EARIN Project: Heart Attack Prediction

Krzysztof Kotowski and Juan Manuel Aristizabal Henao

Warsaw University of Technology, Plac Politechniki 1, Warsaw, Poland

1 Project Description

This project aims to develop a predictive model that identifies individuals at increased risk of experiencing a heart attack based on clinical and demographic attributes. The primary objective is to build a machine learning model that receives patient data– such as age, cholesterol level, blood pressure, etc.– and outputs a binary classification indicating whether the patient is at high risk of a heart attack.

The data is taken from the **Heart Attack Prediction Dataset** available on Kaggle, which contains 8763 records of patients publicly available. The algorithms used will be: Binary Classification for outputting the prediction of whether or not the patient has a high risk of a heart attack, and Supervised Training algorithms will be used to train the model. The algorithms that can be suitable for this study are: Logistic Regression, Decision Trees, Naive Bayes, Support Vector Machines (SVM), and Random Forests.

2 Dataset Description

The dataset used contains 8763 records each representing a patient's medical profile, with 26 parameters each. A sample of this information can be seen on the Table 1, where the record of Patient 1 is visualized.

The description of each parater of the dataset, according to the information provided on Kaggle is the following:

– **Patient ID**: Unique identifier for each patient
– **Age**: Age of the patient in years
– **Sex**: Gender of the patient (Male/Female)
– **Cholesterol**: Cholesterol levels of the patient
– **Blood Pressure**: Blood pressure of the patient (systolic/diastolic)
– **Heart Rate**: Heart rate of the patient
– **Diabetes**: Whether the patient has diabetes (1 = Yes; 0 = No)
– **Family History**: Family history of heart-related problems (1: Yes, 0: No)
– **Smoking**: Smoking status of the patient (1: Smoker, 0: Non-smoker)
– **Obesity**: Obesity status of the patient (1: Obese, 0: Not obese)
– **Alcohol Consumption**: Level of alcohol consumption by the patient (None/Light/Moderate/Heavy)
– **Exercise Hours Per Week**: Number of exercise hours per week
– **Diet**: Dietary habits of the patient (Healthy/Average/Unhealthy)

- **Previous Heart Problems**: Previous heart problems of the patient (1: Yes, 0: No)
- **Medication Use**: Medication usage by the patient (1: Yes, 0: No)
- **Stress Level**: Stress level reported by the patient (1-10)
- **Sedentary Hours Per Day**: Hours of sedentary activity per day
- **Income**: Income level of the patient
- **BMI**: Body Mass Index (BMI) of the patient
- **Triglycerides**: Triglyceride levels of the patient
- **Physical Activity Days Per Week**: Hours of sleep per day
- **Sleep Hours Per Day**: Target variable (1 = risk of heart attack; 0 = no risk)
- **Country**: Country of the patient
- **Continent**: Continent where the patient resides
- **Hemisphere**: Hemisphere where the patient resides
- **Heart Attack Risk**: Presence of heart attack risk (1: Yes, 0: No)

Data preprocessing For the goal of this model some data preprocesing and cleaning have to be done before the training of the model. The following are the changes to be done to the parameters for training the model:

- **Patient ID**: Will be ignored since it has no relevance for this study since it a parameter that is unique for each patient.
- **Sex**: If the imbalance (70:30 for Male:Female) of that parameter can be addressed properly, its values can be converted to integer values that store 0 for Male and 1 for Female, otherwise this parameter should be ignored.
- **Blood Pressure**: Can be divided into two columns since it has tho values, the first one for *Systolic Pressure* and the second one for *Disatolic Pressure*. Both of these columns will have integer as a datatype, and will be grouped into representative ranges.
- **Diet**: Can be converted to integer values (-1 = Unhealthy, 0 = Average, and 1 = Healthy).
- **Country, Continent and Hemisphere**: Only the *Country* column should be kept since from it, the rest of the information contained on the *Continent* and *Hemisphere* columns can be inferred.
- **Smoking**: Should be ignored since the data imbalance in this parameter is of about 90:10 (Yes:No), which may generate a high bias and poor performance on the minority class (non-smokers).
- **Diabetes**: Can be considered for the study even if it has a significant imbalance (65:35 for Yes:No), but a specific metric can be created to measure the bias of the model and its performance on the minority class (no diabetes)
- **Exercise Hours Per Week, Sedentary Hours Per Day and BMI**: Can be rounded to 2 decimal points, and the data grouped into representative ranges.
- **Age, Income, Cholesterol, Heart Rate, and Triglycerides**: Can be grouped into representative ranges.

