**Heart Attack Classification**

**Student**: **Mohammad Jamal Mahmoud Al Jadallah**

**Student Number**: **32009304050**

**Project: Heart Attack Classification**

**Professor: Suleiman M. S. Bani Hani**



**Introduction**

Heart attacks, also known as myocardial infarctions, are serious medical emergencies that occur when there is a blockage in the blood vessels supplying the heart muscle. This blockage restricts blood flow to the heart, leading to damage or death of the heart muscle tissue. Heart attacks can vary in severity and can be classified based on different criteria to better understand and manage the condition. The classification of heart attacks helps healthcare professionals in diagnosing and treating patients effectively.

**Business Problem**

**predict the risk of heart attacks!**

One of the significant business problems related to heart attacks is the need for a predictive risk assessment system. Healthcare organizations and insurance companies face challenges in identifying individuals who are at a higher risk of experiencing a heart attack. By developing a solution to predict the risk of heart attacks, these organizations can proactively intervene, provide targeted interventions, and reduce healthcare costs.

**Solution**

**Machine Learning Classification for Heart Attack Classification**

Machine learning models can be utilized to classify heart attack data based on various parameters and features. By training these models on labeled datasets, they can learn patterns and make predictions on new, unseen data, thereby assisting in the classification of heart attacks.

**Heart Attack Dataset**

The Heart Attack dataset contains information related to patients and various factors that may influence the likelihood of a heart attack. Here are the variables included in the dataset:

1. **Age**: This variable represents the age of the patient, indicating their chronological age.
2. **Sex**: This variable indicates the sex of the patient. It can take two values:

* 0: Female
* 1: Male

1. **exang**: Exercise-induced angina is a condition where chest pain or discomfort occurs during physical activity or exercise. This variable indicates the presence or absence of exercise-induced angina:

* 0: No exercise-induced angina
* 1: Exercise-induced angina present

1. **ca**: The number of major vessels refers to the number of major blood vessels supplying the heart that are blocked or narrowed. This variable represents the count of such vessels and can take values from 0 to 3.
2. **cp**: Chest pain type is a categorical variable representing different types of chest pain experienced by the patient. It can take the following values:

* **1**: Typical angina
* **2**: Atypical angina
* **3**: Non-anginal pain
* **4**: Asymptomatic (no chest pain)

1. **trtbps**: Resting blood pressure is the blood pressure measured when the patient is at rest. It is measured in mm Hg (millimeters of mercury).
2. **chol**: Cholesterol is a fatty substance found in the blood. This variable represents the level of cholesterol in mg/dl (milligrams per deciliter) fetched via a BMI (Body Mass Index) sensor.
3. **fbs**: Fasting blood sugar is the level of glucose in the blood after fasting for a certain period. This variable indicates whether the fasting blood sugar level is greater than 120 mg/dl:

* **0**: Fasting blood sugar level is less than or equal to 120 mg/dl
* **1**: Fasting blood sugar level is greater than 120 mg/dl

1. **rest\_ecg**: Resting electrocardiographic results provide information about the electrical activity of the heart at rest. This variable represents the results of the resting electrocardiogram and can take the following values:

* **0**: Normal
* **1**: Abnormal ST-T wave (T wave inversions and/or ST elevation or depression of > 0.05 mV)
* **2**: Probable or definite left ventricular hypertrophy by Estes' criteria.

1. **thalach**: Maximum heart rate achieved is the highest heart rate recorded during exercise or physical activity.
2. **target**: This variable represents the target variable and indicates the likelihood of a heart attack:

* **0**: Less chance of a heart attack.
* **1**: More chance of a heart attack.

The dataset's variables provide information about the patients' demographic characteristics, medical history, symptoms, and diagnostic results. This data can be used to develop predictive models for classifying the likelihood of a heart attack based on the given features.

