Machine Learning for Sustainable Development Goal 3: Good Health and Well-being

1. Introduction

Project Objective: To use machine learning to address challenges in heart disease , aiming to support SDG 3 by occurrence of heart disease, identifying heart disease sources, and forecasting the survival of patients

Motivation: To forecast the survival of patient is essential for health and well-being. By utilizing machine learning, we aim to create predictive tools that can support the prediction of heart disease or attack classification.

2. Data Collection

Data Source: Kaggle Dataset ("**Heart Disease Health Indicators Dataset** ")

Dataset Description:
- Features:

Features	**Description**
HeartDiseaseorAttack	person faced any Heart Attacks(target variable, binary variable)
HighBP	person has high BP or not(binary variable)
HighCol	person has high cholestrol or not(binary variable)
CholCheck	person has an cholestrol check
BMI	Body Mass Index of a person
Smoker	whether the person is smoker or not
Stroke	whether the person previously faced any stroke
Diabetes	whether the person is a diabetes patient
PhysActivity	Physical Activities(excerise, sports activities etc..) of a person
Fruits	Fruits consumption of the person
Veggies	Consumption of Vegetables
HvyAlcoholConsump	Alcohol consumption
AnyHealthcare	Having any healthcare including any health insurance or any government plans such as medicare etc
NoDocbcCost	NoDocbcCost
GenHlth	General health conidtion or descrition
MentHlth	Mental health problems such as stress, dipression, emotional problems etc
PhysHlth	Physical health problems such as physical illness or injury
DiffWalk	Difficulty while walking or climbing stairs
Sex	indicates gender or sex of respondent
Age	age of respondent
Education	education of respondent
Income	income of respondent

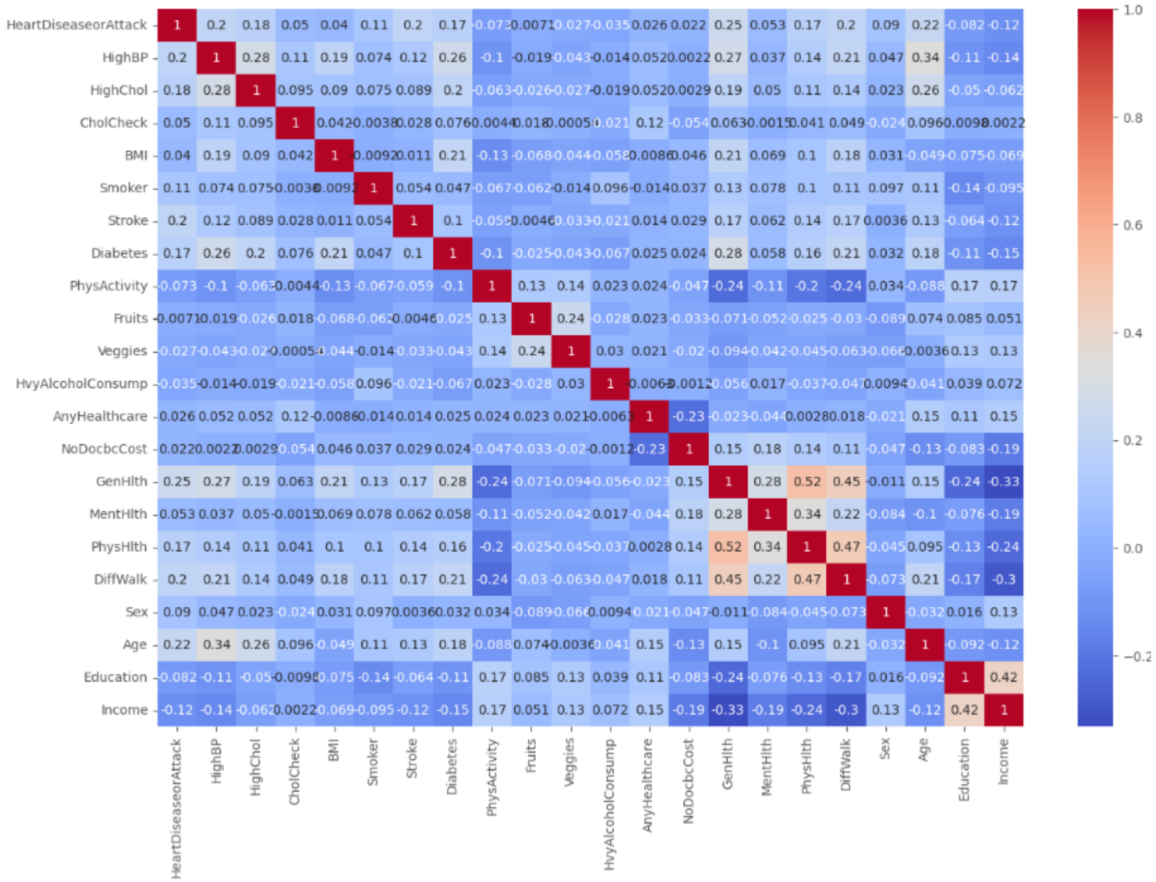
- Size: 253680 rows by 22 columns
- Target Variable: Heart Disease or Attack (binary)

