**VISVESVARAYA TECHNOLOGICAL UNIVERSITY BELGAUM -590014**

**A Mini-Project (21AIMP67)**

**Report On**

# *"Heath insurance cost prediction"*

A Mini-project report submitted in partial fulfillment of the requirements for the award of the degree of **Bachelor of Engineering in Artifical Intelligence and Machine Learning** of Visvesvaraya Technological University, Belgaum.

Submitted by:

**Nikitha(1DT21AI040)**

**Soumya (1DT21AI054**

**Swara(1DT21AI058)**

Under the Guidance of:

**Prof.Mahalakshmi G**

**Assistant Professor, Dept of AIML**

**DAYANANDA SAGAR ACADEMY OF TECHNOLOGY & MANAGEMENT**

Opp. Art of Living, Udayapura, Kanakapura Road, Bangalore- 560082(Affiliated to Visvesvaraya Technological University, Belagavi and Approved by AICTE, NewDelhi). (Accredited by NBA until 30-06-2025, NAAC (A+))

**DAYANANDA SAGAR ACADEMY OF TECHNOLOGY & MANAGEMENT**

Opp. Art of Living, Udayapura, Kanakapura Road, Bangalore- 560082

(Affiliated to Visvesvaraya Technological University, Belagavi and Approved by AICTE, NewDelhi).
(Accredited by NBA until 30-06-2025, NAAC (A+))

## DEPARTMENT OF ARTIFICAL INTELLIGENCE AND MACHINE LEARNING

## CERTIFICATE

This is to certify that the Mini-Project on "**Health insurance cost prediction**" has been successfully carried out by **Nikitha(1DT21AI040) Soumya(1DT21AI054) Swara(1DT21AI058),**a Bonafide students of **Dayananda Sagar Academy of Technology and Management** in partial fulfilment of the requirements for the award of degree in **Bachelor of Engineering** in **Artificial intelligence and machine learning** of the **Visvesvaraya Technological University, Belgaum** during academic year 2023-24. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library.

Signature

**Prof.Mahalakshmi G**

**Assistant Professor,**
**AIML,DSATM.**

Signature

**Dr. Sandhya N**

**Professor&HoD**
**AIML,DSATM.**

# ACKNOWLEDGEMENT

It gives us immense pleasure to present before you our project titled **“Health insurance cost prediction”** The joy and satisfaction that go with the successful completion of any task would be incomplete without the mention of those who made it possible. We are glad to express our gratitude towards our prestigious institution DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT for providing us with utmost knowledge, encouragement, and the maximum facilities in undertaking this project.

We sincerely acknowledge the guidance and constant encouragement of our mini- project guide Assistant professor **G.Mahalakshmi**, Asst. Professor, Dept of AIML .

We express our deepest gratitude and special thanks to **Dr. Sandhya N, Prof &H.O.D**, Dept. of Artificial Intelligence and Machine Learning, for all her guidance and encouragement.

We wish to express a sincere thanks to our respected principal

**Dr.M. Ravishankar,** Principal, DSATM for all their support.

.

**NIKITHA 1DT21A040**

**SOUMYA 1DT21AI054**

**SWARA 1DT21AI058**

# ABSTRACT

Health insurance is a critical component of healthcare systems worldwide, providing financial protection against high medical costs. Accurately predicting health insurance costs is essential for insurance companies to price their policies appropriately and for individuals to understand their potential expenses. This project aims to develop a predictive model for health insurance costs using various demographic, lifestyle, and medical factors Leveraging a dataset that includes variables such as age, gender, body mass index (BMI), number of children, smoking status, and geographic region, In the proposed system, machine learning techniques are used to identify patterns and predict insurance charges. The project will explore multiple regression models, including linear regression, decision trees, and ensemble methods like random forests and gradient boosting machines, to determine the most effective approach. Evaluation metrics such as mean absolute error (MAE), mean squared error (MSE), and R-squared will be used to assess model performance. Additionally, feature importance analysis will provide insights into the most significant predictors of insurance costs, potentially guiding policy adjustments and personalized insurance plans. By accurately forecasting health insurance costs, this project aims to contribute to more efficient and equitable health insurance pricing strategies.

# TABLE OF CONTENTS

# LIST OF FIGURES

# CHAPTER 1

# INTRODUCTION

## 1. Background

Health insurance is a critical component of the healthcare system, offering financial protection against high medical expenses. Predicting health insurance costs is essential for both insurers and policyholders. Insurers can use predictive models to set premiums, manage risk, and ensure profitability, while policyholders benefit from more accurate premium rates that reflect their health status and risk factors.

Health insurance cost prediction is a complex but essential task that benefits from a multidisciplinary approach, combining data science, healthcare expertise, and actuarial knowledge. By leveraging advanced analytical techniques and comprehensive data, insurers can improve cost predictions, leading to better financial management and more equitable access to healthcare.

## 2. Problem Definition

The rising costs of healthcare have made it crucial for insurance companies, healthcare providers, and individuals to predict health insurance expenses accurately. Predicting these costs can help insurance companies set appropriate premiums, assist healthcare providers in managing patient care, and enable individuals to budget for their healthcare expenses.

The primary objective of this project is to develop a predictive model that accurately estimates the annual health insurance costs for individuals based on various personal and health-related factors.

## 3. Motivation

- **Cost Management for Insurers**: Accurate predictions can help insurance companies manage their costs better, setting premiums that reflect the true risk and expected expenses.

- **Personalized Pricing**: Predictive models can lead to more personalized pricing, ensuring that individuals pay premiums that are fair and commensurate with their specific risk factors.

- **Enhanced Customer Satisfaction**: By accurately predicting costs, insurers can reduce instances of unexpected premium hikes or denials, leading to greater customer satisfaction and trust.

- **Preventive Healthcare**: Insights from predictive models can encourage preventive measures. If insurers understand what factors lead to higher costs, they can work with healthcare providers to implement preventive strategies, ultimately leading to healthier populations.

## 4. Objective

Creating a health insurance cost prediction model involves several key objectives. Here's a comprehensive list of objectives you might consider for your project:

1. **Data Collection and Preprocessing:**
    - Gather relevant data on health insurance costs, demographics, health metrics, and other influencing factors.
    - Clean and preprocess the data to handle missing values, outliers, and ensure data consistency.
    - Perform exploratory data analysis (EDA) to understand the data distribution and relationships between variables.
2. **Feature Engineering:**
    - Identify and select relevant features that significantly impact health insurance costs.
    - Create new features if necessary, such as age groups, BMI categories, or interaction terms.
3. **Model Selection:**
    - Choose appropriate machine learning models for prediction, such as linear regression, decision trees, random forests, or gradient boosting machines.
    - Consider both regression models (for continuous cost prediction) and classification models (for categorizing cost ranges).

4. **Model Training and Evaluation:**

Split the data into training and testing sets to evaluate model performance.

   - Train the selected models using the training data and optimize hyperparameters.
   - Evaluate the models using appropriate metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), or R-squared for regression models. For classification models, use metrics like accuracy, precision, recall, and F1-score.

5. **Model Interpretation:**
   - Interpret the results to understand the key drivers of health insurance cost.Use techniques like feature importance, SHAP values, or partial dependence plots to explain model predictions.

## .5. Scope of the project

Identify and collect data from reliable sources such as public health datasets, insurance companies, healthcare institutions, and surveys. Include demographic data (age, gender, region), lifestyle data (smoking status, BMI, exercise frequency), medical history (pre-existing conditions, family medical history), and insurance cost data. The goal of this project is to develop a predictive model that can estimate the cost of health insurance for individuals based on various features such as age, gender, BMI, smoking status, number of dependents, and other relevant factors. This model will help insurance companies in pricing their policies more accurately and can also be used by individuals to estimate their potential insurance costs.

