MACHINE LEARNING BENCHMARKING CREDIT DEFAULT DATASET

ANGGORO Fajar Tri

fajartri.anggoro@ieseg.fr

March 29th, 2022

CONTENTS

LIST OF FIGURES

OVERVIEW

PROBLEM STATEMENT

Machine learning is a well-known technique derived from statistics. The application of Machine learning techniques in nowadays era has been massive. In business, both the effectiveness and the reliability of a Machine Learning model has been utilized to improve business. However, with there are many available models out there, often, we don’t exactly know which model to use. In this case, we’re going to develop several Machine learning models based on a given dataset. The models will then be evaluated and then compared with one another.

The dataset that we’re using is the credit default dataset. A list of customers, their demographic information, as well as their historical track for the credit bill & payment. The target variable is whether if a given customer will default in the following month.

GOAL OF THE REPORT

The goal of the report is to set up a benchmarking experiment between different Machine learning models within a given dataset. From the result of the benchmarking experiment, we will determine the best performing model.

ML MODELS BENCHMARKED

LOGISTIC REGRESSION

The first model of the benchmark is the logistic regression. Logistic regression is the modified version of a linear regression model. Since linear regression doesn't perform well in classification problems, logistic regression modifies the underlying function in linear regression into sigmoid function. It tries to predict the probability of the target variable belonging into a specific category.

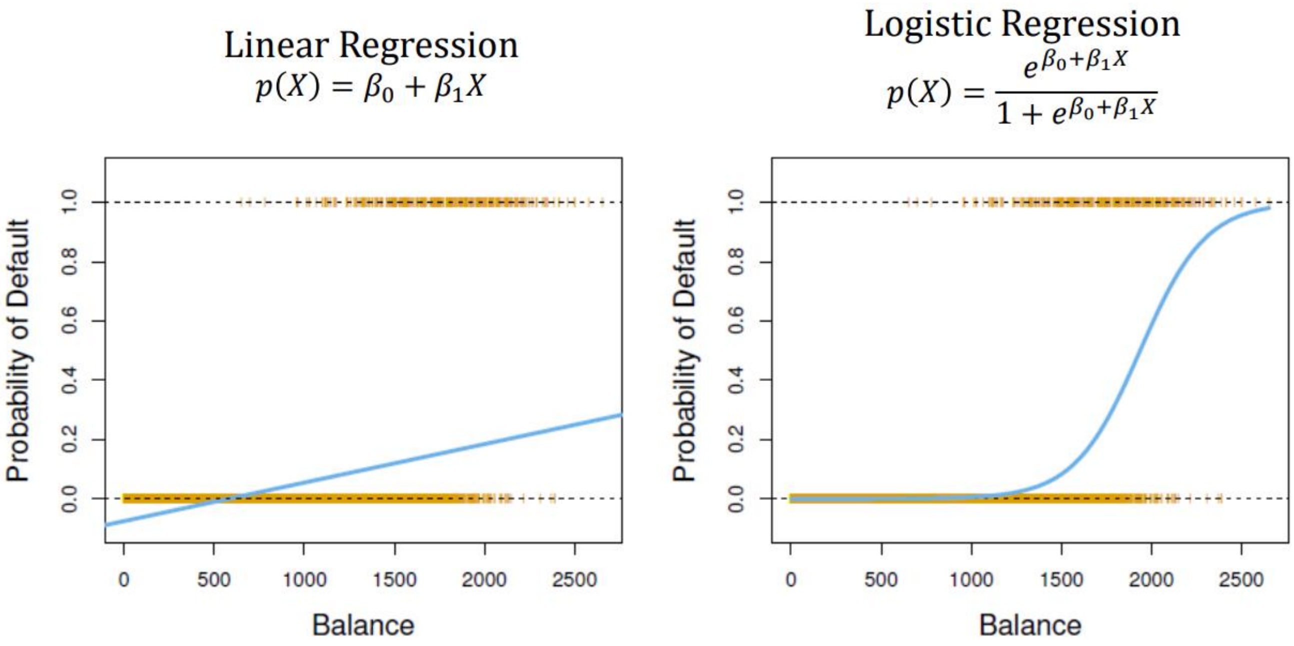


Figure 1 linear regression vs logistic regression

The objective of this model is to maximise the likelihood function. To maximise the likelihood function, in this case, we're going to use Stochastic Gradient Descent optimization algorithm. In practice, this model although usually has lower performance as opposed to the other models, it is still widely used because of its interpretability.

