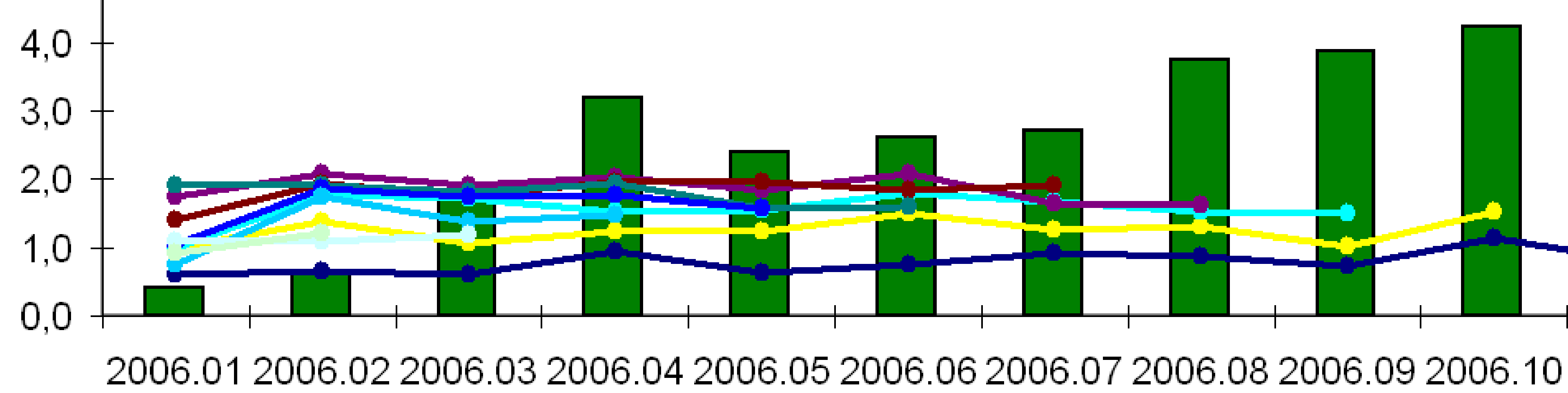


Credit Scoring - business process automation

© dr Karol Przanowski

(c) Copyrights

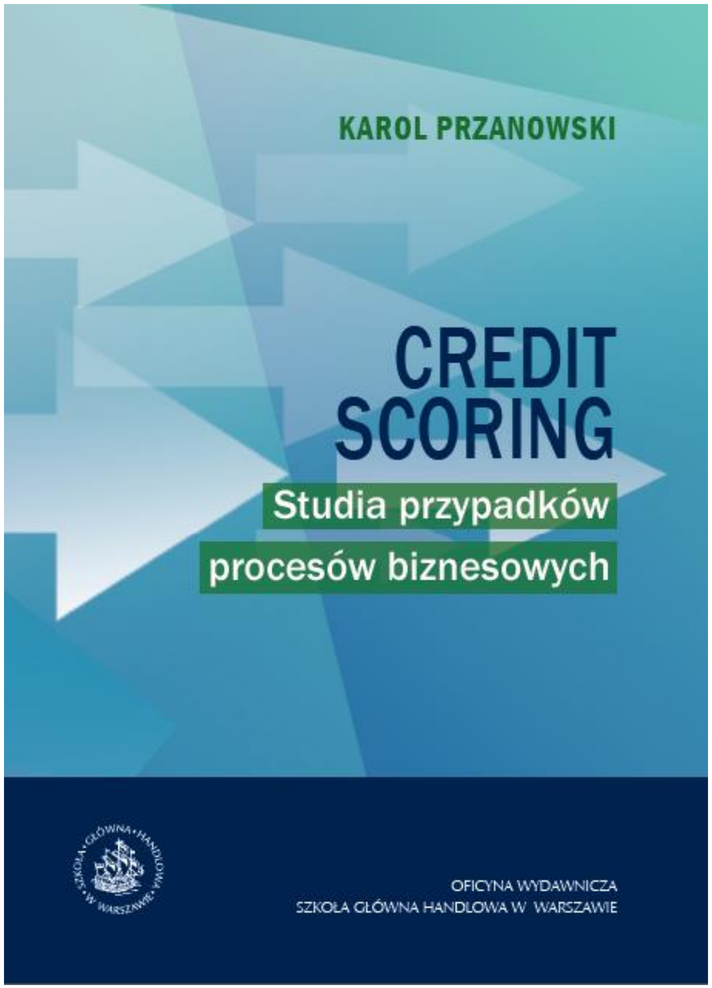
- All SAS and Python codes are made and has a copyrights by trainer
- Can be used in educational and science purpose providing a proper reference
- Usage into business purpose only with agreement with the author

Book in PDF, only in Polish

http://www.wydawnictwo.sgh.waw.pl/produkty/profilProduktu/id/723//CREDIT_SCORING_W_ERZE_BIG-DATA_Karol_Przanowski/

http://administracja.sgh.waw.pl/pl/OW/publikacje/Documents/ostateczny_CreditScoring_KPrzanowski.pdf

Book in PDF, only in Polish



The presented business models of profitability and usability of predictive models in:

- Acceptance of cash loans
- Acceptance of the complex process: acquisition and cross-selling
- Acceptance of mortgage loans
- Amicable debt collection
- Managing BTL campaigns
- Counteracting customer churn

Enclosed Excel files with rules and practical indicators http://administracja.sgh.waw.pl/pl/OW/publikacje/Strony/2015.aspx

In English

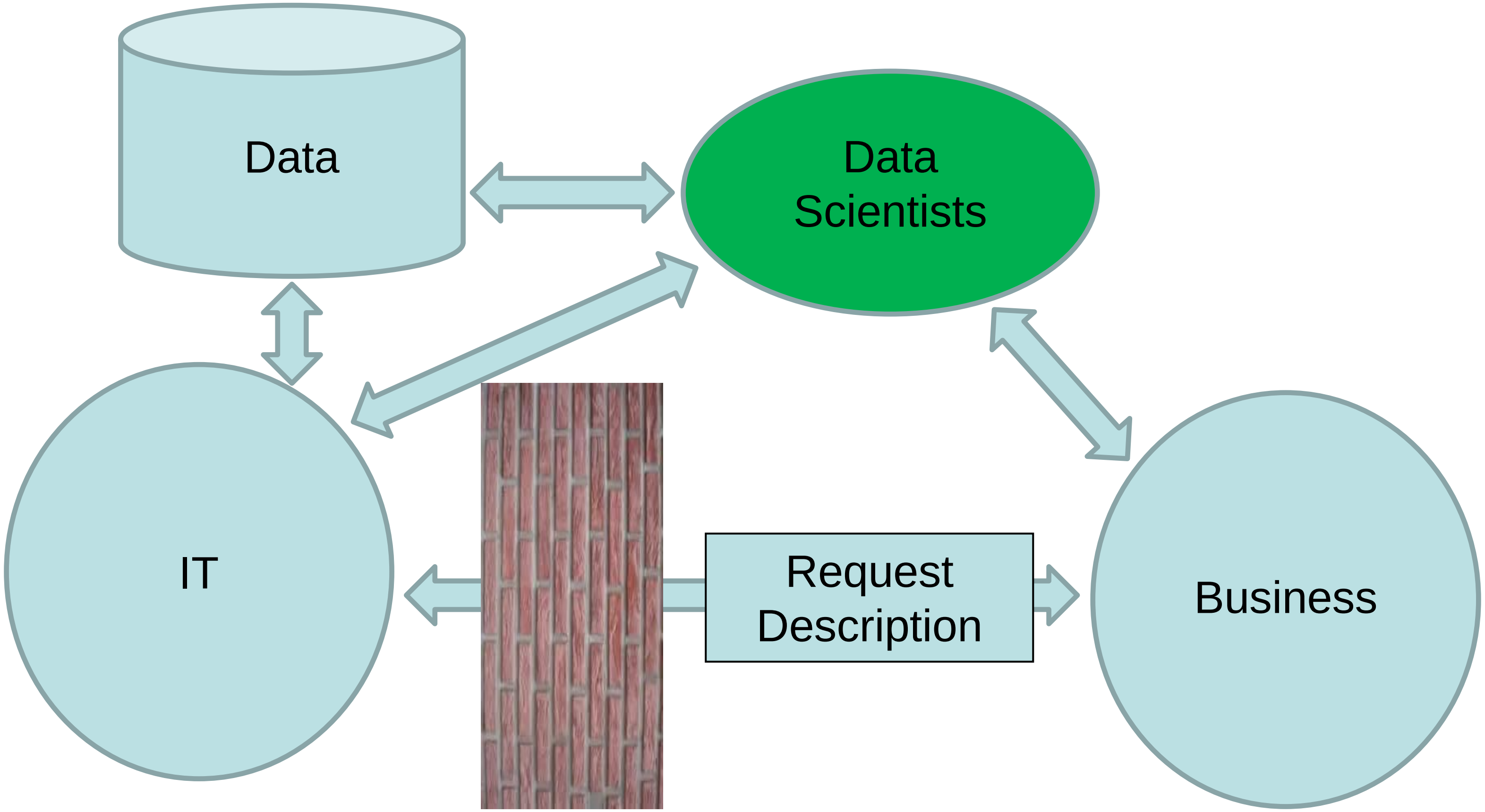
- **Karol Przanowski, Credit acceptance process strategy case studies - the power of Credit Scoring - https://arxiv.org/abs/1403.6531**
- Karol Przanowski, Consumer finance data generator - a new approach to Credit Scoring technique comparison - https://arxiv.org/abs/1210.0057
- Karol Przanowski, Banking retail consumer finance data generator - credit scoring data repository, e–FINANSE, 9(1), pp. 44–59, 2013

Data Scientists - competences

- Data processing (programming):
  - C++, Java, Phyton, Perl, R, SAS 4GL, Julia
- Systems:
  - Oracle, Teradata, SAS, Hadoop, Cloud
- Statistics and Data Mining:
  - Logistic regression, tree decisions, neural networks, random forests, cluster analysis, survival analysis, CLTV models
- Text Mining

The role of Data Scientists

- Middleman, connector, between IT and business



New paradigms

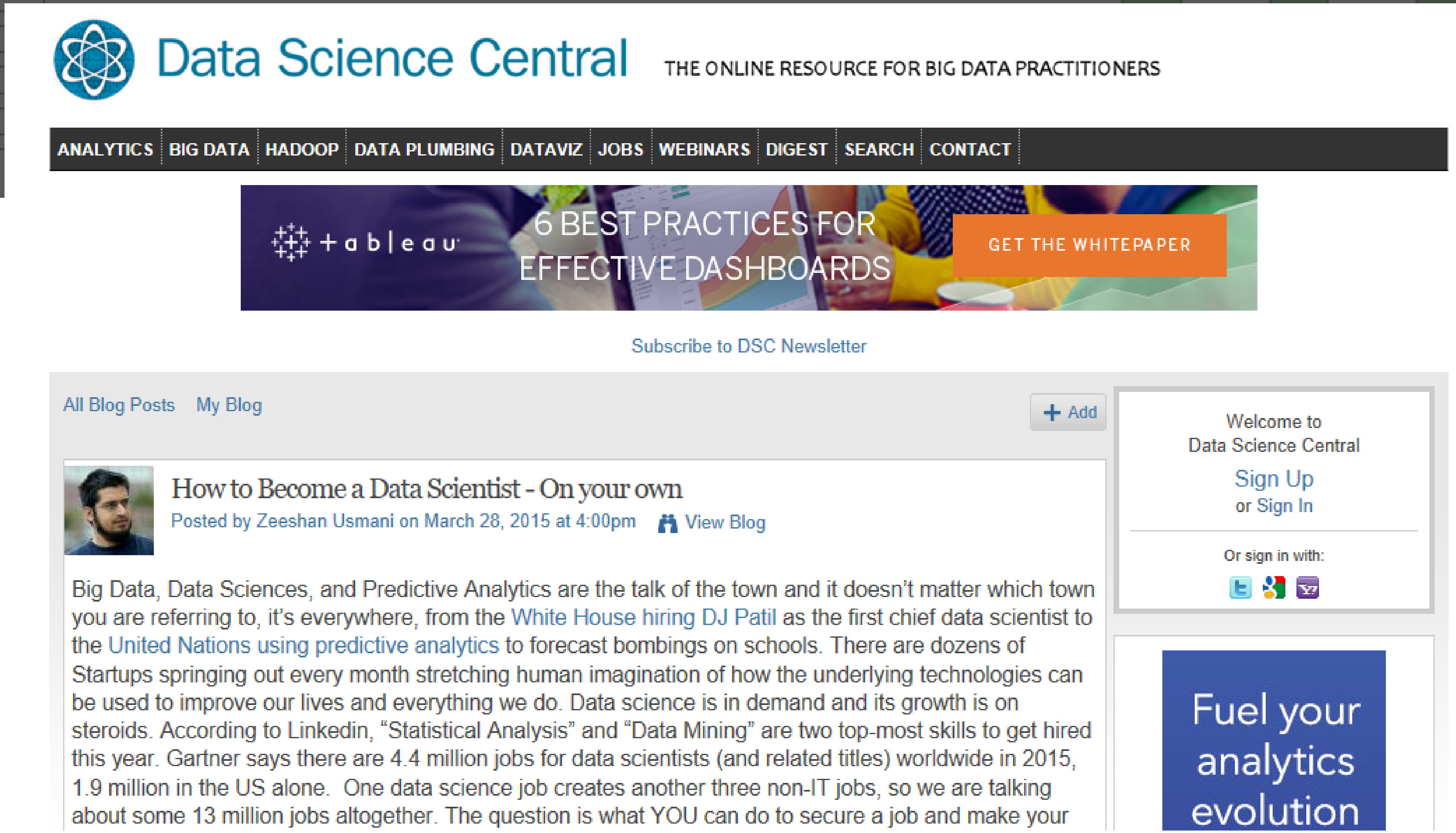
- DWH (Datawarehouse):
  - Clean and then loan (old)
  - Load and then have a troubles (new)
- Modelling (forecasting, predicting):
  - Find the real reason, observe important factors (old)
  - Verify what have already collected data could influence on modeled event, accept relations coming from derivatives, not from sources (new)

Data quality

- What should be collected, corrected?
- Where and how data should be used?
- Measures of quality:
  - Completeness
  - Accuracy
  - Consistency
  - Integrity
  - Utility
  - Intelligibility

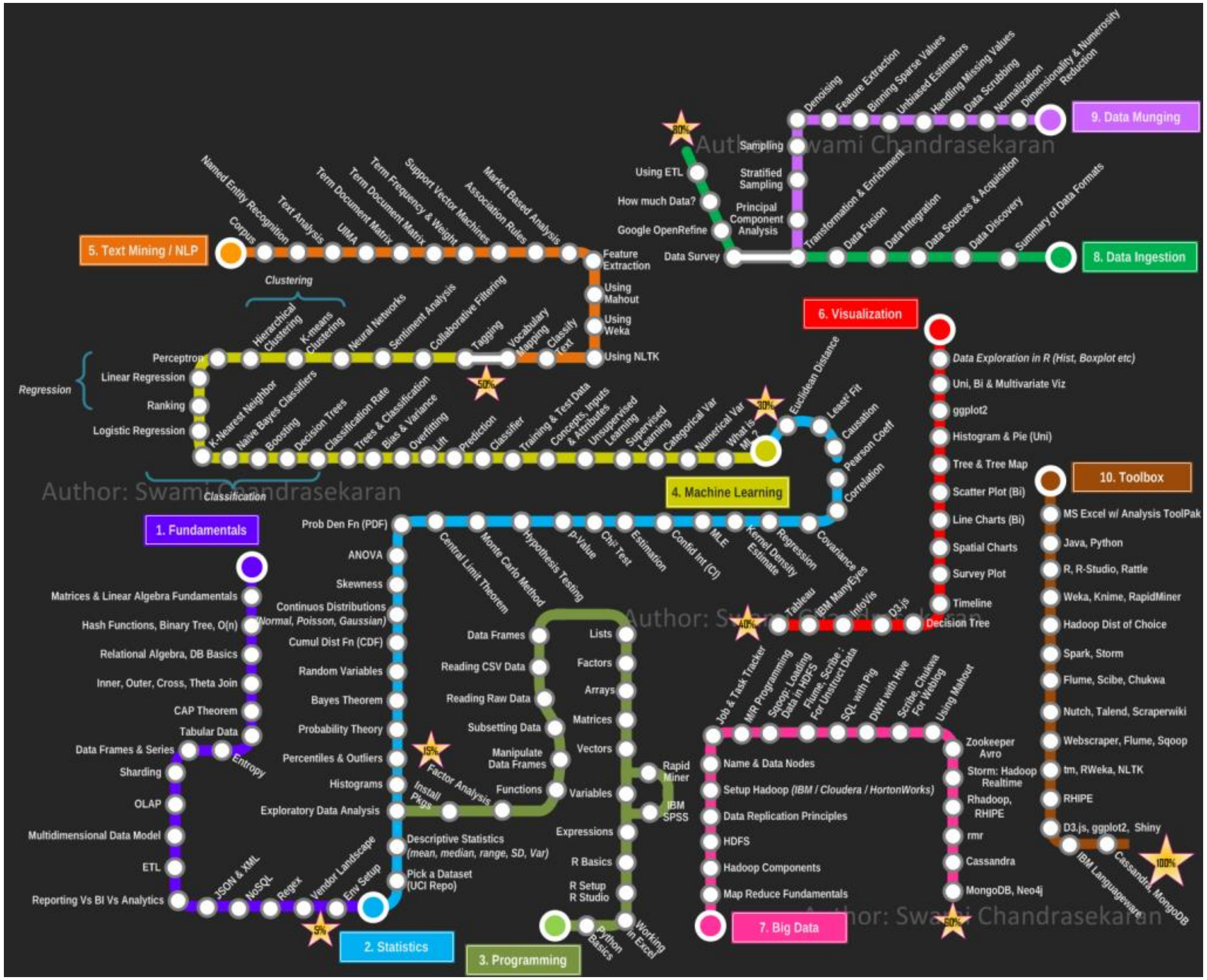
Big Data fails

- Lack of good *business cases*
- Data are collected but nobody knows where and why we need it
- Underestimated quality of data problems
- Omitted problem of biased estimation
- Too strong focus on only technology, IT
- Naïve hope of user friendly software, a few clicks
- Lack of good trainings for data scientists
- Lack of public interesting data, with enough number of rows and columns



http://www.datasciencecentral.com/profiles/blogs/how-to-become-a-data-scientist-for-free

Data Scientist Metro Map



http://nirvacana.com/thoughts/becoming-a-data-scientist/

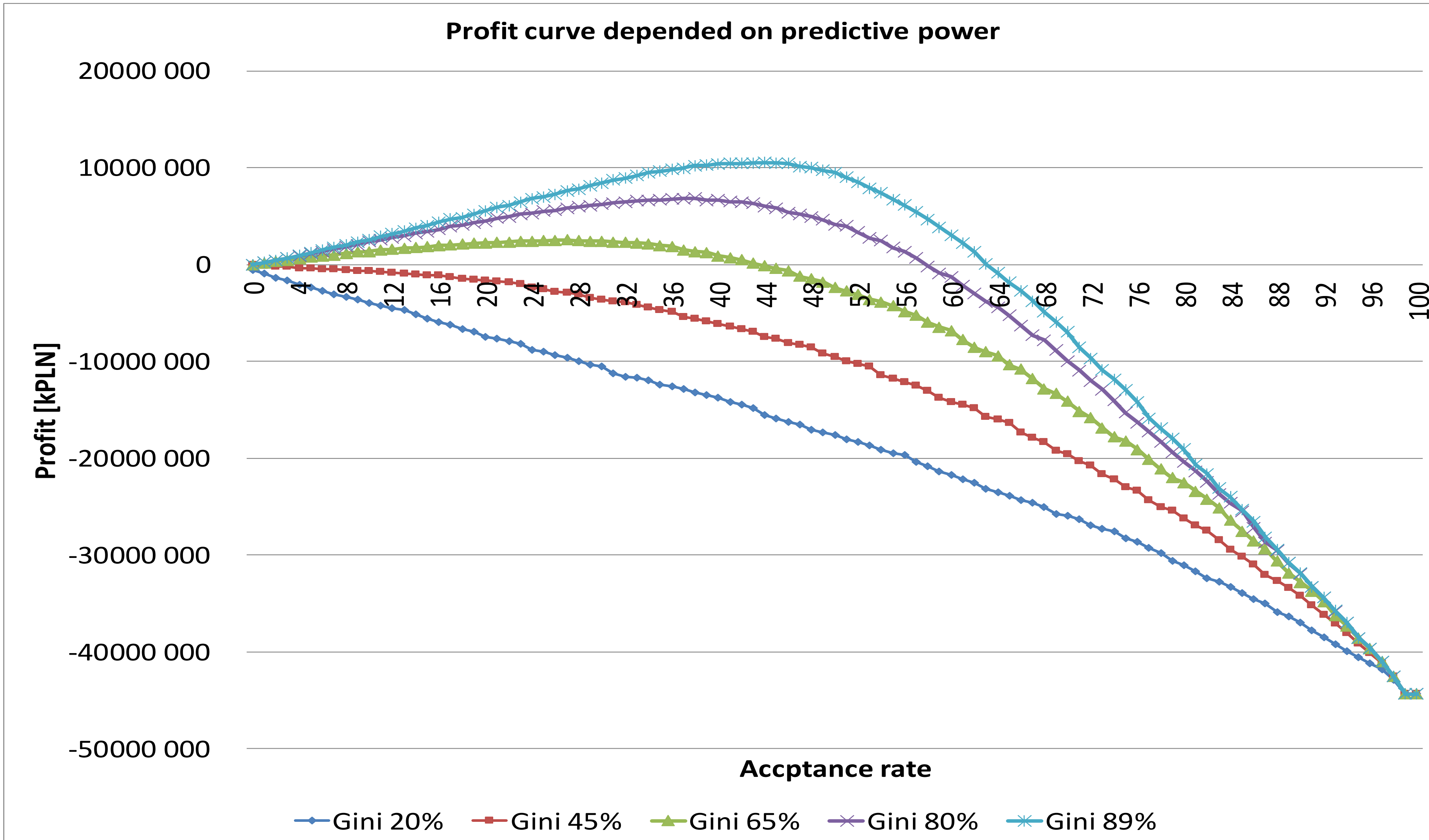
Statistics

- Repeatable and massive events
- Trend and property indication, discovery
- Population research
- Relation analysis
- Forecasting and predictive analysis
- Stability testing
- Not one event but a few thousands

Event prediction

- New purchase
- Conversion into new product
- Instalment or credit payment
- Attrition, Churn
- Fraud, cheater, scam (AML)
- Not legal usage of electric service
- Accident, emergency event

Why are we able to earn money?



