AN INVESTIGATION INTO THE USE OF BIG DATA FEATURES FOR CREDIT SCORING IN ONLINE LENDING

DENYS OSIPENKO, PHD
HEAD OF RISK, TAMGA FINANCE

WWW.PROFITSCORING.COM

CSCCXVI, 28-30 August 2019, Edinburgh

Type of features in online lending

We need extra data for online lending

- Standard credit scoring from banking lending shows poor results for online pay-day loans
- We do not contact an applicant face-to-face, only make a call in some cases
- The key question in the risk assessment process in online lending is

  'What extra data can we collect from a client?
- Client behavior on the network gives features which are non-typical for banking offline lending:
  - Surfing the Internet – web-sites, search results, cookies
  - Usage of mobile devices and laptops – UserAgent
  - Social network – profile, customer preferences, and networks of friends
  - Application form filling – Java-script, which scan and grab customer actions on the web-site
  - Profile changes – history of data which customer provide and what we knew and know about him/her

  These features can be even stronger predictors for the risk assessment than application socio-demographic and financial data

Data Sources in Online Lending

- Credit Bureau:
  - Negative history (e.g., DPD 60+)
  - Positive history (# request, DPD 1-30)
  - Rating
- Transactional data – client usage of credit and debit cards:
  - incoming and outcoming client transactions –
    - Source of Money: Where does money come FROM?
    - Spending profile: Where do you spend your money?
  - Balances – Do you save you money? Is it enough for you?
  - However, can be limited according to regulation (as GDPR)
- Client's behaviour while filling in the application and user data
- Devices, IPs, and client accounts in the network
- Geolocation: IP and Address: deviation and mapping
- Social networks data – . Text, pictures, video of the client… ☺ Who you are and your friends? The legal usage is limited now
- Client photos
- Client voice recording

Data structures: XML и JSON - NoSQL

JSON Example


```
“accountReportList": [{"number": "61 1020 4708 0000 XXXX XXXX XXXX",
                "totalNumberOfTransactions": 663,
                "wholeMonthsAvailable": 11,
                "averageNumberOfTransactionsWholeMonth": 56.72727272727273,
                "averageAmountOfIncomingTransactionsWholeMonth": 9246.59,
                "averageAmountOfOutgoingTransactionsWholeMonth": -9365.02,
                "averageMinimumBalanceWholeMonth": 201.59,
                "cashFlow": [{
                                "calendarMonth": "2017-08-01",
                                "incoming": 1583.06,
                                "outgoing": -1175.89,
                                "minBalance": -995.8,
                                "maxBalance": -304.86,
                "avgBalance": -872.12,
                "isWholeMonth": false
                },
                                {
                                "calendarMonth": "2017-07-01",
                                "incoming": 5897.02,
                                "outgoing": -5928.64,
                                "minBalance": -998.31,
                                "maxBalance": 696.51,
                                "avgBalance": -778.35,
                                "isWholeMonth": true
                                },]
```


XML example


```
<accountReportList>
                <number>61 1020 4708 0000 XXXX XXXX XXXX</number>
                <totalNumberOfTransactions>663</totalNumberOfTransactions>
                <wholeMonthsAvailable>11</wholeMonthsAvailable>
                <averageNumberOfTransactionsWholeMonth>56.72727272727273</averageNumberOfTransact
ionsWholeMonth>
                <averageAmountOfIncomingTransactionsWholeMonth>9246.59</averageAmountOfIncomingTra
nsactionsWholeMonth>
                <averageAmountOfOutgoingTransactionsWholeMonth>-
9365.02</averageAmountOfOutgoingTransactionsWholeMonth>
                <averageMinimumBalanceWholeMonth>201.59</averageMinimumBalanceWholeMonth>
                <cashFlow>
                                <calendarMonth>2017-08-01</calendarMonth>
                                <incoming>1583.06</incoming>
                                <outgoing>-1175.89</outgoing>
                                <minBalance>-995.8</minBalance>
                                <maxBalance>-304.86</maxBalance>
                                <avgBalance>-872.12</avgBalance>
                                <isWholeMonth>false</isWholeMonth>
                </cashFlow>

…
```


For Credit scoring model development – convert NoSQL data base into relation data base, and then – in train/test samples at the account of client level

Methodology of Investigation

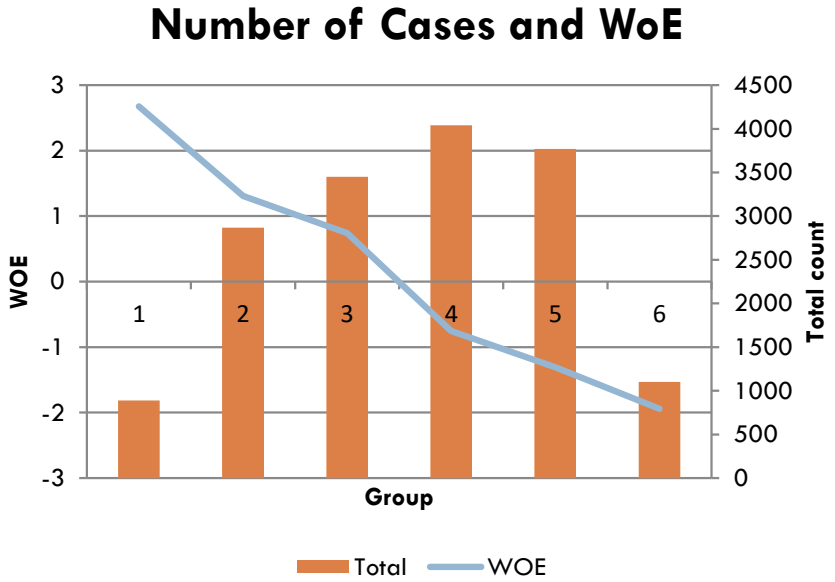
- Sample and target
  - Data sample period – 6 month
  - Train sample: 2680 New и 4597 Existing Clients of Pay-day Loans, Loan Term to 30 days
  - Target (0/1) Variable – Days Past Due 35+ Ever. Performance period for target – 3 month
- Predictors selection and binning
  - Preliminary selection of features approach – Gini: value and stability (an average of monthly Gini of each feature for the selected target)
  - Final selection of predictors – binning and IV > 0.02
  - Features – WoE values for each bin
  - We apply *R::scorecards* for features binning
  - bins = woebin(dt_sel, y="GB", min_perc_coarse_bin = 0.1, min_perc_fine_bin = 0.01, max_num_bin = 4)
  - An investigation of the impact of each group of features (application, behavioural, device etc.) on the final model Gini.
  - Assumption for the feature effect on the model – what if the model is built with a single feature only

