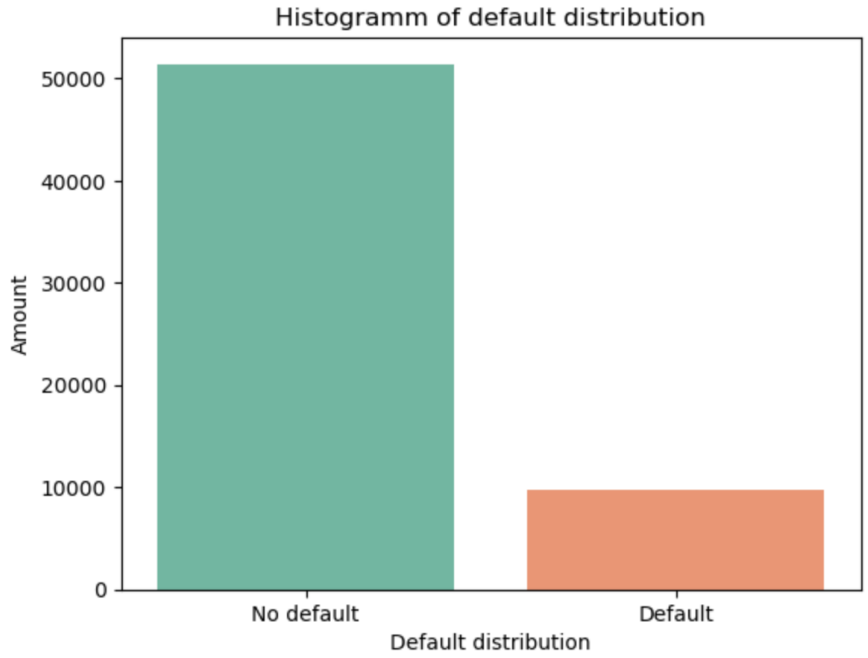
Practical Module on Credit Risk Assessment