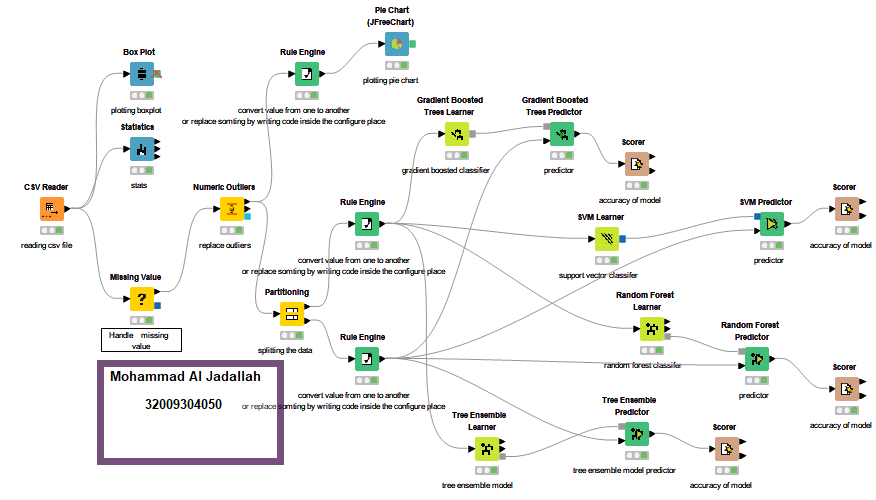
**Methodology for Heart Attack Classification using Python and KNIME**

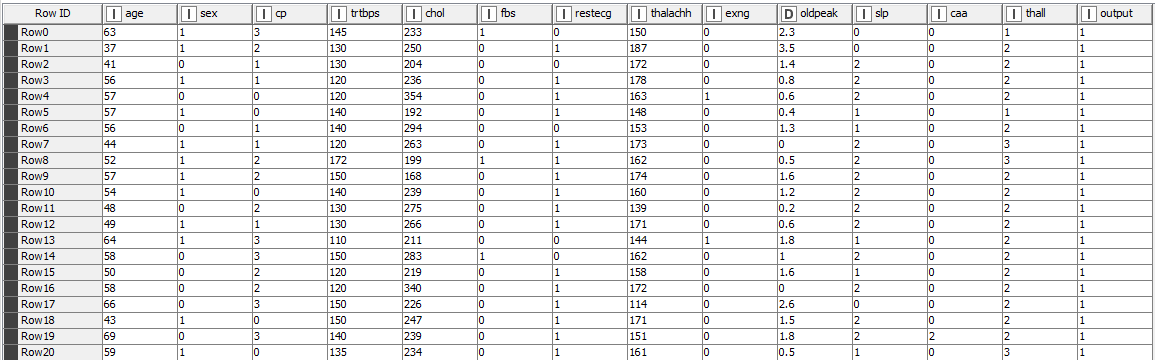
In this methodology, we will use a combination of Python and KNIME to develop and train machine learning models for heart attack classification. Python will be used for data preparation, model creation, and evaluation using libraries such as scikit-learn (sklearn), pandas, and matplotlib. KNIME will serve as a complementary tool to build workflows, preprocess data, and train models. The overall steps involved in the methodology are as follows:

* **Data Preparation**: Firstly, we will load the heart attack dataset into Python using the pandas library. We will perform necessary data preprocessing steps, such as handling missing values, encoding categorical variables, and scaling numerical features. The preprocessed dataset will be split into training and testing sets for model evaluation.
* **Model Creation and Training in Python**: With the preprocessed data, we will proceed to create and train machine learning models using Python. We will utilize libraries like sklearn to implement various classifiers, including Random Forest Classifier, Gradient Boosted Classifier, SVM Classifier, and Decision Tree Classifier. Each model will be trained using the training data and the fit() function provided by sklearn. Model performance will be evaluated using appropriate evaluation metrics, such as accuracy, precision, recall, and F1-score, utilizing the testing set.
* **Model Creation and Training in KNIME**: Moving to KNIME, we will open the tool and create a new workflow. The heart attack dataset will be imported into KNIME, where we can leverage the available nodes to preprocess the data, including handling missing values, encoding categorical variables, and scaling numerical features. The dataset will then be split into training and testing sets within KNIME. Using KNIME's modeling nodes, we will create the same set of classifiers as in Python, including Random Forest Classifier, Gradient Boosted Classifier, SVM Classifier, and Tree Ensemble Classifier. The nodes will be configured to train the models using the training data. Evaluation nodes will be connected to each model to assess their performance using suitable metrics.
* **Model Evaluation and Selection**: The performance of each model, both in Python and KNIME, will be compared based on the evaluation metrics obtained. Factors such as accuracy, error to select the best-performing models.

Throughout this methodology, Python will be primarily used to create and train the models, perform data preparation, and evaluate model performance. KNIME will be utilized to build workflows, preprocess data, and train models, providing a visual interface for effective data analysis and model development. The combined use of Python and KNIME offers flexibility and enables efficient utilization of both tools in the heart attack classification task.

