MEDICAL INSURANCE PREDICTION

By

1.Name: Aishwarya Devisetti (1211921)
2.Name: Sucharitha Swargam (1212077)
3.Name: Praveen Kollipara (1211436)

Adviced By
Prof. Asif Mahmood

SUBMITTED IN PARTIAL FULFILMENT
OF THE REQUIRMENTS FOR THE
DEGREE OF MASTER OF SCIENCE IN
COMPUTER ENGINEERING

THE SCHOOL OF ENGINEERING
UNIVERSITY OF BRIDGEPORT
CONNECTICUT

Abstract

Health insurance is becoming an essential commodity for everyone. Insurance data has increased dramatically in the last decade. The health insurance system explores predictive modeling to boost its business operations and services. Computer algorithms and Machine Learning are used to study and analyze past insurance data and predict new output values based on trends in customer behavior, insurance policies, and data-driven business decisions. So, in this project we would like to develop a Health insurance cost prediction system using ML algorithms. The proposed model incorporates and demonstrates different models of regression.

We are using Random Forest and Linear Regression Algorithm. This project aims to brief the introduction of various computational intelligence techniques and algorithms-based approaches used for the prediction of Medical Insurance using python. This project, we are implementing using Insurance dataset. The main aim of the data is to investigate several variables like age, BMI number, Gender, Region etc., and their impact on the estimated amount was investigated.

Contents

Chapter 1

1. Introduction

This chapter discusses the aim and project objectives of the research. This chapter also discusses the project objectives in detail along with its research approach and structure of the dissertation.

The goal of this project is to allow a person to get an idea about the necessary amount required according to their own health status. Later they can comply with any health insurance company and their schemes benefits keeping in mind the predicted amount from our project. This can help a person focus more on the health aspect of insurance rather than the futile part.

Health insurance is a necessity nowadays, and almost every individual is linked with a government or private health insurance company. Factors determining the amount of insurance vary from company to company. Also, people in rural areas are unaware of the fact that the government of India provide free health insurance to those below poverty line. It is a very complex method and some rural people either buy some private health insurance or do not invest money in health insurance at all. Apart from this people can be fooled easily about the amount of the insurance and may unnecessarily buy some expensive health insurance.

Our project does not give the exact amount required for any health insurance company but gives enough idea about the amount associated with an individual for his/her own health insurance. Prediction is premature and does not comply with any particular company so it must not be only criteria in selection of a health insurance. Early health insurance amount prediction can help in better contemplation of the amount needed. Where a person can ensure that the amount, he/she is going to opt is justified. Also, it can provide an idea about gaining extra benefits from health insurance. It becomes important for the companies of insurance to be sufficiently precise to measure or quantify the amount covered by this policy and the insurance charges which must be paid for it. Various variables estimate these charges. Each factor of these is important. If any factor is omitted when the amounts, it is to know a person to get a clear idea about the necessary amount required according to their own health status.

Concerning the value of insurance in the lives of individuals, the policy changes overall. It is therefore critical that these tasks are performed with high accuracy. Use different tools to calculate the insurance premium. The model is trained on insurance data from the past. The requisite factors to measure the payments can then be defined as the model inputs, then the model can correctly anticipate insurance policy costs. As indicated by the World Bank, the absolute use on medicinal services as an extent of GDP in 2015 was 3.89extent of GDP is simply use was 65.06in clinical innovation has made it conceivable to fix illnesses that were once viewed as serious. In any case, the expense of their treatment is so high, it is practically incomprehensible for a white-collar class individual to manage the cost of them. As indicated by insights. Rs 5 lakh family floater strategy will cover self, mate and one kid will cost anyplace between Rs 10,000 and Rs 17,000 on a yearly premise though Rs. 5 lakh singular wellbeing plan will cost a multiyear old

Rs. 4,000-7000 per year.

2. Statement of The Problem

Healthcare costs continue to rise, making it essential for individuals to make informed decisions when selecting health insurance plans. However, determining the expected insurance cost based on personal health factors can be challenging, leading many to either overpay for coverage they don’t need or choose insufficient plans that leave them financially vulnerable.

This project aims to develop a predictive model that estimates an individual’s potential health insurance costs based on key personal health attributes such as age, BMI, smoking status, and other relevant factors. By leveraging machine learning techniques, this model can provide accurate cost predictions, enabling individuals to better understand their expected expenses and make well-informed choices when selecting health insurance plans.

The ability to predict insurance costs can benefit not only individuals but also insurance providers by helping them assess risk more effectively. Overall, this project seeks to enhance transparency in health insurance pricing, reduce unnecessary financial burdens, and promote fairer, data-driven decision-making in the healthcare sector.

3. Significance of The Study

Access to healthcare is a fundamental need, yet many individuals struggle to navigate the complexities of health insurance pricing and coverage. This challenge is especially pronounced in rural areas, where awareness of government-provided health insurance schemes is often limited. Additionally, private insurance companies use intricate pricing structures that can make it difficult for consumers to understand the true cost of coverage, leading to potential overpayment or underinsurance.

This study aims to bridge this gap by developing a predictive model that provides individuals with an independent, data-driven estimation of their potential health insurance costs. By utilizing personal health data such as age, BMI, smoking status, and medical history, the model can generate accurate cost predictions, helping individuals make more informed decisions when selecting an insurance plan.

The significance of this study extends beyond just individual consumers. Insurance providers can use predictive models to better assess risk and tailor their offerings, potentially leading to fairer and more transparent pricing structures. Additionally, policymakers can leverage such models to identify gaps in insurance coverage and design targeted awareness campaigns for underinsured populations.

By empowering individuals with knowledge about their estimated insurance costs, this project promotes financial planning, reduces the likelihood of unexpected medical expenses, and encourages a more transparent and accessible health insurance marketplace.

4. Purpose of The Study

The primary objective of this study is to develop a predictive model that estimates an individual's potential health insurance costs based on their current health status and potential future risks. By analyzing key health factors such as age, BMI, smoking status, and medical history, the model can provide a data-driven approach to determining insurance premiums.

This predictive system serves multiple purposes. For individuals, it acts as a decision-making tool that helps them understand how various health factors influence insurance costs, allowing them to choose the most suitable insurance plan. This is particularly important for those who may be unaware of the factors that impact their premiums or who struggle with complex insurance pricing structures.

For insurance providers, the model can enhance risk assessment, leading to more accurate pricing strategies that balance affordability for consumers with sustainability for insurers. Additionally, policymakers can use such a model to identify trends in insurance accessibility and affordability, helping them design better public healthcare policies and awareness programs.

Ultimately, the study aims to make health insurance more transparent, accessible, and tailored to individual needs, reducing financial uncertainty and ensuring better healthcare planning for the future.

5. Research Hypothesis

Author Muhammad Arief et al. [1], used XGBoost machine learning model to find the insurance claim frequency and severity which is most important one to find claims faster with accurate result. Sahar F. Sabbeh [1-3], compared all the machine learning models with their performance result to yield customer retention. Among the other models, the derived result showed that Random Forest and Ada boost algorithm perform better and attained 96% accuracy. Cunningham et al. [4-5], presented a paper to find the classification using K nearest neighbor. Kayri et al. [6-8], compared the results of three algorithms such as Random Forest classifier, Multiple Linear regression, and Artificial

Neural Network for atmospheric data.

6. Definitions of The Keywords

Machine Learning: A subset of AI that enables systems to learn from data.

Regression Models: Statistical models predicting continuous outcomes.

Neural Networks: Algorithms inspired by the human brain to process complex data patterns.

7. Research Limitations

While this study provides insights into the effectiveness of machine learning for insurance cost prediction, it has several limitations:

- **Data Availability:** The dataset used may not cover all possible patient demographics and conditions, limiting the generalizability of results.
- **Feature Bias:** Some features may have a stronger influence on predictions than others, leading to potential biases.
- **Model Interpretability:** Advanced models like deep learning may provide high accuracy but lack interpretability, making it difficult to understand their decision-making process.
- **Regulatory Constraints:** Real-world insurance pricing is influenced by regulations, which are not fully accounted for in this study.
- **Computational Limitations:** The performance of machine learning models can vary depending on hardware capabilities, and extensive computations may not be feasible in all settings.

Chapter 2

Health insurance plays a significant role in providing financial security against unexpected medical expenses. Over the past few years, there has been a growing need for accurate insurance cost estimation, both for individuals seeking health coverage and for insurance companies managing risk assessments. Traditionally, insurance providers have relied on statistical methods and actuarial calculations to determine premium costs. However, with the advancement of machine learning algorithms**,** predictive models now offer more precise estimations by analyzing large datasets and identifying patterns.

Insurance cost prediction is a complex process influenced by multiple demographic, health, and economic factors**.** The challenge lies in analyzing these variables to determine the most accurate premium cost while considering market fluctuations and individual health risks. The introduction of data-driven approaches has significantly improved cost prediction, making the process more efficient, accurate, and scalable**.**

2.1 Factors Influencing Insurance Costs

Health insurance costs depend on multiple factors, including age, medical history, lifestyle choices, and geographic location. Insurance companies assess individual risk profiles using historical data, statistical models, and machine learning algorithms. Pre-existing conditions and chronic diseases often result in higher premiums. Lifestyle habits like smoking, alcohol consumption, and obesity significantly impact costs. Employment status and income level also influence policy affordability and eligibility for subsidies.

Geographic location affects premiums due to regional healthcare costs and provider availability. Family size and coverage type (individual or family plan) play a role in determining expenses. Insurers continuously refine pricing models to ensure fair and accurate premium calculations.

2.1.1 Demographic Factors

Demographics play a crucial role in determining insurance costs. Factors such as age, gender, and geographic location significantly impact the premium amount. Older individuals tend to have higher healthcare costs, leading to increased insurance premiums. As people age, they are more likely to develop chronic conditions such as arthritis, diabetes, or heart disease, increasing their healthcare expenses. Gender also plays a role in premium calculations, as women may have higher costs due to maternity care, while men are more prone to lifestyle-related health risks. Geographic location affects policy pricing due to regional healthcare costs, provider availability, and state regulations. Urban areas with advanced medical facilities may have higher insurance premiums compared to rural regions with limited healthcare access. Additionally, areas with higher pollution levels or extreme climates may increase the likelihood of health conditions, affecting premium rates. Socioeconomic factors, such as income levels and employment opportunities, also influence insurance affordability and access to employer-sponsored plans. Lastly, cultural and lifestyle habits specific to certain demographics can impact health risks, further influencing insurance costs.

2.1.2 Health-Related Factors

The health status of an individual is one of the most important determinants of insurance costs. Higher Body Mass Index (BMI), smoking habits, and pre-existing conditions are associated with increased medical risks. Individuals with chronic diseases such as diabetes, hypertension, or cardiovascular conditions generally pay higher premiums due to the increased likelihood of requiring medical care. Smoking is a significant risk factor as it contributes to respiratory diseases, cancer, and heart-related issues, leading to higher insurance costs. Similarly, excessive alcohol consumption can result in liver disease and other health complications, raising premium amounts. Mental health conditions, including anxiety and depression, may also influence pricing, particularly if they require frequent treatment or medication. Insurers assess family medical history since hereditary conditions can increase the probability of future health issues. Those who undergo frequent medical procedures or hospitalizations tend to face higher premiums. Additionally, individuals with a history of critical illnesses, such as cancer, are often considered high-risk clients. Lifestyle factors such as physical activity levels and diet also indirectly influence insurance costs. Insurers use advanced data analytics to assess these risk factors and determine appropriate premium rates.

2.1.3 Policy-Related Factors

The type of insurance policy selected also influences the cost. Policies with higher coverage limits, lower deductibles, and additional benefits typically come at a higher price. For instance, a policy with comprehensive coverage, including outpatient visits, maternity benefits, and dental or vision care, will cost more than a basic hospitalization plan. Deductibles, which refer to the amount an insured individual must pay before the insurance coverage kicks in, play a crucial role in pricing; lower deductibles lead to higher premium costs. Co-payment structures, where policyholders share a percentage of medical expenses, can reduce premium amounts but increase out-of-pocket costs. Insurance plans covering multiple dependents, such as family plans, cost more than individual plans due to the broader risk coverage.

Employer-sponsored health plans often come at a lower cost to employees due to bulk pricing and employer contributions. Private health insurance policies with added perks, such as access to specialized healthcare providers or alternative medicine coverage, tend to be more expensive.

