CAPSTONE PROJECT – PGP-DSBA
HEALTHCARE PROJECT
LIFE INSURANCE COST PREDICTION

By

Roshinipriya Chandrasekaran Bala

CONTENTS

LIST OF FIGURES

1. Introduction

a. Brief introduction about the problem statement and the need of solving it

Defining problem statement:

We all know that Health care is very important domain in the market. It is directly linked with the life of the individual; hence we have to be always be proactive in this particular domain. Money plays a major role in this domain, because sometime 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 the 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 individual eat healthy and do proper exercise the chance of getting ill is drastically reduced.

The problem revolves around predicting the **optimum health insurance cost** for an individual based on their **health and habit-related parameters**. Health insurance companies need to balance affordability for customers while minimizing their own financial risk. The challenge is to create a data-driven model that estimates insurance costs **fairly and accurately**, considering various factors such as age, BMI, smoking habits, exercise routine, and medical history.

Need of the study/project:

This project helps the insurance company in accurate risk assessment of policyholders. Aids in pricing optimization by aligning premium costs with health profiles. Reduces financial losses from under-pricing high-risk individuals. Ensures fair pricing of health insurance based on **actual health risk**. Encourages **healthier lifestyle choices** by showcasing how habits impact premium costs. Provides **financial security** by making health insurance accessible and affordable. Helps **healthcare providers** understand the impact of lifestyle factors on medical costs. Supports **preventive healthcare initiatives** by rewarding healthy behaviour with lower premiums. The objective of this project is to build a model, using data that provide the optimum insurance cost for an individual.

Understanding business/social opportunity

Using data-driven models, insurers can predict the insurance cost of newly joining customers and can create **tailored** health plans. This prediction helps the insurance companies to create affordable and optimized pricing which can attract **more customers** to buy health insurance. This project bridges the gap between affordability and risk management, ultimately benefiting both insurers and customers. It also helps in educating individuals on how their lifestyle impacts insurance costs thereby promotes a **healthier society**. Data from **wearables (Fitbit, Apple Watch, etc.)** can be integrated into insurance models to offer **personalized pricing**. Companies can incentivize customers to maintain **healthy activity levels** through discounts or rewards. Insurance firms can expand beyond policies and offer **subscription-based health services** like telemedicine, diet plans, or mental health support. Such services create an **additional revenue stream** while ensuring better health outcomes for customers.

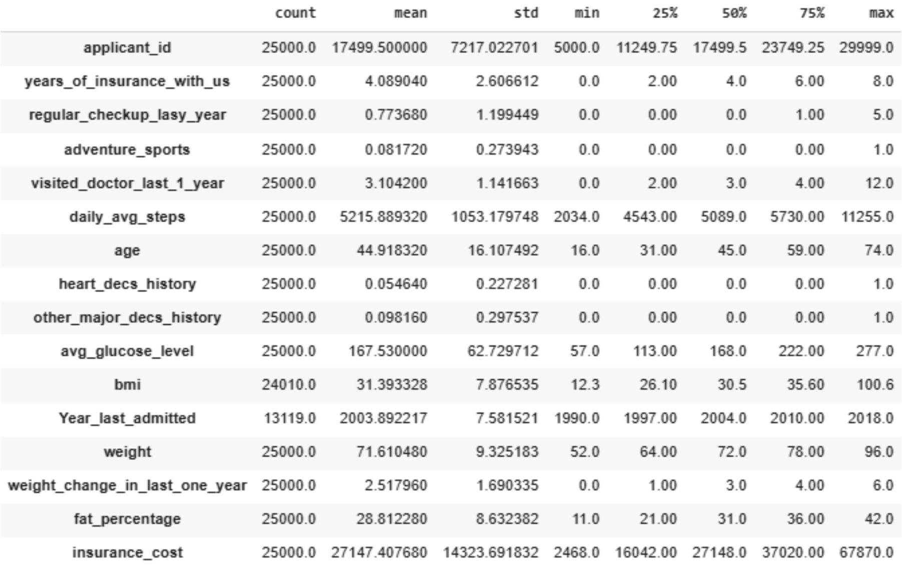
2. EDA and Business Implication

a. Uni-variate / Bi-variate / multi-variate analysis to understand relationship b/w variables - Both visual and non-visual understanding of the data

Understanding data collection helps in handling missing values, ensuring data accuracy, and designing a robust predictive model. Hence, I have made the following assumptions about the given dataset:

Variables	**Possible Frequency, Time**	**Methodology**
applicant_id, age, occupation, location, Gender, covered_by_any_other_company, years_of_insurance_with_us	Collected at **policy application** and updated **annually** or at major life events	Direct Customer Input
bmi, cholesterol_level, avg_glucose_level, fat_percentage, weight, heart_decs_history, other_major_decs_history	Likely collected during **annual medical checkups**	Medical Checkups & Health Records
alcohol, smoking_status, exercise, adventure_sports, daily_avg_steps, weight_change_in_last_one_year	Could be **self-reported** (monthly/yearly surveys) or **automated from fitness devices**	Wearable Devices & Fitness Trackers
visited_doctor_last_1_year, regular_checkup_lasy_year, Year_last_admitted	Likely sourced from **healthcare provider records** or **self-reported annually**	Medical Checkups & Health Records
insurance_cost	Updated **annually at renewal** based on risk assessment	Insurance Company Internal Records

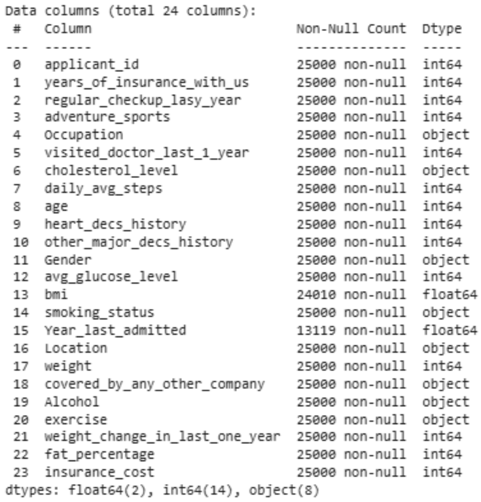
The given dataset has 25000 rows and 24 columns. Shape of the dataset is (25000, 24).



	count	mean	std	min	25%	50%	75%	max
applicant_id	25000.0	17499.500000	7217.022701	5000.0	11249.75	17499.5	23749.25	29999.0
years_of_insurance_with_us	25000.0	4.089040	2.606612	0.0	2.00	4.0	6.00	8.0
regular_checkup_lasy_year	25000.0	0.773680	1.199449	0.0	0.00	0.0	1.00	5.0
adventure_sports	25000.0	0.081720	0.273943	0.0	0.00	0.0	0.00	1.0
visited_doctor_last_1_year	25000.0	3.104200	1.141663	0.0	2.00	3.0	4.00	12.0
daily_avg_steps	25000.0	5215.889320	1053.179748	2034.0	4543.00	5089.0	5730.00	11255.0
age	25000.0	44.918320	16.107492	16.0	31.00	45.0	59.00	74.0
heart_decs_history	25000.0	0.054640	0.227281	0.0	0.00	0.0	0.00	1.0
other_major_decs_history	25000.0	0.098160	0.297537	0.0	0.00	0.0	0.00	1.0
avg_glucose_level	25000.0	167.530000	62.729712	57.0	113.00	168.0	222.00	277.0
bmi	24010.0	31.393328	7.876535	12.3	26.10	30.5	35.60	100.6
Year_last_admitted	13119.0	2003.892217	7.581521	1990.0	1997.00	2004.0	2010.00	2018.0
weight	25000.0	71.610480	9.325183	52.0	64.00	72.0	78.00	96.0
weight_change_in_last_one_year	25000.0	2.517960	1.690335	0.0	1.00	3.0	4.00	6.0
fat_percentage	25000.0	28.812280	8.632382	11.0	21.00	31.0	36.00	42.0
insurance_cost	25000.0	27147.407680	14323.691832	2468.0	16042.00	27148.0	37020.00	67870.0

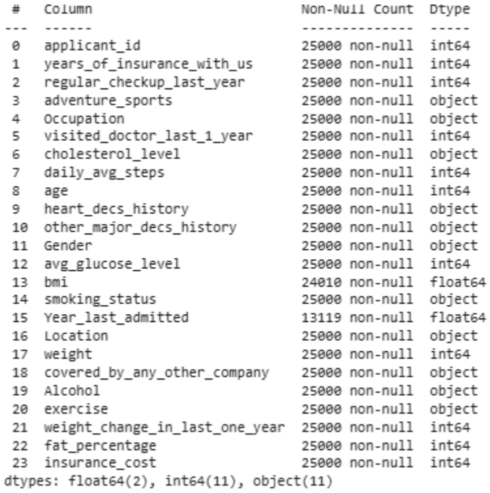
Based on these descriptive statistics, it is observed that most customers have insurance for **4–6 years**, but some are newly insured. Most people go for **1 checkup per year**, while some never visit the doctor. **Only 8% of individuals engage in adventure sports**, which could be a **high-risk factor** affecting insurance costs. Individuals **visit doctors 3 times a year on average**, but some require **frequent visits (up to 12 times)**, possibly indicating chronic conditions. Many individuals have **low step counts**, suggesting potential health risks. **High variance in insurance costs**, suggests that personal health and habits significantly influence pricing. The **average BMI is high**, indicating that many individuals are **overweight or obese.** bmi and Year_last_admitted have **missing values.**

The variables in the given dataset are identified as follows:



```
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)
```

The variables **'adventure_sports', 'heart_decs_history', and 'other_major_decs_history'** were originally identified as integer variables, but they only hold **binary values (0 or 1) which signifies either Yes or No**. Therefore, I have converted them to **categorical variables** to better reflect their nature. Additionally, **'regular_checkup_lasy_year'** has been renamed to correct the spelling, ensuring clarity and consistency in the dataset.



```
 #   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_last_year      25000 non-null  int64  
 3   adventure_sports               25000 non-null  object 
 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  object 
 10  other_major_decs_history       25000 non-null  object 
 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(11), object(11)
```

Now there are **13 continuous variables and 11 categorical variables** in the dataset.

