

# Predicting Mortgage Backed Securities Prepayment Using Machine Learning Methods

*by mohit garg*

---

**Submission date:** 28-May-2022 11:28AM (UTC-0500)

**Submission ID:** 1845969496

**File name:** Conference-template-A4.docx (735.46K)

**Word count:** 2982

**Character count:** 16149

# Predicting Mortgage Backed Securities Prepayment Using Machine Learning Methods

Vinamr Kanodia (19ucso67)

Mohit Garg(19ucco27)

Nitish Khunteta(19ucso96)

[Link of our code](#)

---

## Abstract

Mortgage-backed securities are becoming more important in India as they are likely to be listed and traded in the public domain. This prompted an analysis of various aspects of product pricing. The price of these products is determined by various factors of risk, and prepayment is one of the most important. This research paper seeks to identify and analyze two factors: Prepayment risk associated with mortgage-backed securities in the Indian market where the underlying asset is based on mortgages to individuals in India. The data collected is related to the existing pool of freddie mac single family loan dataset.

---

## 1. INTRODUCTION

Mortgage-backed securities (MBS) are bonds and are one in every of the biggest asset instructions in the economic markets. Mortgages are the maximum commonplace manner to get a fixed amount in a quick time period. for instance, to cowl a huge amount of cash. shopping for a house. A loan is a contract between two parties, a mortgage, which is often a financial institution, and a mortgage lender, who's the individual or company making use of for the loan. Mortgages are creditors and lend cash. In return for this carrier, he gets a part of the hobby calculated based totally on the amount (conceptually) borrowed from the mortgage lender. Loan lenders have the option of repaying a component in their main earlier than the agreed time. These anticipated bills are called prepayments. This prepayment feature makes the loan portfolio period probabilistic. loan forecast coins flows may not match real registered cash flows. Therefore, prepaid options are a risk to the lender. This creates two types of risk, especially liquidity risk and interest rate risk.

## 2. Literature survey

The importance of prepayment assumptions in estimating mortgage-backed securities (CMO) yields is that realized yields can differ materially from estimated yields, depending on the accuracy of such prepayment assumptions. The impact of prepayments depends on the underlying mortgage rate, the current market rate, and whether the prepayment realized is faster or slower than expected. These effects can be summarized as follows .

- High coupon mortgage securities traded at premium "lose" if the realized prepayment is faster than the expected prepayment. Conversely, if the prepayment is slower than expected, you "win".
- Low coupon mortgage securities that are traded at a "profit" discount if the realized prepayment is faster than the expected prepayment. Conversely, if the prepayment is slower than expected, you will "lose".
- Regular coupon mortgage securities traded at face value are not affected by prepayment of principal.

Mortgage players are aware that prepayments will be made, but do not know when and how many. For discounted mortgage securities, it is modest to misunderstand that they are slower than expected upfront payments . In a fully competitive market, if the current market interest rate falls below the existing mortgage interest rate, a well-informed borrower will prepay the mortgage. However, due to various personal factors such as job changes, births, deaths, marriages, divorces, changes in family income, and the enforcement of the sale deadline clause, borrowers are not always able to exercise their early repayment options in a timely manner [ Five]. The difference between current market interest rates and existing mortgage coupon rates is an important factor in predicting mortgage prepayments . The utility discussion is also used to show that various factors such as alternative investment rates, contract rates, penalties imposed, real estate valuations, real estate capital adequacy ratios, housing costs, etc. have a significant impact on the tendency of borrowers to pay in advance. Will be. In addition, fluctuating market interest rates, regional differences, and mortgage age are important factors in explaining prepayment. The prepaid "rule of thumb" assumptions currently in use are inadequate.

### 3. RELATED WORK

The application of machine learning (ML) technology to credit-related issues has been done before. Not necessarily applied specifically to prepayment of loans.

In Han's dissertation , various ML methods are applied to repay loans. In this task, Han uses various methods such as Random Forest, Support Vector Machine (SVM), and Logistic Regression to estimate the accuracy of each model. In the end, Han found that logistic regression was the best predictor of loan repayment, and Fico scores and annual income were the best predictors of repayment.