Weight of Evidence and Information Value

We use ***Weight of Evidence*** (WoE) measure for each bin *i* of the variable

$$WoE_i = \ln\left(\frac{p_i}{q_i}\right)$$

For example, attribute 20-25 years,
good i = 100, bad i = 20,
good=1000, bad=50
WoE i = ln(0.1/0.4) = -1.38629



Information value of attribute – «predictive power»

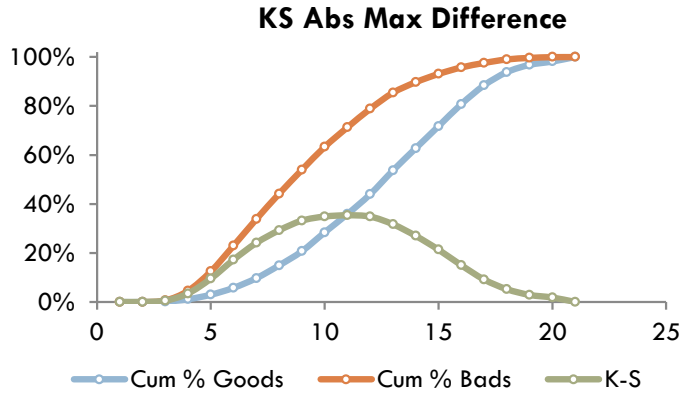
$$IV_i = (p_i - q_i)\ln\left(\frac{p_i}{q_i}\right)$$

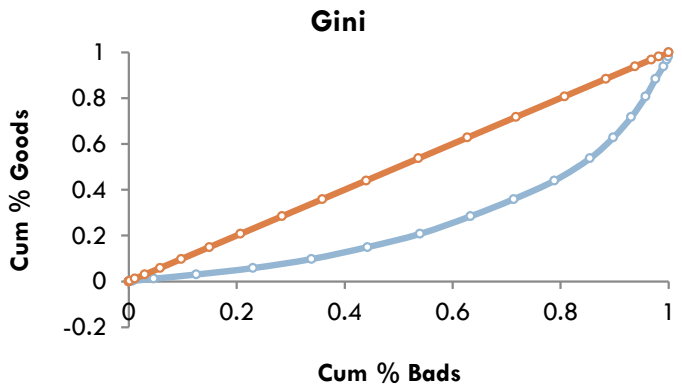
Information value of the feature is computed as:

$$IV = \sum_{i=1}^{n}(p_i - q_i)\ln\left(\frac{p_i}{q_i}\right)$$

Example of Validation: KS and Gini

Population equal split	Score		Marginal goods	Marginal bads	Bad rate	Cum. Clients		Cum. Goods		Cum. Bads		Gini			KS test
	up to	#	#	#	%	#	%	#	%	#	%	Bi-Bi-1	Gi+Gi-1	(Bi-Bi-1) * (Gi+Gi-1)	K-S Difference
0	1	0			0.00%	0	0	0	0	0	0	0	0	0.00%	0%
1	0.95-1	145	7	138	95.17%	145	1%	7	0.15%	138	0.61%	0.0	0.0	0.00%	0%
2	0.9-0.95	7 360	609	6 751	91.73%	7 505	27%	616	13.51%	6 889	30.28%	0.3	0.1	4.06%	17%
3	0.85-0.9	10 447	1 379	9 068	86.80%	17 952	66%	1 995	43.77%	15 957	70.14%	0.4	0.6	22.83%	26%
4	0.8-0.85	3 552	607	2 945	82.91%	21 504	79%	2 602	57.09%	18 902	83.09%	0.1	1.0	13.06%	26%
5	0.75-0.8	1 614	364	1 250	77.45%	23 118	85%	2 966	65.07%	20 152	88.58%	0.1	1.2	6.71%	24%
6	0.7-0.75	1 440	403	1 037	72.01%	24 558	90%	3 369	73.91%	21 189	93.14%	0.0	1.4	6.34%	19%
7	0.65-0.7	1 136	372	764	67.25%	25 694	94%	3 741	82.08%	21 953	96.50%	0.0	1.6	5.24%	14%
8	0.6-0.65	557	200	357	64.09%	26 251	96%	3 941	86.46%	22 310	98.07%	0.0	1.7	2.64%	12%
9	0.55-0.6	216	95	121	56.02%	26 467	97%	4 036	88.55%	22 431	98.60%	0.0	1.8	0.93%	10%
10	0.5-0.55	110	59	51	46.36%	26 577	97%	4 095	89.84%	22 482	98.82%	0.0	1.8	0.40%	9%
11	0.45-0.5	114	67	47	41.23%	26 691	98%	4 162	91.31%	22 529	99.03%	0.0	1.8	0.37%	8%
12	0.4-0.45	168	92	76	45.24%	26 859	98%	4 254	93.33%	22 605	99.36%	0.0	1.8	0.62%	6%
13	0.35-0.4	207	150	57	27.54%	27 066	99%	4 404	96.62%	22 662	99.61%	0.0	1.9	0.48%	3%
14	0.3-0.35	132	76	56	42.42%	27 198	100%	4 480	98.29%	22 718	99.86%	0.0	1.9	0.48%	2%
15	0.25-0.3	48	31	17	35.42%	27 246	100%	4 511	98.97%	22 735	99.93%	0.0	2.0	0.15%	1%
16	0.2-0.25	21	17	4	19.05%	27 267	100%	4 528	99.34%	22 739	99.95%	0.0	2.0	0.03%	1%
17	0.15-0.2	11	7	4	36.36%	27 278	100%	4 535	99.50%	22 743	99.97%	0.0	2.0	0.03%	0%
18	0.1-0.15	27	20	7	25.93%	27 305	100%	4 555	99.93%	22 750	100.00%	0.0	2.0	0.06%	0%
19	0.05-0.1	3	3	0	0.00%	27 308	100%	4 558	100.00%	22 750	100.00%	0.0	2.0	0.00%	0%
20	0-0.05				0.00%	27 308	100%	4 558	100.00%	22 750	100.00%	0.0	2.0	0.00%	0%
Total		**27 308**	**4 558**	**22 750**	**83.31%**	**27 308**	**100%**	**4 558**	**100.00%**	**22 750**	**100.00%**			**35.57%**	**26.37%**





We use GINI measure instead of Information Value for features preselection.

