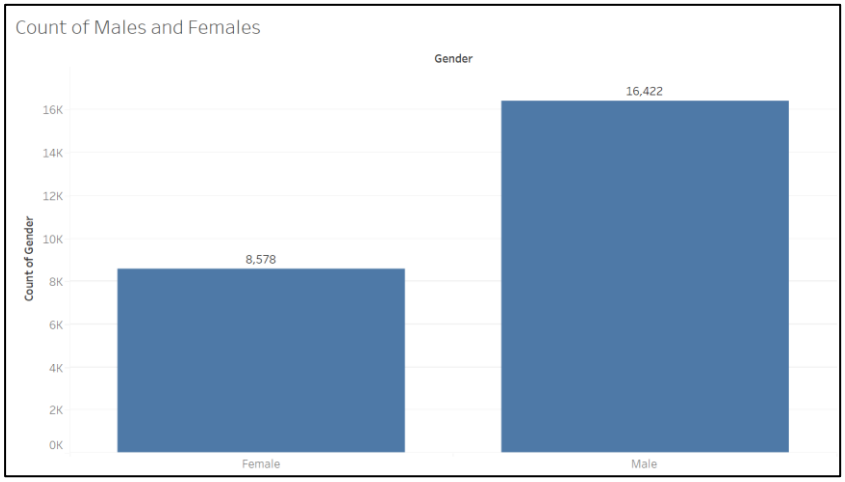


- It can be seen that we have outliers in “daily_avg_steps’, “bmi”, “Year_last_admitted”, regular_checkup_last_year, and “Visited_doctor_last_year” columns.
- We will not be treating the outliers as the data might be beneficial for us also removing them would require consultation from the business or stakeholders.

Some of the EDA steps are performed in Tableau, please refer to the link below to visit the Tableau website

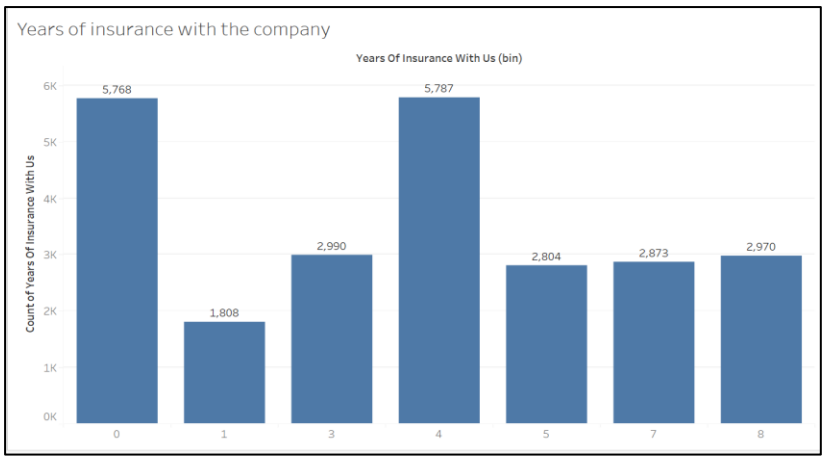
Link:https://public.tableau.com/app/profile/nikita.jamwal/viz/FinalCapstoneEDA_Project_7th-May_2023/Insurancetakenbygenders?publish=yes

Count of Males and Females in the dataset



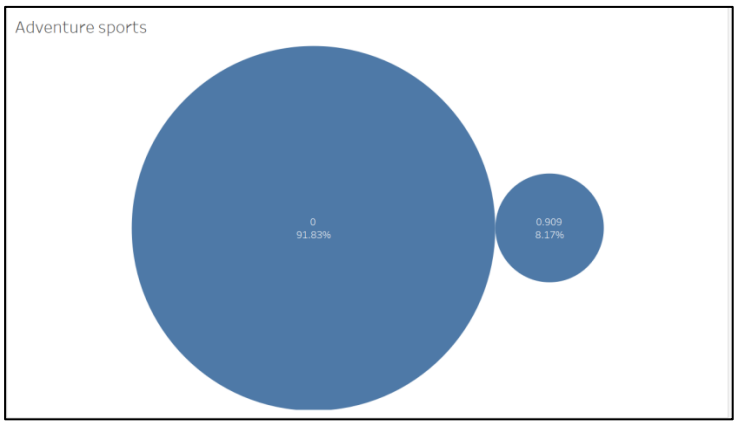
- From the above graph, we can understand that there are 16,422 males and 8,578 females in the dataset.
- Males form the majority.

Years of insurance with the company



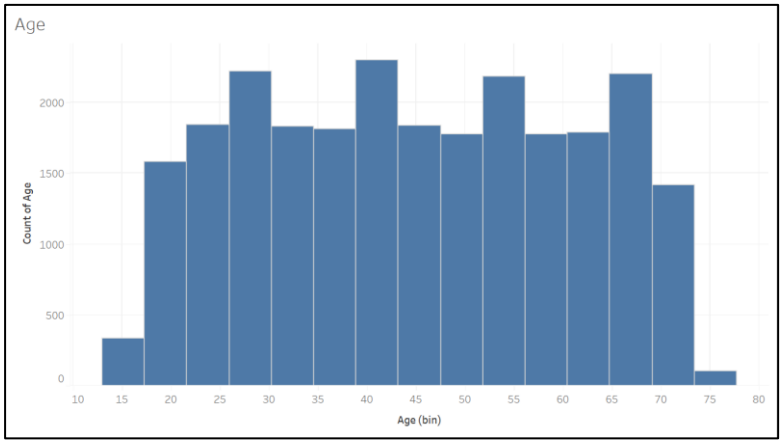
- It is seen that the longer duration of insurance with the company for an individual goes up to 8 years. The people who have been with the company for less than a year form the second major majority after those who have been there for four years.

Adventure Sports



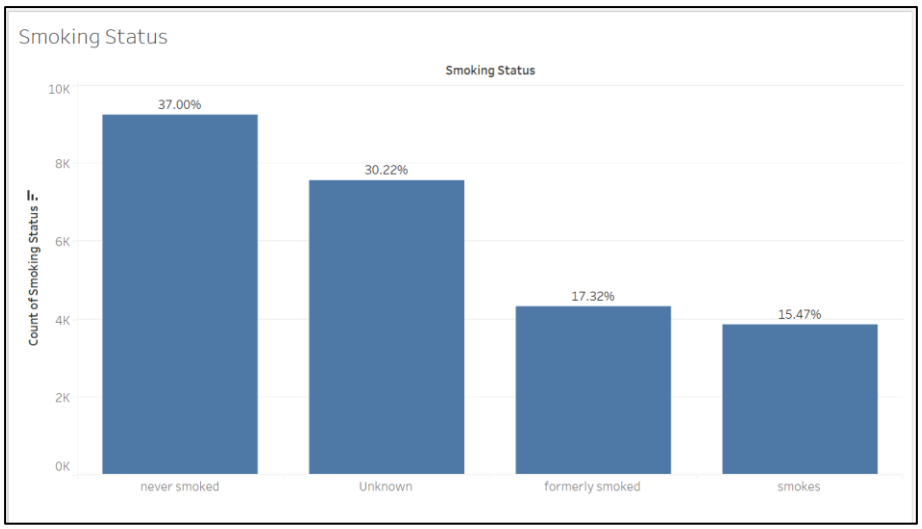
- A significant majority of 91.83% of the population does not participate in any form of adventure sports.

Age



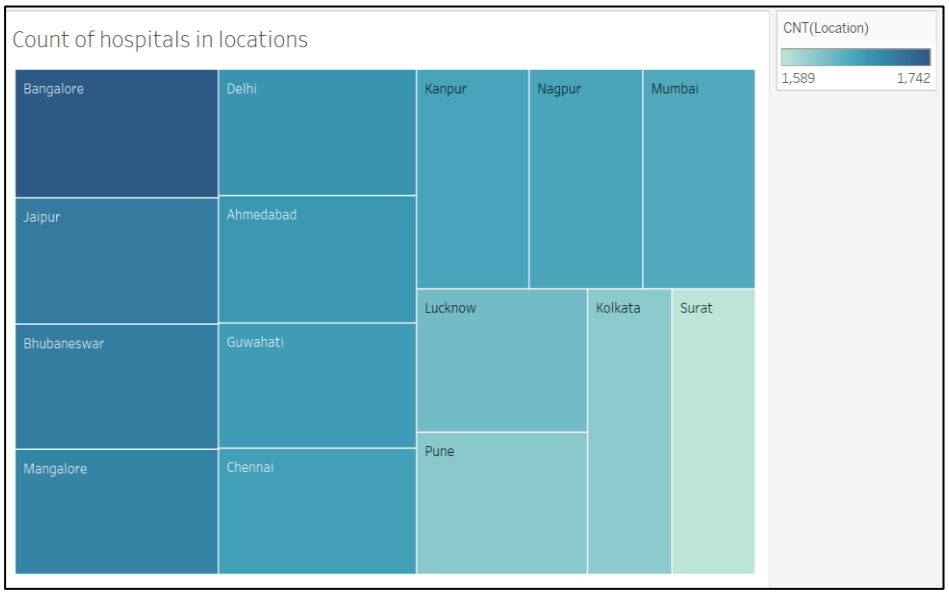
- The data set contains age groups of individuals from 15 to 80 years.

Smoking Status



- The “Smoking status” variable has 30% of the unknown, and 37% of the individuals never smoked. Hence it is advised to get clarity on the data which is not known by the business or stakeholders.