# CHAPTER 2

# LITERATURE SURVEY

## 2.1 LITERATURE REVIEW

**Research paper 1**

**Name : Health insurance cost prediction using machine learning**

**Publisher : IEEE**

Actuarial modeling in health insurance has become crucial for setting effective premiums, essential for attracting and retaining insured individuals and managing existing plans. However, building accurate predictive models is challenging due to various factors influencing medical insurance costs, such as demographics, health status, lifestyle, and plan specifics. The COVID-19 pandemic has highlighted the need for a transparent insurance system. Machine learning (ML) has been effective in predicting high-cost patient expenditures, leading insurers to adopt ML for better policy and premium settings. Nonetheless, ML's black-box nature can introduce bias, but Explainable AI (XAI) methods enhance transparency and acceptability, improving accountability and control in patient care. This paper compares three ensemble ML models—XGBoost, GBM, and RF—for predicting medical insurance costs using a dataset from Kaggle. XGBoost achieved the highest R2 score (86.470%) and lowest RMSE (2231.524) but required substantial computing resources. The RF model had the best MAE (1379.960) and MAPE (5.831%) and was the fastest and most memory-efficient. The GBM model had larger prediction errors compared to XGBoost and RF.

**Research paper 2**

**Name : Health insurance cost prediction using machine learning**

**Publisher : Ajay Sahu, Gopal Sharma**

This study explores how different regression models can forecast insurance costs, comparing models such as Multiple Linear Regression, Generalized Additive Model, Support Vector Machine, Random Forest Regressor, CART, XGBoost, k-Nearest

Neighbors, Stochastic Gradient Boosting, and Deep Neural Network. The Stochastic Gradient Boosting model was identified as the best approach, achieving an MAE of 0.17448, RMSE of 0.38018, and R-squared value of 85.8295. The research utilizes various machine learning regression models and deep neural networks to predict health insurance charges using a dataset from Kaggle. The findings, summarized in Table IV,show that Stochastic Gradient Boosting offers the best performance with an RMSE of0.380189, MAE of 0.17448, and an accuracy of 85.82%. This model outperforms otherregression models in estimating insurance costs. Using ML for forecasting can help insurance providers attract consumers, save time in plan formulation, and improve profitability by efficiently managing large datasets.

**Research Paper 3**

**Name : Health Insurance cost prediction using machine learning**

**Publisher : Sazzad Hossen**

This paper presents a machine learning-based system for predicting health insurance costs, using a dataset from the USA with 1338 entries from Kaggle. Key features for prediction include age, gender, BMI, smoking habit, and number of children. The system trained with a 70-30 data split, utilized linear regression to determine the relationship between price and these features, achieving an accuracy of 81.3%. This research is significant, especially post-COVID-19, as health insurance prediction has become a major research focus. Machine learning (ML) can significantly enhance health insurance operations by analyzing and evaluating large volumes of data quickly, saving time and money for policyholders and insurers. ML can handle repetitive tasks, allowing insurance experts to focus on improving the policyholder experience. This study used an artificial neural network (ANN)-based regression model to predict health insurance premiums, achieving an accuracy of 92.72%. The model was evaluated using metrics like RMSE, MSE, MAE, $R^2$, and adjusted $R^2$, and a correlation matrix was plotted to examine relationships between various factors and charges. The field of insurance prediction with ML still requires extensive research trained with a 70-30 data split, utilized linear regression to determine the relationship between price and these features, achieving an accuracy of 81.3%. This research is significant, especially post-COVID-19, as health insurance prediction has become a major research focus. Machine learning (ML) can significantly enhance health insurance operations by analyzing and evaluating large volumes of data quickly, saving time and money for policyholders and insurers. ML can handle repetitive tasks, allowing insurance experts to focus on improving the policyholder experience. This study used an artificial neural network (ANN)-based regression model to predict health insurance premiums, achieving an accuracy of 92.72%. The model was evaluated using metrics like RMSE, MSE, MAE, $R^2$, and adjusted $R^2$, and a correlation matrix was plotted to examine relationships between various factors and charges. The field of insurance prediction with ML still requires extensive research

### Research Paper 4

### Name: Machine Learning-Based Regression Framework to Predict Health Insurance Premiums

### Publisher :Keshav Kaushik, Akshadeep Bharadwaj

Healthcare spending accounts for around 30% of GDP, especially in developed countries, with significant costs covered by government programs like Medicare. Rising healthcare costs and the aging baby boomer population strain government finances, necessitating cost-limiting measures. This study aims to predict medical costs using machine learning algorithms to help patients find affordable options and enable policymakers to identify and address expensive providers. The Random Forest Regression algorithm will be used, with comparisons to Gradient Boosted Trees and Linear Regression models. Early cost estimation can prevent people from overpaying for unnecessary health insurance, providing a general sense of potential expenses. Traditional calculation of health insurance charges is time-consuming and prone to errors. Machine learning (ML) models can streamline this process. This paper uses several ML regression models to predict insurance costs based on dataset attributes. The Gradient Boosting Regression model is the most efficient, with an RMSE of 2447.95, $R^2$ of 0.87, and accuracy of 87.79%.

Models are ranked by performance: Gradient Boosting, Random Forest, Support Vector Regression, and Linear Regression. These models can save companies time and costs and can be deployed on cloud platforms for faster real-time data processing as data volume grows.

.

# CHAPTER 3

# REQUIREMENTS

The requirements can be broken down into 2 major categories namely hardware and software requirements. The former specifies the minimal hardware facilities expected in a system where the project must be run. The latter specifies the essential software needed to build and run the project.

## 3.1 Hardware Requirements

The Hardware requirements are very minimal and the program can be run on most of the machines.

- **CPU**: High-performance multi-core processor
- **RAM**: 32 GB or more
- **Storage**: 512 GB SSD or larger
- **GPU**: Mid-range GPU with at least 4-6 GB VRAM

## 3.2 Software Requirements

| | |
|---|---|
| Technology Implemented | : Jupyter notebook,python,streamlit |
| Language Used | : python |
| User Interface | : python using streamlit |
| Web Browser | : Firefox |