For preselection we use features with automated binning.

So inputs are GINI on _WOE

Typical application and behavioural features

Gini of Application Features

Variable	2018-10	2018-11	2018-12	2019-01	2019-02	2019-03	Avg Gini
Age_woe	0.0044	0.1209	0.0254	-0.0028	0.0397	0.0585	0.0410
Education_woe	0.2144	0.0296	0.1066	-0.0967	0.0407	0.1017	0.0660
WorkExperience_woe	0.0664	0.2012	0.0729	-0.0693	0.1371	0.0739	0.0804
TotalMonthlyPayments_woe	0.0696	0.0779	0.2452	0.0846	0.0911	-0.0267	0.0903
RequestedAmount_woe	0.1317	0.0046	0.0019	0.0629	0.0436	0.1380	0.0638
RequestedPeriod_woe	0.0570	0.1213	0.0852	0.1792	0.1331	0.0287	0.1007
DaysInCompany_woe	**0.1218**	**0.1340**	**0.0132**	**0.0649**	**0.1376**	**0.1995**	**0.1118**
DTI_woe	**0.1061**	**0.2510**	**0.2854**	**0.1041**	**0.0524**	**-0.0206**	**0.1298**
ReqAmnt_to_PreviousAmt_woe	**0.1911**	**0.1846**	**0.1922**	**0.2456**	**0.1342**	**0.0951**	**0.1738**
age_Gender_2_woe	0.0712	0.1454	0.0689	0.0350	0.0781	0.0636	0.0770
AdditionalPhone_woe	-0.0218	0.0447	0.0984	-0.0728	0.1228	0.1140	0.0476
Income_woe	0.1093	0.0339	0.1612	-0.0074	0.0420	0.1270	0.0776
IncomeSource_woe	0.0172	0.1471	0.0807	0.0136	0.0305	0.0417	0.0551
SourceMainIncome_woe	0.0126	0.1496	0.0732	0.0524	0.0514	0.0473	0.0644
IndustryName_woe	0.0397	0.0462	0.0702	0.0704	0.0862	0.0173	0.0550
PropertyType_woe	**0.0833**	**0.1617**	**0.1963**	**0.0350**	**0.1234**	**0.0905**	**0.1150**
CreatedWeekdayHour_woe	0.0464	0.1641	0.0879	-0.0050	0.0785	0.0291	0.0668
Time_App_woe	0.0793	0.1211	-0.0150	-0.0019	0.0403	0.0847	0.0514

Days with Company, Debt-to-Income, Requested Amount to Previous Loan Amount are the strongest features because of the highest average GINI value.

However, proposed binning have even negative predictive power in some months.

Gini of History of Applications

Variable	2018-10	2018-11	2018-12	2019-01	2019-02	2019-03	Avg Gini
PreviousApproved_woe	**0.051513**	**0.111674**	**0.070768**	**0.255221**	**0.122339**	**0.068607**	**0.113354**
PreviousAutoDeclined_woe	-0.03387	-0.02197	0.070714	0.019563	0.052872	-0.02485	0.010411
PreviousCounterOfferDeclined_woe	0.041025	0.017358	-0.00301	0.09308	0.055203	-0.0049	0.033126
RequestsPerDaySum_woe	0.045313	0.007719	0.123708	0.026734	0.046529	-0.06312	0.031148
RequestsPerDayAvg_woe	0.062056	0.048733	-0.00514	0.207881	0.094462	0.11561	0.087267
RequestsPerDaySD_woe	0.029553	0.029954	-0.01537	0.231342	0.077365	0.103825	0.076112
RequestsPerDayScopeToAvg_woe	0.056648	0.007796	-0.04315	0.160471	0.019604	0.108429	0.051633
RequestsPerDaySdToAvg_woe	0.034934	-0.00545	-0.00077	0.179268	0.124094	0.026109	0.059698
RequestDaysNum_woe	0.045313	0.007719	0.123708	0.026734	0.046529	-0.06312	0.031148
DaysAfterFirstRequest_woe	**0.115864**	**0.160599**	**0.102434**	**0.149889**	**0.16774**	**0.098414**	**0.132490**
RequestDaysProc_woe	**0.076013**	**0.074117**	**0.107629**	**0.107491**	**0.242022**	**0.186853**	**0.132354**
DaysAfterPrevRequestsSum_woe	**0.081995**	**0.183295**	**0.075636**	**0.16423**	**0.192359**	**0.082045**	**0.129927**
DaysAfterPrevRequestsAvg_woe	**0.096553**	**0.068779**	**0.115997**	**0.211014**	**0.174058**	**0.120686**	**0.131181**
DaysAfterPrevRequestsMax_woe	0.034497	0.099962	0.180968	0.104845	0.109479	0.051076	0.096804
DaysAfterPrevRequestsMin_woe	0.003278	0.034332	0.076019	0.004247	0.118679	0.085606	0.053693
DaysAfterPrevRequestsScope_woe	0.055282	0.104647	0.08805	0.139864	0.070095	-0.00474	0.075533
DaysAfterPrevRequestsSD_woe	0.071561	0.029263	0.058682	0.139655	0.06651	0.017098	0.063795
DaysAfterPrevRequestsScopeToAvg_woe	0.063586	0.093625	0.057971	0.152952	0.113415	0.089757	0.095218
DaysAfterPrevRequestsSdToAvg_woe	0.036982	0.158564	-0.0373	0.19166	0.141994	0.061012	0.092152
RequestsNum_woe	0.045313	0.007719	0.123708	0.026734	0.046529	-0.06312	0.031148

The strongest features are
Number of Previous approved applications,
Number of Days from the Previous request,
Average Number of Days from ALL Previous requests,
The Percentage of Days with Requests in total number of days with the Company

Gini of Behavioural Features (1/2)

