**SUMMER TRAINING/INTERNSHIP**

**PROJECT REPORT**

(Term June-July 2025)

## *Medical Premium Insurance Prediction*

Submitted by

**ANSHIKA DUBEY**

**Registration Number :12316623**

**Machine Learning Made Easy: From Basics To AI**

**Course Code PETV79**

Under the Guidance of

MAHIPAL SINGH

# 

# School of Computer Science and Engineering

**CERTIFICATE**

This is to certify that Ms. Anshika Dubey, Registration No.- 12316623 a student of B.Tech in Computer Science and Engineering, has successfully completed her Summer Training / Internship Project titled “Medical Insurance Premium Prediction Using Machine Learning” during the period June 2025 to July 2025 under my supervision.

This project has been carried out in partial fulfilment of the requirements of the Summer Internship Program as prescribed by Lovely Professional University. To the best of my knowledge, the work presented in this report is original and has not been submitted elsewhere for the award of any degree or diploma.

**School of Computer Science and Engineering**

Lovely Professional University

Phagwara, Punjab.

**ACKNOWLEDGEMENT**

I am sincerely thankful to the Lovely Professional University for offering me the opportunity to undertake this Summer Training / Internship Program during which I was able to apply my academic knowledge to a real-world machine learning problem.

I express my deep gratitude to my training/project mentor, Mahipal Singh , for their support, expert guidance, and constant encouragement throughout the project titled “Medical Insurance Premium Prediction Using Machine Learning”.

I would also like to thank all the members of the organization/institute for providing a positive and learning-friendly environment**.**

**Anshika Dubey (12316623)**

**TABLE OF CONTENTS**

**1.** **CHAPTER 1: INTRODUCTION**

1.1 Overview

1.2 Objective of the Project

**2. CHAPTER 2: TRAINING OVERVIEW**

2.1 Tools & Technologies used

2.2 Dataset Description

2.3 Areas covered during Training

2.4 Daily work summary

**3.** **CHAPTER 3: PROJECT DETAILS**

3.1 Title of the project

3.2 Problem definition

3.3 Scope and objectives

3.4System Requirements

3.5 Architecture Diagram

3.6 Data flow

**4. CHAPTER 3: IMPLEMENTATION**

4.1 Methodology

4.2 Modules (screenshots)

4.3 Code snippet (screenshots)

**5. CHAPTER 5: RESULTS AND DISCUSSION**

5.1 Output

5.2 Model Evaluation & Results

5.3 Comparative Analysis

5.4 Challenges Faced

5.5 learnings

**6. CHAPTER 6: CONCLUSION**

6.1 Summary

6.2 Future work

**7. CHAPTER 7: MODEL VALIDATION USING SYNTHETIC DATASET**

7.1 Purpose

7.2 Model Training and Evaluation

7.3 Results

7.4 Interpretation

7.5 Deployment

**CHAPTER 1: INTRODUCTION**

**1.1 OVERVIEW**

The field of Machine Learning (ML) has transformed the way predictive analytics is applied across industries, and the insurance sector is no exception. With the explosion of accessible data, the need for efficient, accurate, and scalable predictive systems has grown significantly. Predicting medical insurance premiums is a critical application area within the insurance domain, allowing companies to estimate costs, manage risks, and personalize offerings based on individual characteristics.

In this project, the domain explored was supervised machine learning with a focus on regression-based predictive modeling. The application scenario was centered around a real-world dataset representing various personal attributes (such as age, gender, BMI, number of children, smoking habits, and region) and the corresponding insurance charges. By applying ML models to this dataset, the goal was to accurately predict medical insurance premiums for new individuals, thereby simulating an operational decision-support tool for insurance companies.

**1.2Objective of the Project**

The primary objective of this project was to:

* Analyze and understand the dataset associated with medical insurance premiums.
* Train and evaluate three different machine learning models:
  + Linear Regression (LR)
  + Random Forest Regressor (RF)
  + Extreme Gradient Boosting (XGBoost)
* Compare the performance of these models using standard evaluation metrics such as R² Score, Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).
* Identify the most influential features that affect insurance premium costs.
* Provide insights and interpretations from the trained models to assist in better policy planning and premium estimation.

This project serves as a foundational learning experience in implementing regression models, understanding data preprocessing pipelines, feature engineering, model training and evaluation, and deriving actionable insights from machine learning workflows.

**CHAPTER 2: TRAINING OVERVIEW**

**2.1 Tools and Technologies used**

To develop, train, and evaluate the insurance premium prediction models, the following tools and technologies were used:

|  |  |
| --- | --- |
| Tool / Technology | Description |
| Python 3.x | Programming language used for data analysis and machine learning model development. |
| Jupyter Notebook | Interactive environment for writing and running Python code, visualizing results, and documenting the workflow. |
| Pandas | Data manipulation and analysis library used for handling structured data. |
| NumPy | Library for numerical computations and array operations. |
| Matplotlib & Seaborn | Libraries used for data visualization and plotting graphs/charts. |
| Scikit-learn | Machine learning library used for implementing Linear Regression and Random Forest models, as well as preprocessing utilities. |
| XGBoost | High-performance gradient boosting library used for implementing the XGBoost model. |

**2.2 Dataset Description**

The dataset used in this project is named insurance.csv. It contains 1,338 records, each representing an individual's demographic and health information, along with the corresponding insurance charges.

**Features in the Dataset**

|  |  |
| --- | --- |
| **Column Name** | **Description** |
| age | Age of the individual (in years) |
| sex | Gender (male, female) |
| bmi | Body Mass Index (float) |
| children | Number of children covered under the insurance |
| smoker | Smoking status (yes, no) |
| region | Residential region in the U.S. (southeast, southwest, etc.) |
| charges | Target variable - Insurance premium (float, in USD) |

**2.3Areas Covered During Training**

The training phase covered a range of foundational and advanced topics in machine learning and data science, particularly within the regression context. Key areas explored include:

**2.3.1. Exploratory Data Analysis (EDA)**

EDA refers to the process of visually and statistically analyzing datasets to understand distributions, trends, relationships, and outliers before applying machine learning. It helps in making informed decisions about feature selection and preprocessing steps.

**In project:**

* We plotted histograms and scatter plots for variables like age, bmi, and charges.
* Observed that smoking status and BMI had strong effects on charges.
* Used correlation heatmaps to identify multicollinearity.

**2.3.2 Feature Engineering and Encoding**

Feature engineering is the process of transforming raw data into meaningful features that enhance a machine learning model’s ability to learn. It involves selecting, modifying, or creating new features to improve model performance. Good feature engineering helps the model better understand patterns and relationships in the data. Common techniques include encoding categorical variables (e.g., converting "smoker" status into 0 or 1), scaling numerical values (like age and bmi), and creating derived features if needed. In this project, we applied One-Hot Encoding to categorical columns like sex, region, and smoker, and performed standardization for numerical features to ensure consistency across all input variables. These steps made the dataset compatible with different machine learning algorithms and improved overall prediction accuracy**.**

**In project:**

* Applied One-Hot Encoding for categorical features like sex, region, and smoker.
* Standardized numeric features like age, bmi, and children for algorithms sensitive to scale.

**2.3.3. Model Training**

Model training involves feeding processed data into algorithms to learn patterns and make predictions.