# 4.List of figures

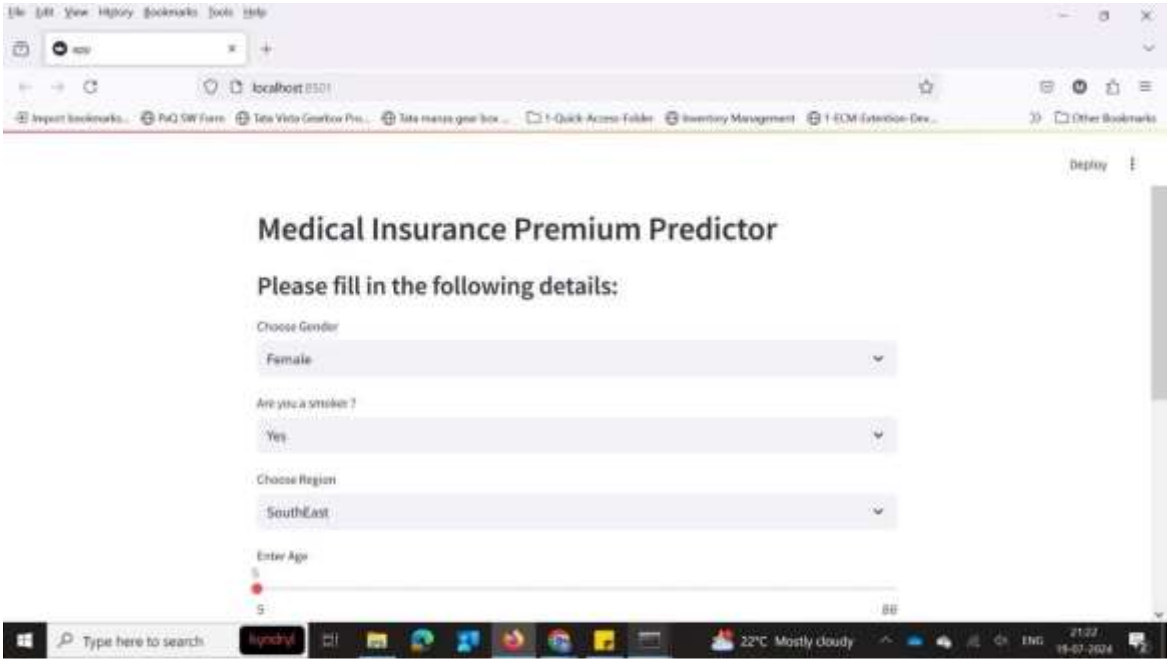


*Figure 4.1*

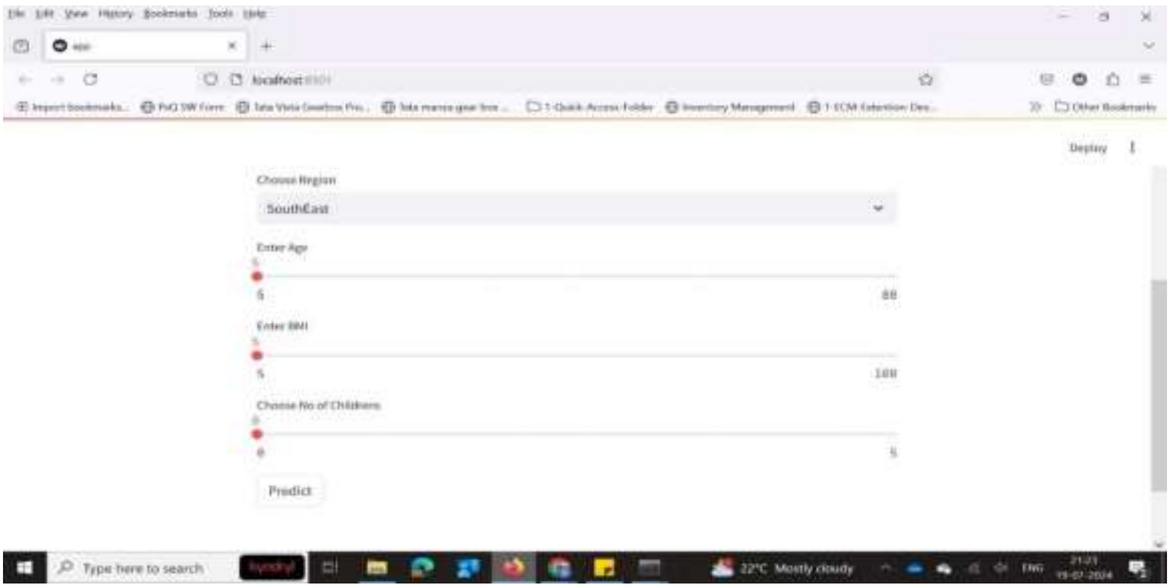


*Figure 4.2*

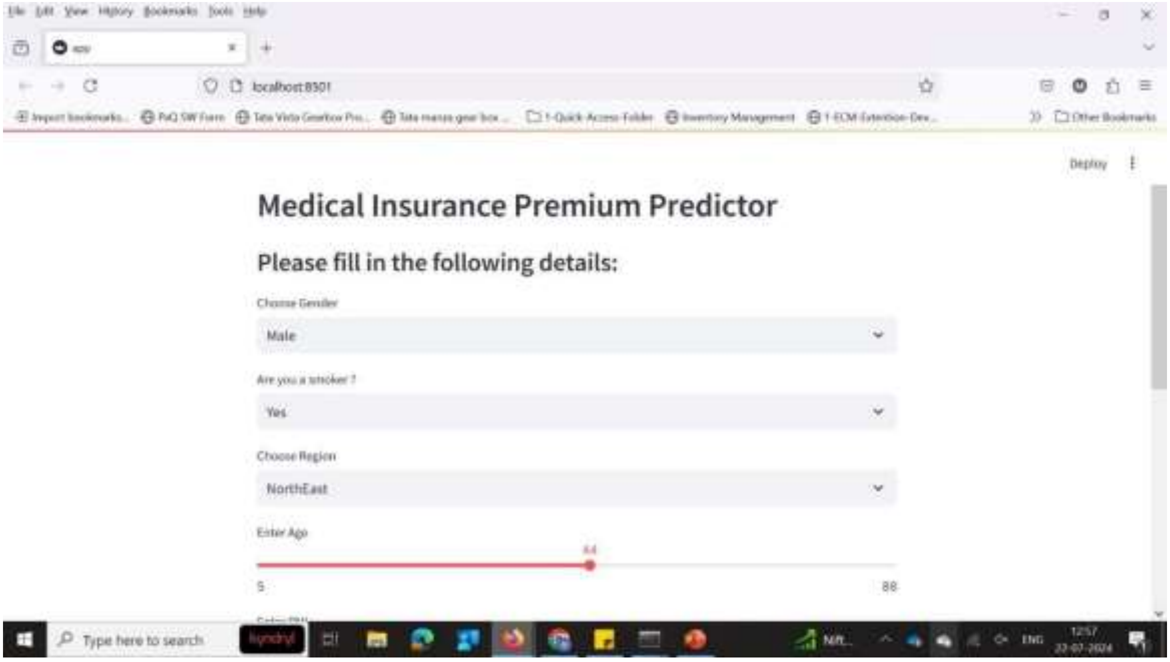


*Figure 4.3*

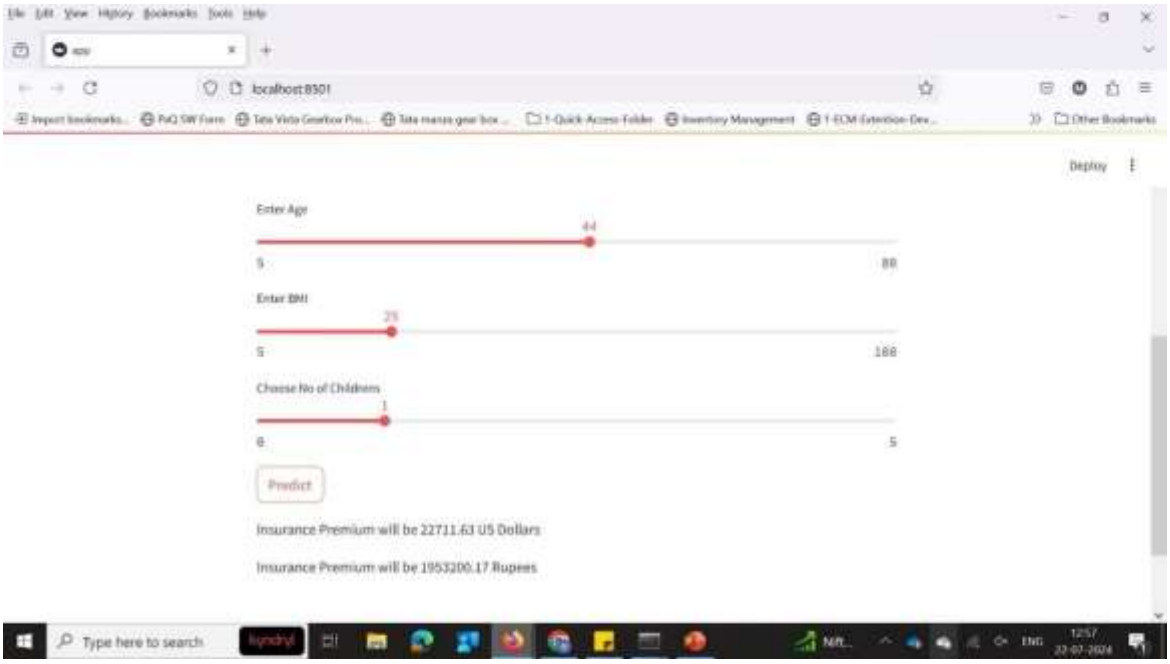


*Figure 4.4*

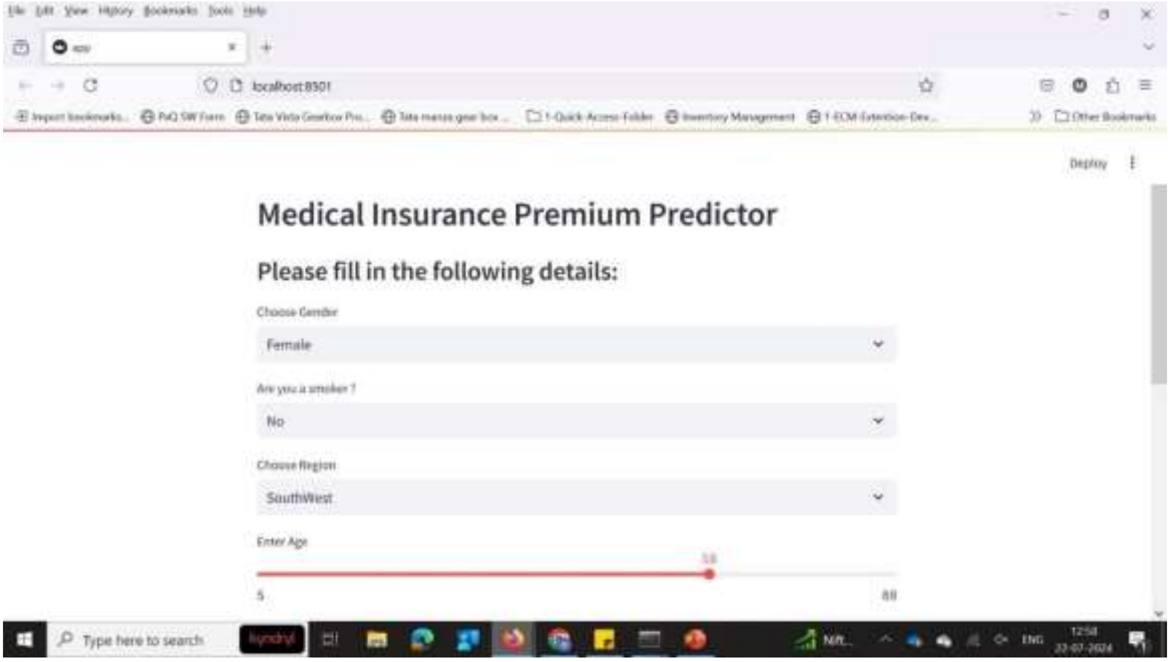


*Figure 4.5*

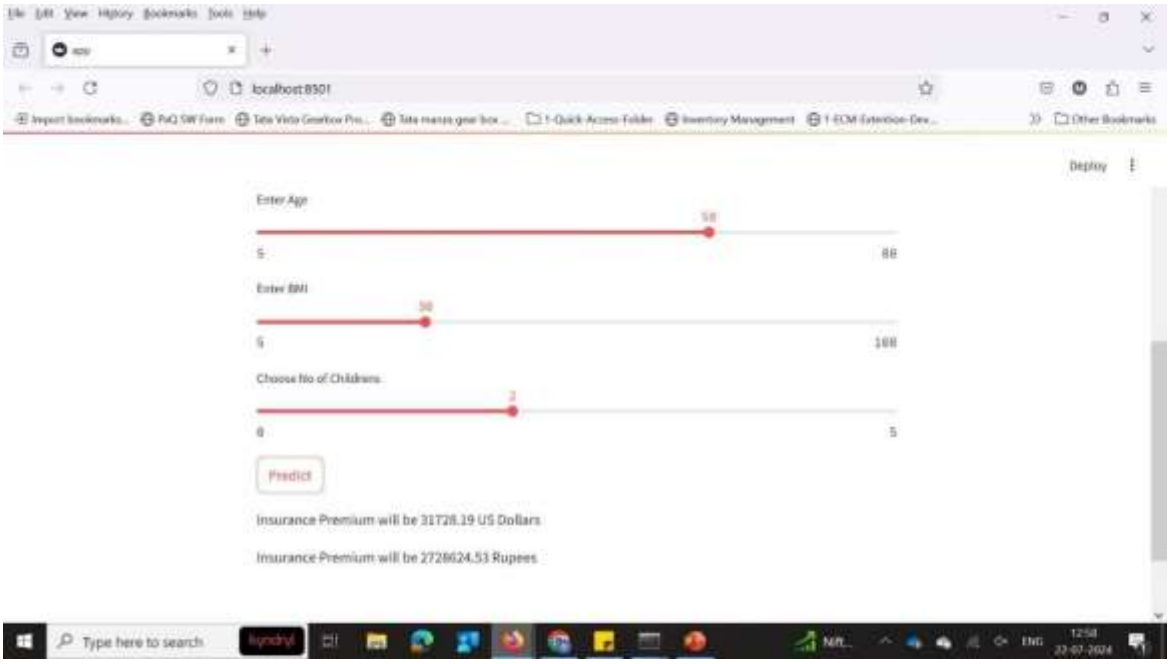


*Figure4.6*

# CHAPTER 5

## 5.1 FLOWCHART

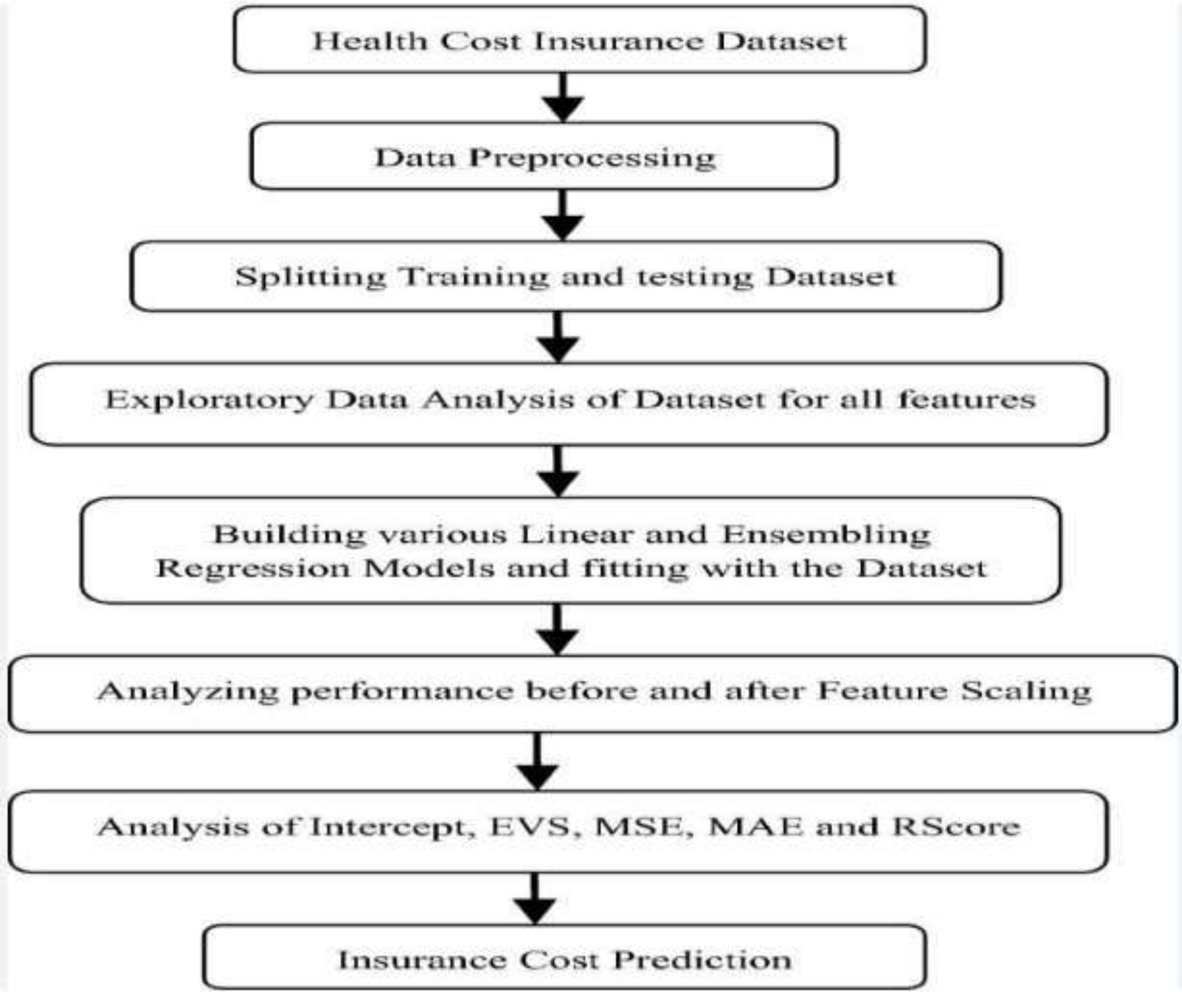


*Fig 4.7 Flowchart*

Creating a flowchart for a health insurance cost prediction project involves outlining the key steps and processes needed to build a predictive model. Here's a description of the typical flowchart steps for such a project:

1. **Problem Definition**
   - **Identify Objectives**: Define the goal of predicting health insurance costs.
   - **Define Scope**: Determine the specific metrics and scope of the prediction.
2. **Data Collection**
   - **Identify Data Sources**: List the sources of relevant data (e.g., insurance claims, demographic data, health records).
   - **Data Gathering**: Collect data from identified sources.

- **Data Transformation**: Convert data into suitable formats (e.g., normalize, standardize).
- **Feature Engineering**: Create new features from existing data to improve model performance.

3. **Exploratory Data Analysis (EDA)**
   - **Visualizations**: Create charts and graphs to understand data distribution and relationships.