Variable	2018-10	2018-11	2018-12	2019-01	2019-02	2019-03	Avg Gini
DaysFromLastPaidBack_woe	0.1409	0.1270	0.1215	0.0010	0.0622	0.0424	0.0825
MaxPreviousDPD_woe	**0.0680**	**-0.1514**	**0.0075**	**0.1035**	**0.0938**	**-0.0130**	**0.0181**
TotalPreviousPaid_woe	**0.0713**	**0.1320**	**0.0360**	**0.2492**	**0.1192**	**0.0636**	**0.1119**
TotalPreviousAmount_woe	**0.0707**	**0.0427**	**0.1140**	**0.2277**	**0.1833**	**0.1306**	**0.1282**
PreviousTotalRepayments_woe	**0.1684**	**0.0712**	**0.0473**	**0.2805**	**0.1454**	**0.1537**	**0.1444**
PreviousTotalOverpayment_woe	0.0046	0.1056	0.0137	0.1485	0.1639	0.0428	0.0799
PreviousMaxAmount_woe	0.0806	0.1064	0.1112	0.1854	0.1753	0.1630	0.1370
PreviousAveragePeriod_woe	0.1025	0.0599	0.0620	0.0840	0.0448	-0.0381	0.0525
PreviousAverageFactPeriod_woe	0.0792	0.1084	0.0311	-0.0104	0.0227	0.0708	0.0503
PreviousPaymentsCount_woe	**0.1197**	**0.1373**	**0.0155**	**0.2415**	**0.1232**	**0.0655**	**0.1171**
PreviousRepaymentFee_woe	**0.0535**	**0.0688**	**0.0127**	**0.1725**	**0.1827**	**0.1173**	**0.1013**
PreviousEarlyPaidBacks_woe	**0.0474**	**0.1294**	**0.0724**	**0.2455**	**0.1186**	**0.0248**	**0.1063**
PreviousTotalOverpayment_prc_woe	0.1138	0.0193	0.1290	-0.0488	0.0894	0.1028	0.0676
PreviousTotalRepayments_prc_woe	0.0959	0.0104	0.1118	0.2203	-0.0533	-0.0618	0.0539
PreviousTotalEarlyPayment_prc_woe	0.0350	0.0858	0.0436	0.1568	0.0226	0.0172	0.0602

The strongest features are

Number of Previous Paid Backs,
Total Previous Loan Amount, Sum of Previous Repayments,
Number of Previous Early Pay Backs (prepayments)

BUT
Not features connected with delinquency (DPD) and arrears amounts.

THUS it's better to predict Good client than Bad client

Gini of Behavioural Features (2/2)

Variable	2018-10	2018-11	2018-12	2019-01	2019-02	2019-03	Avg Gini
DaysAfterPrevPaidBacksSum_woe	0.1060	0.1028	0.0677	0.2465	0.1424	0.0801	0.1243
DaysAfterPrevPaidBacksAvg_woe	**0.1231**	**0.1918**	**0.0956**	**0.0937**	**0.2527**	**0.0939**	**0.1418**
DaysAfterPrevPaidBacksMax_woe	0.1303	0.1228	0.0677	0.2306	0.1492	0.0772	0.1296
DaysAfterPrevPaidBacksMin_woe	0.0728	0.1825	0.0386	0.2012	0.0830	0.0728	0.1085
DaysAfterPrevPaidBacksScope_woe	0.1146	0.0633	0.0454	0.2342	0.1464	0.0763	0.1134
DaysAfterPrevPaidBacksSD_woe	0.1012	0.0875	0.0421	0.2256	0.1417	0.0758	0.1123
PreviousApplications.PaidBack.Last30Days.MinSum_woe	0.0524	0.0728	0.0438	0.1499	0.1406	0.1570	0.1028
PreviousApplications.PaidBack.Last30Days.MaxSum_woe	0.0117	0.0609	0.0535	0.1519	0.0883	0.1266	0.0822
PreviousApplications.PaidBack.Last30Days.AvgSum_woe	-0.0007	0.0463	0.0439	0.1487	0.0928	0.1190	0.0750
PreviousApplications.PaidBack.Last30Days.TotSum_woe	0.0181	0.0595	0.0551	0.1542	0.1021	0.1258	0.0858
PreviousApplications.PaidBack.Last30Days.AvgSumToReqAmt_woe	0.0485	0.1386	0.1204	0.0804	0.1228	0.1541	0.1108

Various aggregates such as Min, Max, Average, Scope or Range (difference between Max and Min values), Standard Deviation might BE GOOD predictors. Or might NOT ☺

Customer behaviour when filling out the online application form

Gini of Web-form Client Behaviour

Variable	2018-10	2018-11	2018-12	2019-01	2019-02	2019-03	Avg Gini
CurrentApplication.SliderAmount.Max	**0.1668**	**0.0518**	**0.0513**	**0.1237**	**0.0296**	**0.0918**	**0.0858**
CurrentApplication.SliderAmount.Changes	**0.0934**	**0.0035**	**0.0760**	**0.1695**	**0.0958**	**0.1101**	**0.0914**
CurrentApplication.SliderTerm.Changes	0.0624	0.0895	-0.0142	0.0804	0.1242	0.0140	0.0594
PersonalDetails.FirstName.Time	-0.0317	0.0789	0.1585	-0.0199	-0.0309	0.1150	0.0450
PersonalDetails.FirstName.Keypress	0.1215	-0.0072	-0.0905	0.0407	0.0485	0.0614	0.0291
PersonalDetails.Surname.Time	0.1093	0.0344	0.1076	0.0661	0.0099	0.0006	0.0546
PersonalDetails.Surname.Keypress	0.0837	-0.0027	0.0032	0.0001	-0.0805	0.0321	0.0060
AddressDetails.City.Time	0.0032	-0.0247	-0.0115	0.0693	0.1476	0.0630	0.0411
AddressDetails.City.Keypress	0.1144	-0.0224	-0.1158	0.0527	0.1374	0.0783	0.0408
AddressDetails.City.Backspace	0.0000	0.0000	-0.0050	0.0042	0.1333	0.0606	0.0322
AddressDetails.HouseNumber.Time	0.1359	-0.1182	0.0158	0.0812	0.1696	0.0648	0.0582
AddressDetails.HouseNumber.Keypress	0.0232	-0.0392	0.0638	0.0267	0.1628	0.0623	0.0422
AddressDetails.HouseNumber.Backspace	0.0354	-0.0425	0.0201	0.0425	0.1441	0.0645	0.0322
PersonalDetails.MobilePhone.Activate	**0.0952**	**0.0207**	**0.1928**	**0.0524**	**0.1158**	**-0.0185**	**0.0764**
PersonalDetails.MobilePhone.Time	**0.1291**	**0.0440**	**0.0087**	**0.1627**	**0.1068**	**0.0736**	**0.0875**
PersonalDetails.MobilePhone.Keypress	0.0298	-0.0640	0.0686	0.1562	0.0477	0.0558	0.0490
PersonalDetails.MobilePhone.Click	0.0778	0.0340	0.0615	0.1114	0.0556	0.0263	0.0611
PersonalDetails.Email.Time	-0.0116	0.0978	0.0319	0.0611	0.1589	0.0552	0.0656
PersonalDetails.Email.Keypress	-0.0121	0.0247	-0.0045	-0.0400	0.1844	0.0558	0.0347
PersonalDetails.Email.Click	0.0066	0.0529	0.0602	0.0640	0.1659	0.0524	0.0670
PersonalDetails.Email.Backspace	0.0000	0.0000	-0.0050	0.0042	0.1333	0.0606	0.0322

The strongest features are Max Loan Amount in Web-form Calculator,
# of Changes of Requested Loan Amount,
# of Activation of Mobile Phone field and
Time of Mobile Phone field filling.