The work of Zhang and Holm focuses primarily on logistic regression. Zhang looks at the probabilities as well as the classification of failures. The results show that the use of "survival analysis" proves to be a formidable enemy of logistic regression. In fact, their survival analysis method replaces logistic regression in the study of probability. Holm's study found that the amount of assets and the length of time he was in the bank were factors in whether a person defaulted. What's interesting here is an introduction to how long a person is in the bank. This is a useful indicator of future work using the machine learning mortgage model. There are some works that focus primarily on prepayment. They provide a narrower framework for how we can tackle our research issues.

In Amar's paper , they see if they are using neural networks to check prepayments. They perform a sensitivity analysis to better understand the results and divide the prepayment into two types. This white paper focuses on the usefulness of neural networks in prepaid decisions.

Another example is from our backyard in Drembles. His work envisions loan defaults as well as prepayments. This document uses the LSTM architecture and trains over a 20-month data period. They found that adding layers improved accuracy, but could lead to overfitting. This paper introduces the idea that prepayments can be predicted using deep neural

### 4. Proposed Approach

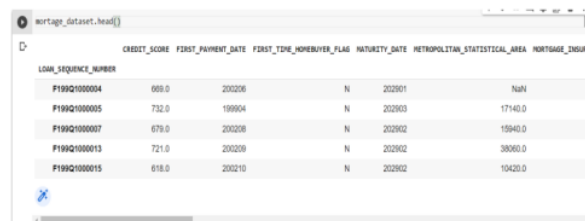
Prepaid modeling is essential for bank balance sheet planning and risk analysis. This is also one of the most complex areas of financial modeling. This is because the potential risk factors are highly non-linear and interactive, which can change the regime in credit availability and borrower behavior. In addition, prepayment with Home sales and prepayment with refinancing incentives may have different sensitivities to risk factors .For example, refinancing incentives tend to increase with the amount of loans, and housing sales often decrease with the amount of loans . There is another reason for partial

prepayment. Therefore, many submodels are required, each with its own risk factor dependencies specified and estimated. However, our data does not include reasons for prepayment. It makes it impossible to separate these models.

Logistic regression techniques are used by most banks to predict whether mortgages will prepay or not. In recent years , Approaches like Random forest and artificial neural networks are also in use. Therefore, the goal of this paper is to create a model to predict the prepayment using random forest and artificial neural networks.

### 5. DATASET AND FEATURES

We have taken Freddie Mac's single family loan level dataset for training , validation and testing of our models. This dataset contains 500137(500k) mortgage data with 26 features like CREDIT\_SCORE, PREPAID, FIRST\_PAYMENT\_DATE, MORTGAGE\_INSURANCE\_PERCENTAGE, NUMBER\_OF\_UNITS, ORIGINAL\_COMBINED\_LOAN\_TO\_VALUE and many more features. And we use 60% , 20% and 20% data splitting for training , validation and testing respectively.

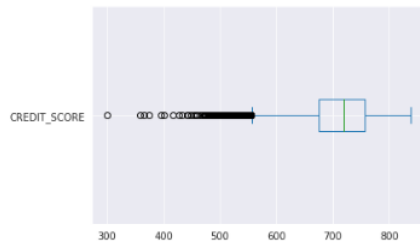


| LOAN_SEQUENCE_NUMBER | CREDIT_SCORE | FIRST_PAYMENT_DATE | FIRST_TIME_HOMEBUYER_FLAG | MATURITY_DATE | METROPOLITAN_STATISTICAL_AREA | MORTGAGE_INSURANCE |
|----------------------|--------------|--------------------|---------------------------|---------------|-------------------------------|--------------------|
| F199Q1000004         | 688.0        | 200206             | N                         | 202901        | NaN                           |                    |
| F199Q1000005         | 732.0        | 199904             | N                         | 202903        |                               | 17140.0            |
| F199Q1000007         | 679.0        | 200208             | N                         | 202902        |                               | 15940.0            |
| F199Q1000013         | 721.0        | 200209             | N                         | 202902        |                               | 30060.0            |
| F199Q1000015         | 618.0        | 200210             | N                         | 202902        |                               | 10420.0            |

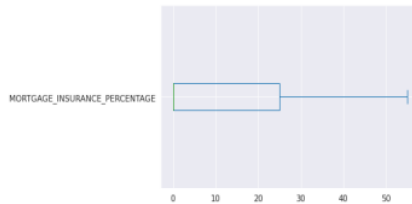
Each vintage year has a source data file and a corresponding monthly performance data file that contains the same loan-level data fields that are contained in the complete dataset. Due to the size of the dataset, the data is split and compressed as described below. The files are chronologically arranged by year and quarter.