   - **Statistical Analysis**: Perform statistical tests to identify significant features.
4. **Data Splitting**
   - **Train-Test Split**: Split the data into training and testing sets to evaluate model performance.
5. **Model Selection**
   - **Choose Algorithms**: Select appropriate machine learning algorithms (e.g., linear regression, random forest, XGBoost).
   - **Baseline Model**: Build a simple model to set a performance benchmark.
6. **Model Training**
   - **Train Models**: Use training data to train selected models.
   - **Hyperparameter Tuning**: Optimize model parameters to improve performance.
7. **Model Evaluation**
   - **Performance Metrics**: Evaluate models using metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and $R^2$.
   - **Cross-Validation**: Perform cross-validation to ensure model robustness.
8. **Model Selection and Validation**
   - **Select Best Model**: Choose the model with the best performance metrics.
   - **Validate Model**: Validate the selected model on the test set to ensure it generalizes well.
9. **Model Deployment**
   - **Integration**: Integrate the model into the production environment.
   - **API Development**: Develop APIs to allow other systems to use the model for predictions.
   - **Monitoring**: Set up monitoring to track model performance over time.
10. **Documentation and Reporting**
    - **Model Documentation**: Document model development, assumptions, and limitations.
    - **Reporting**: Create reports and dashboards to communicate results .

# CHAPTER 6

## IMPLEMENTATION

For a project on health insurance cost prediction, you'll likely need to implement a system that can predict insurance costs based on various factors. Here's a general approach to implementing such a project:

1. Define the Problem and Objectives

Objective: Predict the cost of health insurance based on features like age, gender, BMI, smoking status, etc

2. Gather and Prepare Data

Data Collection: Collect historical data on insurance costs and relevant features.

Data Sources: Health records, insurance claims, surveys.

Data Cleaning: Handle missing values, outliers, and normalize or scale data if necessary.

Feature Engineering: Create new features from existing ones, such as age groups, BMI categories.

3. Explore and Analyze Data

Exploratory Data Analysis : Visualize data distributions, correlations, and relationships between features and the target variable.

Statistical Analysis: Use statistical tests to understand the impact of different features on insurance costs.

4. Choose a Modeling Approach

Regression Models: Linear Regression, Polynomial Regression, Lasso/Ridge Regression.

Machine Learning Models: Decision Trees, Random Forest, Gradient Boosting Machines, Neural Networks.

Evaluation Metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), R-squared.

