How To Prevent Another Financial Crisis On Wall Street

Helin Gao
helingao@stanford.edu

Qianying Lin
qlin1@stanford.edu

Kaidi Yan
kaidi@stanford.edu

Abstract

Riskiness of a particular loan can be estimated based on several of its economic indicators. Although the actual relationship is very complicated, it is possible to construct a simplified model which can roughly classify a loan into one of the three categories - high risk loan, medium risk loan and low risk loan - based on certain criteria. This paper seeks to build such a model based on publicly available information as well as various machine-learning techniques.

1 Introduction

During the 2008-2010 financial crisis, one of the principal culprits that led to the later domino effect of financial meltdown was Collateral Debt Obligations (CDO-s), a type of structured asset-based security, which allows allocation of interest and principal payment based on seniority. What happened during financial crisis was that many CDO-s were overrated by rating agencies due to asymmetric information. This misleading rating was aggravated when a subset of CDO-s pool the junior tranches of Collateral Mortgage Backed Securities (CMBS-s) together and thereby creating systemic risk for security holders. From 2004 to 2007 alone, the issuance of CMBS increased from 93 billion to 230 billion. Eventually, CMBS delinquency rate skyrocketed, paralyzing the entire economy. Curiously, many factors may influence loan default rate, including loan type, LTV ratio (Loan-to-value ratio), benchmark and average spread of interest rate. This project aims to fix the systematic risk created by CDO-s by applying machine learning algorithm to derive a method in classifying risky assets. By comparing the accuracy and F-scores of different machine learning algorithms, we managed to derive an optimal method in classifying loan risk and therefore contributing to improved accuracy of risk classification of CMBS bond.

2 Data Processing and Intepretation

We have 147 data points collected from 2013 CMBS transactions in California and 126 data points in New York state, each of which detailing the characteristics of loaner as well as current loan status. After eliminating those with missing data, we are left with 134 training examples from California and 119 training examples from New York. All of our data are available in the company filings of US Securities and Exchange Commission (SEC) website

In particular, each loan has its associated estimated average spread of interest rate. In general, the higher the average spread of a loan is, the more risky the loan will be (in other words, it will be more likely the loan will default). Based on the average spread, we are able to categorize loan riskiness into low (0-250 bps), medium (250-400 bps), and high (above 400 bps) risk assets. (Hong, Tang, 2008). And our objective is to correctly classify each loan into one of these categories based on its various economic indicators.

We have 61 potential variables for each loan, some of which, such as property name, trustee loan ID, are clearly irrelevant to the project. Based on economic theory, we decide to select the following seven variables for our preliminary analysis of the data:

1. Debt-service-coverage ratio under Net Operating Income (NOI): This feature captures the ratio of NOI available for debt servicing to interest, principal and lease payments. Based on the general trend of market, there is likely to be a negative correlation between NOI and loan default rate. Our DSCR NOI ranges from 0.86-8.16

2. Debt-service-coverage ratio under Net Cash Flow (NCF): This feature captures the ratio of NCF available for debt servicing to interest, principal and lease payments. Similar to NOI, a higher NCF probably indicates stronger financial

strength of the loaner based on economic theory. Our DSCR NCF ranges from 1.19 to 6.49.

3. Debt Yield (DY): Defined as the NCF divided by first mortgage debt expressed as a percentage. It describes the relative size of the loan compared to the companys total cash flow. In this data set, the DY ranges from 6.69 to 23.

4. Major Type (MT): describing the commercial purpose of CMBS bonds. There are 7 main categories in this case, namely, IN (industrial) LO (lodging) MF (multi-family) MU (mixed use) OF (office) RT (retail) SS (self-storage).

5. BenchMark (BM): long-term debt obligation issued by the government. We use 4 types of benchmark, namely, 5, 7, 10 treasury bill rate and Libor 1-month borrowing rate.

6. Loan-To-Value (LTV): loan to value ratio used by lenders to describe the ratio of loans to the value of the asset purchased. The higher the LTV ratio, the higher the risk as perceived by lender. In this project, LTV ratio ranges from 30.3

7. Time to Maturity (TM): Measuring the time between when the bond is issued and when it matures. Higher maturity usually indicates greater default rate risk subject to the fluctuation of macroeconomic condition as well as higher bond durationsensitivity of the price to a change in interest rate. Therefore, higher time to maturity usually corresponds to higher spread.