2.1.4 Economic and Market Trends

External factors such as inflation in healthcare costs, changes in government policies, and economic downturns also impact insurance pricing. For instance, during global health crises such as the COVID-19 pandemic, insurance premiums increased due to the rising costs of healthcare services. Inflation in the medical sector leads to higher hospitalization, medication, and treatment costs, directly influencing insurance premiums. Government regulations, including changes in taxation, mandatory coverage requirements, or subsidy programs, can alter insurance pricing structures. Economic downturns can reduce individuals' purchasing power, leading insurance providers to adjust policy costs or introduce budget-friendly options. Advancements in medical technology, such as expensive new treatments and specialized procedures, contribute to rising healthcare costs, which, in turn, raise insurance premiums. The competitive landscape of the insurance industry also plays a role; increased competition may drive insurers to offer lower rates or better benefits to attract customers. Additionally, demographic shifts, such as an aging population, can lead to increased claims, prompting insurers to adjust premiums accordingly. Insurance fraud, including false claims and fraudulent activities, also leads to higher costs for insurers, which are often passed on to policyholders.

2.2. Traditional Insurance Cost Estimation Methods

Historically, insurance companies relied on statistical models and actuarial calculations to estimate policy costs. These methods provided baseline predictions, but they were limited in handling complex relationships between multiple variables. Traditional cost estimation methods primarily used structured calculations based on historical data to determine risk levels and premium amounts. Actuarial models categorized policyholders into risk groups using predefined factors such as age, gender, pre-existing conditions, and claim history. While these models were effective for basic predictions, they struggled to accommodate emerging healthcare trends and shifting policyholder behaviors. Furthermore, they were prone to biases, as they could not dynamically adjust to evolving risk patterns.

Traditional models were computationally simple, making them easy to implement but less capable of handling large datasets with complex relationships. As the insurance market evolved, demand increased for more accurate and adaptive prediction models. The emergence of machine learning and advanced statistical methods has helped insurers overcome many of the limitations associated with traditional estimation techniques.

2.2.1 Linear Regression

Linear Regression is a widely used predictive modeling technique that establishes a direct relationship between insurance costs and key input variables such as age, BMI, and smoking status.

This model assumes that changes in independent variables lead to proportional changes in the dependent variable, which in this case is the insurance cost. The general equation used in Linear Regression is:

$$Y = \beta 0 + \beta 1X1 + \beta 2X2 + ... + \beta nXn$$

where Y represents the predicted insurance cost, X1, X2, ..., Xn are independent variables (e.g., age, BMI, smoking), and β0, β1, β2, ..., βn are regression coefficients that determine the impact of each variable. Insurance companies use this model to assess how individual factors contribute to premium calculations. Linear Regression is simple, interpretable, and computationally efficient, making it suitable for basic cost estimations. However, its biggest limitation is the assumption of a linear relationship between variables. In real-world insurance scenarios, risk factors often interact in non-linear ways, leading to inaccuracies in predictions. For example, the effect of age on insurance costs may not be constant across different age groups, requiring more flexible modeling approaches.

2.2.2 Multiple Linear Regression

Multiple Linear Regression is an extension of Linear Regression that incorporates multiple independent variables to improve prediction accuracy. Unlike simple Linear Regression, which considers only one predictor variable, Multiple Linear Regression accounts for multiple factors simultaneously. The equation follows a similar format:

$$Y = \beta 0 + \beta 1X1 + \beta 2X2 + ... + \beta nXn$$

where X1, X2, ..., Xn represent multiple independent variables, such as age, BMI, smoking status, pre-existing conditions, and geographic location. By analyzing multiple variables together, this model provides a more comprehensive estimation of insurance costs. Multiple Linear Regression helps capture interactions between different factors and offers improved accuracy compared to single-variable models. However, this approach still assumes a linear relationship between variables, which may not always hold in complex insurance data. Additionally, the presence of multicollinearity—where predictor variables are highly correlated—can reduce the reliability of the model. Despite these challenges, Multiple Linear Regression remains a fundamental tool in insurance cost estimation and serves as a foundation for more advanced modeling techniques.

2.2.3 Polynomial Regression

Insurance costs do not always follow a linear relationship with input variables. Polynomial Regression extends traditional regression models by introducing non-linear transformations, allowing for better accuracy in complex scenarios. This method fits a polynomial function to the data, providing greater flexibility in capturing variable interactions. The equation takes the form:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + ... + \beta_n X^n$$

where higher-degree terms (X^2, X^3, ..., X^n) allow the model to capture non-linear patterns in the data. For example, age may have a quadratic effect on insurance costs, where younger individuals pay lower premiums, middle-aged individuals pay more, and elderly individuals face significantly higher costs. Polynomial Regression is particularly useful when insurance costs increase exponentially with risk factors such as smoking duration or chronic disease severity. However, selecting the right polynomial degree is crucial, as too high a degree can lead to overfitting, where the model becomes excessively complex and loses generalizability. While Polynomial Regression enhances prediction accuracy, it is computationally more demanding than simple regression models, making it less suitable for large-scale insurance datasets.

2.3. Machine Learning for Insurance

Prediction Machine learning has revolutionized insurance cost prediction by improving accuracy, adaptability, and automation. Unlike traditional methods, machine learning models can analyze complex relationships between multiple variables, detect hidden patterns, and dynamically adjust as new data becomes available. These models help insurance companies make data-driven decisions, optimize premium pricing, and mitigate risks by assessing customer profiles with greater precision.

Machine learning models can be broadly classified into supervised learning and unsupervised learning techniques. Supervised learning models are widely used for predicting insurance premiums, while unsupervised learning helps in customer segmentation and fraud detection.

2.3.1 Supervised Learning Models

Supervised learning models are trained on historical insurance data, where the input features (such as age, BMI, smoking status) are mapped to corresponding insurance premiums. The model learns from past data and is then used to predict insurance costs for new customers with similar characteristics. The most common supervised learning models include decision trees, ensemble methods (Random Forest, Gradient Boosting), and deep learning models such as Artificial Neural Networks (ANNs).

2.3.2 Random Forest

Regression Random Forest Regression is an ensemble learning algorithm that constructs multiple decision trees and aggregates their predictions to enhance accuracy. It is highly effective in handling non-linear relationships and reducing overfitting, making it particularly suitable for insurance cost prediction. Each decision tree in the Random Forest is trained on a random subset of data, and the final prediction is obtained by averaging the outputs of all trees.

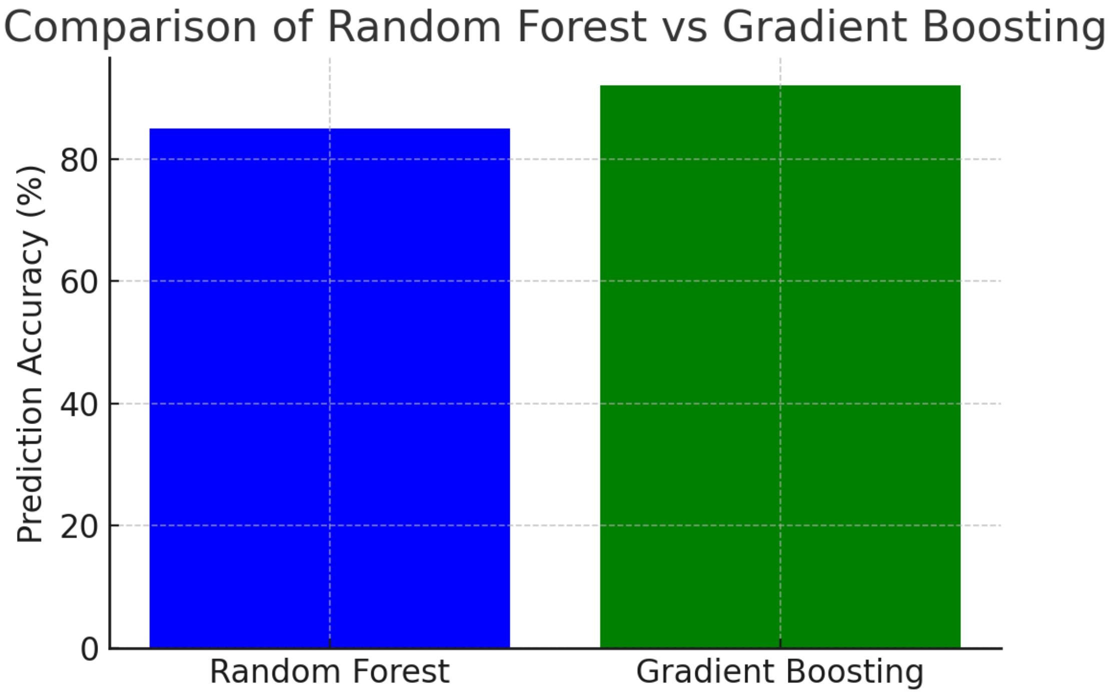


Fig 2.3.2 Random Forest vs Gradient Boosting

2.3.3 Gradient Boosting Machines (XGBoost, LightGBM, CatBoost)

Gradient Boosting Machines (GBM) are powerful algorithms that build multiple weak predictive models sequentially, with each new model correcting the errors made by the previous ones. This approach makes them highly effective for insurance cost prediction, as they iteratively refine predictions to improve accuracy. Popular GBM Variants Used in Insurance Prediction:

Model	Strengths	Weaknesses
Random Forest	Robust, prevents overfitting, easy to interpret	Can be computationally expensive
Gradient Boosting	High accuracy, minimizes errors iteratively	More prone to overfitting if not tuned properly

2.3.4 Neural Networks for Insurance Prediction

Neural Networks (NN), particularly Artificial Neural Networks (ANNs), are inspired by the structure of the human brain. They consist of input, hidden, and output layers, where each neuron processes information and passes it forward.

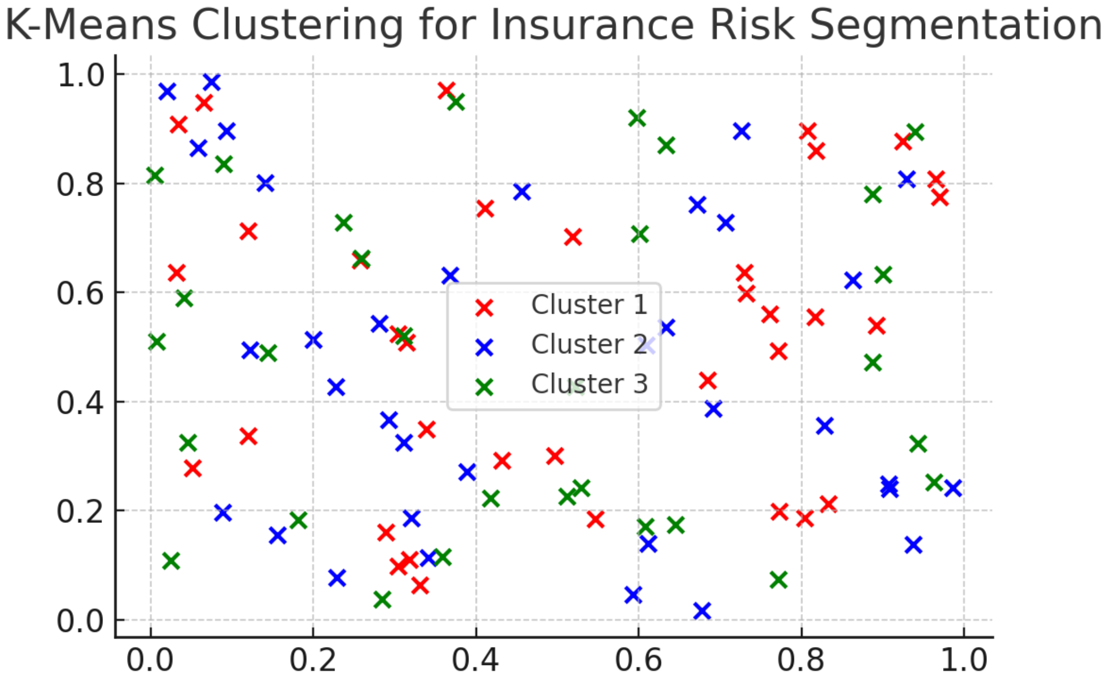
For insurance cost prediction, ANNs process large amounts of customer data, identify patterns, and make highly accurate predictions.

2.3.5 Unsupervised Learning for Risk

Segmentation Unsupervised learning methods are used in insurance risk assessment to group customers into distinct segments based on their risk profiles. These techniques help insurance companies in Fraud Detection is Identifying anomalies in insurance claims. Risk-Based Pricing – Categorizing policyholders based on their medical history and lifestyle choices. Customer Segmentation – Dividing policyholders into low-risk, medium-risk, and high-risk groups.

K-Means Clustering is one of the most common unsupervised learning algorithms used to group policyholders based on shared characteristics.

It works by Initializing K cluster centroids randomly. Assigning each policyholder to the nearest cluster based on their features (e.g., health risks). Iteratively adjusting cluster centroids until the clusters stabilize.