Univariate analysis focuses on analysing and describing the distribution, central tendency, and spread of a single variable. Some common univariate analysis methods are Descriptive statistics, histogram, boxplot, Kernel density estimation, bar chart, pie chart, probability mass function, cumulative distribution function and so on. Here I have used histogram and boxplot.

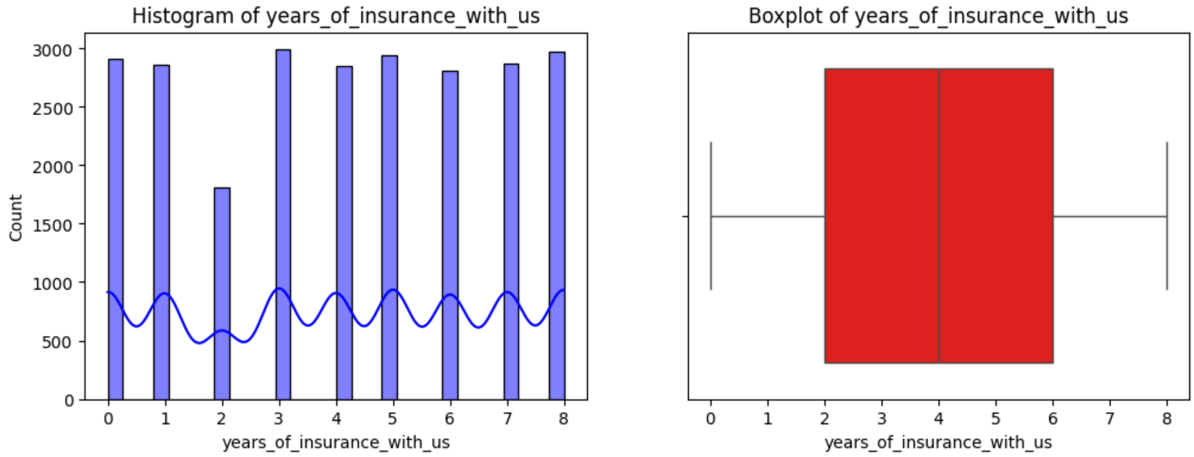


Fig 1. Distribution of 'years_of_insurance_with_us' variable

The histogram shows that the values are mostly uniformly distributed but with noticeable gaps with fewer customers at 2 years. The median appears to be around 4-5 years, indicating that half of the customers have stayed with the company for at least this duration. The boxplot shows no extreme outliers.

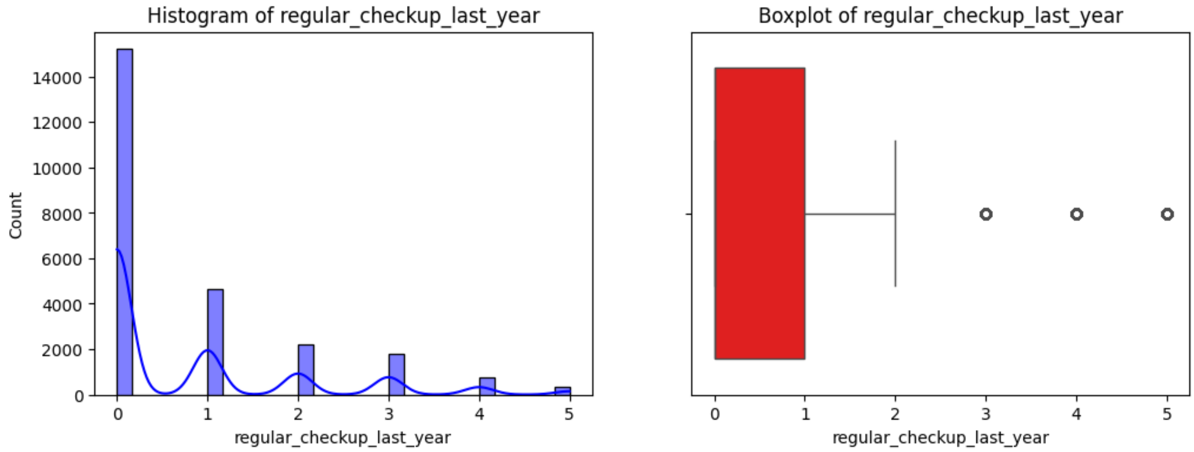


Fig 2. Distribution of 'regular_checkup_last_year' variable

The histogram clearly shows that the data is right skewed. Majority of individuals did not have regular check in the last year. This is further supported by the boxplot where the box is compressed towards the lower values, and the median line within the box is also closer to the lower end. The boxplot reveals outliers on the higher end.

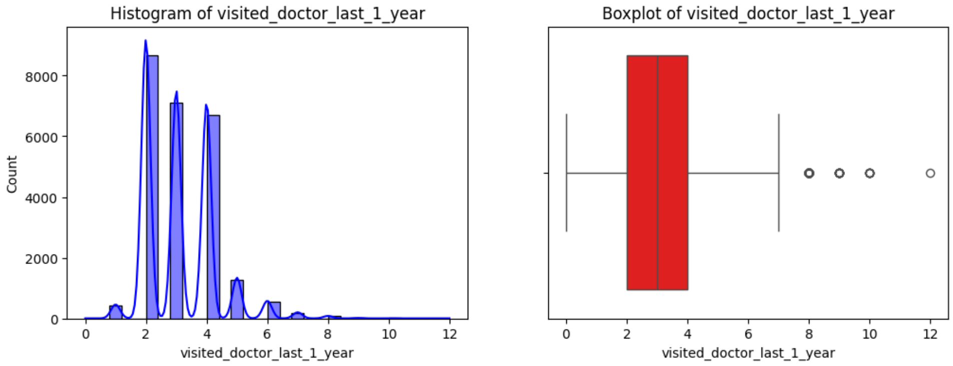


Fig 3. Distribution of 'visited_doctor_last_1_year' variable

The histogram indicates that majority of individuals have visited doctor more than once in the last year. Few outliers are seen at the higher end, indicating those individuals have visited doctor on a monthly basis in the last year.

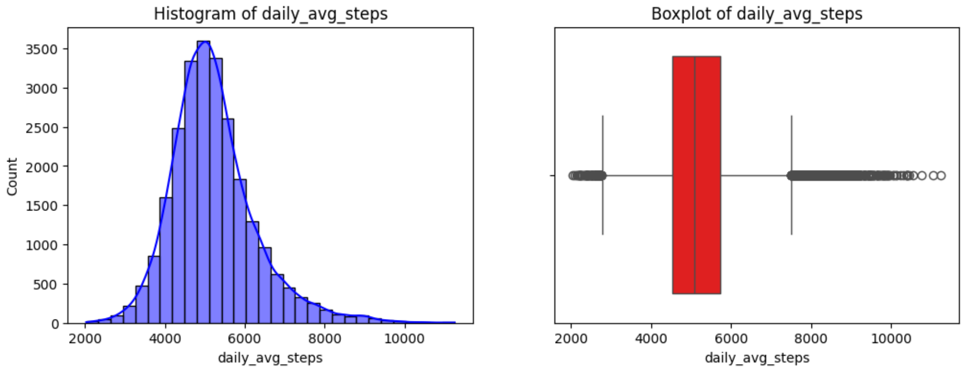


Fig 4. Distribution of 'daily_avg_steps' variable

The data is almost normally distributed and the median lies around 5000 daily average steps. Many outliers are observed at the higher end indicating that many individuals are walking more than 8000 steps a day which is a healthy sign, whereas many outliers are observed at the lower end as well, indicating those individuals are leading a sedentary lifestyle.

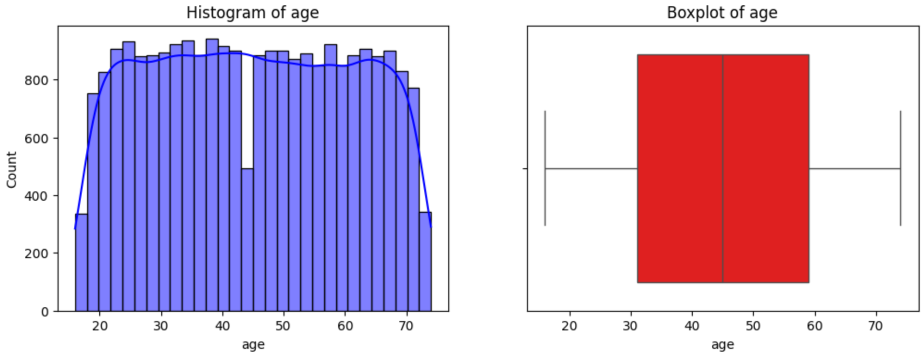


Fig 5. Distribution of 'age' variable

The dataset has individuals aging from 16 to 74. The data is almost equally spread, however few individuals are seen aged less than 20, more than 70 and in mid 40's.

