HomeCredit Project on Credit-risk Machine Learning

Haolin Li

August 25, 2018

1 Introduction

What makes home credit data interesting is that past data could be used to predict the future pay behavior of current loan applicants. This is made possible by taking consideration of many aspects of applicants' attributes and compare them with modeled previous applicants through comprehensive comparison of their similarity and difference. In general the nature of problem is supervised machine learning with classification as the goal.

- **Supervised**: The labels are included in the training data and the goal is to train a model to learn to predict the labels from the features
- **Classification**: The label is a binary variable, 0 (will repay loan on time), 1 (will have difficulty repaying loan)

1.1 Background and Implication

Traditionally loan decision is performed by grant officers who relies on their experience. As the credit economy grows, the outspurring demand of various loans made this approach no longer viable. And credit score system is later used to supplement such approach. *Credit scoring* is used to distinguish different applicants in their liklyhood in defaulting with repatments. Based on the predictive indicators provided by application forms and the scores from third party *credit bureau* an estimation is made on the likelyhood of repayments. Such estmatation could be made based on the statistical methods. The standard form used are discriminant analysis, linear regression, logistic regression and decision trees, which provide basis for accept and reject decision under certain threshold[Hand, D. J. , & Henley, W. E. (1997). Statistical classification models in consumer credit scoring: A review. Journal of the Royal Statistical Society: Series A (General), 160 (3), 523–541 .]. In order to make sure this underserved population has a positive loan experience, Home Credit makes use of a variety of alternative data--including telco and transactional information--to predict their clients' repayment abilities. Home Credit has challenged Kagglers to further explore their past and present home loan information using machine learning.

Provided data came from 7 sources of data. Well over 48000 applicants measured on more than 200 variables. The main trainning and testing dataset is sepratated in multiple rows of loan application identified by `SK_ID_CURR`. The label is charaterized as `TARGET` indicating whether the loan was repaid or not. In contrast to credit score system where application is distinguished by person. Our dataset descriminates on loan case. As we proceed in the data exploration phase, we shall see how this criterial strengths the bond of person-related characterics in the distribution of `TARGET` variables.

The bureau and bureau_balance datasets showed the monitoring of past loan repayment behaviors. It tracks loans applied at different agency across different time. With such information at hand, it provide means to evaluate future repayment behavior in terms of person. Another different angle is to see how applicants repay their past loan at HomeCredit as provided by previous_application dataset. Installments_payments showed payment history at Home credits.

Multiple source of dataset indicates severe overlaping and heavy coherence among one and another as we shall see in the data exploration. The data objective model(fig.1) shows how all of the data is related.

1.2 Approach

This project tries to improve performance of less complex models by improve on the input space. While choosing the right model and tuning parameter settings are important, the machine model can only learn from the data it is given. The first part of EDA analysis is to search most relevant features out of hundreds

of variables in the database. The preliminary knowledge of features is applied into machine learning model training. And we compare the performance before and after feature engineering.

2 Related work

2.1 Journal published

There are many algorithms available for model construction, so one of the main problems in practice is that of algorithm selection or combination. Unfortunately it is hard to choose an algorithm a priori because one might not know the nature and characteristics of the data set, e.g. its intrinsic noise, complexity, or the type of relationships it contains. Algorithms vary enormously in their basic structure, parameters and optimization landscapes but they can roughly be classified in a few groups (Michie et al., 1994; Weiss and Kulikowski, 1991; Mitchell, 1997).

As earlier as last 70s, the experts at finance industry and mathematitians have been looking into the credit granting decisons in the perspective of statistical classification methods.

In align with methods:

- naive Bayesian: Aida.etl(3) used a database of 924 files of credits granted to industrial Tunisian companies by a commercial bank in the years 2003, 2004, 2005 and 2006. The naive Bayesian classifier algorithm was used, and the results show that the good classification rate is of the order of 63.85 per cent. The default probability is explained by the variables measuring working capital, leverage, solvency, profitability and cash flow indicators.

- Logistic regression: Bsesens et al(Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003).) originated empirical evaluation on eight retail credit scoring datasets from Australian credit (AC) and German credit (GC) now archieved in the UCI Library. The data sets include several covariates to develop PD scorecards and a binary response variable, which indicates bad loans. The covariates capture information from the application form (e.g., loan amount, interest rate, etc.) and customer information (e.g., demographic, social-graphic, and solvency data).

- Linear Discriminant Analysis: Performed on the same database as above, Brown et al.(I. Brown, C. Mues / Expert Systems with Applications 39 (2012) 3446–3453) performed linear discriminant analysis (LDA), quadratic discriminant analysis (QDA) and logistic regression (LOG) classification techniques require without tuning parameter.

- Support vector classification : Maldonado et al.(S. Maldonado et al. / European Journal of Operational Research 261 (2017) 656–665)incorporated SVM's margin maximization with variable acquisition costs in the feature selection procedure. Dataset is orginated from a credit scoring project with a Chilean financial institution. The datasets used have six different groups of variables with different costs. Some attributes come from external sources, while others are combinations of the original variables from different groups in the form of financial ratios.

More recently, artificial intelligence technique such as neural network, support vector machine and nearest neighbor methods are used in the classification task. Online leading has disrupted the traditional consumer banking sector with more effective loan processing. Risk prediction and monitoring is critical for the success of the business model.

- Ensemble-method, Random Forests: Yu.(arXiv:1707.04831v1 [q-fin.RM]) from an online vender service in China glued together data with various format and size from public website, third-parties credit agency in China such as Ali's Zhima-credit and assembled with client's loan application information data (borrowers' national photo ID, two emergent contacts' name, mobile phone and relationship, and a banking account information. 7 days and 21 days' short-term loan with/without collateral is provided). Random forest and XGBoost models are developed and trained with historical borrower's information data from our online lending platform. Models are tested subsequently with evaluation metrics for instance K-S curve, accuracy.

- Neural Net: Y.-M. Huang et al(Y.-M. Huang et al. / Nonlinear Analysis: Real World Applications 7 (2006) 720 – 747) in collaboration waith a research project of Taiwan Bank involves training a data mining system with sampling data contain around 1000 records, accounting for roughly 2% of the TNB's current database.Various models of back-propagation algorithms is trainned including Multilayer Perceptron and RBF muti-clusters.

Other approchees tend to investigate systematic credit-risk models. Amir E. et in their paper utilized bank consumer data in household loans to construct forecasting models of consumer credit risk.

2.2 Kaggle kernels:

- One particular thread portrayed good visualization of missing values for all HomeCredit datasets which implied this as another degree of challenge that HomeCredit invites kagglers to solve. Moreover in application_train_test dataset, such feature accounts for more than 30% of the total number of featuers.
- One post have discussed the feature importance in training the Gradient Boosting models (LGM,XgBoost) which is widely adopted by kagglers. The advantage of such models is ability to generalize sound result after tweaking some of the hyper-parameters such as max-depth and learning rate.

3 Methods

3.1 Exploratory Data Analysis

The goal of EDA is to learn from data that helps improve pattern recognition. For example, previously

3.1.1 Examine the Distribution of the Target Column

The target is what we are asked to predict: either a 0 for the loan was repaid on time, or a 1 indicating the client had payment difficulties. We can first examine the number of loans falling into each category.

From this information, we see this is an imbalanced class problem. There are far more loans that were repaid on time than loans that were not repaid. We are more in the label of 1and we can see this label is unevenly compared to its counterpart. All the work onwards will treat this as positive so we do not inherent a high accuracy.

3.1.2 Variable Types

In the previous query of "application-train" table 121 columns of feature were identified. In a deeper look at the number of unique entries in each of the `object` (categorical) columns. Most of the categorical features have a relatively small number of unique entries(70% is median frequency of non-nil and zero entries,out of which unique entries accounts for lesser than 1/3). This makes the feature space interesting because of its categorical values.