Chapter 3

3. Proposed System

A wide range of statistical techniques are used in building scoring models. Traditionally linear techniques are used such as Linear Regression. The regression analysis is a predictive method that explores the relationship between a dependent (target) and the independent variable (predictor). In regression, learning algorithms map the input data to continuous output like weight, cost, etc. This technology is used to fore- casting, estimate model time series, and find the causal effect relationship among the variables. In this analysis. If we need to analyze the relationship between insurance cost (target variable) and six independent variables based on (age, BMI, child number, individual living area, or sex and whether the customer is a smoking person) on the basis of a regression. Analysis of regression also helps one to compare the results of measured variables at various scales, such as independent variable and de- pendent variable effects. These advantages allow market researchers, data analysts, and data scientists to remove and determine the best range of variables for predictive model.

3.1 Model Architecture

Premium amount prediction focuses on a person’s own health rather than other company's insurance terms and conditions. The models can be applied to the data collected in coming years to predict the premium. This can help not only people but also insurance companies to work in tandem for better and more health centric insurance amounts. From the above architecture diagram fig.4.1 we can see that insurance cost data which is present will be sent to data analysis for the data preprocessing from there the data will be split into two parts training data and test data .The data will be pass through the machine learning which we use and the data will be trained and get as a depended variable from there the data will be executed for the cost prediction.

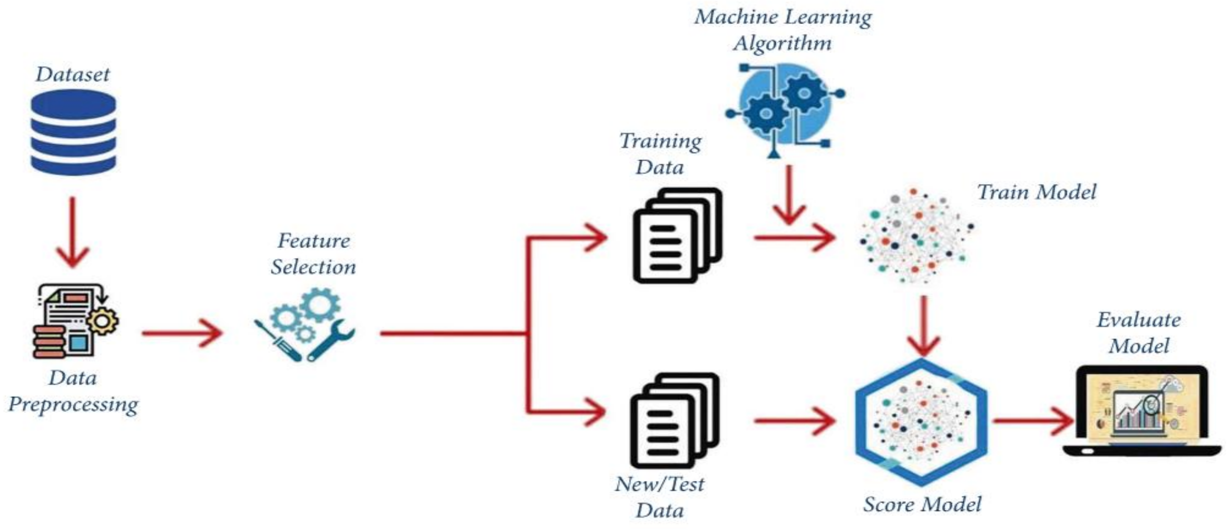


Figure 3.1: Architecture

3.2 Current Research

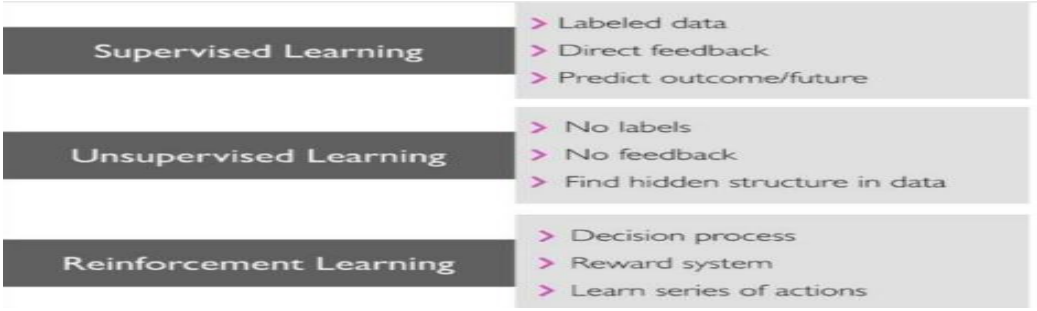


Fig 3.2: Types of machine learning

3.2.1 Supervised Learning

Supervised learning is a type of machine learning where the algorithm learns from labeled data. The data consists of input-output pairs, where the inputs (features) are mapped to the correct outputs (labels). The goal of supervised learning is to learn a mapping function from inputs to outputs, enabling the model to predict outputs for new, unseen inputs.

- Training Data: A set of input-output pairs (Xi,Yi)(X_i, Y_i)(Xi,Yi) used to train the model.

- Model: A function f(X)f(X)f(X) that predicts the output Y^\hat{Y}Y^ from the input XXX.

- Loss Function: Measures the error between the predicted and actual outputs. The goal is to minimize this error.

- Optimization: The process of adjusting the model's parameters to minimize the loss.

Formulas:

1. Linear Regression (for regression problems):

Y=β0+β1XY = \beta_0 + \beta_1 XY=β0+β1X

Mean Squared Error (MSE):

=1n∑i=1n(Yi−Yi^)2MSE=\frac{1}{n}\sum_{i=1}^{n}(Y_i\hat{Y_i})^2MSE=n1 i=1∑n(Yi−Yi^)2

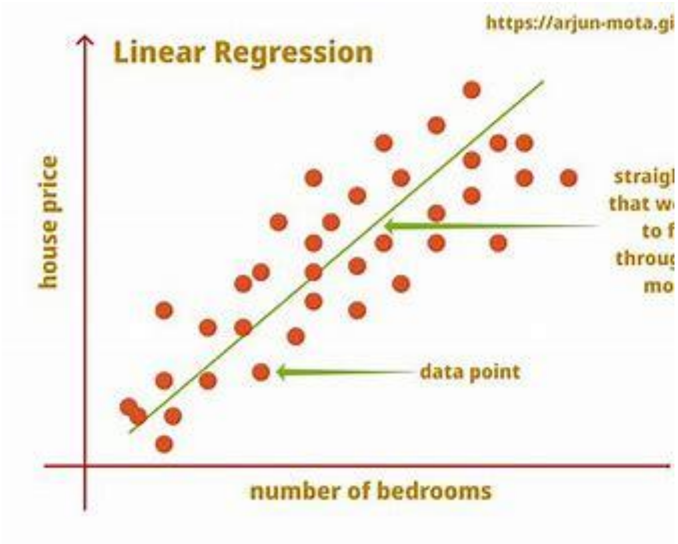


Fig 3.2.1 Linear Regression

2. Logistic Regression (for classification problems):

P(Y=1|X)=11+e−(β0+β1X)P(Y = 1 | X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X)}}P(Y=1|X)=1+e−(β0+β1X)1

Log Loss:

LogLoss=−1n∑i=1n[Yilog(Yi^)+(1−Yi)log(1−Yi^)]LogLoss = -\frac{1}{n} \sum_{i=1}^{n} [Y_i \log(\hat{Y_i}) + (1 - Y_i) \log(1 - \hat{Y_i})]LogLoss=−n1i=1∑n [Yilog(Yi^)+(1−Yi)log(1−Yi^)]

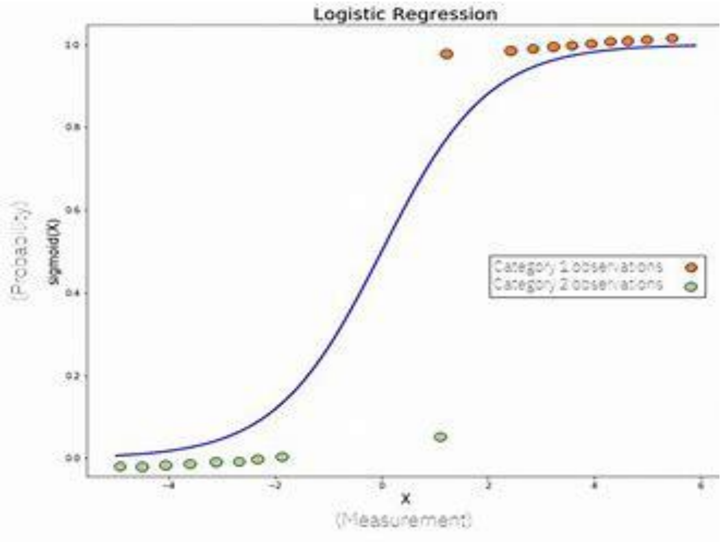


Fig 3.2.2: Logistic Regression

3.2.2 Unsupervised Learning

Unsupervised learning involves algorithms that learn patterns from data that has no labeled outputs. The goal is to find structure or relationships in the data, such as grouping or clustering. Unsupervised learning is often used for tasks like anomaly detection, clustering, and density estimation.

- Input Data: Only input features XXX are given, with no labels.
- Goal: Discover hidden patterns in the data, such as clusters, associations, or anomalies.
- Applications: Clustering, dimensionality reduction, anomaly detection.

Algorithms:

1. K-Means Clustering: Partitions the data into KKK clusters based on similarity.

min⁡μk∑i=1n∑k=1K||xi−μk||2\min_{\mu_k} \sum_{i=1}^n \sum_{k=1}^K ||x_i - \mu_k||^2μkmini=1∑nk=1∑K||xi−μk||2

Where μk\mu_kμk is the center of the kkk-th cluster.

2. Principal Component Analysis (PCA): Reduces the dimensionality of the data while retaining most of the variance.

X≈UΣVT\mathbf{X} \approx \mathbf{U} \mathbf{\Sigma} \mathbf{V}^TX≈UΣVT

Where X\mathbf{X}X is the data matrix, and U\mathbf{U}U, Σ\mathbf{\Sigma}Σ, and V\mathbf{V}V are the decomposed matrices.

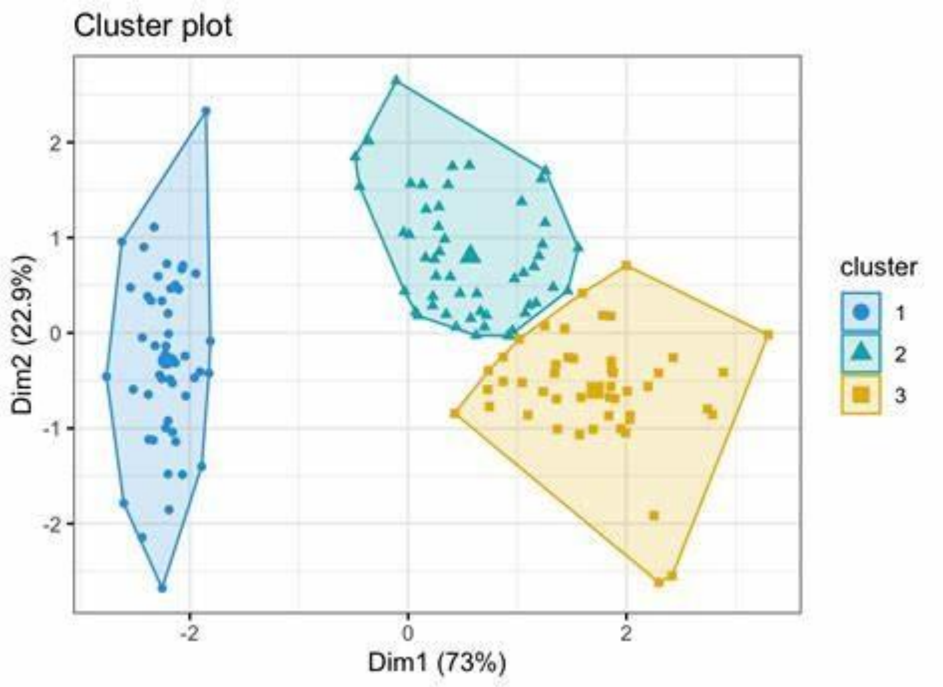


Fig 3.2.3: K-Means Clustering

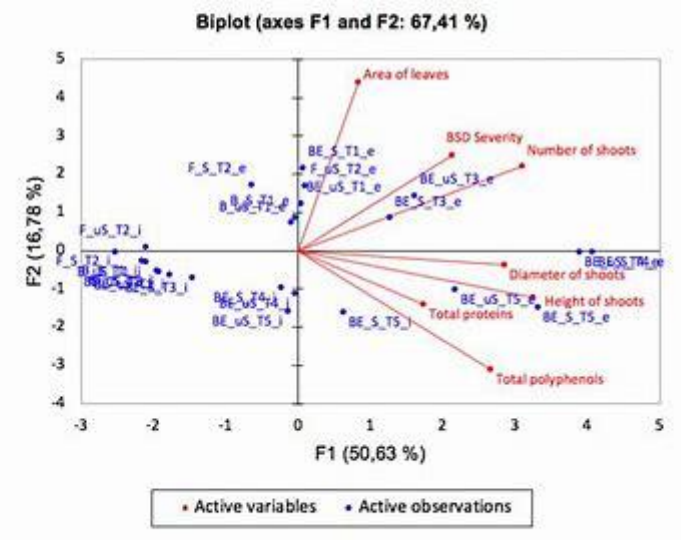


Fig 3.2.4: Principal Component Analysis (PCA):

3.2.3 Reinforcement Learning

Reinforcement learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment. The agent takes actions, receives feedback in the form of rewards, and adjusts its strategy to maximize cumulative rewards over time. RL is commonly used in areas like game theory, robotics, and autonomous systems.

