# Project Phase I

*by*

**Saurabh Katkar**

**Collaborators: None**

## Mode of Execution:

- I have made a few changes to the "simulate_agents_phase2.py" file that are necessary for proper execution of my code, although they don't conflict with the code in agents."py file". A few(not all) of them are mentioned below. As such I am providing the "simulate_agents_phase2.py" file along with my code in the rar folder.
- Placed the agent_skatkar.py file in the same folder as the agents.py and the simulate_agents_phase2.py file
- Added "from agent_skatkar import *" at the beginning of simulate_agents_phase2.py file.
- Added "agents.append(Agent_skatkar("My_classifier"))" to add the user-create agent in simulate_agents_phase2.py file.
- This code was executed in IPython using the Anaconda environment.

## Overview

Given the training and validation set, train the Agent to learn the probability of being either Good(1) or Bad(0)

Maximize the profit received on selling the product for a value by training the agent with the best classifier from a list of classifiers.

## Strategy implemented

### Grid Search for finding best parameters of a classifier:

The grid search provided by **GridSearchCV** exhaustively generates candidates from a grid of parameter values specified with the param_grid parameter.

Using Grid search approach to parameter tuning, we methodically build and evaluate a model for each combination of algorithm parameters specified in a grid.

### Performance evaluation of classifiers to find the best one:

After applying GridSearchCV, we store the resultant classifiers with the derived parameters in a list and compare the classifiers using the validation dataset.

For the purpose of comparison we take into account multiple performance evaluators such as **Accuracy**, **Precision**, **Average Precision**, **Recall**, **F1-Score**, **AUC** to determine the selection of the best classifier.

Applying the aforementioned algorithm gives us the best classifier to be applied to the validation dataset using which we get the profit that beats the Bernoulli – Naïve Bayes algorithm applied by the professor in 9 out of the 12 cases (1,8 and 9) being the exception

Thus to summarize,

- We apply the algorithm from the **find_best()** method in **Agent_skatkar** class to find the best classifier for the given validation dataset.
- Using different performance evaluators we determine the best classifier from the list of classifiers.
- We apply this classifier, called '**best one**' to the algorithm in the **simulate_agents_phase2.py** file to get the profit derived from selling the product at the cost of **value** after buying it at the cost of **price**.

The evaluation results with respect to profit using **different evaluators** is as follows:

| | Precision | Classification Report | Accuracy | Average Precision | F1 Score |
|---|---|---|---|---|---|
| 1 | $2,632,500.00 | $2,956,450.00 | $2,982,750.00 | $3,086,200.00 | $2,982,700.00 |
| 2 | $3,283,650.00 | $3,283,650.00 | $3,280,550.00 | $3,283,650.00 | $3,280,550.00 |
| 3 | $3,478,100.00 | $3,431,600.00 | $3,479,600.00 | $3,470,200.00 | $3,474,900.00 |
| 4 | $3,121,600.00 | $3,044,450.00 | $3,125,650.00 | $3,474,700.00 | $3,130,850.00 |
| 5 | $1,647,250.00 | $1,687,950.00 | $1,691,400.00 | $1,686,450.00 | $1,687,150.00 |
| 6 | $2,155,350.00 | $2,203,250.00 | $2,158,800.00 | $2,204,250.00 | $2,198,300.00 |
| 7 | $1,749,450.00 | $1,749,450.00 | $1,749,450.00 | $1,749,450.00 | $1,749,450.00 |
| 8 | $1,496,300.00 | $1,653,550.00 | $1,653,550.00 | $1,496,300.00 | $1,653,550.00 |
| 9 | $568,600.00 | $712,300.00 | $712,300.00 | $677,800.00 | $712,300.00 |
| 10 | $805,750.00 | $644,200.00 | $805,750.00 | $805,750.00 | $644,200.00 |
| 11 | $851,100.00 | $849,500.00 | $859,750.00 | $843,100.00 | $858,700.00 |
| 12 | $376,650.00 | $489,350.00 | $486,150.00 | $317,800.00 | $486,400.00 |

After finding the best classifier we apply this classifier in the **fit_the_classifier** method. This classifier also predicts probabilities of the Y-label.

## Conclusion

Thus we derived an algorithm to find the best classifier that will maximize the profit obtained by selling the product at the cost of value after buying it at the cost of price by choosing the best classifier for the dataset from a list of classifiers.