Features as the Number of Keypress for email are non-predictable and instable because correlated with the length of email ☺

An example: time of client's First Name filling

New client

Variable	Bin	Count	Count_distr	Good	Bad	BadProb	WoE	Bin_IV	Total_IV
ClientBehaviour_FirstName_Time	[-Inf,4500)	592	0.2209	399	193	0.3260	-0.2005	0.0086	0.0268
ClientBehaviour_FirstName_Time	[4500,7000)	500	0.1866	285	215	0.4300	0.2439	0.0114	0.0268
ClientBehaviour_FirstName_Time	[7000,12500)	815	0.3041	533	282	0.3460	-0.1108	0.0037	0.0268
ClientBehaviour_FirstName_Time	[12500, Inf)	772	0.2881	467	305	0.3951	0.0998	0.0029	0.0268

Existing client

Variable	Bin	Count	Count_distr	Good	Bad	BadProb	WoE	Bin_IV	Total_IV
ClientBehaviour_FirstName_Time	[-Inf,1000)	1560	0.3394	1189	371	0.2378	0.0257	0.0002	0.0122
ClientBehaviour_FirstName_Time	[1000,7500)	1489	0.3239	1179	310	0.2082	-0.1455	0.0066	0.0122
ClientBehaviour_FirstName_Time	[7500,10500)	494	0.1075	360	134	0.2713	0.2021	0.0046	0.0122
ClientBehaviour_FirstName_Time	[10500, Inf)	1054	0.2293	797	257	0.2438	0.0586	0.0008	0.0122

The time of the Client's First Name filling out is NON Monotonic

The time of the Client's First Name filling out has moderate predictive power (Information Value) for New Clients, BUT is not significant for Existing Clients. So it does not matter for the second and subsequent loans how long the client filled out his/her name in the registration form.

Non-monotonic Behaviour

Variable	Bin	Count	Count_distr	Good	Bad	BadProb	WoE	Bin_IV	Total_IV
ClientBehaviour_SliderAmount_Changes	[-Inf,1)	987	0.2147	732	255	0.2584	0.1358	0.0041	0.0235
ClientBehaviour_SliderAmount_Changes	[1,3)	2105	0.4579	1647	458	0.2176	-0.0895	0.0036	0.0235
ClientBehaviour_SliderAmount_Changes	[3,5)	685	0.1490	492	193	0.2818	0.2546	0.0103	0.0235
ClientBehaviour_SliderAmount_Changes	[5, Inf)	820	0.1784	654	166	0.2024	-0.1808	0.0055	0.0235

Number of changes of requested amount – How many times the client moved the requested amount slider:

- more than 5 slider moves is better than 3-5 moves
- 1-2 slider moves is better than 3-4 moves

The Bad Rate and WoE values can be considered as statistically significant because of the observation number.

Options for modelling procedure:

1) To **accept non-monotonic tendencies** and try to explain the reasons of such dependence between predictor and target variable

2) To **merge bins to get the monotonic dependence** between predictor and target, and… get IV -> 0

Behaviour in Network: Devices and Geolocation

Gini of features related to Device Type and Geolocation

Variable	2018-10	2018-11	2018-12	2019-01	2019-02	2019-03	Avg Gini
IpDistance_Sum_woe	-0.0021	-0.0158	-0.0176	0.0882	0.0821	0.0287	0.0272
IpDistance_Avg_woe	0.0190	-0.0255	-0.0514	0.1025	0.0166	0.1302	0.0319
IpDistance_Max_woe	-0.0455	0.0505	-0.0494	0.0255	0.0797	0.0720	0.0221
IpDistance_Min_woe	0.0125	-0.0845	-0.0597	0.0471	0.0003	0.1775	0.0155
IpDistance_Scope_woe	-0.0328	0.1109	-0.1254	0.0308	0.0489	0.0178	0.0084
IpDistance_SD_woe	0.0681	-0.0013	0.0263	-0.0046	0.0234	0.0741	0.0310
IpDistance_ScopeToAvg_woe	**0.0589**	**-0.0024**	**0.2156**	**0.0948**	**0.0273**	**0.0119**	**0.0677**
IpDistance_SdToAvg_woe	**0.0569**	**0.0909**	**0.1219**	**0.0876**	**-0.0323**	**0.0048**	**0.0550**
RequestsFromLastIpNum_woe	-0.0082	0.0147	0.0068	0.0608	0.1063	-0.0146	0.0277
RequestsFromLastIpProp_woe	0.0033	-0.0266	0.0735	0.0850	0.0020	0.1293	0.0444
RequestsByIpSum_woe	0.0453	0.0077	0.1237	0.0267	0.0465	-0.0631	0.0311
RequestsByIpAvg_woe	0.0950	0.0430	0.0286	-0.0022	0.0018	0.0812	0.0412
RequestsByIpSD_woe	**0.0666**	**0.0197**	**0.1033**	**-0.0317**	**0.1301**	**0.0597**	**0.0579**
RequestsByIpScopeToAvg_woe	0.0326	-0.0492	0.1032	0.0931	0.1013	-0.0046	0.0461
RequestsByIpSdToAvg_woe	0.0881	-0.0313	0.0564	-0.0817	0.2040	0.0175	0.0422
PrevEqualIpNum_woe	0.0392	-0.0031	0.1105	-0.0084	0.0885	0.0719	0.0498
Browser_woe	0.0444	0.0172	0.0791	-0.0107	0.0344	0.0608	0.0375
BrowserVersion_woe	0.0642	-0.0255	0.0802	0.0223	0.0155	0.0554	0.0353
OS_woe	**0.1380**	**-0.0481**	**0.0994**	**0.1009**	**0.0616**	**0.1066**	**0.0764**
OsVer_woe	0.1380	-0.0456	0.1019	0.0984	0.0629	0.1075	0.0772

Distance between IP of the previous client applications is not high predictive feature,
BUT the statistical aggregates such as Range to Average Distance and SD to Average Distance Between Application IP can be useful.

