**Methodology**

**1. Introduction**

Heart disease is one of the leading causes of death worldwide, making early detection and intervention critical for improving patient outcomes. Leveraging machine learning techniques can enhance our ability to predict the risk of heart disease based on various health and demographic indicators. This project aims to develop a predictive model using a comprehensive dataset from the [Behavioral Risk Factor Surveillance System (BRFSS) 2015](https://www.kaggle.com/datasets/cdc/behavioral-risk-factor-surveillance-system/data?select=2015.csv) survey. The dataset contains a wealth of information on health-related risk behaviors, chronic health conditions, and use of preventive services.

The methodology outlined here involves several key steps: data preparation, exploratory data analysis (EDA), model training, and model evaluation. Each step is carefully designed to ensure the model is both accurate and generalizable. By identifying significant predictors of heart disease, the model can help in designing targeted interventions and informing public health strategies.

**2. Dataset Description**

The dataset used in this project is derived from the 2015 BRFSS survey, which collects data on health-related risk behaviors, chronic health conditions, and use of preventive services among U.S. adults. The original dataset includes 330 features and 400k plus survey responses. However, not all features are relevant to heart disease prediction. Therefore, based on extensive research, a subset of key features has been selected for this analysis.

|  |
| --- |
|  |

Dataset Overview

**Key Features and Risk Factors:**

Research in the field of cardiology and epidemiology has identified several important risk factors for heart disease. These risk factors include, but are not limited to, high blood pressure, high cholesterol, smoking, diabetes, obesity, age, sex, race, diet, exercise, alcohol consumption, body mass index (BMI), household income, marital status, sleep, time since last medical checkup, education level, healthcare coverage, and mental health status. The selected features from the BRFSS 2015 dataset encompass these risk factors, providing a robust basis for predictive modeling.

|  |
| --- |
|  |

Columns reduce to 22, (Important Feature Extracted)

**Selected Subset of Features:**

To align with the identified risk factors, the following features have been selected from the BRFSS 2015 dataset:

* **High Blood Pressure:** Indicator of whether an individual has been told they have high blood pressure.
* **High Cholesterol:** Includes whether an individual has been told they have high cholesterol and whether they have had a cholesterol check in the past five years.
* **BMI:** Body Mass Index, calculated from self-reported height and weight.
* **Smoking:** Indicates whether an individual has smoked at least 100 cigarettes in their lifetime.
* **Other Chronic Health Conditions**: Includes whether an individual has had a stroke or has diabetes.
* **Physical Activity:** Reports if an individual has engaged in physical activity or exercise other than their regular job in the past 30 days.
* **Diet:** Frequency of fruit and vegetable consumption.
* **Alcohol Consumption:** Reports heavy drinking patterns, defined as more than 14 drinks per week for men and more than 7 drinks per week for women.
* **Healthcare Access:** Includes health care coverage status and whether there was a time in the past 12 months when the individual needed to see a doctor but could not because of cost.
* **General and Mental Health:** Self-reported general health status and the number of days in the past 30 days when mental or physical health was not good.
* **Difficulty Walking:** Reports serious difficulty in walking or climbing stairs.
* **Demographics:** Includes sex, age category, education level, and household income.

**Response Variable:**

The response variable, `\_MICHD`, indicates whether the respondent has ever been told by a health professional that they have coronary heart disease (CHD) or a myocardial infarction (MI). This binary variable serves as the target for our predictive model.

1. **Data Preparation and Preprocessing:**

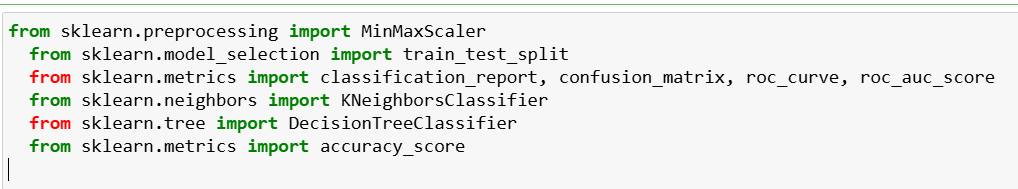
**Importing Necessary Libraries and Data Loading**

The objective of this step is to establish the necessary computational environment by importing essential Python libraries and loading the dataset into a Pandas DataFrame for subsequent analysis.