**KNIME Workflow**

 **Figure-1: KNIME workflow**

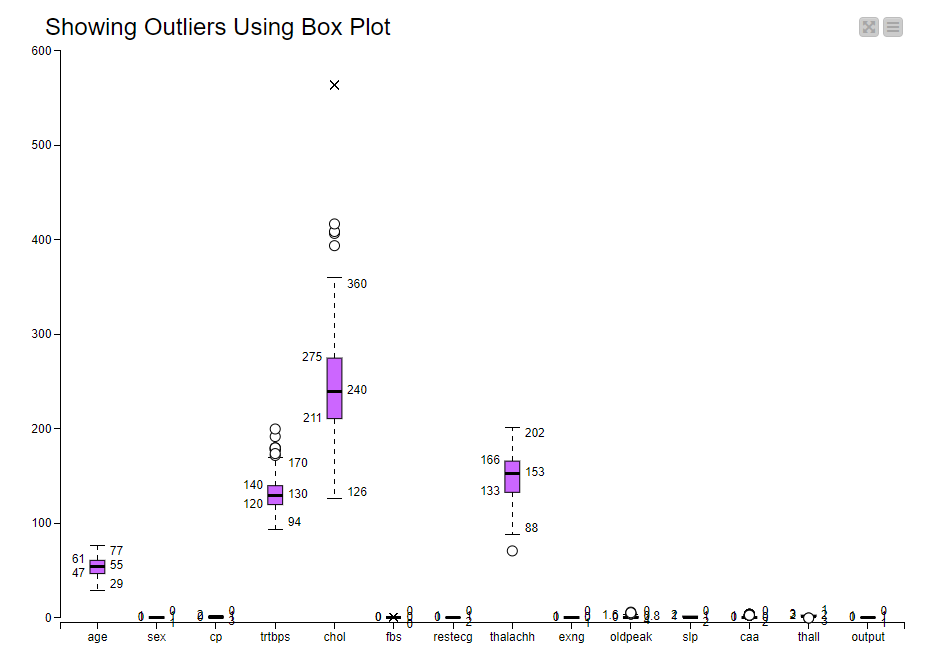
**Sample Dataset**

**Figure-2: Dataset**

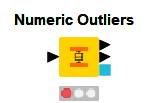
**Statistics**

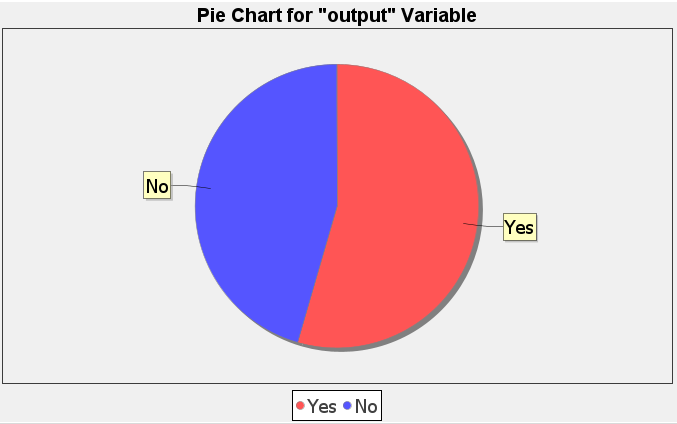
**Figure-3: Statistics**

**BOXPLOT:**

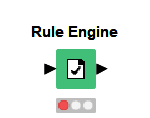
Used to show the outliers.

**Figure-4: BoxPlot**

So, for the above chart we can see that there are outliers in the data so we must fix it using Numeric outliers’ node which used to handle the outliers in the dataset.

**Visualize the target column:**

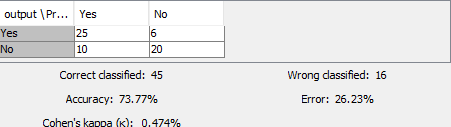
**Figure-5: Pie Chart**

**Rule Engine Node:**

It is a node used to convert the values of a column from something to another. In our project we need it to convert the output column from numeric to categorical data, so when we want to train the models, we will not face a problem like the below one:

[**no possible target in input node not connected unable to configure in KNIME**](https://forum.knime.com/t/decision-tree-table-contains-no-nominal-attribute-for-classification/5388)**.**

**Accuracy of models:**

* **Gradient Boosted Trees Learner**
* **A screenshot of a computer

  Description automatically generated with medium confidenceSVM Learner**
* **A screenshot of a computer

  Description automatically generated with medium confidenceRandom Forest Learner**
* **A screenshot of a computer

  Description automatically generated with medium confidenceTree Ensemble Learner**

**Best of Luck**