Origination Data File :

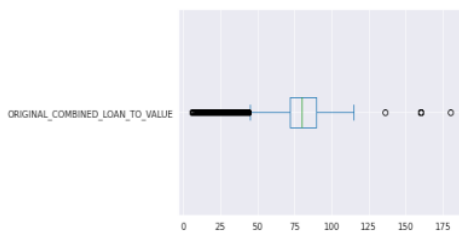
- CREDIT SCORE - Consumer's credit worthiness.



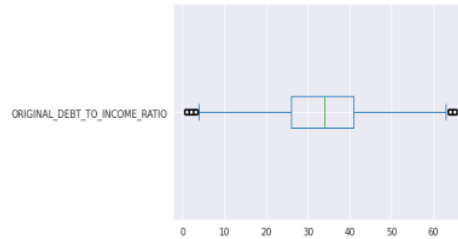
- **FIRST PAYMENT DATE** - first payment date due under the terms of mortgage note.
- **FIRST TIME HOMEBUYER FLAG**- When consumers purchase residential for the very first time.
- **MATURITY DATE**- When a debt comes due and all principal or interest must be repaid.
- **MORTGAGE INSURANCE PERCENTAGE** - The percentage of loss coverage on the loan.



- **NUMBER OF UNITS** - Denotes whether or not the loan is a one, two, three, or 4 unit assets.
- **OCCUPANCY STATUS** - Whether or not the loan kind is proprietor occupied, 2nd home, or investment property.



- **ORIGINAL COMBINED LOAN-TO-VALUE (CLTV)** – Ratio of all secured loans on a belongings to the fee of a belongings.
- **ORIGINAL DEBT-TO-INCOME (DTI) RATIO**- Ratio of all secured loans on a belongings to the cost of assets



## 6. METHODS

### 6.1. Logistic Regression

Logistic regression is one of the most common methods for solving classification problems. In our particular case we check if the mortgage is prepaid ( $Y_i = 1$ ) or not ( $Y_i = 0$ ). Since  $Y_i$  is either 0 or 1, it is distributed as a Bernoulli random variable.

$$Y_i | X_i = x_i \sim \text{Bernoulli}(p_i),$$

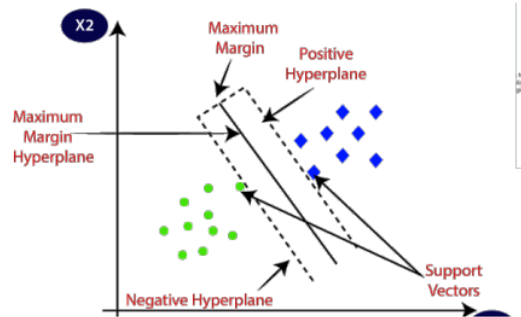
where  $p_i$  is modeled as

$$\text{logit}(p_i) = \beta_0 + \sum_{j=1}^k \beta_j x_{ij} \Leftrightarrow p_i = P(Y_i = 1 | X = x) = \frac{e^{\beta_0 + \sum_{j=1}^k \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^k \beta_j x_{ij}}},$$

where  $\text{logit}(p) = \log(p/1-p)$  is called logistic function. The name of logistic regression came from this function.