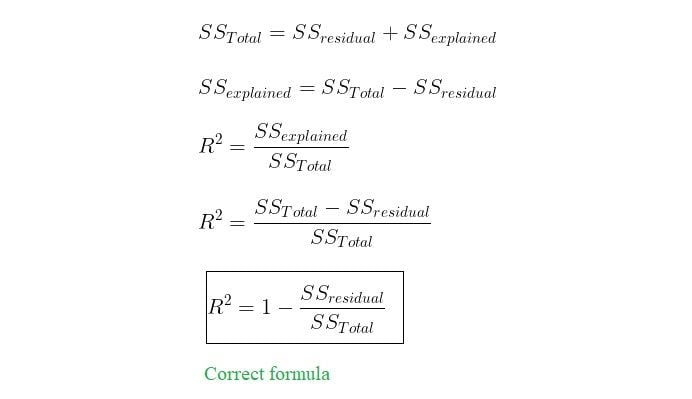
**Models Used:**

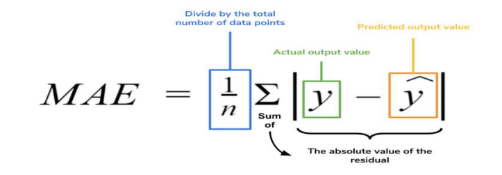
* **Linear Regression:** In [statistics](https://en.wikipedia.org/wiki/Statistics), linear regression is a [model](https://en.wikipedia.org/wiki/Statistical_model) that estimates the relationship between a [scalar](https://en.wikipedia.org/wiki/Scalar_(mathematics)) response ([dependent variable](https://en.wikipedia.org/wiki/Dependent_variable)) and one or more explanatory variables (regressor or [independent variable](https://en.wikipedia.org/wiki/Independent_variable)). A model with exactly one explanatory variable is a [simple linear regression](https://en.wikipedia.org/wiki/Simple_linear_regression); a model with two or more explanatory variables is a multiple linear regression. This term is distinct from [multivariate linear regression](https://en.wikipedia.org/wiki/Multivariate_linear_regression), which predicts multiple [correlated](https://en.wikipedia.org/wiki/Correlated) dependent variables rather than a single dependent variable.
* **Random Forest Regressor:** Random Forest is a popular ensemble learning algorithm used for both classification and regression tasks. It works by building multiple decision trees during training and combining their outputs to produce more accurate and stable predictions. In regression problems, the final prediction is the average of all tree outputs. Random Forest reduces the risk of overfitting by introducing randomness in both data selection (through bootstrapping) and feature selection at each split. It handles non-linear relationships well and is capable of managing both numerical and categorical data. Because of its high accuracy and robustness, Random Forest is widely used in real-world applications such as insurance, finance, and healthcare.
* **XGBoost Regressor:** XGBoost (Extreme Gradient Boosting) is an advanced and efficient implementation of gradient boosting algorithms. It builds decision trees sequentially, where each new tree focuses on correcting the errors made by the previous ones. XGBoost uses techniques like regularization, shrinkage, and column sampling to reduce overfitting and improve generalization. It is known for its high speed, accuracy, and scalability, making it a top choice in many data science competitions and real-world applications. XGBoost is especially effective in handling structured/tabular data, and it supports missing value handling and parallel processing. In this project, it produced the most accurate insurance premium predictions among all models used.

**2.3.4. Hyperparameter Tuning**

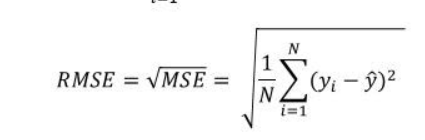
Hyperparameter tuning is the process of selecting the best combination of configuration settings (called hyperparameters) that control how a machine learning model learns from data. Unlike model parameters (which are learned during training), hyperparameters are set before training and can significantly affect a model’s performance. Examples include the number of trees (n\_estimators) and maximum tree depth (max\_depth) in Random Forest or XGBoost. To find the optimal values, methods like Grid Search, Random Search, or Bayesian Optimization are used. In this project, GridSearchCV was applied to systematically test different combinations of hyperparameters for the Random Forest model, helping to improve its prediction accuracy.

**Performance Metrics Explained:**

* **R² Score (Coefficient of Determination):** Measures how well predictions match actual values. A value closer to 1 indicates better fit.

* **MAE (Mean Absolute Error):** The average of absolute differences between predicted and actual values. Lower is better.

* **RMSE (Root Mean Squared Error):** The square root of the average squared differences. Penalizes large errors more heavily. Lower is better.



**2.3.5 Model Evaluation**

Evaluation is the step where we compare actual vs predicted values to assess model accuracy and generalization.

**In project:**

* Used test data to validate model predictions.
* Visualized predicted vs actual charges.
* Compared performance metrics of all three models.

**2.3.6. Deployment**

Deployment refers to taking a trained model and integrating it into an interface or application for real-world usage.

**In project:**

* Created a Streamlit app where users can enter personal info and get premium predictions.
* Deployed the model locally for demonstration.

**2.4 Daily/Weekly Work Summary**

|  |  |
| --- | --- |
| **Days** | **Tasks Completed** |
| Day 1 | Collected and understood the dataset; performed initial cleaning and exploratory analysis. |
| Day 2 | Preprocessed data and engineered features; built and evaluated Linear Regression model. |
| Day 3-4 | Implemented and tested Random Forest model with basic hyperparameter tuning. |
| Day 5-6 | Integrated XGBoost model; performed model comparison and evaluation. |
| Day 7-8 | Interpreted results, generated visualizations, and documented insights. |

**CHAPTER 3: PROJECT DETAILS**

**3.1 Title of the Project**

Medical Insurance Premium Prediction using Machine Learning Models

**3.2 Problem Definition**

In the modern insurance industry, accurate premium estimation is a vital task. Insurance companies determine medical premium amounts based on multiple personal and lifestyle attributes of customers. Traditional actuarial methods, while effective, may not leverage the full potential of modern data analysis and prediction techniques.

This project addresses the problem of **predicting medical insurance premiums** using **machine learning algorithms**. Given a set of features such as age, gender, BMI, number of children, smoking status, and region, the goal is to build models that accurately predict the **insurance cost** for a customer.

**3.3Scope and Objectives**

**3.3.1 Scope**

The project is confined to the use of a structured dataset from the healthcare domain that includes demographic and health-related features. The scope includes:

* Preprocessing and analyzing the dataset
* Training and evaluating multiple ML models
* Interpreting model performance and insights
* Comparing accuracy, efficiency, and generalization of models

**3.3.2 Objectives**

* Understand and clean the insurance dataset
* Train and evaluate three regression models: Linear Regression, Random Forest, and XGBoost
* Compare models using statistical performance metrics
* Identify important features influencing medical charges
* Recommend the best model for deployment based on findings

**3.4 System Requirements**

|  |  |
| --- | --- |
| **Component** | **Specification** |
| Programming Language | Python 3.x |
| IDE / Notebook | Jupyter Notebook |
| Libraries | Pandas, NumPy, Matplotlib, Seaborn, Sklearn, XGBoost, Joblib, |
| Dataset | insurance.csv (age, sex, bmi, children, smoker, region, charges) |

**3.5 ARCHITECTURE DIAGRAM**

Raw Dataset

Prediction Output

(Insurance charges)

Model Evaluation

(MAE,RMSE,R^2)

Model Training and selection

Data Preprocessing

(Cleaning,Encoding )

**3.6 DATA FLOW DIAGRAM**

**DFD – Insurance Premium Prediction System**

User

Prediction Output

ML Algorithms

(LR/RF/XGB)