3.2 More Discovery

3.2.1 Anomalies

For real world dataset, it is beneficial to understand its boundary and edge cases. These may be due to mis-typed numbers, errors in measuring equipment, or they could be valid but extreme measurements. One way to support anomalies quantitatively is by looking at the statistics of a column using the `describe` method. The numbers in the `DAYS_BIRTH` column are negative because they are recorded relative to the current loan application. For a more intuitive look in years, we apply multiplication by -1 and divide by total days in a year. The distribution looks to be much more in line with what we would expect. In addition we also applied clustering algorithm on the these notable features to cluster

3.2.2 Correlations

One way to try and understand the data is by looking for correlations between the features and the target. Pearson correlation coefficient was calculated between every variable and the target using the `.corr` dataframe method.

Take a look in the correlation heatmap at some of more significant correlations: the `DAYS_BIRTH` is the most positive correlation. (second only `TARGET` because the correlation of a variable with itself is always 1) Looking at the documentation, `DAYS_BIRTH` is the age in days of the client at the time of the loan in negative days. The correlation is positive, but the value of this feature is actually negative, meaning that as the client

gets older, they are less likely to default on their loan (ie the target == 0). an age segment that is intensive in loan application and also high on defaults.

3.2.3 Effect of Age on Repayment

By itself, the distribution of age does not tell us much other than that there are no outliers as all the ages are reasonable. To visualize the effect of the age on the target, a kernel density estimation plot (KDE) colored by the value of the target is used. The plot shows the distribution of a single variable in a smoothed setting. histogram supported Gaussian kernels in seaborn `kdeplot`.

The target == 1 curve skews towards the younger end of the range. Although this is not a significant correlation (-0.07 correlation coefficient), this variable is likely going to be useful in a machine learning model because it does affect the target. Let's look at this relationship in another way: average failure to repay loans by age bracket.

To make this graph, first we `cut` the age category into bins of 5 years each. Then, for each bin, we calculate the average value of the target, which tells us the ratio of loans that were not repaid in each age category.

There is a clear trend: younger applicants are more likely to not repay the loan! The rate of failure to repay is above 10% for the youngest three age groups and beolow 5% for the oldest age group.

This is information that could be directly used by the bank: because younger clients are less likely to repay the loan, maybe they should be provided with more guidance or financial planning tips. This does not mean the bank should discriminate against younger clients, but it would be smart to take precautionary measures to help younger clients pay on time.

3.2.4 Exterior Sources and some other findings

The 3 variables with the strongest negative correlations with the target are `EXT_SOURCE_1`, `EXT_SOURCE_2`, and `EXT_SOURCE_3`. According to the documentation, these features represent a "normalized score from external data source". I'm not sure what this exactly means, but it may be a cumulative sort of credit rating made using numerous sources of data.

Let's take a look at these variables.

First, we can show the correlations of the `EXT_SOURCE` features with the target and with each other. In the plot, the red indicates loans that were not repaid and the blue are loans that are paid. We can see the different relationships within the data. There does appear to be a moderate positive linear relationship between the `EXT_SOURCE_1` and the `DAYS_BIRTH` (or equivalently `YEARS_BIRTH`), indicating that this feature may take into account the age of the client.

3.3 Algorithm

3.3.1 Online Logistic Regression on EXIT&Fin-knowlegde

We will first preprocess the data by collecting non-zero values and normalize the range of the features. After it is normalized, we will specifically elevate and cross-product `EXT_SOURCE` features for logistic regression.

$$P(D \mid z) = \prod_{i=1}^{n} h(y=1 \mid z)^{y_i} h(y=0 \mid z)^{1-y_i} = \prod_{i=1}^{n} \frac{1}{1+e^{-z}}^{y_i} (\frac{e^{-z}}{1+e^{-z}})^{(1-y_i)}$$

$$z(X,Y) = \omega \cdot \{x_1 + x_2 + x_3 + x_1^2 + x_2^2 + x_3^2 + x_1 x_2 x_3\}$$

The confidence here is that numerical EXT-SOURCE value taken by inter product of scaled features will be squashed in logistic regression's sigmoid function shown above. For example, we could create new features `EXT_SOURCE_1^2` (powers) and `EXT_SOURCE_2^2` and also variables such as `EXT_SOURCE_1` x `EXT_SOURCE_2`(interaction terms), `EXT_SOURCE_1` x `EXT_SOURCE_2^2`, `EXT_SOURCE_1^2` x `EXT_SOURCE_2^2`. It is our assumption that, while our variables by themselves may not have a strong influence on the target, combining them together into a single interaction variable might show a relationship with the target.

It is from the view of our feature-target distribution, we could make a couple features that attempt to capture what we think may be important for telling whether a client will default on a loan. In a kaggle discussion interesting proposal was made to investigate following domain knowledge. These features were believed to be all-rounded to represent a person's financial status.

- CREDIT_INCOME_PERCENT=AMT_CREDIT/AMT_INCOME_TOTAL: the percentage of the credit amount relative to a client's income
- ANNUITY_INCOME_PERCENT=AMT_ANNUITY/AMT_INCOME_TOTAL: the percentage of loan amount relative to a client's income.
- YEARS_EMPLOYED_PERCENT=abs(DAYS_EMPLOYED/DAYS_BIRTH): the percentage of the days employed relative to the client's age

To try and beat the performance of our baseline, we also could make choice of applying complex models such as Random Forest on the same training data to see how that affects performance. The Random Forest is a much more powerful model especially when we use hundreds of trees. We will use between [10:20] trees in the random forest.

4 Experiment& Discussion

4.1 Feature Engineering and Sampling

The adopted approach of feature engineering mainly consists of two part: choosing the feature that impacts the target variable most(this part we also focused on cleansing input entries that has weak information as zero and nil values) and constructing new features from existing ones. The first step will greatly reduce the dimension of feature space, the second one will prepare a dataset that caters the assumption of our hypothesis. The following adopted techniques to perform feature engineering are introduced in later paragraphs.

- Polynomial features
- Domain knowledge features

For training sampling method, we used one fold of size 10000 to train while hold out rest, this method is mainly considering IO threshold. In stochastic approach one fold is further sliced into mini-batch that is used to train the model. To ensure the convergence of stochastic process multiple iteration is adopted. Detailed setup parameters is concluded in Table1.

Addition adjustment made for categorical values used encoding policy that for any categorical variable (dtype == object) with 2 unique categories, we will use label encoding, and for any categorical variable with more than 2 unique categories, we will use one-hot encoding. This policy will help preserve the original information of feature.

4.2 Model Interpretation: Feature Importances

As a simple method to see which variables are the most relevant, we can look at the feature importances in the formation of decision tree. Given the correlations we saw in the exploratory data analysis, we should expect that the most important features are the EXT_SOURCE and the financial group of information DAYS_BIRTH. This importance is further supported by info-gain ranker we applied to select top 20 features for training adaboost. We may use these feature importances as a method of dimensionality reduction in future work. As expected, the most important features are those dealing with EXT_SOURCE and DAYS_BIRTH. We see that there are only a handful of features with a significant importance to the model, which suggests we may be able to drop many of the features without a decrease in performance, Feature importances are not the most sophisticated method to interpret a model or perform dimensionality reduction, but they let us start to understand what factors our model takes into account when it makes in predictions.

4.3 Evaluation Metric

On this particular problem, the goal is to reduce false alarm out of a unbalanced dataset. Hence accuracy will not better describe the performance as even in random classifier baseline the accuracy would be high. Instead ROC and AUC stress on both false and true positive case, it adopted to evaluate model performance. We would use Receiver

Operating Characteristic curve to visualize the actual performance.On the curve x-axis represent fpr that an ideal model for all positives tpr far greater than fpr so that there is an concavity of this cave above the base diagonal line and AUC measures this concavity. In short, larger the AUC the better the model performance.

4.4 Experimental Evaluation:

From the result perspective. At beginning, the model of logistic regression performed on EXT-SOURCE data had quiet worse performance only achieved 0.50 to 0.49 AUC.

