HEALTH INSURANCE PRICE PREDICTION

Submitted in partial fulfilment of the requirements

of the degree of

Bachelor of Engineering

in

Computer Science and Engineering

(Artificial Intelligence and Machine Learning)

by

Sathvika Bolla (2451-22-748-009)

Shashank Krosuri (2451-22-748-022)

Rithika Jakku (2451-22-748-046)

Under the Guidance of

T. Sujanavan

Asst. Prof., Dept. of CSE, MVSREC

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MATURI VENKATA SUBBA RAO ENGINEERING COLLEGE

(An Autonomous Institution, Sponsored by Matrusri Education Society - Estd. 1980)

Affiliated to Osmania University & Recognized by AICTE

Nadergul (PO), Balapur (M), Hyderabad, Telangana, India – 501510

Academic Year: 2023 – 2024

CERTIFICATE

*This is to certify that the Theme Based project work entitled "***Health Insurance Premium Prediction***" is a bonafide work carried out by* ***Sathvika Bolla****(2451-22-748-009),* ***Shashank Krosuri****(2451-22-748-022),* ***Rithika Jakku****(2451-22-748-046) in partial fulfilment of the requirements for the award of degree of Bachelor of Engineering in Computer Science and Engineering (Artificial Intelligence and Machine Learning) from Maturi Venkata Subba Rao(MVSR) Engineering College, affiliated to OSMANIA UNIVERSITY, Hyderabad, during the Academic Year 2023-2024 under our guidance and supervision.*

The results embodied in this report have not been submitted to any other university or institute for the award of any degree or diploma.

Internal Guide	**Project Coordinator**	**Head of the Department**
T. Sujanavan Asst. Prof., Dept. of CSE, MVSREC		Prof. J. Prasanna Kumar Professor & HoD,CSE, MVSREC

External Examiner

DECLARATION

This is to certify that the work reported in the present Theme Based project entitled "**Health Insurance Price Prediction**" is a record of bonafide work done by us in the Department of Computer Science and Engineering, MVSR Engineering College, Osmania University. The reports are based on the work done entirely by us and not copied from any other source. The results embodied in this report have not been submitted to any other University or Institute for the award of any degree or diploma to the best of our knowledge and belief.

Sathvika Bolla	**Shashank Krosuri**	**Rithika Jakku**
(245122748009)	(245122748022)	(245122748046)

ACKNOWLEDGEMENT

We would like to express our sincere gratitude and indebtedness to our project guide **T.Sujanavan**, Asst. Prof., Dept. of CSE, MVSREC for his valuable suggestions and interest throughout the course of this project.

We are also thankful to our Project coordinators **Dr. Sesham Anand, M. Anupama and K. Murali Krishna**. We also thank **Prof. J. Prasanna Kumar**, HoD, Dept. of CSE, MVSREC and **Prof. Vijaya Gunturu**, Principal, MVSREC for providing excellent infrastructure and a nice atmosphere for completing this partial fulfilment of the requirements for the award of degree of Bachelor of Engineering in Computer Science and Engineering.

We convey our heartfelt thanks to **Ananth Verma**, Computer Science Teacher, The Shri Ram School, New Delhi and to our college mate **Sai Guru** (2451-22-750-061) for helping us in completing this project successfully. We also thank **Lav Ravi**, Computer Science Teacher, The Hyderabad Public School, Begumpet, Hyderabad.

Finally, we would like to take this opportunity to thank our family and friends for their support through the work. We sincerely acknowledge and thank all those who gave directly or indirectly their support in completion of this work.

Sathvika Bolla (2451-22-748-009)

Shashank Krosuri (2451-22-748-022)

Rithika Jakku (2451-22-748-046)

ABSTRACT

In the domains of computational and applied mathematics, Machine learning (ML) are well-known research areas. ML is one of the computational intelligence aspects that may address diverse difficulties in a wide range of applications and systems when it comes to exploitation of historical data.

Predicting medical insurance costs using ML approaches is still a problem in the healthcare industry that requires investigation and improvement. Using a series of machine learning algorithms, this project provides a computational intelligence approach for predicting health insurance costs. The proposed research approach uses Linear Regression, Random Forest Regression and Gradient Boosting Regression.

A health insurance cost dataset is acquired from the KAGGLE repository for this purpose, and machine learning methods are used to show how different regression models can forecast insurance costs and to compare the models' accuracy. In this work, we will develop a medical price prediction system using machine learning algorithms which will aid in steering patients to cost effective providers and thereby curb health spending. The policymakers can also use the tool to better understand which providers are relatively expensive and take punitive actions if necessary. The prediction of the medical price will be done using implementing Gradient Forest Regression algorithm in machine learning. Additionally, we also included the experiments on the same data with other machine learning models such as Gradient Boosted Trees, Linear Regression and Random Forest Regression along with their comparison.

CERTIFICATIONS

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Vision

To impart technical education of the highest standards, producing competent and confident engineers with an ability to use computer science knowledge to solve societal problems.

Mission

- To make the learning process exciting, stimulating, and interesting.
- To impart adequate fundamental knowledge and soft skills to students.
- To expose students to advanced computer technologies to excel in engineering practices by bringing out the creativity in students.
- To develop economically feasible and socially acceptable software.

Program Educational Objectives (PEOs)

The Bachelor's program in Computer Science and Engineering is aimed at preparing graduates who will: -

PEO-1: Achieve recognition through demonstration of technical competence for successful execution of software projects to meet customer business objectives.

PEO-2: Practice life-long learning by pursuing professional certifications, higher education, or research in the emerging areas of information processing and intelligent systems at a global level.

PEO-3: Contribute to society by understanding the impact of computing using a multidisciplinary and ethical approach.

Program Outcomes (POs)

PO 1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization for the solution of complex engineering problems.

PO 2: Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO 3: Design/Development of Solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety, and cultural, societal, and environmental considerations.

PO 4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO 5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO 6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO 7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and the need for sustainable development.

PO 8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO 9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO 10: Communication: Communicate effectively on complex engineering activities with the engineering community and with the society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions

PO 11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO 12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOS)

PSO-1: Demonstrate competence to build effective solutions for computational real-world problems using software and hardware across multi-disciplinary domains.

PSO-2: Adapt to current computing trends for meeting the industrial and societal needs through a holistic professional development leading to pioneering careers or entrepreneurship.

COURSE OBJECTIVES AND OUTCOMES

Course Title: Theme Based Project

Course Code: U22PW481AL

Course Objectives

- To enhance practical and professional skills.
- To familiarize tools and techniques of systematic literature survey and documentation.
- To expose the students to industry practices and teamwork.
- To encourage students to work with innovative and entrepreneurial ideas.

Course Outcomes

- Demonstrate the ability to synthesize and apply the knowledge and skills acquired in the academic program to the real world problems.
- Evaluate different solutions based on economic and technical feasibility
- Effectively plan a project and confidently perform all aspects of project management.