Zhurba Anna

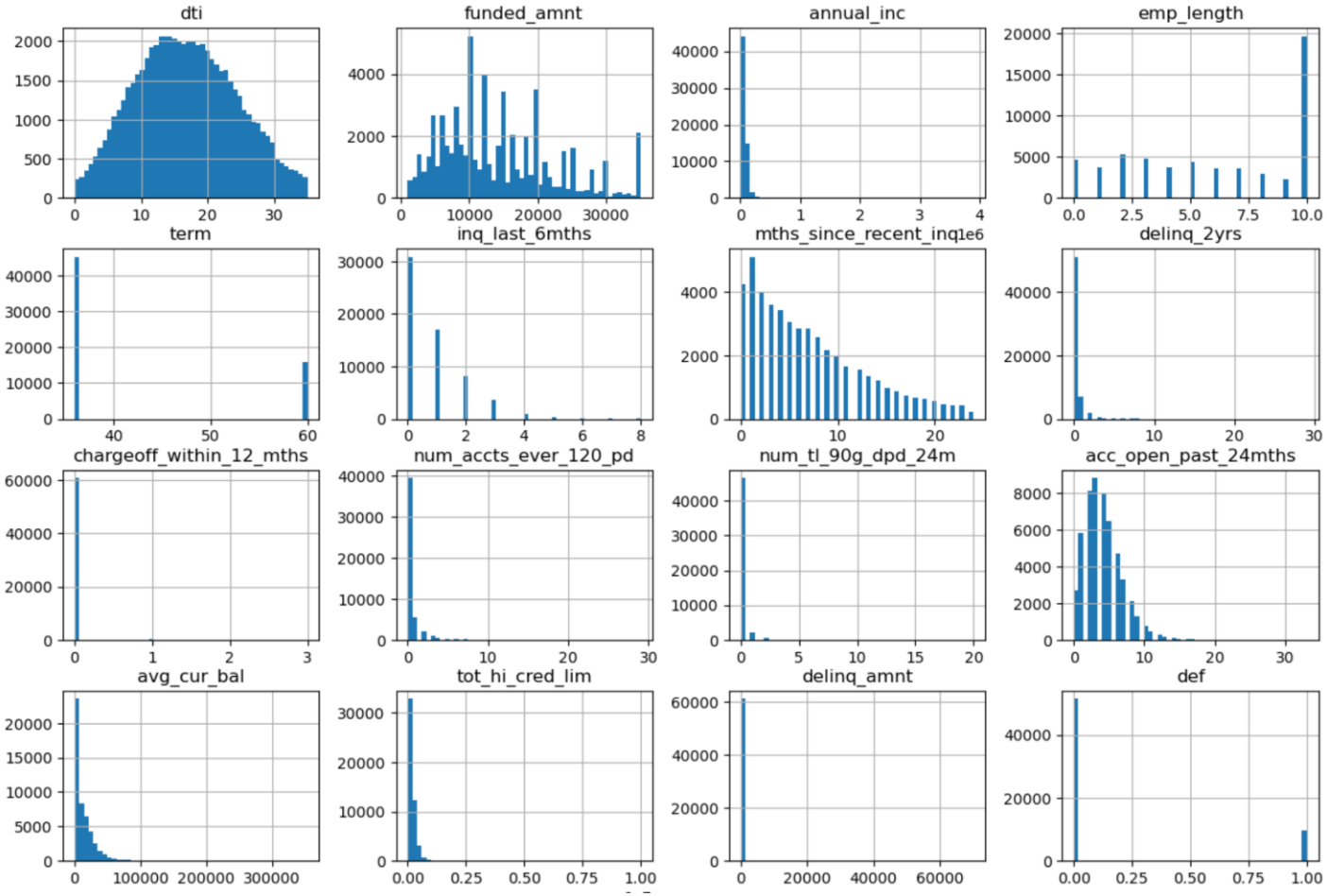
EDA

Unbalanced sample:

	def	count	percent	cumulative_count	cumulative_percent
0	0	51,407	84.04%	51,407	84.04%
1	1	9,762	15.96%	61,169	100.00%



The variables are not normally distributed:



Delete useless features

Deleted:

‘issue_d ‘ – date

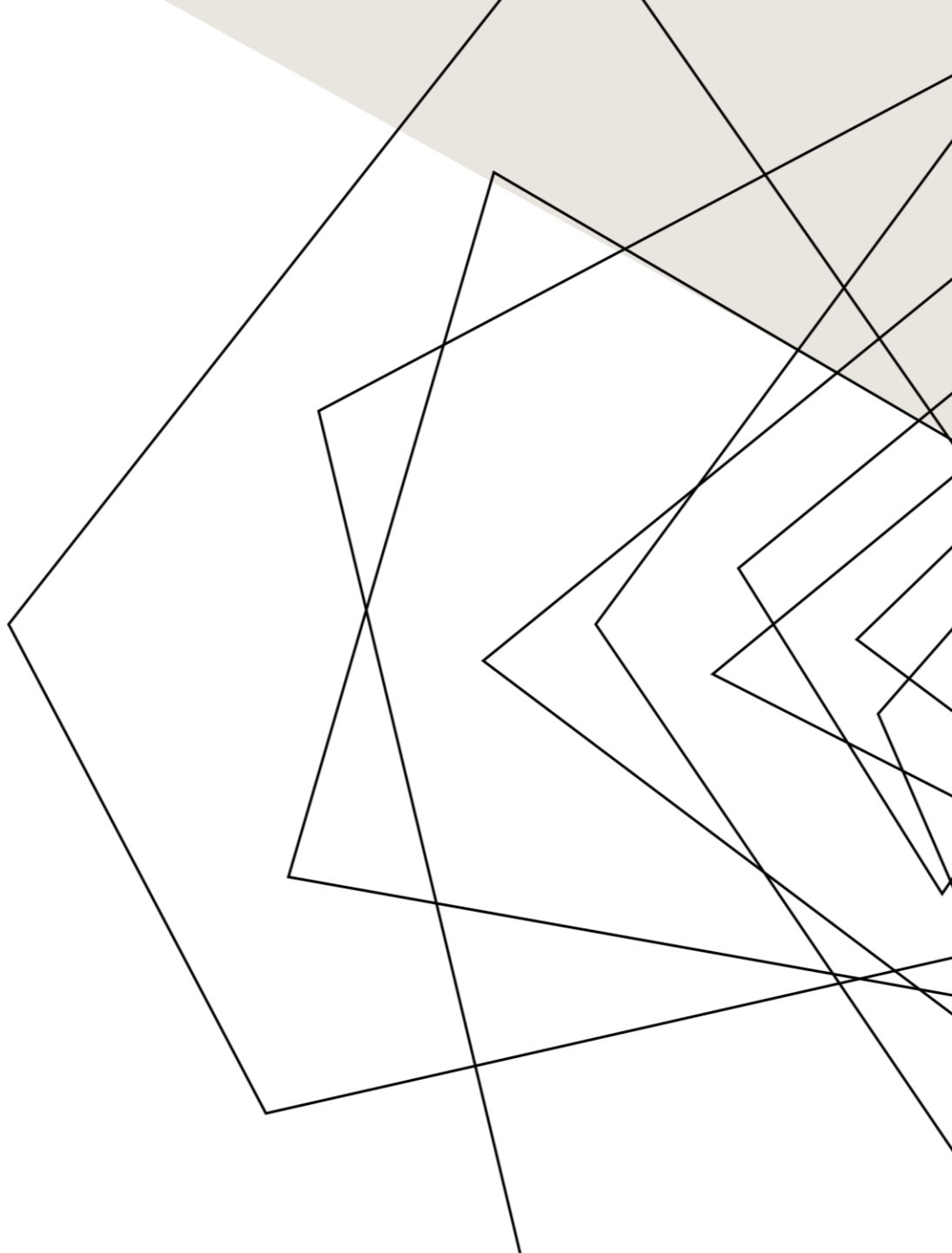
‘addr_state’ – USA state

I delete it as it's not correct use encoder for date and location (it doesn't take into account the distance between states, for dummies –too many unique values (47))

Dummies without 1 category:

- Purpose
- Sub_grade
- Home_ownership

The sample was divided into train (80%) and test (20%)



Feature engineering

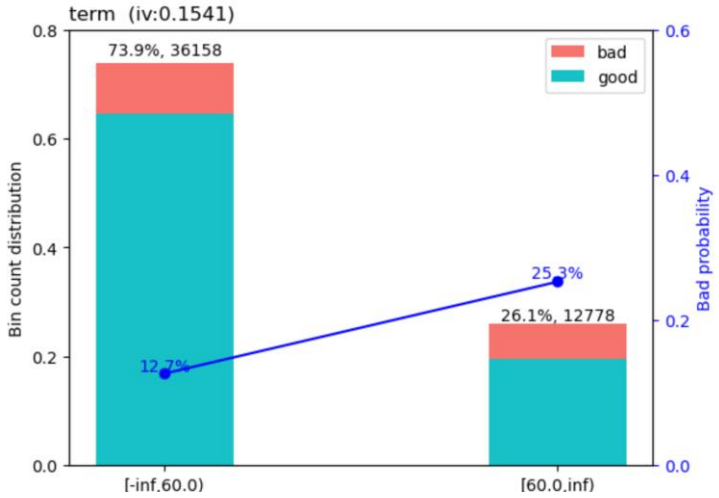
Removing:
Variables that contain **less than 80%** of the values of the entire dataframe were deleted : ‘mths_since_recent_inq

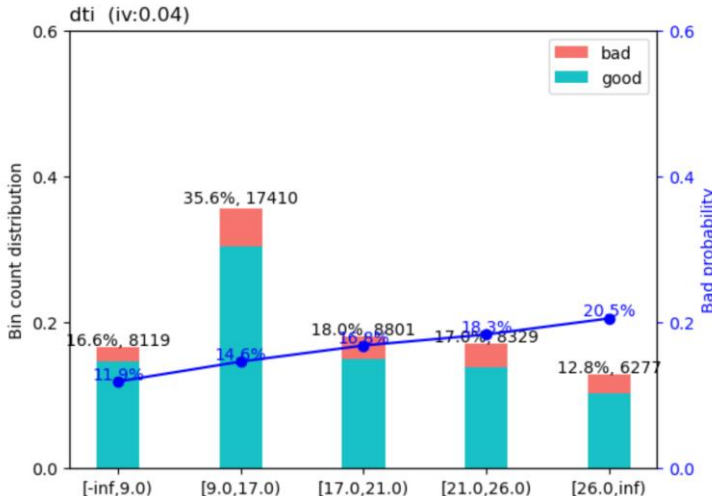
Missing input:
Let's fill in the missing values with the most frequently occurring values of the variable, i.e. the **mode**