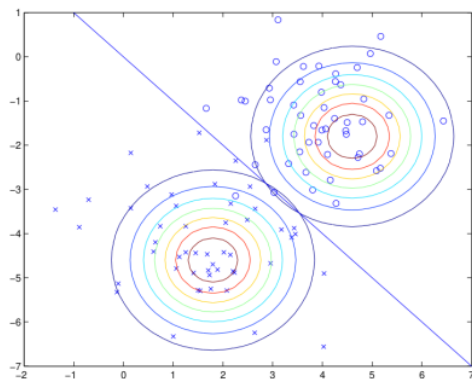
### 6.2. Support Vector Machine

The Supportvector machine works by finding a hyperplane that maximizes the separation between classes. Usually this is a linear hyperplane, but in many cases the data is not linearly separable and often requires a non-linear transformation to higher dimensions. The main problem that arises is that the dimensions can be very large ( $d \gg n$ ) and are computationally infeasible. This is where "kernel tricks" come in handy. With kernel tricks, you don't have to explicitly render the underlying feature map, but you render it through a training example and a linear combination of kernels. This avoids explicitly storing the estimator, which greatly improves run-time efficiency at the cost of requiring all training data when performing inferences. For the purposes of our project, the hyperplane separates the "no prepayment" class and the "prepayment" class and uses kernel tricks to do this.



### 6.3. Gaussian Discriminant Analysis

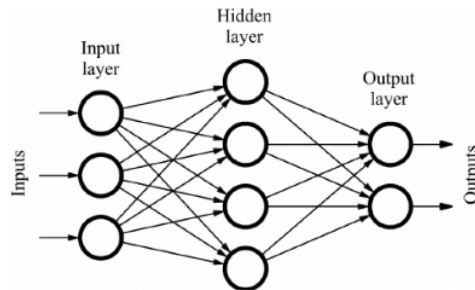
Gaussian discriminant analysis works by specifying training data to create a Gaussian distribution for each class. After these distributions are created, new data points are classified by assessing the probabilities of belonging to each class based on their Gaussian distribution. The distribution to which the new data point is expected to belong is the one most likely to belong to this Gaussian distribution. GDA has two main forms: linear and quadratic. Assuming that all classes have the same covariance matrix, the decision boundaries between the classes are linear. If each class can have a different covariance matrix, a quadratic decision boundary between the classes is established. Both methods were applied and the results were compared using logistic regression. Use GDA to predict prepayments and create two Gaussian distributions. One represents the distribution of unpaid data and the other represents the distribution of prepaid data.



### 6.4. Feed Forward Neural Networks

Neural networks work by stacking and layering many neurons with activation functions to create highly nonlinear functions. Simply put, a single neuron outputs the dot product between the input and the weight and adds a bias to it. This output then passes through the

activation function and is passed to many other neurons. Train these neural networks in a manner very similar to other monitored methods, calculate the slope of the loss function for each parameter, and then perform the stochastic gradient descent method. To predict if a prepayment will occur in a particular month, we will use a dataset to train a neural network that outputs the probability that a prepayment will occur.



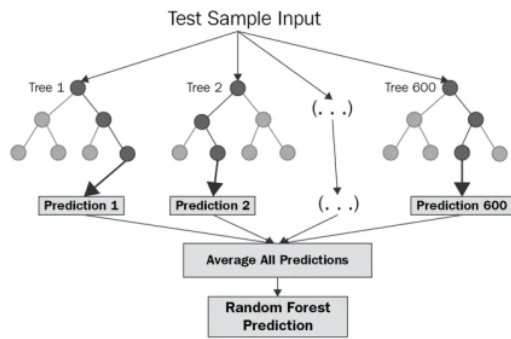
Feedforward neural networks method indicators in a one-way route and have no inherent temporal dynamics. Hence, they're frequently described as being static.

### 6.5. Random Forest

Random forests usually perform well on unbalanced datasets. Another advantage of running a random forest classifier on a dataset is that you can intuitively see the characteristics by listing the importance of the individual characteristics. This provides insight into the factors that affect an individual's ability to repay loans. Also, let's see if introducing some degree of randomness into the classification problem can help improve the accuracy of the results.