Number of Requests from the SAME IP, for example, Standard Deviation of the Number of Requests from the same IP.

These features might have a positive impact on the total Gini of the model because of low correlation with other traditional features.

Example of Features for Device Type and Geolocation

Variable	Bin	Count	Count_distr	Good	Bad	BadProb	WoE	Bin_IV	Total_IV
OS	Android	2860	0.6221	2133	727	0.2542	0.1140	0.0083	0.0291
OS	iOS	580	0.1262	448	132	0.2276	-0.0316	0.0001	0.0291
OS	Linux, MacOS, Windows, Windows Phone	1157	0.2517	944	213	0.1841	-0.2985	0.0206	0.0291

Low IV, but Low correlation with other features

Variable	Bin	Count	Count_distr	Good	Bad	BadProb	WoE	Bin_IV	Total_IV
IpDistanceSdToAvg	missing	916	0.1993	694	222	0.2424	0.0506	0.0005	0.0143
IpDistanceSdToAvg	[-Inf,0.3)	837	0.1821	650	187	0.2234	-0.0555	0.0006	0.0143
IpDistanceSdToAvg	[0.3,0.7)	828	0.1801	610	218	0.2633	0.1614	0.0049	0.0143
IpDistanceSdToAvg	[0.7,1.1)	1368	0.2976	1047	321	0.2346	0.0081	0.0000	0.0143
IpDistanceSdToAvg	[1.1, Inf)	648	0.1410	524	124	0.1914	-0.2509	0.0083	0.0143

The ‘best’ clients apply from Laptops, Computers, and Mobiles with Windows.
The ‘worst’ – use mobiles with Android.
IPhone users are the average group by the level of risk.

Standard Deviation to Mean of the Distance between IPs of Applications is weak feature:
IV ~ 0.014

BUT because of low correlation with other features it can be used as a predictor.

Client's behaviour in Network: example of credit rules (checkpoints)

Credit Rule Category	Examples
Age and Term of Device Usage	How long this device and client account are linked Is it New device for the network
Anomaly in client's device behaviour	Transactions through TOR / VPN Time zones of device and IP are different
Geolocation	Device IP is far from real IP which have been detected for this device Transaction from devices IP associated with high risk countries
Risk Profile	Device language parameters are associated with high risk countries or countries, which are untypical for our market Information about delivery and billing is not corresponding IP is in the list of high-risk IPs
Speed and density of transactions	Number of transactions per device/account Number of e-mail per device Number of countries per account

Client assessment in Social Networks

- Social Profile
  - Completeness of info, detailed information, history etc.
- Client's behavioural model
  - Posts and likes frequency; follows and followers; dynamic of network
  - Photos and pictures;
  - Text analysis (for grammar, content, specific words)
- Visual and text triggers (both manual and automatic) – extremism, depression, aggression, groups like 'How do not pay back my debt?'.
- Psychological portfolio of the client given from
  - personal preferences, communities, posts, comments etc.
  - third persons interaction
- Client links in different social networks (facebook, Instagram, linkedIn, local networks) and third part portfolio

– *Show me your friends and I'll tell you who you are*

External Data: Credit Bureau and Client Transactions

Example of Transactional Features

- AverageIncome3Months – Average sum of Incoming transactions for the last 3 months
- LastMonthOutgoing – Total amount of Outgoing transactions for the last month
- Tran_MaxLastMonthByAccount – Maximum Transactions Amount for the Last Month for all accounts of the client
- MinBalanceLastMonth – Minimum Balance for the Last Month
- Tran_AvailableMonthes – Number of months with transactions
- DTI (requested/income) – Debt to Income
- DaysAgoLastPayment – Number of days since the last payment transaction
- OI_trend6m – The coefficient of the slope of the Monthly Income Amount for the last six months which describes general increase or decrease or no change in Customer's Income
- max_AverageNumberOfTransactions – maximum of the Average number of Transaction by each account of the client

Binning and IV of some transactional features

variable	bin	count	count_distr	good	bad	badprob	woe	bin_iv	total_iv
Income	[-Inf,1250)	2374	0.467968	1380	994	0.418703	0.38284	0.072437	0.204004
Income	[1250,1550)	589	0.116105	393	196	0.332767	0.015246	2.71E-05	0.204004
Income	[1550,2350)	927	0.182732	663	264	0.28479	-0.20988	0.007744	0.204004
Income	[2350, Inf)	1180	0.232604	965	215	0.182203	-0.79055	0.122528	0.204004

variable	bin	count	count_distr	good	bad	badprob	woe	bin_iv	total_iv
max_AverageNumberOfTransactions	[-Inf,20)	1028	0.202641	530	498	0.484436	0.648664	0.092262	0.173191
max_AverageNumberOfTransactions	[20,44)	1674	0.329982	1082	592	0.353644	0.107881	0.003909	0.173191
max_AverageNumberOfTransactions	[44,66)	1145	0.225705	846	299	0.261135	-0.32913	0.022955	0.173191
max_AverageNumberOfTransactions	[66, Inf)	1226	0.241672	944	282	0.230016	-0.49728	0.054065	0.173191

Variable	bin	count	count_distr	good	bad	badprob	woe	bin_iv	total_iv
Outgoing_3Month	missing	826	0.162823	528	298	0.360775	0.138938	0.003214	0.100765
Outgoing_3Month	[-Inf,-18000)	600	0.118273	484	116	0.193333	-0.71755	0.052274	0.100765
Outgoing_3Month	[-18000,-9000)	992	0.195545	726	266	0.268145	-0.29311	0.015892	0.100765
Outgoing_3Month	[-9000,-4500)	1172	0.231027	775	397	0.338737	0.042014	0.000411	0.100765
Outgoing_3Month	[-4500, Inf)	1483	0.292332	889	594	0.400539	0.307723	0.028975	0.100765

Credit Bureau – Here is not included in scoring features, but is used as a part of decision making

- Negative history:
  - Take into account types of products (for example, the mortgage bad debt might be not the key factor for pay-day loan decline)
  - Bad debt term (for example, more than 1 year for the newest Bad Debt loan)
- Positive History:
  - Number of active loans of the client
  - Number of client applications compared to the number of loans
- Credit Rating:
  - Generic credit ratings should be validated and calibrated with your portfolio. Sometimes bad client for a bank can be good client for microfinance.