The advantages of Logistic Regression Model:

- Fairly intuitive and interpretable model
- Simple model

While the disadvantages of Logistic Regression Model:

- Less flexible
- Tend to perform poorly, as compared to other models

K-NEAREST NEIGHBOURS

The next model of the benchmark is K Nearest Neighbors. Essentially, the idea of KNN is that groups that belong to the same class will be located near each other. It makes its prediction based on the nearest neighbors of the point. It doesn't have an objective function since it technically, in its prediction, doesn't construct any equations. It just simply made its prediction based on the distance to the nearest neighbors. In practice this model is not as widely used because of its poor performance.

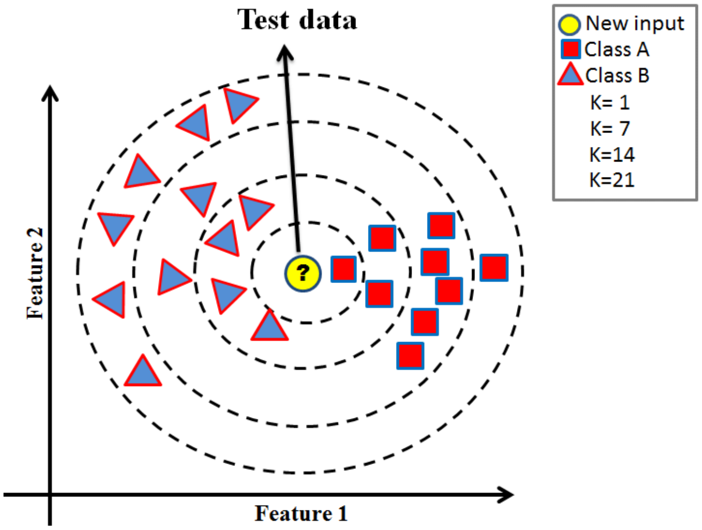


Figure 2 KNN Illustration

The advantages of KNN Model:

- Fairly intuitive and interpretable model
- Doesn’t have an objective function and therefore fast

While the disadvantages of KNN Model:

- The optimum k (number of neighbors) has to be tuned
- Sensitive to outliers

SUPPORT VECTOR MACHINE CLASSIFIER

The next model of the benchmark is Support vector machine. SVM is usually a good model to consider, especially in a higher dimension. The idea of SVM is that it will create a hyperplane that separates the class of our data. The objective function of this model is to find the hyperplane with the maximum distance / margin.

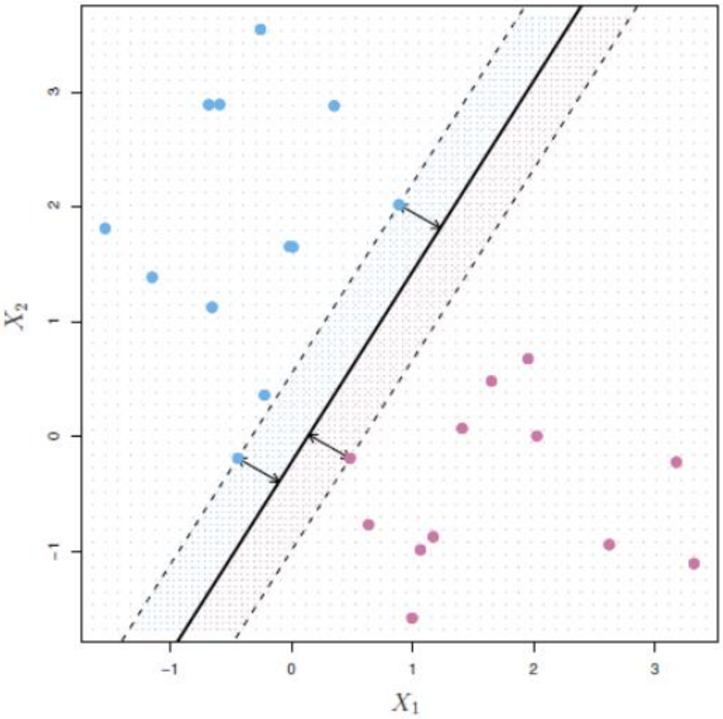


Figure 3 Support Vector Machine Hyperplane & Margin

However, in most real-life cases, our data is not always perfectly separated. That's why in this benchmarking experiment, we're trying different kernels and adjusting the regularization term in order to find the best hyperparameters.