How main factors can be calculated?

$$L_i = \begin{cases} 50\% A_i, & \text{when} \quad \mathrm{default}_{12} = \text{Bad} \\ 0, & \text{when} \quad \mathrm{default}_{12} \neq \text{Bad} \end{cases}$$

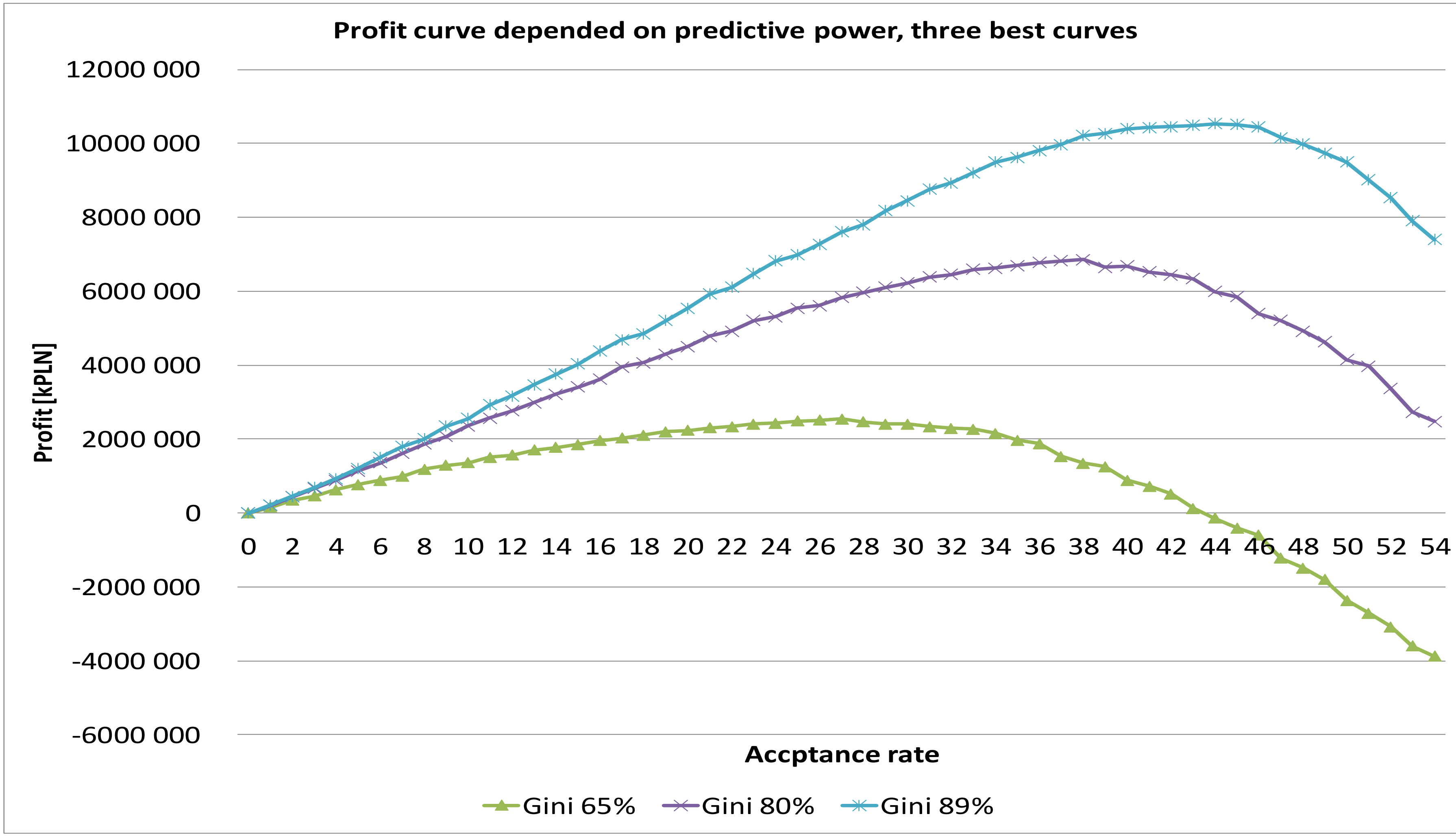
$$I_i = \begin{cases} A_i p, & \text{when} \quad \mathrm{default}_{12} = \text{Bad} \\ A_i(N_i r \frac{(1+r)^{N_i}}{(1+r)^{N_i}-1} + (p-1)), & \text{when} \quad \mathrm{default}_{12} \neq \text{Bad} \end{cases}$$

Total profit can be calculated as follows:

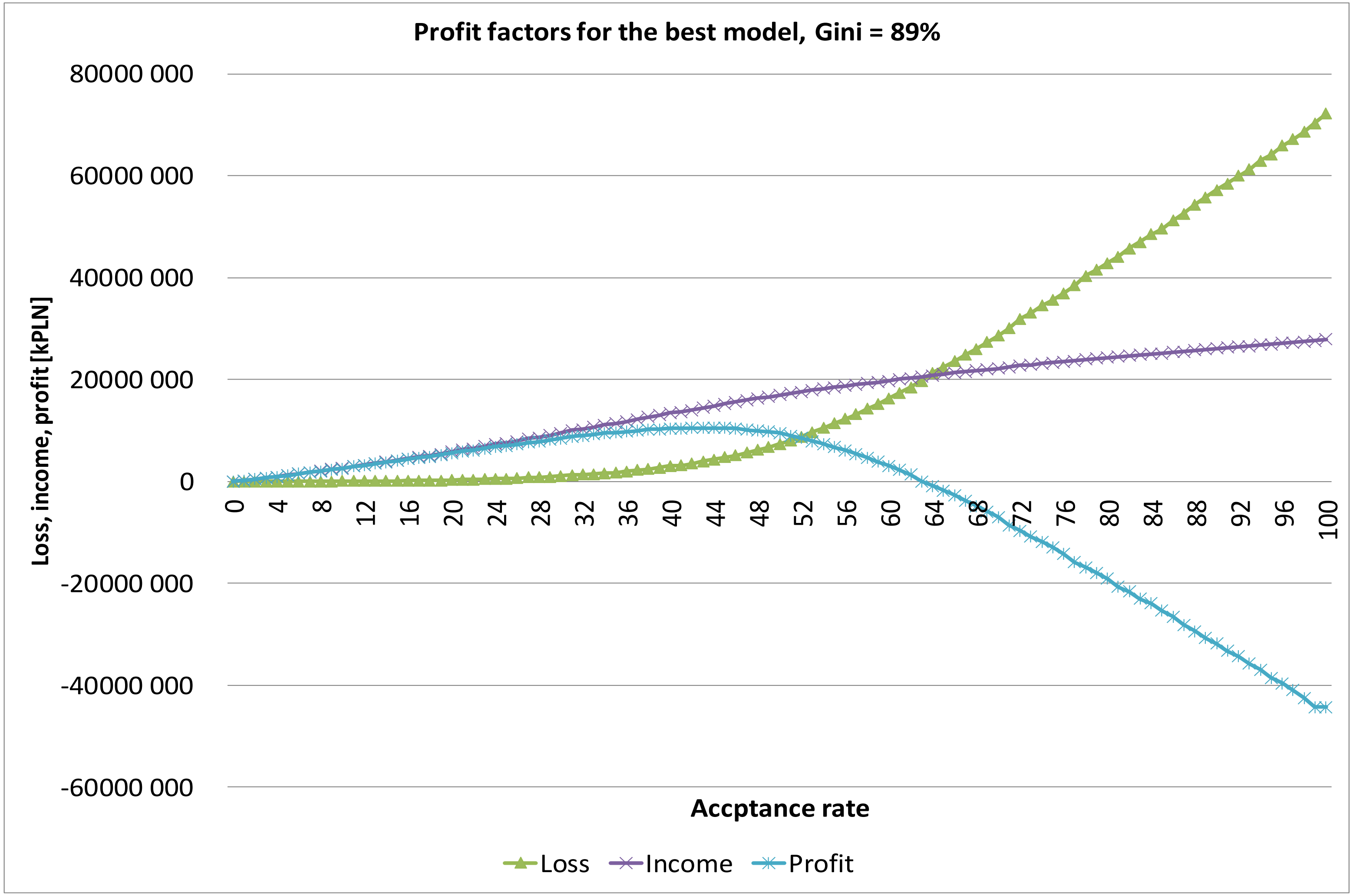
$$P = \sum_i I_i - L_i. \tag{4.1}$$

- EL = PD*LGD*EAD

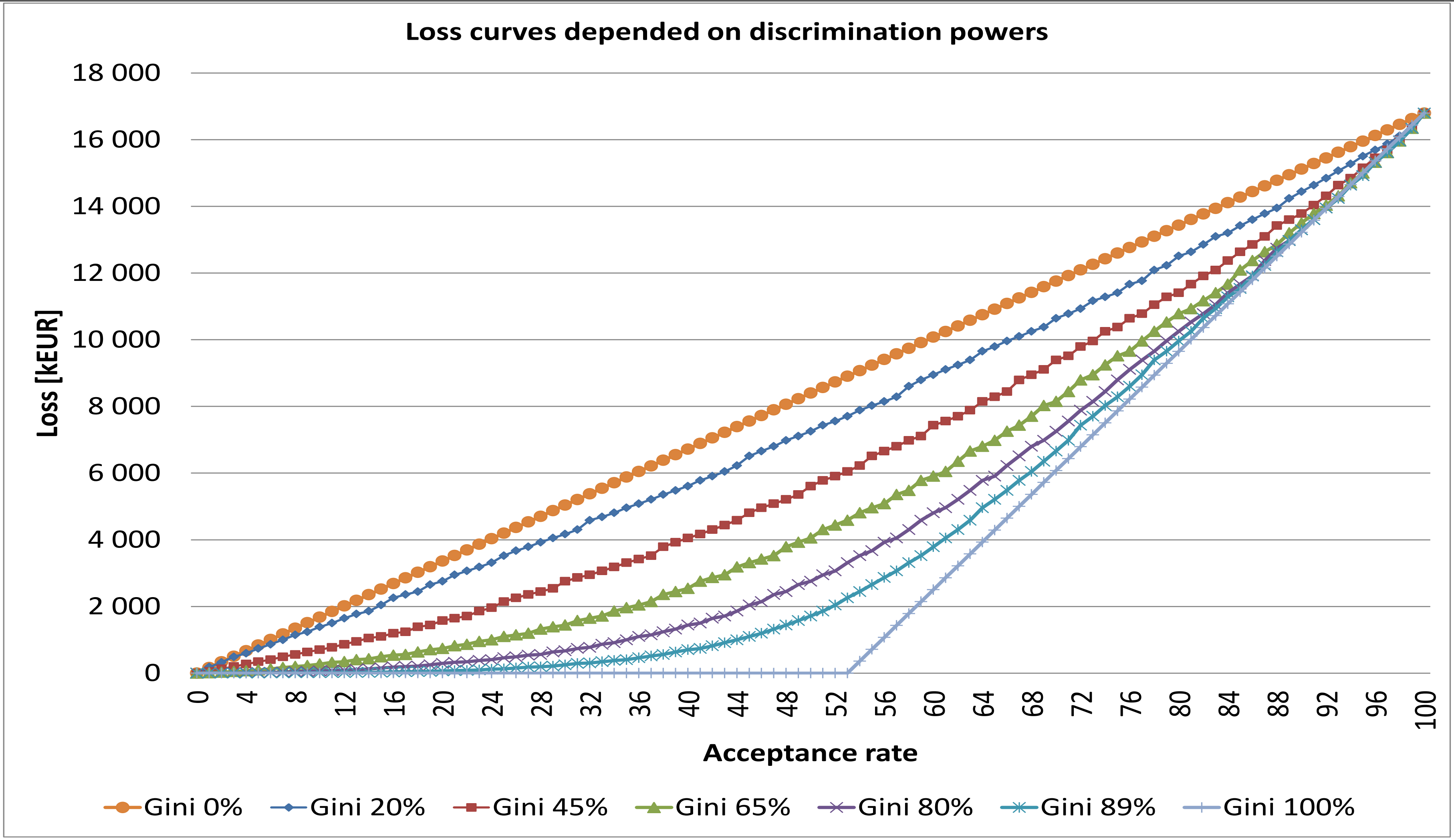
Why are we able to earn money?



Factors of profit



Loss curves



Loss curves depended on discrimination powers
18 000
16 000
14 000
12 000
10 000
8 000
6 000
4 000
2 000
0
Loss [kEUR]
0
4
8
12
16
20
24
28
32
36
40
44
48
52
56
60
64
68
72
76
80
84
88
92
96
100
Acceptance rate
Gini 0%
Gini 20%
Gini 45%
Gini 65%
Gini 80%
Gini 89%
Gini 100%

Impact on financial results

Number of applications per month	50 000
Average loan amount	1.1 kEUR
Average number of instalments	36 months
Annual percentage rate	12%
Provision for loan granting	6%
Global portfolio risk	47%
Increase of predictive power	5%
Increase of acceptance rate	3,5%
Increase of monthly profit	**350 kEUR**
Decrease of monthly loss (AR=20%)	210 kEUR
Decrease of monthly loss (AR=40%)	350 kEUR

Business case in Excel

Credit acceptance process (8)

Conclusions

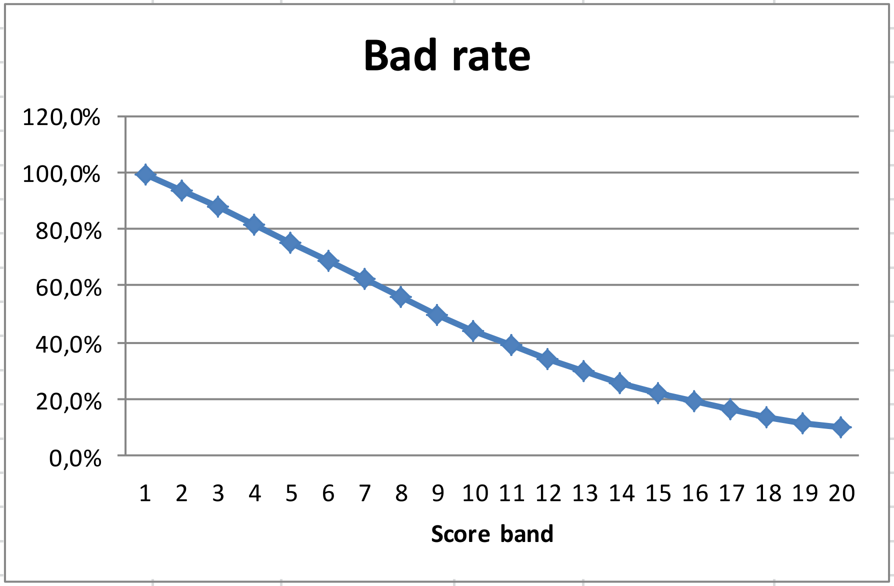
- If delta Gini is increased by 5% then the delta profit of the process can be increased monthly by about 350 kEUR and acceptance rate by 3,5%.
- In the different way, when the increase of acceptance rate is not needed, then bank can save money only by use better scoring model. Namely with acceptance rate on the level 20% the loss can be saved monthly about 210 kEUR. In case 40% of acceptance rate can be saved about 350 kEUR monthly.
- Scoring models are not only a tool to satisfy regulator recommendations, but there are the best tool to earn big money.
- Mentioned above numbers, profit amounts or saved losses persuade to keep and care about analytical teams in our companies and moreover suggest to always try to build better models, always to test a new one, to have always some champion challengers and some parallels acceptance scenarios. Also it is the reason why all analysts should develop their skills, make brain storms, knowledge share to be always on top to cut the edge, because better model means more money and guarantees a better position on the market, to win with competitors.

Business case in Excel

Credit acceptance process (7)

The same exercise in Excel file

Number of applications per month	50 000
Average loan amount [EUR]	1 100
Average number of installments	36
Annual percentage rate (or net margin)	12%
LGD (Loss Given Default)	50%
Provision charged on disbursement day	6%
Gini global	65,54%
Gini on accepted	24,50%
Global risk in market (default12)	47%
Accepted risk	18,49%
Acceptance rate	40,00%
Global loss	12 925 000
Global income	7 454 097
Global profit	-5 470 903
Accepted loss	2 034 280
Accepted income	4 585 341
Accepted profit	2 551 061



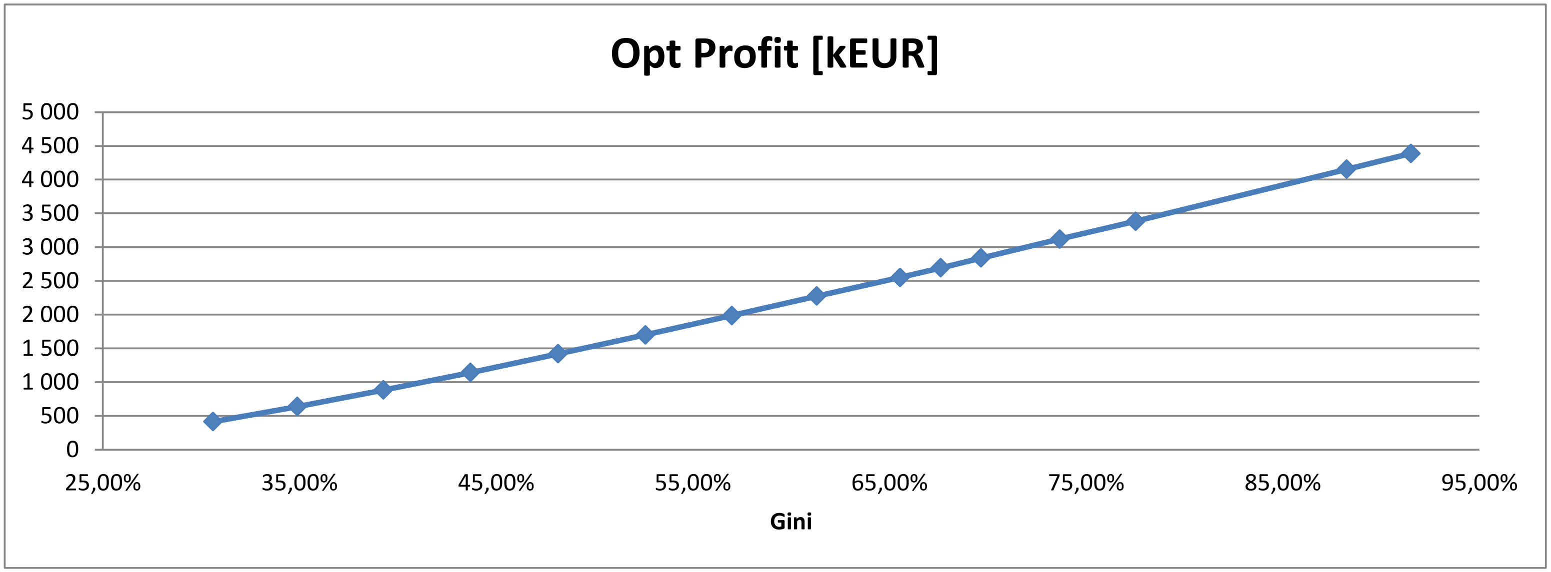
charge provision	66,00
income from interest rates	215,29

http://administracja.sgh.waw.pl/pl/OW/publikacje/Strony/2015.aspx

Business case in Excel

Credit acceptance process (7)

The same exercise in Excel file



1%	Delta Gini	65 219	EUR
5%	Delta Gini	326 096	EUR
10%	Delta Gini	652 192	EUR

Credit risk management

- The entire approval process affects the credit risk! (from the first word with the client to the last contact with him)
- What is the phenomenon of negative selection?
- What is the impact of Risk Based Pricing on the bottom line (financial result)?
- Is risk managed by reducing the numerator?
- How does sales affect credit risk?
- Can Sales and Risk understand each other?
- Can you reduce your credit risk and increase your sales?
- The Credit Risk Director must be a friend of the Sales Director and vice versa !!!

