# MATH 4995 Mini Project 1. Machine Learning Basics
# Kaggle Contest: Home Credit Default Risk
# By Ho Chuen Ho, Chun Lok Him Brian

## Introduction

Home Credit Default Risk is an issue where bank may need to predict if a client is likely to default. In this project, datasets are used from Kaggle and various machine learning algorithm will be used on the dataset. For simplicity, mostly variables listed in application_train.csv will be used.

## Challenges

Firstly, the response data in the training set is imbalanced. Only less than 10% of the data are marked as client having difficulty to repay. Moreover, it is very likely that bank would prefer false positives (wrongly classifying people as defaulters) than false negatives. Therefore, the sensitivity of classification on defaulters will need to be considered. Also, precision may be used instead of accuracy.

Secondly, there are over 130 variables in the training set, and some of them are correlated. Some models may suffer from curse of dimensionality and may lead to overfit. Also, there are 300k sample in the training set, and if most of them are used, the training for some models with high computational complexity may run very slow. Choosing less complex model may be preferred.

## About the data

In the dataset, various information revolving financial status, for example, information relating to their work (e.g. location of work, types of work), and their family (e.g. the type of housing, the size of their family) can be used for predictors Predictor consist of a mixture of variable types, including indicator variable, discrete and continuous variables) , and whether the client will default (1 or 0) is used as the response. It is worth noted that some values of predictors are missing.

## Validation: k-fold Cross Validation

To find the optimal model, k-fold validation is used in the model (Wang & Zheng, 2013). For all of the listed mode, they would be a 5-fold validation considering the size of the testing set and also the number of samples, which will affect the computational time.

## Feature Selection

The first method is using light gradient boosting model, we obtain the top 10 importance feature with parallel tree boosting technique.

The second method is calculating correlation between features. Variables that are highly correlated with one another is redundant and will be removed. Because reducing the number of features can help the model learn during training and generalize better to the testing dataset.

## Solution – Naïve Bayes

Naïve Bayes classifications have multiple advantages on the dataset. It is unlikely to overfit, and given the large dataset, it is unlikely to underfit any of the variables also. Moreover, the computational complexity is also very low (O(n·d)), making it easier for testing. However, as it is a simple model and that it assumes independence of variables, it will lower its performance.

This model gives a score of 0.59.

## Solution – Adaboost

Adaboost is a model that is less likely to suffer from overfit (Schapire, 2013). Therefore, one of the solution is to use adaboost to predict the result. We have used 100 estimators given the amount of variables to reduce the chance of underfit.

It is worth noting that adaboost is the slowest model among all others we have tested, due to the high amount of estimator.

The model gives a score of 0.734

## Solution – Stochastic Gradient Descent

SGD is another model that can achieve classification with high computational efficiency. Also, we have allowed early stopping to further increase efficiency if the model does not improve after multiple iteration. To further improve prediction performance, elastic net regularization have been used. Both methods of encoding is tried out in this model.

The model gives a score of 0.701

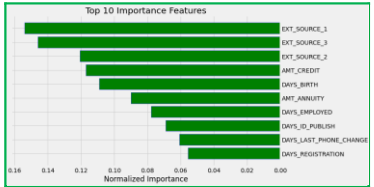

Figure: Top 10 Importance Features

Figure: Heatmap on correlation of variables

## Solution – Light Gradient Boosting

Boosting allow learning to be more accurate using multiple weak learners. Light Gradient Boosting Machine is implemented as a way to try out how effective Light GBM is. We have chosen a learning rate that is lower than the default value, and a significantly higher amount of estimators than default. This model is also computing faster than adaboost.

The model gives a score of 0.744.

## Contribution

Ho Chuen Ho: adaboost, k-fold cross validation, stochastic gradient descent, gradient boosting, report

Chun Lok Him Brian: adaboost, naïve bayes, report

## References

Schapire, R. E. (2013). Explaining adaboost. In Empirical inference (pp. 37-52). Springer, Berlin, Heidelberg.

Wang, H., & Zheng, H. (2013). Model Validation. Machine Learning. In Encyclopedia of Systems Biology, 1406-1407.