Count of Hospitals in Locations

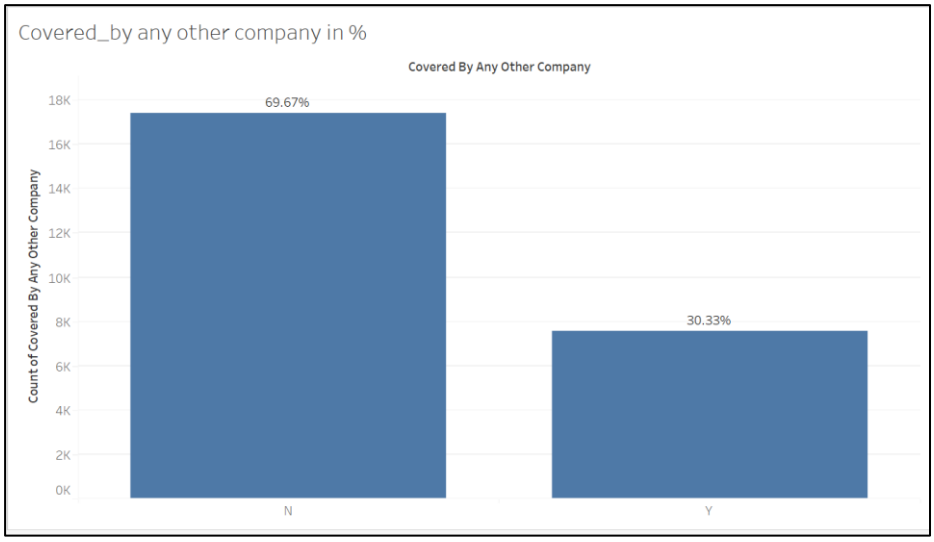


- The above chart shows the count of hospitals present in the location. The darker region indicates a greater number of hospitals and lighter regions indicate a smaller number of hospitals.
- Bangalore, Jaipur, Bhubaneshwar, and Mangalore are the top four with the maximum count.

Insurance cost in the cities

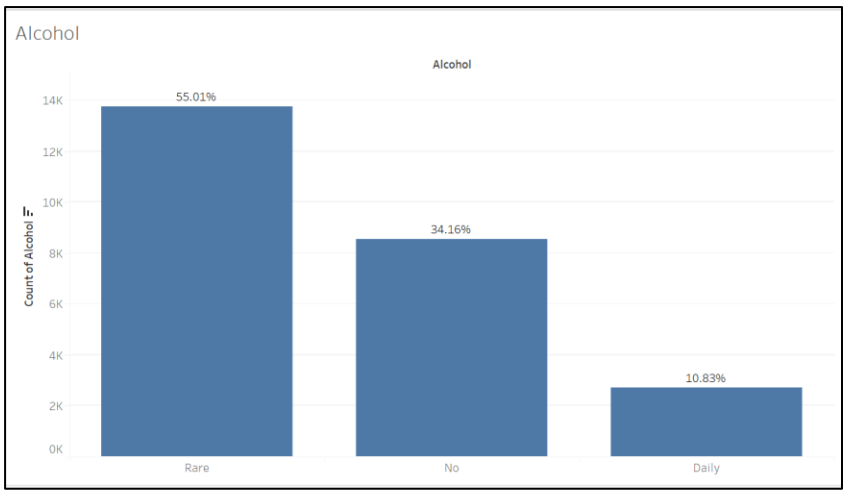
- From the above chart, we can understand that the min range of insurance goes from 2k-3k, the average goes from 26k-27k and the maximum range goes from 66k-67k.

Covered by any other company



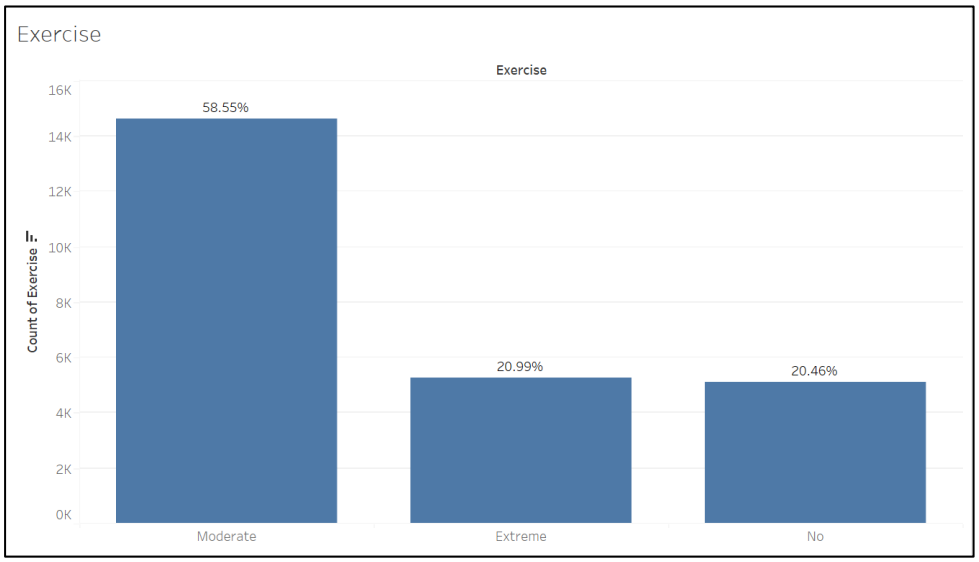
- Almost 70% (69.67%) of the individuals associated with the company do not have insurance policies with any other company.

Alcohol distribution in the data



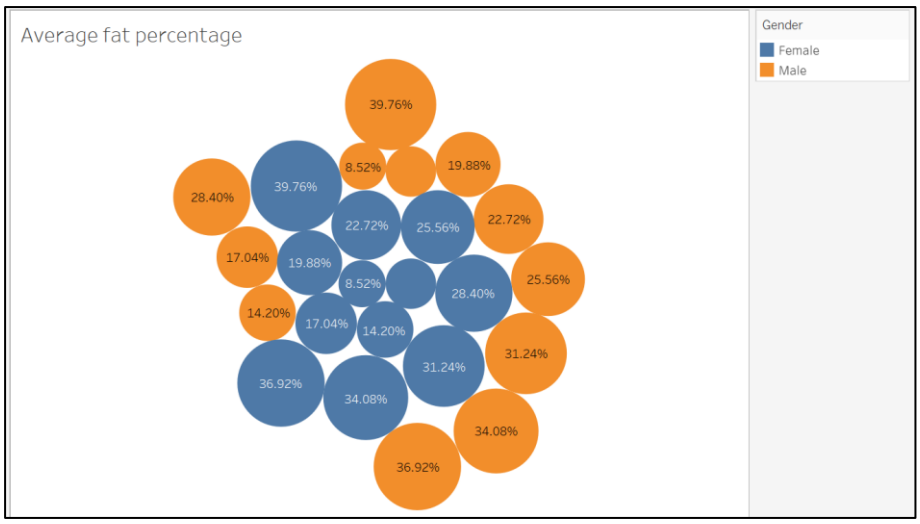
- We have 55% of the individuals who rarely consume alcohol, 34.16% of the individuals who have never consumed, and only 10.83% of the data are daily drinkers.

Exercise



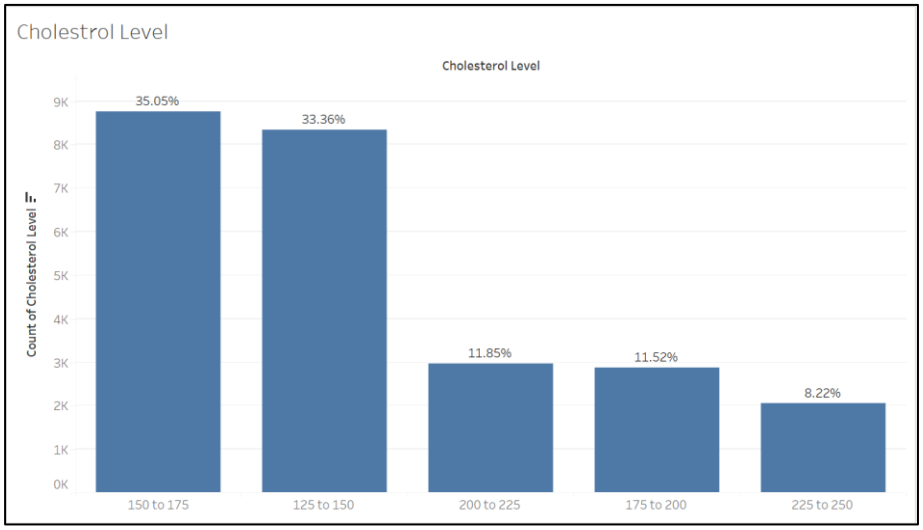
- The data reveals that 58.55% of the population engages in moderate exercise

Average Fat Percentage



- The maximum fat percentage is consistent for both males and females, with a value of 39.76%.
- However, it is important to consider age when evaluating fat percentage, as this is an age-specific parameter.

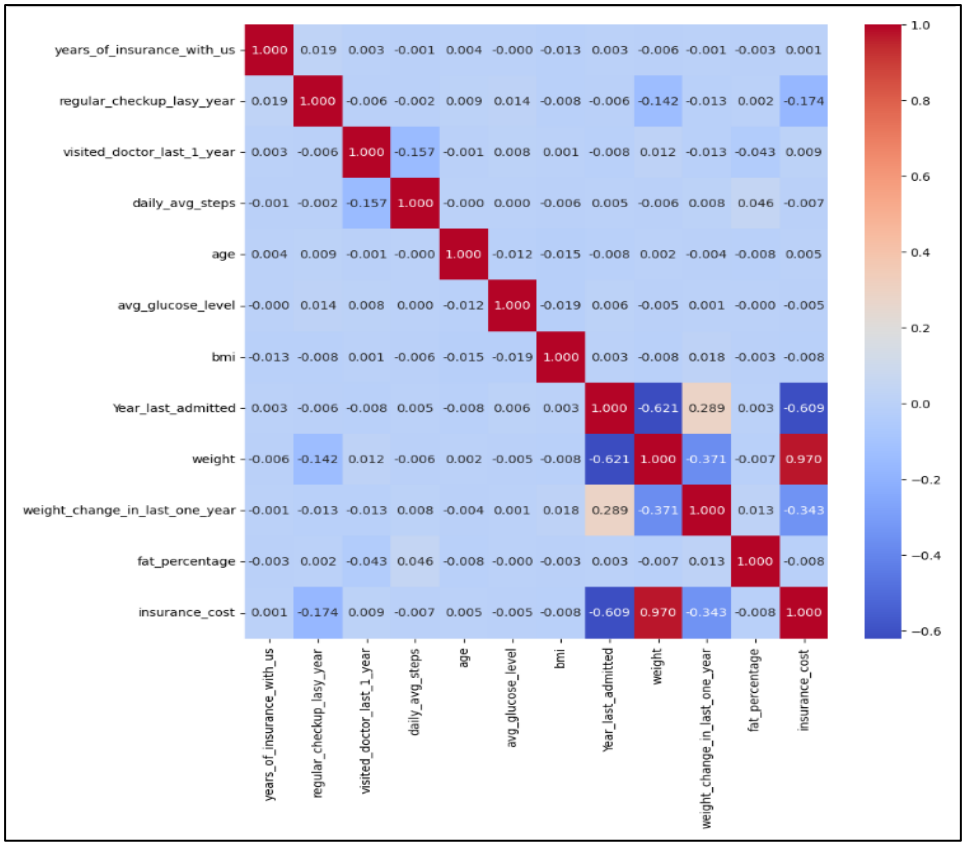
Cholesterol Level



- 79.93% of the population has cholesterol levels within a healthy range. However, a significant proportion of individuals have borderline or high cholesterol levels