3 Feature Selection

3.1 MI Scoring

For feature selection, we first use Mutual Information (MI) to examine the correlation between each feature and the outcome. Based on class, we have the following formula for calculating MI score:

$$MI(x_i, y) = \sum_{x_i} \sum_{y} p(x_i, y) \log \frac{p(x_i, y)}{p(x_i)p(y)} \quad (1)$$

Since some of the features, such as Loan-To-Value, are continuous variables, we use the most straightforward discretizing technique (Butte and Kohane, 2000; Michaels *et al.*,1998) in order to apply equation (1). Below are the MI scores of the seven features we choose in the preliminary analysis, whose correlations with the average spread are all greater than 1:

Feature	MI score
Net Operating Income (NOI)	2.0505
Loan-To-Value (LTV)	1.8289
Debt Yield (DY)	1.7901
Major Type (MT)	1.5795
Net Cash Flow (NCF)	1.5473
Time to Maturity (TM)	1.5090
Bench Mark (BM)	1.1030

Table 1: MI scoring

3.2 Symmetric Uncertainty

Next, we try to reduce certain features which are potentially repetitive. In order to do so we calcualte the *symmetric uncertainty* (SU) (Press *et al.*, 1988) between each pair of distinct features from the seven features above. We choose to calculate SU instead of the linear correlation coefficient since Major Type and Benchmark have no numerical values. We first define the *entropy* of a variable X as:

$$H(X) = -\sum_{x_i} P(x_i) \log P(x_i) \quad (2)$$

and the *conditional entropy* of X based on Y as:

$$H(X|Y) = -\sum_{y_j} P(y_j) \sum_{x_i} P(x_i|y_j) \log P(x_i|y_j) \quad (3)$$

Next, we define the *information gain* of X provided by y (Quinlan, 1993) as:

$$IG(X|Y) = H(X) - H(X|Y) \quad (4)$$

Finally, we can calculate SU using the following formula:

$$SU(X, Y) = \frac{2IG(X|Y)}{H(X) + H(Y)} \quad (5)$$

One can easily check that the range of SU value is $[0, 1]$. Intuitively, the larger $SU(X, Y)$ is, the more likely knowledge of one variable can predict the other variable, hence the more closely two variables are correlated.

Based on the formula given, we are able to calculate the SU score between each pair of features. The results are listed in the following table:

	LTV	MT	BM	NCF	NOI	DY	TM
LTV	1.00	0.26	0.26	0.23	0.36	0.31	0.36
MT	0.26	1.00	0.37	0.23	0.24	0.28	0.27
BM	0.26	0.37	1.00	0.24	0.30	0.29	0.51
NCF	0.23	0.23	0.24	1.00	0.33	0.30	0.45
NOI	0.36	0.24	0.37	0.33	1.00	0.35	0.35
DY	0.31	0.28	0.29	0.30	0.35	1.00	0.40
TM	0.36	0.27	0.51	0.45	0.35	0.40	1.00

Table 2: SU scoring

Based on the table, we can see that Bench Mark (BM) and Time to Maturity (TM) have a relatively high correlation ($SU(BM, TM) > 0.5$). Considering the fact that TM has a higher MI scores than BM does, we decide to eliminate BM from our feature list and use TM instead. Notice also that this result makes economic sense: in defining BenchMark, we have already incorported the factor of time (1-month, 5,7,10 years). Thus it is no surprise that BM will have a high correlation with Time to Maturity, which is also an indicator of time.

4 Model Selection

In the beginning, we consider three different models: Naive Bayes (NB), Logistic Regression (LR),and Support Vector Machine (SVM) using linear kernels. Our SVM model is solved by Sequential Minimal Optimization due to its efficiency. And below are the accuracy rate, precision, recall and F-score for each model (with 10-fold cross-validation):

Model	Accuracy	Precision	Recall	F-Score
NB	58.20%	0.581	0.582	0.569
LR	69.70%	0.673	0.761	0.714
SVM	75.65%	0.679	0.657	0.643

Table 3: Performance of different models

We first try NB because it gives an easy way to see the trend between outcome and different features. Yet based on our result, NB gives a fairly low accuracy rate of less than 60%. This implies that the NB assumption that different features are conditional independent given the outcome is probably not true in our situation. In fact it corresponds to the acutal economic situations in the real world - different features affect each other in a complicated way, thus they can never be conditionally independent with each other.