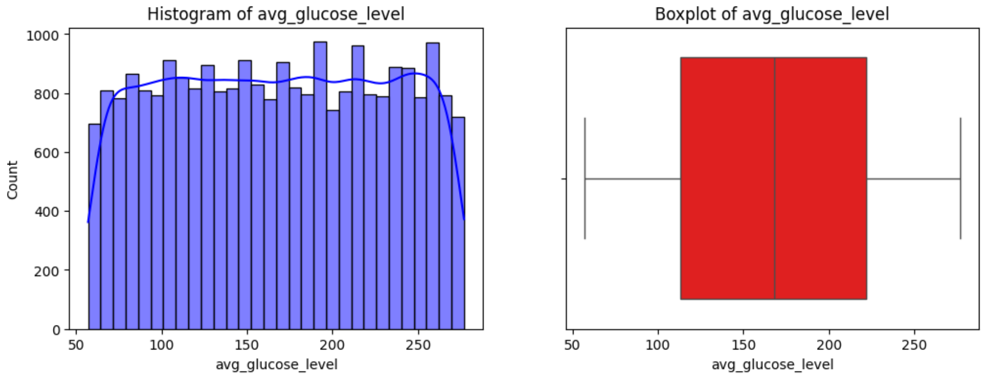


Fig 6. Distribution of 'avg_glucose_level' variable

The histogram of average glucose levels appears roughly uniform across the range, with consistent counts between 50 and 250. The boxplot indicates that the median average glucose level is around 150, showing a balanced central tendency. No outliers observed.

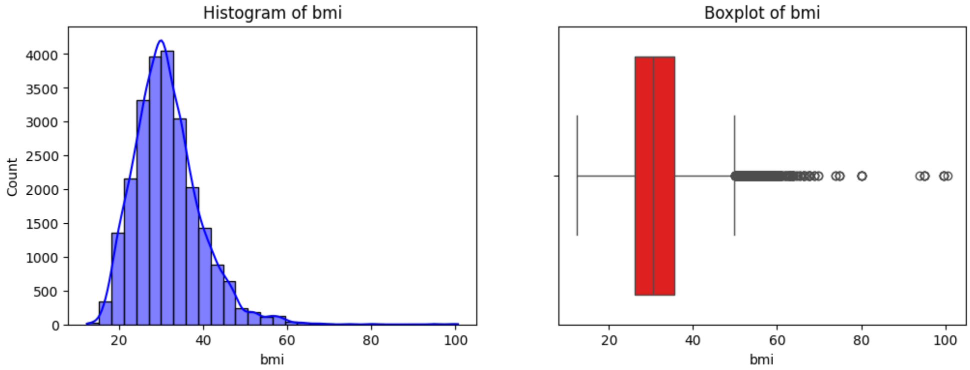


Fig 7. Distribution of 'bmi' variable

The histogram is right skewed. Majority of individuals have bmi ranging between 20 to 40. The median BMI falls around 30. The boxplot indicates the presence of significant outliers above 50.

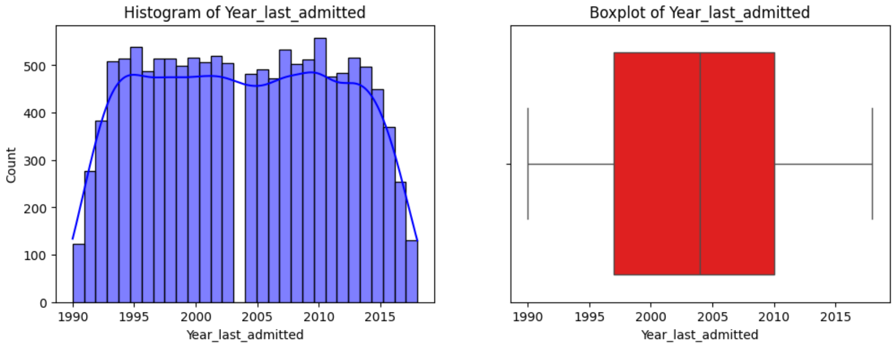


Fig 8. Distribution of 'Year_last_admitted' variable

The histogram shows that the "Year_last_admitted" is approximately uniformly distributed between 1990 and 2015, with consistent counts across the years. The median year is around 2005, indicating that half of the admissions occurred before this year and half after. No outliers seen.

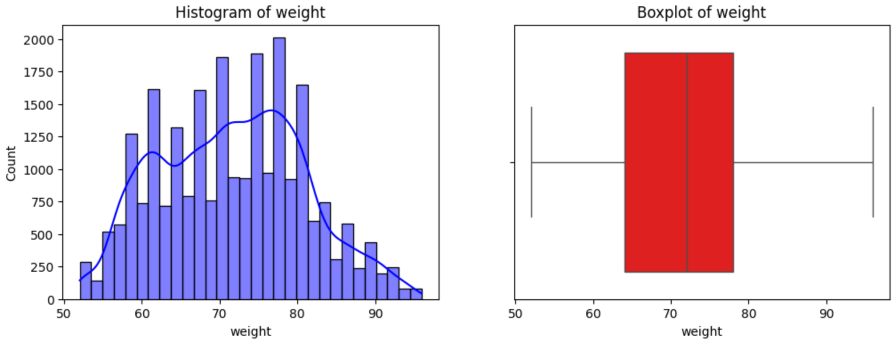


Fig 9. Distribution of ‘weight’ variable

The histogram reveals a slightly right-skewed distribution, with most weights falling between 60 and 80 kg. The peak frequency occurs around 70-75 kg, indicating the most common weight range in the dataset. Median weight around 72-73 kg. No significant outliers.

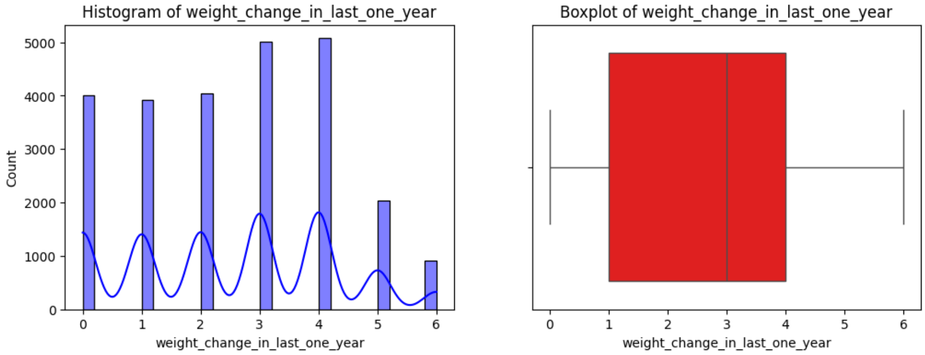


Fig 10. Distribution of ‘weight_change_in_last_one_year’ variable

The histogram shows a discrete distribution with peaks at specific intervals, suggesting that weight changes occur in consistent increments (e.g., 1, 2, 3 kg). The boxplot indicates that the majority of individuals experienced weight changes between 2 and 4 kg, with minimal outliers.

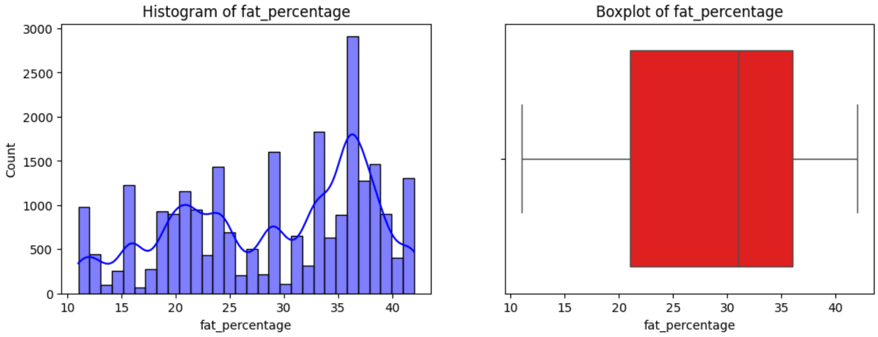


Fig 11. Distribution of ‘fat_percentage’ variable

The fat percentage variable has a slightly right-skewed distribution with most values concentrated between 20% and 40%. The boxplot confirms that the median fat percentage lies near the center of this range, with a reasonably balanced spread and no significant outliers.

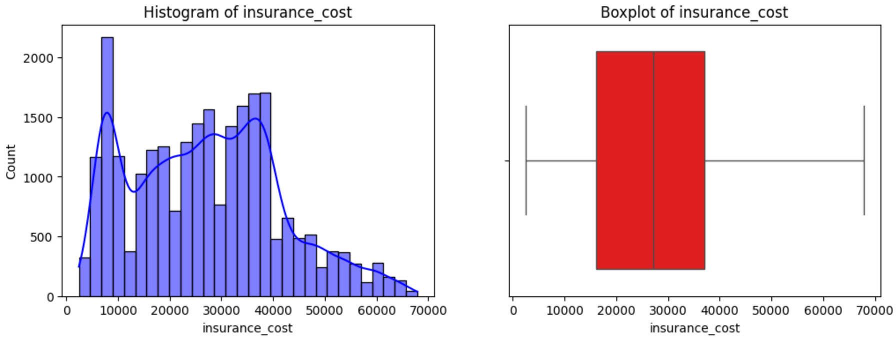


Fig 12. Distribution of 'insurance_cost' variable

The insurance cost distribution is heavily right-skewed, with a significant proportion of individuals having lower insurance costs (below 20,000). The boxplot indicates that a few outliers exist with costs exceeding 50,000, pulling the mean upwards.