The advantages of SVM Model:

- Good in high dimensional settings
- Kernel could be adjusted to meet needs

While the disadvantages of SVM Model:

- Regularization term needs to be adjusted for bias-variance trade off

GRADIENT BOOSTING CLASSIFIER

The next model of the benchmark is gradient boosting classifier. Gradient boosting is a family of the tree-based method in Machine learning models. The idea of gradient boosting is that it does the whole process sequentially. The first tree is built and evaluated, then the next tree is build based on the error from the previous tree. This process is repeated until the learning process is completed. Tree based models are widely used within practice because of its high performance.

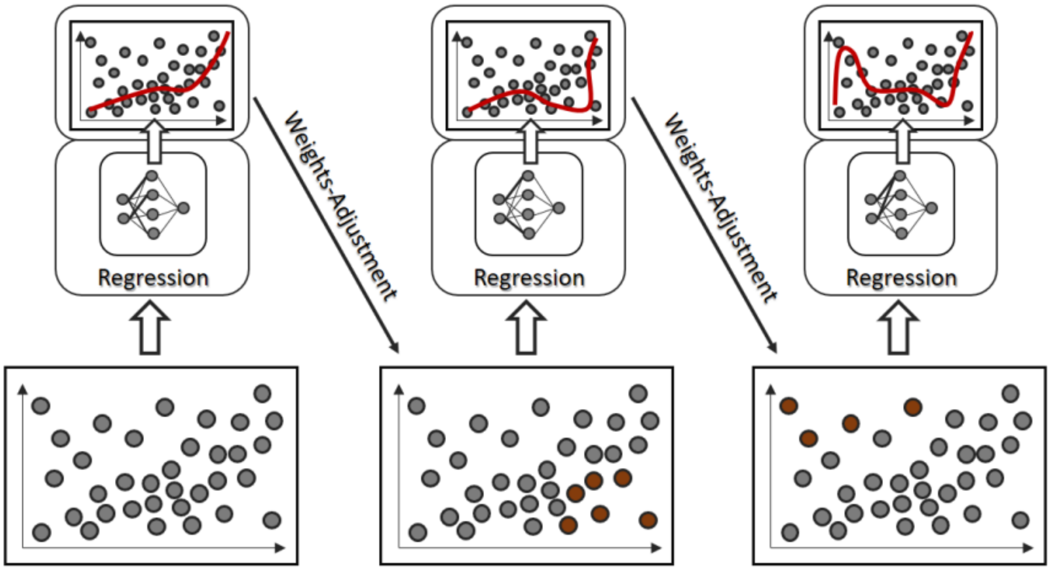


Figure 4 Ada Boosting

The advantages of Gradient Boosting Model:

- More flexible
- Similar to tree-based models, preprocessing is not really necessary

While the disadvantages of Gradient Boosting Model:

- Since it works sequentially, the model is computationally expensive

RANDOM FOREST CLASSIFIER

Our next model of the benchmark is Random Forest. Random forest is another family from the tree-based models. As the name suggest, random forest is a combination of decision tree models. It makes its prediction based on the prediction of these decision trees. Each decision trees are fitted with random instances and features. This what makes it superior as opposed to bagging tree approach as it's less prone to overfitting. In practice, Random Forest usually has high performance.