Stages of building scorecard

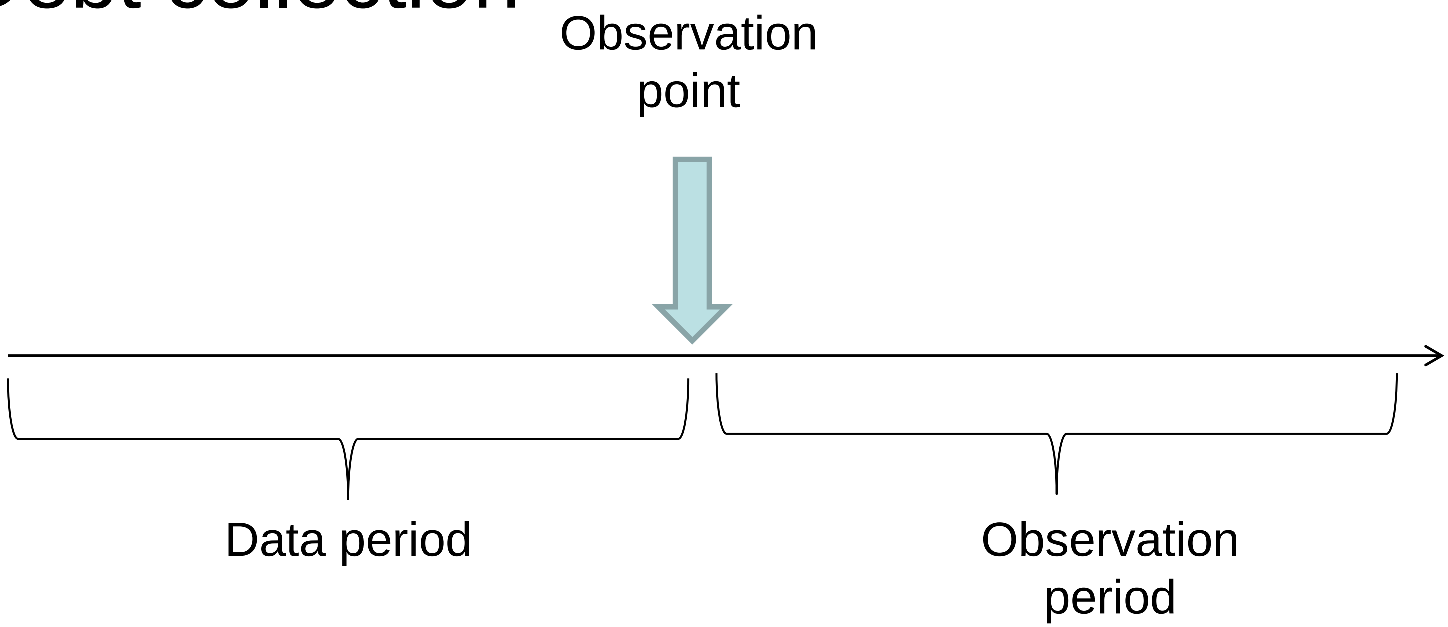
1. Data structure
2. ABT variables
3. Data partition
4. Variable scale
5. Defining default
6. Binning, variables’ categorization
7. Variables’ pre-selection
8. Variables’ reports and visualization
9. Multidimensional variables’ selection and model evaluation
10. Manual remedies and corrections
11. Monitoring and model documentation
12. Scoring code

Default definition

- Every account is tested in 3, 6, 9 and 12 months after loan granting.
- We calculate a MAX of number of past due installments, then we can define default statuses:
  – Good – MAX<=1 or it is paid
  – Bad – MAX>3
  – Indeterminate = the rest of possibility
- Sometimes: Dormant and balance condition

Various portfolios

- Application
- Behavioral
- Debt collection

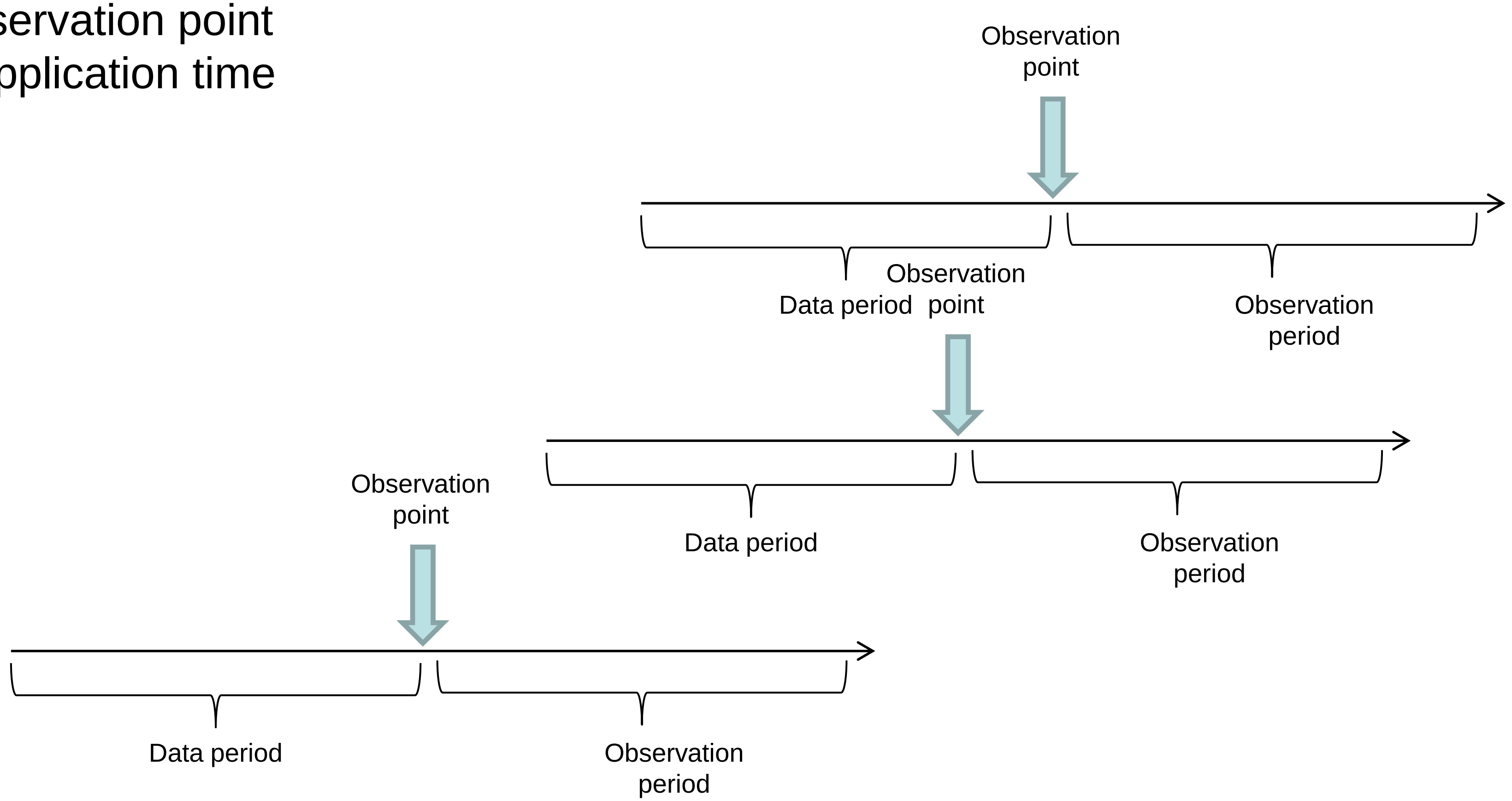


Important problems

- Length of observation period
- Length of data period
  - How do create variables?
  - Evolutions in the time (TTC)
  - Only one time stamp (PIT)
  - As a percentage (relative value)
    - comparison to some gold example
    - Always as a ratio
  - As an absolute value
  - With information included all history

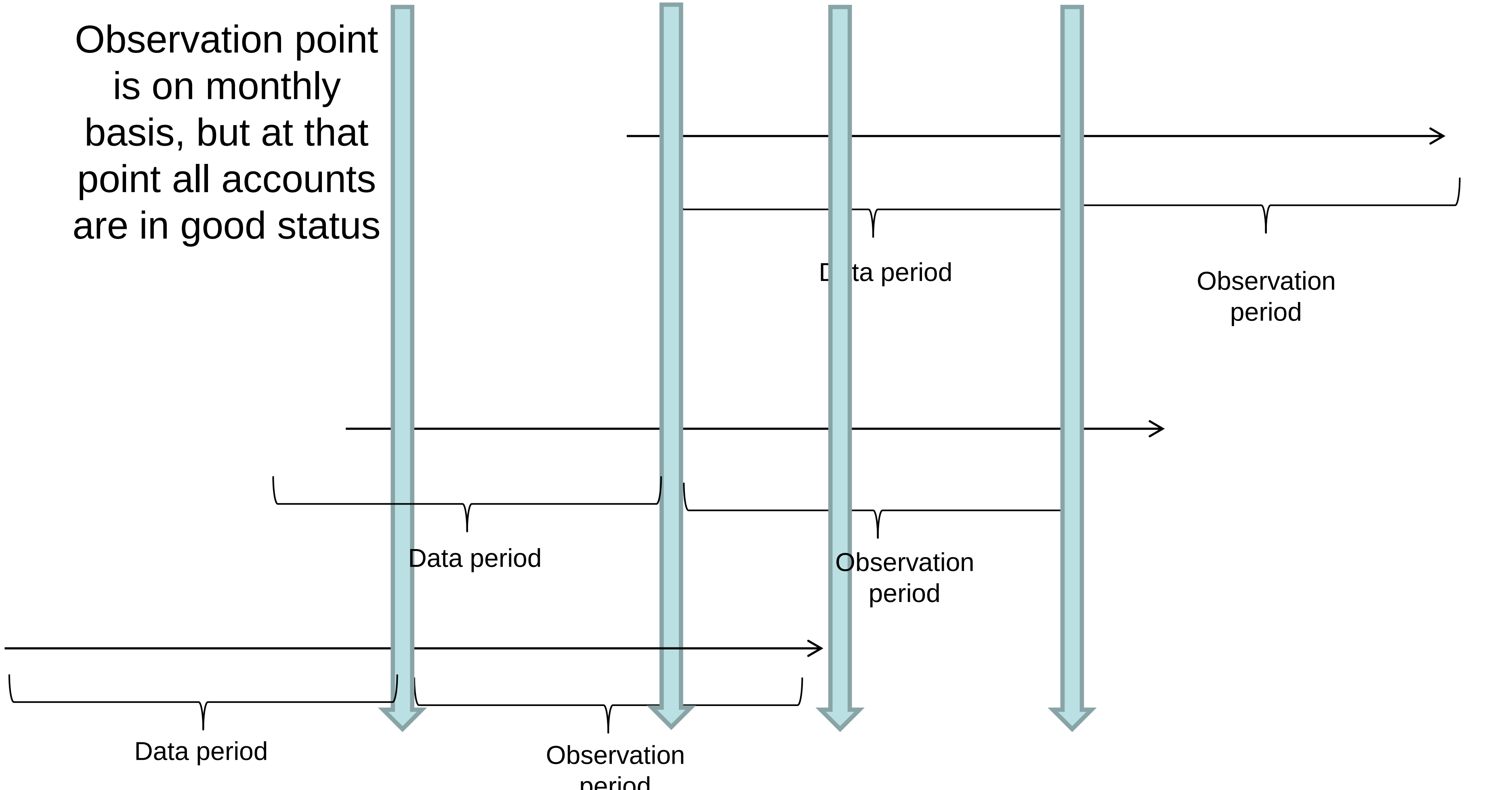
Application (account only ones)

Observation point
is application time



Behavioral (account many times)

Observation point is on monthly basis, but at that point all accounts are in good status



ABT – Analytical Base Table

- One row is the object of modelling, an account, a customer?
- Goal function: 1 – bad, 0 – good, .i – indeterminate, .d – dormant
- Naming: ags3_Min_CMaxI_Due – we count the maximum number of due installments for a given customer on all his installment loans, then we count the minimum value in the last 3 months
- Excel the list of variables
- SAS code abt_behavioral_columns.sas

Data structure

id client or account	default	period	var1	var2	var3
	0	200901			
	1	200902			
	.i	200901			
	.d				

Good

Bad

Indeterminate

Dormant

Scorecard - example

Category	Variable	Partial Score
<20	AGE	10
20>= and <34		20
35>=		30
Bad	Payment history	10
Not good		25
Good		40

- Who is the best customer?
- What variable is the most important?

First steps with ASB

- Main code:
  - SAS: main.sas,
  - Python: ASB_step_by_step.ipynb
- Options, macro-variables
- Batch processing
- Layout of directories and libraries
- Additional variables
- Interaction variables

Data subset

- SAS:
  - where '197501'<=period<='198712' and product='css' and decision='A';
- Python:
  - df=df[('197501'<=df['period']) & (df['period']<='198712') & (df['product']=='css') & (df['decision']=='A')]

Data partition

- Splitting into 2 data sets: train and valid
- Time sampling ↔ Random sampling
- Through the cycle ↔ Point in time

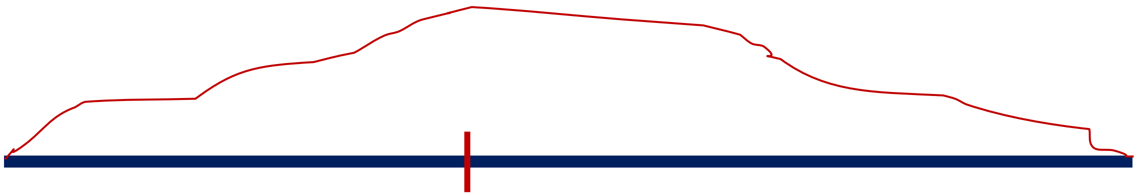
period	train	valid
200801		
200802		
200803		
200804		
200805		
200806		
200807		
200808		
200809		
200810		
200811		
200812		
200901		
200902		
200903		
200904		
200905		
200906		

period	
200801	
200802	
200803	
200804	
200805	
200806	
200807	
200808	
200809	
200810	
200811	
200812	
200901	
200902	
200903	
200904	
200905	
200906	

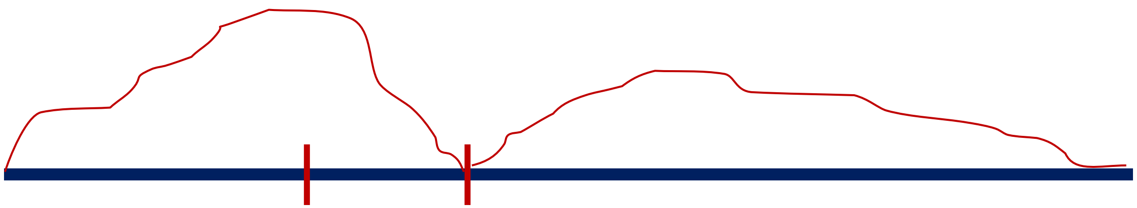
Data partition

- SAS:
  - %include "&dir_codes.train_valid.sas" / source2;
  - uncomment line: /*   agr: ags:*/
- Python:
  - #Splitting for train and test datasets
  - Uncomment line: # vars=[var for var in list(df) if var[0:3].lower() in ['app','act','agr','ags']]

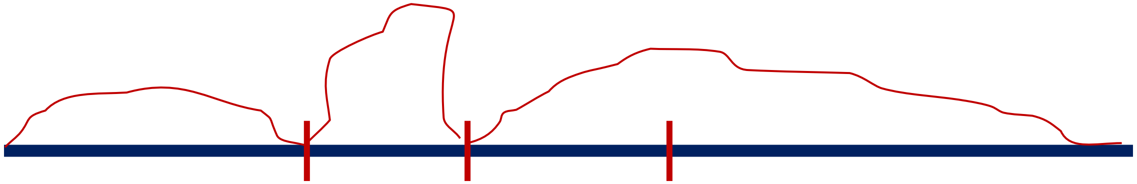
Binning of continuous variables



First point of splitting



Second point



Third in the widest partition

Binning of continuous variables

$$h_a = -\left[\frac{b_a}{s_a}\log_2\left(\frac{b_a}{s_a}\right) + \frac{g_a}{s_a}\log_2\left(\frac{g_a}{s_a}\right)\right]$$

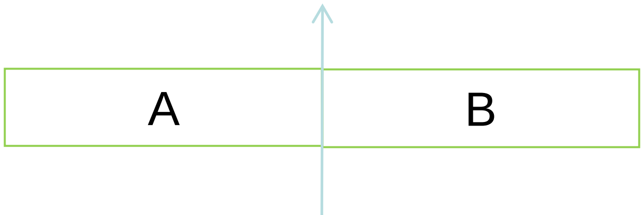
$$h_b = -\left[\frac{b_b}{s_b}\log_2\left(\frac{b_b}{s_b}\right) + \frac{g_b}{s_b}\log_2\left(\frac{g_b}{s_b}\right)\right]$$

$$h = -\left[\frac{b}{s}\log_2\left(\frac{b}{s}\right) + \frac{g}{s}\log_2\left(\frac{g}{s}\right)\right] - \frac{s_a}{s}h_a - \frac{s_b}{s}h_b$$

$$g_a = 1 - \frac{b_a^2 + g_a^2}{s_a^2}$$

$$g_b = 1 - \frac{b_b^2 + g_b^2}{s_b^2}$$

$$g = 1 - \frac{b^2 + g^2}{s^2} - g_a\frac{s_a}{s} - g_b\frac{s_b}{s}$$



b_a – number of bads in A
g_a – number of goods in A
s_a – number of all in A
b_b, g_b, s_b – similar for B
b, g, s – similar for all

Binning of continuous variables

- SAS:
  - %let max_n_splitting_points=5;
  - /*Minimal share of category*/
  - %let min_percent=3;
  - %include "&dir_codes.tree.sas" / source2;
- Python:
  - #Bining for numerical variables

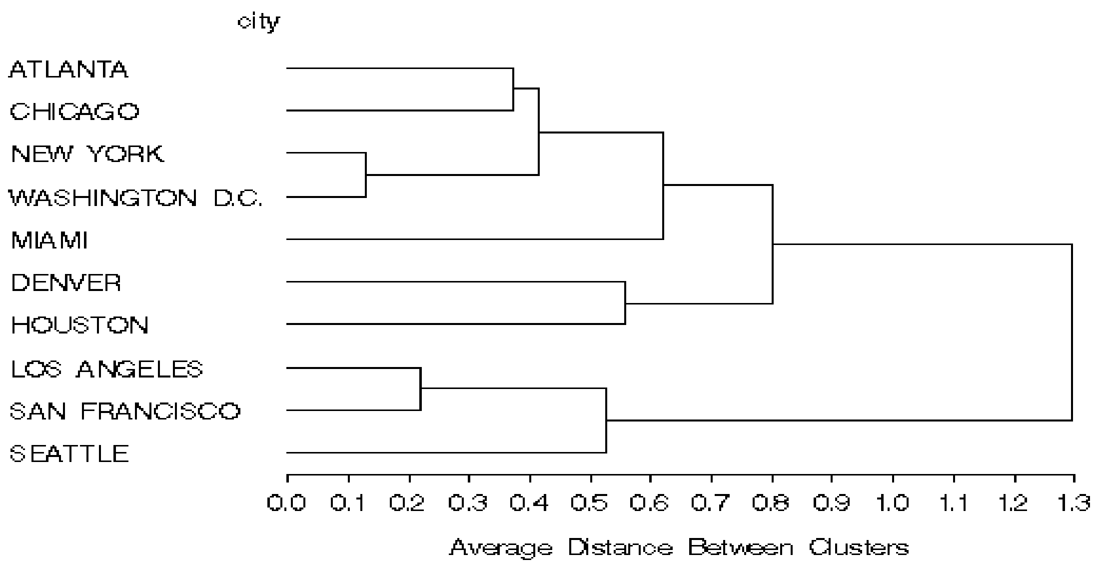
Binning of continuous variables

- Monotonic
- Maximizing Gini
- Constant width or shares
- Generally, one may use different decision trees algorithms

Number	Variable name	Gini_before	Gini_NonMon	Gini_MonNew	Gini_MonOld
1	AGGR6_MEAN_S_CASHUTL_EM	64,72%	63,31%	7,57%	63,04%
2	AGSP6_MAX_BAL_EMCL	60,09%	60,02%	.	59,65%
3	ACT_S_CASHUTL_EM	58,38%	60,03%	20,71%	59,81%
4	AGGR3_MEAN_S_RBAL_EMCL	38,44%	36,11%	39,35%	36,11%
5	AGSP3_MIN_PMT	29,33%	40,36%	31,39%	36,87%
6	AGSP6_MAX_NOTPAID	28,32%	17,85%	17,85%	.
7	ACT_PMT	27,59%	43,33%	32,33%	41,03%
8	ACT_S_RBAL_EM	26,41%	.	26,00%	.
9	AGGR6_MAX_CYCLE_DD	14,36%	7,58%	7,58%	7,58%
10	AGSP3_MAX_PMT	8,88%	.	19,71%	24,65%
11	ACT_NBR_LCF	1,75%	27,51%	27,54%	.

Nominal variables

- The condition for representiveness level:
  - Share of category >= 1% or 3%
- Combine categories by cluster analysis methods based on similar bad rate statistics (proc cluster)



Binning of nominal variables

- SAS:
  - %let max_n_splitting_points=5;
  - /*Minimal share of category*/
  - %let min_percent=3;
  - %include "&dir_codes.bining_nominal.sas" / source2;
  - %include "&dir_codes.bining_nominal_without_joining.sas" / source2;
- Python:
  - #Bining for character variables

Variable preselection

- For every variable are calculated statistics:
  - Quality
  - Descriptive
  - Predictiveness
  - Stability
- SAS:
  - Output: variable_stat – wiele statystyk zmiennych
  - %include "&dir_codes.variable_pre_selection_1step.sas" / source2;
  - %include "&dir_codes.variable_pre_selection_full.sas" / source2;
- Python:
  - Output: Gini_vars.xlsx, Variable_report.xlsx
  - #Calculating Gini values for features

Variable preselection

- Stability statistics:
  - IS (PSI) – index stability,
  - KS - Kolmogorov-Smirnov ,
  - KL - Kullback-Leibler distance,
  - AR_Diff (Delta Gini) = abs (Gini Train – Gini Valid) / Gini Train
- Predictiveness statistics:
  - Gini train, valid
  - IV – information value

Variable preselection

$$\mathrm{IS} = \sum (t_i - v_i) \ln(\frac{t_i}{v_i}),$$

$$\mathrm{KL} = \sum t_i \ln(\frac{t_i}{v_i}),$$

$$\mathrm{IV} = \sum (g_i - b_i) \ln(\frac{g_i}{b_i}),$$

t_i, v_i - shares of *i*-th category in train, valid

g_i, b_i - shares of goods and bads

Preselection - benchmarks

- Acceptance criteria:
  - Gini > 5%
  - AR_diff < 5%, 20%
  - KL, IS (PSI) < 0.1, 0.5
  - KL, IS only for bads < 0.1

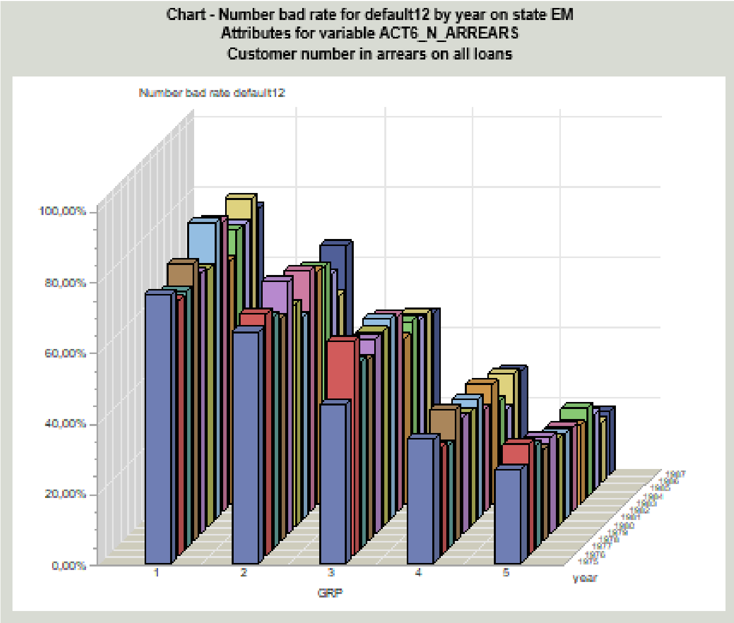
Preselection – data potential

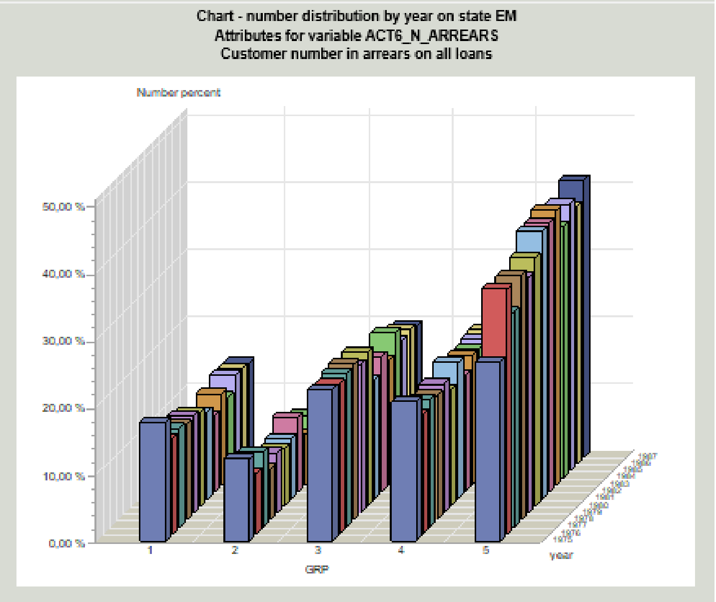
- SAS: out.Variables_stat_1step
- Python: Gini_vars.xlsx

	variable	ar_train
1	WOE_ACT6_N_ARREARS	48.44%
2	WOE_ACT3_N_ARREARS	48.06%
3	WOE_ACT9_N_ARREARS	47.73%
4	WOE_ACT_CCSS_DUEUTL	45.55%
5	WOE_ACT12_N_ARREARS	44.87%
6	WOE_ACT_CCSS_MAXDUE	44.17%
7	WOE_ACT_CCSS_UTL	43.14%
8	WOE_ACT_CCSS_N_LOANS_ACT	42.38%
9	WOE_ACT_CCSS_MIN_LNINST	36.79%
10	WOE_ACT_CCSS_MIN_PNINST	29.71%
11	WOE_ACT_CCSS_N_STATC	27.19%
12	WOE_ACT_CCSS_N_LOANS_HIST	25.62%
13	WOE_ACT_CCSS_SENIORITY	25.22%
14	WOE_ACT_CCSS_MIN_SENIORITY	25.15%