On the other side, at first multinomial Naive Bayesian model did quiet poorly on all categorical feature. On one hand we saw those features falls behind in its ranking of information gain i.e the ability to distinguish target labels which probably explained why in general the performance of model is bad. ROC of 0.45-0.49 is worse than random -classifier but not worse enough to make reversing model prediction parameter a good strategy to turn the performance upside. For fixed output classifier and Random-classifier both had around 0.5 since their bias is on the target distribution representing future result.

And after we applied the feature space transformation and controlled batch size to 100 per batch since after feature engineering the sample size, AUC for improved notably compared to two previous models. SGD Logistic regression reached AUC on average of 0.750 on polynomial features while on domain knowledge dataset it reached 0.547. SGD NaiveBayes also improved drastically for polynomial features which increased to 25% on newly engineered polynomial features. From the ROC we could clearly see the concavity is towards point (0,1).So after feature engineering it generally yields higher f-score. This would be beneficial for HomeCredit loan dep. since their prediction of default behavior under these models would be more accurate and less vague. Also both feature engineering improved compared to baseline The increase on finance domain knowledge features is slight compared to formal. One assumption is that it had limited number of features. And this is good tips for HomeCredit and other banking institutions who also use expert-system to make loan decisions, a limited input could restrict the precision quality even the input is after thoughtful selection.

One final aspect is whether the model generate stable outcome. From Table2 we could compare the stand deviation of train test performance in AUC. Logistic regression in general have higher variance than Multinomial-NaiveBayes. This is quiet reasonable since the formal has less complex model than the latter. The latter controls hypothesis space with the prior.

Model	Method	Data sampling	Parameter	Performance
SGD Logistic-Regression	Online	1batch=500 ,shuffled: 10000	max-iter:50	0.49-0.61
SGD Multinomial-Naive	Online	1batch=100 ,shuffled: 10000	max-iter:20	0.45-0.49
Fixed-output	Full pass	Size=10000	NA	0.49
Random-Classifier	Full pass	size=10000	thres=P-target	0.51
Boost:WeakClassifier	Each Gm at a time	shuffled:10000	MaxIter = 10, lr = 1e-4	0.61
Improved Logistic regression	Online	1batch=500 ,shuffled: 10000	max-iter:20	EXT:0.72-0.75 | FIN:0.52
Improved NaiveBayes	Online	1batch=500 ,shuffled: 10000	max-iter:20	EXT: >0.70 | FIN:0.51

Table1.ImplementationSetting&Result.

5 Conclusion

This project deals with a interesting problem with several different perspectives. In previous sections we have seen how among academics and professionals improved their techniques to deal with this of problems along the course. Now complex models and comprehensive recipes have been adopted to do binary classification. In EDA part, we discovered some strong correlation of defaults features and patterns which could suggests HomeCredit for better granting process. Through training new models and comparing with baseline models we found sampling amount affected model's performance so does the coverage of input features, the model only gets as good as its feed allows it to be. In general, feature engineering improves performance as well as contribute to the understanding of the model. While models have inherent bias,the recipe of controlling input and error feed-back mechanism would still control how well the model generalize its performance(as shown in Adaboost). In further attempt, we could combine boosting method with feature engineering by aggregating more powerful classifiers. In this way we hope to achieve even better result.

Model	Train(Avg)	Test(Avg)	VAR
LR	0.5190	0.4920	10.7%
NB	0.4929	0.4890	0.40%
AdaBoost	0.6315	0.6191	1.24%
LR2	0.7496	0.7474	0.22
NB2	0.7437	0.7404	0.03%
LR3	0.5465	0.5159	3.1%
NB3	0.5223	0.5107	1.16%

Table2.Model Stability.

6 Reference

1 Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning. Information Science and Statistics. Springer-Verlag, Berlin, Heidelberg.

2 Hand D. J. , & Henley, W. E. 1997. Statistical classification models in consumer credit scoring: A review. Journal of the Royal Statistical Society: Series A (General), 160 (3), 523–541 .

3 Aida Krichene, 2017. Using a naive Bayesian classifier methodology for loan risk assessment: Evidence from a Tunisian commercial bank, Journal of Economics, Finance and Administrative Science, Vol. 22 Issue: 42, 3-24.

4 Baesens B., Van Gestel, T.,Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. 2003. Benchmarking state-of-the-art classification algorithms for credit scoring, Journal of the Operational Research Society, 54 (6), 627-635.

5 I.Brown, C. Mues. 2012. An experimental comparison of classification algorithms for imbalanced credit scoring data sets. Journal of Expert Systems with Applications, 39, 3446–3453.

6 S. Maldonado, J. Pérez, C. Bravo, 2017. Cost-based feature selection for Support Vector Machines: An application in credit scoring, European Journal of Operational Research, 261(2),656-665.

7 Y.Xiao, 2017,Machine learning application in online lending risk prediction. arXiv:1707.04831v1 [q-fin.RM].

8 Huang,Y.M.,Hung,C.M.,Jiau,H.C.,2006, Evaluation of neural networks and data minning methods on a credit assessment task for class imbalance problem, Nonlinear Analysis:Real World Applications 7(4) 720 – 747.

9 Khandani, Amir E.,Kim, Adlar J., Lo, Andrew W., 2010. Consumer credit-risk models via machine-learning algorithms, Journal of Banking and Finance, Elsevier, vol. 34(11), 2767-2787.

10 Peter M. A., Dominique G., Bertrand H.2018. Credit Risk Analysis Using Machine and Deep Learning Models, Journal of MDPI :Risks 2018, 6, 38.