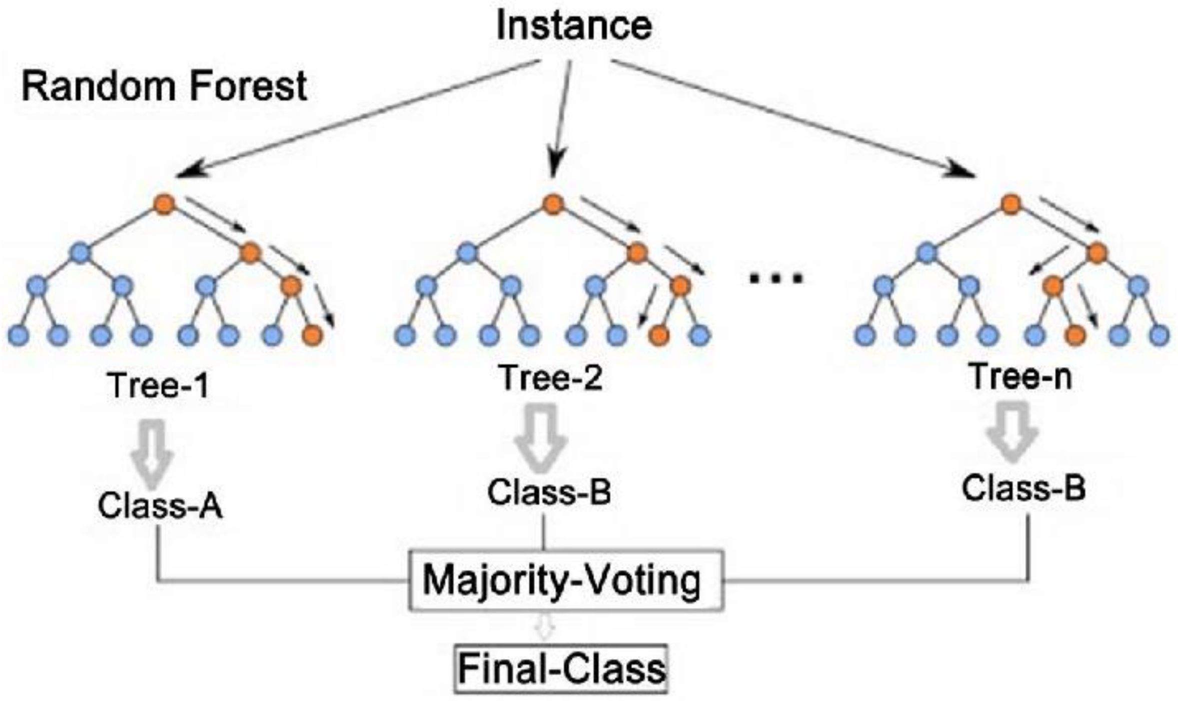


Figure 5 Random Forest Illustration

The advantages of Random Forest Model:

- More flexible
- Similar to tree-based models, preprocessing is not really necessary
- Less prone to overfitting

While the disadvantages of Random Forest Model:

- More complex model

NEURAL NETWORK

The last model of the benchmark is Neural Network, which is represented as multi-layer perceptron in scikit library. The way neural network model works mimics the human brain, it has 3 layers: input layer, hidden layer, output layer. Information first goes into the input layer, processed inside the hidden layer, and then outputted through the output layer. Information are passed through the layers and the preceding layers become the input of the subsequent layers. Within this information passing, a lot of computation are done within the neurons, the model could potentially be computationally expensive, depending on the structure of the layering and the number of neurons.

In practice, Neural Network is one of the most advanced models in terms of application. Image recognition, Text to speech, are some of its applications. It is also known as a deep learning model.

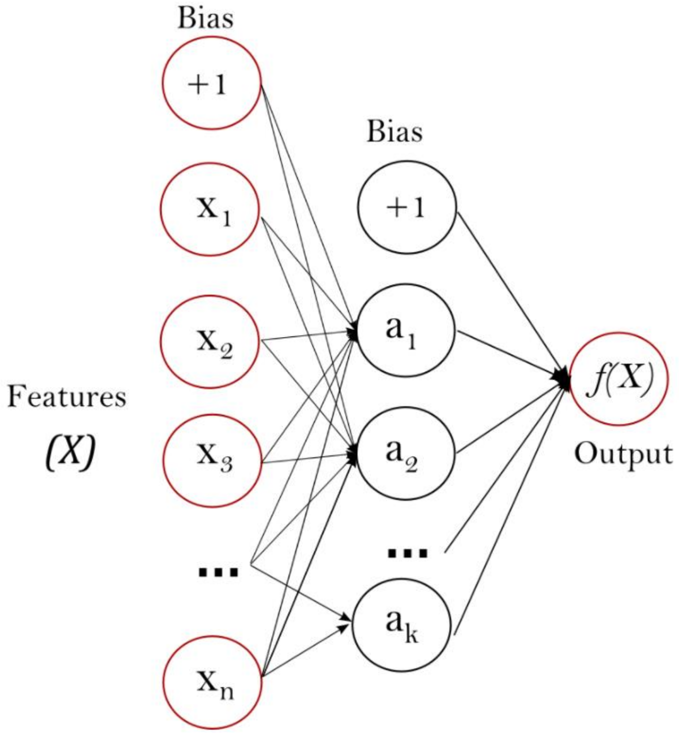


Figure 6 One Hidden Layer Neural Network

The advantages of Neural Network Model:

- More flexible
- Handles non-linearity very well

While the disadvantages of Neural Network Model:

- Less interpretable & complex
- Computationally expensive

EXPERIMENT SETUP

DATA SPLITTING

Before we do any preprocessing on the data, we first split our data into training & testing. The reason to do this is so that we will focus building our model on the basis of the training data, without knowing anything about the testing data, mimicking real life scenarios. The testing data will be used as a final evaluation of the model.