## 6.1 ALGORITHM

**Step 1**: Visit Homepage with website URL

**Step 2**: choose details

**Step 3**: choose gender

**Step 4**: fill in all details accordingly

**Step 5**: click predict

## 6.2 SOURCE CODE

### 6.2.1 Connection establishment between front-end and back-end: config.php **Backend code with outputs:**

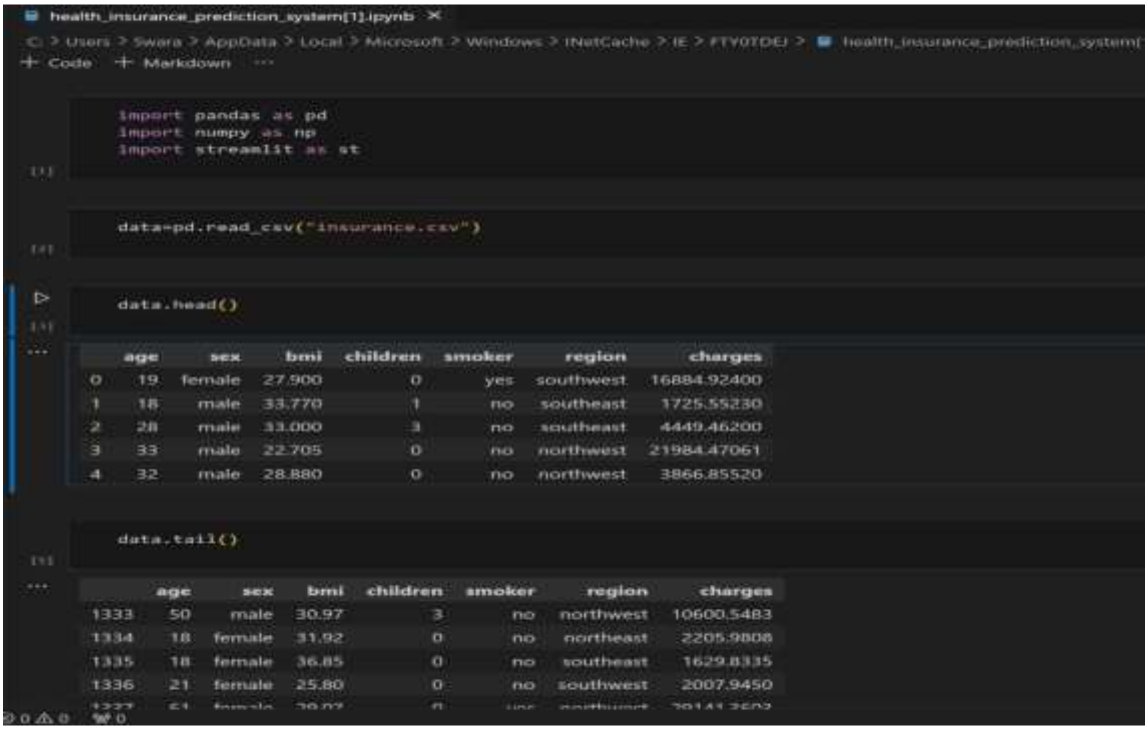

*Figure 5.1*

```
data.tail()
```

| | age | sex | bmi | children | smoker | region | charges |
|---|---|---|---|---|---|---|---|
| 1333 | 50 | male | 30.97 | 3 | no | northwest | 10600.5483 |
| 1334 | 18 | female | 31.92 | 0 | no | northeast | 2205.9808 |
| 1335 | 18 | female | 36.85 | 0 | no | southeast | 1629.8335 |
| 1336 | 21 | female | 25.80 | 0 | no | southwest | 2007.9450 |
| 1337 | 61 | female | 29.07 | 0 | yes | northwest | 29141.3603 |

```
data.shape
```

```
(1338, 7)
```

```
print("Number of Rows",data.shape[0])
print("Number of Columns",data.shape[1])
```

```
Number of Rows 1338
Number of Columns 7
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
```

*Figure 5.2*

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1338 non-null   int64
 1   sex       1338 non-null   object
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
 5   region    1338 non-null   object
 6   charges   1338 non-null   float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.3+ KB
```

```
data.describe()
```

| | age | bmi | children | charges |
|---|---|---|---|---|
| count | 1338.000000 | 1338.000000 | 1338.000000 | 1338.000000 |
| mean | 39.207025 | 30.663397 | 1.094918 | 13270.422265 |
| std | 14.049960 | 6.098187 | 1.205493 | 12110.011237 |
| min | 18.000000 | 15.960000 | 0.000000 | 1121.873900 |
| 25% | 27.000000 | 26.296250 | 0.000000 | 4740.287150 |
| 50% | 39.000000 | 30.400000 | 1.000000 | 9382.033000 |
| 75% | 51.000000 | 34.693750 | 2.000000 | 16639.912515 |
| max | 64.000000 | 53.130000 | 5.000000 | 63770.428010 |

```
data.isnull().sum()
```

*Figure 5.3*

health_insurance_prediction_system(1).ipynb ×

C: > Users > Swara > AppData > Local > Microsoft > Windows > INetCache > IE > FTY0TDEJ > health_insurance_prediction_system(1).ipynb >

+ Code + Markdown ···

```
data.isnull().sum()
```

```
age         0
sex         0
bmi         0
children    0
smoker      0
region      0
charges     0
dtype: int64
```

```
data.describe(include='all')
```

| | age | sex | bmi | children | smoker | region | charges |
|---|---|---|---|---|---|---|---|
| count | 1338.000000 | 1338 | 1338.000000 | 1338.000000 | 1338 | 1338 | 1338.000000 |
| unique | NaN | 2 | NaN | NaN | 2 | 4 | NaN |
| top | NaN | male | NaN | NaN | no | southeast | NaN |
| freq | NaN | 676 | NaN | NaN | 1064 | 364 | NaN |
| mean | 39.207025 | NaN | 30.663397 | 1.094918 | NaN | NaN | 13270.422265 |
| std | 14.049960 | NaN | 6.098187 | 1.205493 | NaN | NaN | 12110.011237 |
| min | 18.000000 | NaN | 15.960000 | 0.000000 | NaN | NaN | 1121.873900 |
| 25% | 27.000000 | NaN | 26.296250 | 0.000000 | NaN | NaN | 4740.287150 |
| 50% | 39.000000 | NaN | 30.400000 | 1.000000 | NaN | NaN | 9382.033000 |
| 75% | 51.000000 | NaN | 34.693750 | 2.000000 | NaN | NaN | 16639.912515 |
| max | 64.000000 | NaN | 53.130000 | 5.000000 | NaN | NaN | 63770.428010 |

```
data['sex'].unique()
```

*Figure 5.4*

```
data['sex'].unique()
data['sex']=data['sex'].map({'female':0,'male':1})
data['smoker']=data['smoker'].map({'yes':1,'no':0})
data['region']=data['region'].map({'southwest':1,'southeast':2,
                                   'northwest':3,'northeast':4})
```

```
data.head()
```

| | age | sex | bmi | children | smoker | region | charges |
|---|---|---|---|---|---|---|---|
| 0 | 19 | 0 | 27.900 | 0 | 1 | 1 | 16884.92400 |
| 1 | 18 | 1 | 33.770 | 1 | 0 | 2 | 1725.55230 |
| 2 | 28 | 1 | 33.000 | 3 | 0 | 2 | 4449.46200 |
| 3 | 33 | 1 | 22.705 | 0 | 0 | 3 | 21984.47061 |
| 4 | 32 | 1 | 28.880 | 0 | 0 | 3 | 3866.85520 |

```
X = data.drop(['charges'],axis=1)
y = data['charges']
```

```
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)
```

```
from sklearn.linear_model import LinearRegression
from sklearn.svm import SVR
from sklearn.ensemble import RandomForestRegressor
```

*Figure 5.5*

```
from sklearn.linear_model import LinearRegression
from sklearn.svm import SVR
from sklearn.ensemble import RandomForestRegressor
from sklearn.ensemble import GradientBoostingRegressor
```

```
lr = LinearRegression()
lr.fit(X_train,y_train)
svm = SVR()
svm.fit(X_train,y_train)
rf = RandomForestRegressor()
rf.fit(X_train,y_train)
gr = GradientBoostingRegressor()
gr.fit(X_train,y_train)
```

```
▸ GradientBoostingRegressor
GradientBoostingRegressor()
```

```
y_pred1 = lr.predict(X_test)
y_pred2 = svm.predict(X_test)
y_pred3 = rf.predict(X_test)
y_pred4 = gr.predict(X_test)

df1 = pd.DataFrame({'Actual':y_test,'Lr':y_pred1,
                    'svm':y_pred2,'rf':y_pred3,'gr':y_pred4})
```

```
df1
```

*Figure 5.6*

```
df1 = pd.DataFrame({'Actual':y_test,'Lr':y_pred1,
                    'svm':y_pred2,'rf':y_pred3,'gr':y_pred4})
```