Bi-variate Analysis

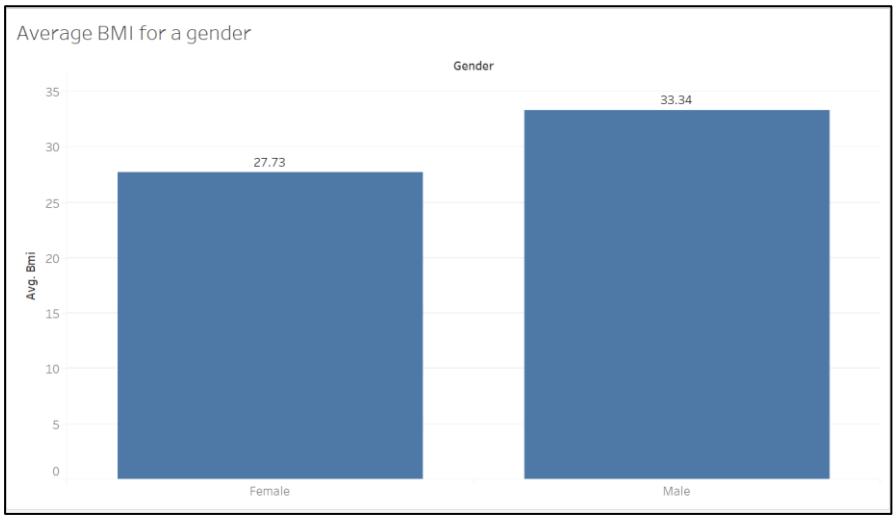
Doing correlation on the continuous data and not on the categorical data



- The correlation plot gives the idea of the presence of collinearity among the variables.
- You can see that the scale ranges from -0.6 (which means extremely weak correlation) till
- 1(extremely high correlation).
- We can see that “Weight” has a high correlation with the target variable, “insurance cost”.
- “Year_last_admitted” and “Weight_change_in_last_one_year” also have a bit of correlation.

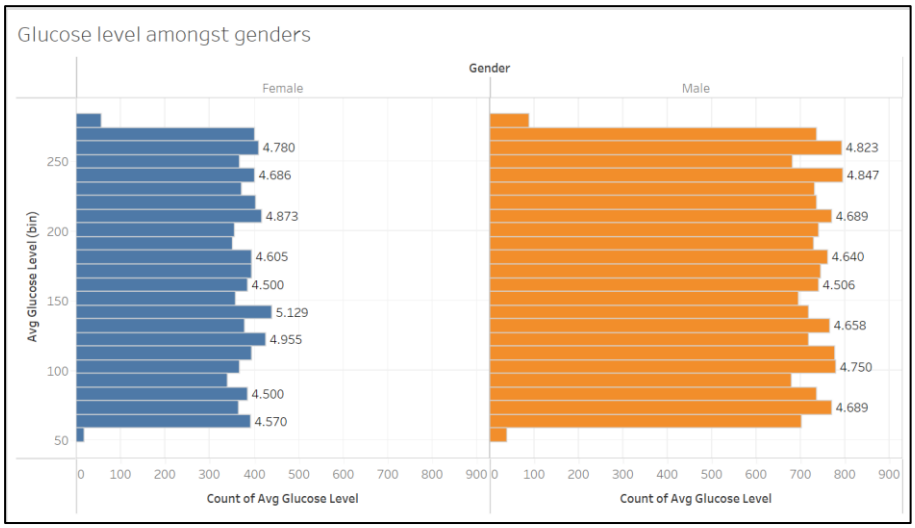
Multi-variate Analysis

Average BMI for a gender



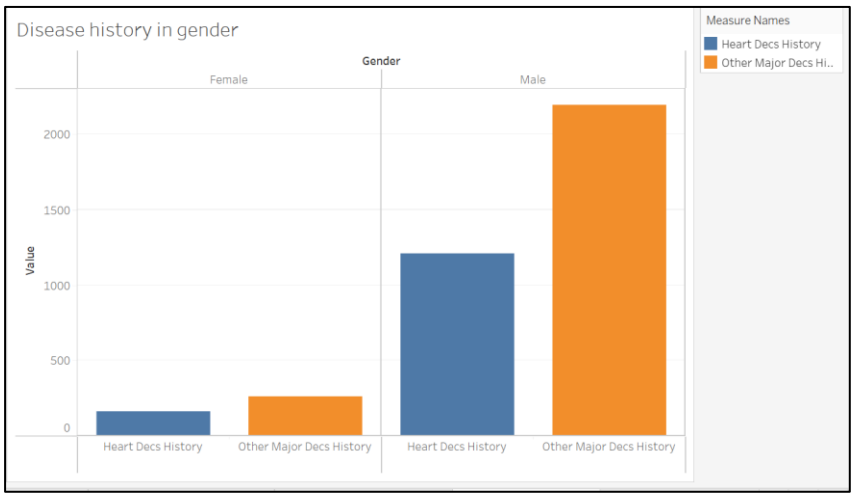
- The average BMI for males is 33.34 and Females is 27.73.

Glucose level amongst genders



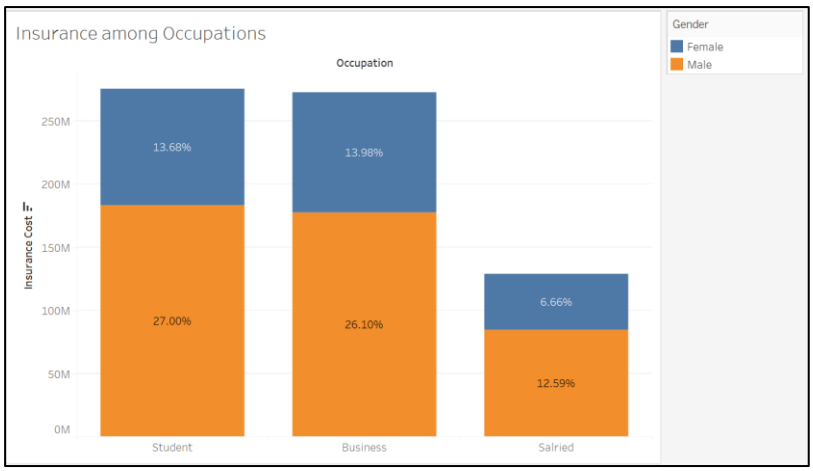
- On average, the glucose levels for the majority of males and females fall within a healthy range. However, there are individuals with lower-than-average glucose levels.

Disease history in gender



- From the above graph, we can see that the count of heart diseases and other major diseases is more in males than in females

Insurance among Occupations



- We can see that overall male contributors are more than female contributors in insurance costs or rather availing insurance services.
- Students are the biggest contributors to the insurance cost, followed by those in the business occupation.

3. Data Cleaning and Pre-processing

a. **Approach used for identifying and treating missing values and outlier treatment (and why).**

```
years_of_insurance_with_us            0
regular_checkup_lasy_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
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
dtype: int64
```

- The dataset contains missing values for some variables. For the “Year_last_admitted” column we have more than 30% of the data missing and hence we will treat it by using the Median filter.
- Median Filter is used particularly when the data is continuous. Here both columns are continuous in nature.
- One reason the median filter is effective for missing value treatment is that it is robust to outliers, which means it is less affected by extreme values in the data compared to other methods such as mean imputation.
- In addition, the median filter preserves the rank order of the data, which can be important in certain applications.
- When using the median filter for missing value treatment, the missing values are replaced with the median value of the neighboring data points. The size

of the neighborhood or window used to calculate the median can be adjusted depending on the nature of the data and the extent of missing values.

Below is the output of the columns with their Median values

```
years_of_insurance_with_us            4.0
regular_checkup_lasy_year             0.0
adventure_sports                      0.0
Occupation                            2.0
visited_doctor_last_1_year            3.0
cholesterol_level                     1.0
daily_avg_steps                    5089.0
age                                  45.0
heart_decs_history                    0.0
other_major_decs_history              0.0
Gender                                0.0
avg_glucose_level                   168.0
bmi                                  30.5
smoking_status                        1.0
Year_last_admitted                 2004.0
weight                               72.0
covered_by_any_other_company          0.0
Alcohol                               2.0
exercise                              2.0
weight_change_in_last_one_year        3.0
fat_percentage                       31.0
insurance_cost                    27148.0
dtype: float64
```

Data after doing the imputation

```
years_of_insurance_with_us         0
regular_checkup_lasy_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
Year_last_admitted                 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
dtype: int64
```

For some columns, we have observed outliers but we won't be imputing them as they might be of any use to us also, we would like to confirm from the stakeholders as to what should be done with them.

b. **Need for variable transformation (if any)**

- Yes, Variable transformation isnecessary as some of the columns in our dataset are categorical.
- Regression analysis is done on continuous data only so to run our models effectively we have done variable transformation also known as one-hot encoding.
- One-hot encoding in machine learning is the conversion of categorical information into a format that may be fed into machine learning

algorithms to improve prediction accuracy. One-hot encoding is a common method for dealing with categorical data in machine learning.

	years_of_insurance_with_us	regular_checkup_lasy_year	adventure_sports	Occupation	visited_doctor_last_1_year	cholesterol_level	daily_avg_steps	age	he
0	3	1	1	2	2	0	4866	28	
1	0	0	0	1	4	1	6411	50	
2	1	0	0	3	4	3	4509	68	
3	7	4	0	3	2	2	6214	51	
4	3	1	0	1	2	1	4938	44	

5 rows × 22 columns

- As we can see we have converted our variable's data into numbers.

c. Variables removed or added and why (if any)
   - Yes, we have removed two variables from the dataset which are applicant id and location. Since the applicant id is just a number used to refer the applicant in the database, we have removed it as it will not add any value to the analysis.
   - Also, the location column doesn't add any help to the models but could be studied thoroughly in the EDA part as we have done above.