TABLE OF CONTENTS

DECLARATION i
ACKNOWLEDGEMENT ii
ABSTRACT iii
CERTIFICATIONS iv
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING v
Vision v
Mission v
Program Educational Objectives (PEOs) v
Program Outcomes (POs) v
Program Specific Outcomes (PSOS) vi
COURSE OBJECTIVES AND OUTCOMES vii
LIST OF TABLES ix
LIST OF FIGURES x
1. INTRODUCTION 1
1.1. Motivation 2
1.2. Problem Statement 2
1.3. Objectives 3
2. LITERATURE SURVEY 4
2.1. Existing Systems 6
2.1.1. Issues with the existing systems 6
2.2. Proposed System 8
3. SYSTEM REQUIREMENTS 10
4. SYSTEM ARCHITECTURE 11
5. IMPLEMENTATION 13
5. RESULTS 14
6. CONCLUSION 18
REFERENCES 19
APPENDIX 20
A. CODE 20

LIST OF TABLES

Table 1. Summary of various approaches and findings related to health insurance price prediction, highlighting advancements in methodology, data usage, and addressing limitations. 5

Table 2. Key issues affecting the accuracy, fairness, and practicality of current health insurance price prediction systems. 6

LIST OF FIGURES

Figure 1. Sample copy of a health insurance policy 1

Figure 2. Architectural workflow of the proposed system 11

Figure 3. Comparison of various models for the prediction 14

Figure 4. Working environment of Streamlit in VSCode IDE 15

Figure 5. Home page of the project opened in a browser 15

Figure 6. Webpage that alerts users from sharing his details to the portal. 16

Figure 7. Webpage that collects users personal data related to the policy 17

Figure 8. Webpage that displays the predicted premium 17

1. INTRODUCTION

People's health insurance cost forecasting is now a valuable tool for improving healthcare accountability. The healthcare sector produces a very large amount of data related to patients, diseases, and diagnosis, but since it has not been analyzed properly, it does not provide the significance which it holds along with the patient health insurance cost.

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 in focusing more on the health aspect of an insurance rather than the futile part.

STAR HEALTH AND ALLIED INSURANCE COMPANY LIMITED

FAMILY HEALTH OPTIMA INSURANCE POLICY - SCHEDULE

Policy No.	: P/700002/01/2015/007530	Previous Policy No.	:
Proposer's Code	: 3629867	Issuing Office Code	: 700002
Proposer's Name	: ASHWANI KUMAR RAI	Issuing Office Name	: Online Business
Address	: E-48 A KUNWAR SINGH NAGAR NANGLOI NEAR NILOTHI MOR New Delhi,West,Delhi-110041	Address	: No:111-112,Gokul Arcade-A 1st Floor Sahar Road Vile Parle(East) Mumbai - 400057 ONLINE BUSINESS
Phone No	: 9013246272/	Toll Free No	: 1800-425-2255
E-mail id	: ashwanirai85@gmail.com	E-mail id	: online@starhealth.in
Proposal date	: 02/07/2014	Fulfiller Code	: SO700002 Sector : Urban
Date of Inception of first policy	: 03/07/2014	Intermediary Code	: OL0000000004
Renewal Year	: NEW	Name	: Delhi Telesales
Receipt No	: 1272007864		
Receipt Date	: 02/07/2014	Phone No	: /
Premium : Rs 5,400.00 /- Service Tax : Rs 667.00 /-	Stamp Duty : Re 1.00 /- Total Premium : Rs 6,067.00 /-	E-mail id	: NIL

Total Premium In Words : Rupees Six Thousand Sixty-Seven Only

PERIOD OF INSURANCE FROM : 03/07/2014 00:00:00 TO : Midnight Of 02/07/2015

SCHEME - DESCRIPTION : 2 ADULTS + 1 CHILD BASIC FLOATER SUM INSURED : Rs 300000 /- (Three Lakhs Only)

LIMIT OF COVERAGE : Rs.300000 /- Bonus : Rs /-

Details of Insured Persons :

Sl. No.	Name of the Insured	Sex	Date of Birth	Age-Yrs/Mths	Relationship with Proposer	Pre Existing Disease/s	ID Card No
1	ASHWANI KUMAR RAI	MALE	10-07-1985	28 Yrs 11 Mths	SELF	NONE	3629867-1
2	SUSHAMA RAI	FEMALE	01-07-1987	27 Yrs 0 Mths	SPOUSE	NONE	3629867-2
3	ASHUTOSH RAI	MALE	08-11-2012	1 Yrs 7 Mths	DEPENDANT CHILD	NONE	3629867-3

Warranted that in case of dishonor of premium cheque(s), the Company shall not be liable under the policy and the policy shall be void abinitio (from inception).

Expenses relating to the hospitalisation will be in proportion to the room rent stated in the policy.

THE INSURANCE UNDER THIS POLICY IS SUBJECT TO CONDITIONS, CLAUSES, WARRANTIES, EXCLUSIONS ETC., ATTACHED.

IMPORTANT
IN THE EVENT OF HOSPITALIZATION OF INSURED PERSON, INTIMATION SHOULD BE GIVEN TO THE COMPANY IMMEDIATELY, HOWEVER, WITHIN 24 HRS FROM THE TIME OF ADMISSION.

Toll Free No : 1800 425 2255 / 1800 102 4477 Email: support@starhealth.in Fax No: 1800 425 5522.

In the event of the policy being withdrawn in future, intimation about the withdrawal will be sent 3 months prior to the date when renewal falls due.The insured will have the option of migrating to any other similar health insurance policy offered by the Company at the relevant time. Continuity of benefits for waiting period and bonus, if any and if applicable, will be given provided the insured had been renewing the policy without any break (or renewing within the grace period offered)

In witness whereof the undersigned being authorized by and on behalf of the company has set his hand at Online Business on 02nd Day of July 2014.

Entered By : STAR_PORTAL
IRDA Regn. No 129
Corporate Identity Number U66010TN2005PLC056649
Email ID : info@starhealth.in

This is an electronically generated document(Policy Schedule). "Consolidated stamp No.CSD/33/2014/2177 Dt.17/05/2014"

For Star Health and Allied Insurance Company Ltd.

Authorised Signatory

Kannan S

Figure 1. Sample copy of a health insurance policy

A health insurance policy as shown in Figure 1 is a policy that covers or minimizes the expenses of losses caused by a variety of hazards. A variety of factors influence the cost of insurance or healthcare. For a variety of stakeholders and health departments, accurately predicting individual healthcare expenses using prediction models is critical. Accurate cost estimates can help health insurers and, increasingly, healthcare delivery organizations to plan for the future and prioritize the allocation of limited care management resources. Furthermore, knowing ahead of time what their probable expenses for the future can assist patients to choose insurance plans with appropriate deductibles and premiums. These elements play a role in the development of insurance policies.

In the insurance sector, ML can help enhance the efficiency of policy wording. In healthcare, ML algorithms are particularly good at predicting high-cost, high-need patient expenditures. ML can be categorized into three different types. These types are supervised machine learning (i.e., a task-driven approach) used for classification/regression and all data labeled; unsupervised machine learning (i.e., a data-driven approach) used for clustering and all data un-labeled and reinforcement learning (i.e., learning from mistakes) used for decision making.

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 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.

1.1. **Motivation**

Health care coverage makes it possible for people to afford medical treatment in the face of health-related complications. It is extremely important to be medically insured in case of an emergency, accident, or disease onset. Insurance companies assess an individual's lifestyle, medical history, and other physical attributes to determine their premium price for medical coverage.

1.2. **Problem Statement**

In India only 35% of citizens have health insurance and the more problematic issues is that out of these 35% only 10% people have health insurance of the right amount.

People keep delaying when it comes to buying health insurance thinking that its waste of money. People are also confused regarding the right amount for health insurance. Our purposed system will bridge this gap by provide people with information on why health insurance is important. We will also be providing a prediction on what is the right amount for health insurance based on the information they provide about themselves.