**1.1 Importing Libraries:**

To facilitate data manipulation, visualization, and machine learning model development, several specialized Python libraries were employed. The following libraries were imported:

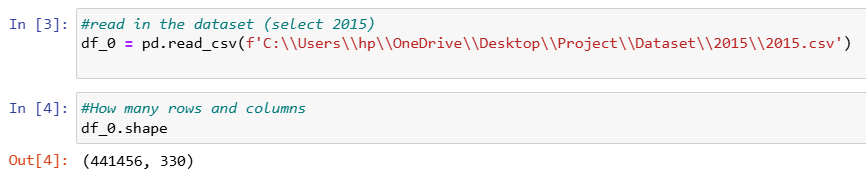
* Pandas: This library is pivotal for data manipulation and analysis, providing data structures and functions designed to handle structured data efficiently.
* NumPy: As a fundamental package for scientific computing in Python, NumPy offers support for arrays and matrices, along with a rich collection of mathematical functions to operate on these arrays.
* Matplotlib: This plotting library is utilized for creating static, animated, and interactive visualizations in Python. It is particularly effective for generating plots and charts.
* Seaborn: Built on top of Matplotlib, Seaborn provides a high-level interface for drawing attractive and informative statistical graphics.
* Scikit-learn: This machine learning library in Python offers simple and efficient tools for data mining and data analysis. It includes modules for model training, evaluation, and preprocessing. The following components were imported:



* SciPy: This library is used for scientific and technical computing, building on NumPy to provide a large number of functions for operating on NumPy arrays.
* Imbalanced-learn (imblearn): This library addresses imbalanced datasets, offering tools for over-sampling, under-sampling, and various techniques to balance the dataset.

**1.2 Loading the Dataset:**

With the necessary libraries imported, the next step involved loading the dataset into a Pandas DataFrame. This structured format allows for efficient data manipulation and analysis.

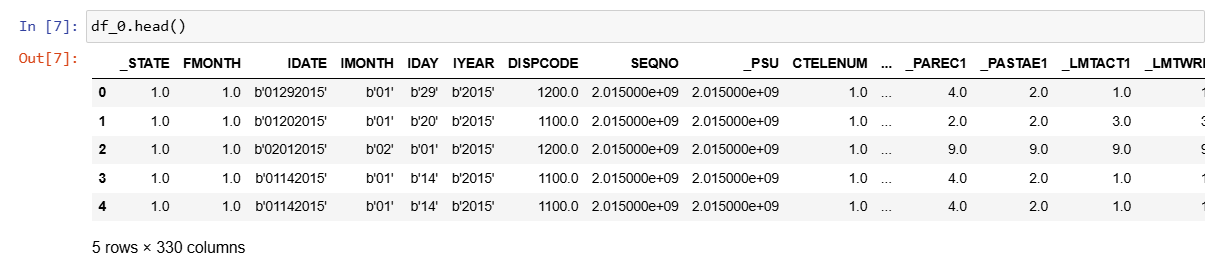


As from the above pic we can clearly see the shape (441456, 330) of the dataframe.

**1.3 Initial Data Inspection:**

Following the loading of the dataset, an initial inspection was conducted to understand its structure and content. This involved displaying the first few rows of the DataFrame, checking the data types of each column, and identifying any missing values or inconsistencies. The methods used for this inspection included:

* Displaying the first five rows of the dataset:



* Obtaining summary statistics of the numerical columns:



**1.4 Removing Duplicates:**

After duplicates removal, we have now 229781 rows and 22 columns.

|  |
| --- |
|  |

1. **Exploratory Data Analysis (EDA):**

**1: Feature Selection**

Based on research and relevance, specific features are selected from the dataset. These features include both the response variable (indicating whether the respondent has reported coronary heart disease or myocardial infarction) and the predictor variables (such as high blood pressure, high cholesterol, BMI, smoking status, etc). The feature we are selecting for our project are 22.

**2: Data Cleaning**

Data cleaning involves checking for and handling any missing values and duplicates. Rows with missing values are either removed. Additionally, columns that are not relevant to the analysis are dropped. This ensures that the dataset is clean and ready for analysis.

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **Before Dropping Missing Values** | **After Dropping Missing Values** |
|  |  |

Modifying and cleaning the values to be more suitable to ML algorithms:

In order to do this part, I referenced the codebook which says what each column/feature/question is: <https://www.cdc.gov/brfss/annual_data/2015/pdf/codebook15_llcp.pdf>.

|  |
| --- |
|  |

After modifying and cleaning the values the dataset looks like:

|  |
| --- |
|  |

|  |
| --- |
|  |

**3. Make feature names more readable:**

To enhance readability, we renamed several dataset columns with more descriptive names. For example, '\_MICHD' was renamed to 'HeartDiseaseorAttack', '\_RFHYPE5' to 'HighBP', and 'DIABETE3' to 'Diabetes'. This makes the dataset more user-friendly and easier to analyze.

Other key renamings include '\_BMI5' to 'BMI', 'SMOKE100' to 'Smoker', 'CVDSTRK3' to 'Stroke', and 'GENHLTH' to 'GenHlth'. These changes help in better understanding the data, facilitating more effective analysis.

|  |
| --- |
|  |

*Now the columns are more readable:*

|  |
| --- |
|  |

**4. Correlation Matrix:**

|  |
| --- |
|  |

The provided correlation matrix shows the pairwise correlation coefficients between various health-related variables. Correlation coefficients range from -1 to 1, where:

* 1 indicates a perfect positive correlation.
* -1 indicates a perfect negative correlation.
* 0 indicates no correlation.

Here’s a summary of key insights from the matrix:

1. **Heart Disease or Attack:**

Positively correlated with **High BP** (0.209), **High Chol** (0.181), b (0.203), and **Age** (0.222).