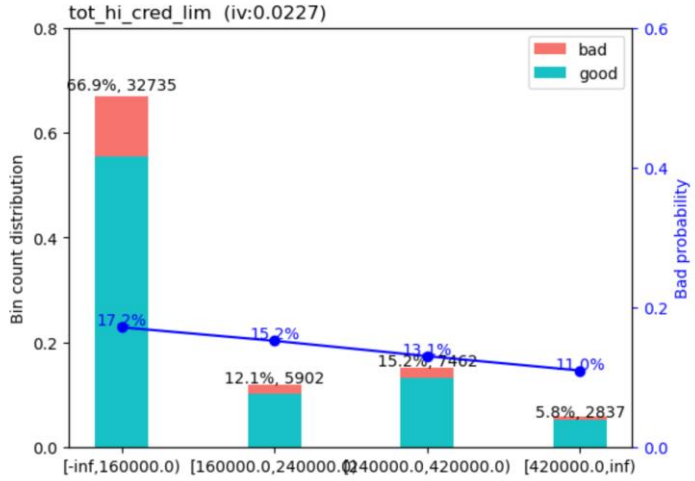
	missing	total	percent
avg_cur_bal	9549	48936	19.5132
num_tl_90g_dpd_24m	9546	48936	19.5071
num_accts_ever_120_pd	9546	48936	19.5071
tot_hi_cred_lim	9546	48936	19.5071
acc_open_past_24mths	6289	48936	12.8515
emp_length	2177	48936	4.4487

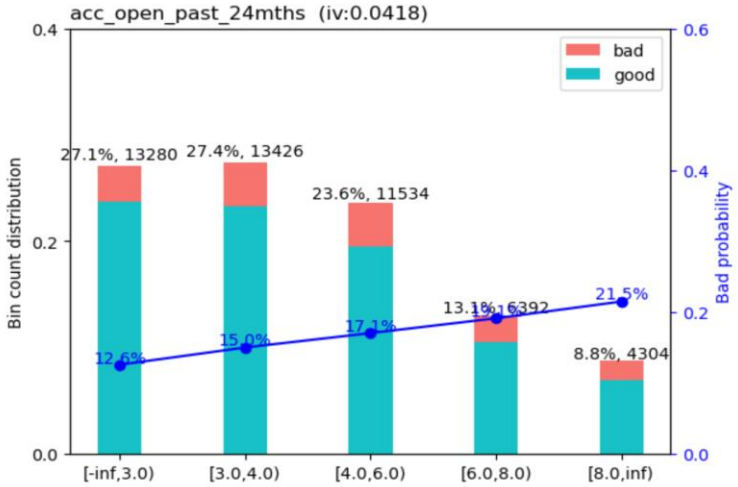
Feature transformation

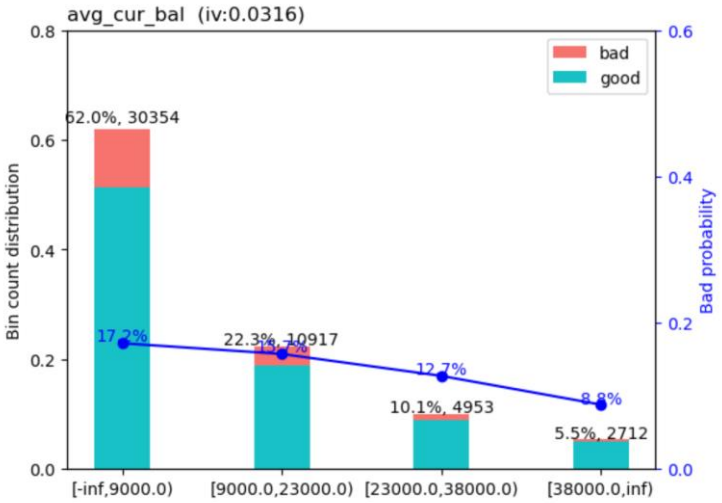
WOE binning for train and test, results for train:

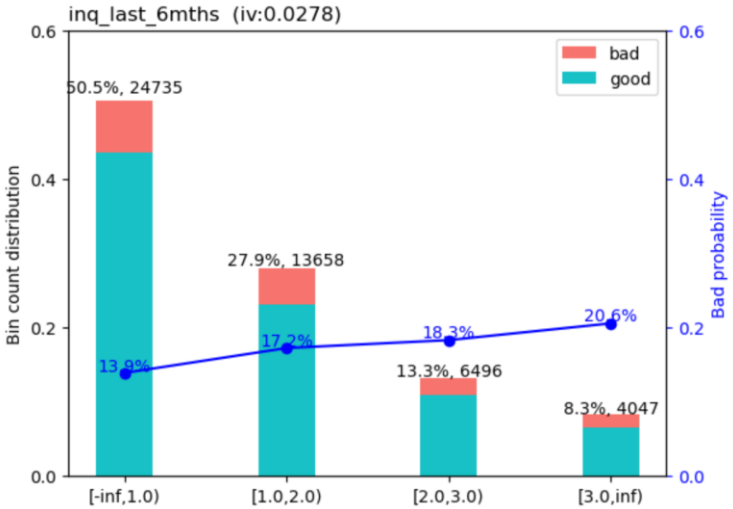












Feature selection– information value

	variable	info_value
26	term_woe	0.1541
9	acc_open_past_24mths_woe	0.0418
18	dti_woe	0.0400
29	annual_inc_woe	0.0379
49	avg_cur_bal_woe	0.0316
47	inq_last_6mths_woe	0.0278
39	sub_grade_B2_woe	0.0227
48	tot_hi_cred_lim_woe	0.0227
40	sub_grade_B1_woe	0.0217

To select variables based on their informativeness, I used the approach of IV's (information values)

The table shows the selected variables for which the IV is **higher than 0.02**

The rule of thumb:

Information Value	Predictive Power
< 0.02	**Useless** predictor
0.02 to 0.1	**Weak** predictor
0.1 to 0.3	**Medium** predictor
> 0.3	**Strong** predictor

Feature selection: correlation matrix
Correlated features removal

	acc_open_past_24mths_woe	dti_woe	term_woe	annual_inc_woe	sub_grade_B2_woe	sub_grade_B1_woe	inq_last_6mths_woe	tot_hi_cred_lim_woe	avg_cur_bal_woe	def
acc_open_past_24mths_woe	1.000000	0.164029	0.026503	-0.068314	0.031317	0.031477	0.199292	-0.114332	0.028783	0.075192
dti_woe	0.164029	1.000000	0.077657	0.187833	0.024374	0.019273	0.009838	-0.026382	0.088838	0.072804
term_woe	0.026503	0.077657	1.000000	-0.114800	0.108022	0.108149	0.033660	-0.107212	-0.072000	0.151508
annual_inc_woe	-0.068314	0.187833	-0.114800	1.000000	0.003120	0.003029	-0.075834	0.441157	0.344484	0.069581
sub_grade_B2_woe	0.031317	0.024374	0.108022	0.003120	1.000000	-0.060477	0.056637	0.014553	0.012454	0.049436
sub_grade_B1_woe	0.031477	0.019273	0.108149	0.003029	-0.060477	1.000000	0.051458	0.027474	0.021500	0.047907
inq_last_6mths_woe	0.199292	0.009838	0.033660	-0.075834	0.056637	0.051458	1.000000	-0.061873	-0.021545	0.061629
tot_hi_cred_lim_woe	-0.114332	-0.026382	-0.107212	0.441157	0.014553	0.027474	-0.061873	1.000000	0.769782	0.052985
avg_cur_bal_woe	0.028783	0.088838	-0.072000	0.344484	0.012454	0.021500	-0.021545	0.769782	1.000000	0.059934
def	0.075192	0.072804	0.151508	0.069581	0.049436	0.047907	0.061629	0.052985	0.059934	1.000000

According to correlation matrix I prefer to delete 'tot_hi_cred_lim_woe‘, because it is less correlated with the target ‘def’ than the ‘Avg_cur_bal_woe’.

Selected features

	acc_open_past_24mths_woe	dti_woe	term_woe	annual_inc_woe	sub_grade_B2_woe	sub_grade_B1_woe	inq_last_6mths_woe	avg_cur_bal_woe
0	-0.0710	-0.3409	-0.2700	0.0783	0.0341	0.0306	-0.1635	0.0900
2	-0.0710	0.0633	-0.2700	-0.2124	0.0341	0.0306	0.1653	0.0900
4	-0.0710	-0.3409	-0.2700	0.0783	0.0341	0.0306	-0.1635	0.0900
5	-0.0710	0.0633	-0.2700	-0.2124	0.0341	0.0306	-0.1635	0.0900
6	-0.0710	-0.1027	-0.2700	-0.4330	0.0341	0.0306	0.0922	0.0900

Acc_open_past_24mnths - Number of trades opened in past 24 months
Dti – A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower's self-reported monthly income
Term
Annual_inc
Sub_grade_B2/B1 - External assigned loan subgrade
Inq_last_6mths - The number of inquiries in past 6 months (excluding auto and mortgage inquiries)
Avg_cur_bal - Average current balance of all accounts

Model construction, model quality

Model	logreg	Decision tree	Random Forest	GBoost	MLP Classifier
ROC-AUC	0.66563298	0.64474354	0.62471941	0.6613518	0.66534098

Having tried all the parameters through the for cycle, the optimal ones for logistic regression were : C=0.1, class_weight = '**balanced**', penalty: **L1-regularisation**.
Also, I decided to use threshold=0.5, as it didn't get better when I changed it.