Appendix :Support illustration

-

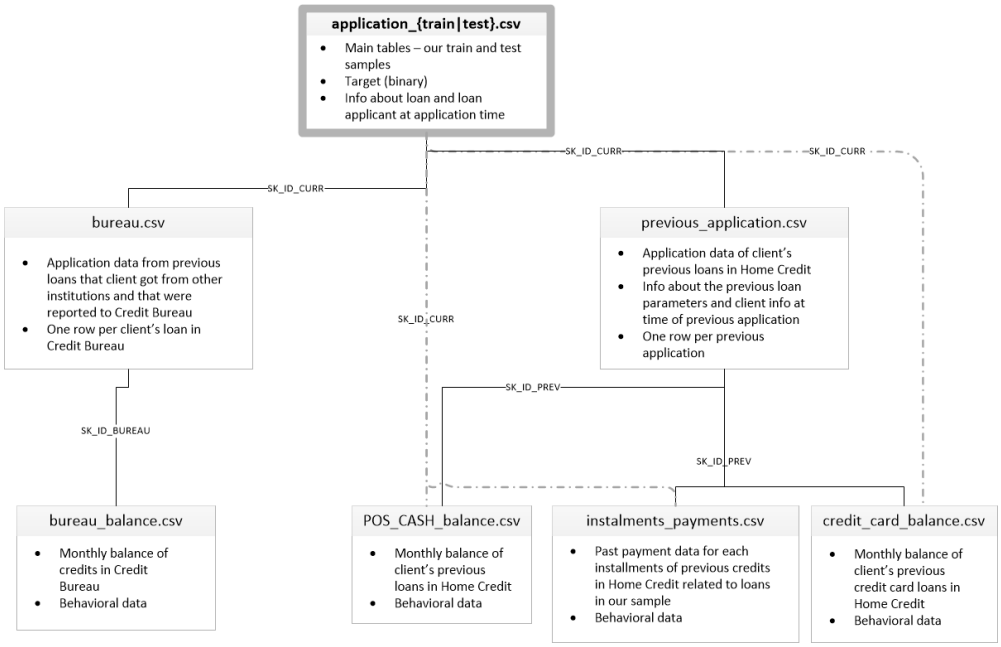

fig.1 Relation DOM

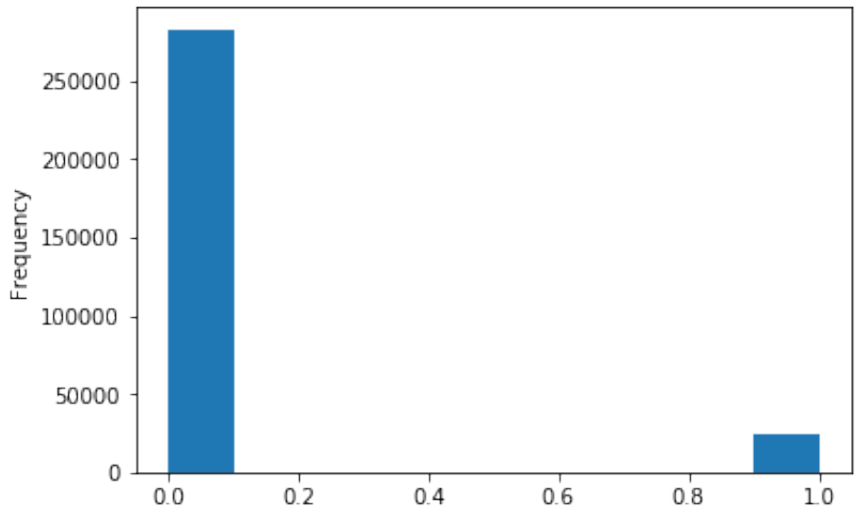

fig.2 Target distribution

	occupation
0	programmer
1	data scientist
2	engineer
3	manager
4	ceo

Label Encoding

	occupation
0	4
1	1
2	2
3	3
4	0

fig.3 label-encoding

fig.4 one-hot-encoding

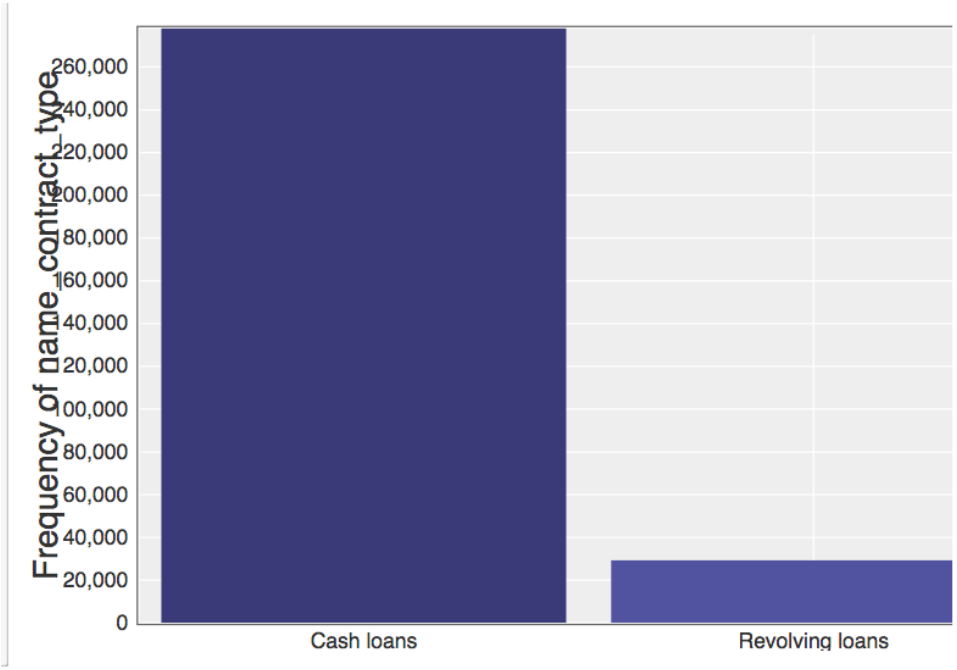

fig.5 loan-distribution

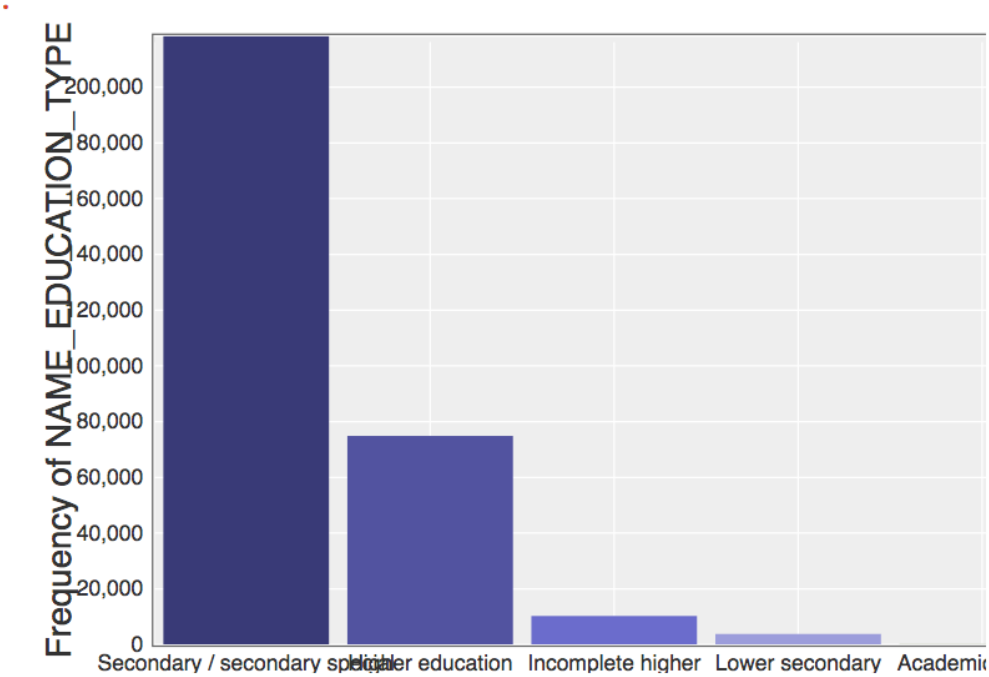

fig.6 education-distribution

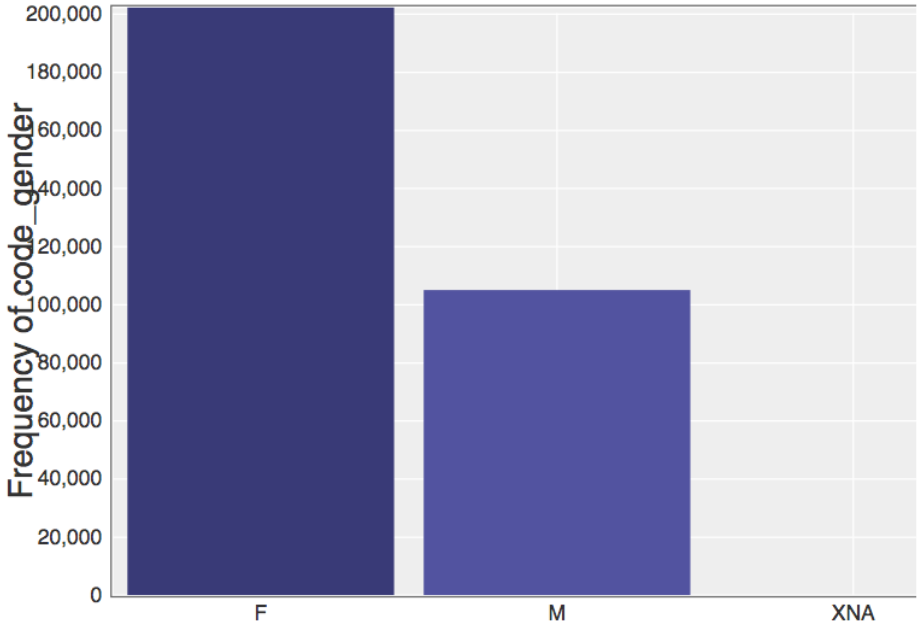

fig.7 Gender-distribution

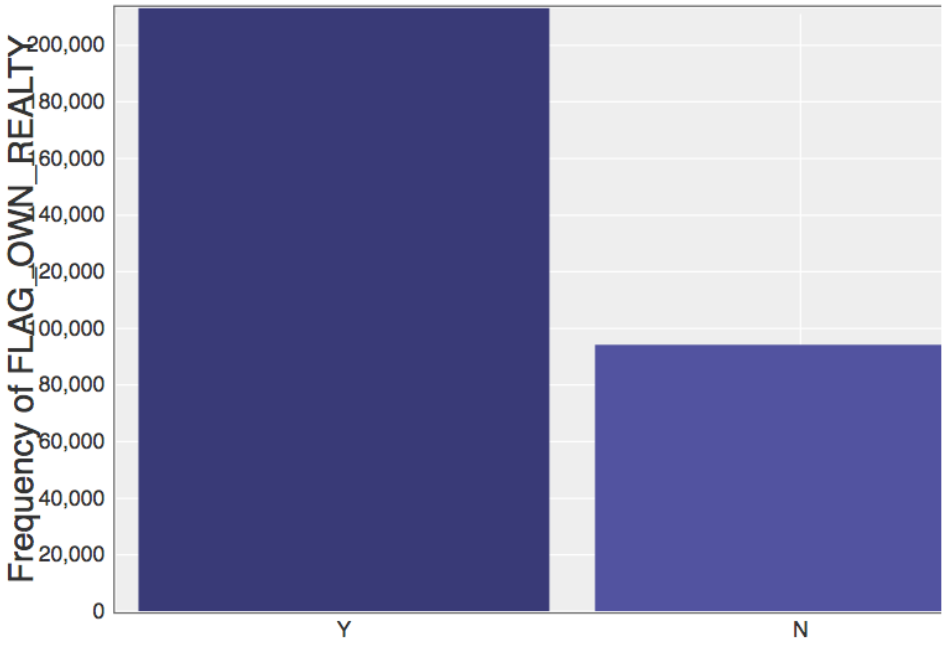

fig.8 Property-distribution

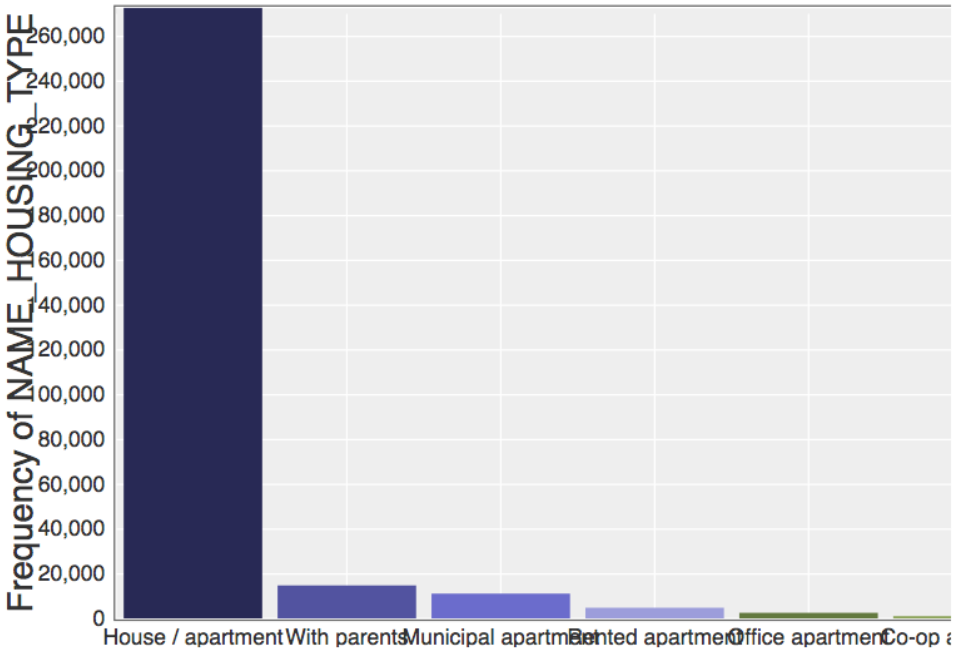

