DeepRating: A Deep Learning Approach to Credit Default Forecasting

Raphael Abbou
Stanford University
MS in Computational & Mathematical Engineering
abbou@stanford.edu

Abstract

Predicting the likelihood of credit events when providing a family with a loan is primordial for loan agencies, in order to price the risk of such an investment, and give a fair interest rate to the person getting it. However, forecasting the probability of a default giving information at loan origination appears to be a difficult task. Research in Deep Learning toward this task has been very sparse, however neural networks are interesting candidates to grasp the complexity of the relationship between the default of a loan and its features. We present below our approach for forecasting the default probability of single-family loans based on an extensive dataset from Freddie Mac. We build a Deep Neural Network to predict if whether or not a loan is going to default for a two year time-horizon from origination.

1 Introduction

Modeling credit risk and being able to provide trustworthy credit ratings for loaner, whether they are people or firms, is crucial for the welfare of the economy. Indeed, loans are the fuel of economic growth, however granting one must be based on solid evidences that the counterparty will be able to pay it back in the future. Being able to provide an accurate measure of the probability of default of a loan is thus crucial: investors would not be able to finance loans through financial instruments like mortgage-backed securities if those could not be priced in an accurate manner thanks to default risk modeling.

While statistical methods and Markovian processes have been used for a long time in the determination of default probability, research in the Deep Learning area toward this problematic is still very sparse. However, as the relation between the default of a loan and its causes is highly non-linear (see Sirignano, 2016), Neural Networks seems pertinent in order to approximate a good predictive model.

The project's code can be found at the address below:

`https://github.com/RaphAbb/DeepRating`

1.1 Problem Overview

First, we want to emphasize the complexity of defining a meaningful problematic. An interesting output would be to get, for a set of states (delinquency status), the probability to transit from one state to the other, given the current feature values for a loan. In this study, we try to establish a more simple model, which aims at showing the relationship between origination features of a loan and its probability of default in the short-term. We have to select different periods for train/dev and test sets, since the model we train should be useful to predict default happening after the neural network creation and tuning. The probability of default of a loan is historically correlated to time-dependent factors (mostly macro-economics), such as unemployment rate, the risk-free rate, movements in global indexes such as S&P500 (see Sirignano, 2016). In our approach, we want to get ride of any time-dependent parameter, since we use a 'static model' that does not take into account the evolution of parameters. Hence, we only focus one 2-year data (we have selected year 2014-2015 for training set, and 2016-2017 for dev set), and assume that changes in those macro-economic data are small enough so that their impact on default probability can be neglected. With almost 3 million loans for 2014-2015, our dataset is still large enough to focus our study on a particular year.

Each loan default falls into 6 different categories (REO disposition, Short Sale, NLP/RLP Loan Sale...). We first wanted to output a classifier that would give te probability for a loan to be in one of those categories, but some classes had too few data. We thus switched to a binary regression to know if whether of not a loan was defaulting during its 2 first years after its origination.

2 About the DataSet

2.1 Dataset Description

Our total dataset consists in 26.3 million fixed-rate mortgages, originated from January 1, 1999 and September 30, 2017, provided by public company Freddie Mac. For each mortgage, 26 input criteria are used to determine the quality of a given loan. We restrained our study to loans originated between 2014 and 2017. Various criteria are used: qualitative data such as Postal Code or the kind of investment the loan has made (called 'Occupancy status' in the dataset) and quantitative data like Credit Score to more complex financial metrics like Original Debt-to-Income Ratio and Original Loan-to-Value (all the criteria are thoroughly described in the dataset's General User Guide).

2.2 Data Preprocessing

Freddie Mac's dataset is contains two kind of files for each year: the origination files, which provide for each line a loan ID and 26 features computed at loan origination time, and monthly data files, which provide for each line monthly information about a loan, that are evolving through time, until a credit event happen (with the corresponding zero-balance value). For our chosen evaluation time-window (that we have fixed to 2 years), we look for each loan if a credit event has occurred in the monthly file, then use this information as our output value. We furthermore normalize all inputs values.

Features with String Values

Many of our features have string values (State in which the loan has been evaluated, Loan purpose...). In order to make them usable in our neural network, we make equivalences between our strings and integers. Each time we encounter a new string value in our training set, we increment the equivalent value by one. In ou testing set, we make all new unknown string values (i.e. that we had not seen in the training set) correspond to 0.

Table 1: Equivalence between Strings and Numeric Values

Feature	Transco
Training Feature 1	1
Training Feature 2	2
...	...
Test Feature 1	1
Test Feature n	n
Unknown Test Feature 1	0
Unknown Test Feature 2	0

Managing Missing Values

Many features have some missing values in our dataset, which can be split in two types:

- Values that are missing because information had not been provided when the loan was originated. We can guess that those missing values will not have too much impact, since it is likely that they are randomly distributed across our dataset, hence only creating noise.

- Values that are missing for confidentiality purpose. Indeed, for some features (including FICO score, Debt-to-Income Ratio...), Freddie Mac does not provide their values for given loans if they are above or below certain thresholds (cf the 'Valid Values' paragraph in Freddie Mac's Dataset user-guide). Those missing values are more problematic, since it remove the tail distribution for those features, that are probably important factors in the potential default of a loan.

Our first approach was to add a column, for each feature displaying missing values, filled with 1's for missing values and 0 for non-missing values, for each loan. In that way, we would add more information in the features related to missing values. However, this technique did not improve the results, and made even more complex (and bigger) the dataset, so we did not keet this idea for the rest of the study. We then set all missing values with 0's.