Model specification: logistic regression

$$P = \frac{1}{1 + e^{y}},$$

y = −1.66 +0.79∗acc_open_past_24mths + 0.45∗dti + 1.017∗term+ 1.087∗annual_inc

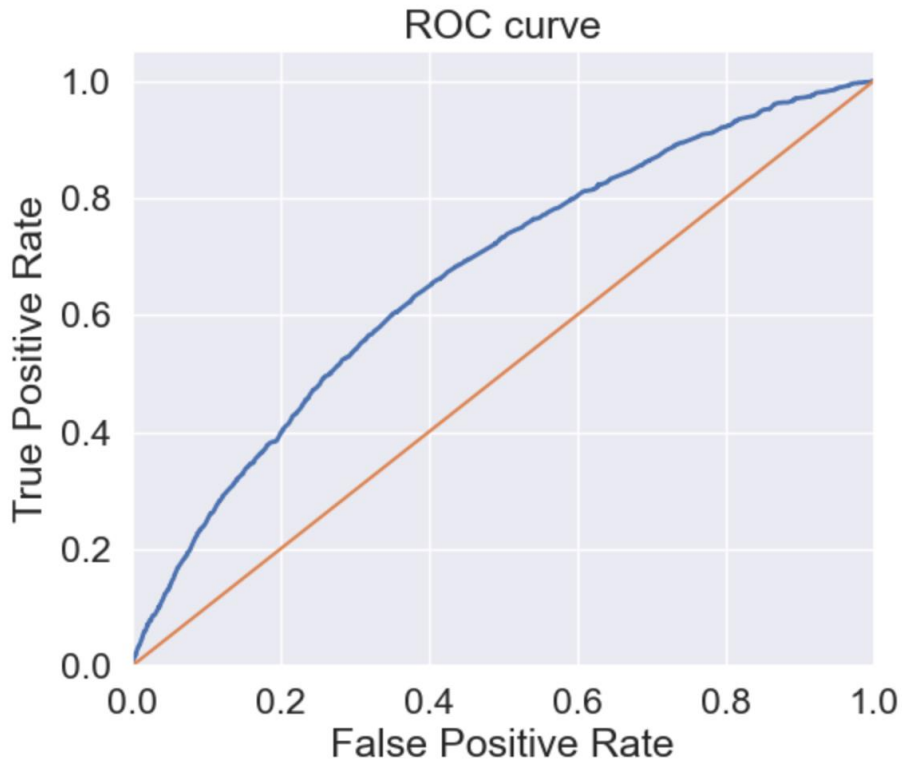
+0.595∗sub_grade_B2 + 0.59∗sub_grade_B1 + 0.77∗inq_last_6mths + 0.75∗ avg_cur_bal

	precision	recall	f1-score	support
no_def	0.90	0.65	0.75	10281
def	0.25	0.60	0.35	1952
accuracy			0.64	12233
macro avg	0.57	0.62	0.55	12233
weighted avg	0.79	0.64	0.69	12233