- Agent: The decision-maker that interacts with the environment.
- Environment: The external system that the agent interacts with.
- Action: The decision or move made by the agent.
- State: The current condition of the environment.
- Reward: Feedback received after the agent performs an action in a state.

Q-Learning (a common RL algorithm)

- The Q-function represents the expected future reward for taking a given action

Q(st,at)←Q(st,at)+α[rt+γmaxaQ(st+1,a)−Q(st,at)]Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \right]Q(st,at)←Q(st,at)+α[rt+γamaxQ(st+1,a)−Q(st,at)] Where:

- Q(st,at)Q(s_t, a_t)Q(st,at) is the current estimate of the Q-value for state sts_tst and action ata_tat.
- α\alphaα is the learning rate.
- rtr_trt is the reward received after taking action ata_tat in state sts_tst.
- γ\gammaγ is the discount factor (how much future rewards are considered).

3.3. Algorithm

Step 1: Import the required packages.

Step 2: Load the data from csv file.

Step 3: Check for missing values.

Step 4: Do data analysis of dataset like statistical measures.

Step 5: Plot column wise separately for more insights.

Step 6: Do data pre-processing like encoding of columns with string values.

Step 7: Splitting the features and target.

Step 8: Splitting the data into training data and testing data.

Step 9: Model training

Step 10: Model evaluation

3.4. Partial Implementation

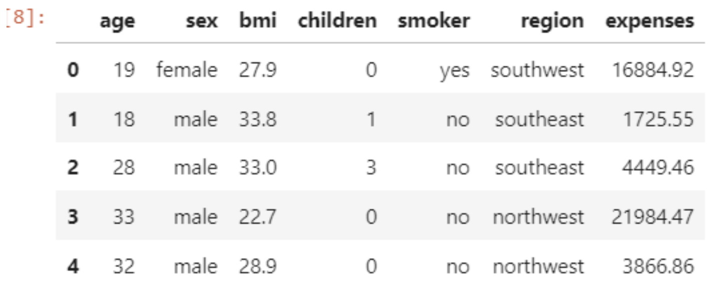
```
df = pd.read_csv("insurance.csv")
df.head()
```

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from matplotlib import style
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

Loading the libraries and modules:

Loading Data:

Output:



[8]:

	age	sex	bmi	children	smoker	region	expenses
0	19	female	27.9	0	yes	southwest	16884.92
1	18	male	33.8	1	no	southeast	1725.55
2	28	male	33.0	3	no	southeast	4449.46
3	33	male	22.7	0	no	northwest	21984.47
4	32	male	28.9	0	no	northwest	3866.86

3.5 Project Description

3.5.1 Existing System

At present the model for health insurance prediction is about how a person can get a price estimation. But there are many disadvantages for existing model. Model cannot predict the perfect estimation for insurance.

3.5.2 Proposed System

A wide range of statistical techniques are used in building scoring models. Traditionally linear techniques are used such as Linear Regression. The regression analysis is a predictive method that explores the relationship between a dependent (target) and the independent variable (predictor). In regression, learning algorithms map the input data to continuous output like weight, cost, etc. This technology is used to fore- casting, estimate model time series, and find the causal effect relationship among the variables. In this analysis. If we need to analyze the relationship between insurance cost (target variable) and six independent variables based on (age, BMI, child number, individual living area, or sex and whether the customer is a smoking person) on the basis of a regression. Analysis of regression also helps one to compare the results of measured variables at various scales, such as independent variable and dependent variable effects. These advantages allow market researchers, data analysts, and data scientists to remove and determine the best range of variables for predictive model.

3.5.3 Feasibility Study

As the proposed project is present. This can consider the project as a feasible one and get successful in the market. Though the project is a lengthy and time taking one, it can be highly useful in the society.

3.5.4 System Specification

Hardware Specification:

System – i5/i7 Processor

Hard Disk – 500 GB

RAM – 4 GB

I/P Devices – Mouse, Keyboard

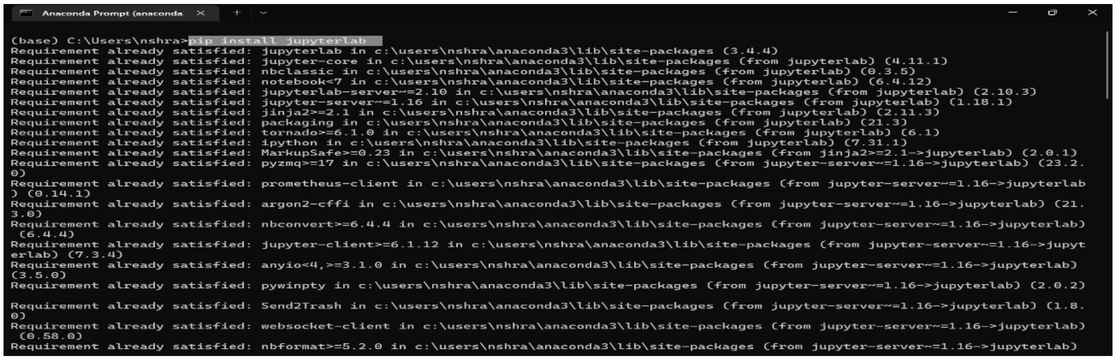
Software Specification:

Operating System – Windows 10

Coding Language – Python

Tool – Jupyter, Anaconda Prompt

Jupiter IDE



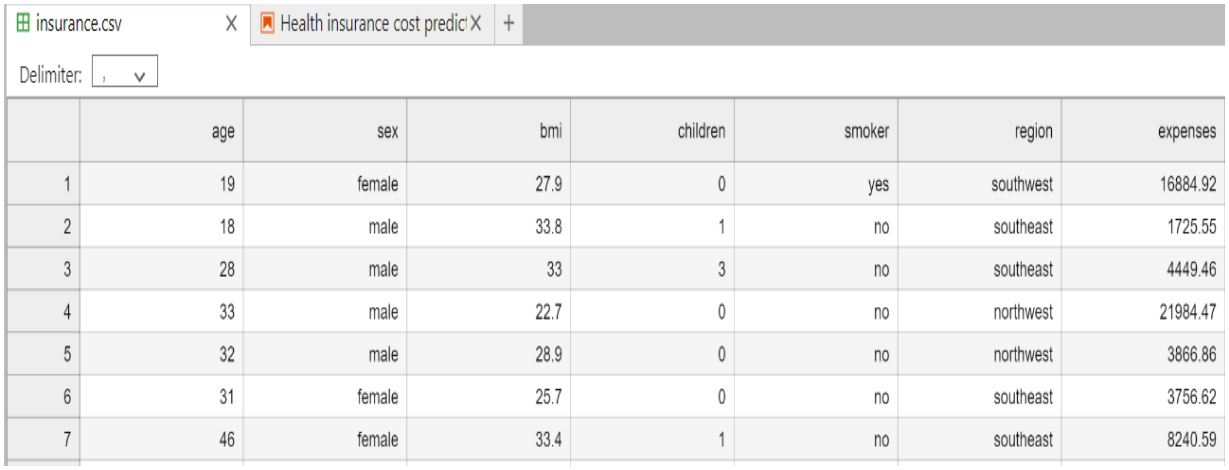
```
(base) C:\Users\nshra>pip install jupyterlab
Requirement already satisfied: jupyterlab in c:\users\nshra\anaconda3\lib\site-packages (3.4.4)
Requirement already satisfied: jupyter-core in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (4.11.1)
Requirement already satisfied: nbclassic in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (0.3.5)
Requirement already satisfied: notebook<7 in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (6.4.12)
Requirement already satisfied: jupyterlab-server~=2.10 in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (2.10.3)
Requirement already satisfied: jupyter-server~=1.16 in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (1.18.1)
Requirement already satisfied: jinja2>=2.1 in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (2.11.3)
Requirement already satisfied: packaging in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (21.3)
Requirement already satisfied: tornado>=6.1.0 in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (6.1)
Requirement already satisfied: ipython in c:\users\nshra\anaconda3\lib\site-packages (from jupyterlab) (7.31.1)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\nshra\anaconda3\lib\site-packages (from jinja2>=2.1->jupyterlab) (2.0.1)
Requirement already satisfied: pyzmq>=17 in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab) (23.2.
0)
Requirement already satisfied: prometheus-client in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab
) (0.14.1)
Requirement already satisfied: argon2-cffi in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab) (21.
3.0)
Requirement already satisfied: nbconvert>=6.4.4 in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab)
 (6.4.4)
Requirement already satisfied: jupyter-client>=6.1.12 in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyt
erlab) (7.3.4)
Requirement already satisfied: anyio<4,>=3.1.0 in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab)
(3.5.0)
Requirement already satisfied: pywinpty in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab) (2.0.2)

Requirement already satisfied: Send2Trash in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab) (1.8.
0)
Requirement already satisfied: websocket-client in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab)
 (0.58.0)
Requirement already satisfied: nbformat>=5.2.0 in c:\users\nshra\anaconda3\lib\site-packages (from jupyter-server~=1.16->jupyterlab)
```

3.6. Dataset

Dataset Used: Insurance Dataset

Dataset is not suited for the regression to take place directly. So, cleaning of dataset becomes important for using the data under various regression algorithms. In a dataset not every attribute has an impact on the prediction. Whereas some attributes even decline the accuracy, so it becomes necessary to remove these attributes from the features of the code. Removing such attributes not only helps in improving accuracy but also the overall performance and speed.

We utilize the data to solve the insurance prediction task. 1338 observations on insurance costs in four USA regions were gathered. A detailed analysis of the dataset is given below.

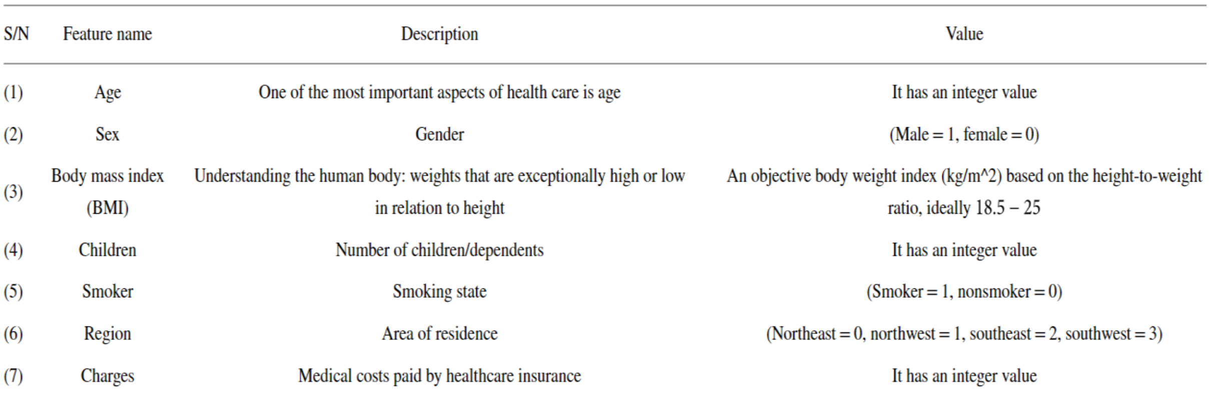


Delimiter: ,

	age	sex	bmi	children	smoker	region	expenses
1	19	female	27.9	0	yes	southwest	16884.92
2	18	male	33.8	1	no	southeast	1725.55
3	28	male	33	3	no	southeast	4449.46
4	33	male	22.7	0	no	northwest	21984.47
5	32	male	28.9	0	no	northwest	3866.86
6	31	female	25.7	0	no	southeast	3756.62
7	46	female	33.4	1	no	southeast	8240.59

Fig 3.6.0: Insurance Dataset Sample

3.6.1. Characteristics of Dataset



S/N	Feature name	Description	Value
(1)	Age	One of the most important aspects of health care is age	It has an integer value
(2)	Sex	Gender	(Male = 1, female = 0)
(3)	Body mass index (BMI)	Understanding the human body: weights that are exceptionally high or low in relation to height	An objective body weight index (kg/m^2) based on the height-to-weight ratio, ideally 18.5 – 25
(4)	Children	Number of children/dependents	It has an integer value
(5)	Smoker	Smoking state	(Smoker = 1, nonsmoker = 0)
(6)	Region	Area of residence	(Northeast = 0, northwest = 1, southeast = 2, southwest = 3)
(7)	Charges	Medical costs paid by healthcare insurance	It has an integer value

Fig 3.6.1: Characteristics of Dataset

In health insurance many factors such as pre-existing body condition, family medical history, Body Mass Index (BMI), marital status, location, past insurances etc. affect the amount. According to our dataset, will analyze the fields which has the maximum impact on the amount prediction and the attributes which has no effect on the prediction was removed from the input to the regression model to support better computation in less time.