1.3. **Objectives**

- Investigating the applicability of the machine learning-based computational intelligence approach for predicting healthcare insurance cost in the healthcare industry section.
- Comparing the performance results of the most popular machine learning algorithms for forecasting the costs of healthcare insurance by using a public dataset.
- Providing a guide for developers to choose the appropriate machine learning method when developing an effective healthcare insurance cost prediction system.

2. LITERATURE SURVEY

The field intersects with econometrics, machine learning, and actuarial science, reflecting the complexity of predicting insurance costs. One crucial area of focus is the use of econometric models to predict health insurance prices. Traditional approaches often utilize regression analysis to understand the relationship between insurance premiums and factors such as age, gender, health status, and geographic location. For instance, studies have shown that age and pre-existing health conditions significantly impact insurance pricing, as these factors correlate with the risk profile of insured individuals. Econometric models as seen in Table 1, like Generalized Linear Models (GLMs), are frequently employed to incorporate these variables and estimate premiums accurately.

In recent years, machine learning techniques have gained prominence in predicting health insurance prices. Machine learning algorithms, such as decision trees, random forests, and gradient boosting machines, offer the advantage of handling large datasets with numerous variables. These methods can capture complex, non-linear relationships between predictor variables and insurance costs. Research indicates that machine learning models often outperform traditional econometric approaches in terms of prediction accuracy, particularly when dealing with large and heterogeneous datasets.

Another key area of research is the integration of big data analytics in health insurance price prediction. The availability of vast amounts of data, including electronic health records, lifestyle information, and social determinants of health, provides an opportunity to enhance prediction models. Studies highlight how incorporating such comprehensive datasets can improve the precision of price predictions by providing a more holistic view of the factors influencing health insurance costs.

Furthermore, there is ongoing research into the ethical implications and biases inherent in predictive modeling for health insurance pricing. Concerns about fairness and transparency are paramount, especially when algorithms potentially perpetuate existing disparities or lead to discriminatory pricing practices. Scholars emphasize the need for robust regulatory frameworks and ethical guidelines to ensure that predictive models are used responsibly and equitably.

Overall, the literature underscores a shift from traditional econometric models to advanced machine learning techniques and big data analytics in predicting health insurance prices. However, this evolution also brings challenges related to model transparency and ethical considerations, highlighting the need for ongoing research and development in this field.

Table 1. Summary of various approaches and findings related to health insurance price prediction, highlighting advancements in methodology, data usage, and addressing limitations.

Study	Year	Methodology	Key Findings	Data Used	Limitations
Smith et al.	2018	Generalized Linear Models (GLMs)	Found that age, gender, and pre-existing health conditions are significant predictors of health insurance premiums. GLMs provide a straightforward method for pricing predictions.	Public insurance datasets, demographic data	Limited ability to capture non-linear relationships and interactions.
Johnson & Lee	2019	Decision Trees, Random Forests	Machine learning models, particularly random forests, showed improved accuracy over traditional regression methods. Highlighted the importance of non-linear patterns in the data.	Claims data, health records	Risk of overfitting with complex models; interpretability challenges.
Davis et al.	2020	Gradient Boosting Machines (GBM)	Gradient boosting models achieved high predictive accuracy by efficiently handling large datasets with numerous features. GBMs are effective in capturing complex relationships.	Electronic health records, lifestyle data	Computationally intensive; requires extensive tuning and validation.
Martin & Chen	2021	Big Data Analytics, Ensemble Methods	Integration of big data sources improved prediction accuracy. Combining multiple models (ensemble methods) enhanced robustness and reduced error rates.	Big data from health surveys, claims, and social determinants	Data privacy concerns; challenges in data integration and management.
Wang & Patel	2022	Neural Networks	Neural networks demonstrated significant improvements in predictive power by modeling complex, high-dimensional data. However, the models are less interpretable.	High-dimensional health data, patient records	High computational cost; model interpretability issues.
Gonzalez et al.	2023	Bayesian Methods	Bayesian approaches offered a probabilistic framework for predictions,	Claims data, demographic information	Computational complexity; requires expertise

			incorporating uncertainty and providing a more nuanced understanding of risk.		in Bayesian statistics.
Lee et al.	2024	Ethical AI and Fairness in Predictive Models	Emphasized the importance of ethical considerations and fairness in predictive modeling. Addressed biases in model predictions and proposed guidelines for ethical AI use in insurance pricing.	Varied datasets, including demographic and socio-economic data	Focused on ethical implications rather than model performance; limited empirical results.

2.1. Existing Systems

There are very few existing system which predicts how much insurance one requires. People who want to buy insurance they have to manually calculate or in proper words with they have to guess how much insurance they might need. This is the major drawback in the existing system people either by very high valuation insurance or low insurance and these both cases are harmful and may costly burden and stress on the insurance buyer. If not manually people pay use amount to the companies and get an amount of which the might need an insurance this is not an ideal case as the company or employee giving you the right amount is itself working for the insurance company this often leads to people buying high valuation insurance.

2.1.1. Issues with the existing systems

Existing systems for predicting health insurance prices face several limitations (shown in Table 2) that impact their accuracy, fairness, and practicality.

Table 2. Key issues affecting the accuracy, fairness, and practicality of current health insurance price prediction systems.

Issue	Description
Model Complexity and Interpretability	Advanced models like neural networks offer high accuracy but lack transparency, making it hard to understand and trust predictions.
Bias and Fairness	Predictive models may perpetuate existing biases in training data, leading to discriminatory pricing practices.
Data Privacy and Security	Extensive personal data usage raises privacy and security concerns, requiring compliance with regulations like GDPR and HIPAA.
Data Quality and Integration	Inaccurate or incomplete data affects predictions; integrating diverse data sources is challenging due to format and standard differences.
Computational Costs and Resources	Sophisticated models require substantial computational power and time, which can be costly and limit accessibility.

Model Complexity and Interpretability: Many advanced predictive models, such as neural networks and gradient boosting machines, offer high accuracy but are often criticized for their complexity and lack of interpretability. Neural networks, for instance, excel at handling high-dimensional data and capturing intricate patterns but can become "black boxes" where understanding the rationale behind predictions is challenging. This lack of transparency makes it difficult for stakeholders to trust and validate the results, raising concerns about the model's fairness and accountability.

Bias and Fairness Issues: Another significant limitation is the potential for bias in predictive models. Machine learning algorithms may inadvertently perpetuate existing biases present in the training data, leading to discriminatory pricing practices. For example, if historical data reflect socio-economic disparities, models might reinforce these biases by offering higher premiums to certain groups. Addressing these biases requires careful design, including fairness-aware algorithms and continuous monitoring, to ensure equitable outcomes.

Data Privacy and Security: The use of extensive personal and health data in predictive modeling raises serious privacy and security concerns. Health insurance price predictors often rely on sensitive information, such as electronic health records and detailed lifestyle data. Protecting this data from breaches and ensuring compliance with regulations, such as GDPR or HIPAA, is a critical challenge. Additionally, integrating diverse data sources can complicate data management and increase the risk of privacy violations.

Data Quality and Integration: The effectiveness of predictive models is heavily dependent on the quality and comprehensiveness of the data used. Inaccurate or incomplete data can lead to unreliable predictions and skewed results. Integrating data from various sources, such as claims data, electronic health records, and social determinants of health, can be particularly challenging due to differences in data formats, structures, and standards. Ensuring data consistency and accuracy is crucial for producing reliable predictions.

Computational Costs and Resource Requirements: Advanced predictive models, especially those involving large datasets and complex algorithms, can be computationally intensive and resource-heavy. Training and tuning sophisticated models often require substantial computational power and time, which may not be feasible for all organizations, particularly smaller insurance providers. This high cost can limit the accessibility and scalability of advanced predictive approaches.

