# MACHINE LEARNING
## ASSIGNMENT

**Heart Harmony_Orchestrating Precision in Predictive Health Analytics**



**PREPARED FOR**
Upgrad TA Priya Sharma Ma'am
**PREPARED BY**
Vishu's Team

## OVERVIEW OF DATASET :

This project utilises a dataset sourced from the Centers for Disease Control and Prevention (CDC), initially comprising about 300 variables. For practicality, it has been streamlined to 40 variables in this dataset. For this project, please use the aforementioned link. You may use the one with NaN values. A notable aspect of this dataset is its class imbalance, which could hinder the effectiveness of standard algorithms and pose a risk of bias in analyses. To enhance the outcomes, an undersampling technique or fixing the weights is recommended. This dataset offers a realistic simulation of real-world data analysis challenges.

## DATA DESCRIPTION:

Please find the description of all the variables present in the dataset. The designated target variable for the analyses should be the 'HadHeartAttack' column.

1. **State:** Geographic location where the individual resides.
2. **Sex:** The biological sex of the individual.
3. **GeneralHealth:** An overall assessment of the individual's health status.
4. **PhysicalHealthDays:** Number of days of poor physical health reported.
5. **MentalHealthDays:** Number of days of poor mental health reported.
6. **LastCheckupTime:** The last time the individual had a medical checkup.
7. **Physical Activities:** Indication of physical activity levels.
8. **SleepHours:** Average number of hours the individual sleeps.
9. **RemovedTeeth:** Whether the individual has had teeth removed.
10. **HadHeartAttack:** Whether the individual has had a heart attack.
11. **HadAngina:** Whether the individual has had angina.
12. **HadStroke:** Whether the individual has had a stroke.
13. **HadAsthma:** Whether the individual has asthma.
14. **HadSkinCancer:** Whether the individual has had skin cancer.

1. **HadCOPD:** Whether the individual has Chronic Obstructive Pulmonary Disease.
2. **HadDepressiveDisorder:** Whether the individual has been diagnosed with depression.
3. **HadKidneyDisease:** Whether the individual has kidney disease.
4. **HadArthritis:** Whether the individual has arthritis.
5. **HadDiabetes:** Whether the individual has diabetes.
6. **DeafOrHardOfHearing:** Hearing difficulties.
7. **BlindOrVisionDifficulty:** Vision difficulties.
8. **DifficultyConcentrating:** If the individual has difficulty concentrating.
9. **Difficulty Walking:** If the individual has difficulty walking.
10. **DifficultyDressingBathing:** If the individual has difficulty dressing or bathing.
11. **DifficultyErrands:** If the individual has difficulty doing errands alone.
12. **SmokerStatus:** The smoking status of the individual.
13. **ECigaretteUsage:** Usage of electronic cigarettes.
14. **ChestScan:** If the individual has had a chest scan.
15. **RaceEthnicityCategory:** The race/ethnicity of the individual.
16. **AgeCategory:** The age range category of the individual.
17. **HeightInMeters:** The individual's height in meters.
18. **WeightInKilograms:** The individual's weight in kilograms.
19. **BMI:** Body Mass Index.
20. **AlcoholDrinkers:** Alcohol consumption status.
21. **HIVTesting:** If the individual has been tested for HIV.
22. **FluVaxLast12:** If the individual received a flu vaccine in the last 12 months.
23. **PneumoVaxEver:** If the individual ever received a pneumococcal vaccine.
24. **TetanusLast10Tdap:** Status of tetanus and Tdap vaccinations.
25. **HighRiskLastYear:** If the individual was at high risk for certain diseases last year.
26. 
27. **CovidPos:** If the individual tested positive for COVID-19.

## PROBLEM STATEMENT

Develop and deploy a predictive model for heart disease using a subset of a dataset. The project involves creating an effective tool for early detection and risk assessment of heart diseases. Utilising key variables such as 'Had Heart Attack' among others, the model should be integrated into a Flask web application with an intuitive user interface. The goal is to leverage predictive analytics for healthcare improvement, making the tool practical and accessible for diverse users.