4. Model building

- Model building refers to the process of creating a statistical or machine learning model that can be used to make predictions or draw insights from data. This involves several steps, including:
- Data preparation: This involves collecting, cleaning, and pre-processing the data to ensure that it is in a format suitable for model building.
- Feature selection: This step involves selecting the most relevant features or variables from the data to be used in the model.
- Model selection: This step involves choosing the appropriate type of model to use, such as a regression model, decision tree, neural network, or support vector machine.
- Training the model: This step involves using a subset of the data, called the training set, to train the model by adjusting its parameters and optimizing its performance.
- Model evaluation: This step involves testing the performance of the model on a separate subset of the data, called the validation set or test set, to determine how well it generalizes to new data.
- Model tuning: This step involves adjusting the model parameters and hyperparameters to improve its performance.
- Deployment: This step involves using the model to make predictions on new data or integrate it into a larger system.

First, we will build the models without scaling or VIF.

Capture the target column into separate vectors for the training set and test set

X data

	years_of_insurance_with_us	regular_checkup_lasy_year	adventure_sports	Occupation	visited_doctor_last_1_year	cholesterol_level	daily_avg_steps	age	he
0	3	1	1	2	2	0	4866	28	
1	0	0	0	1	4	1	6411	50	
2	1	0	0	3	4	3	4509	68	
3	7	4	0	3	2	2	6214	51	
4	3	1	0	1	2	1	4938	44	

5 rows × 21 columns

Y data

```
0    20978
1     6170
2    28382
3    27148
4    29616
Name: insurance_cost, dtype: int64
```

Then we split the data into train and test sets in the ratio of 70:30.

The models which we have used are Random Forest, Lasso regression, Ridge Regression, Elastic Net, and Linear Regression.

Random Forest

The basic idea behind Random Forest is to randomly select a subset of the features and a subset of the data to create each decision tree. This helps to reduce overfitting and improve generalization performance. The decision trees are trained using a process called bootstrapped aggregation, or bagging, which involves randomly sampling the data with replacement to create multiple subsets for training.

Random Forest has several advantages over other machine learning algorithms. It is robust to noise and outliers, can handle high-dimensional data with many features, and provides measures of feature importance that can be used for feature selection. It is also relatively easy to use and requires minimal parameter tuning.

Linear Regression

Linear regression is a type of statistical modeling technique used to establish a relationship between a dependent variable and one or more independent variables. The goal of linear regression is to predict the value of the dependent variable based on the values of the independent variables.

In its simplest form, linear regression assumes that there is a linear relationship between the dependent variable and the independent variables. This means that the relationship can be represented by a straight line, which can be described by an equation in the form:

y = b0 + b1*x1* + *b2*x2 + ... + bn*xn

where y is the dependent variable, x1, x2, ..., xn are the independent variables, b0 is the intercept or constant term, and b1, b2, ..., bn are the regression coefficients that represent the change in y for a unit change in each independent variable.

Linear regression has several advantages, including its simplicity and interpretability, as well as its ability to handle continuous and categorical independent variables. However, it also has limitations, such as the assumption of linearity and the sensitivity to outliers and multicollinearity.

OLS Regression Results

Dep. Variable:	insurance_cost	R-squared:	0.945
Model:	OLS	Adj. R-squared:	0.945
Method:	Least Squares	F-statistic:	6156.
Date:	Fri, 05 May 2023	Prob (F-statistic):	0.00
Time:	20:28:07	Log-Likelihood:	-71460.
No. Observations:	7500	AIC:	1.430e+05
Df Residuals:	7478	BIC:	1.431e+05
Df Model:	21		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	1.52e+04	1.82e+04	0.833	0.405	-2.06e+04	5.1e+04
years_of_insurance_with_us	-7.5249	15.307	-0.492	0.623	-37.532	22.482
regular_checkup_lasy_year	-411.7246	32.895	-12.516	0.000	-476.208	-347.242
adventure_sports	286.5333	140.095	2.045	0.041	11.907	561.160
Occupation	-7.6252	49.993	-0.153	0.879	-105.625	90.375
visited_doctor_last_1_year	-55.2807	33.695	-1.641	0.101	-121.333	10.772
cholesterol_level	21.3949	35.273	0.607	0.544	-47.751	90.541
daily_avg_steps	-0.0056	0.037	-0.151	0.880	-0.078	0.067
age	4.7554	2.390	1.989	0.047	0.070	9.441
heart_decs_history	490.0196	167.348	2.928	0.003	161.970	818.069
other_major_decs_history	59.2786	129.907	0.456	0.648	-195.376	313.934
Gender	82.6452	87.900	0.940	0.347	-89.664	254.955
avg_glucose_level	-0.6839	0.614	-1.114	0.265	-1.887	0.519
bmi	-6.5464	5.340	-1.226	0.220	-17.014	3.922
smoking_status	29.3758	31.715	0.926	0.354	-32.794	91.546
Year_last_admitted	-46.4977	8.989	-5.173	0.000	-64.118	-28.877
weight	1468.9024	5.556	264.370	0.000	1458.011	1479.794
covered_by_any_other_company	1211.3070	87.483	13.846	0.000	1039.815	1382.799
Alcohol	-23.6759	42.461	-0.558	0.577	-106.912	59.560
exercise	-10.8490	48.041	-0.226	0.821	-105.024	83.326
weight_change_in_last_one_year	122.9557	24.681	4.982	0.000	74.575	171.337
fat_percentage	-7.4442	4.477	-1.663	0.096	-16.221	1.332

Omnibus:	107.946	Durbin-Watson:	1.992

Interpretation of R^2

The R-squared value tells us that our model can explain 94.5% of the variance in the training set.

Interpretation of coefficients

The coefficients tell us how one unit change in X can affect y. The sign of the coefficient indicates if the relationship is positive or negative. For example, in the dataset, an increase in 1 kg of weight will result in an increase in the insurance cost

Interpretation of p-values (P > |t|)

For each predictor variable, there is a null hypothesis and an alternate hypothesis.

Null hypothesis: Predictor variable is not significant Alternate hypothesis: Predictor variable is significant (P > |t|) and gives the p-value for each predictor variable to check the null hypothesis. If the level of significance is set to 5% (0.05), the p-values greater than 0.05 would indicate that the corresponding predictor variables are not significant. However, due to the presence of multicollinearity in our data, the p-values will also change.

We need to ensure that there is no multicollinearity in order to interpret the p-values.

Lasso Regression

Lasso (Least Absolute Shrinkage and Selection Operator) is a machine learning algorithm used for feature selection and regularization in linear regression. It is a type of linear regression that adds a penalty term to the cost function in order to encourage simpler models with fewer features.

The penalty term in the cost function is based on the L1 norm of the regression coefficients, which encourages some of the coefficients to be exactly zero, effectively removing the corresponding features from the model. This makes Lasso a useful tool for selecting a subset of the most relevant features from a large set of predictors.

The degree of regularization in Lasso is controlled by a hyperparameter called lambda, which determines the trade-off between the goodness of fit and the simplicity of the model. Larger values of lambda result in more regularization and a sparser model with fewer features.

One of the advantages of Lasso is that it can handle highly correlated features by selecting one representative feature from each group of correlated predictors. It can also be used to perform feature engineering by creating new features that are a combination of the existing features.

```
lasso equation: y = 15410.350402471813 + 0.00*const + -12.49*years_of_insurance_with_us + -473.00*regular_checkup_lasy_year + 1
15.71*adventure_sports+ -38.10*Occupation+ -36.15*visited_doctor_last_1_year+ 40.92*cholesterol_level+...
```

From the above equation, we can see the important variable.

Ridge Regression

Ridge regression is a machine learning algorithm used for regularization in linear regression. It is a type of linear regression that adds a penalty term to the cost function in order to reduce the impact of multicollinearity and overfitting.

The penalty term in the cost function is based on the L2 norm of the regression coefficients, which encourages the coefficients to be small and reduces the impact of large coefficients on the overall model. This helps to prevent overfitting and improves the generalization performance of the model.

The degree of regularization in Ridge regression is controlled by a hyperparameter called alpha, which determines the trade-off between the goodness of fit and the simplicity of the model. Larger values of alpha result in more regularization and a simpler model with smaller coefficients.

One of the advantages of Ridge regression is that it can handle multicollinearity by shrinking the coefficients of correlated predictors towards each other. This can help to improve the stability and interpretability of the model.

```
Root Mean Squared Error: 3331.9407472683565
const : 0.0
years_of_insurance_with_us : -12.491652640541847
regular_checkup_lasy_year : -472.9965039965449
adventure_sports : 115.71533651939559
Occupation : -38.098978328860426
visited_doctor_last_1_year : -36.15169874543777
cholesterol_level : 40.92113924143083
daily_avg_steps : -0.027477160868464988
age : 2.723997667228684
heart_decs_history : 90.93719935619774
other_major_decs_history : 59.97812087318924
Gender : -31.95451102197291
avg_glucose_level : 0.3642326465611445
bmi : -1.3365847410827827
smoking_status : -9.428212755734357
Year_last_admitted : -46.91846466828211
weight : 1473.2685598945486
covered_by_any_other_company : 1212.809238562498
Alcohol : 15.430462239629529
exercise : 8.583170170741871
weight_change_in_last_one_year : 185.65672740692003
fat_percentage : -1.215137779118032
```

Elastic Net

Elastic Net is a machine learning algorithm used for regularization in linear regression. It is a combination of Lasso and Ridge regression that combines their advantages and overcomes their limitations.

The degree of regularization in Elastic Net is controlled by two hyperparameters: alpha, which determines the trade-off between the L1 and L2 penalties, and lambda, which determines the overall strength of the regularization.