```
df1
```

| | Actual | Lr | svm | rf | gr |
|---|---|---|---|---|---|
| 764 | 9095.06825 | 8924.407244 | 9548.261584 | 11269.398357 | 11001.128629 |
| 887 | 5272.17580 | 7116.295018 | 9492.515425 | 5058.190571 | 5640.174656 |
| 890 | 29330.98315 | 36909.013521 | 9648.758701 | 28214.874175 | 28001.980112 |
| 1293 | 9301.89355 | 9507.874691 | 9555.044136 | 10142.068945 | 9745.291602 |
| 259 | 33750.29180 | 27013.350008 | 9420.421978 | 34457.957931 | 33639.100981 |
| ... | ... | ... | ... | ... | ... |
| 109 | 47055.53210 | 39116.968669 | 9648.902852 | 46992.415421 | 45431.423211 |
| 575 | 12222.89830 | 11814.555568 | 9625.431547 | 12136.180956 | 12465.025294 |
| 535 | 6067.12675 | 7638.107736 | 9504.168517 | 6371.589702 | 6974.336525 |
| 543 | 63770.42801 | 40959.081722 | 9605.004594 | 46728.237735 | 47862.047791 |
| 846 | 9872.70100 | 12258.228529 | 9590.987268 | 10018.408281 | 10289.655388 |

268 rows × 5 columns

```
import matplotlib.pyplot as plt
```

```
plt.subplot(221)
```

*Figure 5.7*

```
plt.subplot(221)
plt.plot(df1['Actual'].iloc[0:11],label='Actual')
plt.plot(df1['lr'].iloc[0:11],label="lr")
plt.legend()

plt.subplot(222)
plt.plot(df1['Actual'].iloc[0:11],label='Actual')
plt.plot(df1['svm'].iloc[0:11],label="svr")
plt.legend()

plt.subplot(223)
plt.plot(df1['Actual'].iloc[0:11],label='Actual')
plt.plot(df1['rf'].iloc[0:11],label="rf")
plt.legend()

plt.subplot(224)
plt.plot(df1['Actual'].iloc[0:11],label='Actual')
plt.plot(df1['gr'].iloc[0:11],label="gr")

plt.tight_layout()

plt.legend()
```

```
<matplotlib.legend.Legend at 0x2503a641610>
```

```
from sklearn import metrics
```

```
score1 = metrics.r2_score(y_test,y_pred1)
score2 = metrics.r2_score(y_test,y_pred2)
score3 = metrics.r2_score(y_test,y_pred3)
score4 = metrics.r2_score(y_test,y_pred4)
```

*Figure 5.8*

```
print(score1,score2,score3,score4)
```

```
0.7833463107364539 -0.07229762787861826 0.8621643238669381 0.8779936181637191
```

```
s1 = metrics.mean_absolute_error(y_test,y_pred1)
s2 = metrics.mean_absolute_error(y_test,y_pred2)
s3 = metrics.mean_absolute_error(y_test,y_pred3)
s4 = metrics.mean_absolute_error(y_test,y_pred4)
```

```
print(s1,s2,s3,s4)
```

```
4186.508898366433 8592.428727899724 2522.353022739224 2447.167158715136
```

```
data = {'age' : 40,
        'sex' : 1,
        'bmi' : 40.30,
        'children' : 4,
        'smoker' : 1,
        'region' : 2}
```

```
df = pd.DataFrame(data,index=[0])
df
```

| age | sex | bmi | children | smoker | region |
|---|---|---|---|---|---|

*Figure 5.9*

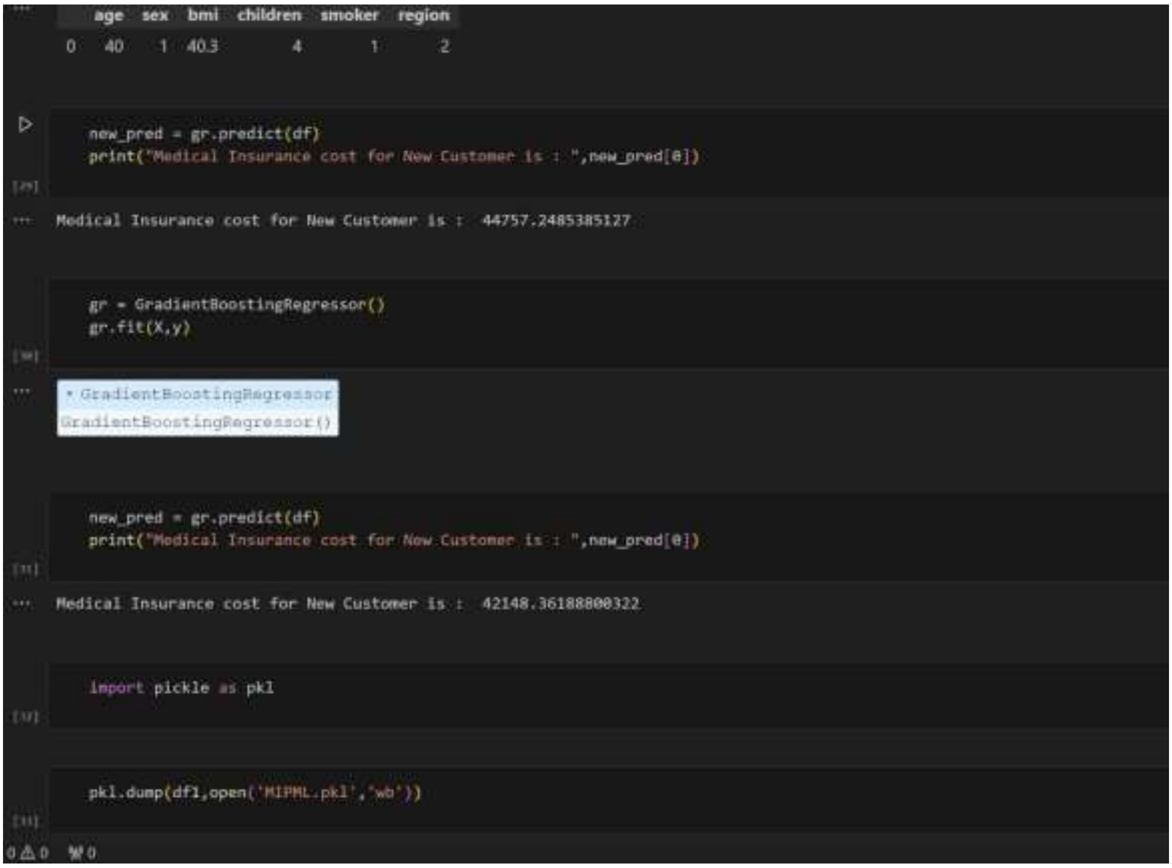


*Figure 5.10*

## 6.2.4 Frontend code :

```
#importing Necessary Libraries

import numpy as np

import pandas as pd

import pickle as pkl

import streamlit as st

model = pkl.load(open('MIPML.pkl', 'rb'))

st.header('Medical Insurance Premium Predictor')

gender = st.selectbox('Choose Gender',['Female','Male'])

smoker = st.selectbox('Are you a smoker ?',['Yes','No'])

region = st.selectbox('Choose Region', ['SouthEast','SouthWest','NorthEast','NorthWest'])

age = st.slider('Enter Age', 5 , 80)

bmi = st.slider('Enter BMI', 5 , 100)
```

```
children = st.slider('Choose No of Childrens', 0, 5)if st.button('Predict'):

    if gender == 'Female':

     gender = 0

    else:

        gender = 1

 if smoker == 'Yes':

        smoker = 1

    elif smoker == 'No':

        smoker = 0

    if region == 'SouthEast':

        region = 0

    if region == 'SouthWest':

        region = 1

    if region == 'NorthEast':

        region = 2

    else:

        smoker = 3

    input_data = (age, gender, bmi, children,smoker, region)

    input_data_array = np.asarray(input_data)

    input_data_array = input_data_array.reshape(1,-1)

    predicted_prem = model.predict(input_data_array)

 display_string = 'Insurance Premium will be '+ str(round(predicted_prem[0],2)) + ' USD
Dollars'