## INTRODUCTION OF SOLUTION

- **Algorithm Selection:** Consider algorithms suitable for binary classification with imbalanced data, such as:
  - ***Logistic Regression with class weights:*** A good baseline model, easily interpretable.
  - ***Support Vector Machines (SVM):*** Effective with high-dimensional data and can handle non-linear relationships.
  - **Random Forest:** Robust to outliers and can handle feature interactions.
  - ***XGBoost:*** A powerful ensemble method known for high accuracy.
- **Cross-Validation and Hyperparameter Tuning:** Employ k-fold cross-validation to obtain reliable performance estimates and fine-tune hyperparameters for each model to optimize their predictive power.
- **Evaluation Metrics and Diagnostics:** Go beyond accuracy and consider metrics like precision, recall, F1-score, and AUC-ROC to account for the class imbalance. Diagnostic tools like confusion matrices and ROC curves provide deeper insights into model performance.

# METHODOLOGY:

The project will follow the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology, which provides a structured approach to data mining projects. The CRISP-DM phases will be adapted to fit the specific needs of this project.

1. Business Understanding:
   - Define the project objectives and scope.
   - Identify the target audience and their needs.
   - Understand the business value of the project.
2. *Data Understanding:*
   - Collect and explore the CDC dataset.
   - Perform data cleaning and preprocessing.
   - Conduct exploratory data analysis to understand data distribution, relationships, and potential issues.
3. *Data Preparation:*
   - Select a representative sample of the data (e.g., 5%).
   - Handle missing values and outliers.
   - Encode categorical variables.
   - Transform variables as needed (e.g., scaling).
4. *Modeling*:
   - Select appropriate machine learning algorithms for binary classification and imbalanced data (e.g., Logistic Regression, SVM, Random Forest, XGBoost).
   - Address class imbalance using undersampling, oversampling, or cost-sensitive learning.
   - Train and tune models using cross-validation and hyperparameter optimization.
   - Evaluate model performance using various metrics and diagnostic plots.
5. *Evaluation:*
   - Assess the model's effectiveness and generalizability.
   - Analyze results and draw insights from the model.
   - Identify limitations and potential areas for improvement.
6. *Deployment:*
   - Develop a Flask web application for model deployment.
   - Design a user-friendly interface for data input and prediction visualization.
   - Deploy the application on a suitable platform (e.g., Hostinger).
7. *Monitoring and Maintenance:*
   - Continuously monitor model performance and collect user feedback.
   - Update and retrain the model as needed to maintain accuracy and relevance.