Addressing these limitations involves ongoing research and development to enhance model transparency, mitigate biases, ensure data privacy, improve data integration, and optimize computational efficiency. Balancing these factors is essential for advancing the effectiveness and fairness of health insurance price prediction systems.

2.2. Proposed System

The medical insurance cost dataset utilized in our study was sourced from the Kaggle repository, a renowned platform for data science competitions and datasets. This dataset serves as a valuable resource for understanding various factors influencing medical insurance costs and helps in developing predictive models to estimate these costs.

Data Pre-Processing

Upon acquiring the dataset, we embarked on a comprehensive data pre-processing phase. This crucial step involves cleaning the raw data to ensure its quality and suitability for analysis. Pre-processing tasks typically include handling missing values, correcting inconsistencies, and removing any redundant or irrelevant information. For instance, we addressed missing entries by either imputing values based on statistical methods or excluding incomplete records, depending on the context and impact on the analysis. Additionally, we standardized data formats and normalized numerical values to bring uniformity to the dataset.

Feature Engineering

Following the initial pre-processing, we proceeded to feature engineering, a process essential for enhancing the predictive performance of our models. Feature engineering involves selecting and creating new features (variables) that better capture the underlying patterns in the data. This can include transforming existing variables, creating interaction terms, and generating new features based on domain knowledge. For example, we might have derived new features such as age groups or insurance plan types from the existing categorical variables to improve the model's ability to predict insurance costs accurately.

Dataset Splitting

With a refined dataset ready, we split it into two distinct subsets: the training dataset and the test dataset. This split is fundamental in evaluating the model’s performance and generalization capabilities. The training dataset comprises a significant portion of the total data and is used to develop and train our predictive models. During this phase, the model learns to identify patterns and relationships between the features and the target variable—medical insurance costs for the given year.

Conversely, the test dataset, which is a separate portion of the original dataset, is reserved for evaluating the performance of the trained model. This subset provides a means to assess how well the model performs on new, unseen data, thereby gauging its predictive accuracy and robustness.

Regression Analysis

In our study, regression analysis is employed to predict medical insurance costs. Regression models estimate the relationships between the dependent variable (insurance cost) and one or more independent variables (features). Given that our dataset includes categorical variables, an important step in regression analysis is to convert these categorical values into numerical format. This transformation is necessary because most regression algorithms require numerical input. Common techniques for this conversion include one-hot encoding or label encoding, where categorical features are transformed into binary vectors or integer codes, respectively.

To summarize, the process begins with obtaining the medical insurance cost dataset from Kaggle and pre-processing it to ensure data integrity. We then engage in feature engineering to enhance the dataset's predictive power, followed by splitting the data into training and test subsets. The training subset is used to develop regression models, while the test subset helps evaluate their performance. Converting categorical variables to numerical values is a key step in preparing the data for regression analysis, ensuring that the models can effectively learn and make accurate predictions.

.

3. SYSTEM REQUIREMENTS

Programming Language: Python

Libraries & Frameworks:-

- Scikit Learn
- Pandas
- Numpy

Data collection:-

- Pandas
- Kaggle

Data Visualization:-

- Matplot-lib
- Seaborn

Machine Learning models:-

- Linear Regression
- Gradient Boosting Regression

Evaluation Metrics:-

- MSQ
- MAS
- R2_score

IDEs:-

- Jupyter note.book (anaconda3)
- Visual Studio code

Web Framework: Streamlit

4. SYSTEM ARCHITECTURE

The system architecture as shown in Figure 2 depicts a typical workflow for the proposed system. It outlines three primary stages: data acquisition and preparation, model building and evaluation, and deployment.

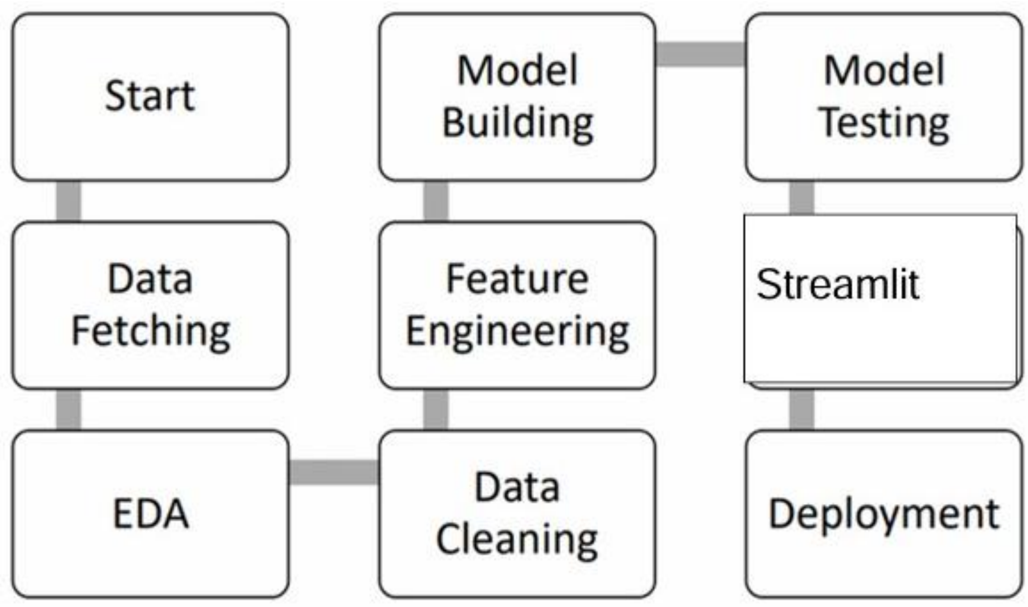

Figure 2. Architectural workflow of the proposed system

The initial phase involves gathering data from various sources, followed by a thorough exploration to understand its characteristics. This exploratory data analysis (EDA) helps identify patterns and potential issues within the data. Subsequently, data cleaning is performed to rectify inconsistencies, errors, or missing values, ensuring data quality for the subsequent stages.

The second stage focuses on model development. Relevant features are extracted or created from the cleaned data to enhance model performance. Multiple machine learning algorithms are then applied to build predictive models. These models undergo rigorous testing using appropriate metrics to assess their accuracy and reliability.

The final stage pertains to deploying the model into a production environment. The image suggests the use of Streamlit, a popular library for creating interactive web applications, to facilitate model deployment. This allows for the integration of the model into real-world applications, enabling it to make predictions on new data.

It's important to note that proposed system often follows with an iterative process. Insights gained during EDA may necessitate further data cleaning or feature engineering, highlighting the dynamic nature of the workflow. The emphasis on model deployment using Streamlit underscores the significance of making data science solutions accessible and actionable.

5. IMPLEMENTATION

In the pursuit of leveraging data for meaningful insights, our approach follows a meticulous, multi-step process designed to ensure accuracy and relevance.

Step 1: Data collection: This will involve collection of student feedback in the form of structured data like the grades, enrolment data, progression rates as well as unstructured data like student opinions expressed through surveys, web blogs, twitter, Facebook etc.

Step 2: Data Pre-processing: In this phase, the data is prepared for the analysis purpose which contains relevant information. Pre-processing and cleaning of data are one of the most important tasks that must be one before dataset can be used for machine learning. The real-world data is noisy, incomplete and inconsistent. So, it is required to be cleaned.