Variable reports

Attribute number	Condition	Bad rate (br)	Percent of population (%POP)
1	5 < ACT6_N_ARREARS	77,78%	13,01%
2	4 < ACT6_N_ARREARS <= 5	67,52%	9,27%
3	2 < ACT6_N_ARREARS <= 4	54,35%	21,01%
4	0 < ACT6_N_ARREARS <= 2	31,37%	19,56%
5	not missing(ACT6_N_ARREARS) and ACT6_N_ARREARS <= 0	23,59%	37,15%
			100,00%





Variable reports

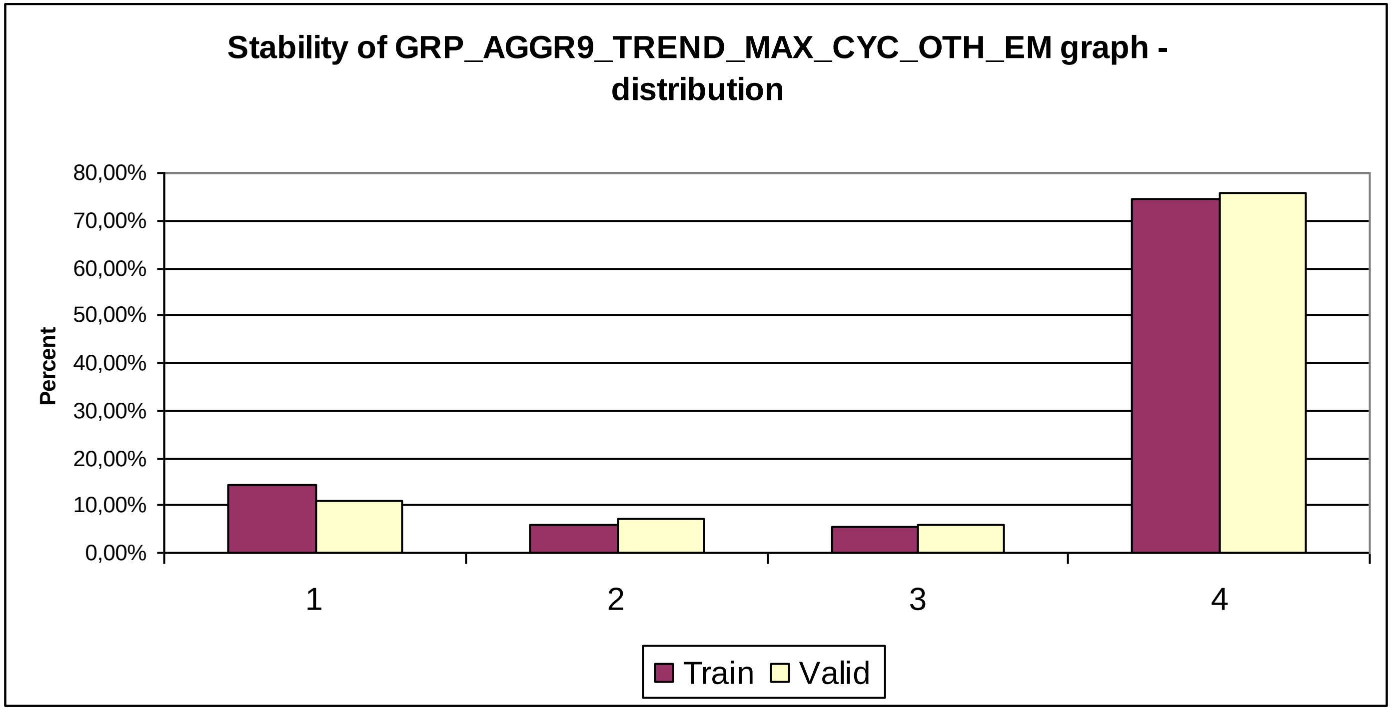
- SAS:
  - in html format - interactive
- Python:
  - In Excel
- Reports like:
  - Descriptive statistics
  - Categories measures
  - Bad rates and shares
  - Shares in time
  - Clusters of variables

Variable reports

- SAS:
  - %include "&dir_codes.variable_reports.sas" / source2;
- Python:
  - #Variable_reportRaporty:

Stability testing on data partition

- Statistics: H_GRP_TV and H_Br_GRP_TV



Variable clustering

- Variables grouped into clusters
- Every variable correlated with other from the same cluster
- Correlation between clusters is minimized
- Statistics like Cumulative proportion explains expected number of clusters

Obs	Number	Eigenvalue	Difference	Proportion	Cumulative
1	1	2,83858046	0,8983386	31,54%	31,54%
2	2	1,94024187	0,90106143	21,56%	53,10%
3	3	1,03918044	0,11832011	11,55%	64,64%
4	4	0,92086032	0,10011509	10,23%	74,88%
5	5	0,82074524	0,26466194	9,12%	84,00%
6	6	0,5560833	0,22605046	6,18%	90,17%
7	7	0,33003284	0,01355258	3,67%	93,84%
8	8	0,31648026	0,07868499	3,52%	97,36%
9	9	0,23779527	_	2,64%	100,00%

Classical logistic regression – model without transformation

$$\operatorname{logit}(p) = Age * \beta_1 + PaymentHistory * \beta_2$$

$$\operatorname{logit}(p) = \ln\left(\frac{p}{1-p}\right)$$

Transformation WOE

Attribute	Variable	Partial Score	Formula	
<20	Age	10	woe1	beta1
20>= and <34		20	woe2	
35>=		30	woe3	
Bad	Payment history	10	woe4	beta2
Not good		25	woe5	
Good		40	woe6	

Transformation Dummy

Attribute	Variable	Partial Score	Formula
<20	Age	10	beta1
20>= and <34		20	beta2
35>=		30	beta3
Bad	Payment history	10	beta4
Not good		25	beta5
Good		40	beta6

What does WoE mean?

- Variable: gender (Man, Woman)

$$WOE_M = \ln\left(\frac{\frac{n\_good_M}{n\_good_{All}}}{\frac{n\_bad_M}{n\_bad_{All}}}\right) = \ldots = \ln\left(\frac{n\_good_M}{n\_bad_M}\right) - \ln\left(\frac{n\_good_{All}}{n\_bad_{All}}\right)$$

$$WOE_M = \mathrm{logit(All)}\ \text{-}\ \mathrm{logit(M)}$$

- WoE for Man is relative risk – of odds – with respect to average level

Without transformation – advantages and disadvantages

- The need to impute missing data
- Possible great collinearity and the need for its reduction
- The challenge of nominal features
- More difficult model interpretation
- Sometimes models are more stable in time
- Sensitivity for outlying values

Transformation WOE

- Little probability of over-training
- No need for missing imputation
- Little collinearity
- Similar approach for nominal and intervals
- Resistance for outlying values
- Always one can produce a good model
- Good estimations – little number of parameters

Transformation Dummy

- Possibility of overtraining
- Difficult assumptions’ verification
- Too many parameters to estimate – challenges with minimal dataset requirements subject to pre-defined predictiveness test
- Naive Bayes – bad assumption on variables’ independence

Multidimensional selection

- Step methods:
  – Python - RFE: Recursive Feature Elimination):
  – SAS: Forward, backward, stepwise
- Heuristics, all combinations
  – Python: all combinations
  – SAS: Best subset selection, method of division and constraints, score method
- Every model should be evaluated by different statistics

Multidimensional selection

- Stability statisics:
  - AR_diff
- Collinearity statistics:
  - Max_VIF – variance inflaction factor,
  - Max_CI – condition index,
  - Max_Pearson – Pearson correlatoin,
  - N_beta_minus – beta sign
- Significance statistics:
  - Max_ProbChiSq
- Predictiveness statistics
  - Gini train, valid

Multidimensional selection - benchmarks

- Stability statisics:
  - AR_diff < 0.1, 0.5
- Collinearity statistics:
  - Max_VIF < 3, 5, 10
  - Max_CI < 10, 50, 100,
  - Max_Pearson < 0.7, 0.8, 0.9
  - N_beta_minus = 0
- Relevance statistics:
  - Max_ProbChiSq < 0.05
- Predictiveness statistics
  - Gini train, valid – depends on model type:
    - Application around 50%, Behavioral – 70%

Multidimensional selection

- SAS:
  - %include "&dir_codes.steps_selection.sas" / source2;
  - %include "&dir_codes.score_selection.sas" / source2;
- Python:
  - #Simple RFE selection method …
  - #Assessment of combinations of features
  - number_vars=12
  - number_features=6

Business criteria for variables and models

- The reliability of the variable
  - Can it be verified?
  - Is this information easy to obtain?
  - Can this data be manipulated?
  - Whether they come from a reliable data source?
- Variable cost
  - How much does it cost to get this data?
- Other criteria:
  - Do we exclude certain groups?
  - Does the client want to provide it?

Partial score calculation

$$score = \log(odds) * factor + offset =$$

$$(-\sum_{i=1}^{n}(woe_i * \beta_i) + \alpha) * factor + offset =$$

$$(-\sum_{i=1}^{n}(woe_i * \beta_i + \frac{\alpha}{n})) * factor + offset =$$

$$\sum_{i=1}^{n}(-(woe_i * \beta_i + \frac{\alpha}{n}) * factor + \frac{offset}{n})$$

$$600 = \log(50) * factor + offset$$

$$620 = \log(100) * factor + offset$$

$$factor = 20 / \log(2)$$

$$offset = 600 - factor * \log(50)$$

Partial score calculation

$$\mathrm{WoE}_k = \ln\left(\frac{G_k/G}{B_k/B}\right) =$$

$$= \ln\left(\frac{G_k}{B_k}\right) - \ln\left(\frac{G}{B}\right),$$

so:

$$\mathrm{WoE}_k = \mathrm{Logit}_k - \mathrm{Logit},$$

where k - stands for any variable category
G and B - counts of good and bad clients in the entire population
G_k and B_k - counts of good and bad clients in the category

Therefore, we have the correlation that Weight of Evidence for a category is the difference between the category logit and the entire population logit. Therefore we call the method of building the model „LOG” and calculate logit instead of WoE.

Partial score calculation

Each variable selected for the model is transformed into pieces of a constant based on the calculated logits of each of its categories. The general logistic regression estimation is given by the formula:

$$\mathrm{Logit}(p_n) = X_n\beta,$$

where p_n is the probability that the client is good. $p_n = P(Y = Good)$ when n is this observation, and β represents a vector of regression coefficients. The Matrix X_n can be written in detail as follows:

$$X_n = l_{ij}\delta_{ijn},$$

where l_{ij} is the logit of j – this category and I – this variable, and δ_{ijn} is a zero-one matrix enclosing the value of one, when n is the observation belonging to j – this category and i – this variable. In addition, a simplified assumption was made that each variable has the same categories so as not to enter more indices, and that the number of categories is the same as the number of variables and is represented by v.

Partial score calculation

The product of the *X* matrix and the *β* vector, standing on the right side of the regression equation, is the point score for the given observation. This assessment is not calibrated and is difficult to interpret. Usually, a few simple transformations are made to give it a more useful form. Note that if the probability value p_n increases, then its logit also increases, and therefore the score will also increase. So, the higher the score, the more likely it is that the client will pay back the loan. Most often, the score value is calibrated through a simple linear function:

$$\mathrm{Logit}(p_n) = \ln\left(\frac{p_n}{1-p_n}\right) = S_n = aS_n^{\mathrm{New}} + b,$$

where S_n^{New} is the new rating and S_n the old one, while *a* and *b* are the coefficients. They are designated in order to obtain an additional property, which is defined in the book as follows: for 300 points the chance of being a good customer should be 50, and when the chance doubles, i.e. it will be 100, the rating should be 320. Chance is defined as the quotient of the number of good to bad customers, or as the ratio $\frac{p_n}{1-p_n}$. A chance of 50 represents, therefore, the customer segment, where there are 50 good ones per one bad.

Partial score calculation

$$\ln(50) = a\,300 + b,$$

$$\ln(100) = a\,320 + b.$$

The solutions have values:

$$a = \frac{\ln\left(\frac{100}{50}\right)}{20} = \frac{\ln(2)}{20},$$

$$b = \ln(50) - \frac{300\ln\left(\frac{100}{50}\right)}{20} = \ln\left(\frac{50}{2^{15}}\right).$$

The second activity when scaling the value of the score is to ensure that all the first category of partial scores have the same number of points. The first category is represented by a group of the most risky clients. The last one represents the best, if partial scores always start with the same value, then the variable that has the highest partial score value can be interpreted as the most important in the model.

Furthermore:

$$S_n = \sum_{i,j=1}^{v} \beta_i l_{ij} \delta_{ijn} + \beta_0.$$

Partial score calculation

We can isolate the segment of associated with the worst customer:

$$\gamma = \sum_{i=1}^{v} \beta_i l_{i1},$$

and thanks to that the intercept coefficient can be divided into two components:

$$\beta_0 = \sum_{i=1} \frac{\beta_0 + \gamma}{v} - \sum_{i=1} \beta_i l_{i1}.$$

This creates a partial score:

$$P_{ij} = \beta_i l_{ij} + \frac{\beta_0 + \gamma}{v} - \beta_i l_{i1}.$$

We notice that for each variable *i* we have:

$$P_{i1} = \frac{\beta_0 + \gamma}{v},$$

So the partial scores begin with the same value.

Partial score calculation

Furthermore:

$$S_n = \sum_{i,j=1}^{v} P_{ij} \delta_{ijn},$$

And finally:

$$S_n^{New} = \frac{S_n - b}{a} = \sum_{i,j=1}^{v} P_{ij}^{New} \delta_{ijn},$$

Where:

$$P_{ij}^{New} = \frac{1}{a} P_{ij} - \frac{b}{v}.$$

The final value of the partial evaluation is often rounded to the nearest total value. This way you get a scoring card with points calculated for each category from the variables selected for the model.

Partial score calculation

- Refer to the code:
- different_betas.sas

Partial score calculation

- SAS:
  - %include "&dir_codes.model_assessment.sas" / source2;

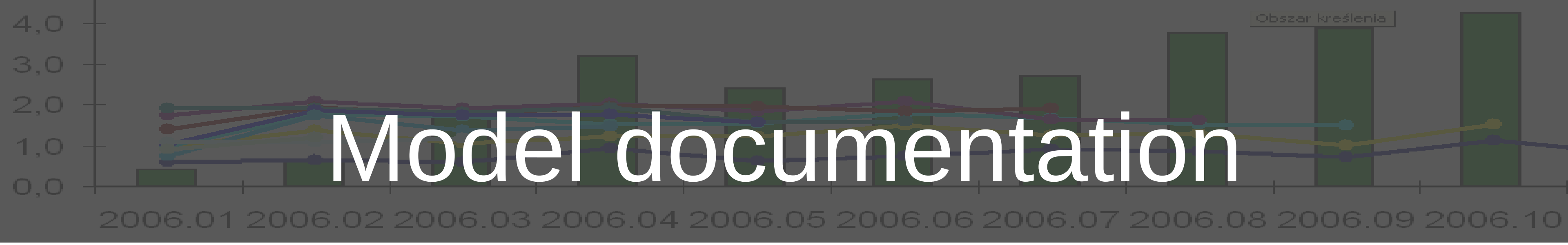
- Python:
  - #Assessment of combinations of features
  - #Creating Scorecard

Properties of the scorecard

- The most important variable has the highest partial rating.

Scale of variable's scorecard points				
Variable	Minimum of scorecard	Maximum of scorecard points	Range of scorecard points	Part of global range
APP_CHAR_JOB_CODE	8	115	107	29.08%
ACT_CCSS_N_STATC	8	79	71	19.29%
ACT_CCSS_DUEUTL	8	70	62	16.85%
ACT_CC	8	61	53	14.40%
ACT12_N_ARREARS	8	59	51	13.86%
ACT_CCSS_MIN_LNINST	8	32	24	6.52%

Gini statistics for variables in the model	
Variable	Gini statistics for variable
ACT_CCSS_DUEUTL	45,53%
ACT12_N_ARREARS	44,87%
ACT_CCSS_MIN_LNINST	36,79%
ACT_CCSS_N_STATC	27,19%
ACT_CC	14,75%
APP_CHAR_JOB_CODE	7,45%



Model documentation

- SAS:
  - %include "&dir_codes.final_report.sas" / source2;
- Puython:
  - #Model report

- The SAS model can be also documented in Excel by Python codes like Python model

Scoring code

- SAS:
  - %include "&dir_codes.scoring_code.sas" / source2;
- Puython:
  - # Scoring code

Confusion Matrix

		Observed	
		True	False
Predicted	True	True Positive (TP)	False Positive (FP)
	False	False Negative (FN)	True Negative (TN)

Basic Concepts

- We set a value for c – cutoff:
  - TP + FN = P (observed positive)
  - TN + FP = N (observed negative)

  - TP + FP = PP (predicted positive)
  - TN + FN = PN (predicted negative)

  - FPrate= FP/N,
  - TPrate=TP/P=Recall,
  - Accuracy=PCC=(TP+TN)/(P+N)

Basic concepts

– Specificity = TN/N

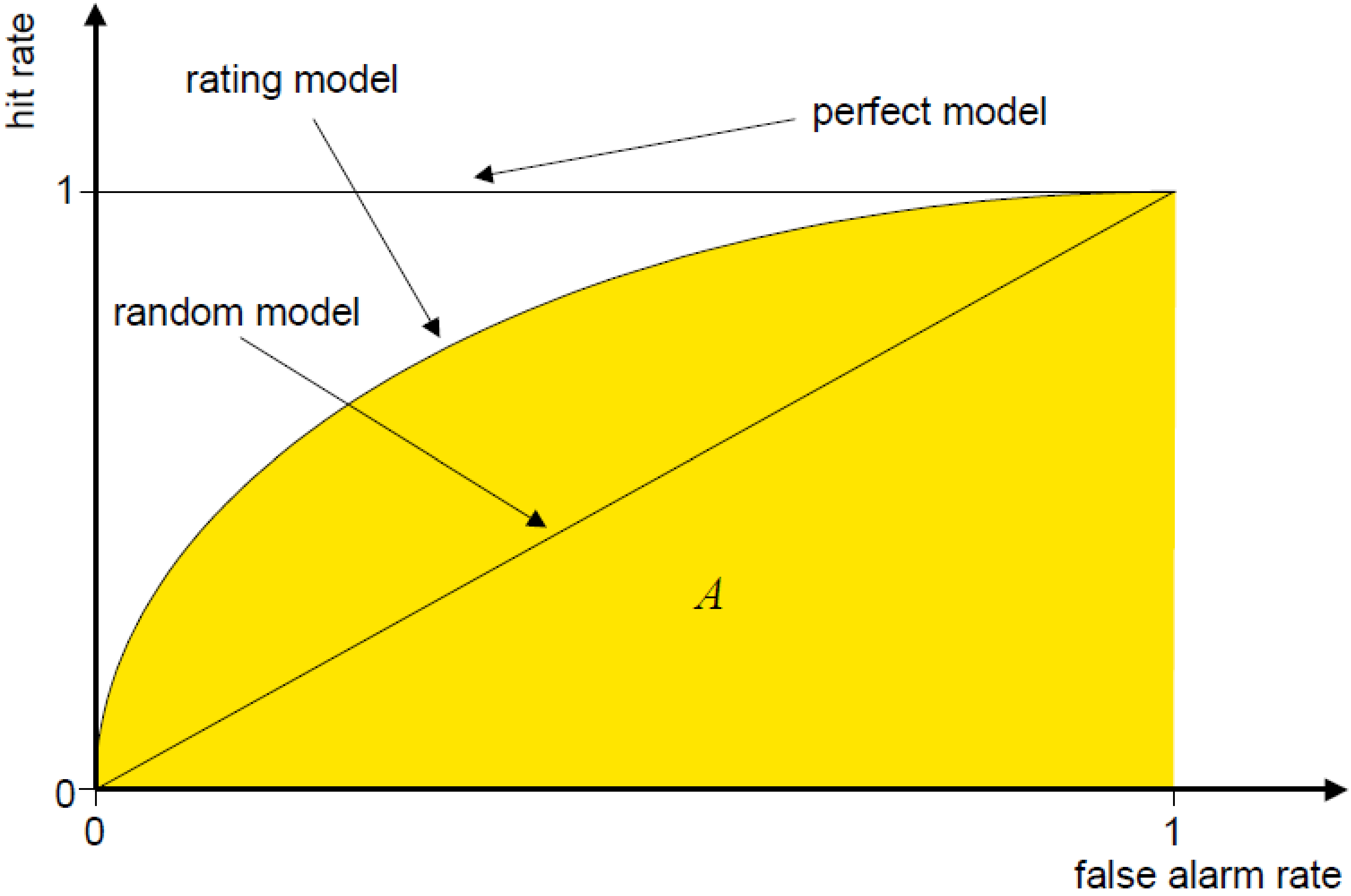
– PV+ = TP/PP (response rate),

– PV- = TN/PN

ROC (Receiver Operating Characteristic):

- x = FPrate = 1-Specificity = false alarm rate
- y = TPrate = Sensitivity = hit rate

ROC (Receiver Operating Characteristic) AUC (Area Under Curve)