# FLOWCHART

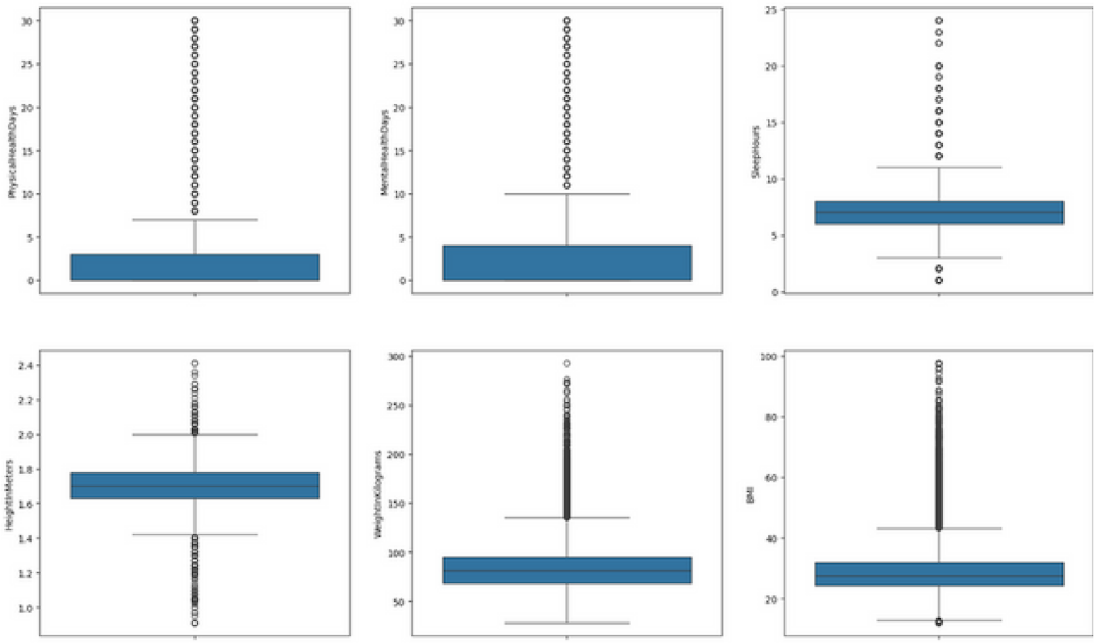
---

Starting with importing dependencies such as 'scikit-learn', 'matplotlib', 'seaborn', 'numpy', etc. which are used to build this model. Next we have loaded the data set using 'pd.read_csv()' function and checking for some of the common measures such as shape, information of the data and description of the data. Then we have visualized some of the common columns such as the 'Sex', 'HadHeartAttack' column. In fact, the 'HadHeartAttack' column has a high number of uneven distribution of classes as there are only 5.46% of the classes are 'Yes' and rest of them belongs to 'No'. Now coming on to the next step to manipulate the dataset such as converting the 'Yes/No' columns into the numeric ones and converting the categorical variables into the numeric variables using 'LabelEncoder' which assigns a numeric value to each of the categories present in the column. Then checking for the outliers present in the data set and for that we have extracted the numeric columns and plot the 'BoxPlot' for each of these numeric columns to find out how many outliers are present and to remove those outliers, we have used the 'clip-in' method to clip the values of the 90th percentile or even higher or 0.01th percentile according to the each of the column. Now after removing the outliers we are checking for the correlation between different variables present in the data set so after creating the 'Heatmap' we have found that there are so many columns which are highly correlated but they are generally related to each other example 'BMI' and 'WeightInKilograms' have a correlation of 0.86 but 'BMI' is to be calculated using 'WeightInKilograms' and 'HeightInMeters' so no need to remove those ones and we are good to go with high values of correlations. Now coming onto the train and test split but before onto that as we have a lot of class imbalance present in the data set so we have to deal with that. So, for that we are using this 'SMOTE' technique, Which is an oversampling technique, which is used to increase the data points for minority classes so that we can have both of the classes in equal proportions. Then after we are just going to split the data set into train and test with the 'random_state' of 42 and then using the 'XGBoostClassifier' to learn from data, and in 'XGBoostClassifier' we are using the number of 'n_estimators' as 500 and as we are solving a binary class problem, so our objective is 'binary:logistic' and Then fitting the 'XG_classifier' on train and test data set and then predicting the X test and finding the 'accuracy_score' which is coming out to be 96.51% and for that as we know that the accuracy is highly affected by class imbalance so for that we are just plotting the 'ROC Curve' to find out the 'AUC Score' (which is coming out to be 99..18%) and also we have find out some of the evaluation metrics such as confusion matrix, Sensitvity, Specificity, True Accuracy and F1-Score which are coming out to be 95.05%, 97.97%, 96.51% and 96.46%.

---

GRAPHS

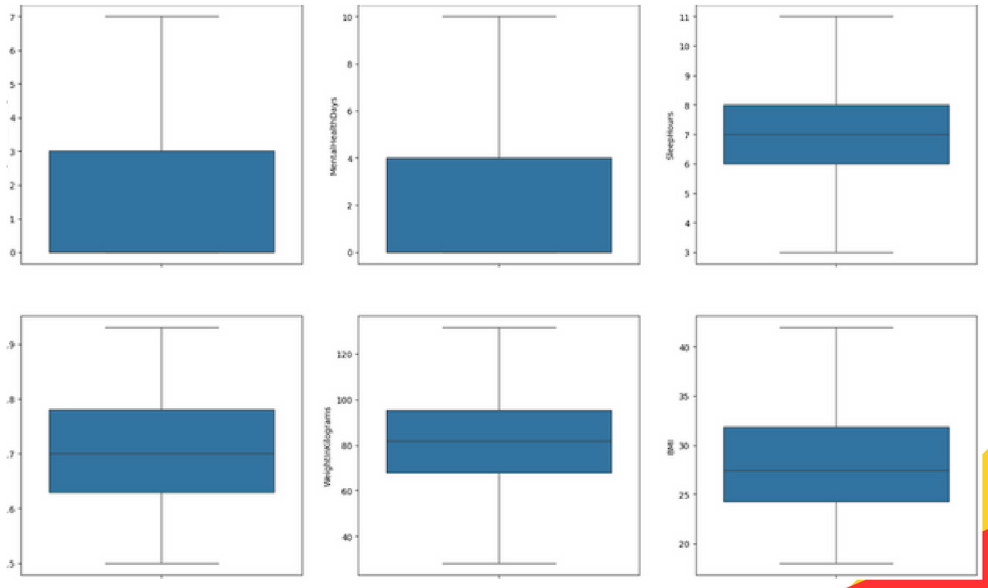
# BOX PLOT:



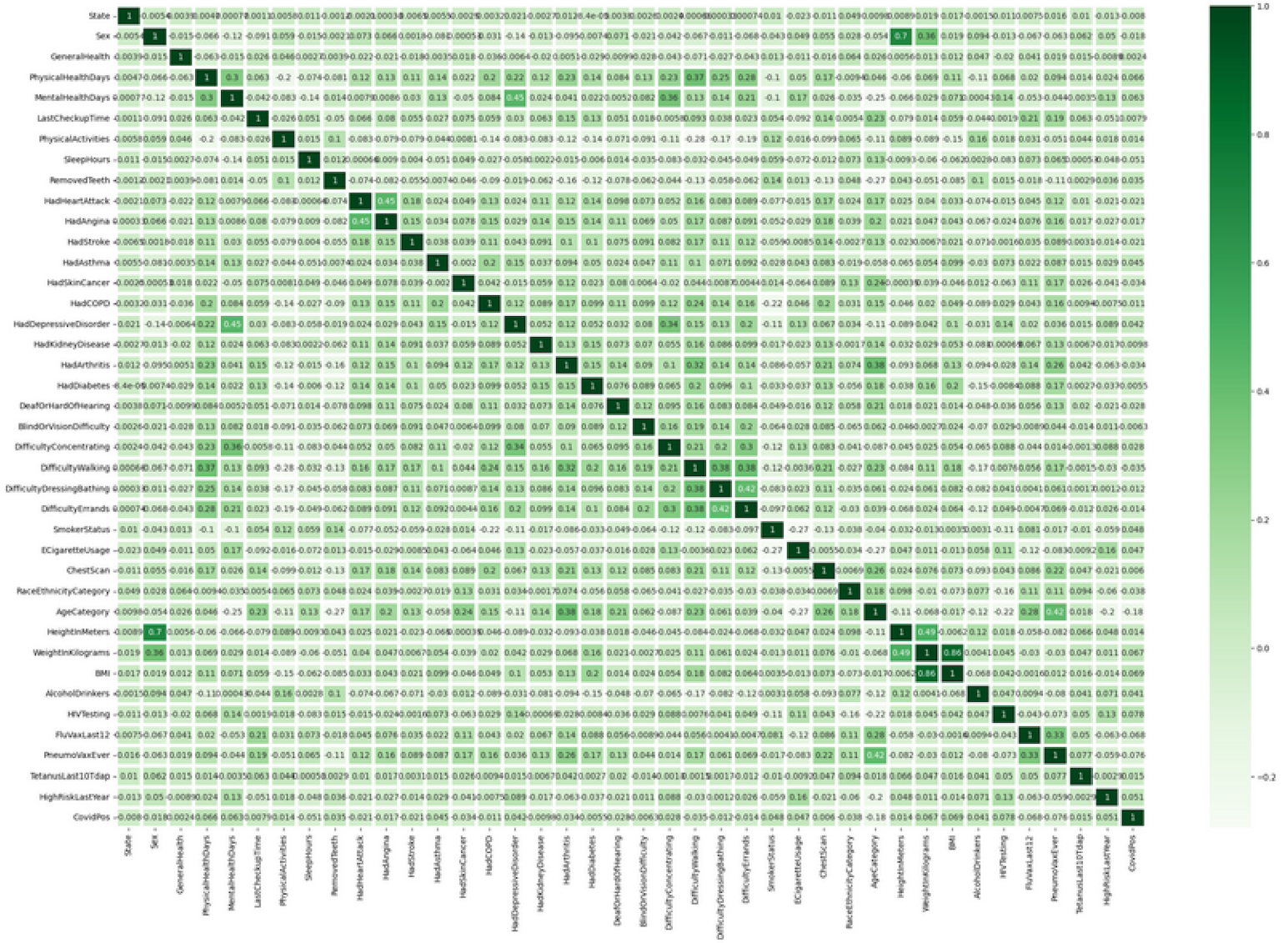

Here as we can see that there are so many outliers present in the numerical columns which can impact statistical analyses and machine learning models in several ways:

1. Skewing Results: Outliers can distort statistical measures such as the mean and standard deviation, making them less representative of the central tendency and variability of the data.
2. Model Performance: Outliers can affect the performance of machine learning models, especially those sensitive to the scale and distribution of the data. They may lead to biased model parameters or degrade prediction accuracy.
3. Robustness: Models trained on datasets with outliers may be less robust when applied to new data, as they might have learned to accommodate the outliers rather than capturing the underlying patterns.