3. Exploratory Data Analysis (EDA)

Summary Statistics: Performed summary statistics using df.describe() method which presents count, mean, median, std, min, max, 25% and 75% of data.

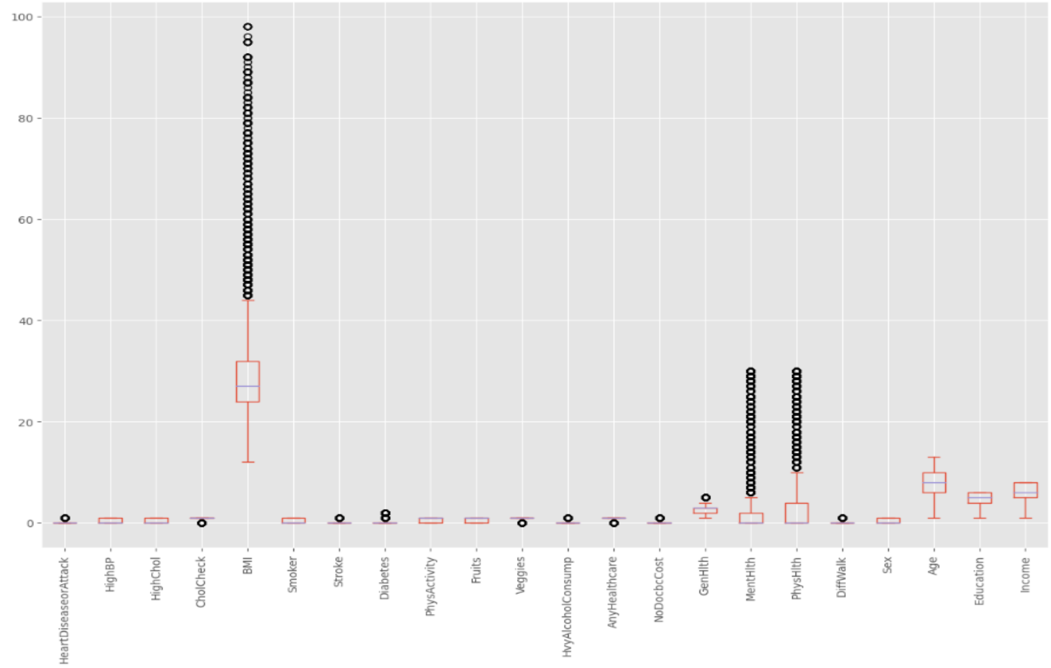
Visualizations:

- Correlation heatmap to understand relationships between variables.