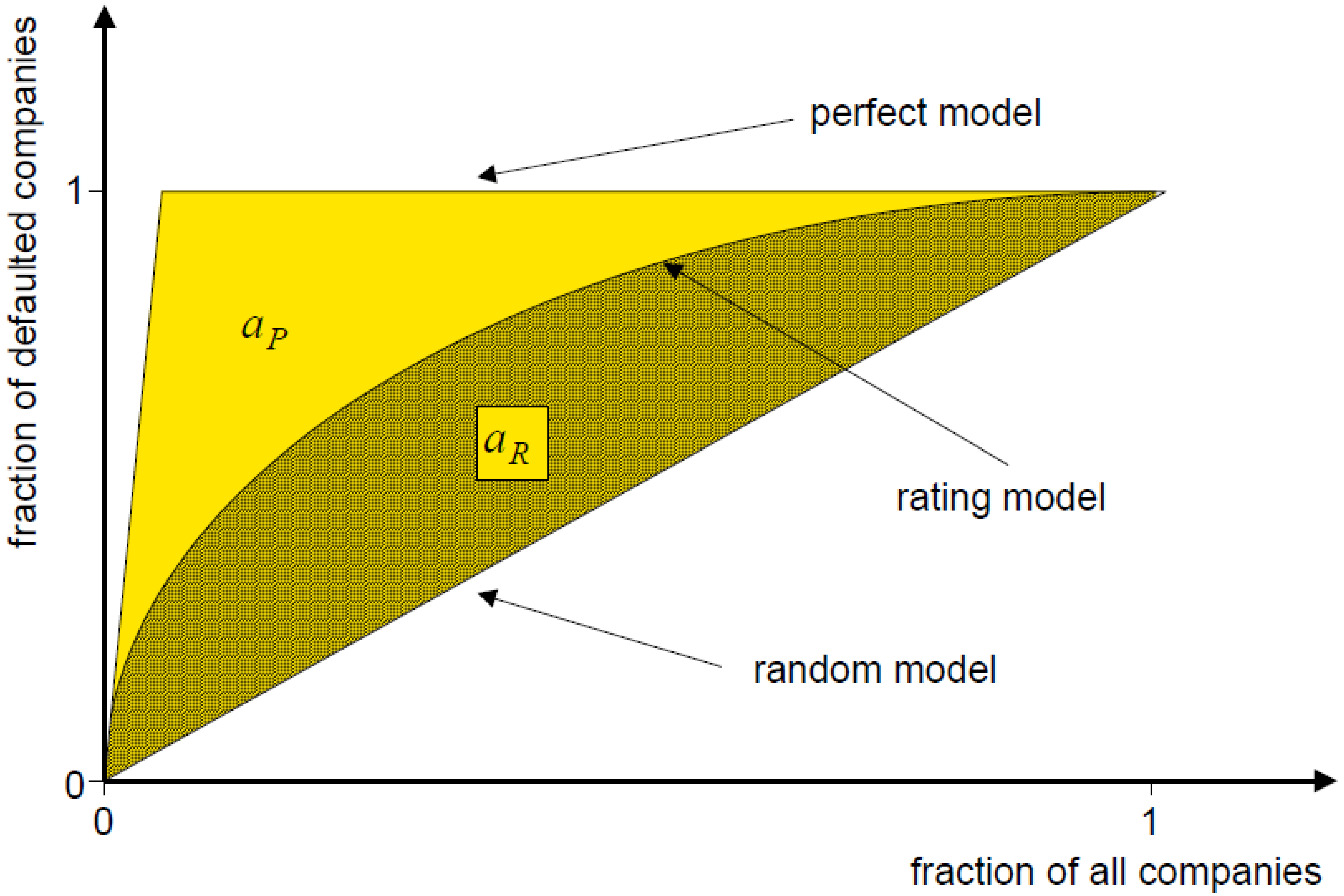


CAP, Lift, Gains and Lorentz Curves

- Depth – penetration rate – population share – the share above cutoff
- Rho1 = P/(P+N) – response rate of the population

- Gains:
  - x = Depth, y=TPrate=TP/P=Recall, how many percent of ones in the selected set of all ones
- Lift:
  - x = Depth, y=PV+/Rho1, how many Times better than the random model
- Lorentz (concentarion curve, CAP) :
  - x = Depth, y = Sensitivity

CAP (Cumulative Accuracy Profiles)



Formulas

- $AR = Gini = a_p/a_r$
- $AUC = C = A$
- $2*C-1= AR$

Gini

$$c = (n_c + 0.5(t - n_c - n_d))/t$$

$$\text{Somers' } D \text{ (Gini coefficient)} = (n_c - n_d)/t$$

$$\text{Goodman-Kruskal Gamma} = (n_c - n_d)/(n_c + n_d)$$

$$\text{Kendall's Tau-}a = (n_c - n_d)/(0.5N(N-1))$$

- n_c – Number of matches ($P_i > P_j$, where i-bad, j-good) concordant, P=P(being bad, Y=1)
- n_d – Number not matching discordant?
- t – Count of all pairs
- Gini= P_c - P_d

Gini - interpretation

- Gini= P_c - P_d
- $P_c + P_d + P_t = 100\%$
- Assuming $P_t = 0$ we have:

  $P_c + P_d = 100\%$

  Gini=2 P_c - 1

  $P_c = (\text{Gini}+1)/2$

Cost Matrix

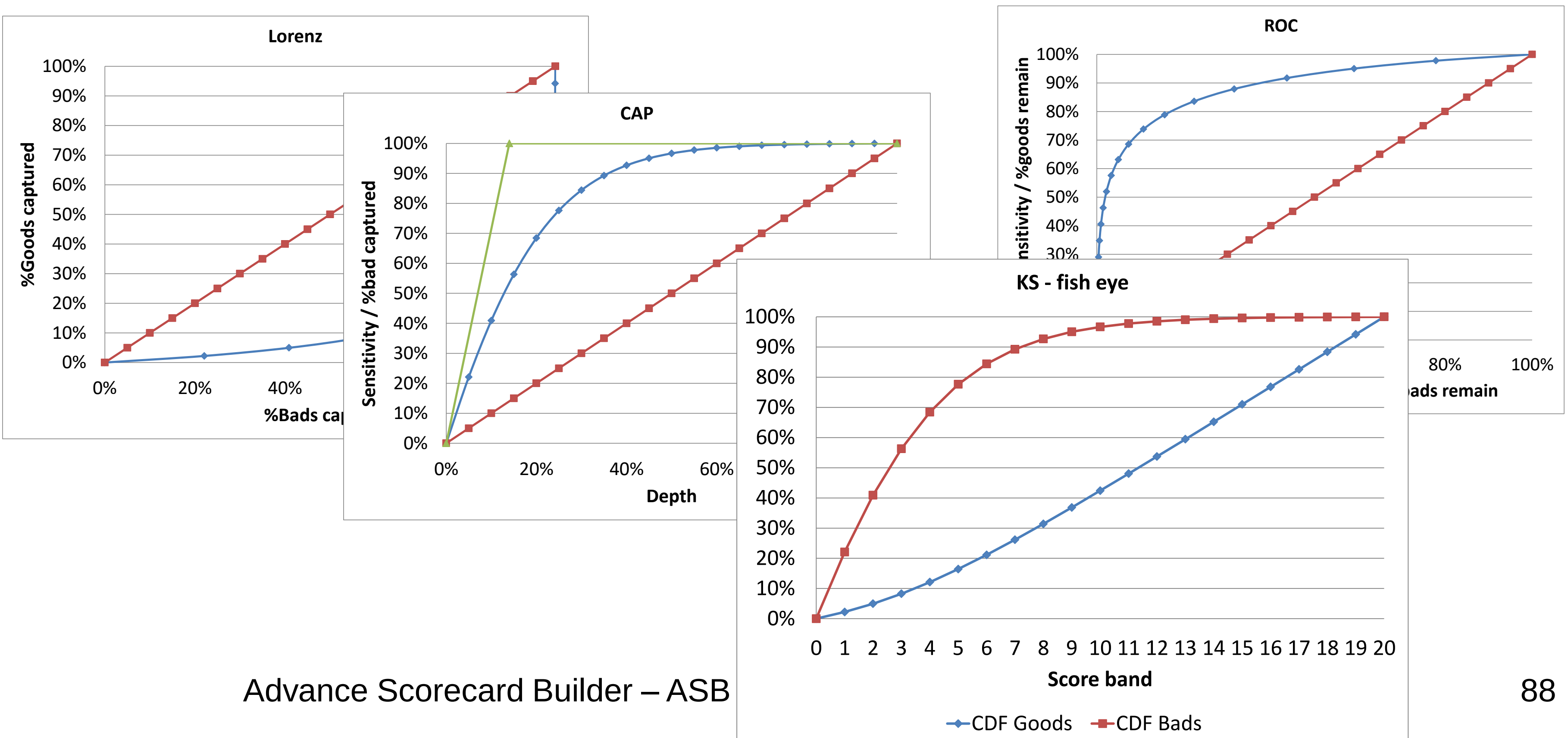
		Observed	
		True	False
Predicted	True	C_{TP} (TP)	C_{FP} (FP)
	False	C_{FN} (FN)	C_{TN} (TN)

All known curves in Excel

All Excels with various statistics and curves.

http://administracja.sgh.waw.pl/pl/OW/publikacje/Strony/2015.aspx

http://administracja.sgh.waw.pl/pl/OW/publikacje/Documents/gini_curves.xlsx



Model lifecycle

- Application for a new model (model request)
- Model building
- Validation
- Implementation
- Monitoring
- Monitoring review
- Decision to change the model

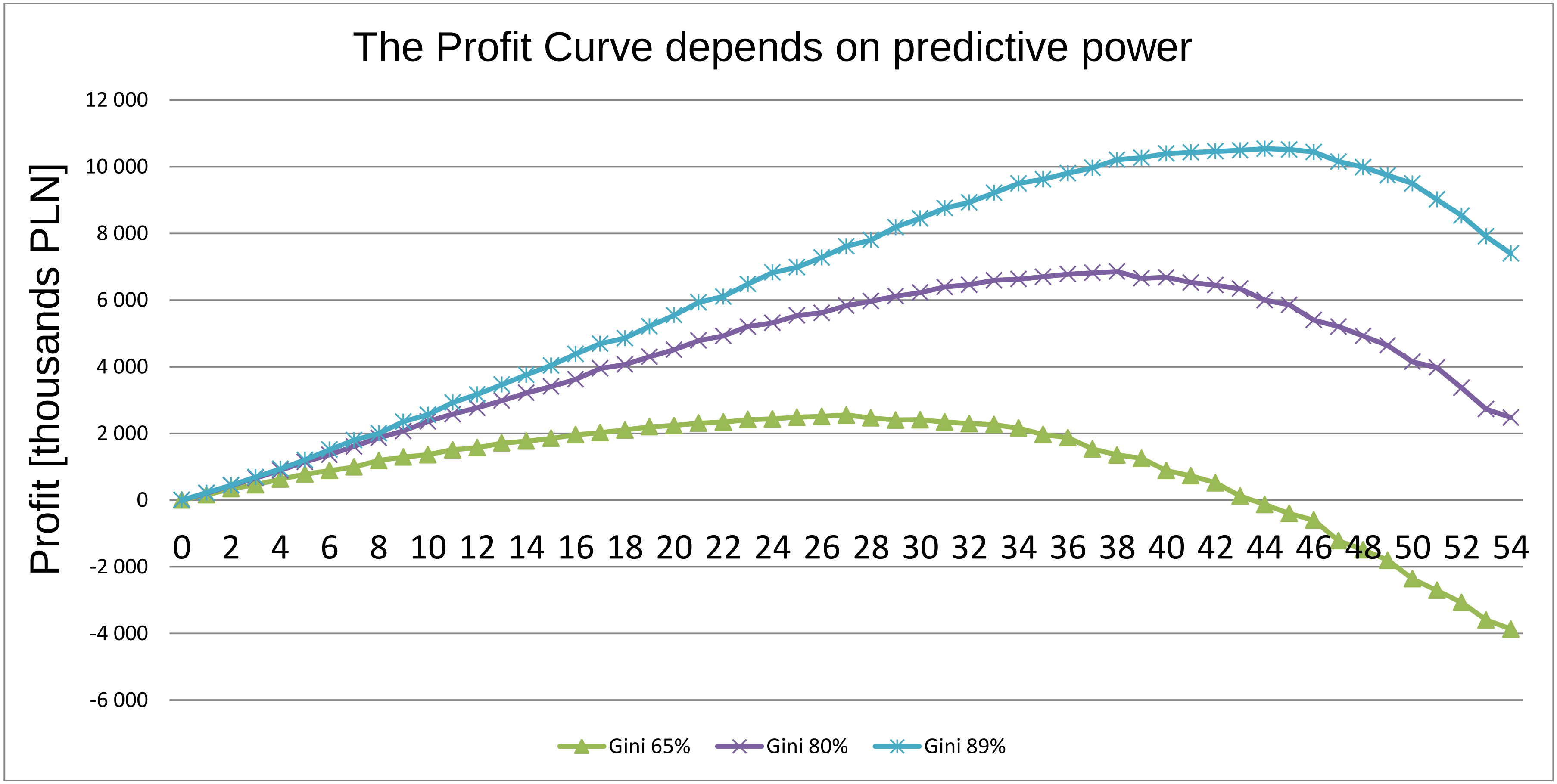
- Each point is a different document

Monitoring of models

- SAS code monitoring.sas
- Folder:

…\CS-AUT\software\ASB_SAS\monitoring\

Profit-Loss Curve



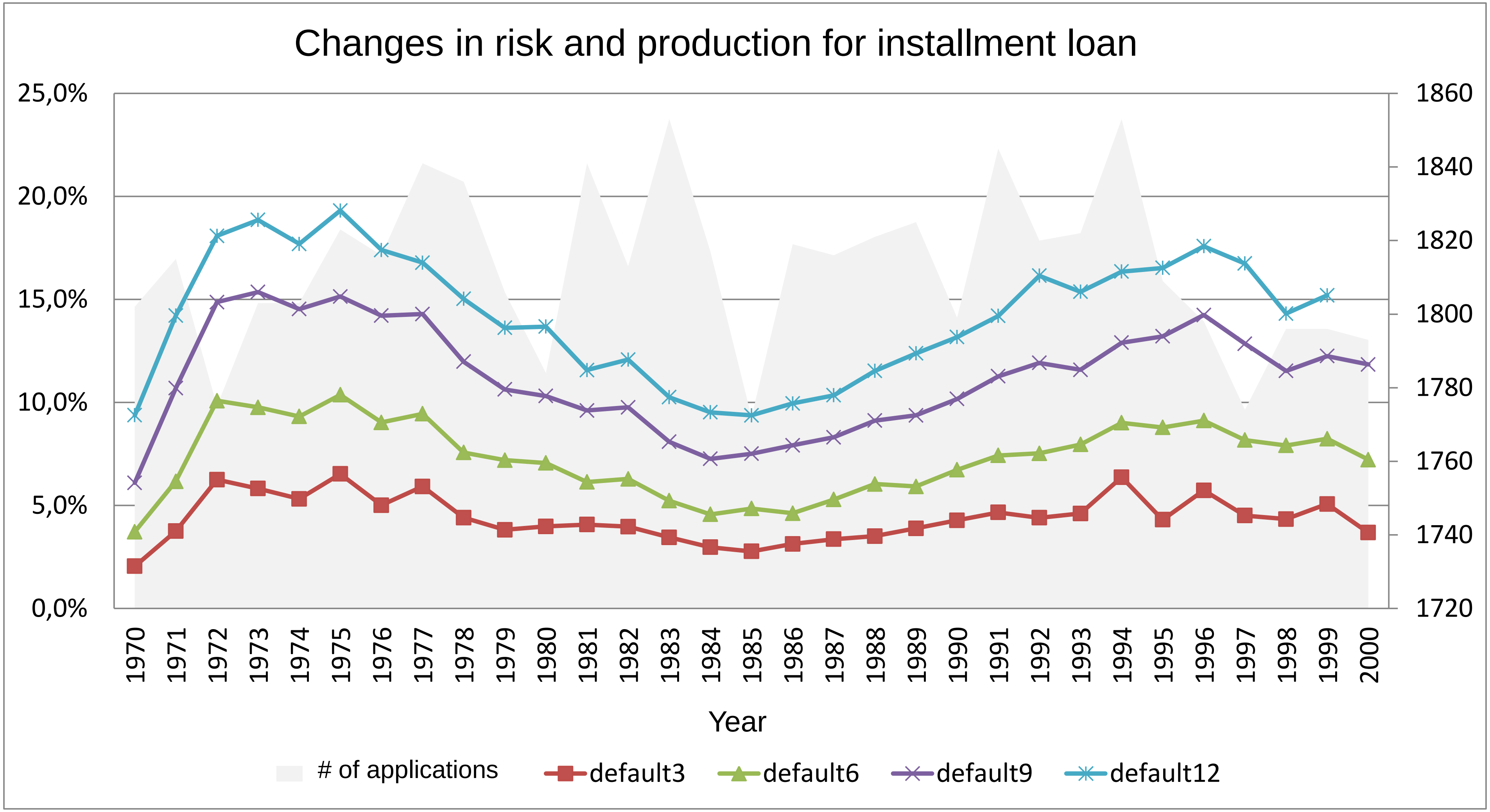
Data construction assumptions

- The customer will always get a loan somewhere, if not in a bank, in a consumer bank or from friends or family
- The client has his priorities. He repays some loans and does not repay others
- The repayment of cash loans depends on previous history, including the repayment of installment loans
- We therefore have the potential of data already generated with the entire repayment history

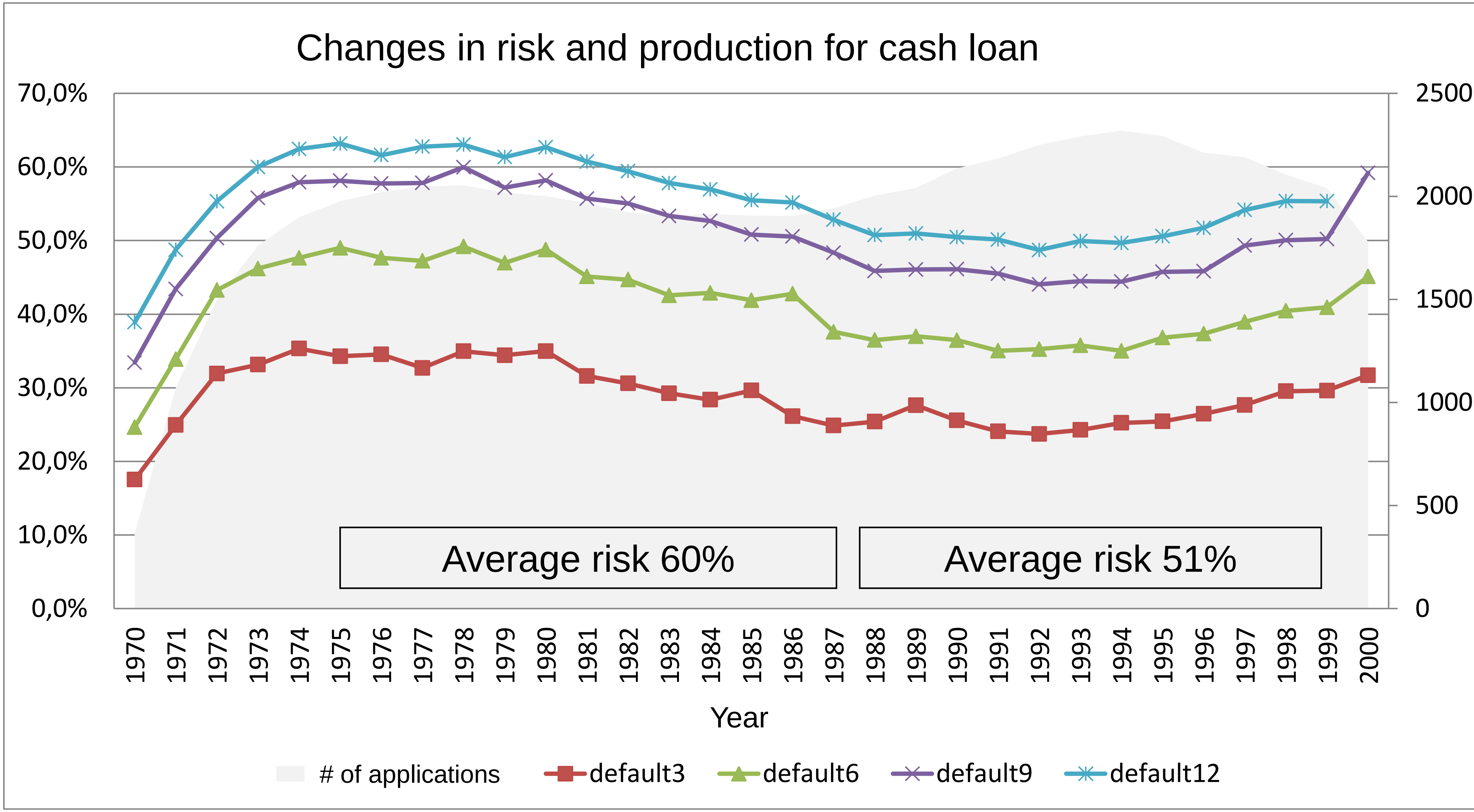
Data construction assumptions

- The bank can choose which customer loans to accept, thereby reducing their losses
- If the bank does not accept some loans for the client, the bank loses valuable information about the client. It only knows about the better side of the client.
- Therefore, the problem of Reject Inference arises
- In addition, there is also a lack of opportunities to sell a cash loan because the customer was rejected earlier when applying for installment loan