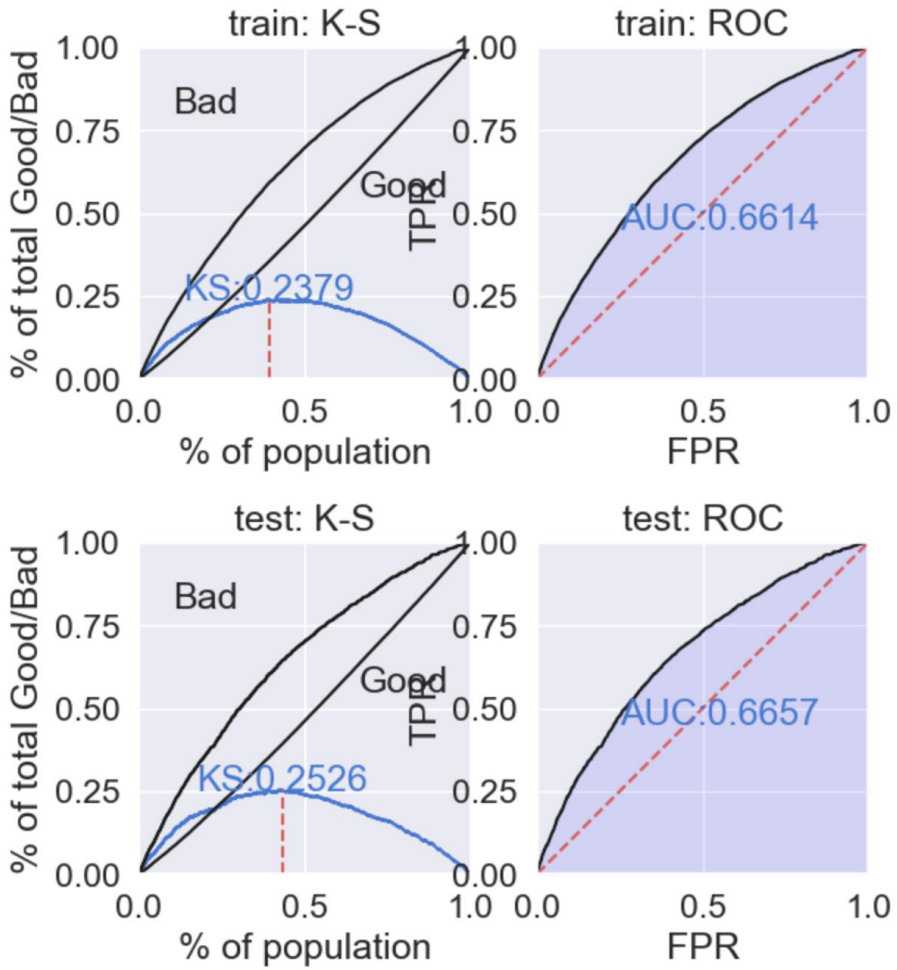
Area under the ROC curve - 0.6248

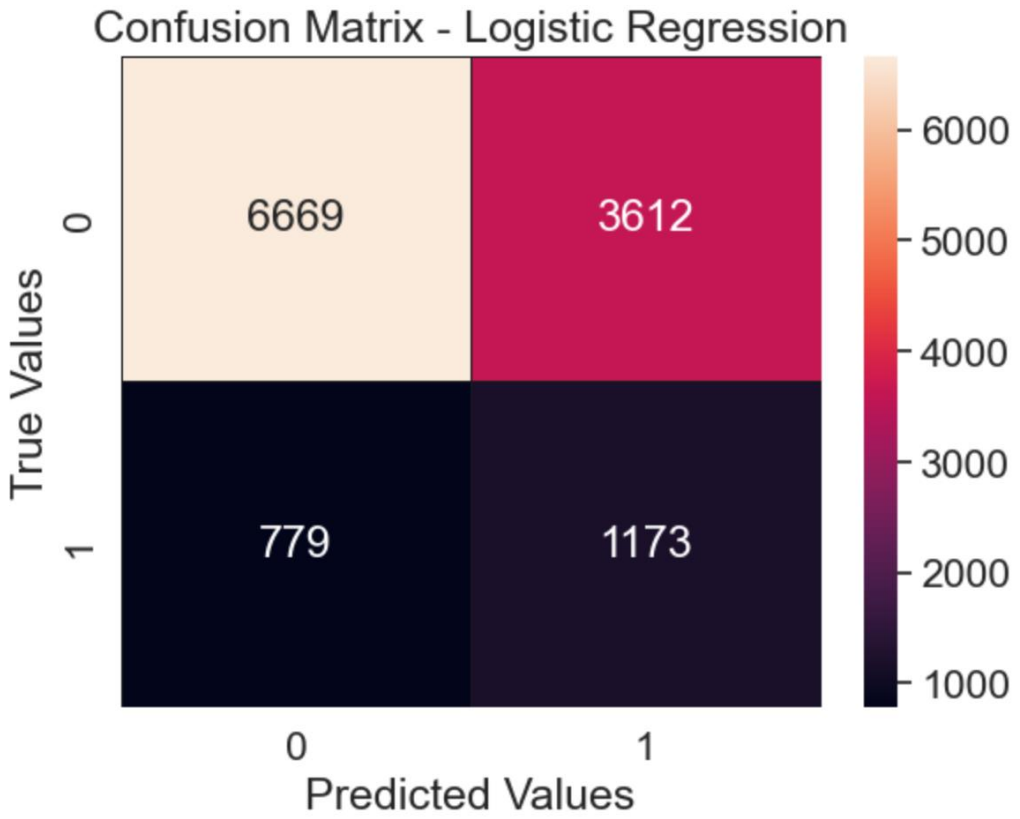
Area under the ROC curve for probabilities, test sample - 0.6657030924863787