Negatively correlated with **Diabetes** (-0.162) and **PhysActivity** (-0.087).

1. **High Blood Pressure (HighBP):**

Strongly positively correlated with **Age** (0.344) and moderately with **High Chol** (0.298), **GenHlth** (0.301), and **DiffWalk** (0.224).

Negatively correlated with **PhysActivit**y (-0.125) and **Diabetes** (-0.238).

1. **High Cholesterol (HighChol):**

Moderately positively correlated with **HighBP** (0.298), **HeartDiseaseorAttack** (0.181), and **GenHlth** (0.208).

Negatively correlated with **PhysActivity** (-0.078) and **Diabetes** (-0.178).

1. **Cholesterol Check (CholCheck):**

Shows weak correlations with most variables, slightly positively correlated with **AnyHealthcare** (0.118) and **Age** (0.090).

Negatively correlated with **NoDocbcCost** (-0.058).

1. **Body Mass Index (BMI):**

Positively correlated with **HighBP** (0.214) and **GenHlth** (0.239).

Negatively correlated with **Diabetes** (-0.198) and **PhysActivity** (-0.147).

1. **Smoker:**

Positively correlated with **HeartDiseaseorAttack** (0.114), **HighBP** (0.097), and **GenHlth** (0.163).

Negatively correlated with **PhysActivity** (-0.087).

1. **Stroke:**

Positively correlated with **HeartDiseaseorAttack** (0.203), **HighBP** (0.130), and **GenHlth** (0.178).

Negatively correlated with **Diabetes** (-0.097).

1. **Diabetes:**

Negatively correlated with **HeartDiseaseorAttack** (-0.162), **HighBP** (-0.238), **HighChol** (-0.178), **GenHlth** (-0.267), and **DiffWalk** (-0.198).

Positively correlated with **BMI** (0.106) and **PhysActivity** (0.107).

1. **Physical Activity (PhysActivity):**

Negatively correlated with \*\*HighBP\*\* (-0.125), \*\*HeartDiseaseorAttack\*\* (-0.087), and GenHlth\*\* (-0.266).

1. **General Health (GenHlth):**

Strongly positively correlated with **PhysHlth** (0.525) and **DiffWalk** (0.457).

Negatively correlated with **Diabetes** (-0.267), **PhysActivity** (-0.266), and **Income** (-0.370).

1. **Age:**

Positively correlated with **HighBP** (0.344), **HeartDiseaseorAttack** (0.222), and **HighChol** (0.272).

Negatively correlated with **Income** (-0.128).

1. **Education and Income:**

Positively correlated with each other (0.449).

Generally, show negative correlations with most health issues, indicating that higher education and income levels are associated with better health outcomes.

This matrix helps identify relationships between health factors, which can be valuable for health professionals and researchers in understanding risk factors and targeting interventions.

**5. Data Visualization:**

**Histograms:**

We employed histograms to explore the distribution of key health-related and demographic variables within the dataset.

|  |  |
| --- | --- |
| BMI | Age |
| HeartDiseaseorAttack | GenHlth |
| Education | Income |
| MentHlth |  |

*Interpretation of Histograms for Key Variables:*

**BMI:**

The histogram for BMI shows that most respondents have BMI values in the lower to mid-range, with the distribution slightly right-skewed. The average BMI is 28.38, indicating that most respondents fall into the overweight category. BMI values range from 12 to 98, revealing a wide variation from underweight to extremely obese. The presence of very high BMI values suggests potential outliers that may need further examination.

**Age:**

The age histogram illustrates the distribution of respondents across different age groups. The distribution appears multimodal, reflecting the different encoded age ranges. The mean age group is around 55-59 years, showing that most respondents are middle-aged. The ages span from 18-24 to 80+, indicating a broad age range. The standard deviation of 3.05 highlights considerable age variation, providing a diverse demographic for analyzing the interaction between age and health variables.

**Heart Disease or Attack:**

The histogram for heart disease or heart attack incidence shows that the majority of respondents do not have a history of heart disease or heart attack, as indicated by a large peak at zero. This binary feature has relatively few respondents with a value of one, aligning with the overall mean of 9.4%, suggesting a lower prevalence of heart disease within the sample.

**General Health:**

The general health histogram uses a scale from 1 (excellent) to 5 (poor). Most respondents rate their health around the mid-range value of 2.5, indicating a general perception of fair to good health. The distribution suggests a fairly even spread across the ratings, with some skew towards better health ratings.

**Education:**

The education histogram shows the distribution of respondents' highest education level, encoded from 1 (never attended school) to 6 (college graduate). The mean value of around 5 indicates that many respondents have some college education or are college graduates. The distribution is slightly left-skewed, suggesting higher educational attainment among the respondents.

**Income:**

The income histogram displays the distribution of respondents' income levels, encoded from 1 (less than $10,000) to 8 ($75,000 or more). The mean income level is around 6, suggesting that many respondents fall into the moderate to high-income brackets. The distribution is slightly right-skewed, indicating a greater number of respondents in higher income categories.