fig.9 household-types

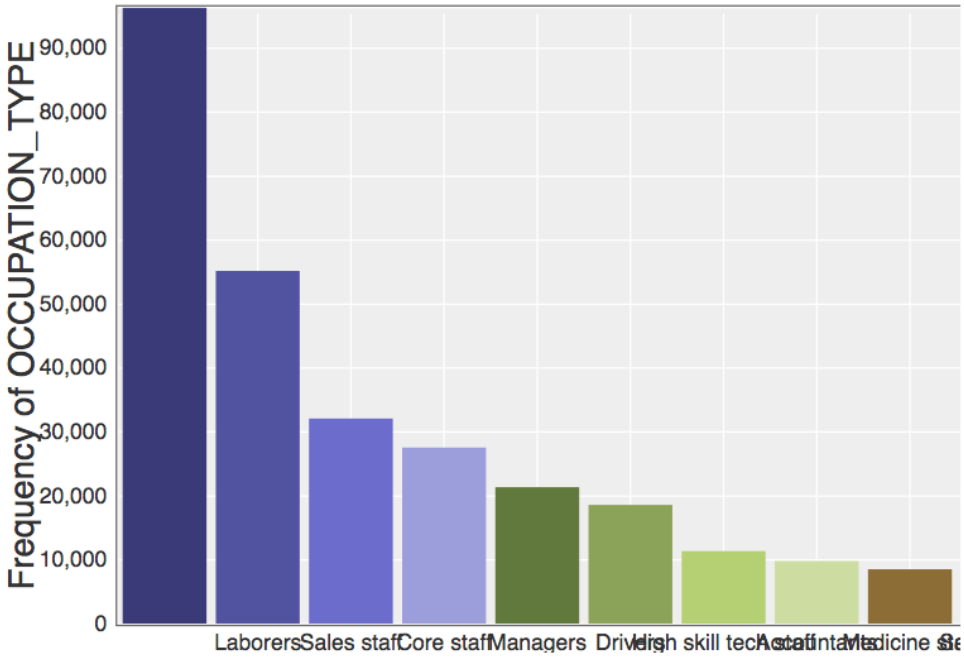

fig.10 Occupation-types

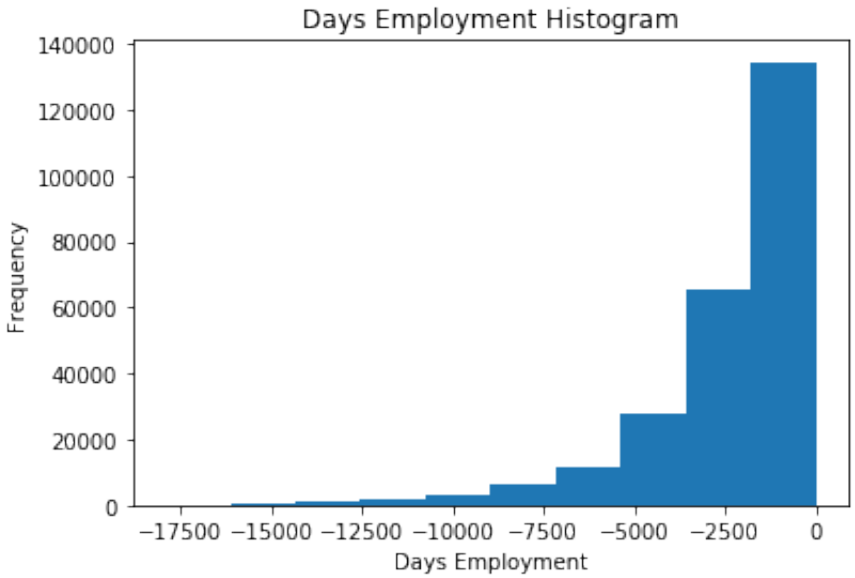

fig.11 hist-days-employed

fig.12 hist-age-client

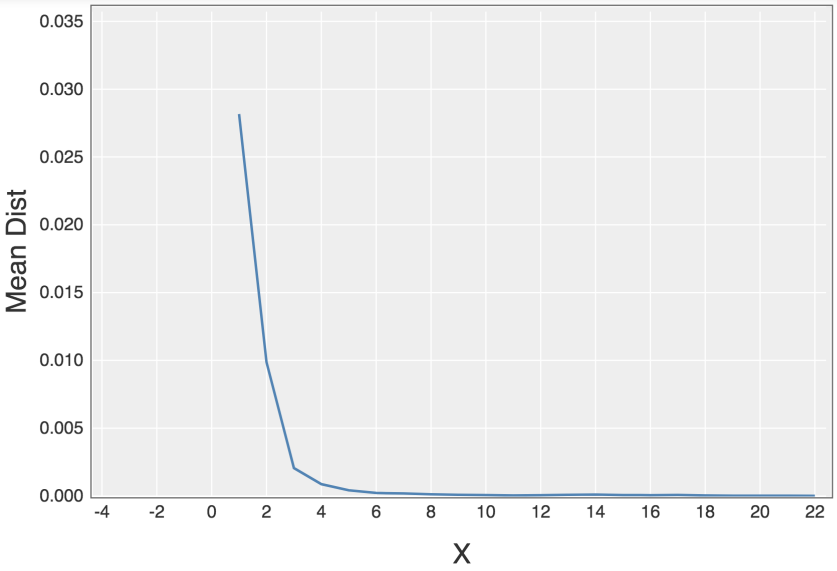

fig.13 Separate age by 5 clusters

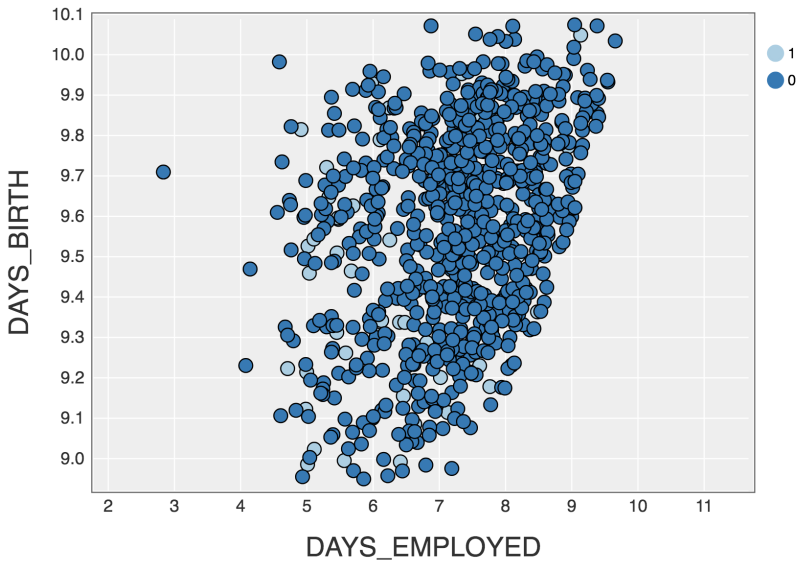

fig.14 Use age group to find default

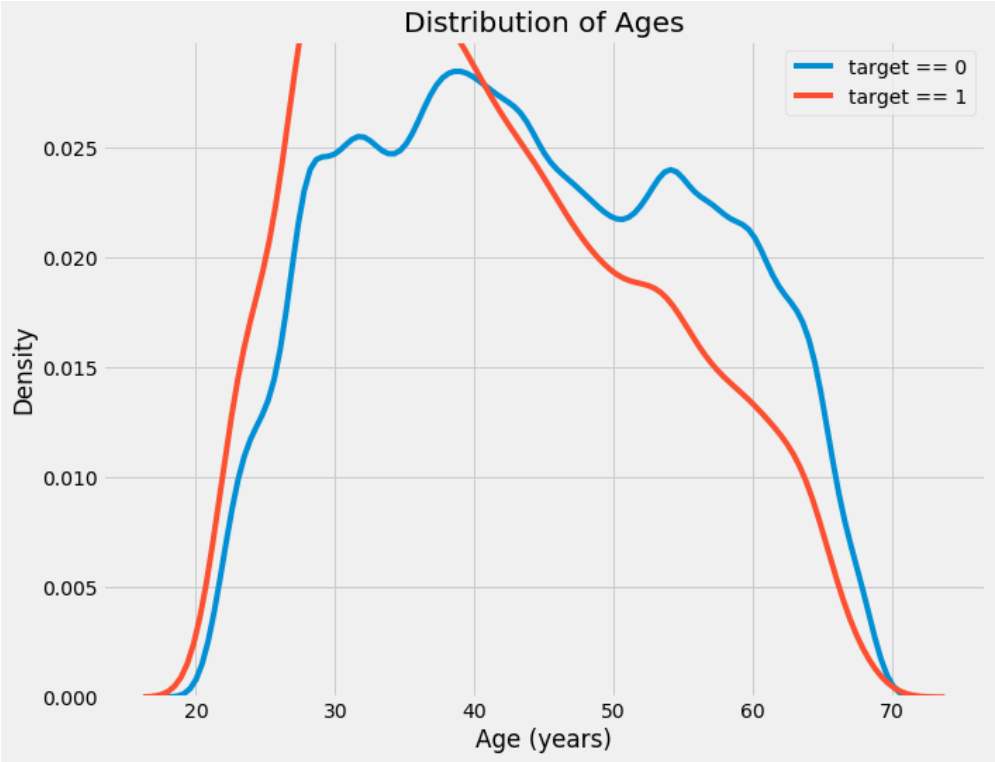

fig.15 dist-target-vs-age

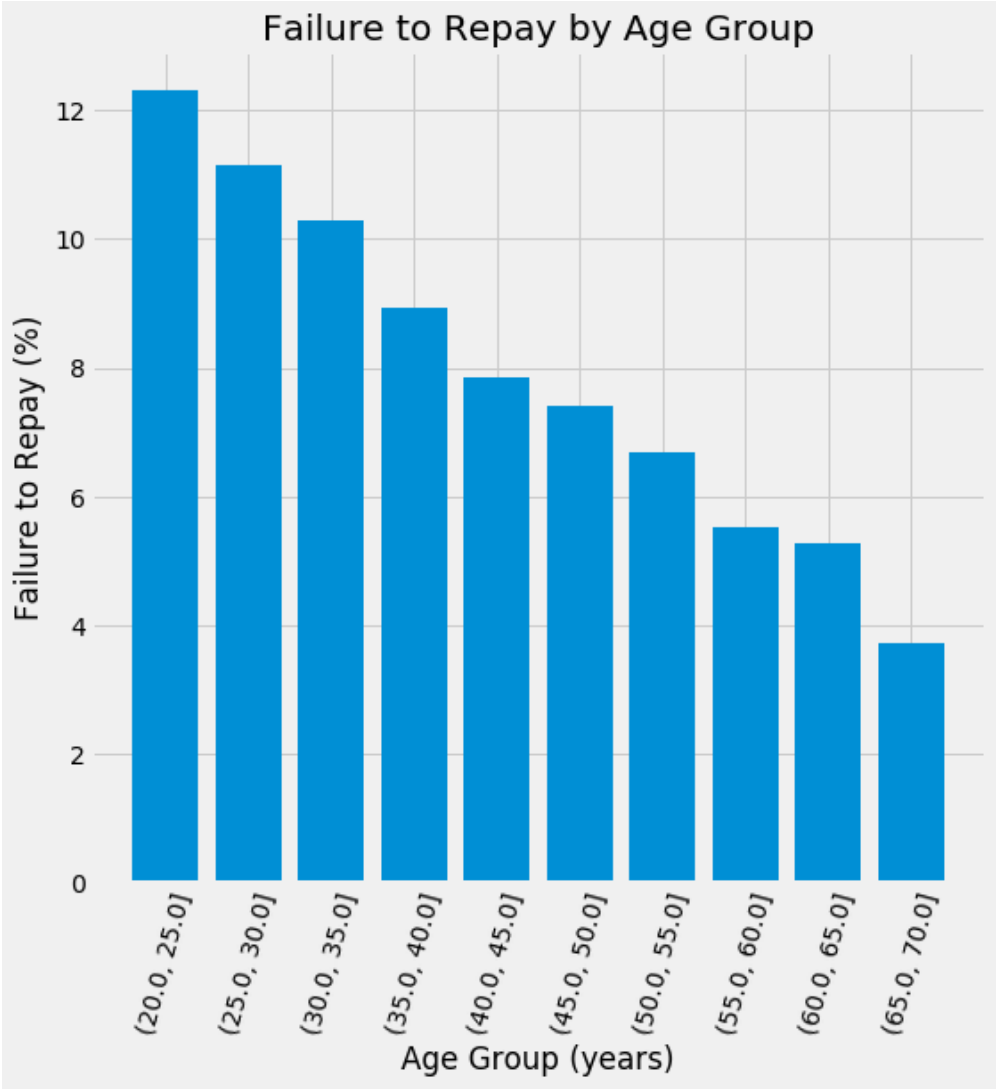

fig.16 RepayFailuerRate-Age

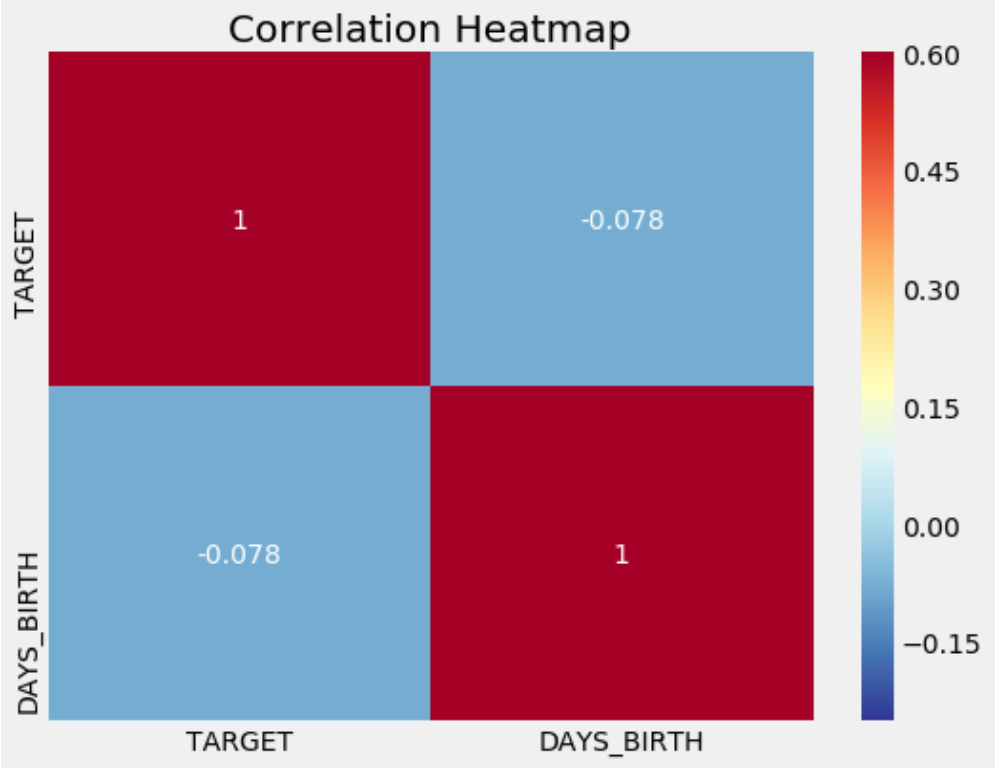

fig.17 Correlation-HeatMap

	Feature Name	Feature Type	Information Gain	Frequency
0	EXT_SOURCE_3	NUMERIC	0.015993	0.801747