Elastic Net is particularly useful when the data has many correlated predictors, as it can select a group of predictors that are jointly associated with the response variable. It is also more stable than Lasso when the number of predictors is greater than the number of samples, and more flexible than Ridge when the predictors have different levels of importance.

Now we will be checking the models after doing scaling and VIF

For scaling we have used Z-score. The Z-score, also known as the standard score, is a statistical measure that indicates how many standard deviations a given data point is from the mean of a distribution. It is a normalized score that enables the comparison of data points from different distributions on a common scale.

	years_of_insurance_with_us	regular_checkup_lasy_year	adventure_sports	Occupation	visited_doctor_last_1_year	cholesterol_level	daily_avg_steps	ag
0	-0.417807	0.188690	3.352150	0.006632	-0.967205	-1.002742	-0.332228	-1.05036
1	-1.568750	-0.645043	-0.298316	-1.106181	0.784661	-0.210186	1.134787	0.31549
2	-1.185102	-0.645043	-0.298316	1.119445	0.784661	1.374926	-0.671209	1.43300
3	1.116783	2.689890	-0.298316	1.119445	-0.967205	0.582370	0.947731	0.37757
4	-0.417807	0.188690	-0.298316	-1.106181	-0.967205	-0.210186	-0.263863	-0.05701

5 rows × 22 columns

Z-scores are commonly used in statistics for various purposes, such as outlier detection, hypothesis testing, and data normalization.

Linear Regression

Here we have done VIF for the variables so as to include only the important variables in the analysis.

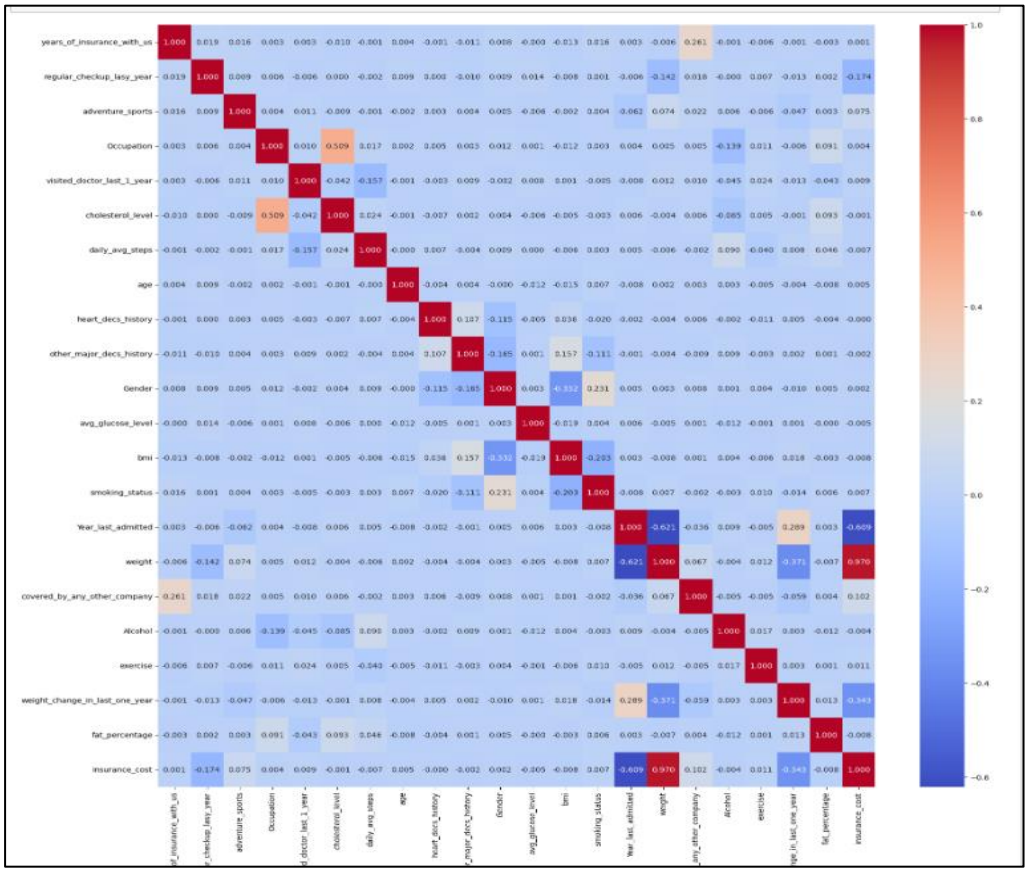
```
VIF for const : 3.7203162056522863
VIF for years_of_insurance_with_us : 1.490637466976036
VIF for regular_checkup_lasy_year : 1.0970448964361632
VIF for adventure_sports : 8.13581370495046
VIF for Occupation : 8.658225522203262
VIF for visited_doctor_last_1_year : 2.7257880009838416
VIF for cholesterol_level : 26.46902273843567
VIF for daily_avg_steps : 8.784371099737237
VIF for age : 1.0817725322509146
VIF for heart_decs_history : 1.168532072980666
VIF for other_major_decs_history : 1.8055419429143538
VIF for Gender : 8.141663321800342
VIF for avg_glucose_level : 20.276322019823667
VIF for bmi : 2.432779304986014
VIF for smoking_status : 725.6835155363706
VIF for Year_last_admitted : 1072.267146268522
VIF for weight : 1.5912108885530991
VIF for covered_by_any_other_company : 2.805488465670437
VIF for Alcohol : 3.9700206170123638
VIF for exercise : 3.7871405563614498
VIF for weight_change_in_last_one_year : 12.322517800319645
VIF for fat_percentage : 83.27181398185708
```

Variance Inflation Factor (VIF) is a measure of collinearity in regression analysis. It quantifies how much the variance of the estimated regression coefficient for a particular independent variable increases due to multicollinearity with other independent variables in the model.

The VIF ranges from 1 (no collinearity) to infinity (perfect collinearity), with values above 5 or 10 indicating high collinearity. A high VIF indicates that the variance of the estimated regression coefficient for the corresponding independent variable is inflated due to the presence of correlated independent variables in the model.

The threshold chosen is 5, if VIF > 5 the variable will be removed else not.

Below is the correlation matrix of the data



Let's remove/drop multicollinear columns one by one and observe the effect on our predictive mode

Variable Name	Value
Adventure_sport	R-squared: 0.945 Adjusted R-squared: 0.945
Occupation	R-squared: 0.945 Adjusted R-squared: 0.945
Cholesterol level	R-squared: 0.945 Adjusted R-squared: 0.945
Daily_avg_steps	R-squared: 0.945 Adjusted R-squared: 0.945
Gender	R-squared: 0.945 Adjusted R-squared: 0.945

As we can see after dropping all the columns mentioned above, there is no impact on the r-squared value of the model. Hence, we will proceed with dropping the columns.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:         insurance_cost   R-squared:                       0.945
Model:                            OLS   Adj. R-squared:                  0.945
Method:                 Least Squares   F-statistic:                 1.992e+04
Date:                Fri, 05 May 2023   Prob (F-statistic):               0.00
Time:                        20:34:12   Log-Likelihood:                 445.30
No. Observations:               17500   AIC:                            -858.6
Df Residuals:                   17484   BIC:                            -734.3
Df Model:                          15
Covariance Type:            nonrobust
=================================================================================================
                                    coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------------------
const                             0.0010      0.002      0.562      0.574      -0.002       0.005
years_of_insurance_with_us       -0.0023      0.002     -1.231      0.218      -0.006       0.001
regular_checkup_lasy_year        -0.0395      0.002    -21.758      0.000      -0.043      -0.036
visited_doctor_last_1_year       -0.0027      0.002     -1.517      0.129      -0.006       0.001
heart_decs_history                0.0015      0.002      0.804      0.421      -0.002       0.005
other_major_decs_history          0.0014      0.002      0.758      0.449      -0.002       0.005
avg_glucose_level                 0.0015      0.002      0.853      0.394      -0.002       0.005
bmi                              -0.0004      0.002     -0.232      0.817      -0.004       0.003
smoking_status                   -0.0010      0.002     -0.566      0.571      -0.005       0.003
Year_last_admitted               -0.0181      0.002     -7.837      0.000      -0.023      -0.014
weight                            0.9592      0.002    398.521      0.000       0.954       0.964
covered_by_any_other_company      0.0390      0.002     21.051      0.000       0.035       0.043
Alcohol                           0.0008      0.002      0.472      0.637      -0.003       0.004
exercise                          0.0005      0.002      0.271      0.787      -0.003       0.004
weight_change_in_last_one_year    0.0218      0.002     11.300      0.000       0.018       0.026
fat_percentage                   -0.0007      0.002     -0.401      0.689      -0.004       0.003
==============================================================================
Omnibus:                      639.866   Durbin-Watson:                   1.978
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              793.264
Skew:                           0.418   Prob(JB):                    5.56e-173
Kurtosis:                       3.624   Cond. No.                         2.34
==============================================================================
```

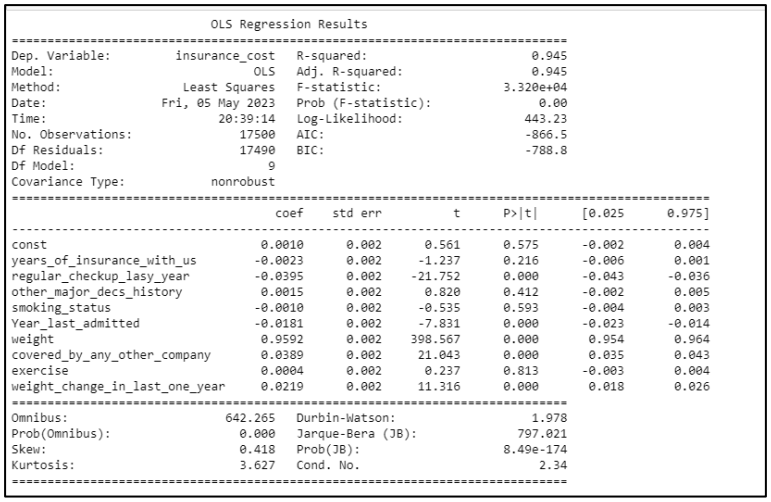
Now this is an iterative process, we will do it till we have the required set of variables.