Preprocessing Pipeline

Input Customer Data

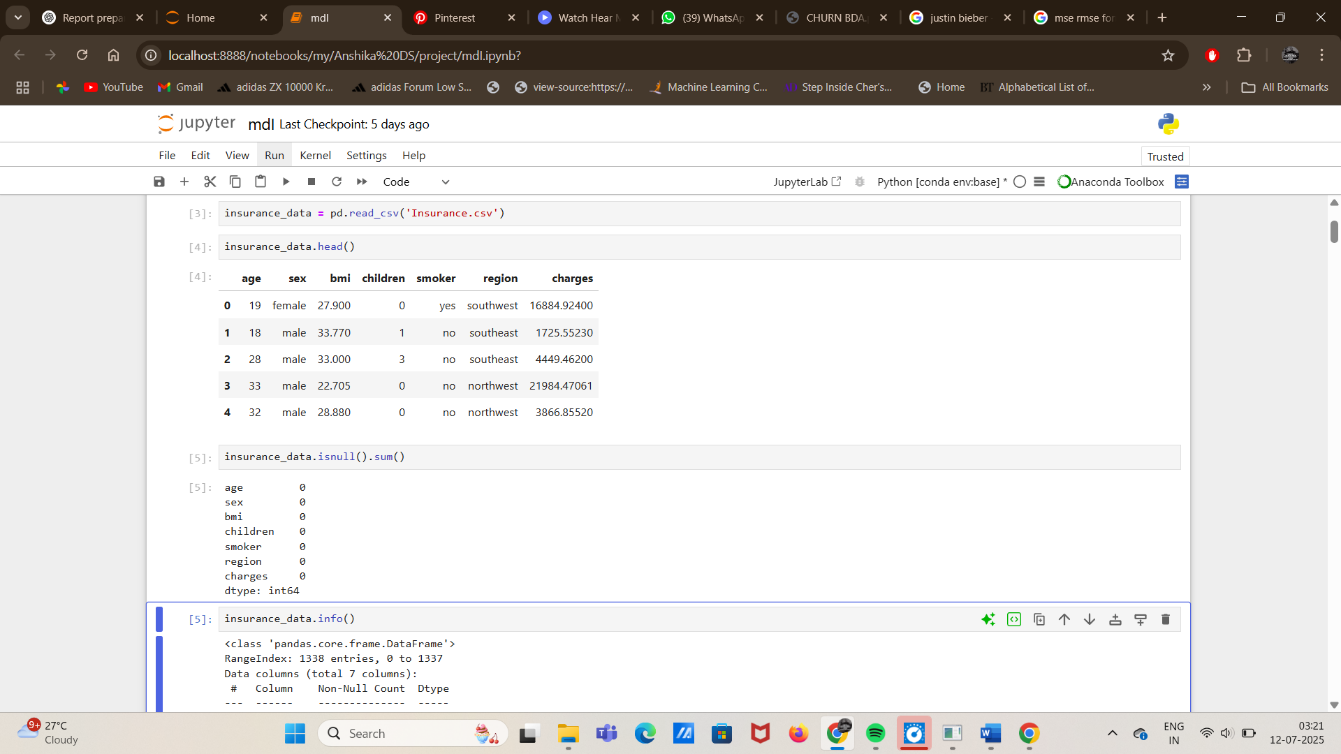
**CHAPTER 4: IMPLEMENTATION**

**4.1Methodology**

The following methodology was used in this project:

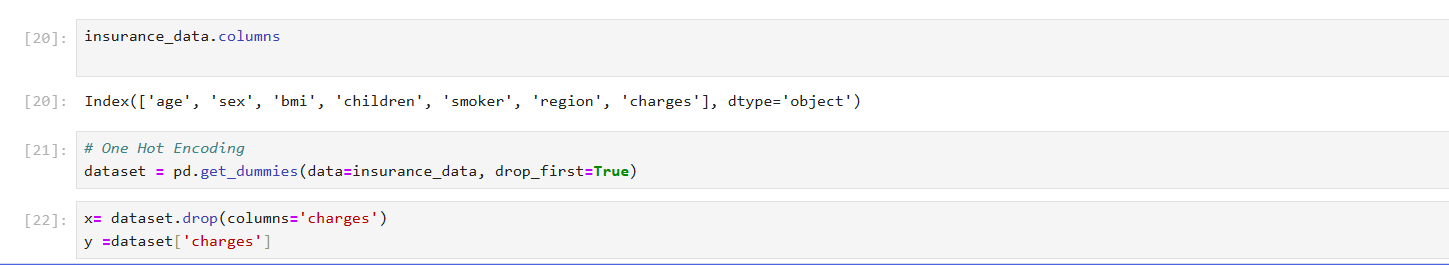
**4.1.1. Dataset Loading and Initial Exploration**

* Loaded the insurance.csv dataset containing 1,338 records,7 columns.
* Features included: age, sex, bmi, children, smoker, region, and charges (target).
* Checked for null values and basic descriptive statistics**.**

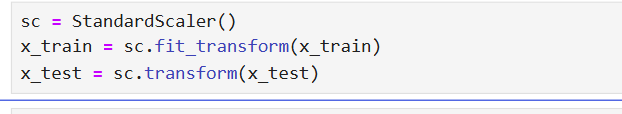
**Implementation:-**

**4.1.2. Data Preprocessing**

* **Encoding:**
  + sex, smoker, and region were categorical features.
  + Used Label Encoding or One-Hot Encoding depending on the model requirements.

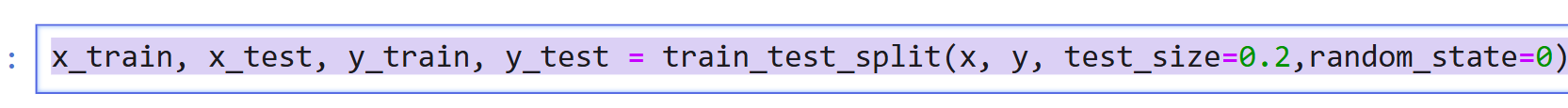
**Implementation:-**

* **Feature Scaling:**
  + Scaling was applied for models like Linear Regression to ensure all variables were on a comparable scale.

**Implementation:-**

* **Train-Test Split:**
  + Dataset split into 80% training and 20% testing data using train\_test\_split.

**Implementation:-**

****

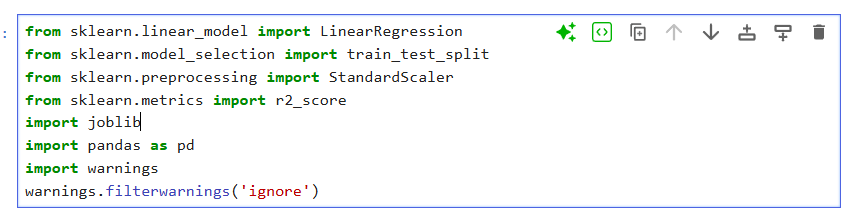
**4.1.3. Model Training**

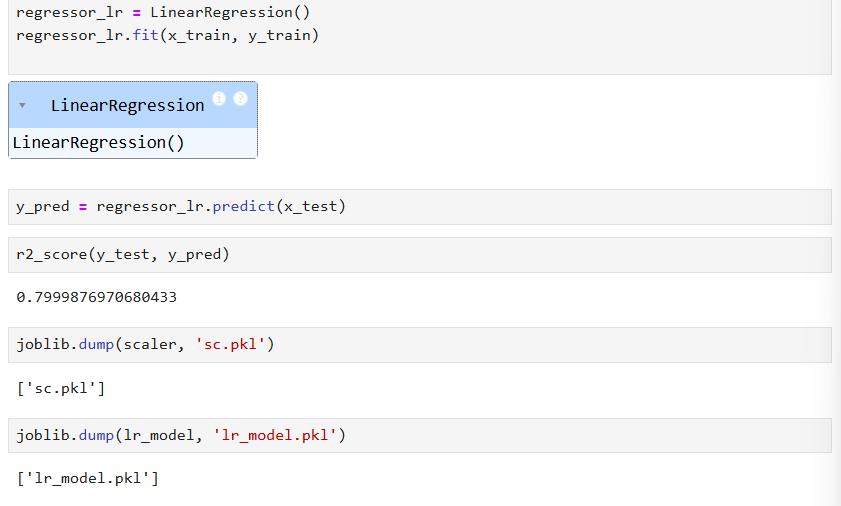
Three machine learning regression models were trained:

**a) Linear Regression**

* Used as the baseline model.
* Interpretable coefficients allowed insights into how features impact the premium.

**Implementation:-**

 **Libraries need to train the model**

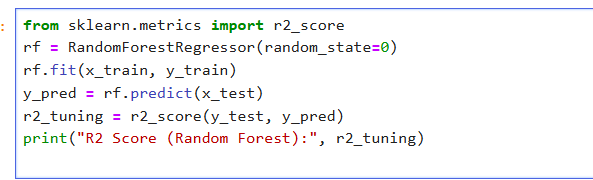
 **Training and Output of Linear Regression model**

**b) Random Forest Regressor**

* An ensemble of decision trees with bagging.
* Controlled overfitting with max\_depth, n\_estimators, etc.
* Good generalization and robustness to outliers.