- EXPERIMENTS:
  - Limitation Period for Bad Debt. For example, MicroFinance with more than 12 month since BadDebt record and total amount less than 500 EUR
  - Type of Product: Mortgage, Car Loan, Business Credit Line are not obligatory say about a bad customer for pay-day loan

Credit Scoring Models: Traditional Approach and Machine Learning

Comparative analysis of models for various groups of features – new and existing clients – WITHOUT Credit Bureau

New Clients

Variable	KS	ROC	Gini
App + Behaviour Application + Device	0.26	0.67	0.34
+ Client Behaviour on Web-Form	0.29	0.69	0.38
Transactions only	0.28	0.68	0.36
+ Transactions = ALL	**0.32**	**0.73**	**0.46**

Existing Clients

Variable	KS	ROC	Gini
Application	0.20	0.65	0.30
Client Behaviour on Web-Form	0.13	0.58	0.16
Behaviour - Payment	0.25	0.67	0.35
Behaviour - Application	0.22	0.63	0.26
Devices	0.14	0.59	0.18
ALL	**0.32**	**0.73**	**0.46**

Transactional scoring provides the model for New clients with the SAME predictive power, as Behavioural model has WITHOUT transactional scoring

It's more difficult to create good model for existing clients (Behavioural Scoring) for PDL than for Banking Lending
where GINI > 0.5 is expected

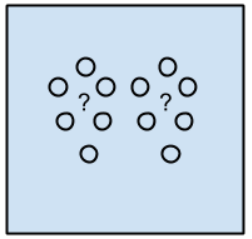
Machine Learning Algorithm Overview

Machine Learning Algorithms *(sample)*

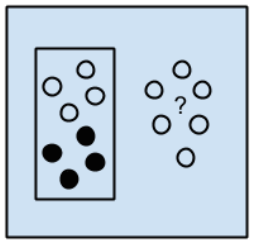
	Unsupervised	Supervised

Continuous	• Clustering & Dimensionality Reduction
	  ○ SVD
	  ○ PCA
	  ○ K-means
		• Regression
		  ○ Linear
		  ○ Polynomial
		• Decision Trees
		• Random Forests

Categorical	• Association Analysis
	  ○ Apriori
	  ○ FP-Growth
	• Hidden Markov Model
		• Classification
		  ○ KNN
		  ○ Trees
		  ○ Logistic Regression
		  ○ Naive-Bayes
		  ○ SVM



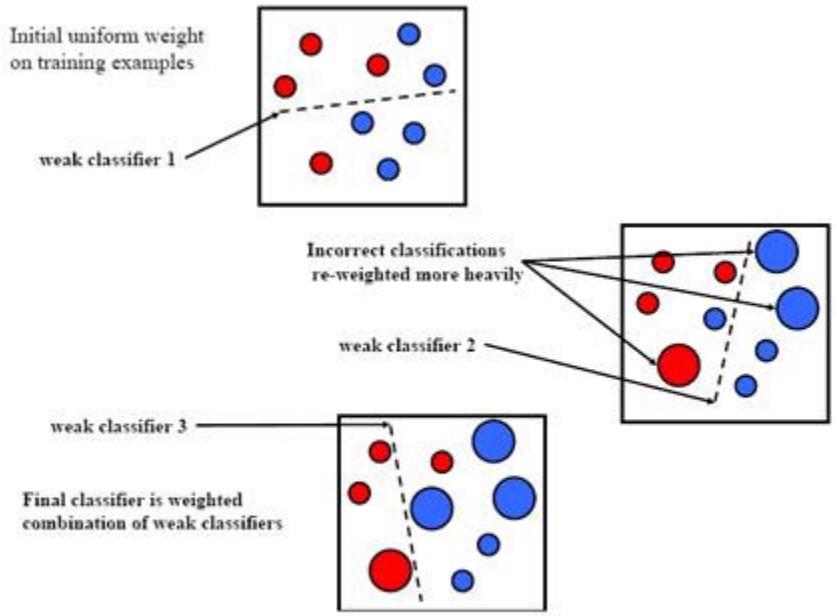
Unsupervised Learning Algorithms



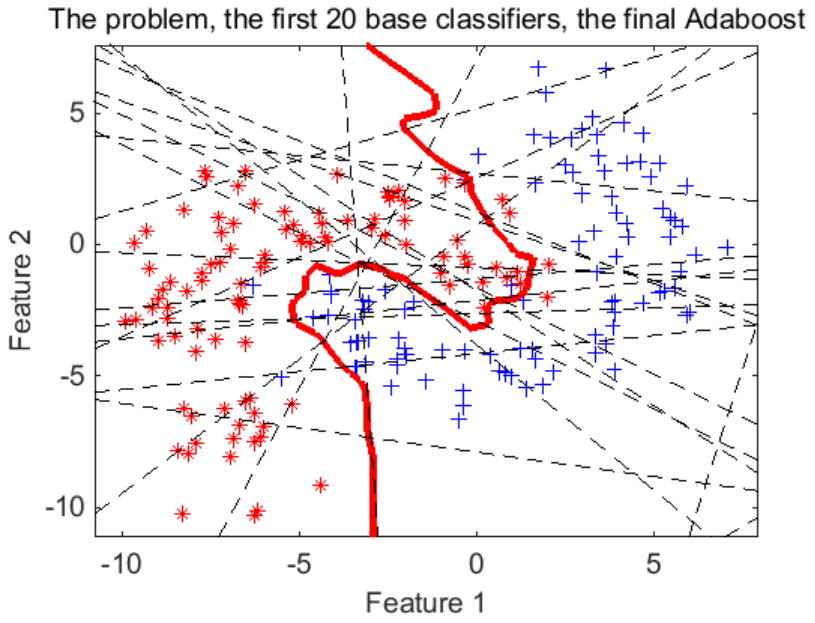
Supervised Learning Algorithms

ML: for example, boosting (XGB)