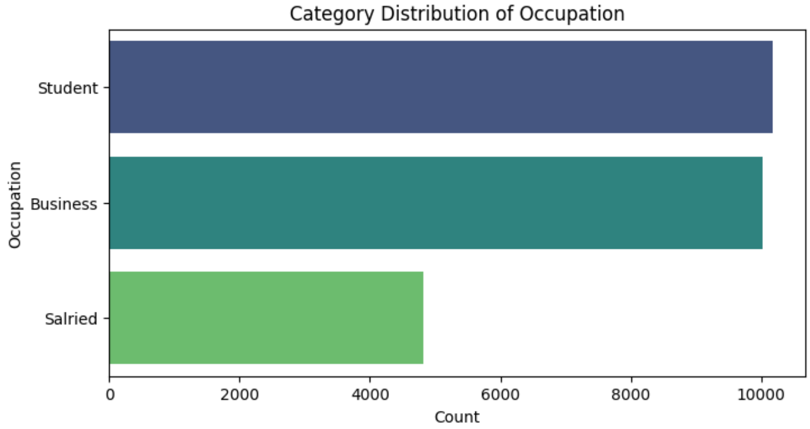


Fig 13. Distribution of 'Occupation' variable

The plot shows that most of the individuals (nearly 10000) are either Students or Business professionals. Only around 5000 individuals in the dataset are salaried.

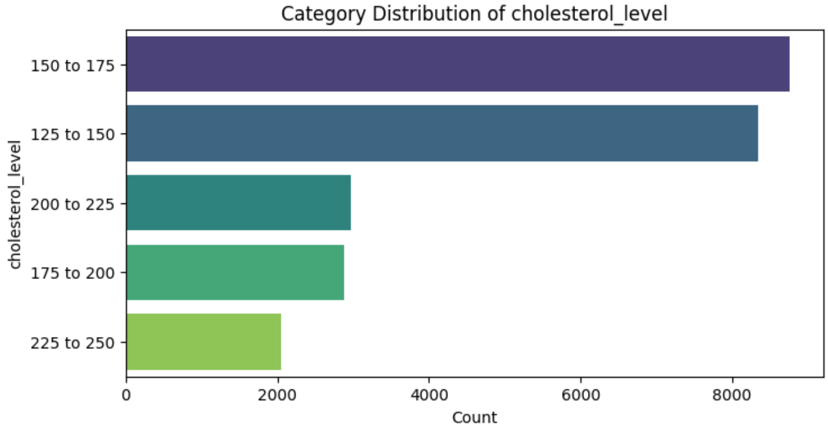


Fig 14. Distribution of 'cholesterol_level' variable

The "150 to 175" category is the most prevalent, with the highest count among all ranges. Fewer individuals have high cholesterol levels in the dataset (200 to 225, 225 to 250)

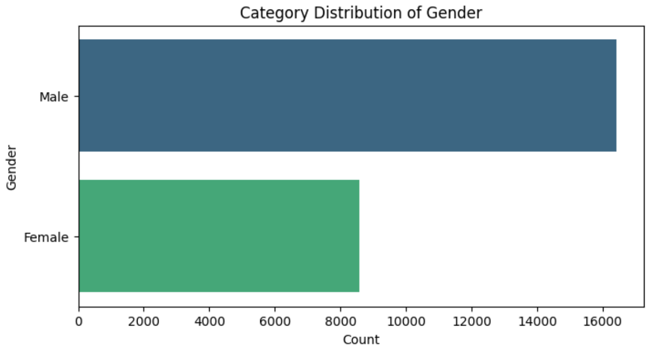


Fig 15. Distribution of ‘Gender’ variable

It is clearly seen that the dataset has a higher representation of males compared to females, as indicated by the larger count for the "Male" category.

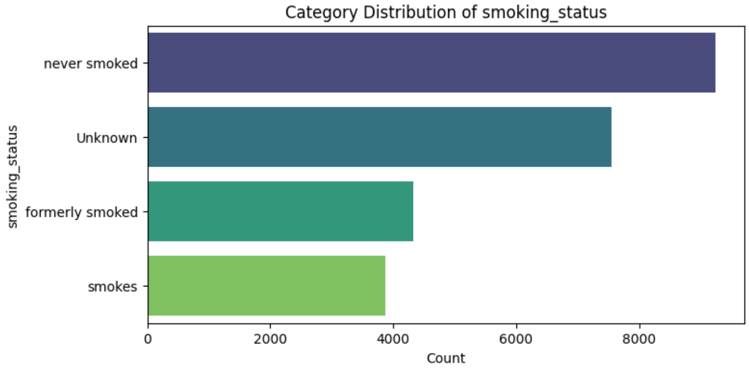


Fig 16. Distribution of ‘smoking_status’ variable

The majority of individuals in the dataset have never smoked, as "never smoked" is the most frequent category. However, a significant portion of the smoking status data is labeled as "Unknown," which could indicate missing or incomplete information.

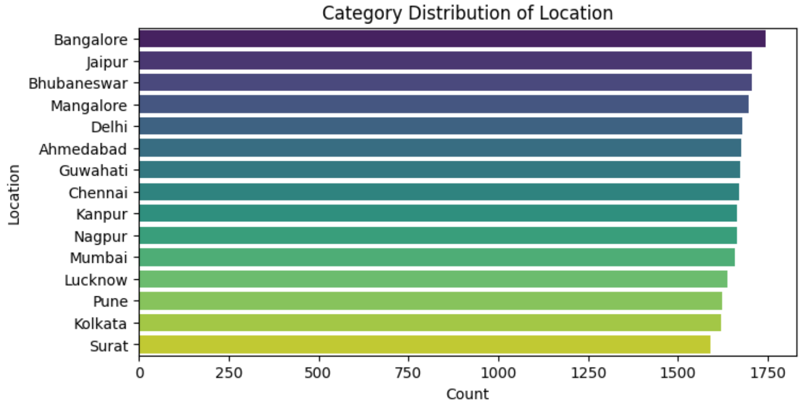


Fig 17. Distribution of ‘Location’ variable

The distribution of individuals across different locations is relatively uniform, with Bangalore having the highest representation and Surat the least. This suggests a balanced dataset across cities with minor variations in counts.

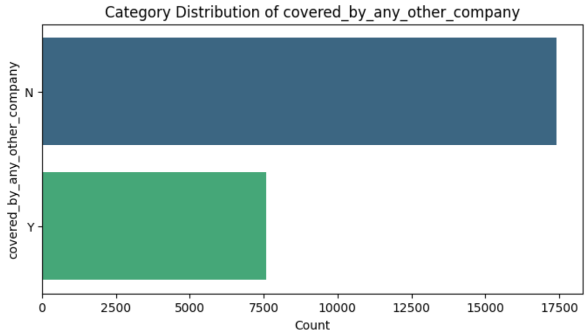


Fig 18. Distribution of 'covered_by_any_other_company' variable

The majority of individuals are not covered by any other company, with those covered forming a smaller but significant portion. This suggests that most depend on the current company for coverage, potentially impacting policy decisions.

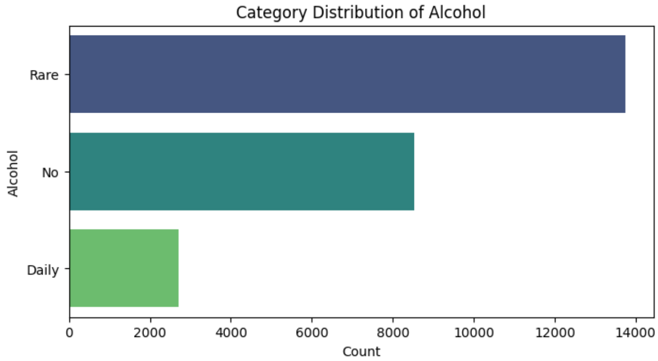


Fig 19. Distribution of 'alcohol' variable

The majority of individuals in the dataset consume alcohol rarely, followed by a substantial number of non-drinkers. Daily alcohol consumption represents the smallest group, indicating a lower prevalence of regular drinking habits. This could have implications for health-risk assessments and targeted wellness initiatives.

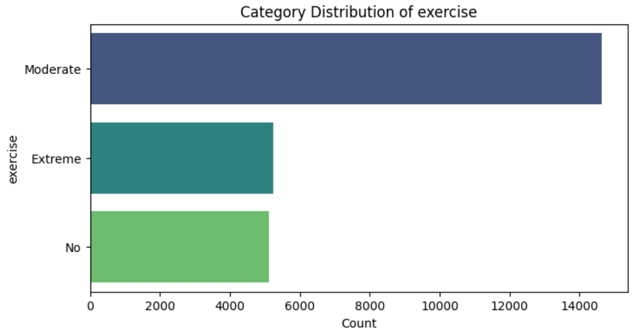


Fig 20. Distribution of 'exercise' variable

The majority of individuals in the dataset engage in moderate exercise, with fewer people engaging in extreme exercise and the smallest group not exercising at all. This suggests that moderate exercise is the most common activity level, potentially aligning with general health guidelines for physical activity.