**Implementation:-**

****

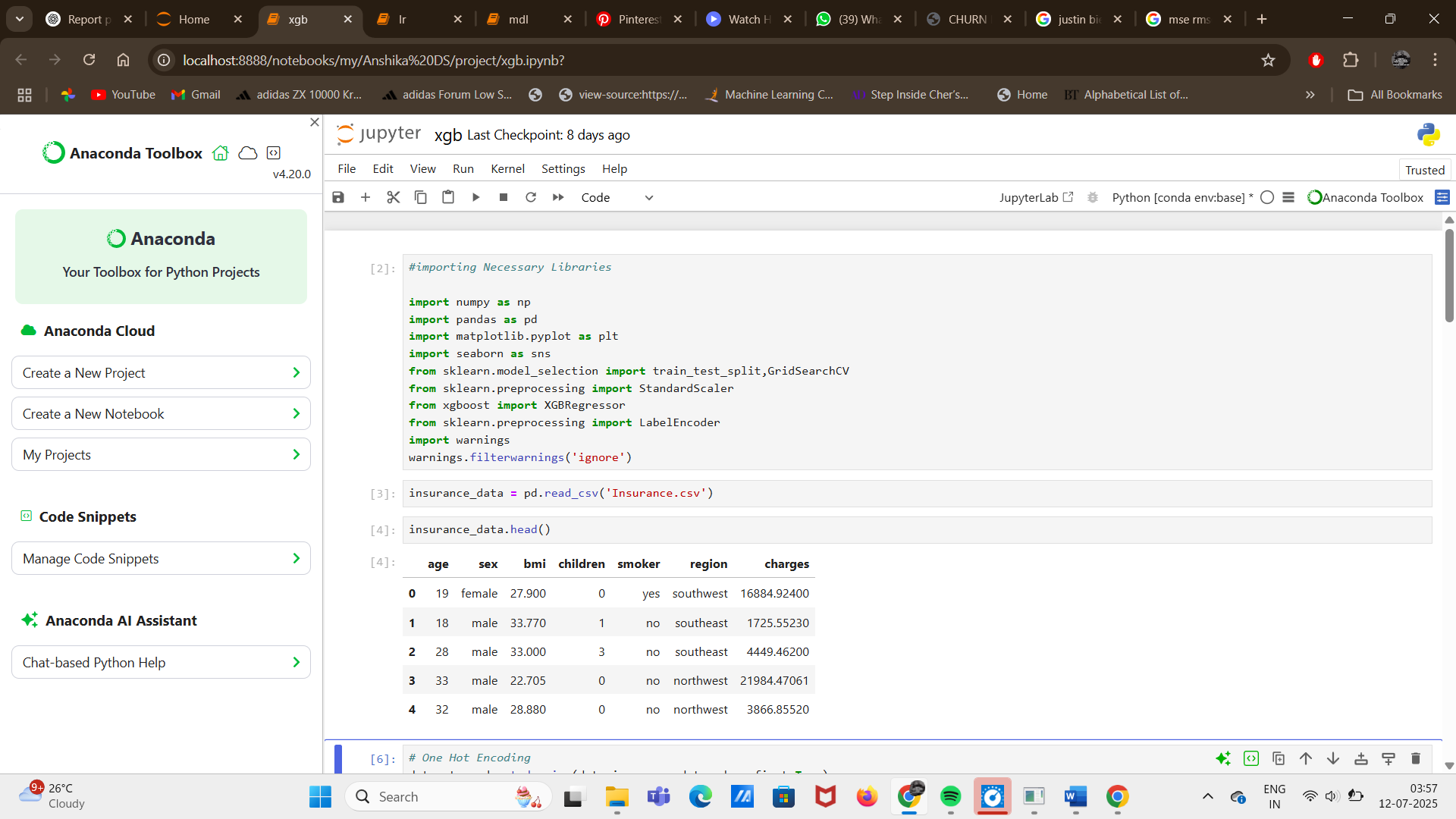
** Training Random Forest Model**

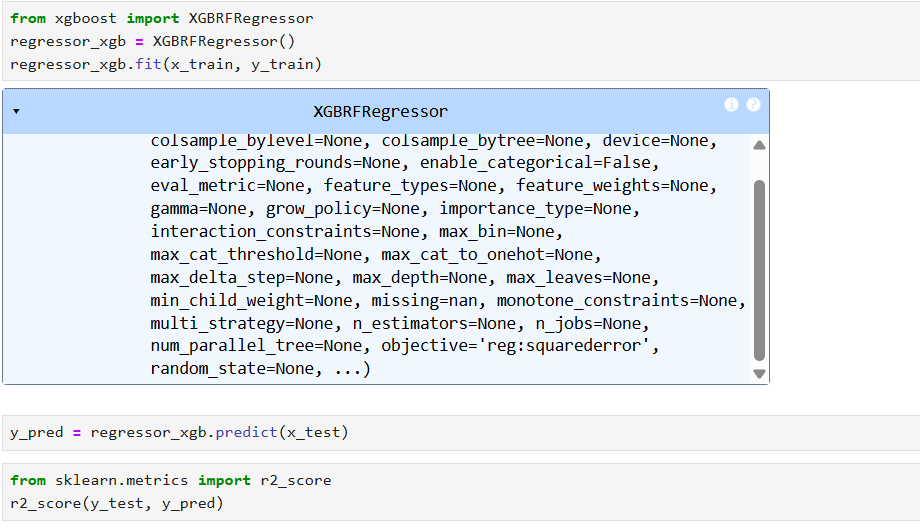
** Hyper Tuning the model using GridsearchCV**