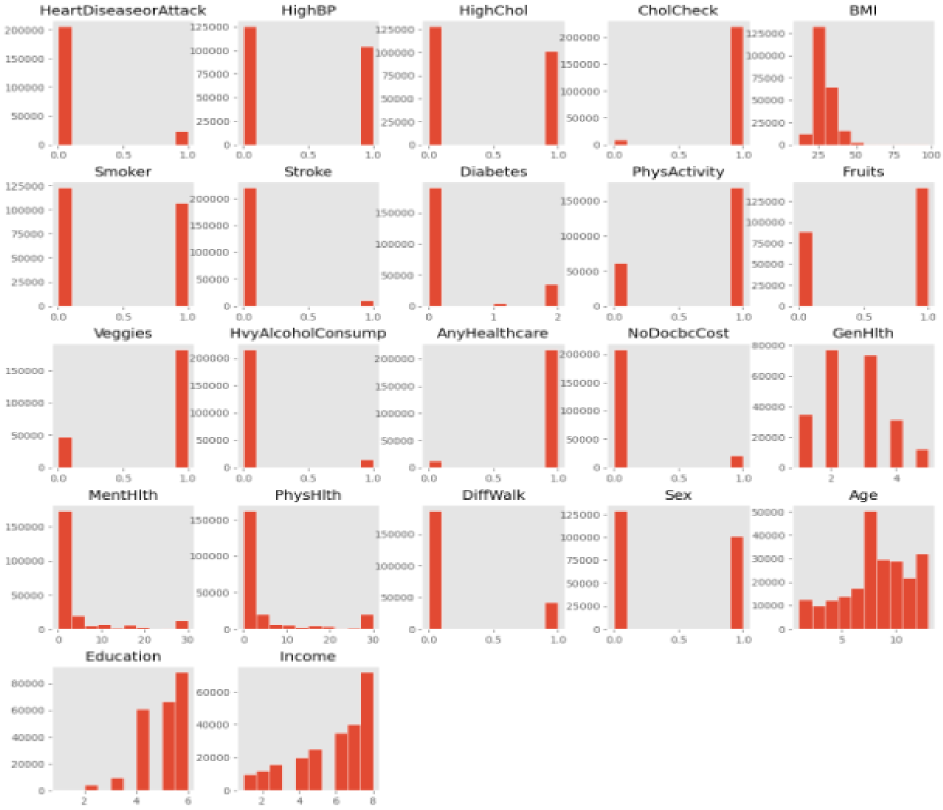


- Most of the variables does not have any strong corelation between them
- variables like GenHlth has some strong correlation with PhysHlth, DiffWalk and Education with income compare to other variables(But not much better for stronger correlation)

- Boxplots for outlier detection.



- Histograms to assess the distribution of each variable.



Insights: Key insights from HighBP, HighChol, CholCheck, BMI, Smoker, Stoke, Diabetes, PhysActivity, Fruits, Veggies, HvyAlcoholConsump, AnyHealthcare, NoDocbcCost, DiffWalk.

4. Data Preprocessing

Handling Missing Values: Null values detection using df.isnull().sum() method, found no null values.
Encoding Categorical Variables: Categories were encoded in present in float converting them into integer.
Feature Scaling: Standardized features using `StandardScaler` for better performance in machine learning models.

5. Machine Learning Model Selection

Model Choices:
- Logistic Regression (for binary classification).
- Decision Tree (predicting the classes).
- K Nearest Neighbors (predicting the classes).

Why Scikit-Learn: Easy implementation, variety of algorithms, and effective performance metrics.
Evaluation Metric: Accuracy, Classification report(Precision, Recall, and F1-Score) to find the accuracy of model performance

6. Model Implementation

Data Splitting: Split dataset into 80% training and 20% testing sets using `train_test_split` from Scikit-Learn.
Hyperparameter Tuning:
- Used GridSearchCV for Logistic Regression to identify best parameters for model efficiency and optimization
- Cross-validation with 5 folds to improve model generalization.