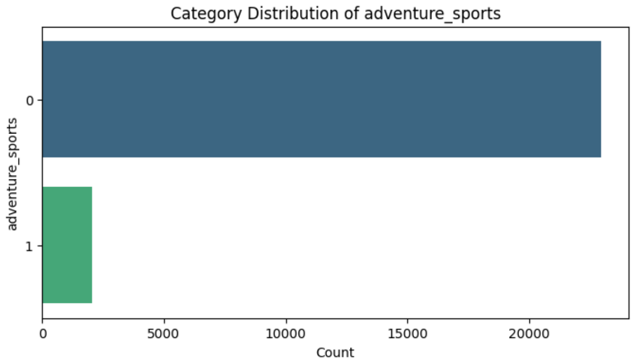


Fig 21. Distribution of ‘adventure_sports’ variable

The majority of individuals do not engage in adventure sports. This highlights that participation in high-risk physical activities is relatively uncommon.

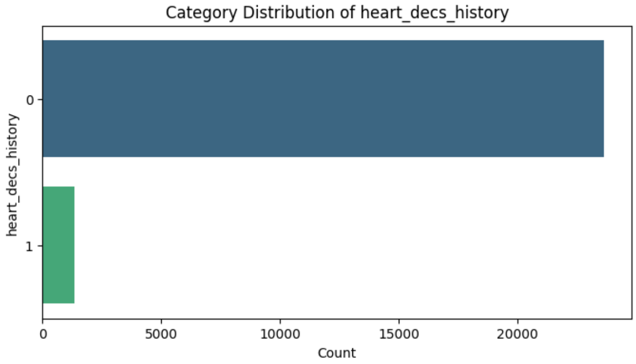


Fig 22. Distribution of ‘heart_decs_history’ variable

Most individuals do not have a history of heart diseases. However, the presence of a small group with heart disease history can indicate potential risk factors or considerations for health-related insights.

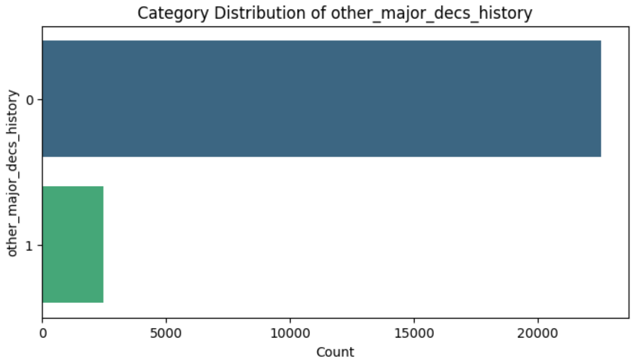


Fig 23. Distribution of ‘other_major_decs_history’ variable

Similarly, most individuals do not report other major diseases in their history, suggesting a relatively healthy population in the dataset. However, the minority with major diseases requires focused health support or interventions.

Bivariate analysis is a statistical method used to examine the relationship between **two variables**. It helps in understanding how one variable change concerning another. This type of analysis is useful for detecting associations, correlations, or dependencies between variables.

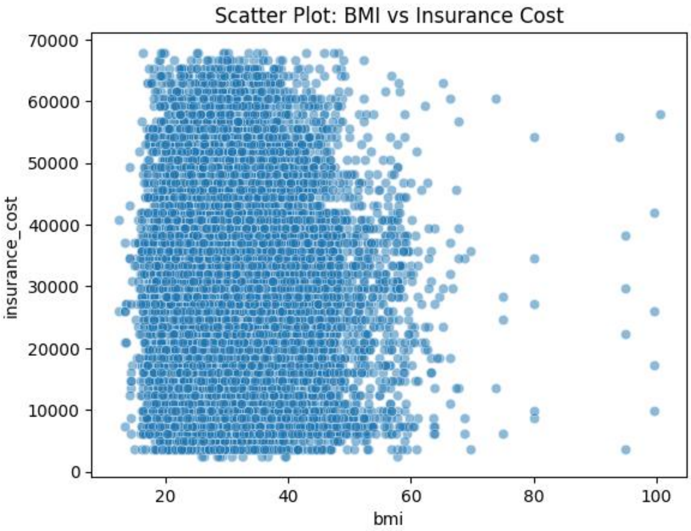


Fig 24. Scatter Plot: BMI vs Insurance cost

Most individuals fall within a moderate insurance cost range approximately 10,000–40,000, regardless of their BMI. A few individuals with very high BMI (greater than 60) exhibit notably higher insurance costs, suggesting that extreme BMI may be associated with elevated healthcare expenses in certain cases.

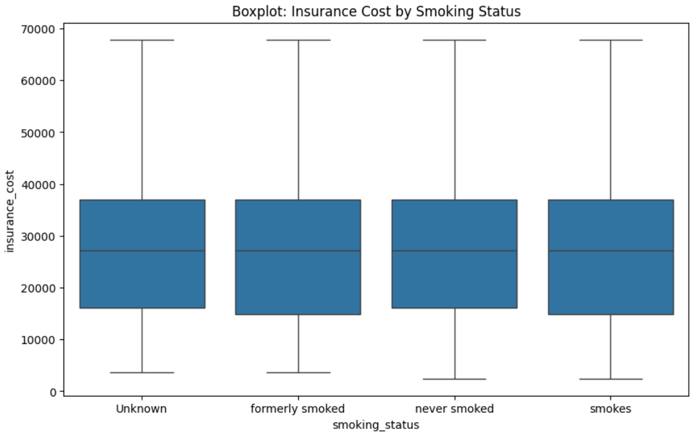


Fig 25. Boxplot: Insurance cost by smoking status

The median insurance cost is fairly consistent across all smoking status categories (Unknown, Formerly Smoked, Never Smoked, and Smokes), indicating that smoking status does not appear to significantly impact the central insurance cost value. All groups exhibit a wide range of insurance costs, with values spanning from nearly 0 to 70,000, suggesting substantial variability in costs regardless of smoking status.

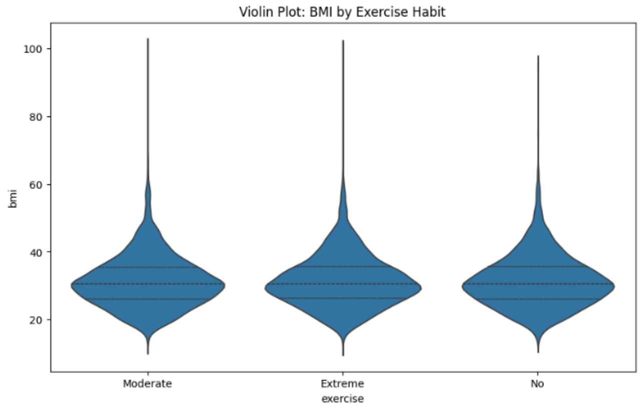


Fig 26. Violin Plot: BMI by exercise habit

The overall BMI distribution appears similar across the three exercise categories (Moderate, Extreme, and No exercise), with a comparable shape and central tendency. The median BMI is almost identical across all groups, suggesting that exercise habit does not significantly influence the central BMI value. The range of BMI values is largest for individuals in the "Moderate" and "Extreme" exercise categories, indicating greater variability compared to the "No exercise" group.

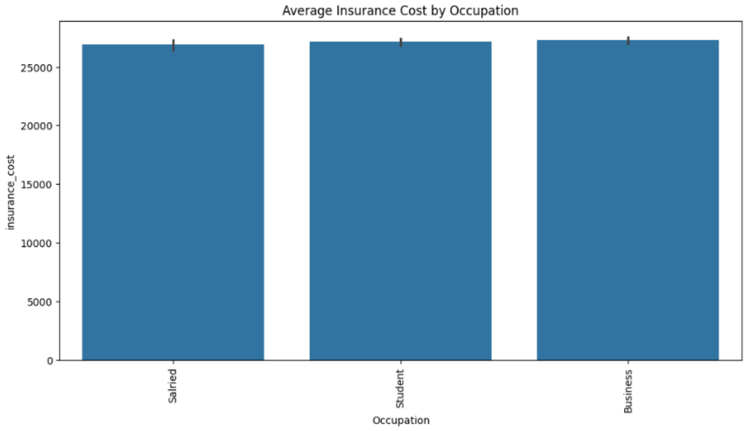


Fig 27. Bar Plot: Average Insurance Cost by Occupation

The average insurance cost is nearly the same for all occupation groups (Salaried, Student, and Business), suggesting no significant variation in costs based on occupation. Since the averages are almost identical, occupation likely does not play a major role in determining insurance costs.