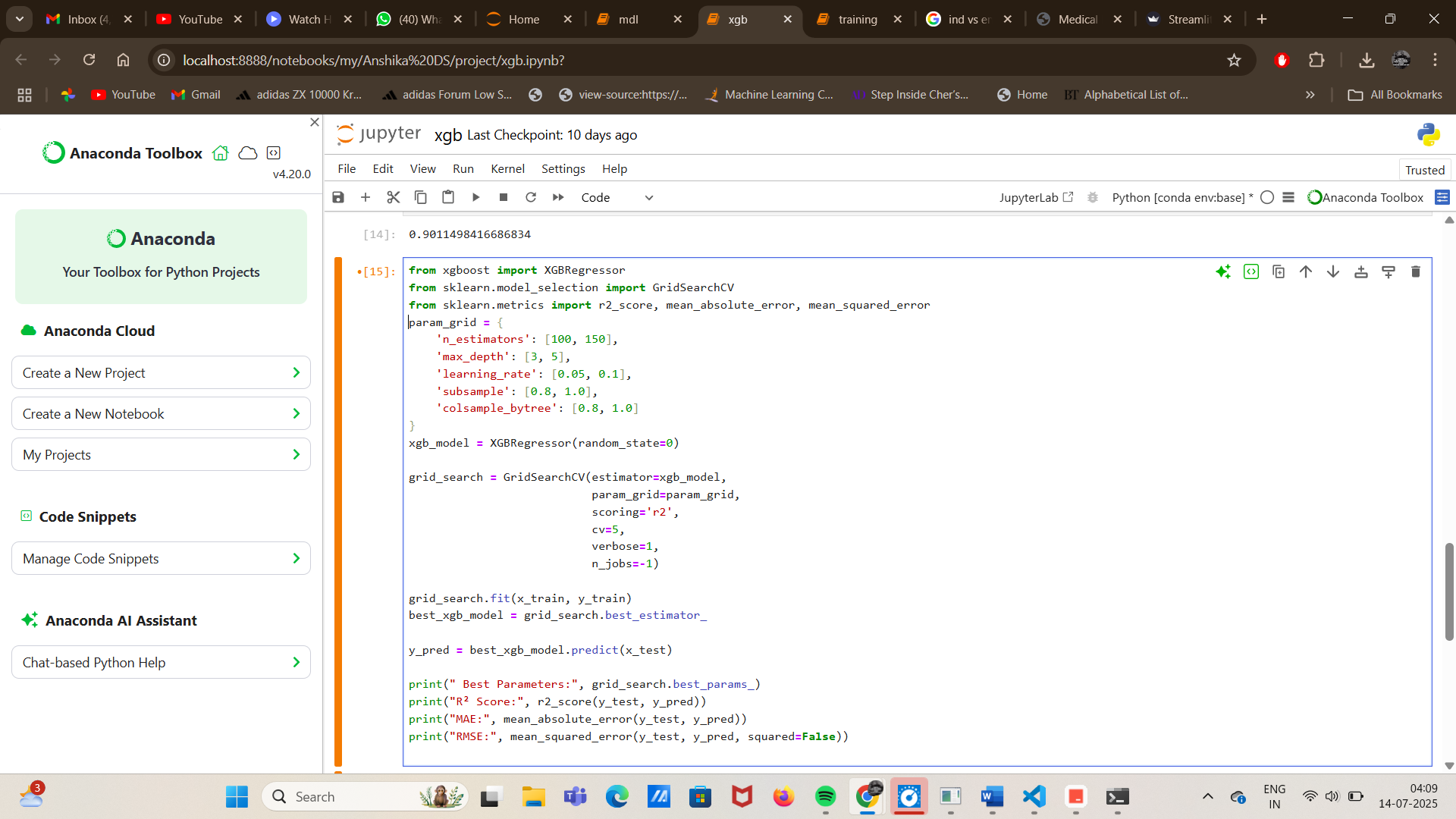
**c) XGBoost Regressor**

* Boosted tree-based model.
* Implemented gradient boosting with regularization.
* Tuning parameters like learning\_rate, n\_estimators, max\_depth led to significant performance improvement.

**Implementation:-**

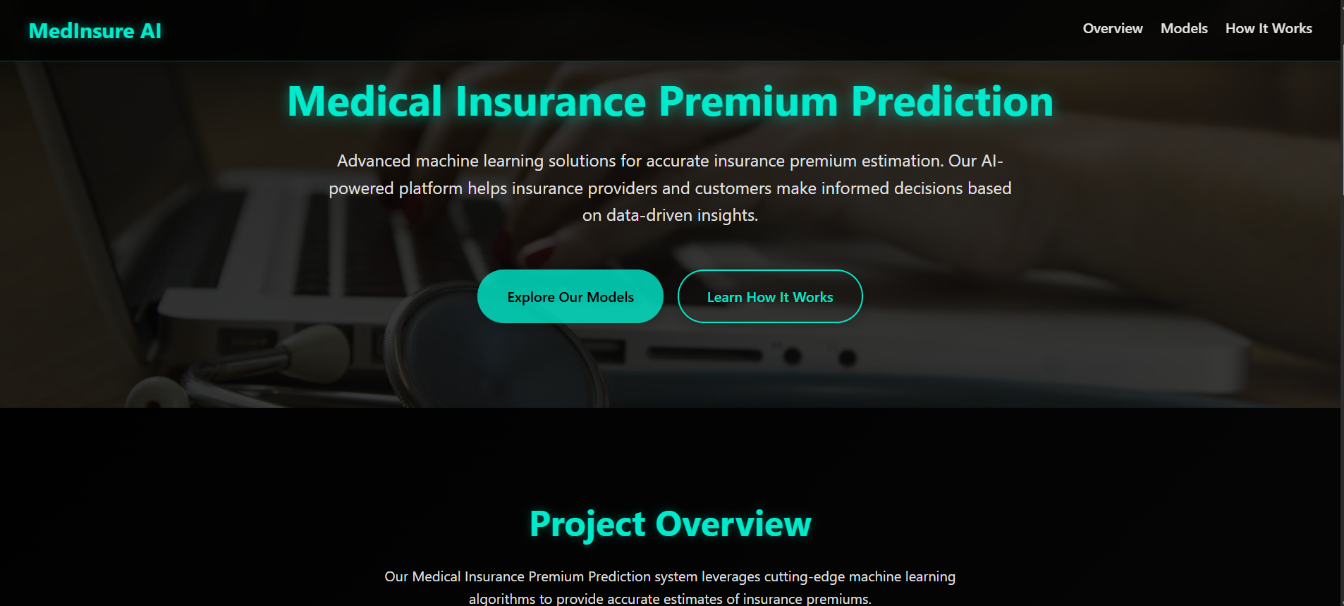
****

**** **Training the Model using XGBoost**

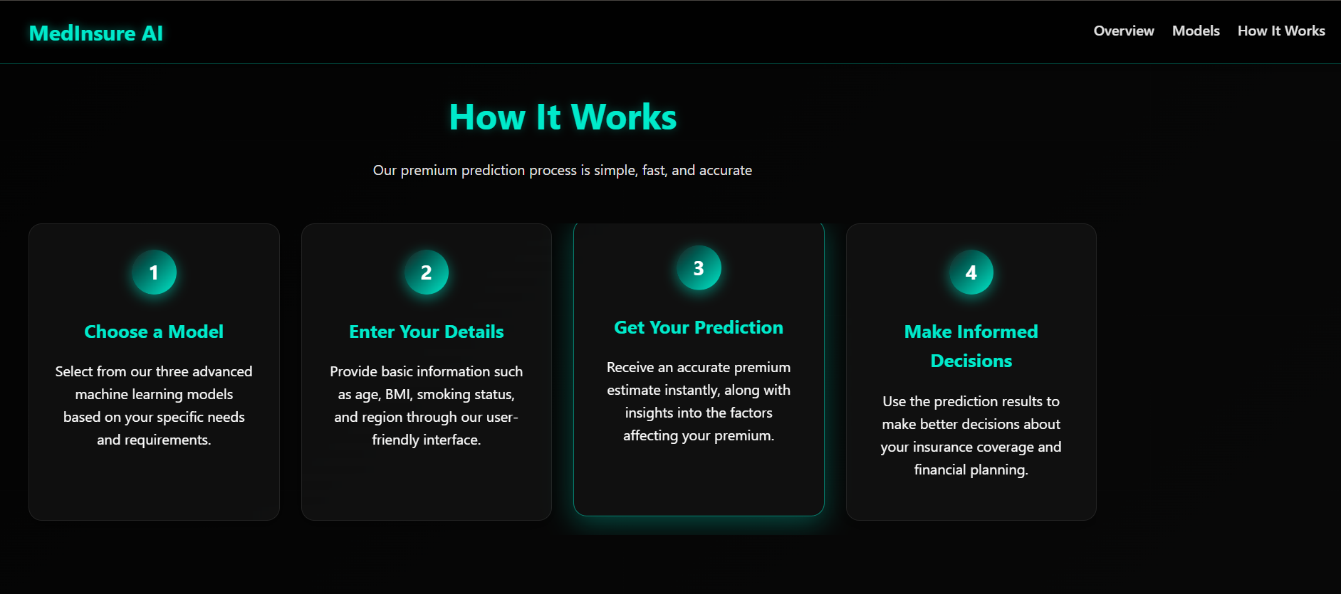
 **Hyper Tuning the model using GridSearchCv**