Logistic Regression and SVM, on the other hand, give higher accuracy rate on the training examples. However, these two models are still unsatisfying because their F-scores are low. In other words, our LR and SVM models are not robust enough to handle with skewed class cases. (The problem with skewed class is present since many of the loans fall into the 'medium risk' category) Besides, by manually checking the data, we see the following problems:

Hayward, CA	LO	435
Oakland - Airport, CA	LO	435
Oakland - Embrcadr, CA	LO	435
Pinole, CA	LO	435

Figure 1: Problem of similar data (4 listed here)

In total, there are 30 training examples in our California loan data with the same type (LO), very similar LTV values and the same average spread (435bps) as shown in Figure 1 (Due to space limit, the LTV values are not displayed in Figure 1). Therefore our training samples are far from being uniformly distributed. Also, we have anomalous data points with unusually high LTV values. These factors contribute to the low F-score of our LR and SVM models, and affect their accuracy rates negatively.

In considering the disadvantages of all the models presented above, we decide to try Decision Tree as our new model. The reasons for choosing Decision Tree are:

1. This model is time-efficient since we only need to build the tree once;

2. Unlike NB, LR or SVM, Decision Tree is non-parametric. This allows us to deal with outliers (data which has anomalous behaviour) with higher confidence, thus boosting the accuracy of our model.

3. Based on certain splitting criteria, Decision Tree can be relatively insensitive to an unbalanced data distribution. (Drummond, Holte, 2000)

In particular, we use C4.5 Algorithm (Quinlan, 1993). It makes use of the *information gain* which we have calculated above. And splitting criteria based on information gain has relatively good performances in

dealing with skewed class and unbalanced data distribution (Monard, Batista, 2003). The outlined C4.5 Algorithm for building a decision tree is:

1. For each feature X in the list, find the normalized information gain ratio from splitting on X
2. Choose $X_{optimal}$ which gives the largest normalized information gain ratio
3. Create the decision node which split the training data based on $X_{optimal}$
4. For each subset of the training data, repeat Step 1 until the base case is reached (meaning that all the training data belong to the same class)

5 Result and Analysis

5.1 Training Result

The Decision Tree built based on C4.5 Algorithm has the following accuracy rate, precision, recall and F-score (In reality, we use an open-source package which allows us to run the algorithm with 10-fold cross-validation):

Accuracy	Precision	Recall	F-Score
83.15%	0.848	0.815	0.831

Table 4: Performance of Decision Tree

As we can see from the table, Decision Tree yields the highest accuracy among the four models (NB, LR, SVM and Decision Tree). This is expected since Decision Tree, as argued in the previous section, deals with outliers with highest confidence. Decision Tree also gives the highest F-score, indicating that our Decisoin Tree does not produce too many true-negative or false-positive predictions. It shows that our Decision Tree is relatively insensitive to the skewed class as well as the uneven distribution of data.
A graphical illustration of the Decision Tree we build in shown in Figure 2 (figure on next page). An importance observation is that in this Decision Tree, only four features are concerned in the end - NOI, NCF, TM and MT. One possible reason is that the rest two features are not able to split the data as much as the first four features do, thus absent in the Decision Tree model.

5.2 Generalization Error on New York Data

Finally, we examine the generalization error of our Decision Tree model on the New York Data. And below is the accuracy rate, precision, recall and F-score when the model is run against New York Data:

Accuracy	Precision	Recall	F-Score
76.47%	0.781	0.768	0.774

Table 5: Generalizatoin Error of Decision Tree

In total 91 out of 119 examples are correctly classified. The generalization error of our Decision Tree model on New York Data is slightly higher than our training error. This is an acceptable result, and furthermore it also produces a high F-score on the New York data.

6 Conclusion

In conclusion, we have built a model based on Decision Tree which has reasonable accuracy and F-score. The model is by no means sophisticated, since we are only concerned with less than 10 features here (in reality there can be many more features which play a part on the riskiness of a loan, but that require a much more complicated model and also much longer time for training and lerning). Nevertheless, this model is still useful to a certain extent, as it provides a simplified perspective on the type of loans and range of NCF and NOIs which will likely give a higher risk loan. And cross-reference with New York data shows that this model has a relatively small generalization error and thus is less likely to suffer from the problem of over-fit.

7 Future Work

Despite the simplicity and relatively high accuracy, there are still some problems associated with our Decision Tree Model, motivating us to work further on this project:

1. The New York and California data turns out to suffer from the problem of skewed class. If the data are more evenly distributed among different classes, SVM and LR might give us appropriate models - the advantages of Decision Tree might not manifest themselves in those circumstance.