Multivariate analysis is essential for understanding the relationship between variables in the dataset, how the variables interact and influence each other. Some of the commonly used methods are scatterplot, heatmap, correlation coefficient and so on. Here let’s visualize the correlation matrix using the heatmap as follows:

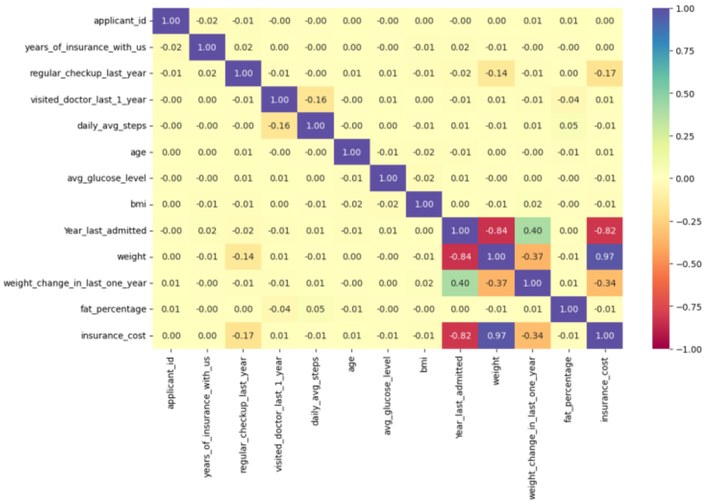


Fig 28. Heatmap – numerical variables

There is a strong positive correlation between BMI and weight (0.97), indicating that BMI is closely related to an individual's weight. Weight change in the last year and Fat percentage are highly correlated with both BMI and weight respectively, suggesting that significant weight changes as well as fat percentage are directly impacting these measures. Daily average steps have weak correlations with most health and cost variables, indicating that physical activity does not have a direct relationship with BMI, weight, or insurance cost. Age shows a weak but positive correlation (0.20) with insurance cost, indicating that older individuals might face slightly higher insurance costs.

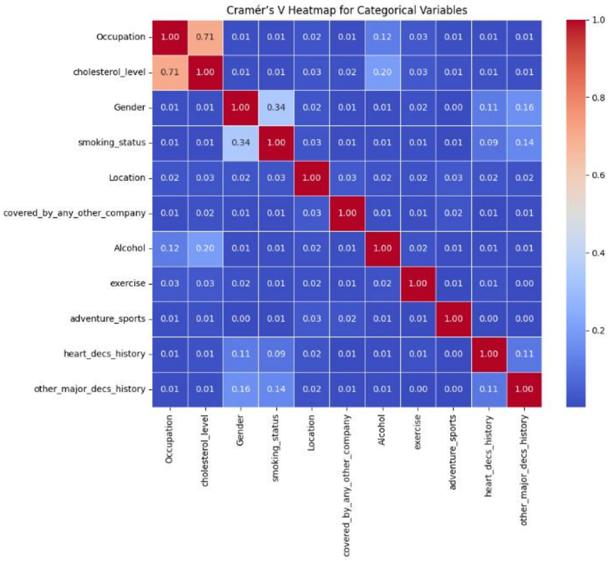


Fig 29. Heatmap – categorical variables

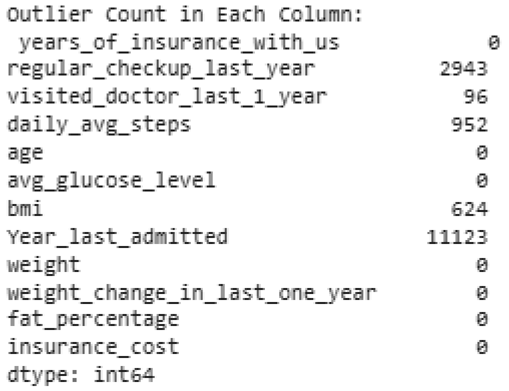
Occupation and cholesterol level show a strong association (0.71), suggesting that certain occupational categories might be linked to cholesterol levels. Gender and smoking status have a moderate association (0.34), indicating potential differences in smoking habits between genders. There is a moderate association (0.34) between gender and cholesterol level, implying that cholesterol levels may vary by gender. Alcohol and exercise have a weak association (0.19), potentially hinting at overlapping lifestyle factors.

3. Data cleaning and Pre-processing

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

The ‘bmi’ and ‘Year_last_admitted’ variables have missing values, accounting for approximately 4% (990) and 48% (11881) of the dataset, respectively. Assuming the missing values are due to data collection errors or random reasons, filling with the **median** makes more sense. I have imputed these missing values using their respective median values.

Outliers are extreme values in the dataset that significantly deviate from other observations. They can be **too high** or **too low** compared to the majority of the data. Outliers can arise due to **measurement errors, data entry mistakes, or genuine extreme cases**. **Interquartile Range** (IQR) Method is used for outlier detection. Identifies outliers based on the **1st quartile (Q1)** and **3rd quartile (Q3)**. Uses the **1.5×IQR rule** to detect outliers.

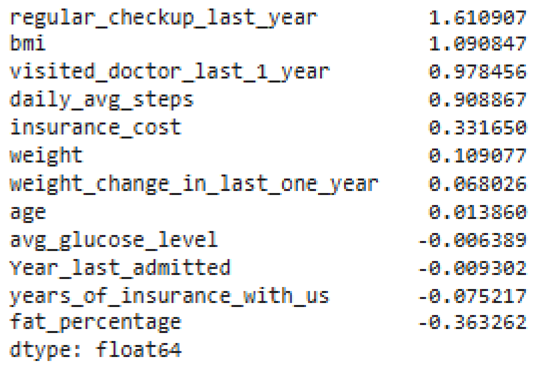


```
Outlier Count in Each Column:
 years_of_insurance_with_us          0
regular_checkup_last_year        2943
visited_doctor_last_1_year         96
daily_avg_steps                   952
age                                 0
avg_glucose_level                   0
bmi                               624
Year_last_admitted              11123
weight                              0
weight_change_in_last_one_year      0
fat_percentage                      0
insurance_cost                      0
dtype: int64
```

Since these outliers indicate specific medical conditions or concerns that need to be addressed, I prefer not to treat them. Retaining these values helps ensure the model accurately reflects the impact of such factors on insurance pricing.

b. Need for variable transformation (if any)

Variable transformation is the process of applying mathematical functions to variables in a dataset to improve the performance of statistical models or machine learning algorithms. It is commonly used to correct/reduce skewness, stabilize variance, handle outliers, and make relationships more linear. Skewness for each column is as follows:



```
regular_checkup_last_year         1.610907
bmi                               1.090847
visited_doctor_last_1_year        0.978456
daily_avg_steps                   0.908867
insurance_cost                    0.331650
weight                            0.109077
weight_change_in_last_one_year    0.068026
age                               0.013860
avg_glucose_level                -0.006389
Year_last_admitted               -0.009302
years_of_insurance_with_us       -0.075217
fat_percentage                   -0.363262
dtype: float64
```

The skewness isn't large enough to significantly affect model performance or interpretability. Hence, I choose to skip variable transformations.

c. Variables removed or added and why (if any)

At this stage of the analysis, I prefer not to add any new variables to the model. The existing features provide a sufficient foundation for predicting insurance costs, and adding additional variables could introduce unnecessary complexity. I have dropped the ‘applicant_id’ variable from the dataset as it does not provide any valuable insight for predicting insurance cost. Shape of the dataset is now (25000, 23)

4. Model building

a. Clear on why was a particular model(s) chosen:

Model building refers to the process of developing a predictive or descriptive model using the dataset, typically as part of a machine learning or statistical analysis pipeline. For this project, **I have built 3 models: Linear Regression, Decision Tree and Random Forest.** The models built using the given dataset, are designed to **predict health insurance costs** based on various features (like customer’s age, health status, lifestyle, etc.)

Linear regression is one of the simplest and most widely used models in machine learning, particularly for predicting continuous numeric outcomes. It assumes a linear relationship between the input variables (features) and the target variable (insurance cost). Since the target variable, insurance_cost, is continuous, linear regression is a basic choice to predict it. It works by finding the best-fit line (or hyperplane in higher dimensions) that minimizes the sum of squared errors between the predicted and actual target values. Errors are normally distributed and have constant variance. It is easy to implement and interpret. Provides coefficients that show the direct relationship between each feature and the target variable. Few disadvantages are it assumes linear relationships and may not perform well if the true relationship between features and the target is non-linear. The model is sensitive to outliers.

Decision Tree Regressor is a non-linear model that makes decisions by splitting the data at each node based on feature values, creating branches and leaves that lead to the predicted output. Decision trees can capture non-linear relationships in the data. Since this problem involves predicting a continuous target, a regression tree is used. Each split in the tree is based on a threshold value of a feature that leads to the most homogeneous groups of data. This hierarchical splitting process is straightforward to interpret and visualize. Handles non-linearity well. Can capture complex interactions between features. But it is prone to overfitting, especially with deep trees and noisy data. Not as smooth in its predictions compared to linear models.

Random Forest Regressor is an ensemble learning method that combines multiple decision trees. It builds many decision trees on random subsets of the data and averages their predictions to improve accuracy and reduce overfitting. It is a more robust and powerful version of the decision tree model. It works well when there are complex interactions and non-linear relationships in the dataset. Although it's harder to visualize or interpret individual trees, it aggregates the results of many trees to improve performance. It uses majority voting or averaging for classification or regression tasks. Reduces overfitting compared to a single decision tree by averaging results across multiple trees. Works well with many input features.

Ensemble modelling is a machine learning technique where multiple models are combined to improve overall prediction accuracy and reduce errors. The main idea is that different models capture different patterns in the data, and their combination results in a more robust and generalized model.