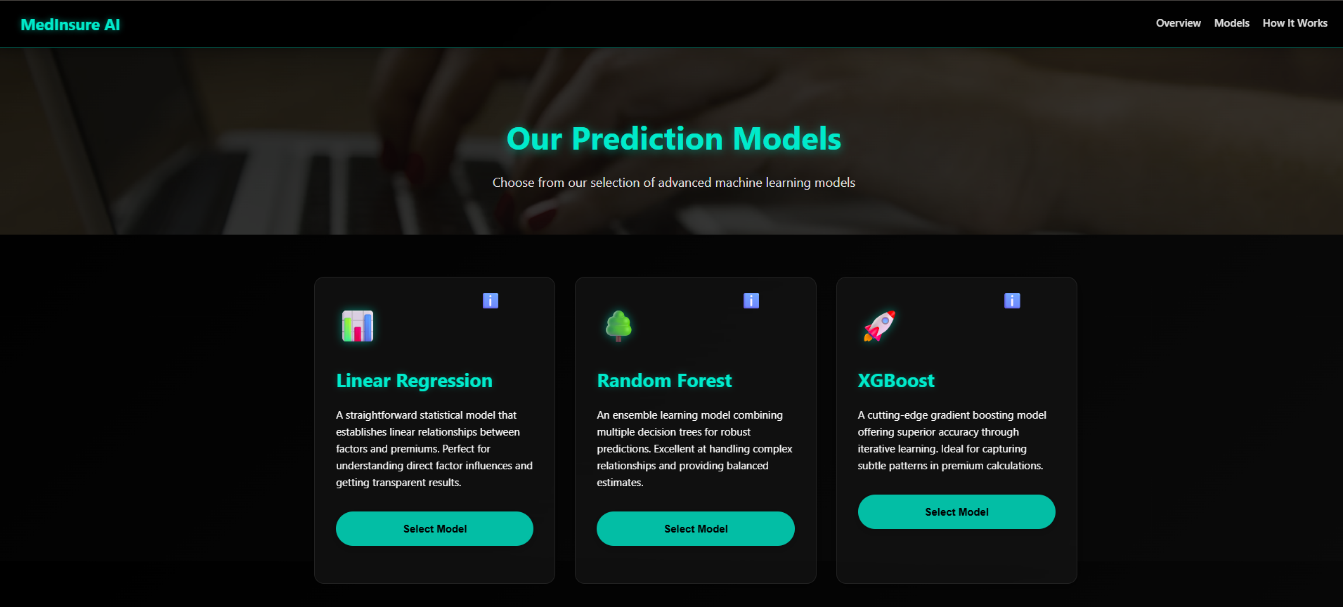
**4. Model Deployment (Streamlit Web App)**

To enhance accessibility and demonstrate real-world application, the final trained model was deployed using Streamlit, a lightweight open-source Python framework for creating interactive web applications.

* A user-friendly web interface was built using streamlit.
* Users can input feature values (e.g., age, BMI, smoker status, etc.) and receive instant predictions for insurance premiums.
* The deployed model runs inference in real-time, making it ideal for business users, customers, or insurance agents.

**(i)Landing Page of the Deployed Model**

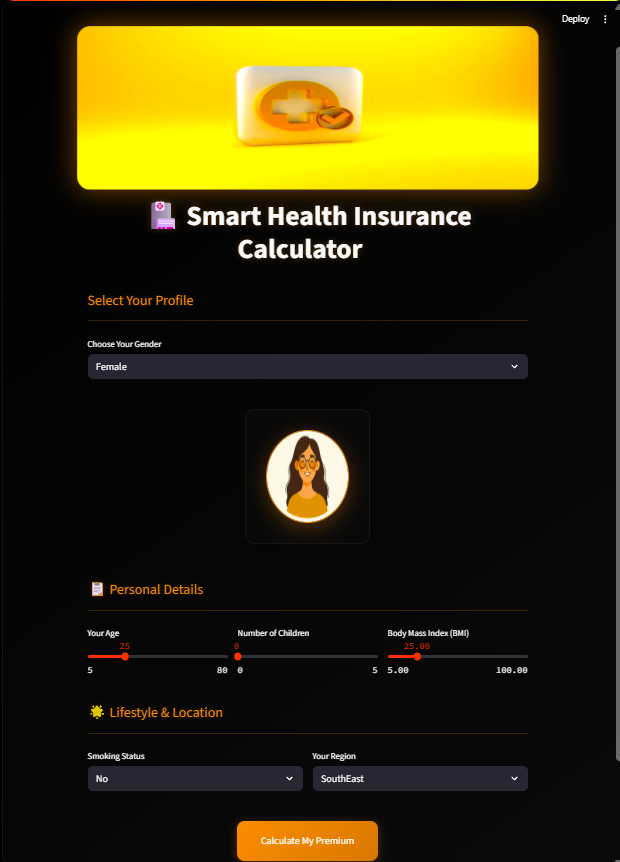
 **(ii)Here’s how the interface works**

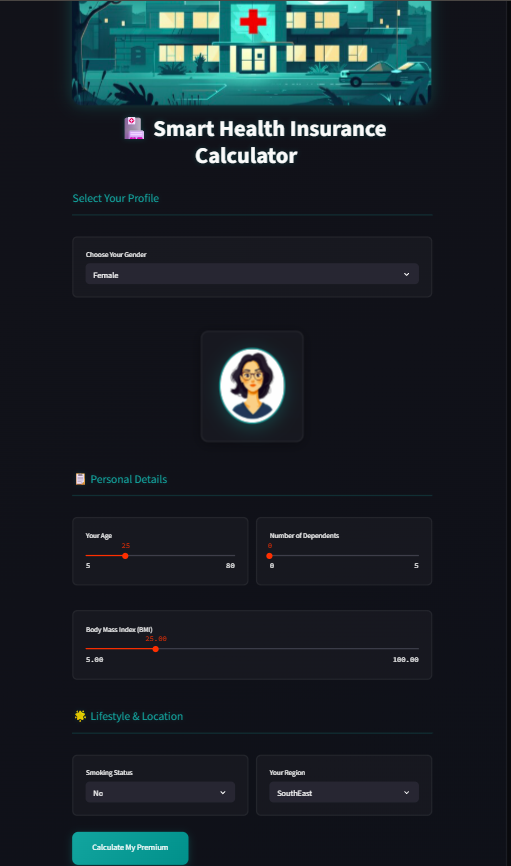
** (iii)Clicking on info icon we can get know the accuracy of each model**

**Pick any model and wait for it to initialize
 (iv) Pick any model and wait for it to initialize**

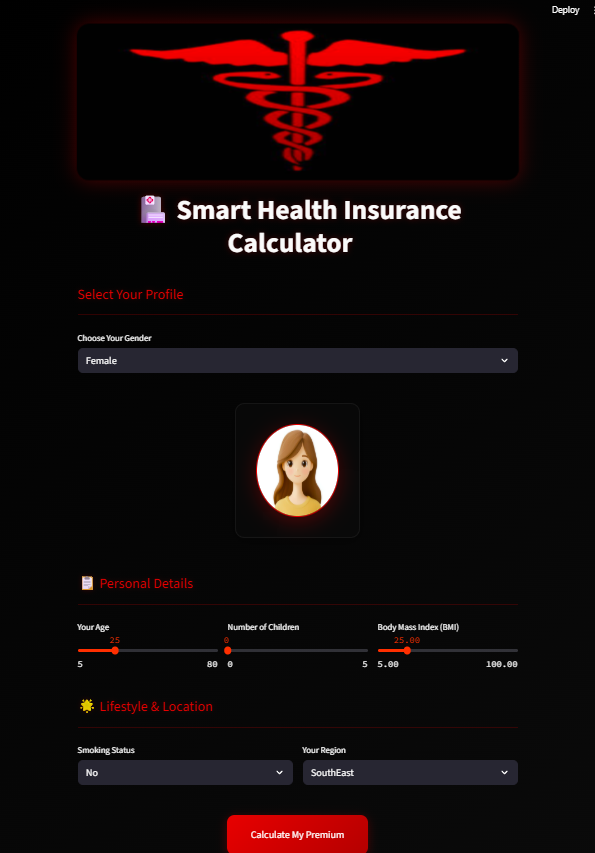
**MODELS:-**

**LINEAER REGRESSION MODEL**

****

** RANDOM FOREST MODEL**

**XGBOOST MODEL**

****

**CHAPTER 5: RESULTS AND DISCUSSION**

**5.1 Output / Report**

After successfully training the three machine learning models—**Linear Regression**, **Random Forest**, and **XGBoost**—the results were evaluated using the following metrics:

* **Mean Squared Error (MSE)**: Penalizes larger errors by squaring the differences.
* **Root Mean Squared Error (RMSE)**: Square root of MSE; interpretable in the same units as the target variable.
* **R² Score (Coefficient of Determination)**: Measures how well predictions approximate the actual values.