**Mental Health:**

The histogram for mental health, representing the number of days with poor mental health in the past month, shows that most respondents report very few days of poor mental health, with a significant peak at zero. The average of about 3.2 days indicates that while the majority experience few issues, there is a notable portion of the population with frequent poor mental health days. The high variability suggests a diverse range of mental health experiences among respondents.

**Distribution of Data:**

|  |
| --- |
|  |

**Normalized Distribution (Gaussian):**

To determine whether the data distributions of each variable approximate a normal (Gaussian) distribution, we conducted a normality test using the D'Agostino and Pearson's test. This test combines skew and kurtosis to provide an omnibus test of normality. The results include the k-squared (k2) statistic and the p-value for each variable in the dataset.

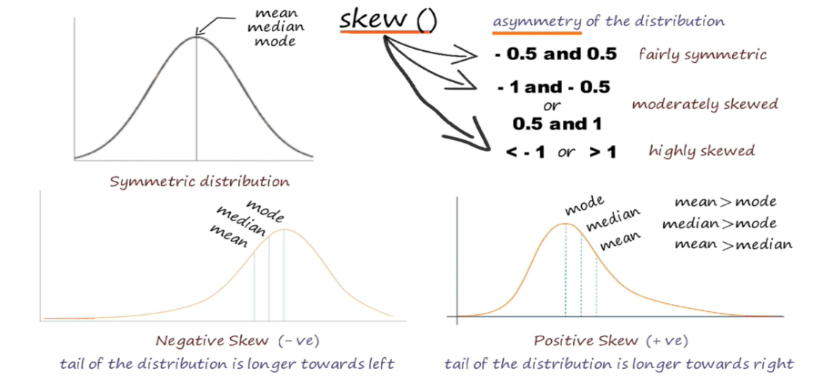
The output shows the k2 statistic and the p-value for each variable:

|  |
| --- |
|  |

Each p-value is 0.0 (to the precision reported), indicating that the null hypothesis (that the data follows a normal distribution) is rejected for all variables. This means that none of the variables in the dataset follow a normal distribution, as per the normality test. These results suggest that the data distributions are significantly different from a Gaussian distribution, which is important to consider for any subsequent statistical analysis or modeling that assumes normality.

**Skewed Distribution Analysis:**

Using the D'Agostino and Pearson's test, we assessed the normality of each variable's distribution. The results indicate significant deviations from normality, with all variables showing p-values of 0.0. This means the data distributions are skewed. Skewness refers to the asymmetry in the distribution of data. A positively skewed distribution has a long tail on the right, while a negatively skewed distribution has a long tail on the left. Recognizing skewness is essential because it impacts the choice of statistical methods and data transformations required for accurate analysis and interpretation.



|  |
| --- |
|  |

**Outliers Detection:**

In our data analysis process, we utilized the Z-score method to detect and subsequently remove outliers specifically from the BMI column. This approach involved calculating the Z-scores for each BMI data point, measuring their deviation from the mean BMI value in terms of standard deviations. By applying a predetermined Z-score threshold, set at 2 in this case, we identified BMI data points that fell beyond this threshold as outliers.

|  |
| --- |
| **Count of Outliers for Each Column** |

Following outlier detection, we proceeded to remove these identified outliers exclusively from the BMI column. The removal process resulted in a refined dataset with outlier-free BMI values, ensuring that extreme data points were effectively addressed and mitigated. This targeted outlier removal strategy was instrumental in enhancing the robustness and reliability of our dataset, particularly in relation to the BMI feature, thereby contributing to the integrity of our subsequent analyses and model development endeavors.

Visualizations, including box plots, were then employed to visually inspect the distribution of the data and pinpoint the exact location of outliers within the dataset. This meticulous approach not only facilitated a comprehensive understanding of the dataset's integrity but also informed subsequent data preprocessing and analysis steps.

|  |
| --- |
| Boxplot Before Outliers Removal |

After removal of outliers the boxplot looks like as shown below;

|  |
| --- |
| C:\Users\lenovo\OneDrive\Desktop\boxplot_01.png  After removal of outliers |

Distribution of BMI Column:

|  |  |
| --- | --- |
|  |  |

Skewness of BMI column:

|  |  |
| --- | --- |
|  |  |

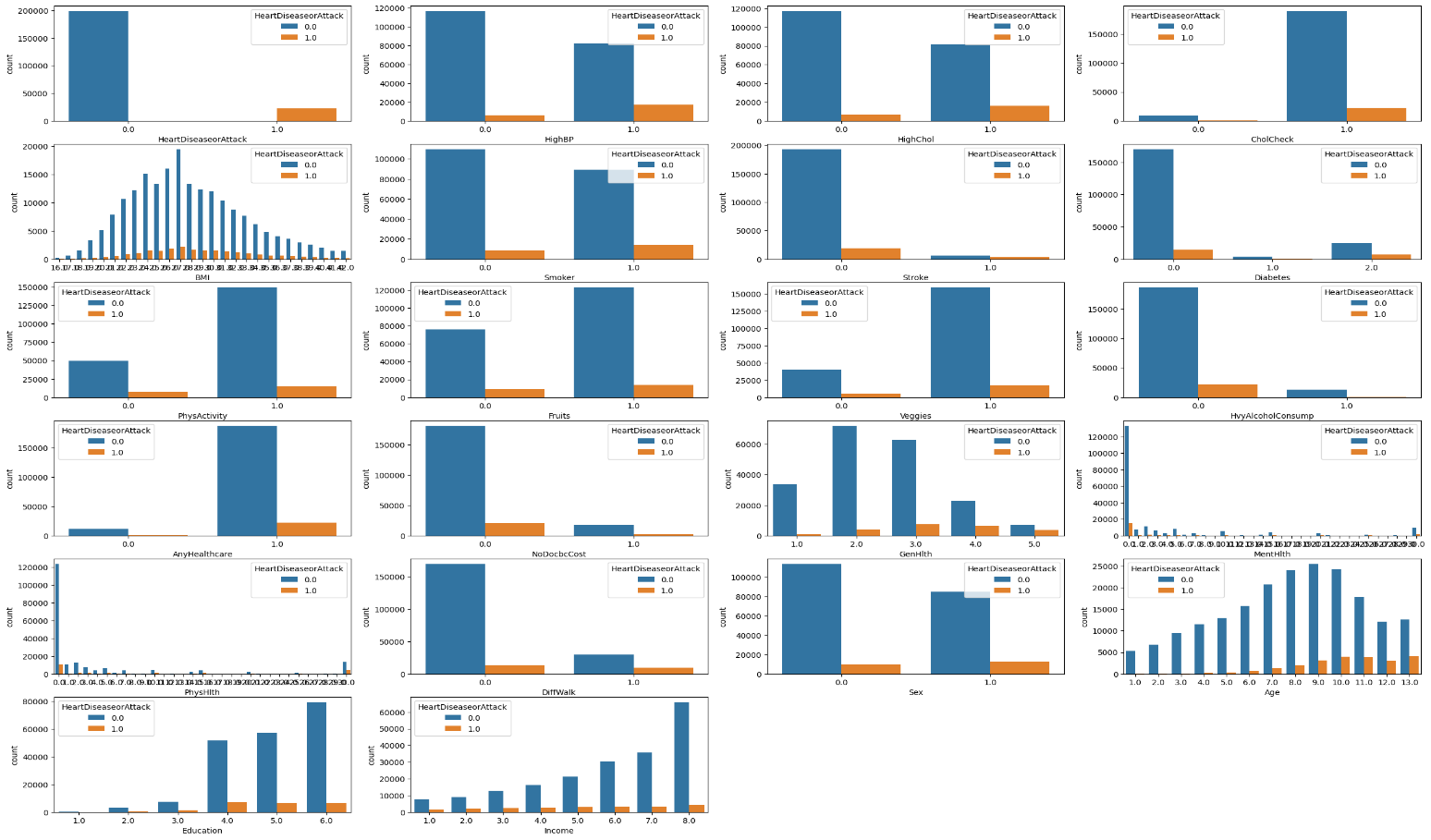
As we see the skewness has been decreased after removal of outliers.

**Removal of Outliers:**

|  |
| --- |
|  |

After removing outliers from the BMI column using the Z-score method with a threshold of 2, we refined our dataset significantly. Initially, the dataset comprised 229,781 observations across 22 features. By applying the outlier removal function, we filtered out extreme BMI values, resulting in a cleaned dataset with 221,401 observations. This process effectively excluded data points with BMI values beyond the range of approximately 16 to 42. Post-cleaning, the descriptive statistics of the BMI column showed a mean of 27.91, a standard deviation of 5.17, and quartiles at 24, 27, and 31, respectively. The removal of outliers ensured a more accurate and reliable dataset for subsequent analyses and model development.

**Count Plots and Their Importance:**



In our analysis, we utilized count plots to visualize the distribution of various features in the filtered dataset, segmented by the incidence of heart disease or attack. Each subplot represents a different feature, enabling us to observe the relationship between feature categories and heart disease presence. By differentiating cases with the "HeartDiseaseorAttack" hue, we gained insights into potential risk factors and correlations, such as the incidence of heart disease across different age groups, BMI categories, and levels of physical activity. These plots revealed data distribution patterns, helped identify feature significance, and highlighted potential imbalances, providing a foundation for informed feature selection and robust model development in our predictive analysis.

1. **Data Preprocessing:**

**Split the dataset into features (X) and labels (y):**

The dataset was divided into a feature matrix (X) by removing the 'HeartDiseaseorAttack' column and a target vector (y) containing the 'HeartDiseaseorAttack' values. This separation is crucial for supervised learning tasks where the model needs to learn from the features to predict the target variable.