Bagging Regressor: Bagging (Bootstrap Aggregating) works by training multiple base learners (e.g., decision trees) on different random subsets of the training data. These base models are trained in parallel, and their predictions are averaged to make the final prediction. This helps in reducing variance and preventing overfitting, especially in high-variance models like decision trees.

AdaBoost Regressor: AdaBoost (Adaptive Boosting) works by sequentially training base models (often decision trees) with a focus on the errors made by the previous models. It adapts to the data by giving more weight to the samples that were misclassified, improving the model's performance iteratively.

Gradient Boosting: Gradient Boosting builds models sequentially, where each new model corrects the errors of the previous ones by focusing on the residuals. It is more sophisticated than AdaBoost, as it minimizes a loss function (e.g., MSE) in each iteration. Gradient boosting models generally perform well, especially for non-linear relationships.

XGBoost: XGBoost (Extreme Gradient Boosting) is an optimized version of gradient boosting that uses regularization techniques and more efficient algorithms. It is known for its performance, speed, and scalability. XGBoost typically outperforms other models on structured data.

b. Effort to improve model performance:

Model tuning is the process of optimizing hyperparameters to improve model performance. Unlike model parameters (which are learned from the data), **hyperparameters** are set before training and significantly impact the model's accuracy, generalization, and efficiency. Tuning **improves accuracy & generalization** on unseen data. **Reduces overfitting** (too complex models) or underfitting (too simple models). **Enhances business decision-making** by providing reliable predictions. Among all the base models and ensemble models built, I chose **Random Forest, Gradient boosting and XGBoost** models for tuning based on their performance metrics results. While all three models are strong performers, tuning their hyperparameters can significantly enhance their predictive power. Hyperparameter tuning allows the models to fit better to the training data and improve generalization to new unseen data. These models are known for their robust performance on a wide range of machine learning tasks, especially in predicting numerical values like health insurance costs.

5. Model validation

a. How was the model validated? Just accuracy, or anything else too?

Predictive models are being evaluated on the **test set** using various appropriate performance metrics, which is essential to assess how well they generalize to new, unseen data. The following metrics to evaluate your models:

1. Mean Absolute Error

MAE represents the average absolute difference between the actual and predicted values. It's easy to interpret since it directly shows how much the predictions deviate from the true values on average. It gives a sense of the average "error" in the insurance cost prediction. A lower MAE indicates better predictive performance.

2. Mean Squared Error (MSE)

MSE penalizes larger errors more heavily due to the squaring of differences. It's useful when large errors are more undesirable and should be avoided. Large errors in predicting the insurance cost might lead to significant financial miscalculations or mispricing of insurance policies, so minimizing MSE is important.

3. Root Mean Squared Error (RMSE)

RMSE provides a more interpretable error measurement in the same units as the target variable (insurance cost). Like MSE, it penalizes larger errors more, but it scales it back to the original unit. It is particularly useful if large errors are very costly for the business. A lower RMSE means better prediction accuracy with fewer significant miscalculations.

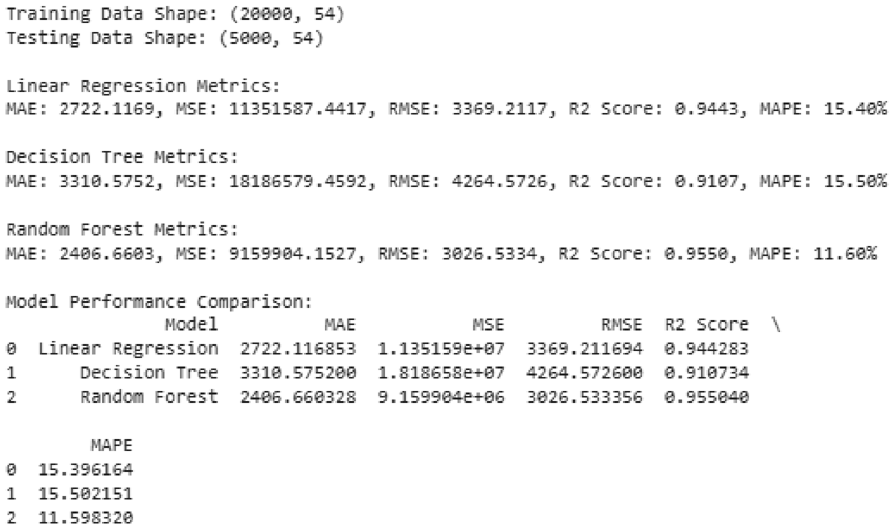
4. R-Squared (R^2)

R^2 indicates how well the model explains the variability in the target variable. It measures the proportion of the variance in the target variable that is predictable from the features. An R^2 closer to 1 indicates that the model is explaining most of the variance in insurance cost, which is ideal. It gives a sense of how well your model fits the data, with higher R^2 showing that the model does a better job of capturing the underlying patterns.

5. Mean Absolute Percentage Error (MAPE)

MAPE measures the average percentage difference between the actual and predicted values. It provides an intuitive way to understand prediction errors in relation to the actual values. Since it expresses errors as a percentage, it allows easy comparison across different scales. A lower MAPE indicates that the model's predictions are, on average, closer to the actual insurance costs in relative terms. MAPE is particularly useful when evaluating cost predictions, as it allows stakeholders to assess errors **in percentage terms rather than absolute amounts**. Lower MAPE means better predictive performance, ensuring that cost estimates remain within an acceptable range of deviation from true values.

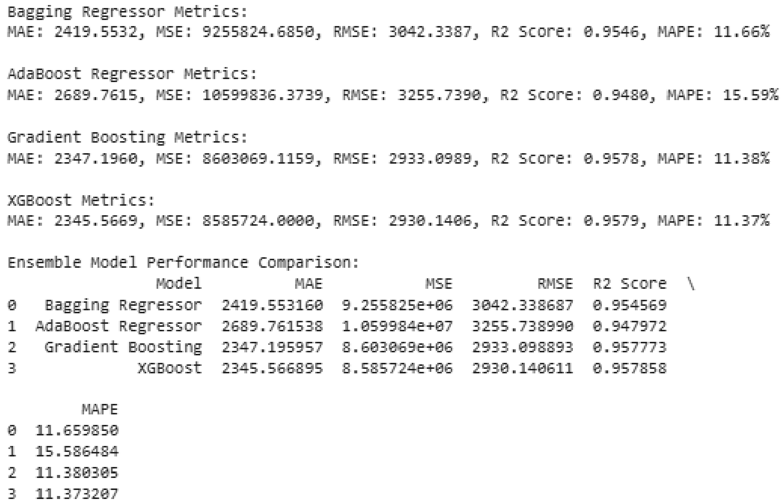
Performance metrics of base models before tuning are as follows:



```
Training Data Shape: (20000, 54)
Testing Data Shape: (5000, 54)

Linear Regression Metrics:
MAE: 2722.1169, MSE: 11351587.4417, RMSE: 3369.2117, R2 Score: 0.9443, MAPE: 15.40%

Decision Tree Metrics:
MAE: 3310.5752, MSE: 18186579.4592, RMSE: 4264.5726, R2 Score: 0.9107, MAPE: 15.50%

Random Forest Metrics:
MAE: 2406.6603, MSE: 9159904.1527, RMSE: 3026.5334, R2 Score: 0.9550, MAPE: 11.60%

Model Performance Comparison:
               Model          MAE           MSE         RMSE  R2 Score  \
0  Linear Regression  2722.116853  1.135159e+07  3369.211694  0.944283
1      Decision Tree  3310.575200  1.818658e+07  4264.572600  0.910734
2      Random Forest  2406.660328  9.159904e+06  3026.533356  0.955040

        MAPE
0  15.396164
1  15.502151
2  11.598320
```

- **Random Forest** has the best performance across all metrics, with the **lowest MAE, MSE, MAPE and RMSE** and the **highest R² score**, making it the most accurate and reliable model for predicting insurance costs in your case. This suggests that Random Forest does the best job of capturing the complexities in the data and minimizing both average error and large errors, which is crucial for the business where accurate pricing is important.
- **Linear Regression** has a lower R² and higher error metrics than Random Forest, which is expected since it assumes a linear relationship between the features and the target. This model might be useful for initial exploratory analyses, but it's not as accurate as the Random Forest model.
- **Decision Tree** performs worse than both Linear Regression and Random Forest, with a higher MAE, MSE, MAPE and RMSE. While decision trees can be valuable for interpretability, they tend to overfit the data, as seen here, leading to worse generalization on the test set.