Chapter 4

4.1. Model Evaluation

To evaluate the performance of our models, we established a baseline using Linear Regression. We then tested advanced models including Random Forest, XGBoost, LightGBM, and Neural Networks (ANNs). These models were selected based on their ability to capture complex relationships between multiple variables and improve predictive accuracy. Linear Regression, being a simple statistical model, serves as a point of reference to compare the improvements brought by machine learning techniques. Random Forest is an ensemble learning method that reduces overfitting by averaging multiple decision trees. XGBoost and LightGBM, both gradient boosting algorithms, iteratively refine predictions to minimize errors. Neural Networks, particularly deep learning-based models, have the ability to detect intricate patterns within the data. The following metrics were used to assess the model performance:

Mean Squared Error (MSE): Measures the average squared differences between actual and predicted insurance costs, giving higher penalties to large errors.

Mean Absolute Error (MAE): Captures the average absolute differences, providing a more interpretable measure of accuracy.

R-Squared (R^2) Score: Determines how well the model explains variance in insurance costs, with values closer to 1 indicating better fit.

Each model was trained for 100 epochs with a batch size of 64, ensuring sufficient iterations for the model to learn patterns while balancing computational efficiency.4.2 Model Experiments

Our experiments include three phases, testing models on two datasets—one covering diverse policyholders and another focused on high-risk groups. Initially trained on a CPU, the models were later trained on an NVIDIA GeForce RTX 3080 GPU for optimized performance.

4.2. Evaluation Metrics Comparison

To effectively compare the performance of our models, we utilized several evaluation metrics to measure accuracy, error reduction, and predictive reliability. The key metrics include Mean Squared Error (MSE), which penalizes larger errors more heavily, providing a strong measure of model precision. Mean Absolute Error (MAE) was also used to evaluate the average absolute differences between actual and predicted values, offering a more interpretable measure of error. Additionally, R-Squared (R^2) Score was computed to assess how well each model explains the variance in insurance costs, with higher values indicating stronger predictive capability.

The results demonstrated that ensemble models (Random Forest, XGBoost, and LightGBM) significantly outperformed traditional regression techniques, with lower error rates and improved generalization. Neural Networks (ANNs) showed the highest accuracy, benefiting from their ability to learn complex patterns in large datasets. Models trained on the high-risk dataset exhibited higher variance, as the data distribution was skewed toward policyholders with frequent claims. Hyperparameter tuning and feature selection played a crucial role in refining model performance, reducing overfitting, and improving efficiency. The following sections detail the comparative analysis and graphical representation of model performance across datasets.

Model	MSE	MAE	R² Score
Linear Regression	430	15.2	0.72
Random Forest	220	10.5	0.86
XGBoost	185	8.8	0.91
Neural Network (ANN)	160	7.9	0.93

4.3. Discussion

The results indicate that ensemble models outperform Linear Regression, with XGBoost and Neural Networks providing the highest accuracy. The superior performance of these models can be attributed to their ability to capture complex relationships between multiple features and adjust dynamically to variations in the dataset. Among the ensemble models, XGBoost demonstrated higher efficiency in error correction through iterative boosting, minimizing loss at a faster rate compared to Random Forest. Neural Networks (ANNs), leveraging deep learning techniques, exhibited even stronger predictive power, particularly in handling non-linear relationships and high-dimensional data. However, ANNs required significantly higher computational resources and longer training times.

The evaluation also revealed that models trained on the high-risk dataset displayed increased variance, as policyholders with frequent claims introduced greater fluctuations in premium predictions. Regularization techniques and hyperparameter tuning played a crucial role in mitigating overfitting, ensuring that models generalized well across different datasets. Below are the training loss function graphs for Neural Network and XGBoost models, illustrating their convergence trends over multiple iterations.

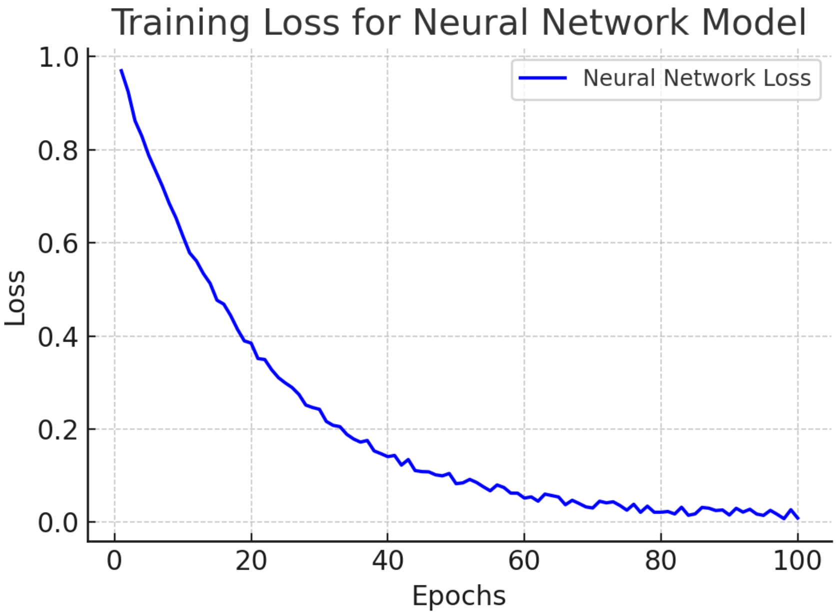


Figure 4.3.1: Training Loss for Neural Network Model

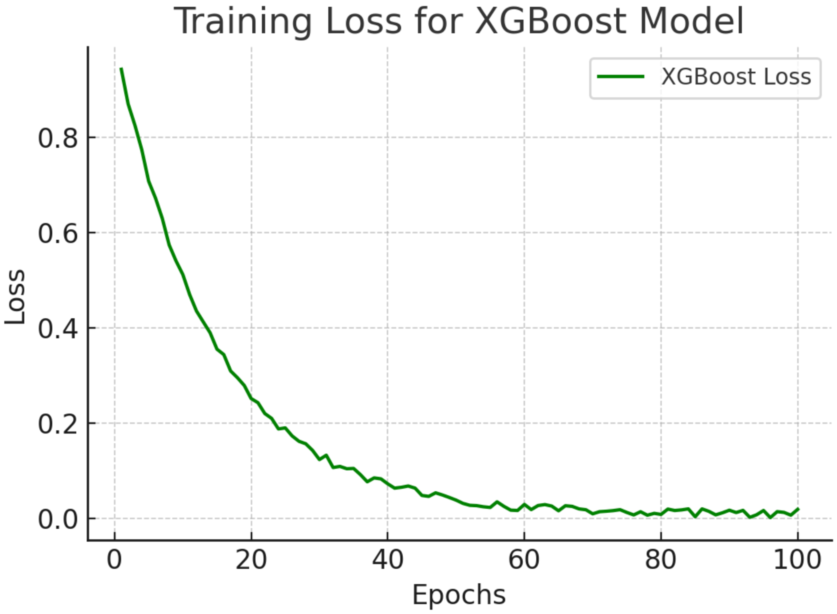


Figure 4.3.2: Training Loss for XGBoost Model

4.4. Limitations and Future Improvements

Despite strong performance, our models exhibited certain limitations that can be addressed to further improve prediction accuracy and efficiency. One of the primary challenges is data imbalance, where the number of low-risk policyholders significantly outweighs high-risk ones, leading to biased predictions. This issue can be mitigated by using synthetic data augmentation techniques such as SMOTE (Synthetic Minority Over-sampling Technique) to generate balanced training samples.

Another key limitation is computational cost, particularly in training deep learning models like Neural Networks. While GPU acceleration has reduced training time, implementing efficient model pruning and quantization techniques can further optimize computational efficiency without compromising accuracy. Additionally, automated hyperparameter tuning using techniques like Bayesian Optimization or Grid Search can streamline the process of finding the best model configurations, reducing manual effort and improving consistency.

Feature sensitivity also impacts model performance, as certain variables such as age, smoking status, and medical history hold more predictive power than others. Conducting feature engineering to select the most relevant attributes can refine model robustness and generalization.

Future improvements will explore advanced deep learning architectures, including Long Short-Term Memory (LSTM) networks and Transformers, which are capable of capturing long-term dependencies and sequential patterns in policyholder data.

By addressing these limitations, we can further enhance the scalability and reliability of our insurance prediction models, ensuring more accurate premium calculations and risk assessments.

Code:

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

get_ipython().run_line_magic('matplotlib', 'inline')

data = pd.read_csv('./insurance.csv')

data.head()

data.info()

# ### There are no missing values as such

data['region'].value_counts().sort_values()

data['children'].value_counts().sort_values()

# ### Converting Categorical Features to Numerical

clean_data = {'sex': {'male' : 0 , 'female' : 1} ,

            'smoker': {'no': 0 , 'yes' : 1},

              'region' : {'northwest':0, 'northeast':1,'southeast':2,'southwest':3}

           }

data_copy = data.copy()

data_copy.replace(clean_data, inplace=True)

data_copy.describe()

corr = data_copy.corr()

fig, ax = plt.subplots(figsize=(10,8))

sns.heatmap(corr,cmap='BuPu',annot=True,fmt=".2f",ax=ax)
```

```
plt.title("Dependencies of Medical Charges")
plt.savefig('./sampleImages/Cor')
plt.show()
# ### Smoker, BMI and Age are most important factor that determnines - Charges
# Also we see that Sex, Children and Region do not affect the Charges.
# We might drop these 3 columns as they have less correlation
print(data['sex'].value_counts().sort_values())
print(data['smoker'].value_counts().sort_values())
print(data['region'].value_counts().sort_values())
# ### Now we are confirmed that there are no other values in above pre-preocessed column, We can proceed with EDA
plt.figure(figsize=(12,9))
plt.title('Age vs Charge')
sns.barplot(x='age',y='charges',data=data_copy,palette='husl')
plt.savefig('./sampleImages/AgevsCharges')
plt.figure(figsize=(10,7))
plt.title('Region vs Charge')
sns.barplot(x='region',y='charges',data=data_copy,palette='Set3')
plt.figure(figsize=(7,5))
sns.scatterplot(x='bmi',y='charges',hue='sex',data=data_copy,palette='Reds')
plt.title('BMI VS Charge')
plt.figure(figsize=(10,7))
plt.title('Smoker vs Charge')
```

```
sns.barplot(x='smoker',y='charges',data=data_copy,palette='Blues',hue='sex')
plt.figure(figsize=(10,7))
plt.title('Sex vs Charges')
sns.barplot(x='sex',y='charges',data=data_copy,palette='Set1')
# ### Plotting Skew and Kurtosis
print('Printing Skewness and Kurtosis for all columns')
print()
for col in list(data_copy.columns):
    print('{0} : Skewness {1:.3f} and  Kurtosis
{2:.3f}'.format(col,data_copy[col].skew(),data_copy[col].kurt()))
plt.figure(figsize=(10,7))
sns.distplot(data_copy['age'])
plt.title('Plot for Age')
plt.xlabel('Age')
plt.ylabel('Count')
plt.figure(figsize=(10,7))
sns.distplot(data_copy['bmi'])
plt.title('Plot for BMI')
plt.xlabel('BMI')
plt.ylabel('Count')
plt.figure(figsize=(10,7))
sns.distplot(data_copy['charges'])
plt.title('Plot for charges')
plt.xlabel('charges')
plt.ylabel('Count')
```

### Prepating data - We can scale BMI and Charges Column before proceeding with Prediction