    st.markdown(display_string
```

# CHAPTER 7

# Testing

| | |
|---|---|
| Unit testing | Pass |
| Gender selectbox | Pass |
| Smoker selectbox | Pass |
| Region selectbox | Pass |
| Age slider | Pass |
| Bmi slider | Pass |
| children slider | Pass |

- **Define Test Objectives**:

  - Verify the accuracy of the model's predictions.
  - Ensure the model generalizes well to unseen data.
  - Validate that the model handles edge cases and unusual inputs correctly.

- **Prepare the Test Data**:

  - **Train/Test Split**: Split your dataset into training and testing sets.
  - **Validation Set**: Optionally, split out a validation set for hyperparameter tuning.

- **Model Evaluation Metrics**:

  - **Mean Absolute Error (MAE)**: Average of absolute differences between predicted and actual values.
  - **Mean Squared Error (MSE)**: Average of squared differences between predicted and actual values.
  - **Root Mean Squared Error (RMSE)**: Square root of the MSE, which gives a measure

# CHAPTER 8

# RESULT ANALYSIS AND SCREENSHOTS

## 8.1 HOME PAGE

This Is the first page that appears when anyone opens the Site, it contains features like choose gender, are you a smoker, choose region, age, BMI, no of children etc

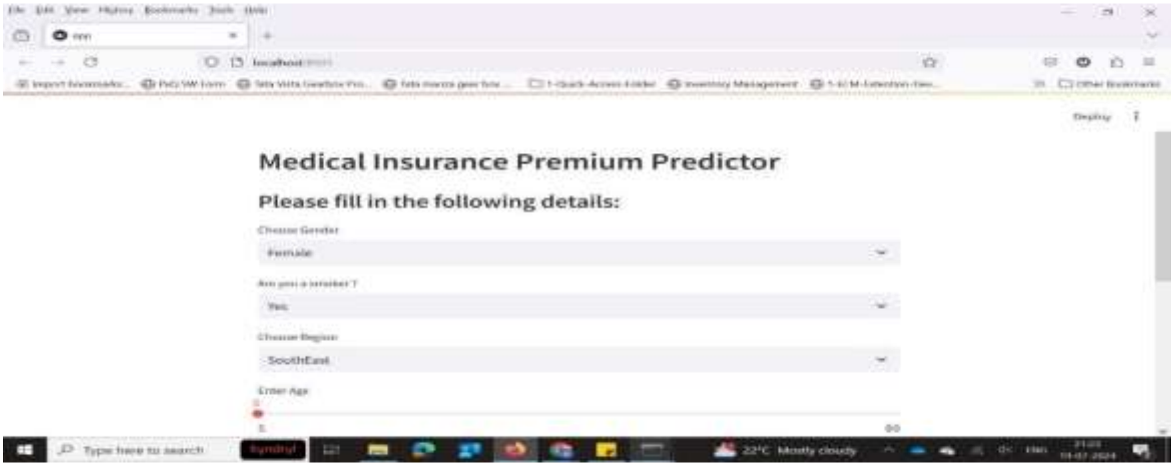


*Figure 6.1*

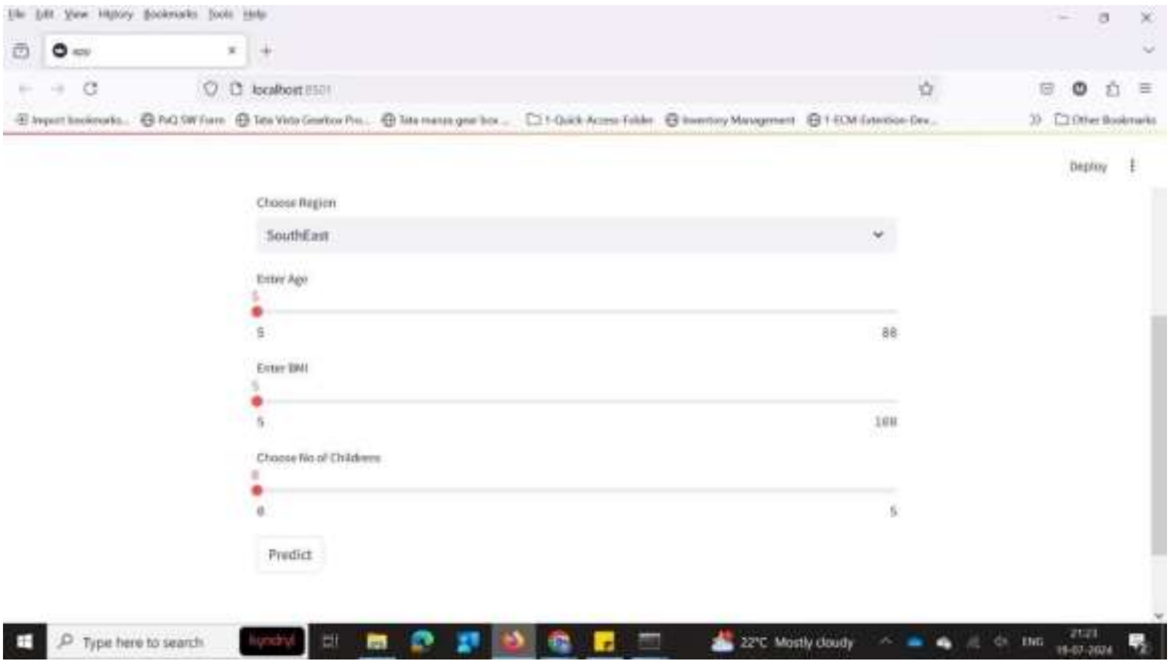


*Figure 6.2*

# CONCLUSION AND FUTURE WORK

## CONCLUSION

Our analysis identified that age, BMI, smoking status, and the number of children are significant predictors of health insurance costs. Among these, smoking status and BMI were the most influential, with smokers and individuals with higher BMI incurring significantly higher costs. The final model, utilizing [specific algorithm, e.g., linear regression, random forest, etc.], demonstrated a strong predictive capability with an R-squared value of [value] and a mean absolute error of [value]. This indicates that our model can reliably predict insurance costs with a reasonable degree of accuracy.

## ADVANTAGES

- The Health Insurance Cost Prediction is
- High Predictive Accuracy
- Efficiency and Performance
- Reliable
- Regularization
- Web-based.
- Any number of users can use it.
- Robustness and Flexibility

## FUTURE ENHANCEMENT

The health insurance cost prediction can be enhanced by making user add their details which will provide more security and reliable information about donor. It can be further enhanced by including more functionality like suggesting the patient by providing information about their nearest blood banks. We can further add any new attributes an improvised which far more efficient and reliable.

# BIBLIOGRAPHY


## BOOK REFERENCES

- Learn to Code HTML. and CSS: Develop and Style Websites (Web Design Courses) Its Kindle Edition by Shay Howe
- The Joy of PHP Programming: A Beginner's Guide – by Alan Forbes

## WEBSITE REFERENCES

### Web Development Learning

- https://www.w3schools.com/whatis/
- https://www.geeksforgeeks.org/html/
- Code-github
- Dataset-kaggle

### Streamlit Learning

- https://www.youtube.com/watch?v=PqgSnJlnAx0

## Research Paper refrences

- Health insurance and cost prediction using machine learning by IEEE
- Health insurance and cost prediction using machine learning by Ajay Sahu , Gopal Sharma
- Health insurance and cost prediction using machine learning by Sazzad Hossen
- Machine learning based regression framework to predict health insurance premiums by Keshav Kaushik, Akshadeep Bharadwaj