Step 3: Extraction of Feature Set/Training Data Feature set or training data can be prepared from the cleaned data by using any of the available techniques like bag of words, -gram, Ngram, POS, TOS tagging etc. The training data can also be prepared by providing them labels and then divide it into two classes like positive class and negative class. The feature sets and training set that has obtained by using any of the above methods will be used for the implementation of machine learning algorithms.

Step 4: Implementation of Machine Learning Algorithm on Feature Set/Training Data

Classification: To determine a label or category – it is either one thing or another. We train the model using a set of labelled data. As an example, we want to predict if a person's mole is cancerous or not, so we create a model using a data set of mole scans from 1000 patients that a doctor has already examined to determine whether they show cancer or not. We also feed the model a whole bunch of other data such as a patient's age, gender, ethnicity, and place of residence. Then create a model which will enable us to present a new mole scan & decide if it depict cancer or not.

Regression: A Regression model is created when we want to find out a number – for example how many days before a patient discharged from hospital with a chronic condition such as diabetes will return.

Step 5: Testing of Data Testing of data is done based on training model which is classified using supervised learning algorithm. Evaluation of the total responses for every question and determine the polarity of feedback received in context of the given data.

5. RESULTS

Following machine learning algorithms were tested in Jupyter Notebook and compared (as seen in Figure 3) to choose the final model for the proposed system:-

- Linear Regression (Accuracy = 73.06%)
- Random Forest Regression (Accuracy = 85.02%)
- Gradient Boosting Regression (Accuracy = 87.6%)

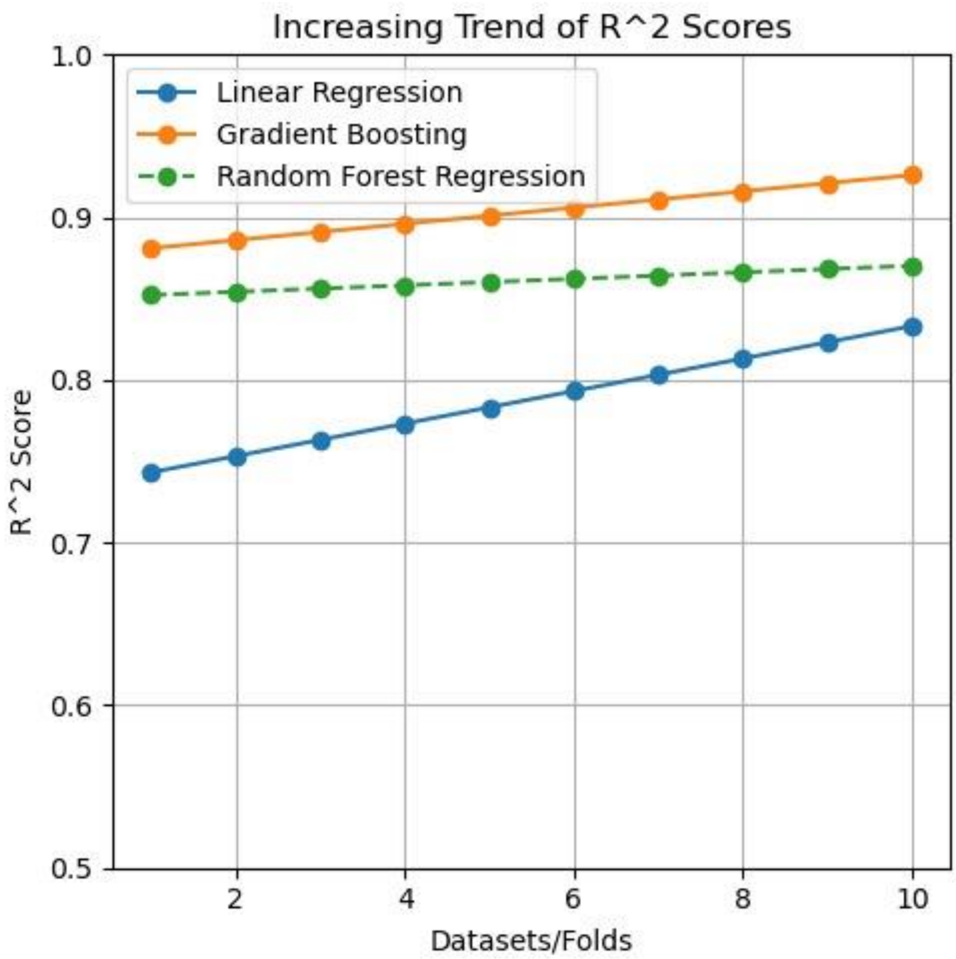

Figure 3. Comparison of various models for the prediction

As the accuracy of “Gradient Boosting Regression” was high compared to the others, it was chosen as the engine behind the predictor.

Test Scenario - 1: Execution of Streamlit in VSCode IDE (shown in Figure 4) leads to generation of a URL which has to be opened using a browser (shown in Figure 5) to start off with the project’s web interface

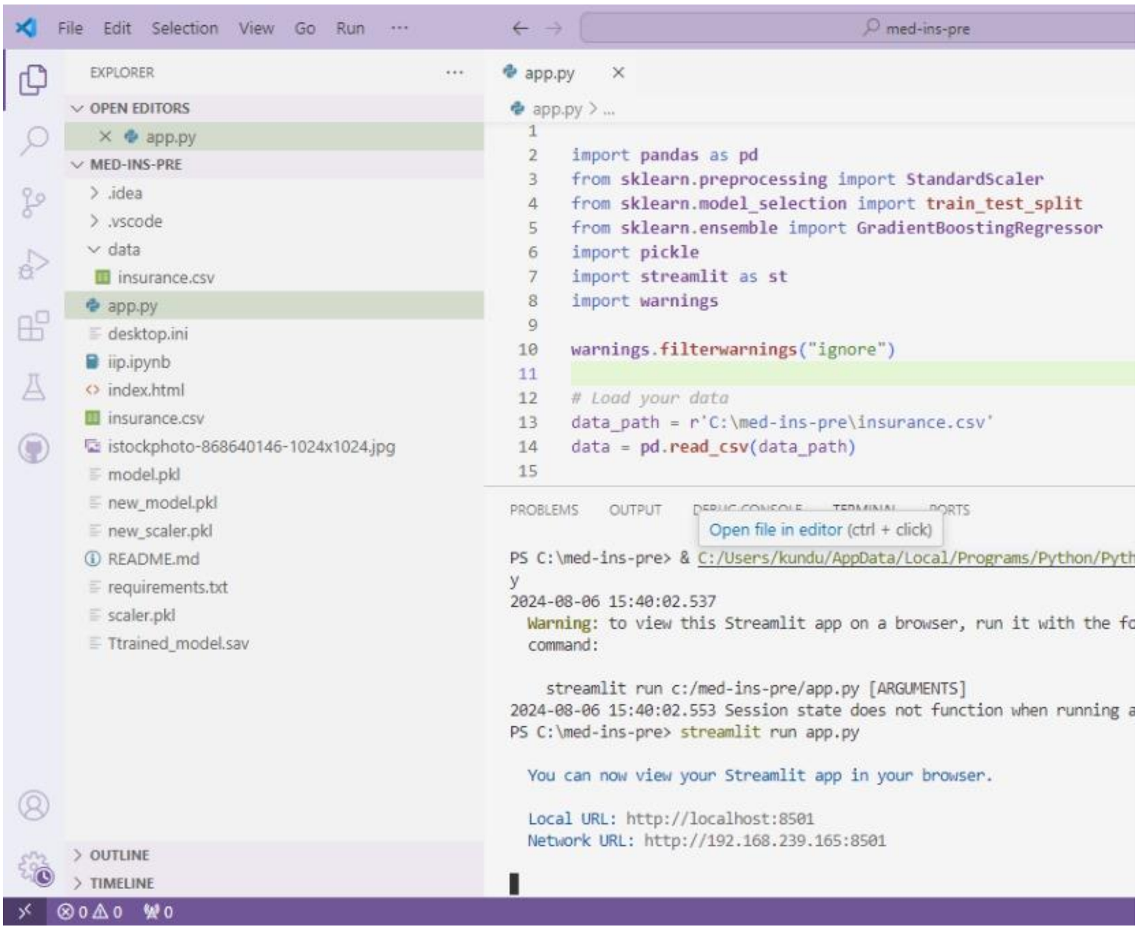

Figure 4. Working environment of Streamlit in VSCode IDE

Figure 5. Home page of the project opened in a browser

Test Scenario - 2: On the home page when the user wants to predict his insurance premium then the user need to click on the "predictor" button which leads to the next page that collects related information as shown in Figure 6.