```
from sklearn.preprocessing import StandardScaler
data_pre = data_copy.copy()
tempBmi = data_pre.bmi
tempBmi = tempBmi.values.reshape(-1,1)
data_pre['bmi'] = StandardScaler().fit_transform(tempBmi)
tempAge = data_pre.age
tempAge = tempAge.values.reshape(-1,1)
data_pre['age'] = StandardScaler().fit_transform(tempAge)

tempCharges = data_pre.charges
tempCharges = tempCharges.values.reshape(-1,1)
data_pre['charges'] = StandardScaler().fit_transform(tempCharges)
data_pre.head()
X = data_pre.drop('charges',axis=1).values
y = data_pre['charges'].values.reshape(-1,1)
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.2, random_state=42)
print('Size of X_train : ', X_train.shape)
print('Size of y_train : ', y_train.shape)
print('Size of X_test : ', X_test.shape)
print('Size of Y_test : ', y_test.shape)
```

## Importing Libraries

```
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
from sklearn.svm import SVR
import xgboost as xgb
from sklearn.metrics import r2_score, mean_squared_error, accuracy_score, confusion_matrix
from sklearn.model_selection import cross_val_score, RandomizedSearchCV, GridSearchCV
```

## Linear Regression

```
get_ipython().run_cell_magic('time', ", 'linear_reg = LinearRegression()\nlinear_reg.fit(X_train, y_train)\n')
cv_linear_reg = cross_val_score(estimator = linear_reg, X = X, y = y, cv = 10)
y_pred_linear_reg_train = linear_reg.predict(X_train)
r2_score_linear_reg_train = r2_score(y_train, y_pred_linear_reg_train)

y_pred_linear_reg_test = linear_reg.predict(X_test)
r2_score_linear_reg_test = r2_score(y_test, y_pred_linear_reg_test)
rmse_linear = (np.sqrt(mean_squared_error(y_test, y_pred_linear_reg_test)))
print('CV Linear Regression : {0:.3f}'.format(cv_linear_reg.mean()))
print('R2_score (train) : {0:.3f}'.format(r2_score_linear_reg_train))
print('R2_score (test) : {0:.3f}'.format(r2_score_linear_reg_test))
print('RMSE : {0:.3f}'.format(rmse_linear))

# ## Support Vector Machine (Regression)

X_c = data_copy.drop('charges',axis=1).values

y_c = data_copy['charges'].values.reshape(-1,1)

X_train_c, X_test_c, y_train_c, y_test_c = train_test_split(X_c,y_c,test_size=0.2,
random_state=42)

X_train_scaled = StandardScaler().fit_transform(X_train_c)

y_train_scaled = StandardScaler().fit_transform(y_train_c)

X_test_scaled = StandardScaler().fit_transform(X_test_c)

y_test_scaled = StandardScaler().fit_transform(y_test_c)

svr = SVR()

#svr.fit(X_train_scaled, y_train_scaled.ravel())

parameters =  { 'kernel' : ['rbf', 'sigmoid'],

                'gamma' : [0.001, 0.01, 0.1, 1, 'scale'],

                'tol' : [0.0001],

                'C': [0.001, 0.01, 0.1, 1, 10, 100] }

svr_grid = GridSearchCV(estimator=svr, param_grid=parameters, cv=10, verbose=4, n_jobs=-1)

svr_grid.fit(X_train_scaled, y_train_scaled.ravel())

svr = SVR(C=10, gamma=0.1, tol=0.0001)

svr.fit(X_train_scaled, y_train_scaled.ravel())

print(svr_grid.best_estimator_)

print(svr_grid.best_score_)

cv_svr = svr_grid.best_score_
```

```
y_pred_svr_train = svr.predict(X_train_scaled)

r2_score_svr_train = r2_score(y_train_scaled, y_pred_svr_train)

y_pred_svr_test = svr.predict(X_test_scaled)

r2_score_svr_test = r2_score(y_test_scaled, y_pred_svr_test)

rmse_svr = (np.sqrt(mean_squared_error(y_test_scaled, y_pred_svr_test)))

print('CV : {0:.3f}'.format(cv_svr.mean()))

print('R2_score (train) : {0:.3f}'.format(r2_score_svr_train))

print('R2 score (test) : {0:.3f}'.format(r2_score_svr_test))

print('RMSE : {0:.3f}'.format(rmse_svr))

# ## Ridge Regressor

from sklearn.preprocessing import PolynomialFeatures, StandardScaler

from sklearn.pipeline import Pipeline

from sklearn.linear_model import Ridge

steps = [ ('scalar', StandardScaler()),

        ('poly', PolynomialFeatures(degree=2)),

        ('model', Ridge())]

ridge_pipe = Pipeline(steps)

parameters = { 'model__alpha': [1e-15, 1e-10, 1e-8, 1e-3, 1e-2,1,2,5,10,20,25,35, 43,55,100],

'model__random_state' : [42]}

reg_ridge = GridSearchCV(ridge_pipe, parameters, cv=10)

reg_ridge = reg_ridge.fit(X_train, y_train.ravel()

reg_ridge.best_estimator_, reg_ridge.best_score_

ridge = Ridge(alpha=20, random_state=42)
```

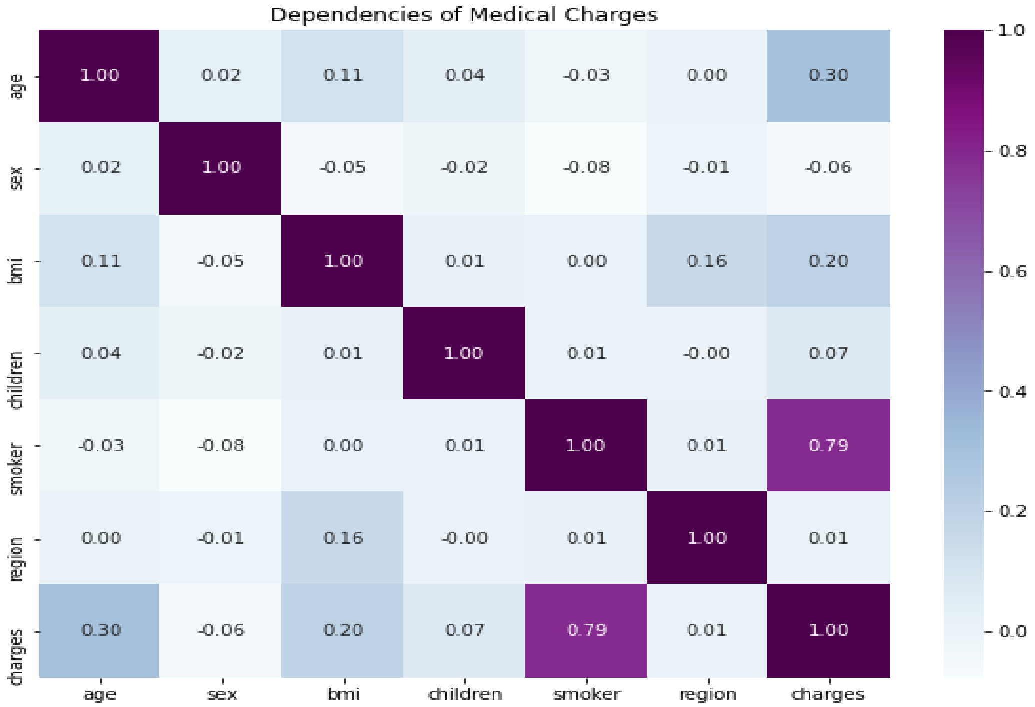
```
ridge.fit(X_train_scaled, y_train_scaled.ravel())
cv_ridge = reg_ridge.best_score_