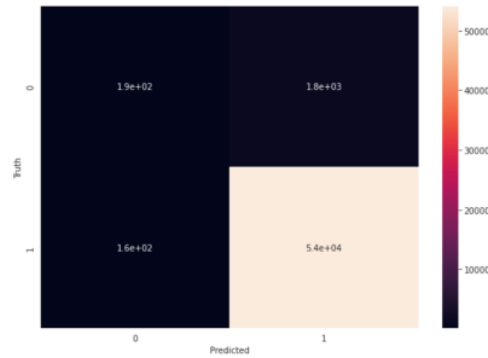
Random forests do not require cross-validation or separate test sets to obtain unbiased estimates of test set errors. It is estimated internally during execution as follows:





the logistic regression and built the model using all the features necessary for the target feature.

The accuracy of model is 0.9647833437138535



## 7. EXPERIMENTS AND RESULTS

### 7.1 Data Classification

We have taken Freddie Mac's single family loan level dataset for our model and using fast\_ml we split our dataset for training , validation and testing . Then perform the classification on the dataset firstly we perform isnull() operation by which we found which features data is null or not present then we drop that row from our dataset by performing dropna() function . Then we wrote the function of mean and median and found the mean and median of every feature and filled the NaN value with the median of that feature.

```
[ ] mortgage_dataset["Missing_CREDIT_SCORE"] = mortgage_dataset.CREDIT_SCORE.isna()
mortgage_dataset.CREDIT_SCORE.fillna(mortgage_dataset.CREDIT_SCORE.median(),inplace=True)
get_plot_and_stat(mortgage_dataset.CREDIT_SCORE)
```

We find the datatype of every feature and then separate the number and bool from the category and objects. To convert category and object data type into the performable data we use OneHotEncoder() and convert it into an array . Finally combine all datatypes together and separate the features of prepaid and delinquent for our target to the dataset.

|                      | CREDIT_SCORE | FIRST_PAYMENT_DATE | PROPERTY_DATE | METROPOLITAN_STATISTICAL_AREA | MORTGAGE_INSURANCE_PERCENTAGE | NUMBER_OF_UNITS | ... |
|----------------------|--------------|--------------------|---------------|-------------------------------|-------------------------------|-----------------|-----|
| LOAN_SEQUENCE_NUMBER |              |                    |               |                               |                               |                 |     |
| P1000210000005       | 752.0        | 199904             | 200003        | 17740.0                       | 0.0                           | 1               |     |
| P1000210000007       | 679.0        | 200006             | 200002        | 10945.0                       | 30.0                          | 1               |     |
| P1000210000012       | 721.0        | 200009             | 200002        | 18060.0                       | 0.0                           | 1               |     |
| P1000210000018       | 610.0        | 200210             | 200002        | 10420.0                       | 20.0                          | 1               |     |
| P1000210000016       | 730.0        | 200211             | 200003        | 10420.0                       | 0.0                           | 1               |     |

### 7.2 Logistic Regression

We first split the dataset into train\_validate\_test\_split . Once we got the target dataset which in our case is 'PREPAID' , we applied

After finding the accuracy of the model via logistic regression we went for L1 regularization. The regularization penalty is the sum of the absolute values of the coefficients, so you want to scale the facts so that all the coefficients are primarily based at the equal scale.

The usefulness of L1 is that you can set the characteristic element to zero to create a feature choice method. The code below plays a logistic regression of 4 instances with an L1 penalty, every time decrementing the value of C. As C decreases, we need to count on greater coefficients to be zero.

Table for training and testing data value for different C values:

| Dataset Type      | C=10                           | C=1                            | C=0.1                          | C=0.001                    |
|-------------------|--------------------------------|--------------------------------|--------------------------------|----------------------------|
| Training accuracy | 0.555<br>21019<br>31986<br>024 | 0.555<br>92794<br>05396<br>749 | 0.556<br>19487<br>13689<br>993 | 0.6630<br>146574<br>68428  |
| Test accuracy     | 0.556<br>73992<br>34807<br>368 | 0.558<br>12794<br>73262<br>746 | 0.559<br>55156<br>15268<br>263 | 0.6639<br>558679<br>597829 |

As C decreases the model coefficient will also become smaller And accuracy may vary but accuracy difference between test and train accuracy will be become smaller and smaller.