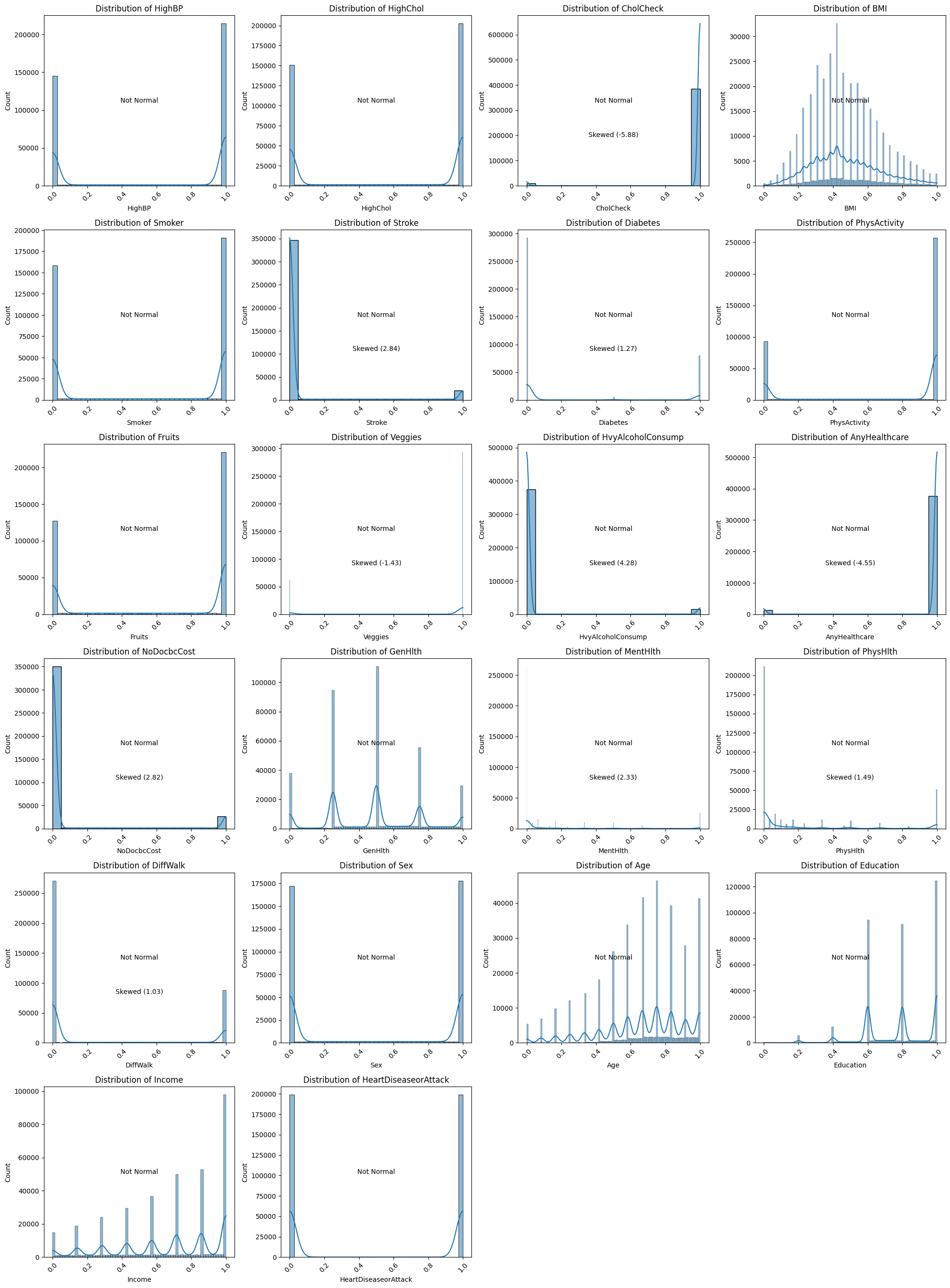
**Apply techniques like SMOTE to handle class imbalance if applicable:**

To address the class imbalance in the target variable, the Synthetic Minority Oversampling Technique (SMOTE) was employed. SMOTE generates synthetic samples for the minority class, ensuring an equal distribution of classes. This process increased the dataset to 397,454 observations, with an equal 50-50 split between the classes(0 and 1).

|  |
| --- |
|  |

**Perform feature scaling using normalization techniques such as Min-Max scaling:**

Feature scaling was performed using Min-Max scaling to normalize the feature values to a range between 0 and 1. This step is essential to ensure that all features contribute equally to the model training process and to improve the convergence rate of the algorithms.



**Split the data into training and testing subsets**:

The normalized data was then split into training and testing subsets using the train\_test\_split() function, with 70% of the data allocated for training and 30% for testing. This division allows for the evaluation of the model's performance on unseen data, ensuring that the model generalizes well to new data. The training set comprised 278,217 observations, while the testing set included 119,237 observations.

|  |
| --- |
|  |

1. **Model Building and Stacking:**

In this section, we developed and evaluated individual machine learning models using various algorithms. Following the training and evaluation of these models, we implemented stacking techniques to combine models for potentially improved performance.