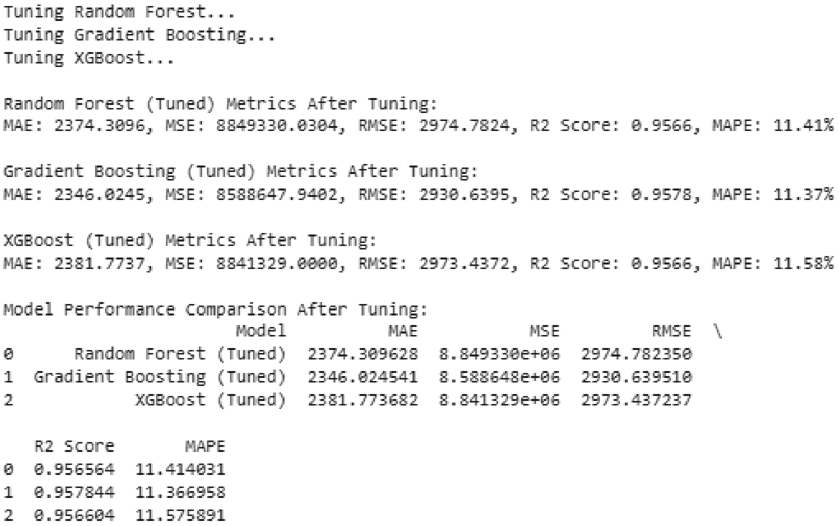
Performance metrics of ensemble models before tuning are as follows:



```
Bagging Regressor Metrics:
MAE: 2419.5532, MSE: 9255824.6850, RMSE: 3042.3387, R2 Score: 0.9546, MAPE: 11.66%

AdaBoost Regressor Metrics:
MAE: 2689.7615, MSE: 10599836.3739, RMSE: 3255.7390, R2 Score: 0.9480, MAPE: 15.59%

Gradient Boosting Metrics:
MAE: 2347.1960, MSE: 8603069.1159, RMSE: 2933.0989, R2 Score: 0.9578, MAPE: 11.38%

XGBoost Metrics:
MAE: 2345.5669, MSE: 8585724.0000, RMSE: 2930.1406, R2 Score: 0.9579, MAPE: 11.37%

Ensemble Model Performance Comparison:
                Model          MAE           MSE         RMSE  R2 Score  \
0   Bagging Regressor  2419.553160  9.255825e+06  3042.338687  0.954569
1  AdaBoost Regressor  2689.761538  1.059984e+07  3255.738990  0.947972
2   Gradient Boosting  2347.195957  8.603069e+06  2933.098893  0.957773
3             XGBoost  2345.566895  8.585724e+06  2930.140611  0.957858

        MAPE
0  11.659850
1  15.586484
2  11.380305
3  11.373207
```

- **XGBoost** is the top performer, with the highest R2 score (0.9579), meaning it is explaining nearly 96% of the variance in the target variable. It also has the lowest MAE, MSE, MAPE and RMSE values, which indicates it's making the most accurate predictions among the ensemble models.
- **Gradient Boosting** is a close second, with an R2 score of 0.9578. It slightly underperforms XGBoost in terms of prediction accuracy but is still an excellent model.
- **Bagging Regressor** comes third in terms of R2 score and performs similarly to Gradient Boosting but is outperformed by both Gradient Boosting and XGBoost in terms of MAE, MSE, and RMSE.
- **AdaBoost Regressor** has the lowest performance among the ensemble models in this case, with a higher MAE and RMSE, and a slightly lower R2 score than the others.

Performance metrics of tuned models are as follows:



```
Tuning Random Forest...
Tuning Gradient Boosting...
Tuning XGBoost...

Random Forest (Tuned) Metrics After Tuning:
MAE: 2374.3096, MSE: 8849330.0304, RMSE: 2974.7824, R2 Score: 0.9566, MAPE: 11.41%

Gradient Boosting (Tuned) Metrics After Tuning:
MAE: 2346.0245, MSE: 8588647.9402, RMSE: 2930.6395, R2 Score: 0.9578, MAPE: 11.37%

XGBoost (Tuned) Metrics After Tuning:
MAE: 2381.7737, MSE: 8841329.0000, RMSE: 2973.4372, R2 Score: 0.9566, MAPE: 11.58%

Model Performance Comparison After Tuning:
                       Model          MAE           MSE         RMSE  \
0      Random Forest (Tuned)  2374.309628  8.849330e+06  2974.782350
1  Gradient Boosting (Tuned)  2346.024541  8.588648e+06  2930.639510
2            XGBoost (Tuned)  2381.773682  8.841329e+06  2973.437237

   R2 Score       MAPE
0  0.956564  11.414031
1  0.957844  11.366958
2  0.956604  11.575891
```

- After tuning, Random Forest showed a **high R^2 score of 0.9566**, indicating that it explains about **95.66%** of the variance in the target variable (health insurance cost). The error metrics (MAE, MSE, MAPE and RMSE) are fairly low, showing good accuracy in predictions. This model performs solidly and is suitable for high-variance, complex datasets.
- **Gradient Boosting** performed slightly better than Random Forest in terms of **R^2 score (0.9578)**. The model explains **95.78%** of the variance, slightly outperforming Random Forest. Additionally, it has a **lower MAE, MAPE and RMSE**, indicating it makes smaller errors. Gradient Boosting works well on structured datasets, and this model has optimized the loss function to achieve these impressive results.
- **XGBoost**, a more efficient and optimized version of Gradient Boosting, achieved **almost identical results to Random Forest**, with an R^2 score of **0.9566**, indicating that it explains **95.66%** of the variance in the target variable. While its **MAE and RMSE** are slightly higher than Gradient Boosting, it still performed exceptionally well overall. XGBoost's efficient use of resources and model regularization techniques make it a top contender for large datasets and tasks where speed and accuracy are crucial.
- **Gradient Boosting (Tuned)** has the best performance, with the lowest **MAE** and **RMSE**, and the highest **R^2 score (0.9578)**. This model explains the most variance and makes the fewest prediction errors, making it the best choice for predicting health insurance costs.

6. Final interpretation/recommendation

a. Very clear and crisp on what recommendations do you want to give to the management / client.

By comparing all models built and their performance metrics results, **Gradient Boosting (Tuned)** has the best performance, with the lowest **MAE**, **MAPE** and **RMSE**, and the highest **R^2 score (0.9578)**. The feature importance result is as follows:

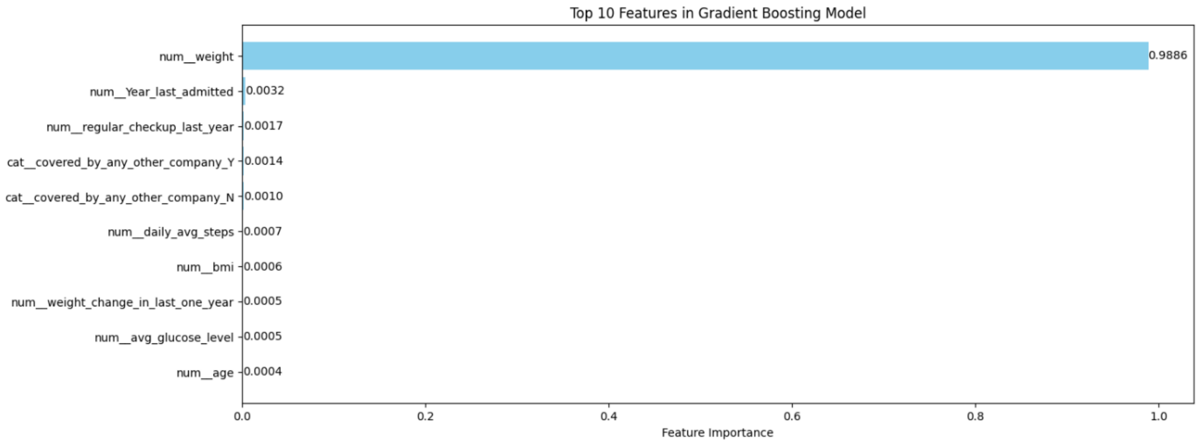


Fig 30. Top 10 Features in Gradient Boosting model

- Similar to the feature importance from random forest model before tuning, **num_weight** has an extremely high importance (0.9886), overshadowing all other features. This suggests that **weight** is the primary driver of insurance costs. Heavier individuals may be at higher risk for conditions such as heart disease, diabetes, and other comorbidities, which lead to higher insurance claims.
- num__Year_last_admitted is second most important (0.0032), but still **far behind weight**. If someone has been admitted to the hospital recently, it could signal ongoing health issues, which can increase insurance costs.
- num__regular_checkup_last_year shows moderate importance (0.0017). Having regular checkups might reduce long-term costs through early detection and management of health issues.
- **BMI, Weight Change, Glucose Level, Age** are all standard health risk indicators. While they rank lower than weight, they still play some role in fine-tuning cost predictions.

Recommendations:

1. With **weight** emerging as the dominant predictor, the insurer might consider **tiered premium structures** based on weight-related metrics (e.g., BMI).
2. Customers with higher weight may represent a higher risk pool, potentially leading to increased premiums or special wellness requirements.
3. Offering **discounts or incentives** for policyholders who maintain a healthy weight or show consistent weight reduction could help **reduce long-term claims**.

4. Features like **Year Last Admitted** and **Regular Checkup** highlight that **recent hospital admissions** or **lack of preventive care** can be indicators of rising costs.
5. The insurer can encourage **regular screenings** or provide **post-discharge care** to mitigate future risks.
6. Insights from daily steps, BMI, and glucose levels allow for **personalized insurance plans**.
7. Customers with healthier lifestyles could be rewarded with **lower premiums**, while those at higher risk might need targeted interventions or specialized plans.
8. Insurers can work with healthcare providers to design **joint initiatives** focused on **weight management**, **preventive care**, and **early detection** of weight-related health issues. These collaborations can help keep policyholders healthy and avoid **expensive claims** down the road.
9. Regular retraining of the Gradient Boosting model with up-to-date customer and health data will help the insurer remain flexible in adjusting to **new patterns** in weight-related claims and risk management.