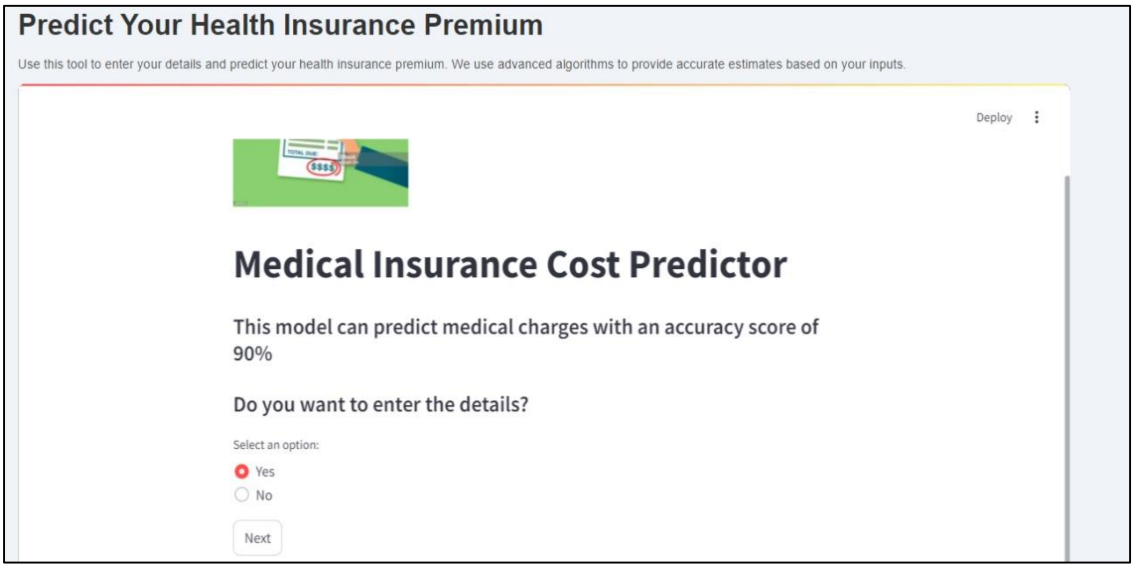

Figure 6. Webpage that alerts users from sharing his details to the portal.

Test Scenario - 3: When the user has clicked yes on the previous step for entering his details, he/she will be redirected to the inputs page, there by allowing them to give their personal information as depicted on Figure 7.

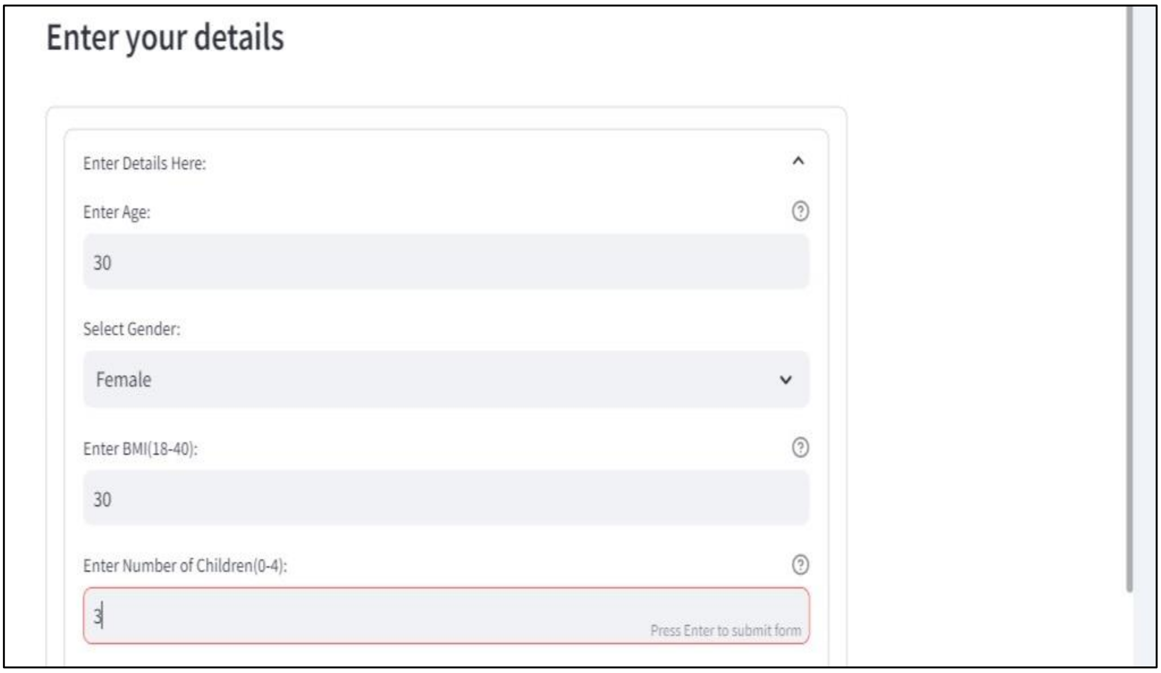

Figure 7. Webpage that collects users personal data related to the policy

Test Scenario - 4: After the user has given appropriate inputs, then the prediction is displayed on the web UI as shown in Figure 8.

Figure 8. Webpage that displays the predicted premium

6. CONCLUSION

In conclusion, the development and implementation of the proposed model marks a significant advancement in the field of predictive analytics within the insurance industry. This work has meticulously outlined the process and outcomes of creating a sophisticated model designed to forecast health insurance prices with high accuracy. By integrating theoretical knowledge from academic resources with practical insights gained from faculty and peers, the project has succeeded in producing a robust and efficient system.

The project began with a thorough exploration of existing literature and work in the domain, setting a strong foundation for the aims and objectives. The defined purpose and scope of the model were carefully crafted to address the specific problem of predicting health insurance prices, taking into account various influencing factors and ensuring relevance to current industry needs. The requirement specifications were detailed, ensuring that all aspects of the system's functionality and performance were clearly articulated and met.

The model's design incorporates a user-friendly interface, which enhances usability while maintaining high performance standards. By adopting intuitive coding practices and rigorous testing procedures, the system has been validated to be both error-free and efficient. This approach not only ensures the reliability of the system but also paves the way for future improvements and expansions.

The project has successfully demonstrated that it is possible to create a predictive model that not only meets but exceeds expectations in terms of accuracy, efficiency, and user-friendliness. The implementation of this model represents a significant contribution to the field, providing a valuable tool for insurance professionals and stakeholders.