Firstly, we built individual models using K-Nearest Neighbors (KNN), Decision Trees, Random Forest, Logistic Regression, and Naive Bayes algorithms. The KNN model, configured with 10 neighbors, achieved an accuracy of 86%, demonstrating balanced precision and recall for both classes. The Decision Tree models, using both Gini Impurity and Entropy criteria, showed a testing accuracy of 91%. Both criteria provided consistent precision, recall, and F1 scores, with a ROC AUC of 91%. The Random Forest classifier outperformed the individual models with a high testing accuracy of 94% and a ROC AUC of 97%, indicating strong discriminative power. Logistic Regression, though slightly less accurate with a testing accuracy of 77%, provided a balanced performance with a ROC AUC of 85%. Lastly, the Naive Bayes model achieved a testing accuracy of 74% and a ROC AUC of 80%, with reasonable precision and recall metrics.

To potentially enhance performance, we combined models using stacking techniques. The stacking approach involved training individual models (base models) first and using their predictions as input features for a meta-model, which in our case was Logistic Regression. The combination of Random Forest and Decision Tree classifiers with Logistic Regression as the meta-classifier achieved a near-perfect training accuracy of 99.8% and a high testing accuracy of 94%, with an impressive ROC AUC of 98%. Similarly, the combination of Random Forest and Naive Bayes classifiers achieved the same high testing accuracy and ROC AUC, confirming the robustness and effectiveness of the stacking technique. These stacked models demonstrated excellent performance in predicting heart disease, with high precision, recall, and F1 scores, making them reliable and robust tools for clinical applications in identifying at-risk patients.

1. **Model Evaluation:**

In this section, we evaluate the performance of the developed machine learning models using various metrics, compare individual and stacked models, and present the results and insights gained from the evaluation process.

To assess model performance, we employed metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. These metrics provide a comprehensive understanding of how well each model performs in classifying instances correctly. Let’s Evaluate each model performance in detail..

1. **Accuracy**: Measures the proportion of correctly classified instances out of the total instances.
2. **Precision**: Indicates the proportion of true positive predictions out of all positive predictions made by the model.
3. **Recall**: Represents the proportion of true positive predictions out of all actual positive instances.
4. **F1**-**score**: The harmonic mean of precision and recall, providing a single metric that balances both.
5. **ROC-AUC**: The area under the Receiver Operating Characteristic curve, illustrating the model's ability to discriminate between positive and negative classes.
6. **K-Nearest Neighbors (KNN)**

Training and Testing Accuracy:

|  |
| --- |
| Training Accuracy: 0.8905566518221388 |
| Testing Accuracy: 0.8581648313862308 |

Confusion Matrix:

|  |
| --- |
|  |

ROC Curve:

|  |
| --- |
|  |

Accuracy vs. K-value:

|  |
| --- |
|  |

Error Rate vs. K-value:

|  |
| --- |
|  |

The KNN model, configured with 10 neighbors, demonstrates strong performance with an overall accuracy of 86%. The confusion matrix reveals balanced precision and recall for both classes, with 85% precision and recall for class 0 and 86% precision and recall for class 1. This balance is reflected in the F1-score of 0.86 for both classes, indicating robust performance in correctly classifying instances. These results highlight the model's effectiveness in accurately predicting both positive and negative instances, making it a reliable tool for classification tasks.

1. **Decision Trees**

*Decision Tree Models Accuracies and Classification Reports:*

|  |  |
| --- | --- |
|  |  |

ROC AUC: 91%

|  |
| --- |
|  |

Confusion Matrix:

|  |
| --- |
|  |

The Decision Tree models, employing Gini Impurity and Entropy criteria, demonstrate remarkable accuracy, with both achieving a testing accuracy of 91% while maintaining a training accuracy of 100% and 91%, respectively. Precision, recall, and F1 scores are consistent across both models, reflecting balanced performance in classifying instances of both classes. Additionally, their ROC AUC values of 91% underscore their robust discriminatory power. Minimal disparities are observed in the confusion matrices, indicating consistent performance. In conclusion, both models exhibit comparable effectiveness in classification tasks, with the choice between Gini Impurity and Entropy criterion potentially influenced by factors such as interpretability or computational efficiency, given their largely indistinguishable predictive capabilities.

1. **Random Forest**

Training and Testing Accuracy:

|  |
| --- |
|  |

Confusion Matrix:

|  |
| --- |
|  |

Classification Report:

|  |
| --- |
|  |

ROC Curve:

|  |
| --- |
|  |

The Random Forest classifier achieved a high testing accuracy of 0.94, demonstrating strong performance in predicting heart disease. The model showed high precision and recall for both classes, with a balanced F1-score. The feature importance analysis revealed that high blood pressure, high cholesterol, and age are significant predictors of heart disease. This model's robustness and interpretability make it a valuable tool for healthcare professionals in identifying at-risk patients.