3 Modelization

Each loans constitutes an example in our training set, and the features are the neurons of our input. We then send it to a Deep Neural Network, with a binary output, giving 0 for a loan that is still live after two years from its origination, and 1 if it has defaulted during this period. Metrics

Our data is highly unbalanced: we have much more examples of non-defaulting loans than defaulting ones (96% and 4% respectively for our training set). Therefore an accuracy metric would be pretty high even if we fail to predict accurately defaulting loans. We thus choose to valid our model with an AUC metric.

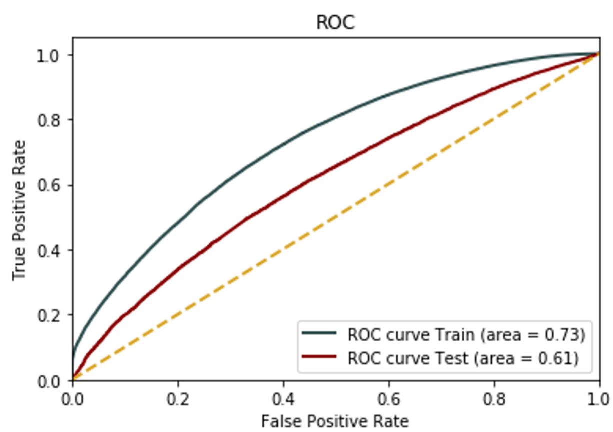

Figure 1: First Results: High Bias and High Variance

3.1 First Results

Reducing Bias

We can observe at first that our network lacks the ability of predicting accurately defaults (its learns too much to predict surviving loans only). We thus add weights in the cost function to the term representing the penalty of incorrectly predicting non defaults. Furthermore, in order to reduce bias, we increase the size of our dataset (going from data on a one year period to two years). Data augmentation appeared to be highly profitable for our bias reduction. We also increase the depth of our neural network (going from a 3-layer DNN to a 10 one).

Reducing Variance

We can observe that even if our Neural Network is now doing a good job on the test set, it is still doing quite poorly on the dev set. We thus add dropout and L2 normalization in order to reduce it.

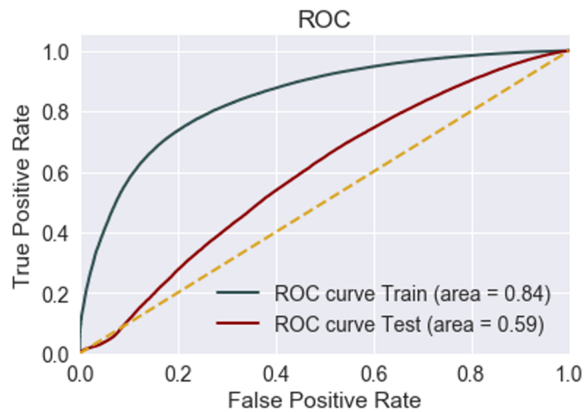

Figure 2: ROC After Bias Reduction

After weighting our cost function and applying a L2 normalization, our cost function is:

$$\mathcal{J}(\hat{y}_i, y_i) = -\frac{1}{m}\sum_{i=1}^{m}[\omega y_i log(\hat{y}_i) + (1-y_i)log(1-\hat{y}_i)] + \frac{\lambda}{L}\sum_{l=1}^{L}||W^{[l]}||_F^2 \tag{1}$$

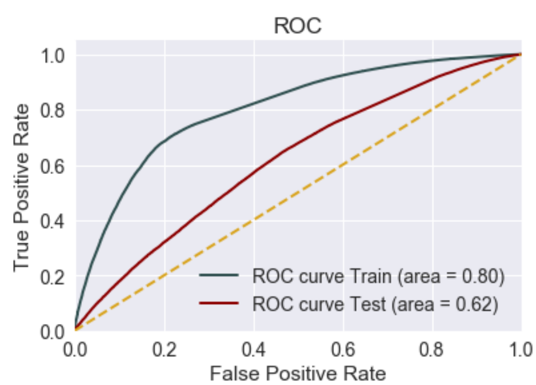

Figure 3: ROC After Bias and Variance Reduction

We did not manage to improve a lot our performance on the dev set, even after tuning hyperparameters like the dropout rate, λ.. Therefore, our features are not sufficient to generalize to loans in a different time-window, despite taking two nearby time periods. However, we still manage to get results accurate enough to show the correlation between our features on the probability of default of a loan in the short term. We believe that what lacks ton our model is the fact to take into account the evolution of macro-economics trends, that would have a important impact in the ability of a loaner to pay its debt back.

4 Conclusion & Future Work

In a more complete study, we would like to implement a model that is not constraint to the forecasting of a default on a fix (two years) time window, but a vanilla Deep Neural Network is not sufficient for taking time-dependent parameters. Another approach would be to consider Recurrent Neural Networks to take into account time series (including macro-economics parameters) as inputs: however, we would have to change our baseline approach, and try to forecast a transition matrix (providing the probability of going from one delinquency status to an other), more than a default probability on a fix period, in order to make the model useful in the industry.

References

[1] Sirignano, J.A., Sadhwani, A. & Giesecke, J. (2018), *Deep Learning for Mortgage Risk*

[2] Swanson, N.R., & White, A. (1997), *A Model Selection Approach to Real-Time Macroeconomic Forecasting Using Linear Models and Artificial Neural Networks*

[3] Chen, G., & Yang, S. (2018), *Application of Deep Learning to Credit Risk Modeling*