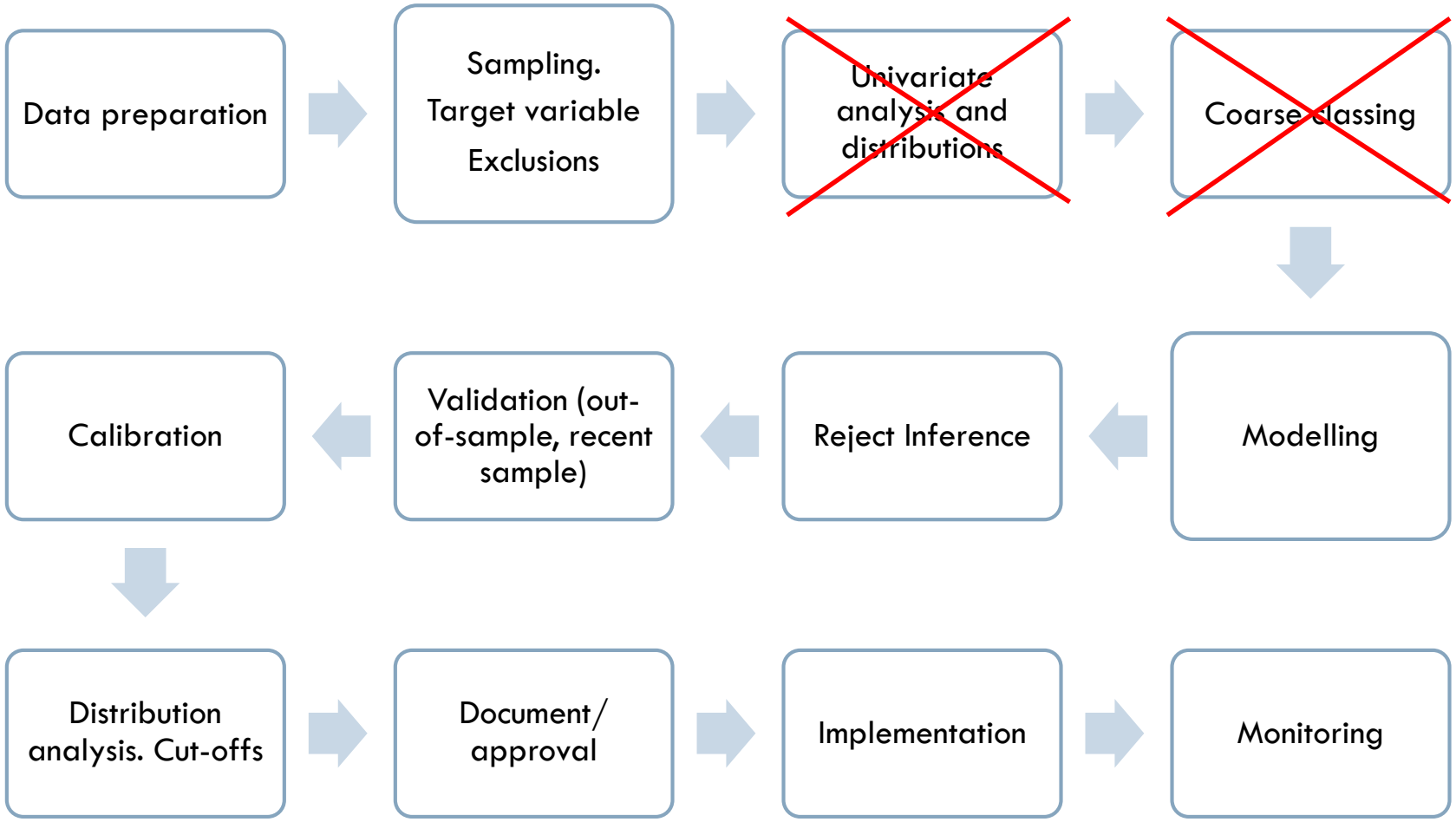
Boosting - Boosting is a machine learning ensemble meta-algorithm for primarily reducing bias, and also variance in supervised learning, and a family of machine learning algorithms that convert weak learners to strong ones. (https://en.wikipedia.org/wiki/Boosting_(machine_learning))
Boosting is so called 'greedy algorithm'.



$$H(x) = sign(\alpha_1 h_1(x) + \alpha_2 h_2(x) + \alpha_3 h_3(x))$$



Development Process



R Code - XGBoost


```
params_xgb <- expand.grid(
  # we would change eta and max_depth params
  eta = 0.1, #c(1,0.5, 0.1, 0.01),
  max_depth = c(1,2,3),
  # other parameters are fixed
  booster = 'gbtree',
  objective = 'binary:logistic',
  min_child_weight = 100,
  subsample = 1,
  colsample_bytree = .2,
  #scale_pos_weight = 1.0,
  stringsAsFactors = FALSE
)
mbst <- xgb.train(params = params_xgb[1,],
                  data = Xtr,
                  eval_metric = 'auc',
                  nround = 100,
                  watchlist = list(test = Xte, #oot = Xoo,
                                   train = Xtr),
                  print.every.n = 20,
                  early.stop.round = 50,
                  prediction = TRUE,
                  maximize = TRUE)

mcv <- xgb.cv(params = params_xgb[1,],
                  data = Xtr,
                  eval_metric = 'auc',
                  nround = 100,
                  nfold = 5,
                  watchlist = list(test = Xte, #oot = Xoo,
                                   train = Xtr),
                  print.every.n = 20,
                  early.stop.round = 50,
                  prediction = FALSE,
                  maximize = TRUE)
```

Model validation results

Model	Train AUC	OOT AUC
Logistic regression	0.69	0.67
Boosting – 1st iteration: set of XGB control parameters 1	0.75	**0.65**
Boosting – 2nd iteration: set of XGB control parameters 2	0.73	**0.71**

Non-linear algorithms is better than linear models in sense of **fitting accuracy** – we can extract the dependencies which are 'hidden' for linear models such as logistic regression

BUT – High Train Performance may cause Low Test Performance

The solution should be balanced: Performance VS Stability

Resume

- The use of **specific features for online lending** can increase predictive power of the scoring models, approximately, **PLUS 10-20% Gini**, and even more
- However, we may face the legal problems due to privacy and typical problem with Big Data storage and processing

- **Machine Learning** non-linear algorithms can add **PLUS 10% Gini and more**
- However, we may face the following problems with ML:
  - **Overfitting** (a model fits closely a particular data set, but may fail to predict other data)

    Solution: use Out-of-Time validation and Cross-validation
  - **Complex technical implementation**

    Solution: e.g. use R or Python procedures with libraries on server
  - **Complex support** of the Machine Learning models

- However, if the problem can be solved by money, it's not a problem, it's the costs... ☺

Thank you for your attention!

QUESTIONS?

Denis Osipenko, PhD

Denis.Osipenko@gmail.com

www.profitscoring.com