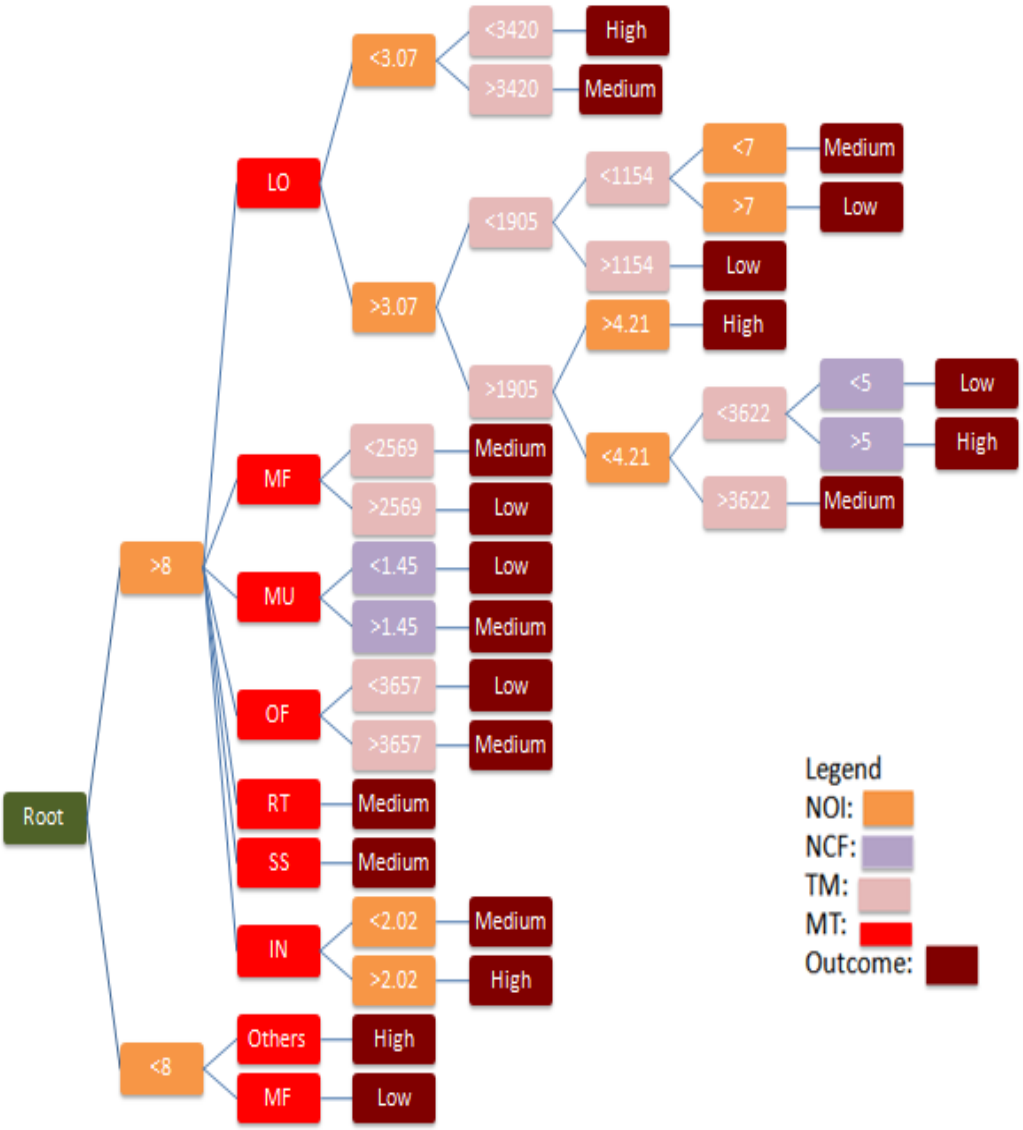

Figure 2: Decision Tree trained using California Data (self-drawn)

2. The riskiness of loans are dicretized into three categories - in reality, this might be over-simplified. Thus we want to see if we can quantify the riskiness through other parameters and construct a quantitative prediction algorithm based on that.

8 Acknowledgement

We would like to express our gratitude to Keith Silates (ICME, Computer Engineering, Stanford University) who accepted to mentor our project.

9 Reference

L.Yu and H.Liu, "Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution", *ICML*,2003.
R.Steuer and J.Selbig, "The Mutual Information: Detecting and Evaluating Dependencies Between Variables", *Bioinformatics*, Vol. 18, pp 231-240.
D.Tang and H.Yan, "Market Conditions, Default Risk and Credit Spreads", *Discussion Paper, Banking and Financial Studies*, 2008.
Andrew Ng, CS229 Lecture Notes, 2013.