Code Example:

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.metrics import accuracy_score

# Splitting the data
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)

# Hyperparameter tuning for Logistic Regression
param_grid = {
  ‘penalty’ : [‘l2’],
   ‘C’ : [0.01, 0.1, 1, 10],
  ‘solver’ : [‘liblinear’, ‘lbfgs’, ‘newton-cg’, ‘sag’],
  'max_iter': [100, 500, 1000]
}
model = LogisticRegression()
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, cv=5, scoring='accuracy',
n_jobs = -1)
grid_search.fit(x_train, y_train)

# Best model and evaluation
```

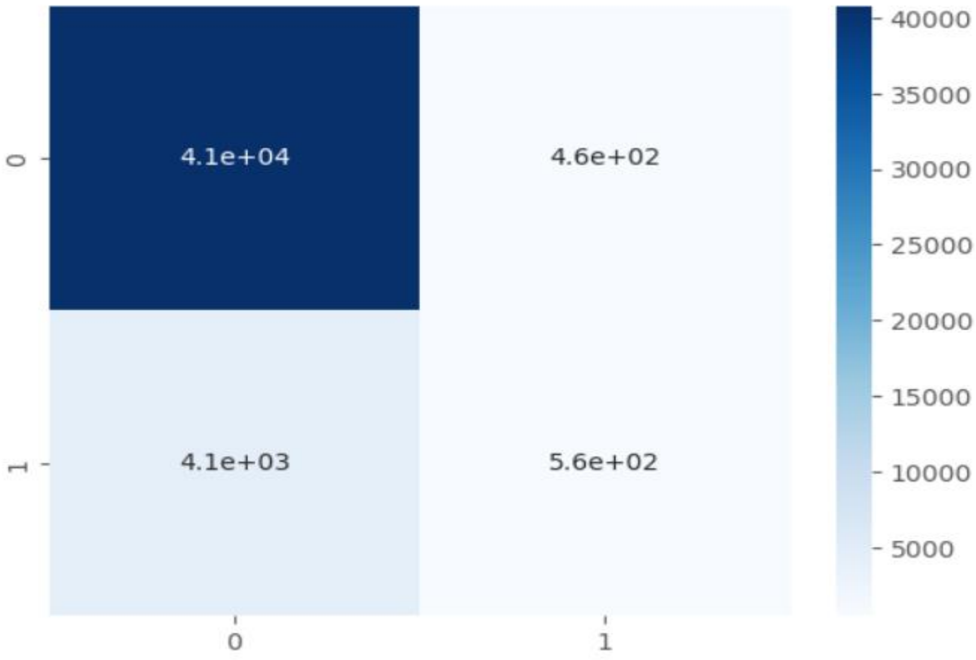
```
best_params = grid_search.best_params_
best_model = grid_search.best_estimator_
print("Best Parameters:", best_params)
Y_pred = best_model.predict(x_test)
accuracy = accuracy_score(y_test, Y_pred)
print("Accuracy : ", accuracy)
```

7. Results and Evaluation

Model Performance:
- Logistic Regression achieved an accuracy of 90%, F1-score of 87%, and precision(0.87) and recall(0.9) values indicating the model's strength in predicting the survival outcome of patient.

Feature Importance:
- Insights into which features such as HighBP, HighChol, CholCheck, BMI, Smoker, Stoke, Diabetes, PhysActivity, Fruits, Veggies, HvyAlcoholConsump, AnyHealthcare, NoDocbcCost, DiffWalk has an influence on target variable('Heart Disease or Attack').

Confusion Matrix: Visualized true vs. predicted values to identify common misclassifications.



8. Conclusion and Future Work

Key Takeaways: Machine learning models effectively predict survival outcome of a patient based on health indicators, diet and patient smoking, alcohol consumption and other variables. The project demonstrates potential for accurately predicting the survival outcomes.

Future Improvements:
- Incorporating real-time data for continuous learning.
- Expanding to a broader dataset covering more insights.
- Implementing models for better accuracy.

9. References

- Kaggle Dataset
- Scikit-Learn Documentation