**Objective: Build a generalized ML class for prediction tasks.**

Please do the following steps (hint: use numpy, pandas, sklearn, xgboost, seaborn and matplotlib)

1. Design a class named Prediction with data and labels as its instance variables to compare multiple classifiers such as **Multinomial Naïve Bayes, Gaussian Naïve Bayes, K nearest neibors, logistic regression, random forest**, **gradient boost**, xgboost, **decision tree**, svm, etc. At least one classification algorithm is required, but good to have as many as possible.
2. Create member functions for **feature selection** by
   1. Drop the features with missing values larger than 60%
   2. Drop the features whose correlation with y(dependent) variable is smaller than 0.2
3. Create a member function to **split data into train and test sets (with parameter to define the proportion between train and test)**.
4. Create a member function to predict, and output each model’s performance metric of accuracy, confusion matrix, roc auc, pr auc
5. Write a python wrapper to use your class and predict the target variable in the provided dataset. (see heart.csv, Predict ‘output’ using other variables.),

Following is optional, but good to have.

1. Create a member function to perform **data\_normalization** with either minmax scaler or Zscore scaler
2. Create a member function for **parameter tuning**