Installment Loan

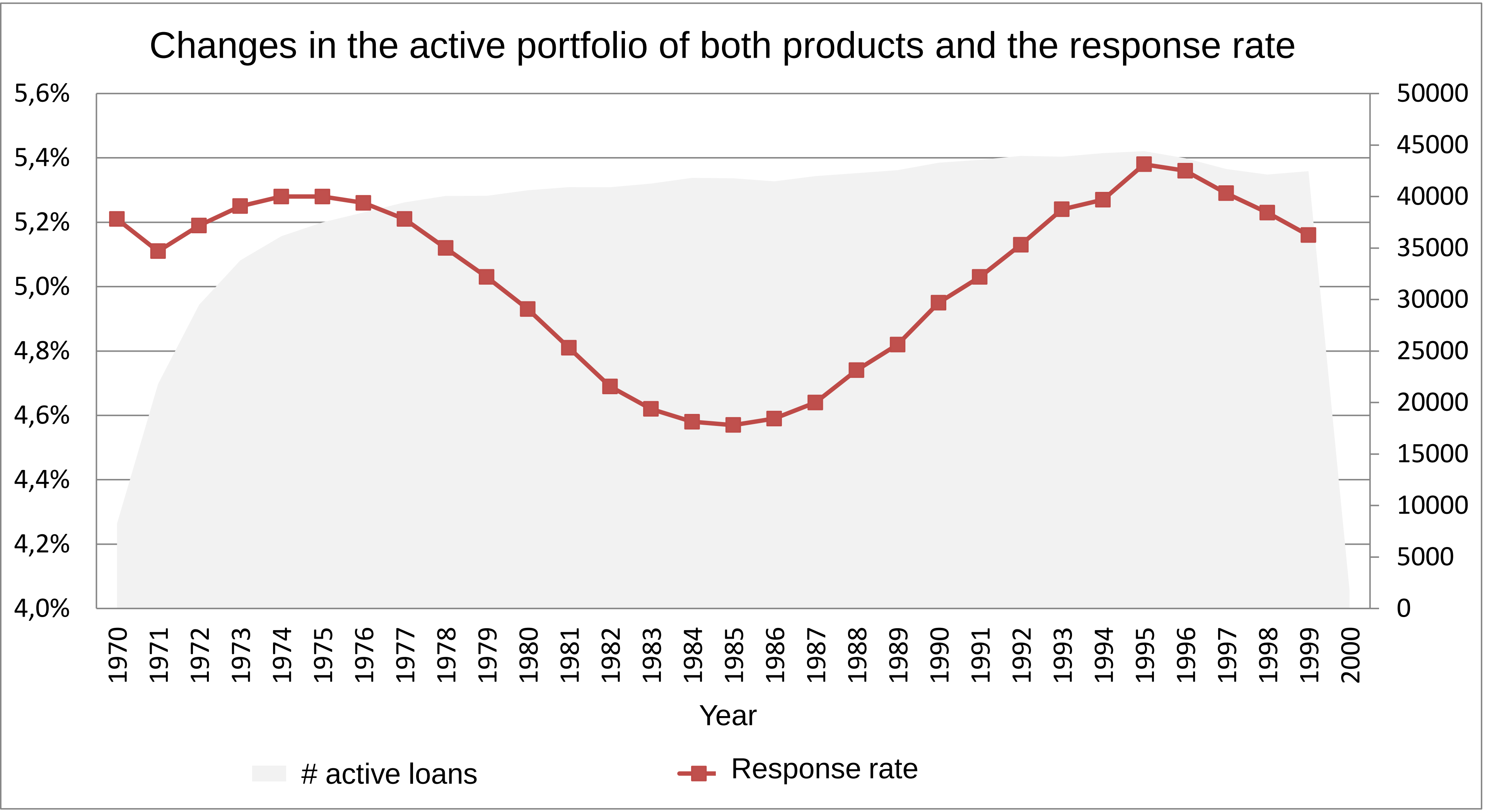


Cash Loans

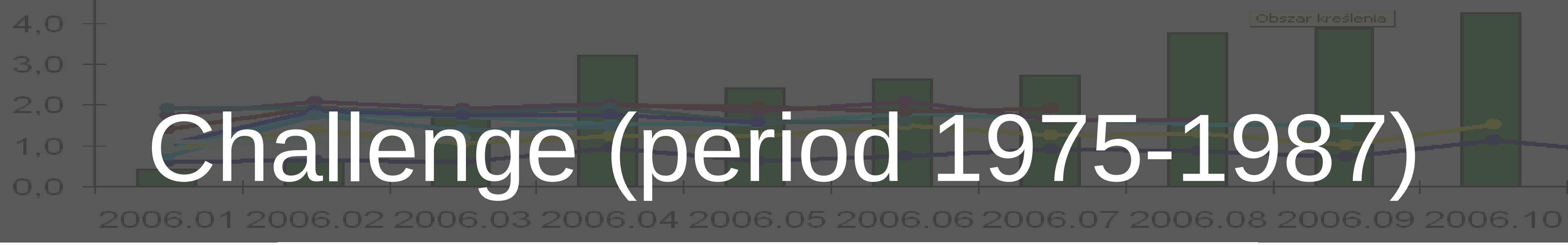


Changes in risk and production for cash loan
70,0%
60,0%
50,0%
40,0%
30,0%
20,0%
10,0%
0,0%
2500
2000
1500
1000
500
0
Average risk 60%
Average risk 51%
1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000
Year
# of applications
default3
default6
default9
default12

Monthly portfolio



Challenge (period 1975-1987)



KPI	Installment	Cash	Total
Profit	-7,824,395	-31,627,311	-39,451,706
Income	969,743	10,260,689	11,230,432
Loss	8,794,138	41,888,000	50,682,138

- 4 models of scoring cards (estimated on the entire population during the period 1975-1987):
  - Installment loan risk model (PD Ins)
  - Cash loan risk model (PD Css)
  - Risk model for a cash loan when applying for installment loan (Cross PD Css)
  - Model of the propensity to use a cash loan when applying for installment loan (PR Css) (response model)

Period 1975-1987

- Calibration of models to probability:

PD_Ins=1/(1+exp(-(-0.032205144*risk_ins_score+9.4025558419)))

PD_Css=1/(1+exp(-(-0.028682728*risk_css_score+8.1960829753)))

Cross_PD_Css=1/(1+exp(-(-0.028954669*cross_css_score+8.2497434934)))

PR_Css=1/(1+exp(-(-0.035007455*response_score+10.492092793)))

Model	Gini
Cross PD Css	74,01%
PD Css	74,21%
PD Ins	73,11%
PR Css	86,37%

Cash optimization

- Studying the entire population from the period 1975-1987, we determine the profit curve and find the optimal point:
  - rejections rule *PD_Css* > 27,24%
  - cash acceptance percentage 18,97%
  - profit for cash 1 591 633 PLN
- Can we do the same with installment loans?

Customer Life Time Value (CLTV)

- Every installment loan is a chance to earn more, if the customer takes a cash loan.
- Therefore, you have to consider the product sequence: first installment loan, second cash loan.
- We create rules by splitting the population into groups determined by installment risk estimation and an estimation of cash propensity

Segmentation of CLTV

GR PR Css	GR PD Ins	# of applications Ins	Global Profit	Min PR Css	Max PR Css	Min PD Ins	Max PD Ins
4	0	1 277	372 856	4,81%	96,61%	0,02%	2,18%
4	1	581	96 096	4,81%	96,61%	2,25%	4,61%
1	0	2 452	67 087	1,07%	1,07%	0,32%	2,18%
3	0	907	46 685	2,80%	4,07%	0,07%	2,18%
3	1	734	14 813	2,80%	4,07%	2,25%	4,61%
3	2	307	12 985	2,80%	4,07%	4,76%	7,95%
4	2	361	8 039	4,81%	96,25%	4,76%	7,95%
3	3	446	-1 283	2,80%	4,07%	8,19%	18,02%
4	3	417	-5 774	4,81%	95,57%	8,19%	18,02%
1	1	3 570	-82 886	1,07%	1,07%	2,25%	4,61%
1	2	4 044	-408 644	1,07%	1,07%	4,76%	7,95%
3	4	726	-946 937	2,80%	4,07%	18,50%	99,62%
4	4	1 054	-1 108 313	4,81%	96,25%	18,50%	99,83%
1	3	3 883	-1 270 930	1,07%	1,07%	8,19%	18,02%
1	4	2 878	-4 306 859	1,07%	1,07%	18,50%	97,00%

Rules for CLTV with Installment Loans

- Rejection Rules:
  - *PD_Ins* > 8,19%
  - 8,19% >= *PD_Ins* > 2,18% and (*PR_Css* < 2,8% or *Cross_PD_Css* > 27,24%)
- Estimated global profit from the combined process:

  1 686 684 PLN

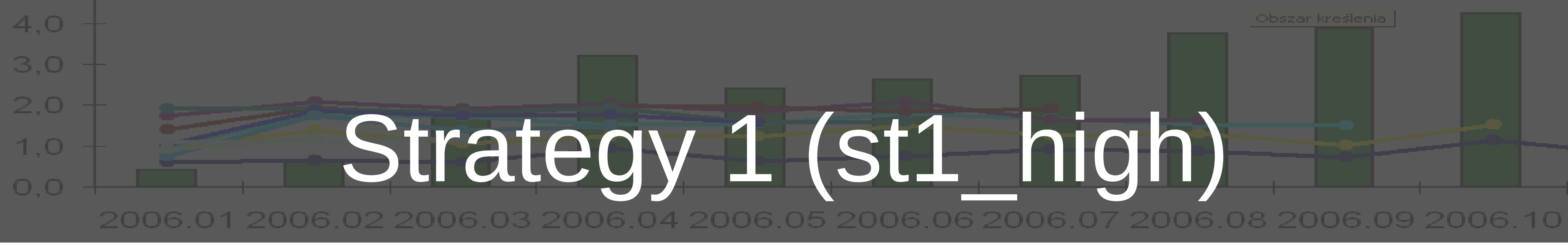
- Rules without PR_Css:
  - PD_Ins > 8,19%
- Estimated global profit from the combined process :

  1 212 261 PLN, or 30% less!

System/Engine for Decisions

- Each set of rules needs to be processed, because depending on credit decisions, the distribution of scoring changes, and because the distribution of variables describing clients changes
- Therefore, we are testing several strategies
  - Strategy 1 – previously found rules
  - Strategy 2 – no rule for PR_Css
  - Strategy 3 – rejection of a bad customer (who defaulted)
  - Strategy 4 – new rules based on strategy 3

Strategy 1 (st1_high)



Rule	Description
PD_Ins Cutoff	$PD\_Ins > 8,19\%$
PD_Css Cutoff	$PD\_Css > 27,24\%$
PD & PR	$8,19\% >= PD\_Ins > 2,18\%$ & ($PR\_Css < 2,8\%$ or $Cross\_PD\_Css > 27,24\%$)

product css

Decline reason	N	Pct	Amount	Risk	Profit
1 PD cut-off on css	8 436	32,97%	42 180 000	67,99%	-13 098 591
998 not active custo	12 999	50,80%	64 995 000	65,91%	-19 171 357
999ok	4 152	33,33% 16,23%	20 760 000	22,35%	642 637
All	25 587	100,00%	127 935 000	59,53%	-31 627 311

product ins

Decline reason	N	Pct	Amount	Risk	Profit
2 PD cut-off on ins	9 289	39,30%	60 214 008	26,95%	-7 339 423
3 PD,PDCross and PR	8 131	34,40%	31 340 808	5,37%	-505 662
999ok	6 217	26,30%	22 698 240	2,14%	20 690
All	23 637	100,00%	114 253 056	13,00%	-7 824 395

Strategy 1

Period	Income	Loss	Profit
1975-1987	3 407 745	2 744 418	663 327
1988-1998	3 761 299	2 246 844	1 514 455

Should have been
1 686 684 PLN

Average Parameter Values		
Parameter	Accepted	All
PD (Both Ins and Css)	7.93%	28.87%
PR Css	17.15%	21.76%
Cross PD Css	21.71%	17.73%

Strength of Prediction

Model	Gini	
	Accepted	**All**
Cross PD on cross	21,34%	40,72%
PD on css	31,66%	53,28%
PD on ins	41,93%	68,58%
PR on cross	72,56%	68,88%

Significant estimation error

- Ins -> Css -> Css -> Css
- Ins -> Css -> Ins -> Css> Css

- Ins -> Css -> Css -> Css
- Ins -> Css -> Ins -> Css> Css

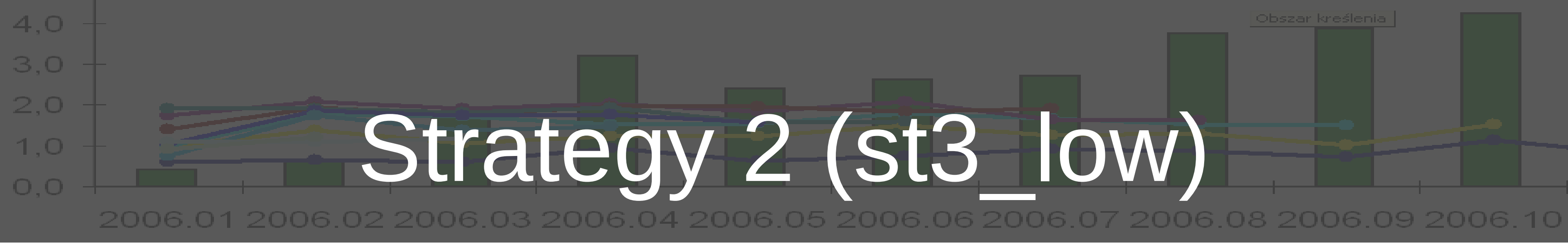
- Ins -> Css -> Css -> Css
- Ins -> Css -> Ins -> Css> Css

Significant estimation error

- Why did we earn only 1 686 684 PLN instead of 663 327 PLN?
- Where has our million gone?

- Impact of the rejected (revolution in the process, from 100% acceptance):
  - Unknown client – 50,8%
  - Approve Installment – 26,3%
  - Approved Cash – 16,23%
  - PD (both PD_Ins & PD_Css) from 37,19% to 28,87%

Strategy 2 (st3_low)



	Income	Loss	Profit
1975-1987	4 008 258	3 896 818	111 441
1988-1998	4 539 328	3 829 634	709 694

551 886 PLN less, which is 83% less!

Rule	Description
PD_Ins Cutoff	$PD\_Ins > 8,19\%$
PD_Css Cutoff	$PD\_Css > 27,24\%$

product css

Decline reason	N	Pct	Amount	Risk	Profit
1 PD cut-off on css	9 297	36,33%	46 485 000	67,84%	-14 381 482
998 not active custo	11 661	45,57%	58 305 000	67,34%	-17 822 432
999ok	4 629	18,09%	23 145 000	23,16%	576 604
All	25 587	100,00%	127 935 000	59,53%	-31 627 311

product ins

Decline reason	N	Pct	Amount	Risk	Profit
2 PD cut-off on ins	9 325	39,45%	60 221 856	26,98%	-7 359 232
999ok	14 312	60,55%	54 031 200	3,89%	-465 163
All	23 637	100,00%	114 253 056	13,00%	-7 824 395

Strategy 3 (st4_bad_due3)

	Income	Loss	Profit
1975-1987	7 496 614	21 801 230	-14 304 616
1988-1998	7 881 992	18 510 342	-10 628 350

Rule	Description
Bad Client	agr12_Max_CMaxA_Due > 3

product css

Decline reason	N	Pct	Amount	Risk	Profit
1 bad customer	7 114	27,80%	35 570 000	79,83%	-14 195 320
998 not active custo	7 036	27,50%	35 180 000	67,04%	-10 673 871
999ok	11 437	44,70%	57 185 000	42,28%	-6 758 120
All	25 587	100,00%	127 935 000	59,53%	-31 627 311

product ins

Decline reason	N	Pct	Amount	Risk	Profit
1 bad customer	483	2,04%	2 047 188	27,74%	-277 899
999ok	23 154	97,96%	112 205 868	12,69%	-7 546 496
All	23 637	100,00%	114 253 056	13,00%	-7 824 395

Strategy 3

Average Parameter Values

Parameter	Acceptance	All
PD (Both Ins and Css)	21.81%	32.70%
PR Css	21.79%	28.83%
Cross PD Css	43.09%	24.48%

Strength of Prediction

Model	Gini	
	Accepted	All
Cross PD on cross	64,83%	63,59%
PD on css	63,67%	64,82%
PD on ins	71,94%	72,56%
PR on cross	79,96%	64,72%

- We do not earn with this strategy, but we are already modifying the scoring patterns on the accepted part

Strategy 4 (st5_from_due3)

Rule	Description
Bad client	$agr12\_Max\_CMaxA\_Due > 3$
PD_Ins Cutoff	$PD\_Ins > 7,95\%$
PD_Css Cutoff	$PD\_Css > 19,13\%$
PD & PR	$7,95\% >= PD\_Ins > 2,8\%$ & $(PR\_Css < 2,8\%$ or $Cross\_PD\_Css > 19,13\%)$

product css

Decline reason	N	Pct	Amount	Risk	Profit
0 bad customer	2 253	8,81%	11 265 000	74,26%	-4 026 033
1 PD cut-off on css	5 375	21,01%	26 875 000	53,66%	-5 462 687
998 not active custo	15 739	61,51%	78 695 000	65,29%	-22 845 756
999ok	2 220	8,68%	11 100 000	17,97%	707 165
All	25 587	100,00%	127 935 000	59,53%	-31 627 311

product ins

Decline reason	N	Pct	Amount	Risk	Profit
0 bad customer	209	0,88%	891 720	27,75%	-121 550
2 PD cut-off on ins	9 253	39,15%	60 130 704	26,46%	-7 208 030
3 PD,PDCross and PR	8 029	33,97%	31 118 232	5,49%	-519 531
999ok	6 146	26,00%	22 112 400	2,05%	24 717
All	23 637	100,00%	114 253 056	13,00%	-7 824 395

Strategy 4

	Income	Loss	Profit
1975-1987	2 010 242	1 278 361	731 882
1988-1998	2 452 716	1 134 729	1 317 986

Average Parameter Values

Parameter	Acceptance	All
PD (Both Ins and Css)	4.24%	25.17%
PR Css	11.37%	15.68%
Cross PD Css	17.02%	14.61%

Strength of Prediction

Model	Gini	
	Accepted	All
Cross PD on cross	3.23%	19.19%
PD on css	33.15%	47.81%
PD on ins	36.79%	67.67%
PR on cross	70.59%	64.89%

Strategy 1 vs. 4

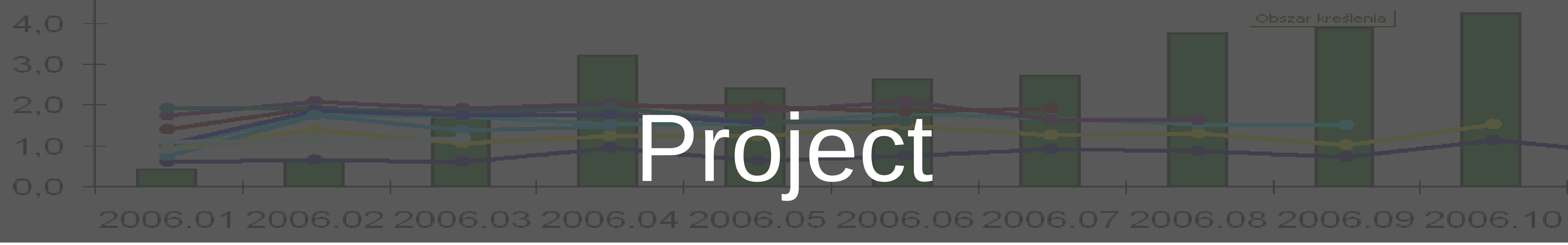
Strategy 1 | Strategy 4

Period	Income	Loss	Profit	Income	Loss	Profit
1975-1987	3 407 745	2 744 418	663 327	2 010 242	1 278 361	731 882
1988-1998	3 761 299	2 246 844	1 514 455	2 452 716	1 134 729	1 317 986

- In a period of prosperity Strategy 1 is better.
- In a period of greater risk Strategy 4 is better.

Conclusions

- The impact of rejected applications in the approval process is difficult to predict
- A safety solution in process management is slow policy change
- Never make revolutionary changes!
- Strategies must change
- Continuous improvement, continuous testing of new models and rules



Project

- How to run a project?
- How to change rules in scoring engine?
- Main reports

- Only in SAS: all_contents.sas
  - Folder:

…\CS-AUT\software\PROCSS_SIMULATION\codes\

What calibration is?

We can define two kinds of calibration. First it is a transformation from probability of default into scoring points, where logit function is used and second, from scoring points into probability, where is used inverse logit function in the following form:

$$p_n = \frac{1}{1 + e^{-(\omega_s S_n^{New} + \omega_0)}},$$

where ω_s & ω_0 are coefficients

Segmentation

- All codes in:

…\CS-AUT\software\PROCSS_SIMULATION\process\segmentation\

Segmentation

Observed - expected risk					
Segments	**N**	**Pct**	**Risk**	**PD**	**PD Seg**
All	23 637	100,00%	13,00%	13,00%	13,00%
Miss	16 827	71,19%	12,61%	11,34%	12,61%
NMiss	6 810	28,81%	13,96%	17,09%	13,96%

Predictive powers		
Segments	**PD**	**PD seg**
All	71,13%	76,06%
Miss	63,54%	68,80%
NMiss	85,92%	88,33%

Segmentation – two models

Categories of variable in case of model for known customer

Condition	Nobs	PcT	Risk
ACT_CINS_N_STATC ⩽ 0	666	16,3%	28,5%
0 < ACT_CINS_N_STATC ⩽ 2	2 616	63,8%	13,2%
2 < ACT_CINS_N_STATC ⩽ 3	367	9,0%	9,3%
3 < ACT_CINS_N_STATC ⩽ 4	222	5,4%	5,4%
4 < ACT_CINS_N_STATC	227	5,5%	2,2%

Categories of variable in case of PD INS model for all customers

Condition	Nobs	PcT	Risk
ACT_CINS_N_STATC ⩽ 0	535	4,7%	29,0%
0 < ACT_CINS_N_STATC ⩽ 1	1 528	13,4%	12,6%
Missing	8 105	71,2%	12,4%
1 < ACT_CINS_N_STATC ⩽ 2	604	5,3%	11,4%
2 < ACT_CINS_N_STATC	607	5,3%	6,1%

Segmentation – two models

Categories of variable in case of model for unknown customer

Condition	Nobs	PcT	Risk
Contract	823	8,2%	43,1%
Owner company	1 236	12,3%	15,0%
Retired	4 276	42,5%	10,3%
Permanent	3 725	37,0%	8,8%

Categories of variable in case of PD INS model for all customers

Condition	Nobs	PcT	Risk
Contract	768	6,7%	42,1%
Owner company	1 265	11,1%	15,3%
Retired	5 754	50,6%	10,5%
Permanent	3 592	31,6%	9,4%

Variable corrections

- In some cases, especially due to instability of category shares or risk, we need to make some corrections on categories definitions, to change some conditions.
- SAS:
  - %include "&dir_codes.variable_corrections.sas" / source2;
- Python:
  - #labsn['app_number_of_children']=[-np.inf, 1, 1, 2, np.inf]

Interaction

```
/*Important macro to create new variables and define where statement*/
%macro Additional_variables;
length app_IGJM $ 30;
outstanding=app_loan_amount;
credit_limit=app_loan_amount;
app_IGJM = trim(app_char_gender)||'-'||trim(app_char_job_code)||
           '-'||trim(app_char_marital_status);
where '197501'<=period<='198712' and product='css' and decision='A';
%mend;
```