Table 1. Sample of the dataset to show the parameters present and the data format and types.

Parameter	**Patient 1**
Patient ID	BMW7812
Age	67
Sex	Male
Cholesterol	208
Blood Pressure	158/88
Heart Rate	72
Diabetes	0
Family History	0
Smoking	1
Obesity	0
Alcohol Consumption	0
Exercise Hours Per Week	4.168188835442079
Diet	Average
Previous Heart Problems	0
Medication Use	0
Stress Level	9
Sedentary Hours Per Day	6.61500145291406
Income	261404
BMI	31.2512327252954
Triglycerides	286
Physical Activity Days Per Week	0
Sleep Hours Per Day	6
Country	Argentina
Continent	South America
Hemisphere	Southern Hemisphere
Heart Attack Risk	0

For the predicted parameter (**Heart Attack Risk**) there is a considerable imbalance of 64:36 for No: Yes. This imbalance can be treated by implementing an imbalance treatment technique that can bring closer both values, so that to minimize the risk of poor performance over people that have high risk of heart attack. A proposed solution to this issue can be using an oversampling approach that will duplicate the entries that have high risk, so that the ratio may look more like 50:50.

If the above-mentioned approach happens to yield poor results, more data could be scraped from other websites that also publicly offer access to this data.

Metrics Our model's results will be analysed using:

- Accuracy - Percentage of correct predictions in general
- Confusion matrix - Plotting model's results against actual results
- Precision - How many positive classifications (results) are actually correct?
- Recall - How many actually positive cases were detected by the model?

3 Solution

Common methods used for Binary Classification problems are Logistic Regression, Decision Trees, Naive Bayes, Support Vector Machines (SVM) and Random Forests.

3.1 Logistic Regression

Logistic regression is a supervised machine learning algorithm that performs binary classification tasks by predicting the probability of an outcome, event, or observation. The model delivers probabilities mapped to classes (typically 0 or 1).

This is one of the methods which will be implemented. A sigmoid function will be used to map the probabilities. For the data preprocessing, The previously explained data preprocessing will be used in addition to measure the linearity of the relationships between the parameters, to determine if they are suitable to train the model in this way.

3.2 Decision Trees

A decision tree is a flowchart-like model that maps out the possible outcomes of a series of related choices, helping individuals or organizations evaluate potential actions based on their costs, probabilities, and benefits. It's a visual tool that can also be used to build predictive models in machine learning.

3.3 Random Forests

A Random Forest is a machine learning algorithm that combines multiple decision trees to improve prediction accuracy and robustness. It's an ensemble method that utilizes bagging, where each tree is trained on a random subset of the data and considers only a random subset of features at each node split. By averaging or voting the predictions of these trees, Random Forests produce more reliable results than a single decision tree.

This is one of the methods that will be used. The previously mentioned data processing will be done, with a focus on some data augmentation if the results are of poor quality.

3.4 Support Vector Machines

Support Vector Machines (SVMs) are supervised machine learning algorithms used for classification and regression tasks, particularly effective in high-dimensional spaces. They work by finding a hyperplane that best separates different classes of data, maximizing the margin between them. This "margin" represents the distance between the hyperplane and the closest data points from each class, known as support vectors.

3.5 Naive Bayes

Naive Bayes is a simple yet powerful classification algorithm based on Bayes' Theorem, assuming that features are independent of each other. It calculates the posterior probability of each class given the input features and selects the one with the highest probability. It's widely used for tasks like text classification, spam filtering, and sentiment analysis due to its speed and efficiency, even with small datasets. Despite the simplifying "naive" assumption of feature independence, it often performs surprisingly well in practice.