1. **Logistic Regression**

Testing and Training accuracy:

|  |
| --- |
|  |

Confusion Matrix:

|  |
| --- |
|  |

Logistic Regression Classification Report:

|  |
| --- |
|  |

The Logistic Regression model, used for predicting heart disease, demonstrates a balanced performance with both training and testing accuracy at 77%. The model's confusion matrix reveals a precision of 79% for non-disease cases (class 0) and 76% for disease cases (class 1), alongside recall values of 74% and 80%, respectively. The F1-scores are 77% for non-disease cases and 78% for disease cases, indicating effective handling of both false positives and false negatives. With a ROC curve area of 0.85, the model shows good capability in distinguishing between patients with and without heart disease.

1. **Naive Bayes**

Naive Bayes Confusion Matrix:

|  |
| --- |
|  |

ROC Curve:

|  |
| --- |
|  |

Naïve Bayes Classification Report:

|  |
| --- |
|  |

The Naive Bayes model, specifically Gaussian Naive Bayes, achieves a testing accuracy of 74%, with a training accuracy matching this value. Precision, recall, and F1 scores are 77%, 69%, and 73% for class 0, and 72%, 80%, and 76% for class 1, respectively. Despite slightly imbalanced precision and recall values, the model demonstrates an overall balanced performance in classifying instances. The ROC curve area of 0.80 indicates moderate discriminatory power. The confusion matrix illustrates higher false negatives for class 0 and higher false positives for class 1, highlighting areas where the model could be further optimized for improved performance. Overall, the Naive Bayes model offers a moderate level of accuracy and may benefit from adjustments to address class imbalances and optimize classification outcomes.

**II. Implementing Stacking**

We will combine the models using stacking techniques. For simplicity, we will use Logistic Regression as the meta-model to combine the base models.

1. **RF + NB**

|  |
| --- |
| Training Accuracy: 0.9981381439667597 |
| Testing Accuracy: 0.9394483256036298 |

Training and testing Evaluation:

|  |  |
| --- | --- |
|  |  |

Confusion Matrix:

|  |
| --- |
|  |

**ROC Curve:**

|  |
| --- |
|  |

The stacked model, combining Random Forest and Naive Bayes classifiers with Logistic Regression as the meta-classifier, demonstrates exceptional performance in predicting heart disease. It achieved near-perfect accuracy on the training data (99.81%) and maintained high accuracy on the testing data (93.94%). The model's F1 score (93.78%), precision (96.13%), and recall (91.55%) on the test set indicate its effectiveness in correctly identifying both positive and negative cases of heart disease. Furthermore, the ROC curve and AUC score of 0.98 confirm the model's strong discriminative ability. Overall, this stacked model serves as a reliable and robust tool for heart disease prediction, suitable for clinical applications.

1. **RF + DT**

|  |
| --- |
| Training Accuracy: 0.9981345496500933 |
| Testing Accuracy: 0.939322525726075 |

Classification Report:

|  |  |
| --- | --- |
|  |  |

Confusion Matrix:

|  |
| --- |
|  |

ROC Curve:

|  |
| --- |
|  |

The stacked model combining Random Forest and Decision Tree classifiers, with Logistic Regression as the meta-classifier, demonstrates strong performance in predicting heart disease. On the training data, it achieves nearly perfect accuracy (0.998), indicating an excellent fit. For the testing data, the model maintains a high accuracy (0.939), showcasing its generalization ability. Key performance metrics such as F1 score (0.937), precision (0.96), and recall (0.91) on the test set underline its balanced effectiveness in identifying both positive and negative cases. The ROC curve of 0.98 further validate the model's robustness and reliability. This stacked model thus offers a potent solution for heart disease prediction, combining the strengths of both Random Forest and Decision Tree classifiers to enhance prediction accuracy and reliability in clinical settings.

|  |
| --- |
| **Models Comparison** |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Model** | **Accuracy** | **Precision (Class 0)** | **Precision (Class 1)** | **Recall (Class 0)** | **Recall (Class 1)** | **F1-score (Class 0)** | **F1-score (Class 1)** | **ROC-AUC** | | K-Nearest Neighbors (KNN) | 0.86 | 0.85 | 0.86 | 0.86 | 0.85 | 0.86 | 0.86 | 0.86 | | Decision Tree (Gini) | 0.91 | 0.91 | 0.9 | 0.9 | 0.91 | 0.9 | 0.91 | 0.91 | | Decision Tree (Entropy) | 0.91 | 0.91 | 0.9 | 0.9 | 0.91 | 0.9 | 0.91 | 0.91 | | Random Forest | 0.94 | 0.91 | 0.97 | 0.98 | 0.91 | 0.94 | 0.94 | 0.98 | | Logistic Regression | 0.77 | 0.79 | 0.76 | 0.74 | 0.8 | 0.77 | 0.78 | 0.85 | | Naive Bayes | 0.74 | 0.77 | 0.72 | 0.69 | 0.8 | 0.73 | 0.76 | 0.801 | | Stacked Model (RF + DT) | 0.94 | 0.92 | 0.96 | 0.96 | 0.92 | 0.94 | 0.94 | 0.98 | | Stacked Model (RF + NB) | 0.94 | 0.92 | 0.96 | 0.96 | 0.92 | 0.94 | 0.94 | 0.98 | |