```
VIF for const : 3.7200706706326474
VIF for years_of_insurance_with_us : 1.453974373104186
VIF for regular_checkup_lasy_year : 1.0968500744153182
VIF for visited_doctor_last_1_year : 8.135469658831871
VIF for heart_decs_history : 8.656487092430767
VIF for other_major_decs_history : 2.7254254194827077
VIF for avg_glucose_level : 26.4677871087109
VIF for bmi : 8.782110029566887
VIF for smoking_status : 1.081539762342331
VIF for Year_last_admitted : 1.1685017706949798
VIF for weight : 1.8055419066251361
VIF for covered_by_any_other_company : 8.141658142572833
VIF for Alcohol : 20.275492333643935
VIF for exercise : 2.4327767150698176
VIF for weight_change_in_last_one_year : 179.58967673433935
VIF for fat_percentage : 69.88371134923634
```

Variable Name	Value
Visited_doctor_last_1_year	R-squared: 0.945 Adjusted R-squared: 0.945
Heart_decs_history	R-squared: 0.945 Adjusted R-squared: 0.945
avg_glucose_level	R-squared: 0.945 Adjusted R-squared: 0.945
bmi	R-squared: 0.945 Adjusted R-squared: 0.945
covered_by_any_other_company	R-squared: 0.943 Adjusted R-squared: 0.943
Alcohol	R-squared: 0.945 Adjusted R-squared: 0.945
weight_change_in_last_one_year	R-squared: 0.944 Adjusted R-squared: 0.944

fat percentage	R-squared: 0.945 Adjusted R-squared: 0.945

As we can see after dropping “covered_by_any_other_company” and “weight_change_in_last_one_year” there is a slight decrease in the -squared value of the columns. Hence, we will proceed with retaining those columns and dropping the others.



```
                            OLS Regression Results
==============================================================================
Dep. Variable:         insurance_cost   R-squared:                       0.945
Model:                            OLS   Adj. R-squared:                  0.945
Method:                 Least Squares   F-statistic:                 3.320e+04
Date:                Fri, 05 May 2023   Prob (F-statistic):               0.00
Time:                        20:39:14   Log-Likelihood:                 443.23
No. Observations:               17500   AIC:                            -866.5
Df Residuals:                   17490   BIC:                            -788.8
Df Model:                           9
Covariance Type:            nonrobust
=================================================================================================
                                    coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------------------
const                             0.0010      0.002      0.561      0.575      -0.002       0.004
years_of_insurance_with_us       -0.0023      0.002     -1.237      0.216      -0.006       0.001
regular_checkup_lasy_year        -0.0395      0.002    -21.752      0.000      -0.043      -0.036
other_major_decs_history          0.0015      0.002      0.820      0.412      -0.002       0.005
smoking_status                   -0.0010      0.002     -0.535      0.593      -0.004       0.003
Year_last_admitted               -0.0181      0.002     -7.831      0.000      -0.023      -0.014
weight                            0.9592      0.002    398.567      0.000       0.954       0.964
covered_by_any_other_company      0.0389      0.002     21.043      0.000       0.035       0.043
exercise                          0.0004      0.002      0.237      0.813      -0.003       0.004
weight_change_in_last_one_year    0.0219      0.002     11.316      0.000       0.018       0.026
==============================================================================
Omnibus:                      642.265   Durbin-Watson:                   1.978
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              797.021
Skew:                           0.418   Prob(JB):                    8.49e-174
Kurtosis:                       3.627   Cond. No.                         2.34
==============================================================================
```

```
VIF for const : 3.7200706706326474
VIF for years_of_insurance_with_us : 1.453974373104186
VIF for regular_checkup_lasy_year : 1.0968500744153182
VIF for other_major_decs_history : 8.135469658831871
VIF for smoking_status : 8.656487092430767
VIF for Year_last_admitted : 2.7254254194827077
VIF for weight : 26.4677871087109
VIF for covered_by_any_other_company : 8.782110029566887
VIF for exercise : 1.081539762342331
VIF for weight_change_in_last_one_year : 1.1685017706949798
```

Variable Name	Value
other_major_decs_history	R-squared: 0.945 Adjusted R-squared: 0.945
smoking_status	R-squared: 0.945 Adjusted R-squared: 0.945
weight	R-squared: 0.442 Adjusted R-squared: 0.442
covered_by_any_other_company	R-squared: 0.943 Adjusted R-squared: 0.943

As we can see after dropping “Weight” there is a drastic decrease in the r-squared value of the columns. Hence, we will proceed with retaining that column and dropping the others.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:         insurance_cost   R-squared:                       0.945
Model:                            OLS   Adj. R-squared:                  0.945
Method:                 Least Squares   F-statistic:                 4.268e+04
Date:                Fri, 05 May 2023   Prob (F-statistic):               0.00
Time:                        20:44:57   Log-Likelihood:                442.69
No. Observations:               17500   AIC:                            -869.4
Df Residuals:                   17492   BIC:                            -807.2
Df Model:                           7
Covariance Type:            nonrobust
=================================================================================================
                                    coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------------------
const                             0.0010      0.002      0.554      0.579      -0.003       0.004
years_of_insurance_with_us       -0.0023      0.002     -1.255      0.210      -0.006       0.001
regular_checkup_lasy_year        -0.0395      0.002    -21.751      0.000      -0.043      -0.036
Year_last_admitted               -0.0181      0.002     -7.835      0.000      -0.023      -0.014
weight                            0.9592      0.002    398.578      0.000       0.954       0.964
covered_by_any_other_company      0.0389      0.002     21.047      0.000       0.035       0.043
exercise                          0.0004      0.002      0.237      0.813      -0.003       0.004
weight_change_in_last_one_year    0.0219      0.002     11.330      0.000       0.018       0.026
==============================================================================
Omnibus:                      643.036   Durbin-Watson:                   1.978
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              798.317
Skew:                           0.419   Prob(JB):                     4.44e-174
Kurtosis:                       3.628   Cond. No.                         2.34
==============================================================================
```

Assumptions of Linear Regression

These assumptions are essential conditions that should be met before we draw inferences regarding the model estimates or use the model to make a prediction.

For Linear Regression, we need to check if the following assumptions hold: -

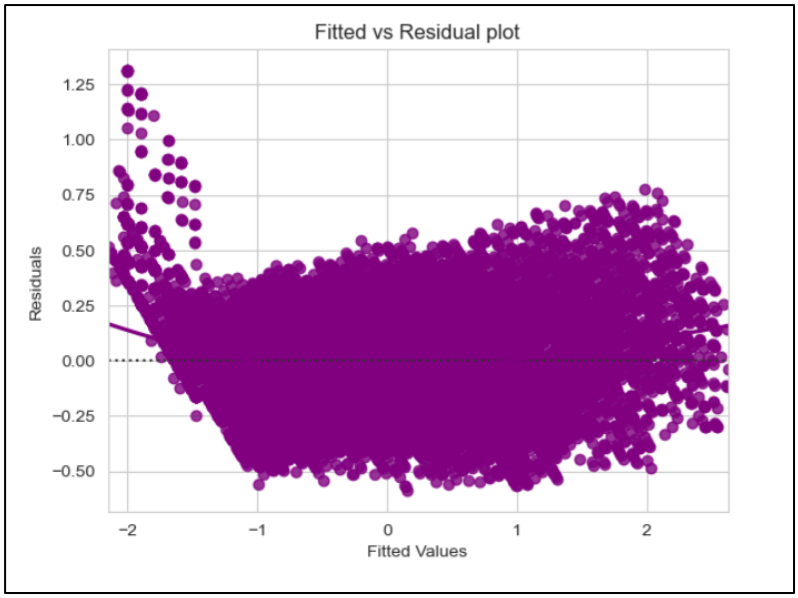
Linearity Independence Homoscedasticity Normality of error terms No strong Multicollinearity

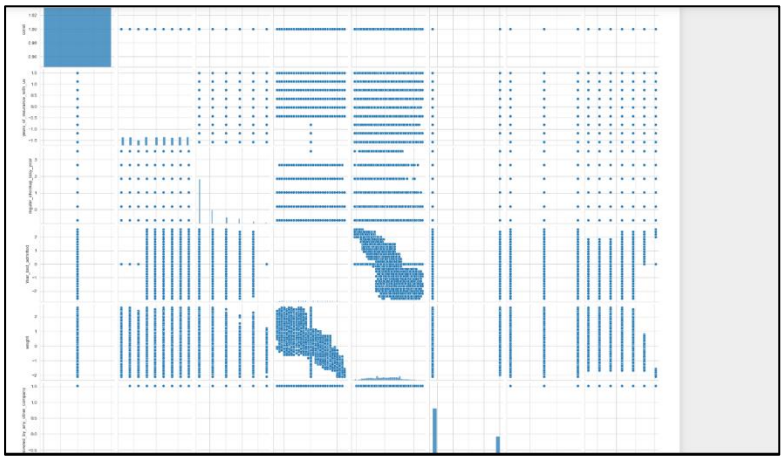
	Actual Values	Fitted Values	Residuals
0	0.086194	0.164791	-0.078597
1	-0.258417	-0.449945	0.191528
2	1.378485	1.226125	0.152359
3	0.947721	1.276295	-0.328574
4	0.344652	0.639438	-0.294785

TEST FOR LINEARITY AND INDEPENDENCE

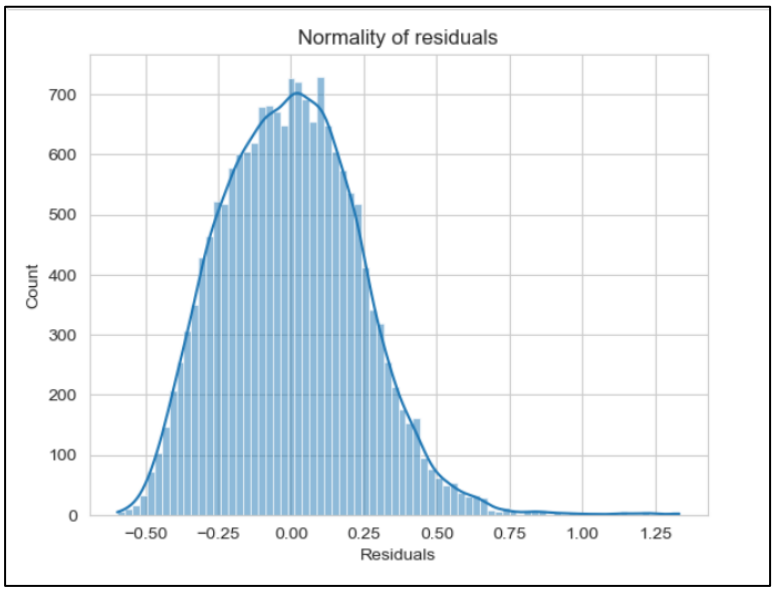
Linearity describes a straight-line relationship between two variables, predictor variables must have a linear relation with the dependent variable.