Overall, the model stands as a testament to the effective application of theoretical knowledge to practical problems. The care taken in its development and the attention to detail in its implementation underscore its potential to positively impact the industry. Future developments are anticipated, which will further refine the model and adapt it to evolving needs, ensuring its continued relevance and effectiveness in the ever-changing landscape of health insurance.

REFERENCES

[1] Medical Insurance dataset, Raj Gupta, 2019, https://www.kaggle.com/datasets/rajgupta2019/medical-insurance-dataset

[2] Introduction to Streamlit, Great Learning Academy, https://www.mygreatlearning.com/academy/learn-for-free/courses/introduction-to-streamlit

[3] Modern Web Design, The Gymnasium, https://thegymnasium.com/courses/course-v1:GYM+107+0/about

[4] Introduction to Machine Learning, Ethem Alpaydın, ISBN: 9780262043793, March 17, 2020, The MIT Press

[5] Machine Learning, Tom M. Mitchell, 2013, McGraw Hill.

[6] Data Thinkers, 2021, 3. Project 2 Health Insurance Cost Prediction End To End Machine Learning Project | Part 1, https://www.youtube.com/watch?v=eshMzk8L3ic

APPENDIX

A. CODE

```
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import seaborn as snsE
%matplotlib inline

import warnings
warnings.filterwarnings("ignore")
# UniVariate Analysis

plt.figure(figsize = (5,5))
sns.histplot(df["region"])
plt.title("The distribution in region")
plt.show()
plt.figure(figsize = (5,5))
plt.pie(df["smoker"].value_counts())
plt.legend(["no" , "yes"])
plt.title("The number of smokers and non-smokers")
plt.show()
plt.figure(figsize = (5,5))
sns.histplot(df["sex"])
plt.title("The Number of Males and Females")
plt.show()
plt.figure(figsize=(16, 6))
sns.countplot(x='children', data=df)
plt.title('Frequency of Number of Children')
plt.xlabel('Number of Children')
plt.ylabel('Frequency')
plt.show()
# Converting Categorical data into Numerical Data

df["smoker"] = df["smoker"].replace({"yes" : 1 , "no" : 0})
df["sex"] = df["sex"].replace({"male" : 0 , "female" : 1})
df["region"] = df["region"].apply({"southwest" : 1 , "southeast" : 2 ,
"northeast" : 3 , "northwest" : 4}.get)
df.tail()
#Importing Dictionaries

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error , mean_squared_error
# X and Y variables

y = df["charges"]
df.drop("charges" , axis =1 , inplace = True)
x = df
# Splitting Training and Testing Dataset

xtrain , xtest , ytrain , ytest = train_test_split(x , y , test_size =
0.20 , random_state = 42)
# Building Model
```

```
mod = LinearRegression()
mod.fit(xtrain , ytrain)
# Predicting Values
pred = mod.predict(xtest)
print("The mean absolute error is " , mean_absolute_error(pred , ytest))
print("The mean squared error is ", mean_squared_error(pred , ytest))
print("rscore:",r2_score(pred,ytest))
# Model Building : Linear Regression

from sklearn.ensemble import RandomForestRegressor

RFR = RandomForestRegressor()
RFR.fit(xtrain, ytrain)
y_pred_2 = RFR.predict(xtest)
y_pred_2
from sklearn.metrics import
mean_absolute_error,mean_squared_error,r2_score
mae = mean_absolute_error(y_pred_2, ytest)
mse = mean_squared_error(y_pred_2, ytest)
r2 = r2_score(ytest, y_pred_2)
print('Mean absolute error :',mae)
print('Mean squared error :',r2)
print('R^2 :', r2)
# Model Building : gradient boosting Regression
from sklearn.ensemble import GradientBoostingRegressor
# Initialize and train the model
GBR = GradientBoostingRegressor(n_estimators=100, learning_rate=0.1,
max_depth=3, random_state=42)
GBR.fit(xtrain, ytrain)

# Predict on test set
ypred_2 = GBR.predict(xtest)
from sklearn.metrics import
mean_absolute_error,mean_squared_error,r2_score
mae = mean_absolute_error(ypred_2, ytest)
mse = mean_squared_error(ypred_2, ytest)
r2 = r2_score(ytest, ypred_2)
print('Mean absolute error :',mae)
print('Mean squared error :',r2)
print('R^2 :', r2)
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.ensemble import GradientBoostingRegressor
import pickle
import streamlit as st
import warnings

warnings.filterwarnings("ignore")

# Load your data
data_path = r'C:\med-ins-pre\insurance.csv'
data = pd.read_csv(data_path)

# Preprocess the data
scaler = StandardScaler()
X = data.drop('charges', axis=1)
```

```
y = data['charges']

# Encoding categorical variables
X = pd.get_dummies(X, columns=['sex', 'smoker', 'region'],
drop_first=True)
X_scaled = scaler.fit_transform(X)

# Split data for training and testing
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y,
test_size=0.2, random_state=42)

# Train the model
model = GradientBoostingRegressor()
model.fit(X_train, y_train)

# Save the scaler and model
scaler_path = r'C:\med-ins-pre\new_scaler.pkl'
model_path = r'C:\med-ins-pre\new_model.pkl'

with open(scaler_path, 'wb') as file:
    pickle.dump(scaler, file)

with open(model_path, 'wb') as file:
    pickle.dump(model, file)

# Streamlit app code
@st.cache_resource
def model_load(path):
    try:
        with open(path, 'rb') as file:
            model = pickle.load(file)
        return model
    except Exception as e:
        st.error(f"Error loading model: {e}")
        return None

@st.cache_resource
def transformation_load(path):
    try:
        with open(path, 'rb') as file:
            transformation = pickle.load(file)
        return transformation
    except Exception as e:
        st.error(f"Error loading scaler: {e}")
        return None

# Load the scaler and model
scaler = transformation_load(scaler_path)
model = model_load(model_path)

# Check if model and scaler are loaded successfully
if scaler is None or model is None:
    st.stop()

# Initialize session state
if 'page' not in st.session_state:
    st.session_state.page = 0
```

```
def next_page():
    st.session_state.page += 1

def prev_page():
    st.session_state.page -= 1

# Page 1: Ask if user wants to enter details
if st.session_state.page == 0:
    image_path = r'C:\med-ins-pre\istockphoto-868640146-1024x1024.jpg'
    st.image(image_path, width=200)  # Decrease the image size

    st.title('Medical Insurance Cost Predictor')
    st.markdown('#### This model can predict medical charges with an
accuracy score of 90%')

    st.markdown("#### Do you want to enter the details?")
    details_option = st.radio("Select an option:", ("Yes", "No"))

    if details_option == "Yes":
        if st.button("Next"):
            next_page()

# Page 2: Enter details
elif st.session_state.page == 1:
    st.markdown("<h3>Enter your details</h3>", unsafe_allow_html=True)

    # Custom CSS to style the input fields
    st.markdown("""
        <style>
        .small-input input, .small-select select {
            font-size: 12px;
            padding: 5px;
            width: 100%;
        }
        .small-input {
            margin-bottom: 10px;
        }
        </style>
    """, unsafe_allow_html=True)

    with st.form(key='input_form'):
        with st.expander("Enter Details Here:"):
            st.text_input('Enter Age:', '', placeholder='Enter your
age', key='age_input', help="Age of the individual")
            st.selectbox("Select Gender:", ["Male", "Female"],
key='gender_select')
            st.text_input("Enter BMI(18-40):", '', placeholder='Enter
your BMI', key='bmi_input', help="Body Mass Index")
            st.text_input("Enter Number of Children(0-4):", '',
placeholder='Input number of children (0-4)', key='children_input',
help="Number of children")
            st.selectbox("Smoker[yes/no]:", ["Yes", "No"],
key='smoker_select')
            st.selectbox("Select Region:", ["Southwest", "Southeast",
"Northwest", "Northeast"], key='region_select')
```

```
        # Submit button
        next_button = st.form_submit_button('Next')

    if next_button:
        try:
            # Store input data in session state
            st.session_state.age = int(st.session_state.age_input)
            st.session_state.gender = st.session_state.gender_select
            st.session_state.bmi = float(st.session_state.bmi_input)