Gini - 0.3314061849727574

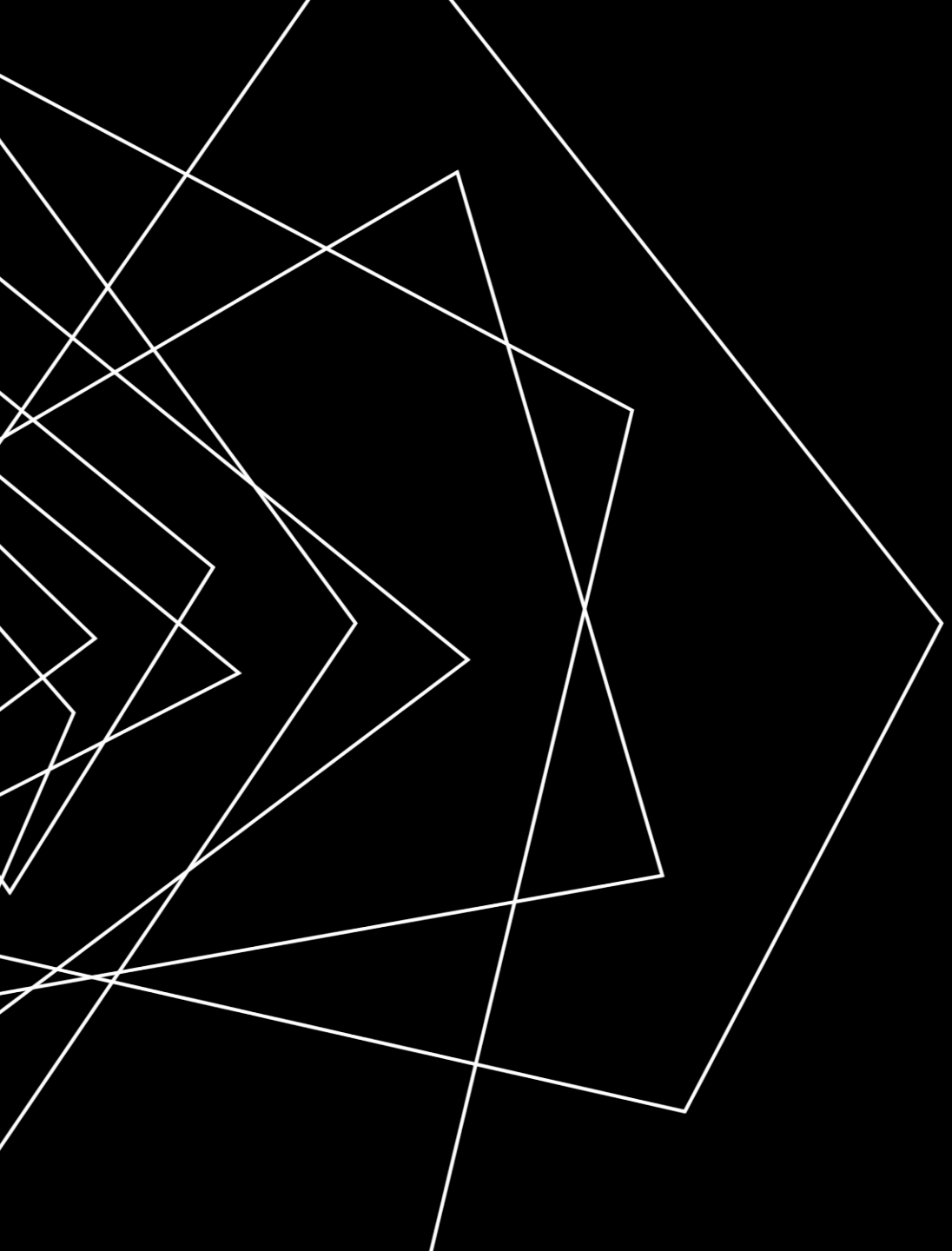


Results of the logistic regression





I consider, it should be better. But other parameters of logreg leads to the situation when model predict only 0, so forecasts to issue loans for anybody.



Thanks!