DATA PREPROCESSING

Before developing the model, we first do the basic preprocessing to our data. For categorical variables, we set up dummy variables through one hot encoding technique. This ensures that for our categorical data, all values are treated equally. In addition to this, missing values are imputed with most common occurrence. A flag variable to indicate rows where missing values occur is also created. For continuous variables, missing values are imputed with average values. Finally, the data are scaled using robust scaler so it becomes less outlier sensitive. Data preprocessing are done both for the training and testing data, however, it is worth noting that for testing data, we used the fitted transformer from the training set. This approach is done in order to make sure there is no potential for information leak, on our training set.

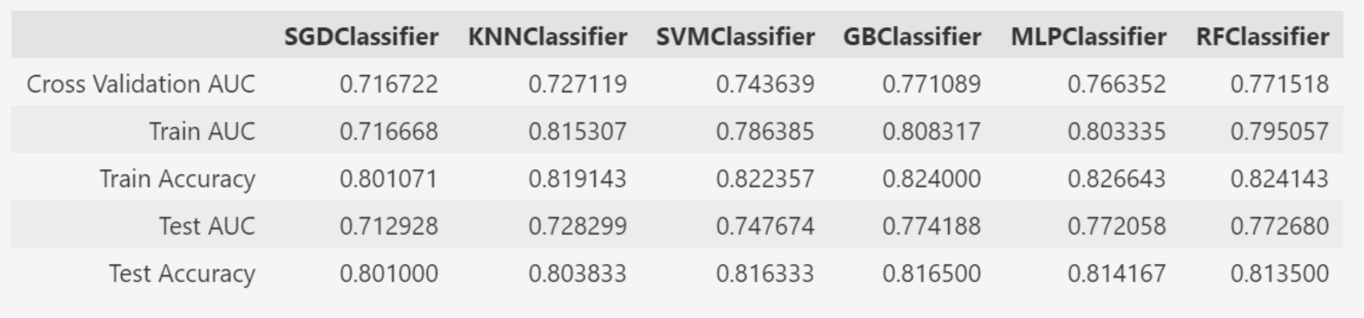
FEATURES SELECTION

For the feature selection, we're taking the univariate selection approach. We're comparing each feature with the target variable, calculate its pearson correlation, and evaluate the corresponding p-value. From this, we will determine whether if a feature is significant to our target variable. In case it's not, we will drop them from our model.

MODEL DEVELOPMENT & BENCHMARKING

Each model is developed using the same processed dataset. For each model, a predefined range of hyperparameter is declared. The models are developed using the gridsearchCV function from sklearn. This automatically build models with all possible combination of hyperparameter. The function also incorporates Cross validation techniques, which means we will also report the Cross validation AUC score. For the benchmarking process, the scoring criteria compared is Area Under the Curve (AUC) and accuracy.

RESULT



	SGDClassifier	KNNClassifier	SVMClassifier	GBClassifier	MLPClassifier	RFClassifier
Cross Validation AUC	0.716722	0.727119	0.743639	0.771089	0.766352	0.771518
Train AUC	0.716668	0.815307	0.786385	0.808317	0.803335	0.795057
Train Accuracy	0.801071	0.819143	0.822357	0.824000	0.826643	0.824143
Test AUC	0.712928	0.728299	0.747674	0.774188	0.772058	0.772680
Test Accuracy	0.801000	0.803833	0.816333	0.816500	0.814167	0.813500

Figure 7 Benchmarking Result

From the of the benchmark, we know that the best performing model for this particular dataset is either Random Forest or Gradient Boosting, the difference in terms of the metrics is not significant, this is followed by the Neural Network model. While on the other hand, Logistic regression, KNN, and SVM does not perform as good as the other models for this dataset. Another thing that we notice from the result is the difference between Training & Testing AUC. The difference between these number could indicate whether a model has been underfitted or overfitted, if the difference is significant (say, above 0.05), then we can conclude there might be an indication of overfitting. This phenomenon could be seen in the KNN model, where the AUC training & testing has a significant difference.

CONCLUSION

To conclude our benchmarking experiment, for this dataset, Either Gradient Boosting or Random Forest is the best performing model. Some possible recommendation to further improve this project:

- Try different algorithms
- Try more robust hyperparameter combinations
- Try evaluating the performance of the model on an out of sample, independent test set
- Try considering hybrid model as well

REFERENCE

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2014). An Introduction to Statistical Learning: With Applications in R. Springer Publishing Company, Incorporated.

Benjamin Aunkofer (2017). Ensemble Learning. Link: https://data-science-blog.com/blog/2017/12/03/ensemble-learning/

Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.