1	EXT_SOURCE_1	NUMERIC	0.012197	0.436189
2	EXT_SOURCE_2	NUMERIC	0.011311	0.997854
3	DAYS_BIRTH	NUMERIC	0.003222	1.0
4	NAME_EDUCATION_TYPE	TEXT	0.00251	1.0
5	NAME_INCOME_TYPE	TEXT	0.002407	1.0
6	DAYS_LAST_PHONE_CHANGE	NUMERIC	0.002159	0.999997
7	CODE_GENDER	TEXT	0.002093	1.0
8	AMT_GOODS_PRICE	NUMERIC	0.002083	0.999096
9	OWN_CAR_AGE	NUMERIC	0.002063	0.340092
10	DAYS_EMPLOYED	NUMERIC	0.001995	1.0
11	REG_CITY_NOT_WORK_CITY	INTEGER	0.001769	1.0
12	ORGANIZATION_TYPE	TEXT	0.001672	1.0
13	FLAG_EMP_PHONE	INTEGER	0.001672	1.0
14	REGION_RATING_CLIENT_W_CITY	INTEGER	0.001637	1.0
15	DAYS_ID_PUBLISH	NUMERIC	0.001609	1.0
16	REGION_RATING_CLIENT	INTEGER	0.001535	1.0
17	FLAG_DOCUMENT_3	NUMERIC	0.001489	1.0
18	REG_CITY_NOT_LIVE_CITY	INTEGER	0.001259	1.0
19	REGION_POPULATION_RELATIVE	NUMERIC	0.00125	1.0
20	AMT_CREDIT	NUMERIC	0.001192	1.0
21	DAYS_REGISTRATION	NUMERIC	0.001102	1.0
22	FLOORSMAX_AVG	NUMERIC	0.001062	0.502392
23	FLOORSMAX_MEDI	NUMERIC	0.001044	0.502392
24	ELEVATORS_AVG	NUMERIC	0.001024	0.46704
25	ELEVATORS_MEDI	NUMERIC	0.001012	0.46704
26	OCCUPATION_TYPE	TEXT	0.000989	0.686545
27	FLOORSMAX_MODE	NUMERIC	0.000985	0.502392
28	ELEVATORS_MODE	NUMERIC	0.000935	0.46704
29	TOTALAREA_MODE	NUMERIC	0.000924	0.517315
...	...	...	...	...
120	FLAG_DOCUMENT_20	NUMERIC	0.0	1.0