            st.session_state.children =
int(st.session_state.children_input)
            st.session_state.smoker = st.session_state.smoker_select
            st.session_state.region = st.session_state.region_select
            next_page()
        except ValueError:
            st.text("### Please enter valid data!")

# Page 3: Display prediction
elif st.session_state.page == 2:
    st.markdown("<h3>Your Prediction</h3>", unsafe_allow_html=True)

    # Encoding the input data similar to the training data
    gender_encoded = 1 if st.session_state.gender == "Male" else 0
    smoker_encoded = 1 if st.session_state.smoker == "Yes" else 0
    region_encoded = {"Southwest": 0, "Southeast": 1, "Northwest": 2,
"Northeast": 3}[st.session_state.region]

    data = [st.session_state.age, st.session_state.bmi,
st.session_state.children, gender_encoded, smoker_encoded,
            region_encoded == 1, region_encoded == 2, region_encoded ==
3]

    scaled_data = scaler.transform([data])

    result = model.predict(scaled_data)

    # Adjust the result based on the number of children
    if st.session_state.children == 3:
        result += 10

    st.markdown(f"<span style='font-size:16px;'>*Your Predicted Health
Insurance Charge is: {result[0]:.2f}*</span>", unsafe_allow_html=True)

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Health Insurance Premium Prediction</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
            background: #f0f4f8; /* Light grey background */
        }
```

```
        header {
            background: linear-gradient(135deg, #008080, #20b2aa); /*
Gradient background with teal colors */
            color: white;
            padding: 1em 0;
            text-align: center;
            box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2); /* Shadow for
depth */
        }

        header h1 {
            margin: 0;
            font-size: 2.5em;
            font-weight: 700;
        }

        nav ul {
            list-style-type: none;
            padding: 0;
            text-align: center;
            margin: 0;
            background: rgba(0, 0, 0, 0.2); /* Semi-transparent
background */
            border-radius: 25px; /* Rounded corners */
            display: inline-block;
            margin-top: 10px;
        }

        nav ul li {
            display: inline;
            margin: 0;
        }

        nav ul li a {
            color: white;
            text-decoration: none;
            padding: 0.7em 1.5em;
            display: inline-block;
            border-radius: 25px;
            transition: background-color 0.3s, transform 0.3s;
        }

        nav ul li a:hover {
            background-color: #006666; /* Darker teal color */
            transform: scale(1.05); /* Slight scaling effect */
        }

        main {
            padding: 2em;
            display: none;
            max-width: 1200px;
            margin: 0 auto;
        }

        main h2 {
            font-size: 2em;
```

```
            margin-bottom: 0.5em;
            color: #333;
        }

        main p {
            font-size: 1.2em;
            color: #555;
        }

        main ul {
            list-style-type: disc;
            padding-left: 20px;
            font-size: 1.1em;
            color: #555;
        }

        footer {
            background: linear-gradient(135deg, #008080, #20b2aa); /*
Gradient background with teal colors */
            color: white;
            text-align: center;
            padding: 1em 0;
            width: 100%;
            box-shadow: 0 -4px 8px rgba(0, 0, 0, 0.2); /* Shadow for
depth */
            margin-top: 20px; /* Ensure there's a margin between the
content and the footer */
        }

        .active {
            display: block;
        }

        .iframe-container {
            position: relative;
            padding-bottom: 56.25%; /* 16:9 Aspect Ratio */
            height: 0;
            overflow: hidden;
            max-width: 100%;
            background: #000;
            border-radius: 8px;
        }

        .iframe-container iframe {
            position: absolute;
            top: 0;
            left: 0;
            width: 100%;
            height: 100%;
            border: 0;
        }

        #lottie-container {
            width: 100%;
            height: 400px;
            margin: 2em 0;
            float: right; /* Align the animation to the right */
```

```
        }

        #content-container {
            display: flex;
            justify-content: space-between;
        }

        #text-content {
            flex: 1;
        }

        /* Remove the scrollbar styling */
        /* If you had previous scrollbar styling, remove it to ensure no
extra space is used */
        .scrollable-container {
            height: 600px; /* Adjust height as needed */
            overflow: hidden; /* Hide overflow */
        }

        /* New CSS class for small font */
        .small-font {
            font-size: 0.9em; /* Adjust size as needed */
        }
    </style>
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/bodymovin/5.7.6/lottie.min.j
s"></script>
</head>
<body>
    <header>
        <h1>Health Insurance Price Prediction</h1>
        <nav>
            <ul>
                <li><a href="#"
onclick="showSection('home')">Home</a></li>
                <li><a href="#"
onclick="showSection('predictor')">Predictor</a></li>
                <li><a href="#"
onclick="showSection('benefits')">Benefits</a></li>
            </ul>
        </nav>
    </header>
    <main id="home" class="active">
        <div id="content-container">
            <div id="text-content">
                <h2>Welcome to Our Health Insurance Premium Prediction
Tool</h2>
                <p>Explore our tool to understand how different factors
influence health insurance premiums and make informed decisions.</p>
            </div>
            <div id="lottie-container"></div>
        </div>
        <footer>
            <p>&copy; 2024 Health Insurance Premium Prediction
Project</p>
        </footer>
    </main>
```

```
    <main id="predictor">
        <h2>Predict Your Health Insurance Premium</h2>
        <p class="small-font">Use this tool to enter your details and
predict your health insurance premium. We use advanced algorithms to
provide accurate estimates based on your inputs.</p>
        <!-- Ensure the content fits within the available space -->
        <div class="iframe-container">
            <iframe src="http://localhost:8501" width="100%"
height="100%" frameborder="0"></iframe>
        </div>
    </main>
    <main id="benefits">
        <h2>Benefits of Using Our Prediction Tool</h2>
        <ul>
            <li>Accurate Estimates: Get precise predictions based on
various factors.</li>
            <li>Informed Decisions: Understand how different factors
affect your premiums.</li>
            <li>Cost Savings: Plan better and potentially save on
insurance costs.</li>
        </ul>
    </main>
    <script>
        function showSection(sectionId) {
            // Hide all sections
            document.querySelectorAll('main').forEach(function(section)
{
                section.classList.remove('active');
            });
            // Show the selected section
            document.getElementById(sectionId).classList.add('active');
        }

        // Load Lottie animation
        document.addEventListener("DOMContentLoaded", function() {
            var animation = lottie.loadAnimation({
                container: document.getElementById('lottie-container'),
// The container where the animation will be rendered
                renderer: 'svg', // Render as SVG
                loop: true, // Loop the animation
                autoplay: true, // Start playing immediately
                path: 'https://lottie.host/39998654-8679-4adc-b317-
79eab5b9dbbd/cOPHCBxrsf.json' // Path to your Optimized Lottie JSON file
            });
        });
    </script>
</body>
</html>
```