**5.2 Model Evaluation Results**

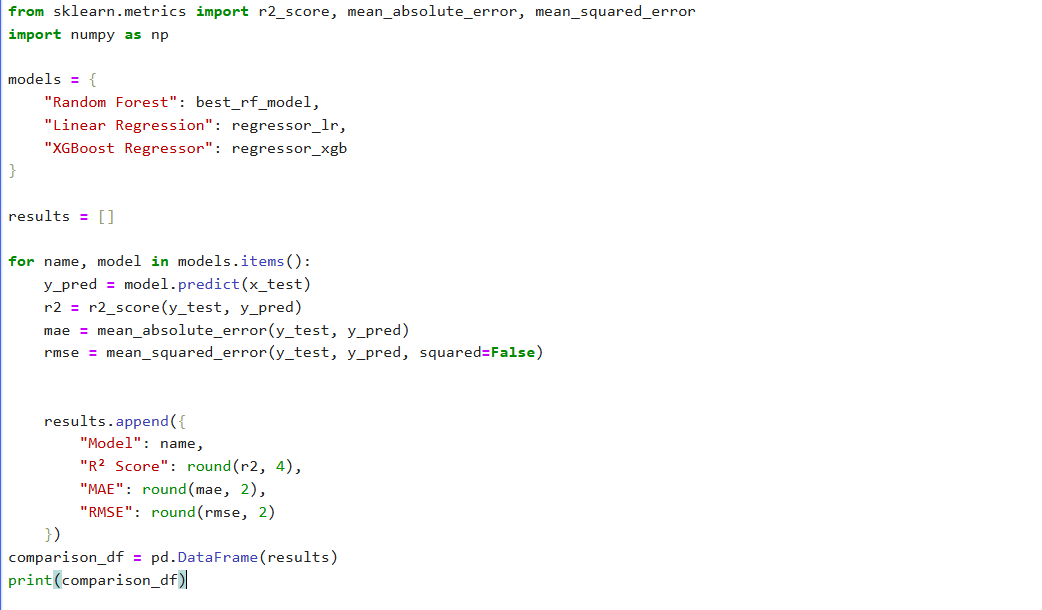
|  |  |  |  |
| --- | --- | --- | --- |
| Model | MAE | RMSE | R² Score |
| Linear Regression | 3933.27 | 5641.63 | 0.80 |
| Random Forest | 3010.29 | 4459.85 | 0.87 |
| XGBoost | 2407.73 | 3,966.11 | 0.90 |

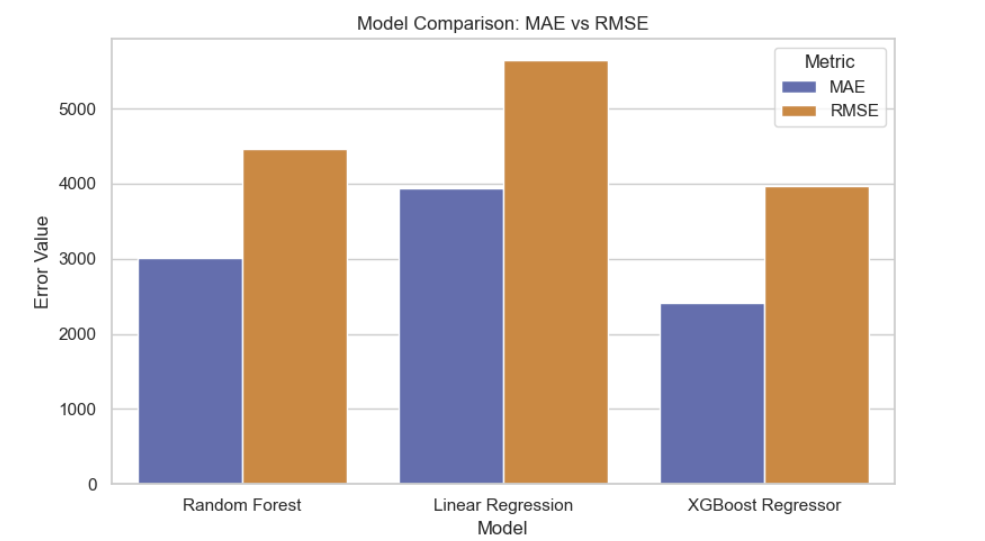
**5.3 Comparative Analysis**

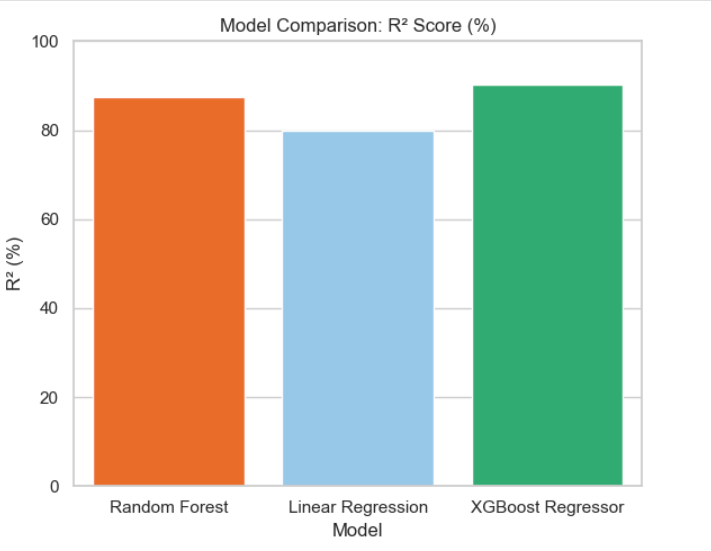
* **Linear Regression** showed decent performance but was limited by its linear assumption and was more prone to underfitting.
* **Random Forest** delivered significantly improved predictions due to its ensemble approach and ability to model nonlinear relationships.
* **XGBoost** slightly outperformed Random Forest in terms of RMSE and R², making it the **best-performing model** overall.

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **Linear Regression** | **Random Forest** | **XGBoost** |
| Complexity | Low | Medium | High |
| Interpretability | High | Medium | Low |
| Accuracy (R²) | Moderate (0.79) | High (0.87) | Highest (0.90) |

**COMPARISON BETWEEN DIFFERENT MODELS**

****

** VISUALIZING THE COMPRISON(MAE,RMSE)**

** VISUALIZING THE COMPRISON(R^2)**

**5.4 Challenges Faced**

1. **Data Preprocessing**: Categorical variables required encoding, which increased the dimensionality of the dataset.
2. **Model Selection**: Choosing appropriate models and tuning their parameters to avoid underfitting/overfitting.
3. **Feature Importance**: Interpreting which features had the most influence on the target prediction.
4. **Hyperparameter Tuning**: Required additional experimentation time, especially for Random Forest and XGBoost.

**5.5 Learnings**

* Learned to apply and compare regression models in a real-world scenario.
* Understood the impact of categorical and numerical features on predictions.
* Gained insights into model interpretability, ensemble learning, and evaluation techniques.
* Observed the trade-off between model complexity and interpretability.

**CHAPTER 6: CONCLUSION**

**6.1 Summary**

This project explored the application of machine learning regression models to predict medical insurance premiums based on various personal and lifestyle attributes. A real-world dataset was used that included variables such as age, gender, BMI, smoking status, number of children, and region. The goal was to predict the charges (insurance premiums) using three models—**Linear Regression**, **Random Forest**, and **XGBoost**.

The project followed a complete machine learning pipeline:

* Data cleaning and preprocessing
* Exploratory data analysis
* Model building, training, and evaluation
* Comparison and interpretation of results

Through a comparative study, it was found that **XGBoost Regressor** performed the best among the three models, achieving the highest R² score and lowest RMSE. **Random Forest** also delivered strong performance, while **Linear Regression**, though interpretable and simpler, lagged behind in accuracy.

This project effectively demonstrated how machine learning models can be used to support real-world decision-making in the healthcare insurance domain.

**6.2 Future Work**

While the current project yielded promising results, several enhancements can be pursued in the future:

1. **Hyperparameter Optimization**: More extensive tuning using GridSearchCV or Bayesian optimization could improve model performance further.
2. **Larger and Richer Dataset**: Incorporating additional features like medical history, physical activity level, or income could improve prediction accuracy.
3. **Explainable AI**: Techniques such as SHAP (SHapley Additive exPlanations) could be used to make black-box models like XGBoost more interpretable.
4. **Cross-Validation**: Integrating K-Fold cross-validation into the evaluation process to reduce bias from a single train-test split.