- Using the clip method can help mitigate the effects of outliers by "clipping" or capping extreme values within a specified range. This method sets all values below a certain threshold to that threshold's value and similarly sets all values above a certain threshold to that threshold's value. Now, we can see that all of the outliers has been removed now.



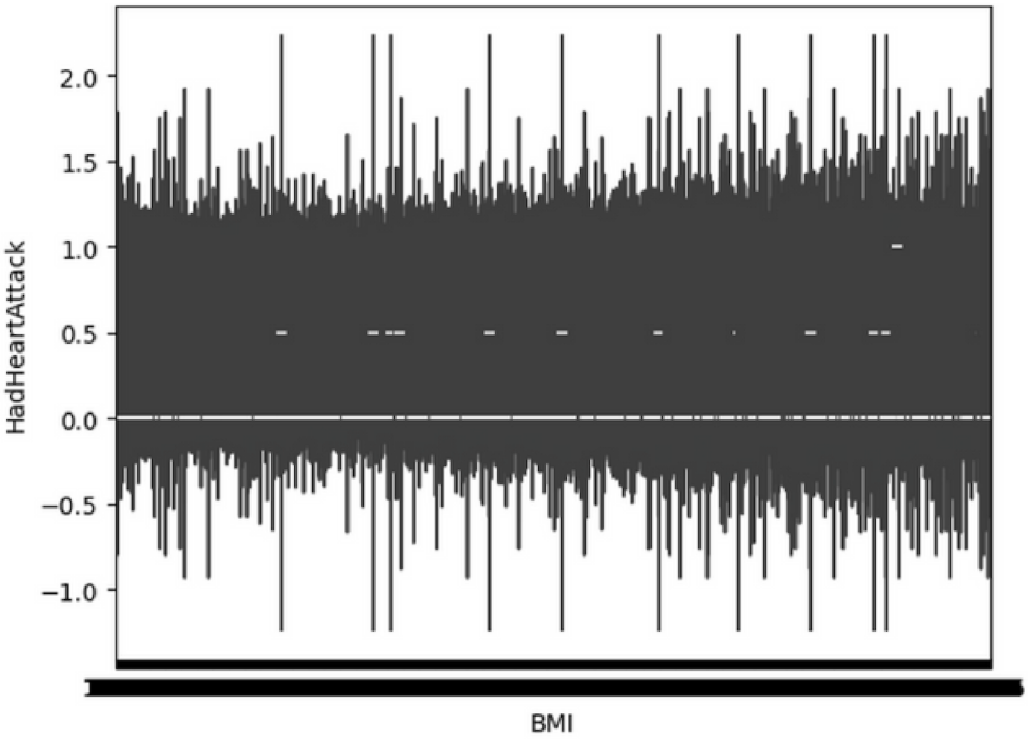
# HEATMAP




**Now we have checked for the correlation using heatmap and found that there are some columns which are highly correlated but they are generally related to each other e.g. BMI and WeightInKilograms have a correlation of 0.86 but BMI is to be calculated usign Weight and Height.**

**That's why we don't need to drop these highly correlated columns.**

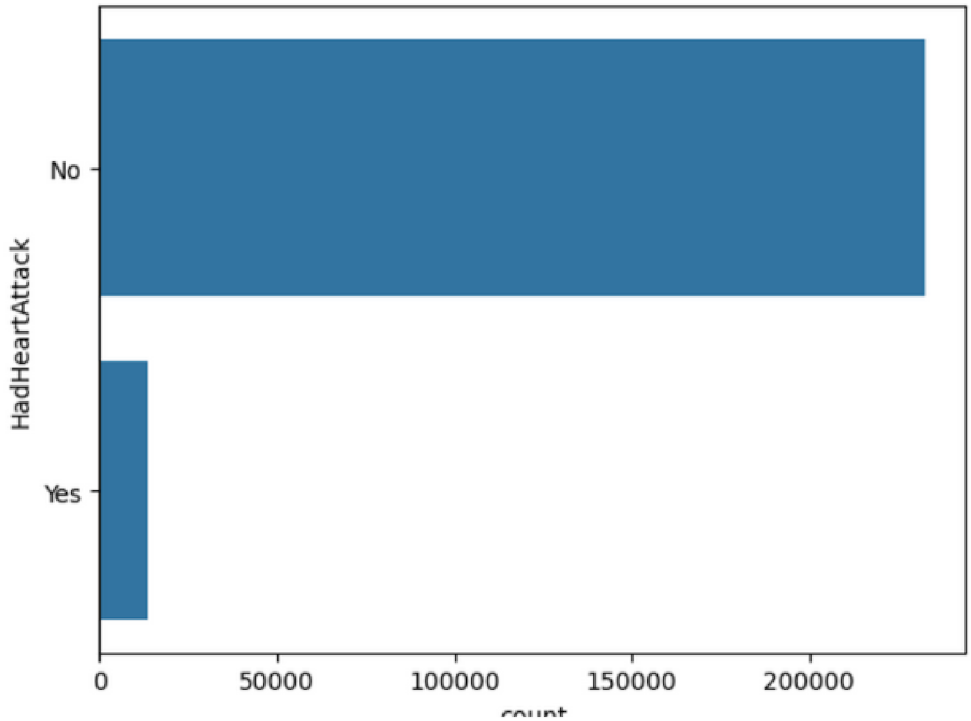
# VIOLIN PLOT:




A **Volin plot** is a type of plot that shows the distribution of data. It is similar to a box plot, but instead of a box ar whiskers, it shows a kernel density estimation of the data's distribution.

We have a line graph showing a person's BMI (Body Mass Index) over time. The BMI values are plotted against a time axis, with a series of data points corresponding to each specific BMI value .

# COUNT PLOT




**Countplot:** As we can see this uneven distribution of classes and only 5.46% of the classes are Yes and rest are no's.

For that, we'll be using OverSampling technique called as **SMOTE** which stands for **Synthetic Minority Over-sampling Technique** . It's a popular technique used in machine learning to address class imbalance in datasets, particularly in classification problems where one class (the minority class) is significantly underrepresented compared to the other classes (the majority class or classes). By generating synthetic samples rather than just duplicating existing ones, SMOTE helps to address the problem of overfitting that can occur when simply duplicating minority class samples. This technique is particularly useful when you have limited data in the minority class and want to improve the performance of your classifier by balancing the class distribution.

# RESULT ANALYSIS AND EVALUATION

*1. Model Performance Comparison:*

- Create a table summarizing the performance of each model (Logistic Regression, SVM, Random Forest, XGBoost) based on the chosen evaluation metrics (accuracy, precision, recall, F1-score, AUC-ROC).
- Visualize the results using bar charts or line graphs to compare the models side-by-side.
- Analyze the strengths and weaknesses of each model based on the results.
- Example:
  - Logistic Regression might have lower accuracy but higher interpretability compared to XGBoost.
  - Random Forest may show good performance overall but may be less interpretable than Logistic Regression.
- Discuss the impact of class imbalance handling techniques on model performance.

*2. Error Analysis:*

- Examine the confusion matrix for each model to identify the types of errors being made (false positives vs. false negatives).
- Investigate instances where the model made incorrect predictions.
- Analyze if certain subgroups within the data are more prone to misclassification (e.g., based on age, sex, or specific health conditions).
- Use this information to understand potential biases and areas where the model can be improved.

*3. Feature Importance Analysis:*

- If applicable (e.g., for Random Forest or XGBoost), extract and analyze feature importance scores to understand the relative contribution of each predictor to the model's predictions.
- Visualize feature importance using bar charts or other suitable methods.
- Discuss the insights gained from feature importance, such as identifying the most significant risk factors for heart attacks.
- Consider using these insights for feature selection in future iterations of model development.

*4. Model Selection and Justification:*

- Based on the performance evaluation and error analysis, select the best-performing model for deployment.
- Consider the trade-offs between different models, such as accuracy, interpretability, and computational efficiency.
- Justify the choice of the final model based on its performance, alignment with project goals.

# Limitations and Conclusion:

## Limitations:

**Data Sample Size:** While a 5% sample was used for computational efficiency, it may not fully capture the complexity and diversity of the entire population. Increasing the sample size, if feasible, could improve the model's generalizability and robustness.

**Potential for Bias:** The dataset may contain inherent biases reflecting existing healthcare disparities and social determinants of health. These biases could be inadvertently learned by the model, leading to unfair or inaccurate predictions for certain subgroups.

## Conclusion:

The Heart Harmony project demonstrates the potential of predictive analytics in healthcare, specifically for assessing heart disease risk. By leveraging machine learning algorithms and a comprehensive data analysis process, the project developed a model that can identify individuals at higher risk of heart attacks.

Despite limitations related to data sample size and potential bias, the project provides valuable insights into heart disease risk factors and contributes to the ongoing efforts for early detection and intervention. Future work could explore incorporating additional data sources, mitigating biases, and improving model interpretability to enhance the model's impact and ensure equitable access to predictive healthcare tools.

# THANK YOU