fig.18 Correlation-HeatMap

```
0.546595802127659
```

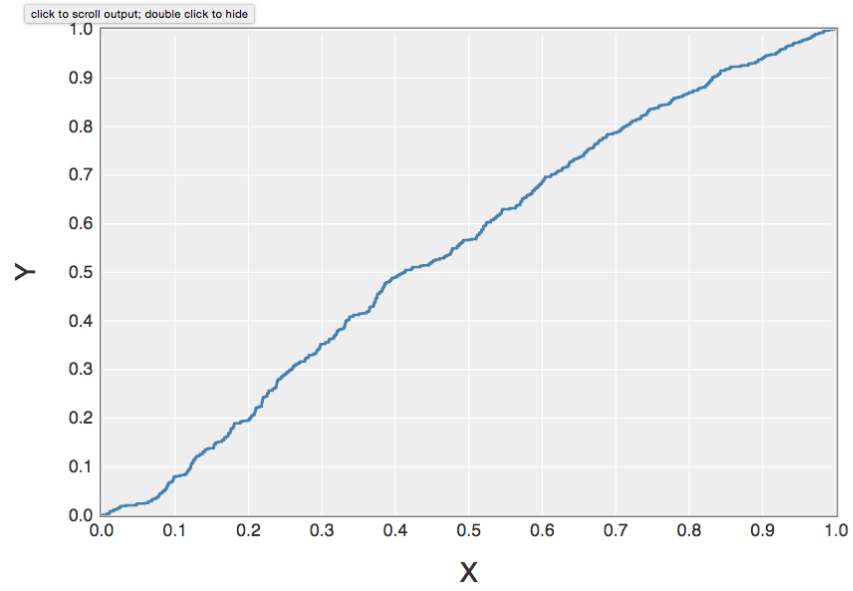

fig.19 ROC-curve-LR

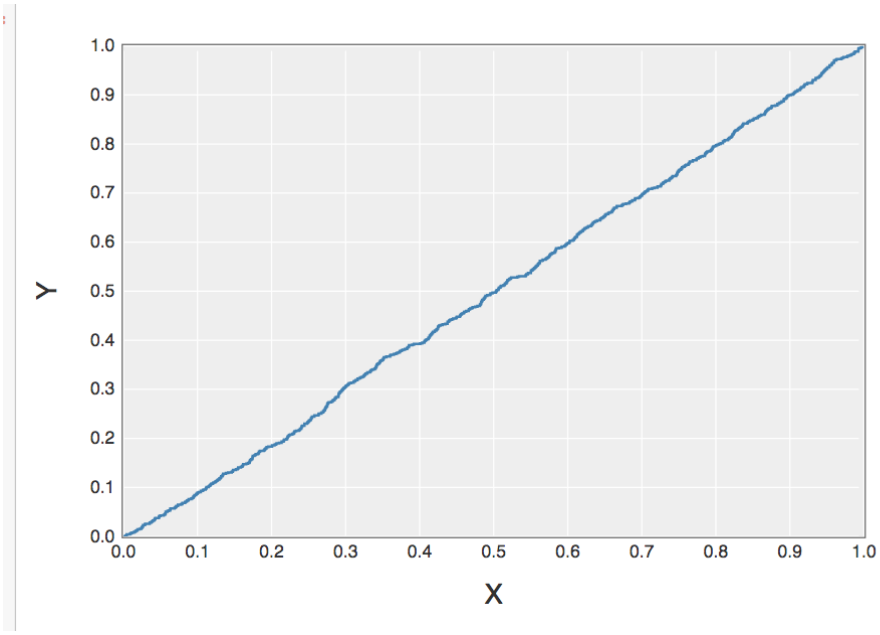

fig.20 ROC-curve-fixed

```
0.7234778360509174
Out[202]:
```

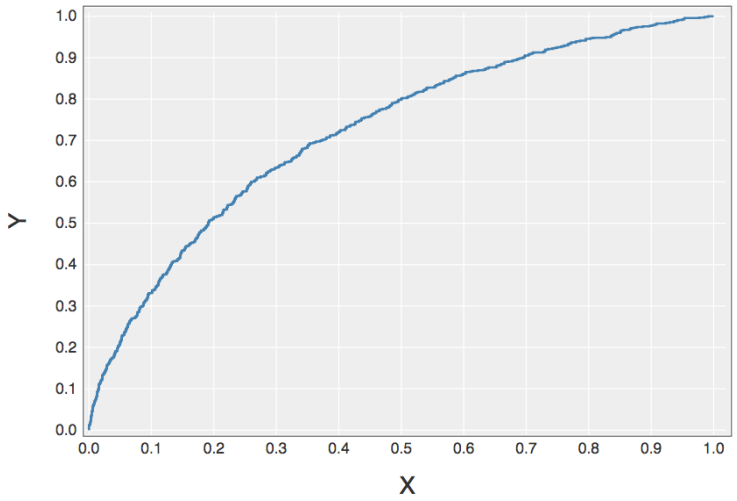

fig.21 ROC-curve-Improved-lr-poly

```
10000
[-0.6952077024314699, 0.0]
0.7239372683201564
Out[206]:
```

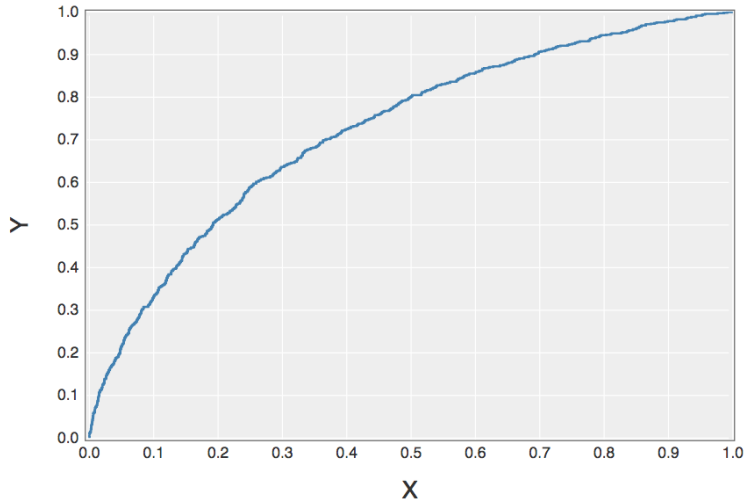

fig.22 ROC-curve-Improved-Naive-poly

```
0.7497689060136771
```

Out[213]:

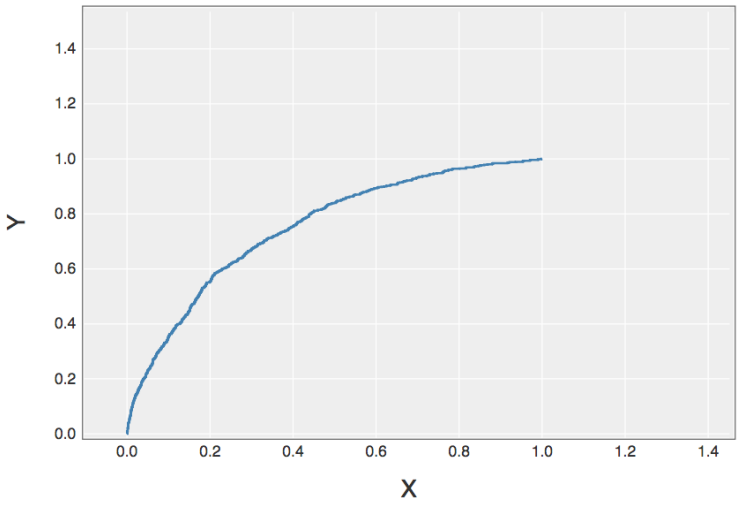

fig.23 ROC-curve-Improved-Naive-poly

```
In [461]: 1 df.plot type: :line, x: [:fpr1, :fpr2, :fpr3], y: [:tpr1, :tpr2, :tpr3]
```

Out[461]:

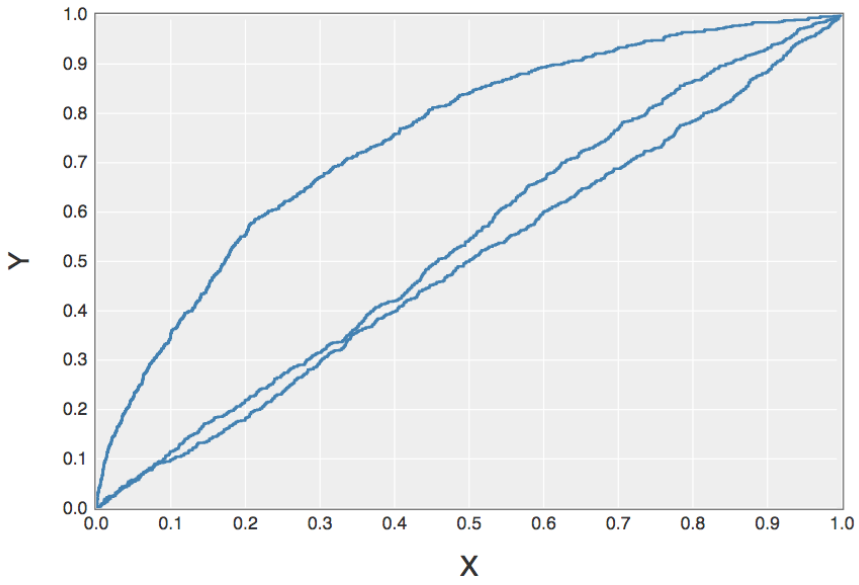

fig.24 ROC-curve