Interaction

Attributes for variable APP_IGJM						
Attribute number	**Condition**	**Bad rate (br)**	**%POP**	**%GD**	**%BD**	**%IND**
1	otherwise	60,64%	10,12%	6,74%	14,37%	7,64%
2	when ('Female-Retired-Divorced')	49,89%	12,95%	11,32%	15,13%	11,35%
3	when ('Female-Permanent-Divorced','Male-Permanent-Maried','Male-Retired-Maried')	46,33%	13,25%	12,53%	14,37%	12,01%
4	when ('Male-Retired-Divorced','Male-Retired-Widowed')	43,48%	10,18%	10,38%	10,37%	8,95%
5	when ('Female-Permanent-Maried')	37,02%	14,67%	15,36%	12,72%	18,56%
6	when ('Female-Retired-Maried','Female-Retired-Widowed')	36,32%	38,83%	43,67%	33,03%	41,48%

Reject Inference

- Wrong estimation of risk
- Biased sample, not included rejected cases -> wrong risk estimation, especially on rejected part
- External databases supporting to minimize mentioned problem:
  - Credit Bureau data
  - Data with bad customers, blacklists, unreliable customers

Bank Consumer Seniority

- Typical conclusion observed in any bank: longer customer seniority – lower risk value.
- Is it a customer property or impact of process?
- It is the result of cleaning process implemented in every bank. Every bad customer is rejected in next processes.
- Let's study categories of mentioned variable on two strategies:
  - All accepted, heaven strategy
  - Strategy 3 (st4_bad_due3)

Categories of variable

Categories for ACT_CCSS_CENIORITY in case of strategy all

Group number	Condition	Risk	PcT	Number of cases
1	$25 < \text{ACT\_CCSS\_SENIORITY} \leqslant 57$	71,50%	19,42%	2 684
2	$18 < \text{ACT\_CCSS\_SENIORITY} \leqslant 25$	68,74%	6,50%	899
3	$57 < \text{ACT\_CCSS\_SENIORITY} \leqslant 67$	61,40%	6,00%	829
4	$67 < \text{ACT\_CCSS\_SENIORITY} \leqslant 140$	59,66%	37,00%	5 114
5	$140 < \text{ACT\_CCSS\_SENIORITY}$	54,86%	17,55%	2 426
6	$\text{ACT\_CCSS\_SENIORITY} \leqslant 18$	49,47%	6,14%	849
7	missing(ACT_CCSS_SENIORITY)	34,90%	7,38%	1 020
		59,36%	100,00%	13 821

Categories for ACT_CCSS_CENIORITY in case of strategy 3

Group number	Condition	Risk	PcT	Number of cases
1	$18 < \text{ACT\_CCSS\_SENIORITY} \leqslant 41$	59,73%	16,34%	1 125
2	$41 < \text{ACT\_CCSS\_SENIORITY} \leqslant 53$	47,97%	3,94%	271
3	$\text{ACT\_CCSS\_SENIORITY} \leqslant 18$	46,14%	11,30%	778
4	$53 < \text{ACT\_CCSS\_SENIORITY} \leqslant 142$	42,51%	37,42%	2 576
5	$142 < \text{ACT\_CCSS\_SENIORITY} \leqslant 184$	34,53%	12,12%	834
6	missing(ACT_CCSS_SENIORITY)	31,65%	15,24%	1 049
7	$184 < \text{ACT\_CCSS\_SENIORITY}$	25,10%	3,65%	251
		42,69%	100,00%	6 884

Conclusions

- In case strategy all a customer with longer history is riskier than with short history. If you more roll the dice, you can finally see 6.
- Some properties of a customer relate to the process
- You must include cleaning process of bad customers in your scoring analysis to estimate the risk in a better way

Results

- Correct risk value of customers with missing(ACT CCSS SENIORITY) is 34,90%
- Category created in strategy 3 has 31,65%, it is correct value only if you consider two rules together:
  – missing(ACT CCSS SENIORITY) and *agr*12*_Max_CMaxA_Due* > 3
- We need to study and include information about the old process rules when we build a new model, because we estimate on biased sample

Reject Inference

- Model KGB – known good bad
- Analysis, estimation of risk on rejected customers, preparation of ABT for all cases
- Model All
- Calibration and validation
- Folder:

…\CS-AUT\materials_all\reject_inference_modeling\

Reject Inference

Target / Segments / Gini		New score	Old score
default12	**Accepted**	36,15%	41,29%
	All	24,73%	65,55%
	Rejected	14,09%	48,29%
default12_ind	**Accepted**	37,34%	42,77%
	All	26,12%	67,60%
	Rejected	15,17%	50,70%

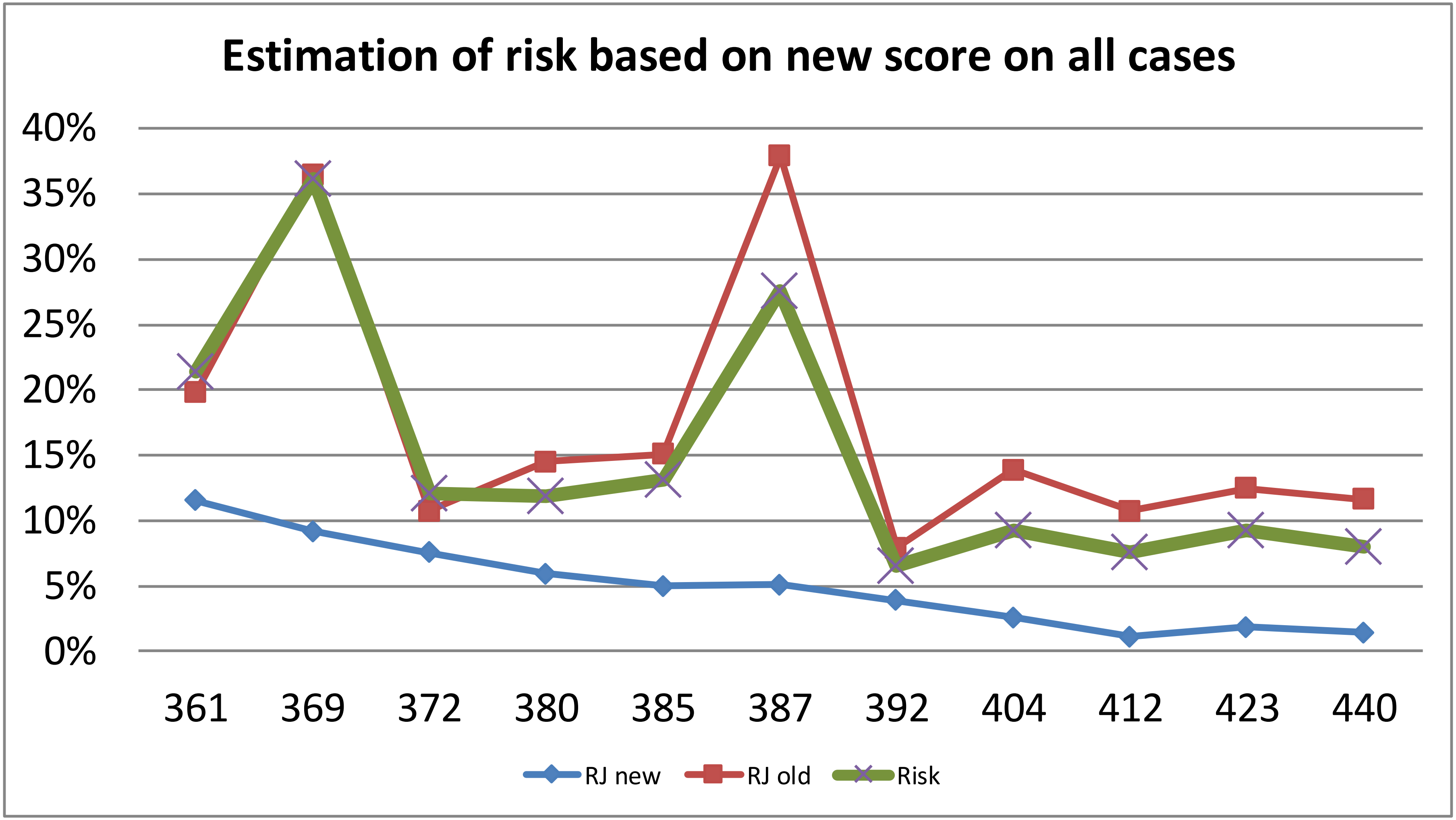
Reject Inference

- RJ New – new PD calibrated on new model only on accepted part
- RJ Old – PD on old model (PD Ins) calibrated only on accepted part
- PD Ins – old model build and calibrated on all cases (in case of strategy all)

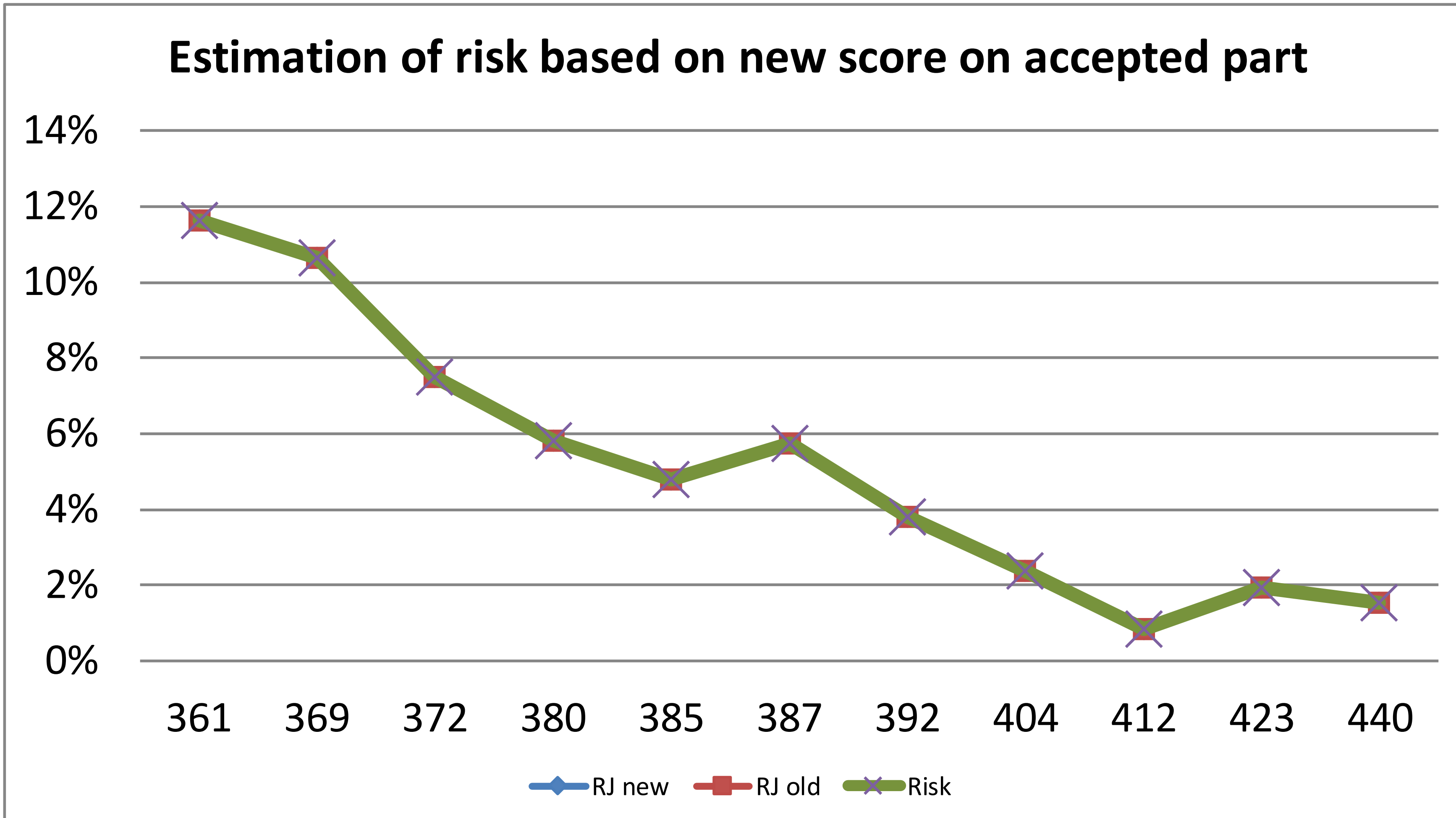
Reject Inference

Group - Condition		Pct			Risk			RJ new			RJ old			PD Ins		
		A	D	All	A	D	All	A	D	All	A	D	All	A	D	All
1	missing(ACT_CINS_N_STATC)	70,72%	76,68%	72,92%	5,61%	27,69%	14,17%	5,61%	5,71%	5,65%	5,61%	33,35%	16,37%	5,61%	22,53%	12,17%
2	not missing(ACT_CINS_N_STATC) and ACT_CINS_N_STATC <= 1	16,20%	16,57%	16,34%	3,52%	36,87%	16,01%	3,52%	6,08%	4,48%	3,52%	42,03%	17,94%	3,52%	31,03%	13,82%
3	1 < ACT_CINS_N_STATC	13,08%	6,74%	10,74%	2,20%	30,32%	8,71%	2,20%	2,32%	2,22%	2,20%	48,77%	12,98%	2,20%	35,39%	9,88%
All		100,00%	100,00%	100,00%	4,82%	29,39%	13,89%	4,82%	5,54%	5,09%	4,82%	35,83%	16,26%	4,82%	24,80%	12,20%

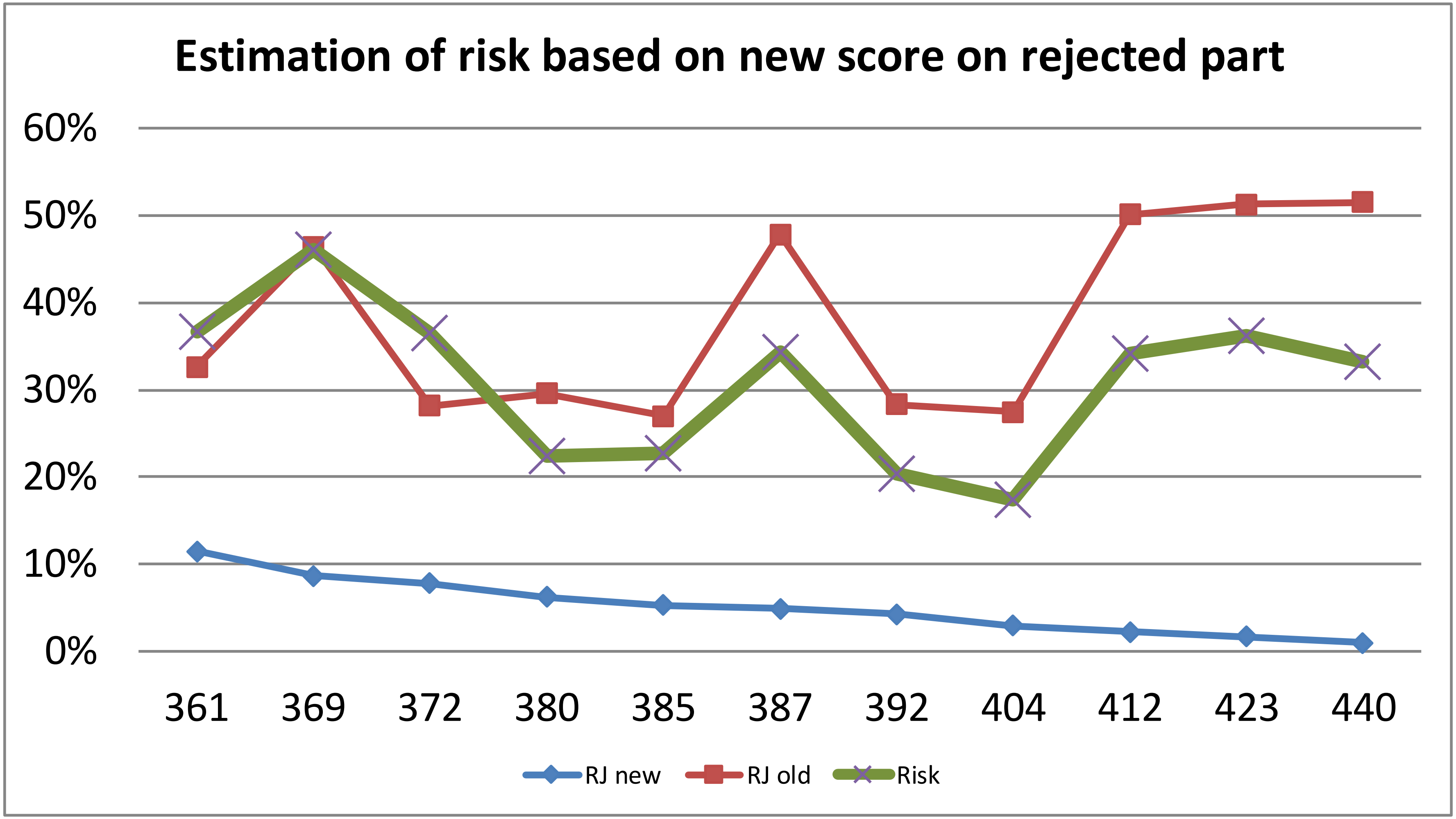
Reject Inference



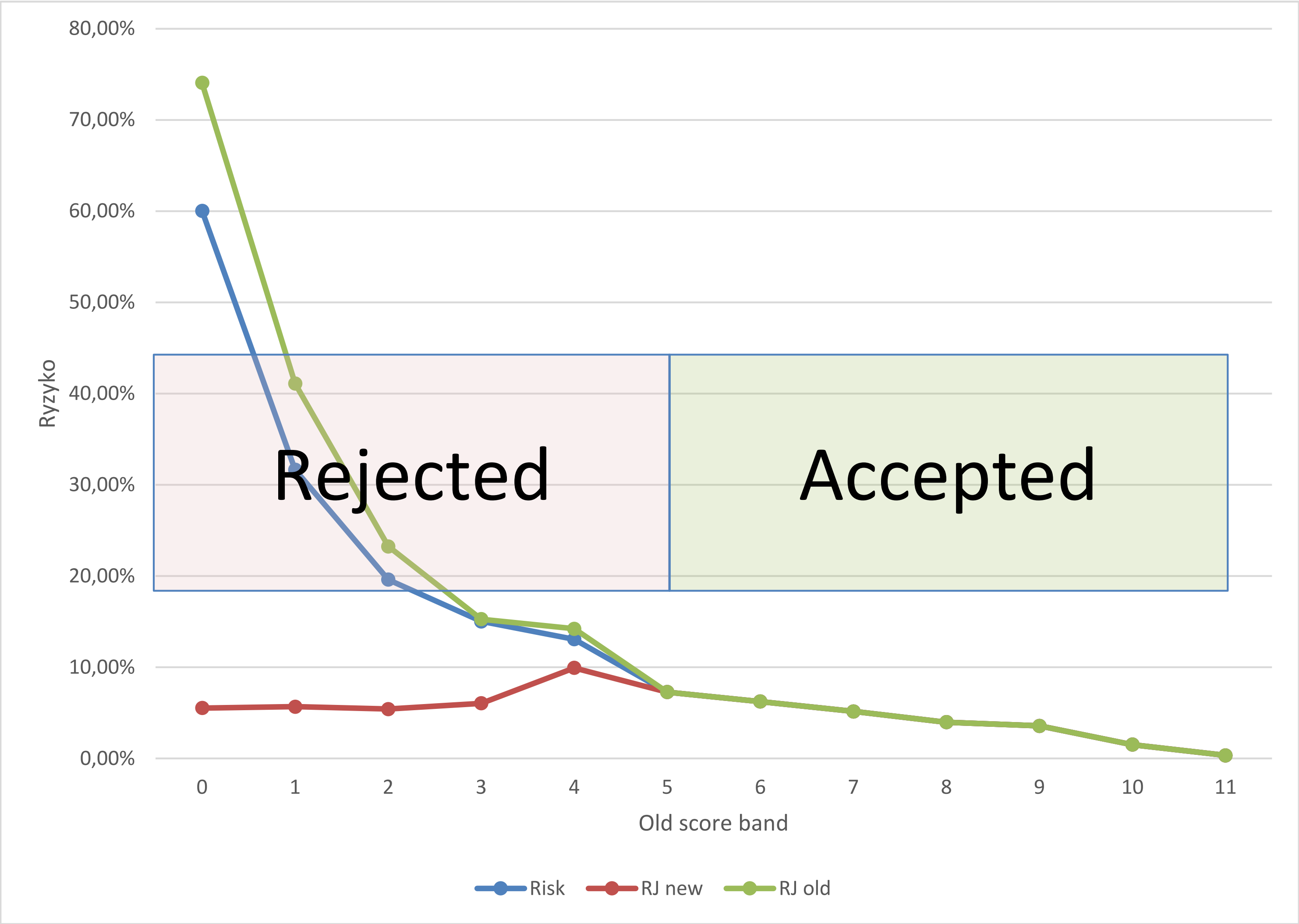
Reject Inference



Reject Inference

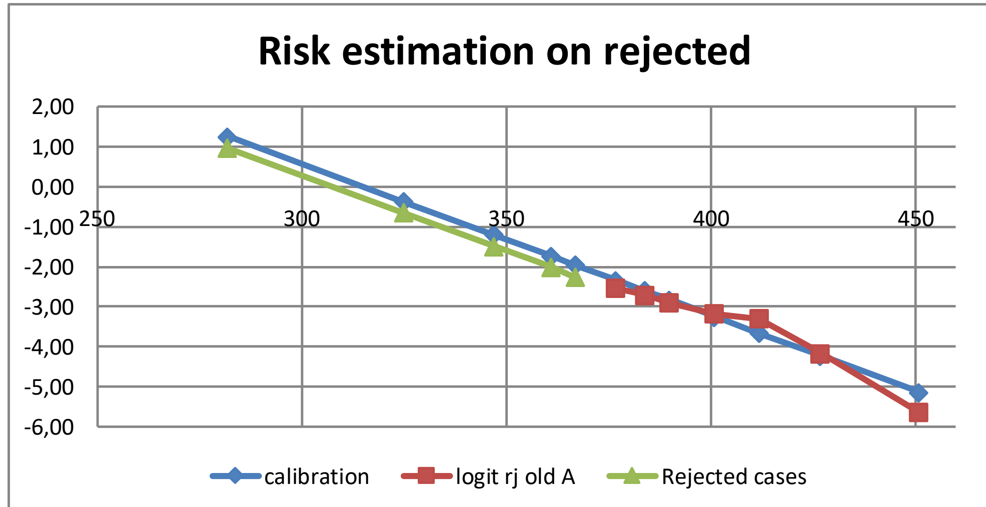


Reject Inference

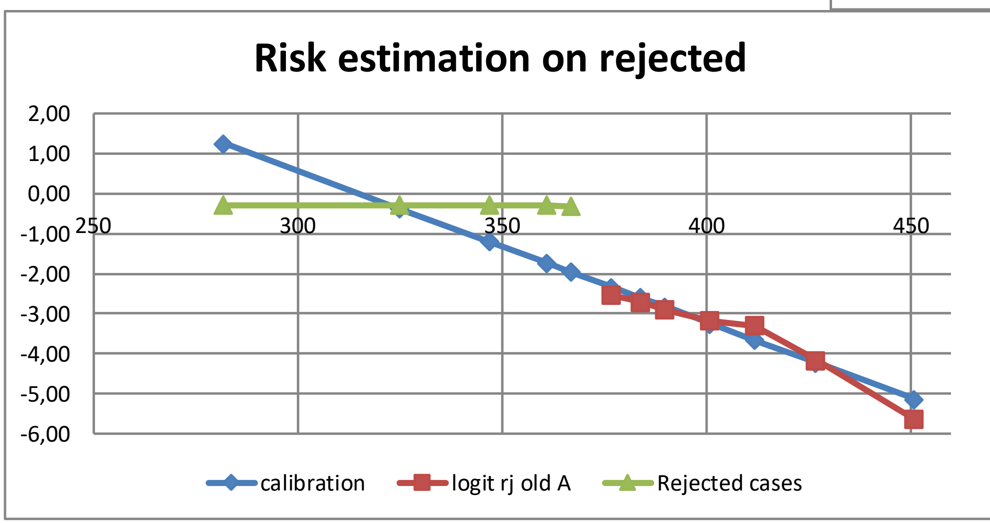


Reject Inference

Beta = 1



Beta = 0



Risk estimation – new target variable

PD=5%

- Exact method (two rows with weights):
  - row1 default=1 weight=5%
  - row2 default=0 weight=95%