The plot of fitted values vs residuals checks the linearity of the model. If they don't follow any pattern (the curve is a straight line), then we say the model is linear otherwise model is showing signs of non-linearity.



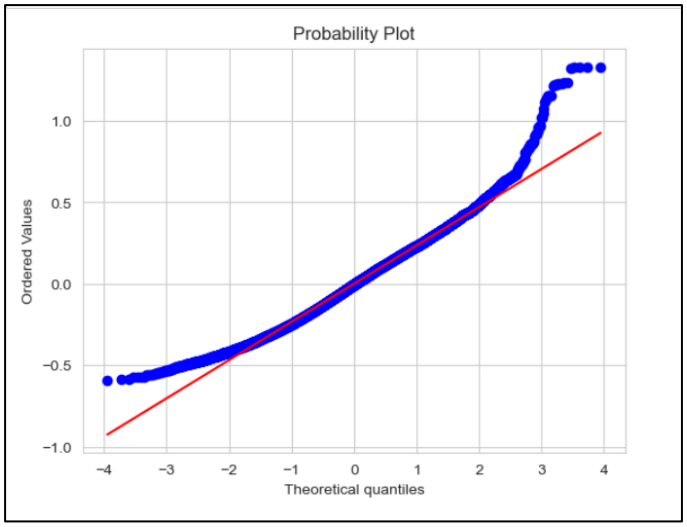


We can see that the year_last_admitted and weight have slightly non-linear relationship.



The residual terms are normally distributed.

The QQ plot of residuals can be used to visually check the normality assumption. The normal probability plot of residuals should approximately follow a straight line.



Most of the points are lying on a straight line in the QQ plot The Shapiro-Wilk test can also be used for checking the normality.

The null and alternate hypotheses of the test are as follows:

Null hypothesis - Data is normally distributed.

Alternate hypothesis - Data is not normally distributed.

```
ShapiroResult(statistic=0.9869423508644104, pvalue=7.818082241103213e-37)
```

Since p-value > 0.05, the residuals are normal as per Shapiro test.

TEST FOR HOMOSCEDASTICITY

Homoscedacity - If the variance of the residuals is symmetrically distributed across the regression line, then the data is said to be homoscedastic.

Heteroscedasticity- If the variance is unequal for the residuals across the regression line, then the data is said to be heteroscedastic. In this case, the residuals can form an arrow shape or any other non-symmetrical shape.

Null hypothesis: Residuals are homoscedastic Alternate hypothesis: Residuals have heteroscedasticity

```
[('F statistic', 1.000980045519812), ('p-value', 0.48173772827388733)]
```

Since p-value > 0.05 we can say that the residuals are homoscedastic.

All the assumptions of linear regression are now satisfied. Let's check the summary of our final model

```
                            OLS Regression Results
==============================================================================
Dep. Variable:         insurance_cost   R-squared:                       0.945
Model:                            OLS   Adj. R-squared:                  0.945
Method:                 Least Squares   F-statistic:                 4.268e+04
Date:                Fri, 05 May 2023   Prob (F-statistic):               0.00
Time:                        21:21:57   Log-Likelihood:                442.69
No. Observations:               17500   AIC:                            -869.4
Df Residuals:                   17492   BIC:                            -807.2
Df Model:                           7
Covariance Type:            nonrobust
=====================================================================================================
                                    coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------------------------
const                                0.0010      0.002      0.554      0.579      -0.003       0.004
years_of_insurance_with_us          -0.0023      0.002     -1.255      0.210      -0.006       0.001
regular_checkup_lasy_year           -0.0395      0.002    -21.751      0.000      -0.043      -0.036
Year_last_admitted                  -0.0181      0.002     -7.835      0.000      -0.023      -0.014
weight                               0.9592      0.002    398.578      0.000       0.954       0.964
covered_by_any_other_company         0.0389      0.002     21.047      0.000       0.035       0.043
exercise                             0.0004      0.002      0.237      0.813      -0.003       0.004
weight_change_in_last_one_year       0.0219      0.002     11.330      0.000       0.018       0.026
==============================================================================
Omnibus:                      643.036   Durbin-Watson:                   1.978
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              798.317
Skew:                           0.419   Prob(JB):                    4.44e-174
Kurtosis:                       3.628   Cond. No.                         2.34
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

The above values against the coefficients indicate that, if there is an increase in years or insurance, regular checkup and year last admitted that will lead to a decrease in some amount of the cost. While the other parameters will increase the cost of the insurance.

Linear Regression final equation

```
insurance_cost = 0.000989097789103339 + -0.0023195632110428574 * ( years_of_insurance_with_us ) +  -0.0395246003723278 * ( regu
lar_checkup_lasy_year ) +  -0.018066030038938842 * ( Year_last_admitted ) +  0.9591661094999135 * ( weight ) +  0.0389453621891
1404 * ( covered_by_any_other_company ) +  0.00042240860003724514 * ( exercise ) +  0.021887596376167724 * ( weight_change_in_l
ast_one_year )
```

And hence we can say the above variables are important for predicting insurance cost and Weight is of higher significance.

Efforts to improve model performance

Performing hyperparameter tuning on all the models

Hyperparameter tuning, also known as model selection, is the process of selecting the optimal hyperparameters for a machine learning algorithm to achieve the best performance on a given task.

```
Running grid search for Linear Regression
Best parameters: {}
RMSE on test set: 0.5098293578750587

Running grid search for Lasso
Best parameters: {'alpha': 0.1}
RMSE on test set: 0.6072889602445415

Running grid search for Elastic Net
Best parameters: {'alpha': 0.1, 'l1_ratio': 0.9}
RMSE on test set: 2.4628031974402678

Running grid search for Ridge
Best parameters: {'alpha': 0.1}
RMSE on test set: 0.5110775005046037

Running grid search for Random Forest
Best parameters: {'max_depth': None, 'n_estimators': 100}
RMSE on test set: 105.6814947836617
```

Hyperparameter tuning is done because the choice of hyperparameters can have a significant impact on the performance of a machine-learning model. Choosing the wrong hyperparameters can lead to poor performance, overfitting, or underfitting of the model while selecting the optimal hyperparameters can improve the accuracy, generalization, and robustness of the model.

The above image shows the best parameters chosen to improve model performance.

Training and evaluating with Ada boost

AdaBoost, short for Adaptive Boosting, is a machine learning algorithm that combines multiple weak learners (models with only slightly better than random performance) to create a strong learner with high accuracy.

```
Training and evaluating Linear Regression with AdaBoost
RMSE: 0.5157495226560284
Accuracy: 0.62

Training and evaluating Lasso with AdaBoost
RMSE: 1.841551550354643
Accuracy: 0.22333333333333333

Training and evaluating Elastic Net with AdaBoost
RMSE: 49.89120067242322
Accuracy: 0.016666666666666666

Training and evaluating Ridge with AdaBoost
RMSE: 0.547347551897734
Accuracy: 0.59

Training and evaluating Random Forest with AdaBoost
RMSE: 91.8021001706508
Accuracy: 0.0
```

Some of the advantages of AdaBoost are:

1. It can handle high-dimensional data with a large number of features.
2. It is less prone to over-fitting than some other machine learning algorithms.
3. It is a flexible algorithm that can be used.

The above image shows the best parameters chosen to improve model performance by the Ada Boost classifier

Grid search cv

Grid search is a hyperparameter tuning technique used in machine learning to exhaustively search over a predefined set of hyperparameters for a given model and identify the combination that results in the best performance. Grid search is typically done using cross-validation to evaluate the performance of each combination of hyperparameters.

Grid search is a powerful technique for optimizing the hyperparameters of a machine learning model, as it allows the user to explore a large space of hyperparameters without manually testing each combination. However, grid search can be computationally expensive, especially for models with a large number of hyperparameters or large datasets. To address this, more advanced techniques such as randomized search or Bayesian optimization can be used to more efficiently search the hyperparameter space.

```
Performing grid search and evaluation for Linear Regression
Best parameters: {'fit_intercept': False}
RMSE: 0.5076792951575849
R^2 score: 0.9999956380583914

Performing grid search and evaluation for Lasso
Best parameters: {'alpha': 0.1}
RMSE: 0.6049551851843423
R^2 score: 0.9999938063401451

Performing grid search and evaluation for Elastic Net
Best parameters: {'alpha': 0.1, 'l1_ratio': 0.75}
RMSE: 6.339751858115141
R^2 score: 0.999319786161242

Performing grid search and evaluation for Ridge
Best parameters: {'alpha': 0.1}
RMSE: 0.5078764830583072
R^2 score: 0.9999956346692866