### 7.3 Support-Vector Machine

### 7.3.1 Using Different Kernel for better accuracy :

Using the kernel for rbf we have the following test code and result :

```
rbf_model = SVC(kernel='rbf')
```

```
Accuracy = 0.9614022599875434
```

Using the kernel for sigmoid we have the following test code and result :

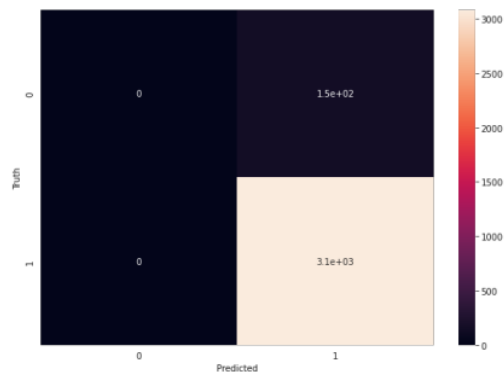
```
sigmoid_model = SVC(kernel='sigmoid')
```

```
Accuracy = 0.9400124566242548
```

Using the kernel for polynomial we have the following test code and result :

```
polynomial_model = SVC(kernel='poly')
```

```
Accuracy = 0.9626835127680399
```



### 7.3.2 LIMITATIONS OF SVM :

1. SVM can take a long time to train.
2. The time complexity of SVM can be  $O(n^2)$ .
3. Finding the appropriate kernel for SVM is not an easy task.
4. If the number of observations is less than the number of features may lead to overfitting.

## 7.4 Gaussian Discriminant Analysis

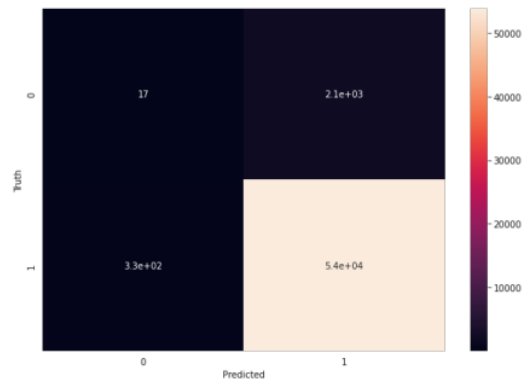
Gaussian Discriminant Analysis is divided into two of LDA and QDA and is one of the most easy to use algorithms.

LDA(Linear Discriminant Analysis) is used for the initial GDA and here covariance matrices per class is a very important part and linear decision boundary or LDA represents the same covariance matrix per class. After

applying the model the accuracy of the model is 0.9521665628614645

**Confusion matrix:**

```
array([[ 17, 2081],
       [ 334, 53763]])
```



## 7.5 Feed-Forward Neural Network

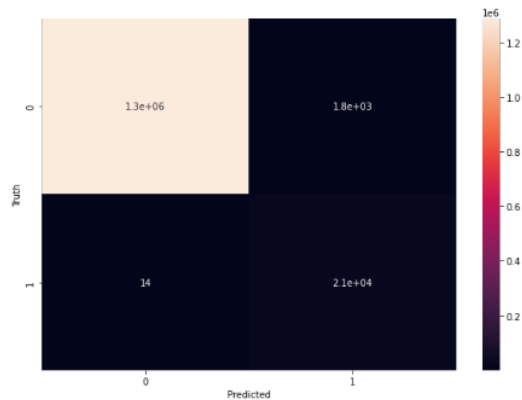
Here we instantiate by the Sequential class this implies that the layers will be stacked on top of each other with the output of each other with the output of the previous layer feeding into next. Then define the first fully connected layer in the network. The input\_shape is set to 131. We then learn 100 weights in this layer and apply the relu function. The next layer learns 50 weights. finally another fully connected layer, with learning of 1 weight

**Confusion matrix:**

```
model = keras.Sequential([
    keras.layers.Dense(100, input_shape=(161,), activation='relu'),
    keras.layers.Dense(50, activation='relu'),
    keras.layers.Dense(1, activation='sigmoid')
])
```

```
array([[1285970, 1824],
       [ 14, 20757]])
```