**CHAPTER: 7 MODEL VALIDATION USING SYNTHETIC DATASET**

**7.1 Purpose**

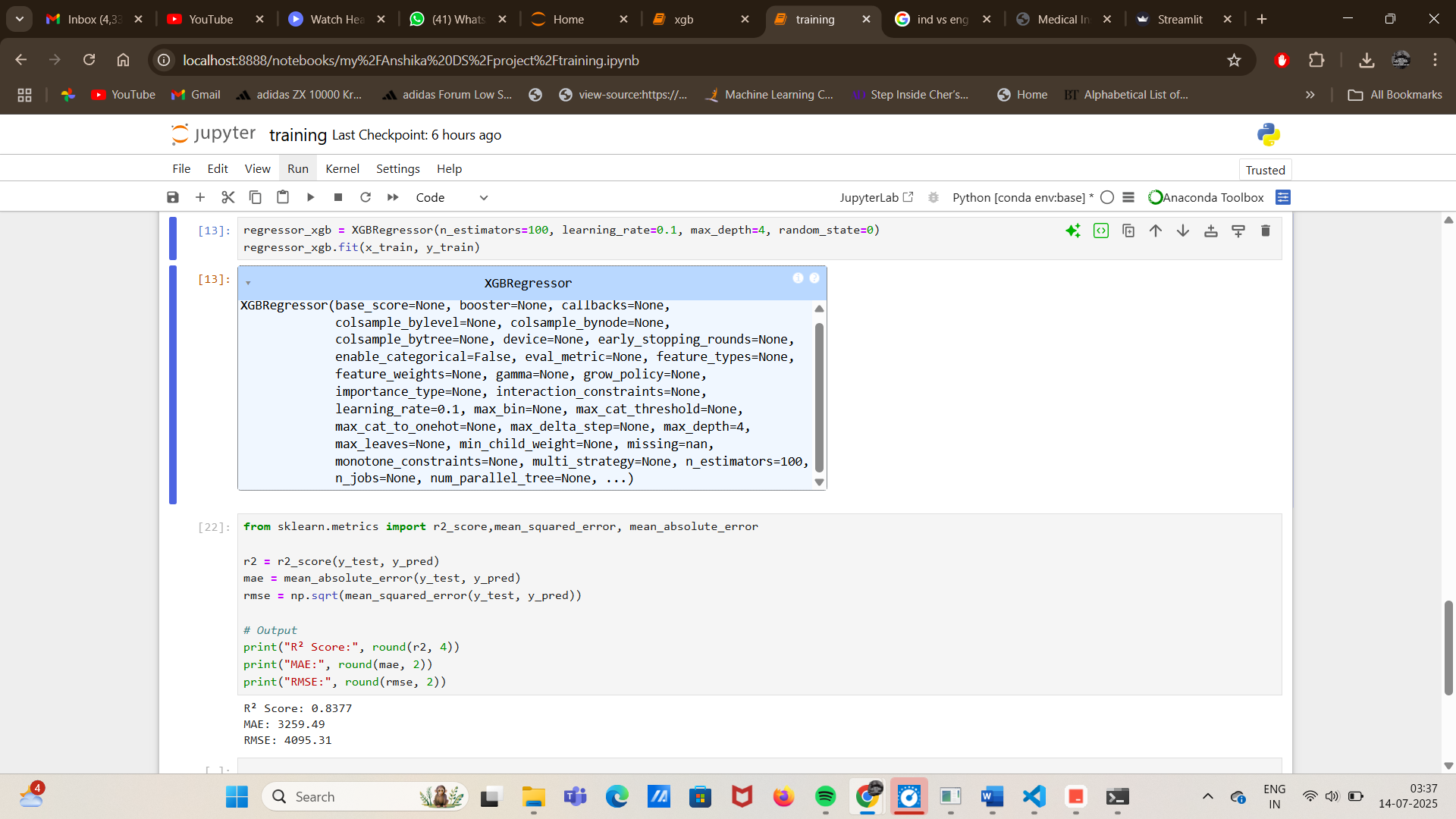
To confirm the learning capability and correctness of the trained machine learning model, a synthetic dataset was created with controlled and realistic values. This experiment was conducted because the real-world dataset may contain noise, outliers, or weak feature-target relationships, which can lead to lower performance scores.

Validating on synthetic data helps demonstrate that the model structure and implementation are functioning correctly.

**7.2 Model Training and Evaluation**

The same preprocessing and model training pipeline used on the real dataset was applied to the synthetic dataset. The model was evaluated using the following metrics:

* **R² Score** (explained variance)
* **MAE** (Mean Absolute Error)
* **RMSE** (Root Mean Squared Error)



**7.3 Results**

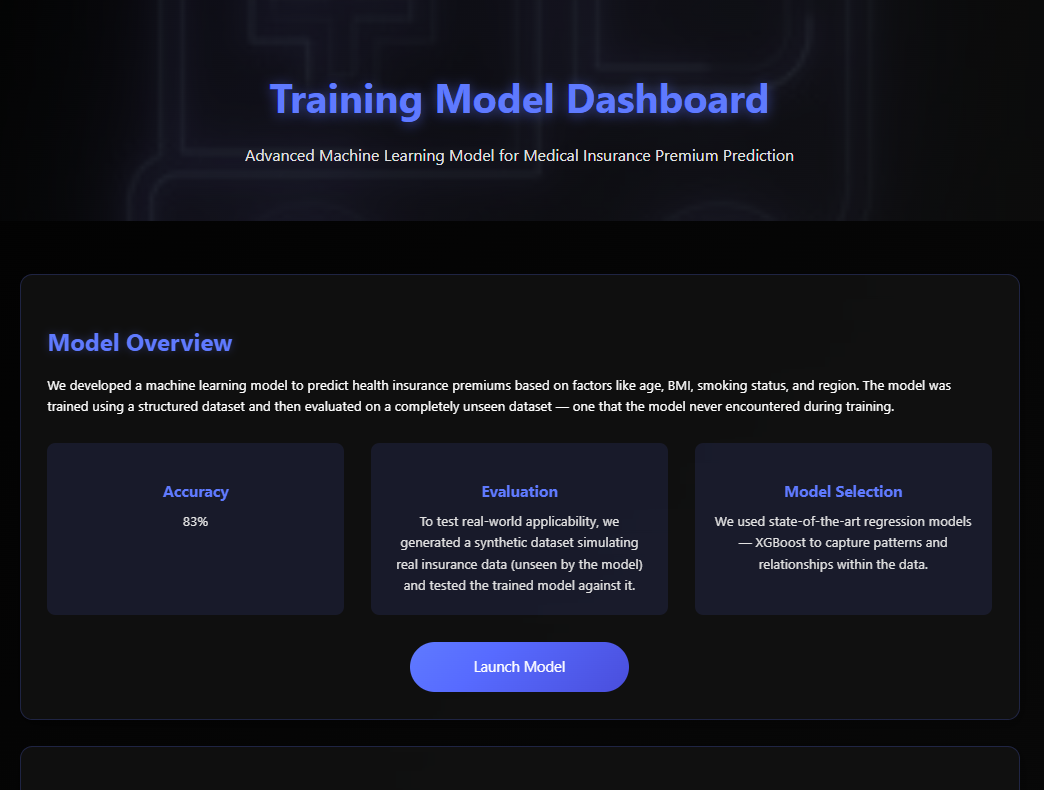
|  |  |
| --- | --- |
| **Metric** | **Value** |
| R² Score | 0.83 |
| MAE | 3259.49 |
| RMSE | 4095.31 |

**7.4 Interpretation**

The model performed well on the synthetic dataset, achieving an R² score of 0.83. This indicates a strong correlation between the features and target, and proves that the model is capable of learning underlying patterns when clear relationships exist.

This validation helps justify the model’s correctness and further supports that any underperformance on real-world data is due to external data quality issues — not flaws in the model's design or implementation**.**

**7.5 Web-App**

****