Performing grid search and evaluation for Random Forest
Best parameters: {'max_depth': None, 'n_estimators': 100}
RMSE: 127.74799631609027
R^2 score: 0.7238092444073922
```

To perform a grid search CV, the user specifies a set of hyperparameters and a range of values for each hyperparameter, and the algorithm systematically tests all possible combinations of hyperparameters using cross-validation. The best combination of hyperparameters is then selected based on the average performance across all folds.

From the above images we can see that grid search cv is giving good results as compared to Ada boost as the accuracy has drastically fallen down but in grid search cv it is really good.

5. Model validation

Evaluating the performance of a Machine learning model is one of the important steps while building an effective ML model. To evaluate the performance or quality of the model, different metrics are used, and these metrics are known as performance metrics or evaluation metrics.

These performance metrics help us understand how well our model has performed for the given data. In this way, we can improve the model's performance by tuning the hyperparameters. Each ML model aims to generalize well on unseen/new data, and performance metrics help determine how well the model generalizes on the new dataset.

In machine learning, each task or problem is divided into classification and Regression. Not all metrics can be used for all types of problems; hence, it is important to know and understand which metrics should be used. Different evaluation metrics are used for both Regression and Classification tasks.

A predictive regression model predicts a numeric or discrete value. The metrics used for regression are different from the classification metrics. It means we cannot use the Accuracy metric (explained above) to evaluate a regression model; instead, the performance of a Regression model is reported as errors in the prediction.

The metrics which we are going to evaluate for our regression models are:

- Mean absolute percentage error (MAPE)

  Mean absolute percentage error (MAPE) is a metric that defines the accuracy of a forecasting method. It represents the average of the absolute percentage errors of each entry in a dataset to calculate how accurate the forecasted quantities were in comparison with the actual quantities.
  The closer the MAPE value to 0, the better the predictions.

- Root Mean Square error (RMSE)

  RMSE stands for Root Mean Square Error, and it is a commonly used measure of the difference between the predicted and actual values in statistical analysis and machine learning.
  The RMSE is a type of performance metric used to evaluate the accuracy of a regression model. It measures the square root of the average squared difference between the predicted values and the actual values in a dataset.
  The value of RMSE that can be considered good or acceptable depends on the scale and range of the data being analyzed. For example, a low RMSE may be around 1-2 units for data on a small scale, while a high RMSE may be around 50-100 units for data ona large scale.
  It is also common to compare the RMSE of different models or methods applied to the same data, and the one with the lowest RMSE is generally considered to be the best-performing model.

- R2 Score

  The R2 score is a very important metric that is used to evaluate the performance of a regression-based machine learning model. It is pronounced as R squared and is also known as the coefficient of determination. It works by measuring the amount of variance in the predictions explained by the dataset.
  If the value of the r-squared score is 1, it means that the model is perfect, and if its value is 0, it means that the model will perform badly on an unseen dataset. This also implies that the closer the value of the r-squared score is to 1, the more perfectly the model is trained.

- Model Score
  It determines the accuracy of the model.

Metrics of all the models without scaling

	RMSE-Train	RMSE-Test	R2-train	R2-test	Model Score-train	Model Score-test	MAPE-train	Mape-Test
Lasso Regression	3377.90	3331.940	0.94	0.94	0.94	0.94	15.25	15.00

Ridge Regression	3377	3331.94	0.94	0.94	0.94	0.94	15.25	15.00
Elastic Net	7.74	8.32	0.99	0.99	0.99	0.99	7.99	8.46

	RMSE-Train	RMSE-Test	R2-train	R2-test	MAPE-Train	MAPE Test -test	Model score - train	Model score -Test
Linear Regression	3377.90	3331.94	0.94	0.94	0.1525	0.15	0.94	0.95
Random Forest	1151.78	3033.35	0.99	0.95	0.04	0.11	0.97	0.95

The table provided shows the performance metrics of five different machine learning models: Lasso Regression, Ridge Regression, Elastic Net, Linear Regression, and Random Forest. The performance metrics used to evaluate the models are Root Mean Squared Error (RMSE), R-squared (R2), Model Score, and Mean Absolute Percentage Error (MAPE).

Based on the provided metrics, it is clear that Elastic Net has the best performance among the five models, with the lowest RMSE (both for train and test), highest R-squared (both for train and test), and lowest MAPE (both for train and test). Elastic Net is a combination of Lasso and Ridge regression techniques and is known to perform well in situations where both Lasso and Ridge fail.

Random Forest also performed well with a low RMSE (although higher than Elastic Net) and a high R-squared for train data. However, it has a much higher RMSE for test data compared to Elastic Net, indicating overfitting on the training data.

Both Lasso and Ridge regression techniques have similar performance with a higher RMSE compared to Elastic Net and Random Forest. Linear Regression has the lowest R2 score and relatively higher RMSE compared to the other models, indicating poor performance.

Based on the given performance metrics, Elastic Net seems to be the best model to choose for the given problem. It has the lowest RMSE, highest R-squared, and lowest MAPE, indicating better performance on both training and testing data. However, other factors such as the interpretability of the model, computational complexity, and time required to train the model also need to be considered before choosing the best model.

Metrics comparison of all the models with Scaling and VIF

	RMSE-Train	RMSE-Test	R2-train	R2-test	Model Score-train	Model Score-test	MAPE-train	Mape-Test
Lasso Regression	30557.64	30444.40	0.94	0.95	0.94	0.94	12349.12	10069.8
Ridge Regression	0.23	0.23	0.94	0.94	0.94	0.94	12357.94	10090.56
Elastic Net	9.11	9.77	0.99	0.99	0.99	0.99	6.58	6.69

	RMSE-Train	RMSE-Test	R2-train	R2-test	MAPE-Train	MAPE Test -test	Model score - train	Model score -Test
Linear Regression	0.23	0.23	0.94	0.94	NA	100.83	0.94	0.94
Random Forest	0.079	0.214	0.99	0.95	46.182	102.32	0.97	0.95

Based on the provided metrics, it is clear that Elastic Net has the best performance among the five models, with the lowest RMSE (both for train and test), highest R-squared (both for train and test), and lowest MAPE (both for train and test). Elastic Net is a combination of Lasso and Ridge regression techniques and is known to perform well in situations where both Lasso and Ridge fail.

Random Forest also performed well with a low RMSE for training data, but a slightly higher RMSE for test data, indicating slight overfitting. Random Forest has the highest R-squared for training data, but a slightly lower R-squared for test data compared to Elastic Net.

Both Lasso and Ridge regression techniques have similar performance, with a higher RMSE compared to Elastic Net and Random Forest. Linear Regression has the lowest R2 score and relatively higher RMSE compared to the other models, indicating poor performance.

6. Final interpretation/recommendation

- The main variables identified during analysis are, “Weight”, exercise, covered by any other company, and weight change in the last 1 year. Therefore, while creating insurance plans company should keep these parameters in mind.

- **Encourage Long-Term Insurance**:

  Offering discounts or loyalty programs to long-term customers can be a win-win situation for both the insurance company and its customers. Long-term customers are more likely to refer the company to their friends and family, which can lead to more business for the company. Furthermore, loyal customers can act as brand ambassadors, helping to increase the company's reputation and brand awareness. In return, customers who have been with the company for a certain number of years can enjoy benefits such as reduced premiums, extended coverage, or exclusive perks that are not available to new customers.
  For example, the insurance company can create a loyalty program where customers who have been with the company for a certain number of years are rewarded with points that can be redeemed for discounts, free upgrades, or other benefits. Alternatively, the company can offer reduced premiums for long-term customers, where the discount increases with the number of years the customer has been with the company. These types of incentives not only encourage long-term insurance but also provide customers with a sense of value and appreciation for their loyalty to the company.

- **Develop Tailored Products for Students**
  Students have unique needs when it comes to insurance, as they are more likely to engage in risky behaviors and participate in activities that put them at a higher risk of accidents or injuries. By creating insurance products that cater to the needs of

students, the insurance company can better serve this demographic and potentially attract more student customers.

For example, the insurance company can create insurance plans that cover accidents or injuries that are common among students, such as sports-related injuries, mental health issues, or accidental damage to electronic devices. Additionally, the insurance company can offer flexible plans that cater to students' changing needs, such as study abroad coverage or temporary insurance for internships.

To further target the student demographic, the insurance company can collaborate with universities or colleges to offer insurance plans to students at a discounted rate. This not only provides students with more affordable insurance options but also helps to increase the company's visibility and reputation among the student population. The insurance company can also leverage social media and other digital channels to reach out to students and promote their tailored insurance products.

By developing insurance products that cater to the unique needs of students and offering them at a discounted rate, the insurance company can potentially attract more student customers and build a stronger presence in the education market.

- **Target Business Professionals**

  Since business professionals are the second biggest contributors to the insurance cost, the company can create marketing campaigns that target this demographic. For instance, the company can partner with business organizations or events to promote their insurance products to business professionals.

- **Address the One-Year Drop-Off**

The fact that most individuals have been with the company for less than a year suggests that there may be issues that are causing them to switch to other insurance providers. To address this issue, the insurance company can investigate the reasons for the drop-off and create targeted strategies to retain customers.

For instance, the company can conduct surveys or interviews with its customers to gather feedback on their experiences with the company. Based on this feedback, the company can identify areas that need improvement, such as customer service, claims handling, or insurance plan coverage. The company can then create targeted strategies to address these issues, such as providing more training to customer service representatives or expanding insurance plan coverage to better meet customer needs.

Additionally, the company can personalize its approach to customer retention by offering personalized insurance recommendations based on the customer's individual needs and preferences. For example, the company can leverage customer data to offer customized insurance plans that are tailored to the customer's specific lifestyle and risk profile.

Finally, the company can consider offering incentives or rewards to customers who stay with the company beyond the first year. For instance, the company can offer a discount on premiums for customers who renew their policy after the first year or provide loyalty points that can be redeemed for exclusive benefits.

By addressing the issues that are causing the one-year drop-off and creating targeted strategies to retain customers, the insurance company can improve its customer retention rates and build a stronger relationship with its customers.

- **Individual Profile Analysis**

It's important to maintain healthy glucose, cholesterol, and body fat levels for overall health. Consult with a healthcare provider to determine any underlying conditions or necessary lifestyle changes. Maintaining a balanced diet, regular exercise, and proper hydration can help regulate these levels and improve overall well-being. Age-specific recommendations should also be considered for body fat percentage.

- **Population Behavior Parameters**

Since 30.22% of the information is not available for the smoking status column, it may be useful to try to obtain more data to gain a better understanding of the population and their behaviors.

As the data suggests, a smaller percentage of the population are daily alcohol consumers and a majority engage in moderate exercise. Encouraging and promoting these healthy lifestyle habits can have positive impacts on overall health and well-being.

Encourage adventure sports participation: While a significant majority of the population does not participate in adventure sports, it may be beneficial to encourage and provide opportunities for individuals to try these activities. Participation in adventure sports can provide numerous benefits such as increased physical activity, improved mental health, and a sense of accomplishment.