Accuracy of the model is  
0.99859540794868



## 7.6 Random Forest

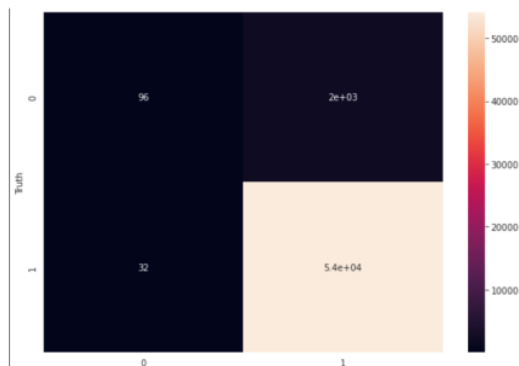
Here we use the function of RandomForestClassifier for the basic data modeling using the train\_validate\_test\_split and also uses the oob function which is basically same as validation but the only difference is it takes the value of left over data after implementation of model

The accuracy of the model in the given model is :

0.9644096449862087

Confusion matrix:

```
array([[ 96, 1968],
       [ 32, 54099]])
```



## 8. CONCLUSION AND FUTURE WORK

In our work we have made a different model for prepayment in MBS and the primary goal for each model is to predict whether the individual will prepay or not. This project contains various ML techniques and we find the accuracy for each technique used in our model.

**Techniques and their Accuracy:-**

Logistic Regression :- 96.478%

Support-Vector Machine

RBF:-96.140%

Sigmoid :- 94.001%

Poly :- 96.268%

Gaussian Discriminant Analysis:- 95.216%

Neural Network :- 99.859%

Random forest :- 96.440%

Out of all techniques we found that most of our model gives accuracy around 96% but the neural network technique to be the top performer with accuracy 99%. Further work needs to be done to explore more feature engineering of the attempted model and compare the results with the pool-level model. This work is a big step in the right direction, and future work will compare these models more closely with questions as well as prepaid.

## 9. References

- [1] Han et. al "Loan Repayment Prediction Using Machine Learning Algorithms"
- [2]Zhang, "Modeling The Probability Of Mortgage Default Via Logistic Regression And Survival Analysis"
- [3] Holm, "Default Prediction of a Swedish Mortgage Portfolio using Logistic Regression"
- [4] Amar, "Modeling of Mortgage Loan Prepayment Risk with Machine Learning"
- [5] Drembles, "Time to Default & Time to Prepayment Estimation Using LSTM and Deep Learning Techniques"
- [https://www.stat.berkeley.edu/~breiman/RandomForests/cc\\_home.htm](https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm)
- [https://cs229.stanford.edu/proj2019aut/data/assignment\\_308832\\_raw/26644913.pdf](https://cs229.stanford.edu/proj2019aut/data/assignment_308832_raw/26644913.pdf)
- [https://www.researchgate.net/publication/354367264\\_Analysis\\_of\\_Loan\\_Availability\\_using\\_Machine\\_Learning\\_Techniques](https://www.researchgate.net/publication/354367264_Analysis_of_Loan_Availability_using_Machine_Learning_Techniques)



- Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems

# Predicting Mortgage Backed Securities Prepayment Using Machine Learning Methods

## ORIGINALITY REPORT

7%

SIMILARITY INDEX

5%

INTERNET SOURCES

1%

PUBLICATIONS

4%

STUDENT PAPERS

## PRIMARY SOURCES

1

[repository.tudelft.nl](https://repository.tudelft.nl)

Internet Source

1%

2

[www.freddiemac.com](http://www.freddiemac.com)

Internet Source

1%

3

Submitted to Universidade Nova De Lisboa

Student Paper

1%

4

Submitted to City University

Student Paper

1%

5

Submitted to University of Southampton

Student Paper

<1%

6

Pengcheng Li, Haohan Hu, Haitao Zhang, Wanlong Liu, Li Zhang. "Abnormal recognition of massive electric power data based on RF and CNN", 2020 7th International Conference on Information Science and Control Engineering (ICISCE), 2020

Publication

<1%

7

[ebin.pub](http://ebin.pub)

Internet Source

<1%

8

academic.oup.com

Internet Source

<1 %

9

digitalcommons.uri.edu

Internet Source

<1 %

10

www.aei.org

Internet Source

<1 %

Exclude quotes On

Exclude matches Off

Exclude bibliography On