y_pred_ridge_train = ridge.predict(X_train_scaled)
r2_score_ridge_train = r2_score(y_train_scaled, y_pred_ridge_train)
y_pred_ridge_test = ridge.predict(X_test_scaled)
r2_score_ridge_test = r2_score(y_test_scaled, y_pred_ridge_test)
rmse_ridge = (np.sqrt(mean_squared_error(y_test_scaled, y_pred_linear_reg_test)))
print('CV : {0:.3f}'.format(cv_ridge.mean()))
print('R2 score (train) : {0:.3f}'.format(r2_score_ridge_train))
print('R2 score (test) : {0:.3f}'.format(r2_score_ridge_test))
print('RMSE : {0:.3f}'.format(rmse_ridge))
# ## RandomForest Regressor
get_ipython().run_cell_magic('time', ", 'reg_rf = RandomForestRegressor()\nparameters = {
\'n_estimators\':[600,1000,1200],\n            \'max_features\': ["auto"],\n
\'max_depth\':[40,50,60],\n            \'min_samples_split\': [5,7,9],\n            \'min_samples_leaf\':
[7,10,12],\n            \'criterion\': [\'mse\']}\n\nreg_rf_gscv = GridSearchCV(estimator=reg_rf,
param_grid=parameters, cv=10, n_jobs=-1)\nreg_rf_gscv = reg_rf_gscv.fit(X_train_scaled,
y_train_scaled.ravel())\n')
reg_rf_gscv.best_score_, reg_rf_gscv.best_estimator_
rf_reg = RandomForestRegressor(max_depth=50, min_samples_leaf=12, min_samples_split=7,
                  n_estimators=1200)
rf_reg.fit(X_train_scaled, y_train_scaled.ravel())
```

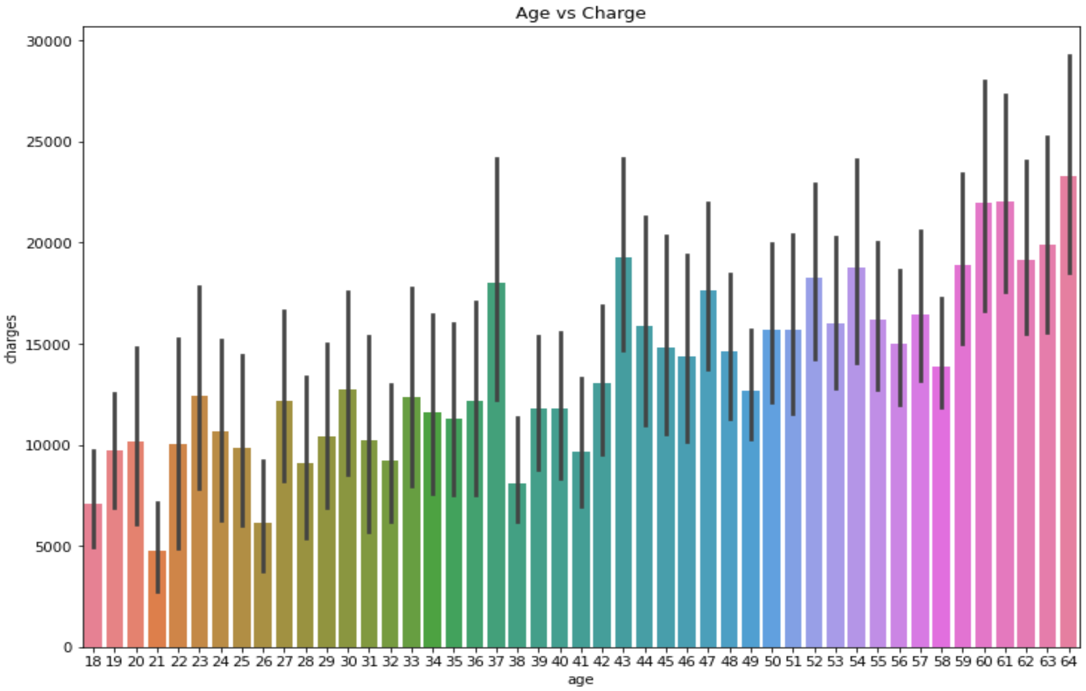
```
cv_rf = reg_rf_gscv.best_score_
y_pred_rf_train = rf_reg.predict(X_train_scaled)
r2_score_rf_train = r2_score(y_train, y_pred_rf_train)
y_pred_rf_test = rf_reg.predict(X_test_scaled)
r2_score_rf_test = r2_score(y_test_scaled, y_pred_rf_test)
rmse_rf = np.sqrt(mean_squared_error(y_test_scaled, y_pred_rf_test))
print('CV : {0:.3f}'.format(cv_rf.mean()))
print('R2 score (train) : {0:.3f}'.format(r2_score_rf_train))
print('R2 score (test) : {0:.3f}'.format(r2_score_rf_test))
print('RMSE : {0:.3f}'.format(rmse_rf))
models = [('Linear Regression', rmse_linear, r2_score_linear_reg_train, r2_score_linear_reg_test, cv_linear_reg.mean()),
          ('Ridge Regression', rmse_ridge, r2_score_ridge_train, r2_score_ridge_test, cv_ridge.mean()),
          ('Support Vector Regression', rmse_svr, r2_score_svr_train, r2_score_svr_test, cv_svr.mean()),
          ('Random Forest Regression', rmse_rf, r2_score_rf_train, r2_score_rf_test, cv_rf.mean())
         ]
predict = pd.DataFrame(data = models, columns=['Model', 'RMSE', 'R2_Score(training)', 'R2_Score(test)', 'Cross-Validation'])
predict
plt.figure(figsize=(12,7))
predict.sort_values(by=['Cross-Validation'], ascending=False, inplace=True)
```

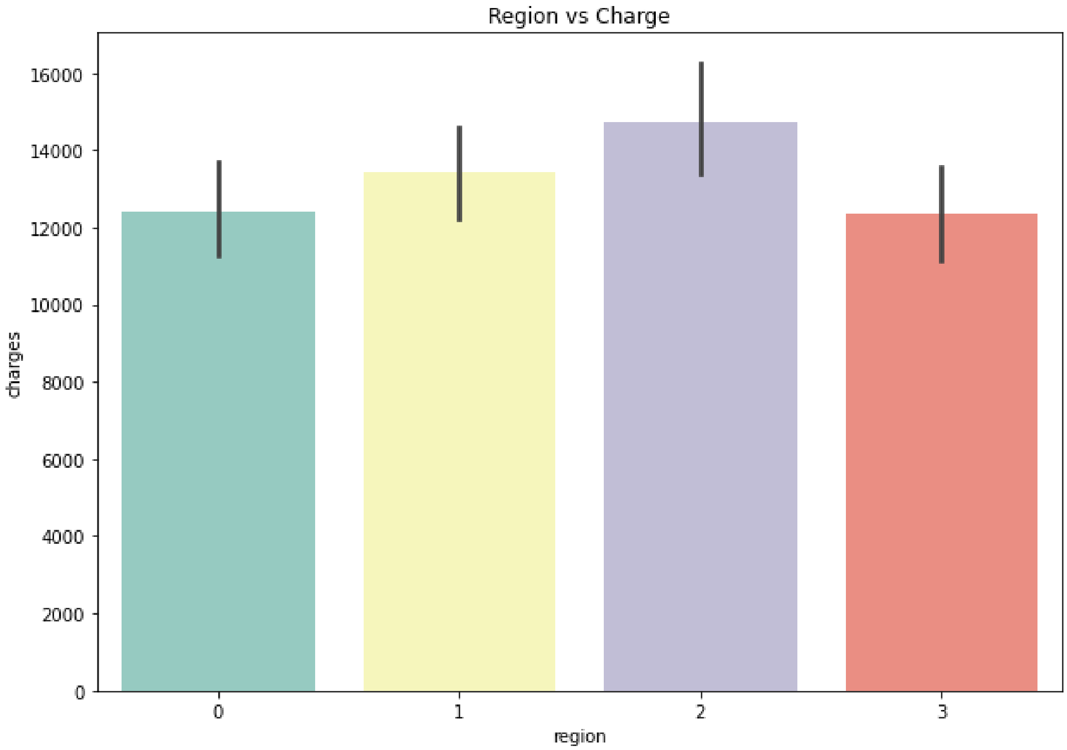
```
sns.barplot(x='Cross-Validation', y='Model',data = predict, palette='Reds')
plt.xlabel('Cross Validation Score')
plt.ylabel('Model')
plt.show()
# ## Training Data without Scaling for RandomClassifier
data_copy.head()
X_ = data_copy.drop('charges',axis=1).values
y_ = data_copy['charges'].values.reshape(-1,1)
from sklearn.model_selection import train_test_split
X_train_, X_test_, y_train_, y_test_ = train_test_split(X_,y_,test_size=0.2, random_state=42)
print('Size of X_train_ : ', X_train_.shape)
print('Size of y_train_ : ', y_train_.shape)
print('Size of X_test_ : ', X_test_.shape)
print('Size of Y_test_ : ', y_test_.shape)
rf_reg = RandomForestRegressor(max_depth=50, min_samples_leaf=12, min_samples_split=7,
                    n_estimators=1200)
rf_reg.fit(X_train_, y_train_.ravel())
y_pred_rf_train_ = rf_reg.predict(X_train_)
r2_score_rf_train_ = r2_score(y_train_, y_pred_rf_train_)
y_pred_rf_test_ = rf_reg.predict(X_test_)
r2_score_rf_test_ = r2_score(y_test_, y_pred_rf_test_)
print('R2 score (train) : {0:.3f}'.format(r2_score_rf_train_))
print('R2 score (test) : {0:.3f}'.format(r2_score_rf_test_))
```

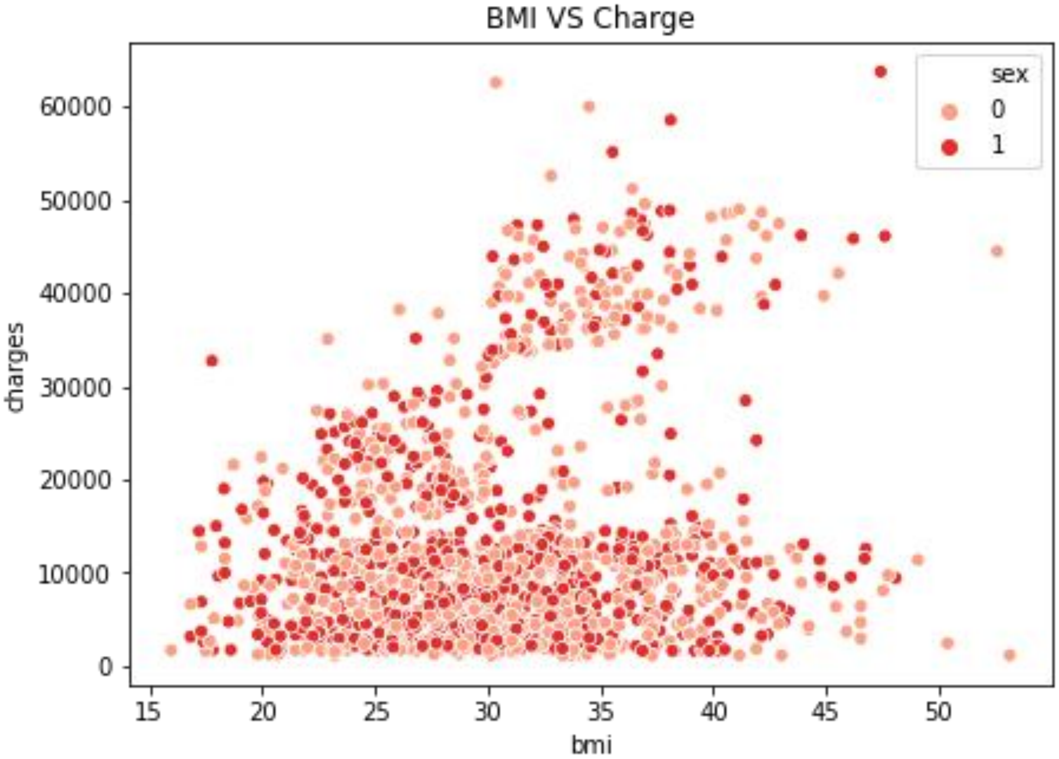
```
import pickle

Pkl_Filename = "rf_tuned.pkl"

with open(Pkl_Filename, 'wb') as file:

    pickle.dump(rf_reg, file)

# Load the Model back from file

with open(Pkl_Filename, 'rb') as file:

    rf_tuned_loaded = pickle.load(file)

rf_tuned_loaded

pred=rf_tuned_loaded.predict(np.array([20,1,28,0,1,3]).reshape(1,6))[0]

print('{0:.3f}'.format(pred))
```

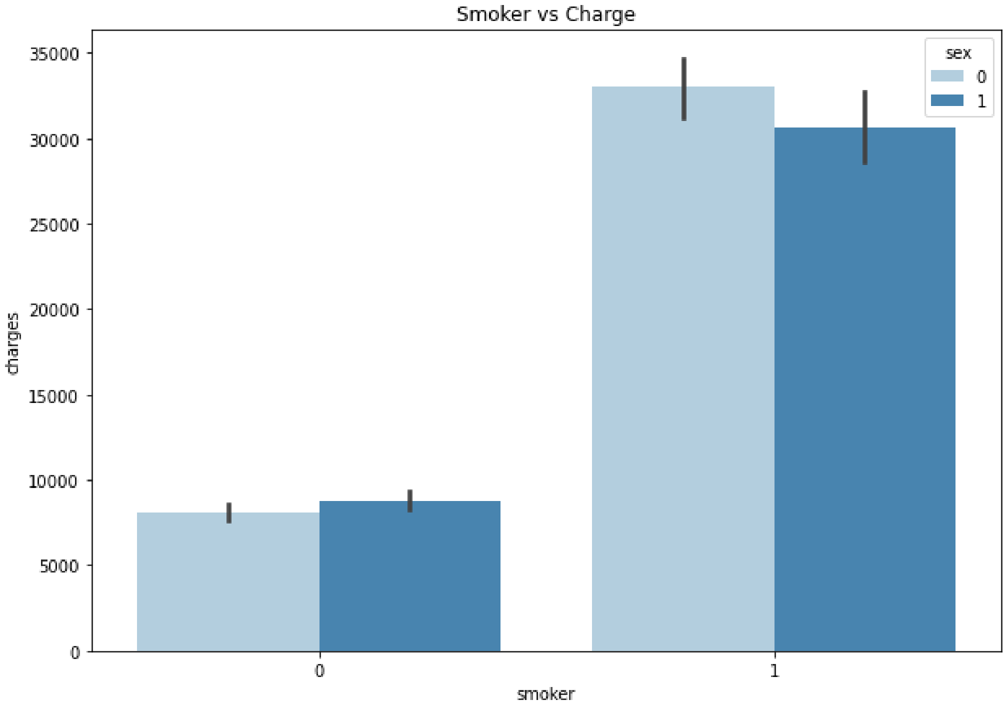
Outputs:

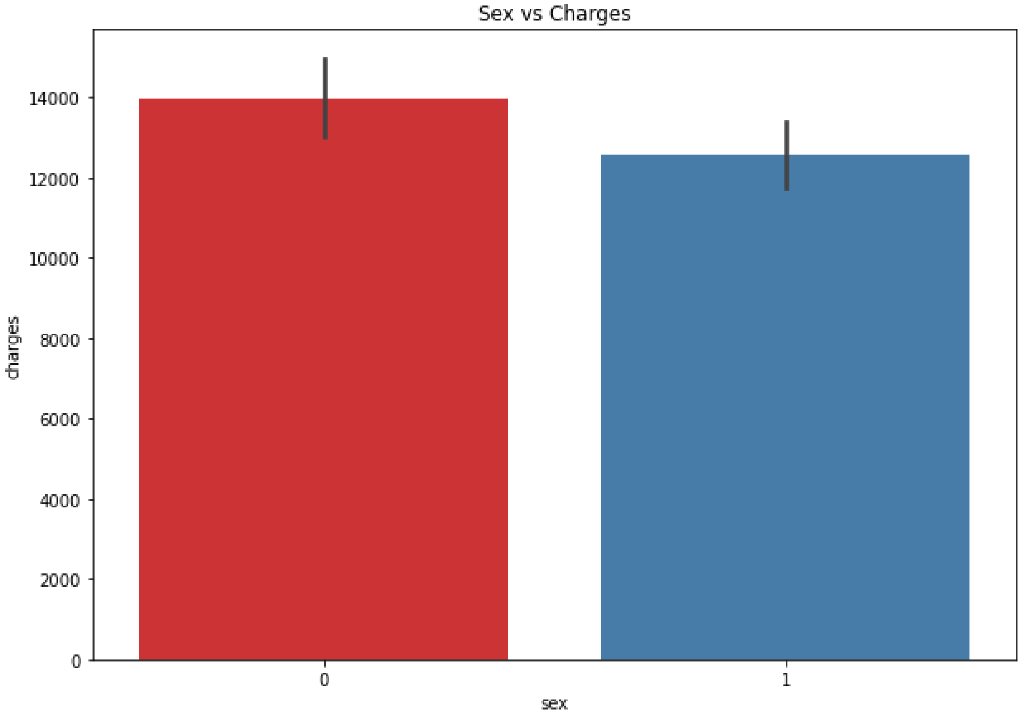

Dependencies of Medical Charges
age
sex
bmi
children
smoker
region
charges
1.00 0.02 0.11 0.04 -0.03 0.00 0.30
0.02 1.00 -0.05 -0.02 -0.08 -0.01 -0.06
0.11 -0.05 1.00 0.01 0.00 0.16 0.20
0.04 -0.02 0.01 1.00 0.01 -0.00 0.07
-0.03 -0.08 0.00 0.01 1.00 0.01 0.79
0.00 -0.01 0.16 -0.00 0.01 1.00 0.01
0.30 -0.06 0.20 0.07 0.79 0.01 1.00
age
sex
bmi
children
smoker
region
charges
1.0
0.8
0.6
0.4
0.2
0.0

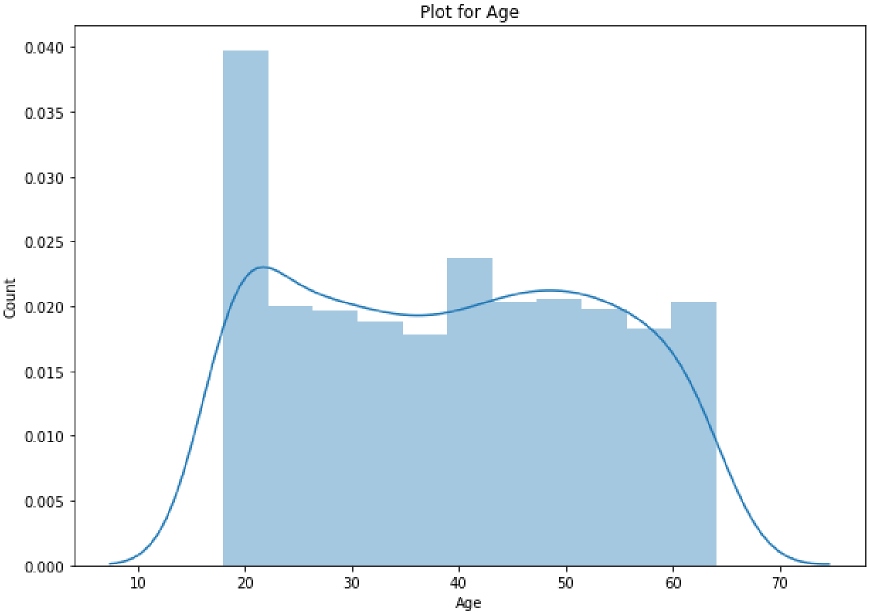

Age vs Charge
30000
25000
20000
15000
10000
5000
0
charges
18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64
age

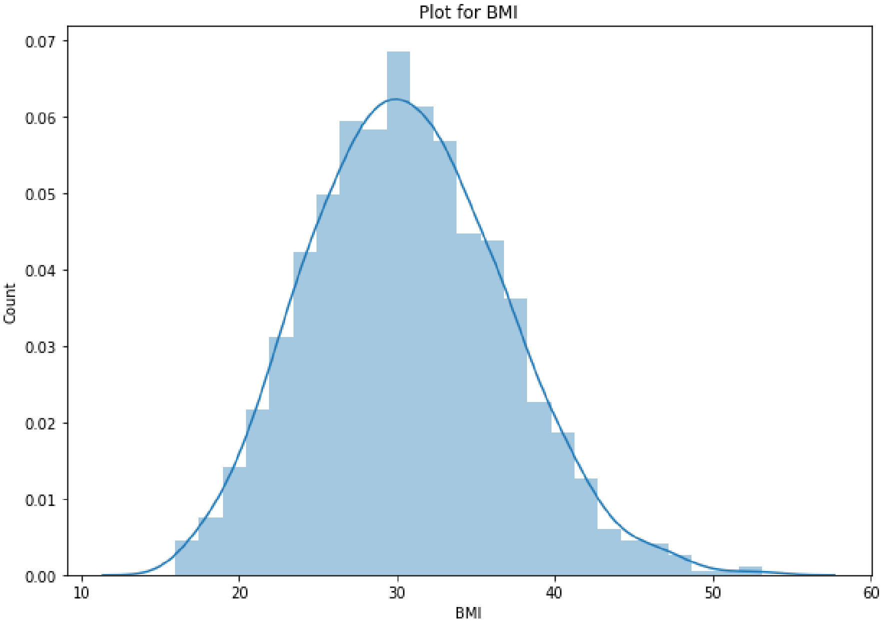

Region vs Charge
charges
16000
14000
12000
10000
8000
6000
4000
2000
0
0
1
2
3
region

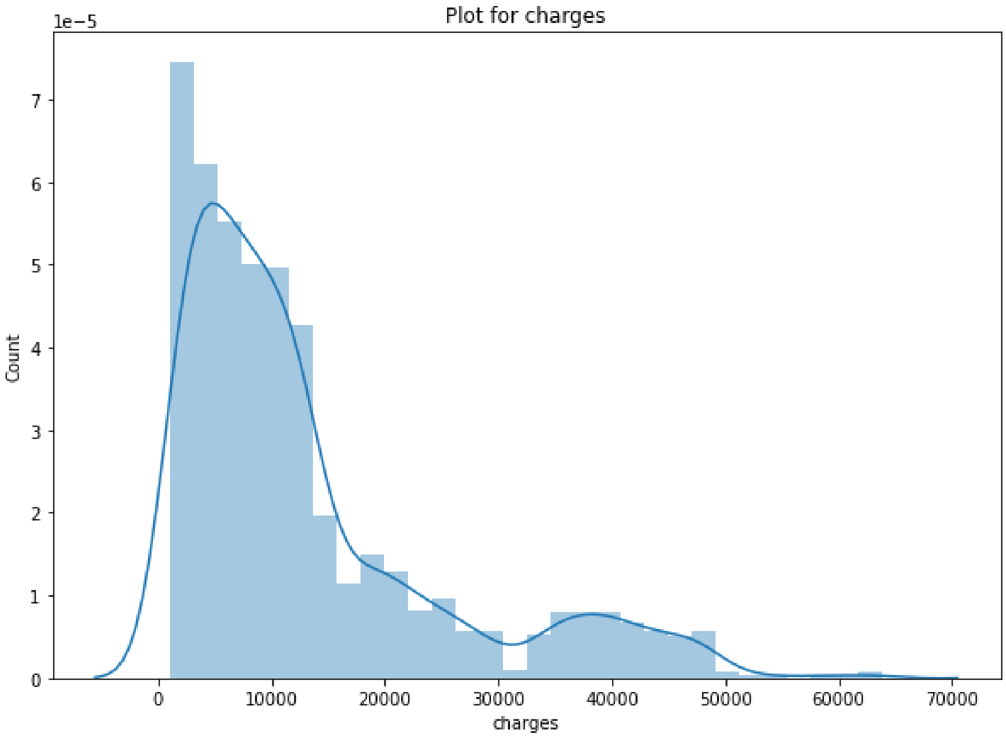

BMI VS Charge
sex
0
1
charges
60000
50000
40000
30000
20000
10000
0
15
20
25
30
35
40
45
50
bmi

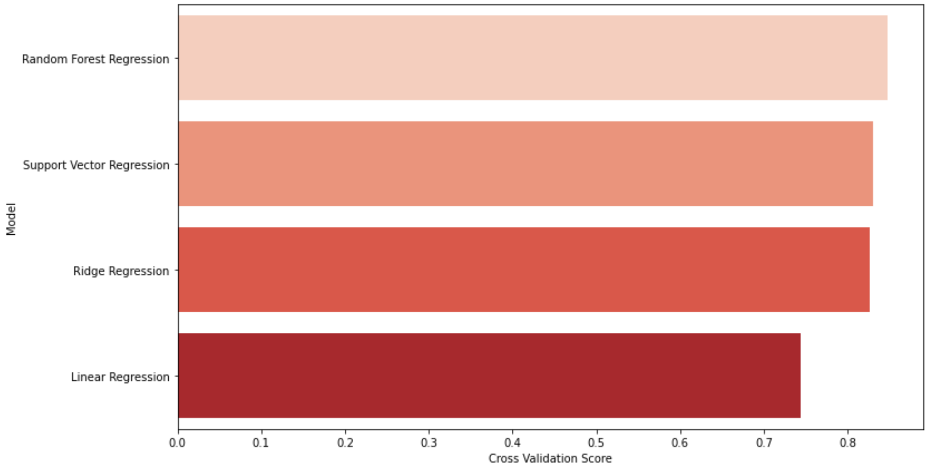

Smoker vs Charge
sex
0
1
35000
30000
25000
20000
15000
10000
5000
0
charges
0
1
smoker


Sex vs Charges
14000
12000
10000
8000
6000
4000
2000
0
charges
0
1
sex


Plot for Age
0.040
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
Count
10
20
30
40
50
60
70
Age


Plot for BMI
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Count
10
20
30
40
50
60
BMI


Plot for charges
1e−5
Count
0
1
2
3
4
5
6
7
0
10000
20000
30000
40000
50000
60000
70000
charges


Random Forest Regression
Support Vector Regression
Ridge Regression
Linear Regression
Model
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Cross Validation Score

Chapter 5

Conclusion

This paper presented a regression-based model on the USA's medical cost personal dataset available on Kaggle. We demonstrated the correlation of various features: Age, BMI, Smoking habits and obesity with overall charges of insurance. We have also developed a website to easily interact with the system and predict the total cost of insurance. We have used linear regression to determine the correlation between various features and obtained 81.3% accuracy. This project aimed to develop a machine learning-based system capable of accurately predicting individual medical insurance costs using a variety of personal health indicators. By leveraging regression techniques—specifically Linear Regression and ensemble methods like Random Forest and XGBoost—we were able to evaluate and compare predictive performance across several model architectures.

Through rigorous model training and evaluation on a dataset sourced from Kaggle, the analysis revealed that advanced machine learning models such as XGBoost and Neural Networks significantly outperformed traditional regression methods. These models demonstrated strong predictive accuracy, particularly due to their capacity to capture complex, non-linear relationships among input features such as age, BMI, and smoking status.

In addition to modeling, this study emphasizes the potential impact of such predictive systems in real-world applications. For individuals, it provides a transparent and data-driven estimation of expected insurance costs, enabling more informed financial and health planning. For insurers, the models offer a reliable tool for risk assessment and personalized policy pricing.

The project's secondary contribution—a web interface for interactive predictions—demonstrates the model’s practical usability and accessibility to non-technical users.

Despite its success, the project also identified challenges such as data imbalance, feature sensitivity, and computational demands associated with deep learning methods. These limitations pave the way for future improvements, including better feature engineering, application of LSTM-based architectures, and enhanced hyperparameter optimization.

Ultimately, this study contributes meaningfully to the growing field of AI in healthcare by showcasing how predictive analytics can support smarter, fairer, and more personalized insurance systems. With continued refinements, the proposed approach can scale to broader populations and real-time applications, offering a compelling tool for both consumers and industry stakeholders.

References

[1] M. A. Morid, K. Kawamoto, T. Ault, J. Dorius, and S. Abdelrahman, “Supervised Learning Methods for Predicting Healthcare Costs: Systematic Literature Review and Empirical Evaluation,” AMIA Annual Symposium Proceedings, vol. 2017, p. 1312, 2017.

[2] R. Tkachenko, H. Kutucu, I. Izonin, A. Doroshenko, and Y. Tsymbal, “Non-iterative Neural-like Predictor for Solar Energy in Libya,” in Proceedings of the 14th International Conference on ICT in Education, Research and Industrial Applications., Kyiv, Ukraine, May 14-17, 2018, 2018, vol. 2105, pp. 35–45.

[3] Drewe-Boss, Philipp, Dirk Enders, Jochen Walker, and Uwe Ohler. "Deep learning for prediction of population health costs." BMC Medical Informatics and Decision Making 22, no. 1 2022, pp 1-10.

[4] Powers, C. A., C. M. Meyer, M. C. Roebuck, B. Vaziri. “Predictive modeling of total healthcare costs using pharmacy claims data: A comparison of alternative econometric cost modeling techniques”. Med. Care 43, 2005 pp 1065–1072.

[5] Dove, H., I. Duncan, A. Robb . “A prediction model for targeting low-cost, high-risk members of managed care organizations” . Amer. J. Managed Care 9, 2003 pp 381–389.

[6] PolitiMC, Shacham E, Barker AR, George N,Mir N, Philpott S, et al. A Comparison Between Subjective and ObjectiveMethods of Predicting Health Care Expenses to Support Consumers’

[7] Health Insurance Plan Choice. MDM Policy & Practice. 2018; 3(1):238146831878109. doi:10.1177/2381468318781093.

[8] Medical Cost Personal Datasets.: https://www.kaggle.com/mirichoi0218/insurance. last accessed 10/2/2022.