- Simplified method (100 rows):
  - rows 1-5 default=1
  - rows 6-100 default=0

Reject Inference

- New PD based on old and new score and chosen calibration parameters results better estimation than first based only on new score and accepted cases

Decision	Risk	Estimation
A	4,82%	4,82%
D	29,39%	35,36%
All	13,89%	16,09%

Reject Inference

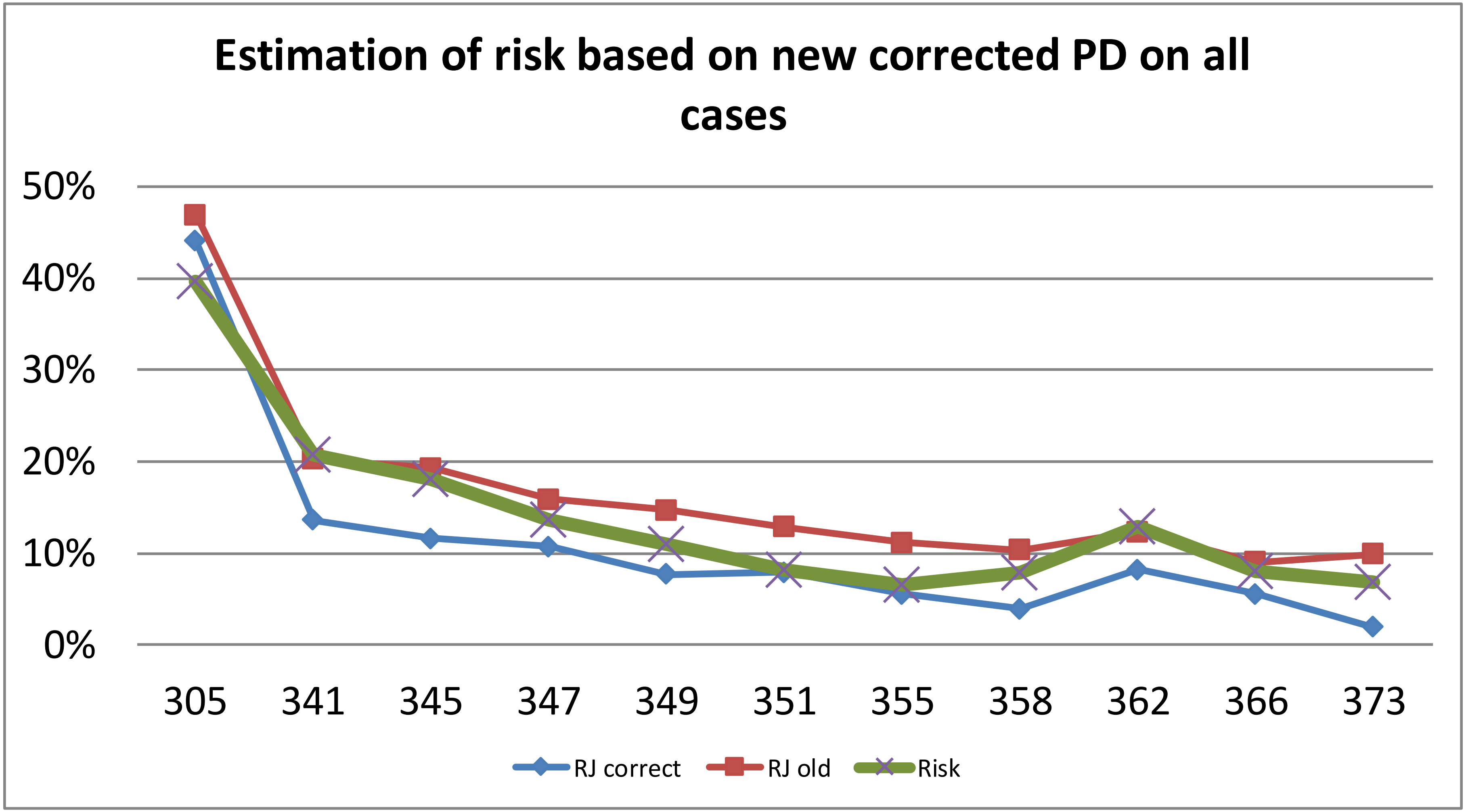
Target / Segments / Gini		New score	New score rj	Old score
default12	**Accepted**	36,15%	3,85%	41,29%
	All	24,73%	31,30%	65,55%
	Rejected	14,09%	17,14%	48,29%
default12_ind	**Accepted**	37,34%	4,17%	42,77%
	All	26,12%	32,23%	67,60%
	Rejected	15,17%	17,64%	50,70%

The first model All is build on the variables selected in KGB model.
Only risk and categories are changed. The list of variables is the same.

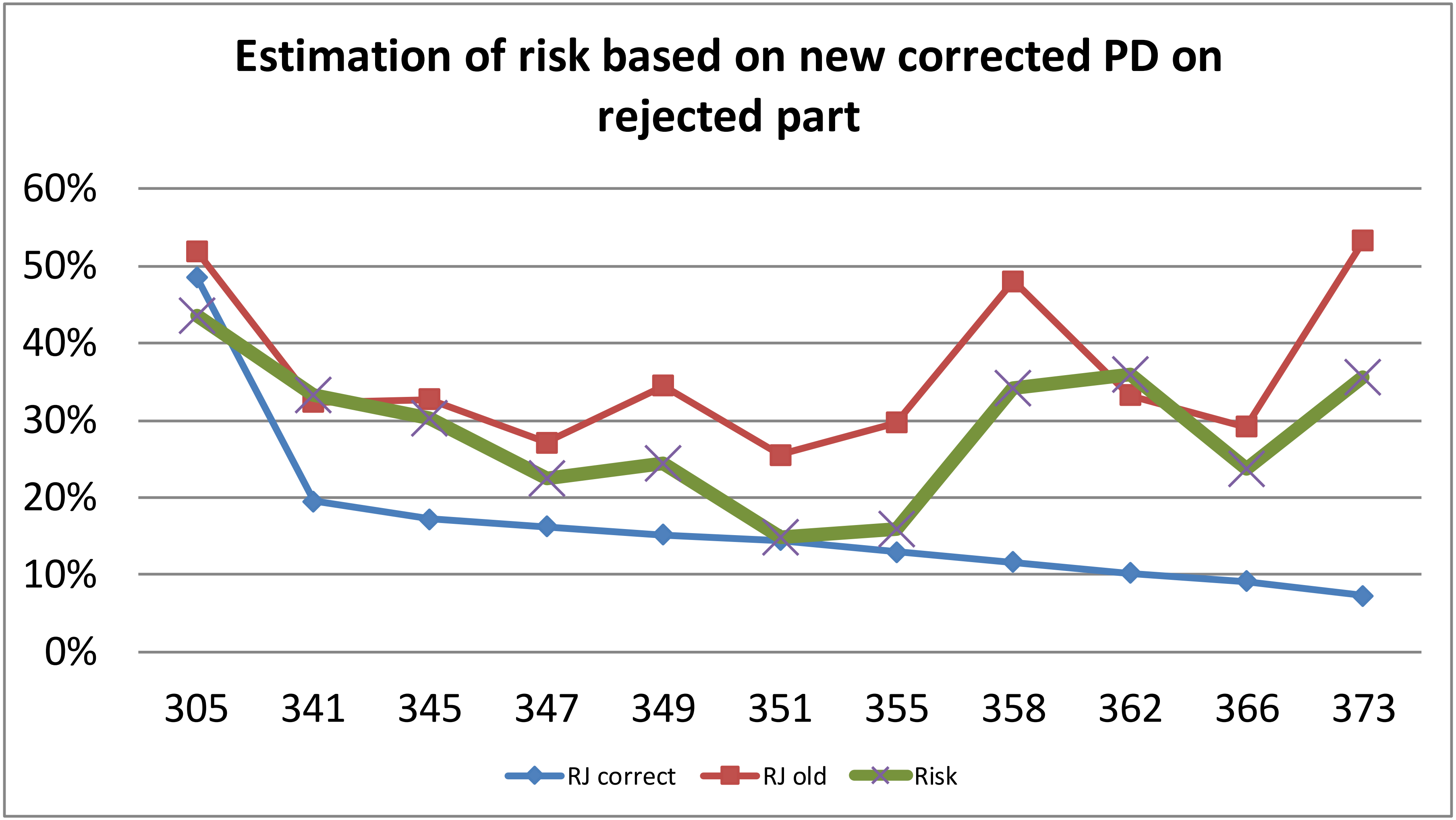
Reject Inference

Group - Condition		Pct			Risk			RJ all			Correct RJ new		
		A	D	All	A	D	All	A	D	All	A	D	All
1	**missing(ACT_CINS_N_STATC) or ACT_CINS_N_STATC <= 0**	72,58%	87,09%	77,94%	5,72%	28,23%	15,00%	5,72%	22,93%	12,82%	5,72%	22,70%	12,72%
2	**0 < ACT_CINS_N_STATC <= 2**	20,78%	10,13%	16,85%	2,69%	39,63%	10,89%	2,69%	38,55%	10,65%	2,69%	14,93%	5,41%
3	**2 < ACT_CINS_N_STATC**	6,64%	2,77%	5,21%	1,70%	28,25%	6,91%	1,70%	33,31%	7,90%	1,70%	13,04%	3,92%
All		100,00%	100,00%	100,00%	4,82%	29,39%	13,89%	4,82%	24,80%	12,20%	4,82%	21,64%	11,03%

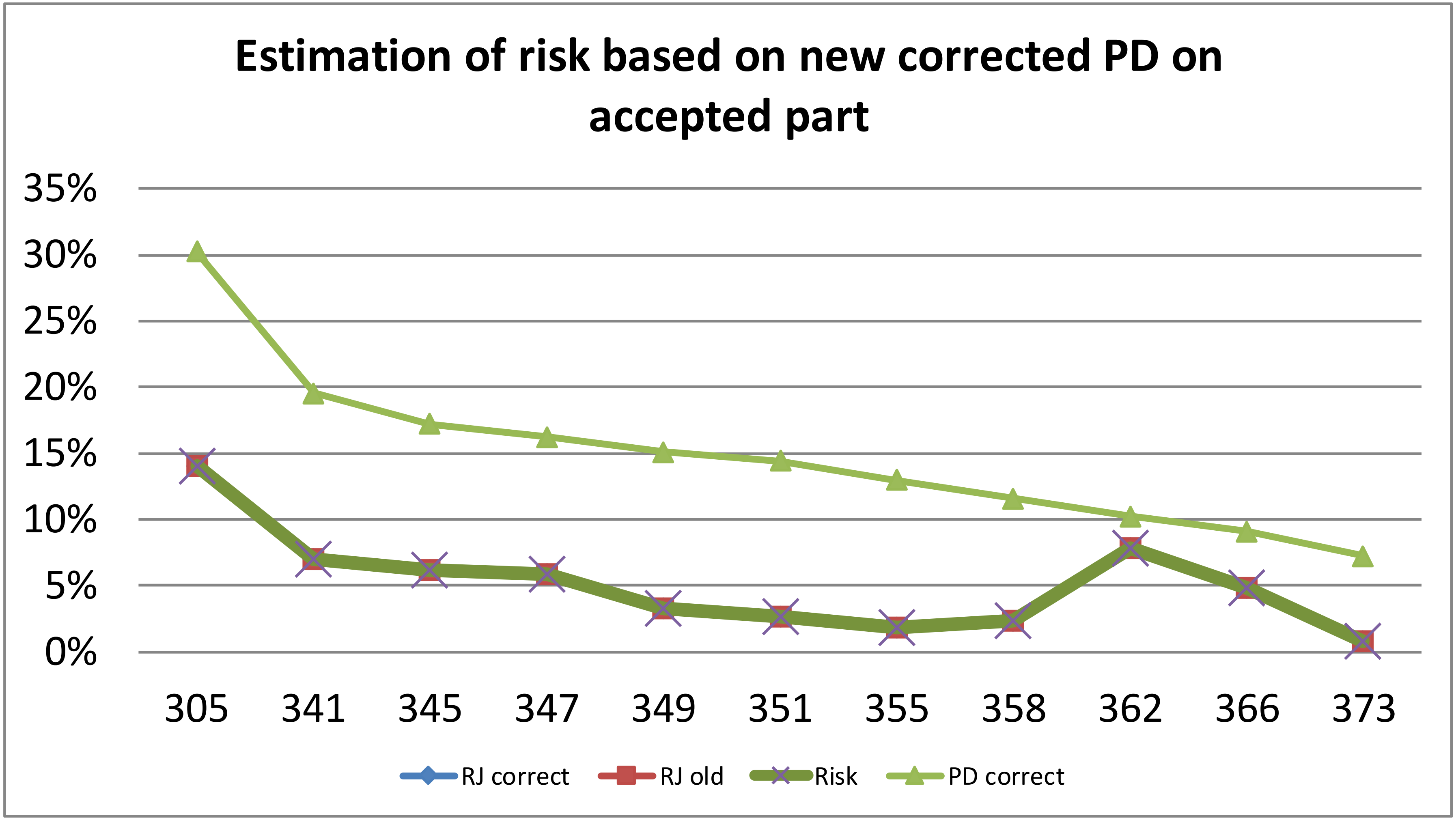
Reject Inference



Reject Inference



Reject Inference



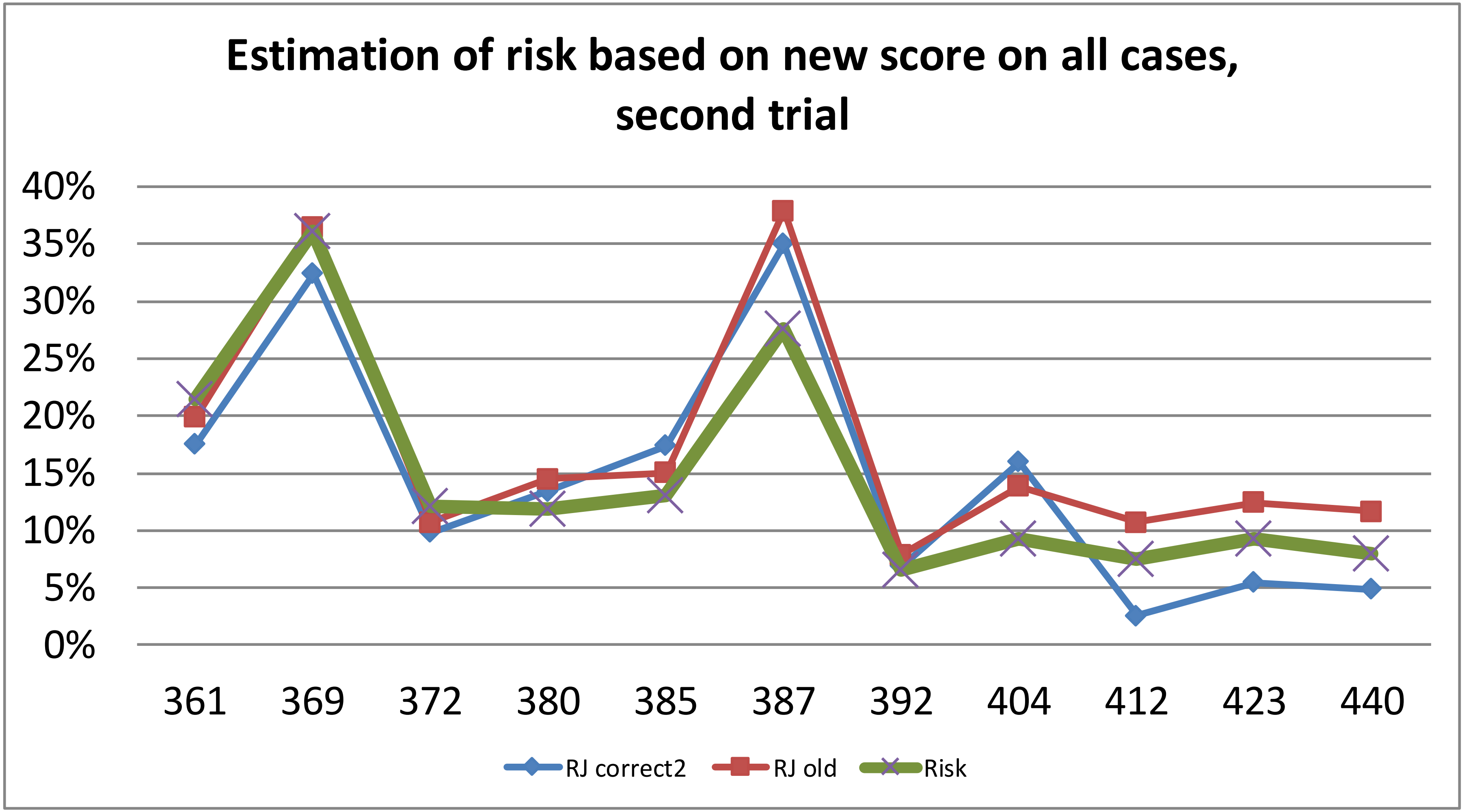
Reject Inference – second trial

- Model All is built on variables selection method starting from all.
- Model has 60% Gini.
- There are chosen different variables than on KGB model
- Model has better properties

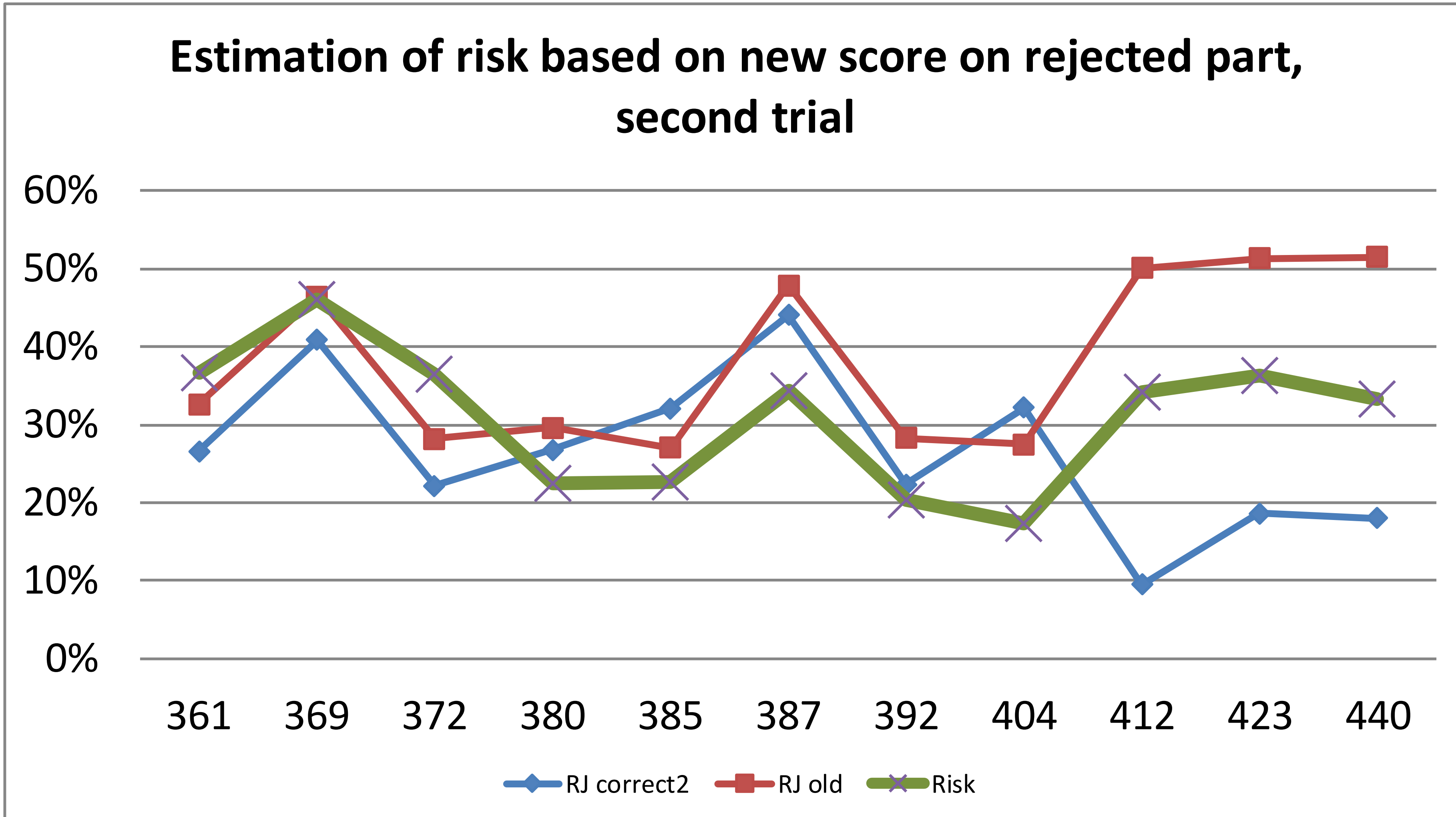
Reject Inference – druga próba

Target / Segments / Gini		New score	New score rj2	Old score
default12	**Accepted**	36,15%	32,81%	41,29%
	All	24,73%	54,08%	65,55%
	Rejected	14,09%	24,93%	48,29%
default12_ind	**Accepted**	37,34%	34,11%	42,77%
	All	26,12%	56,13%	67,60%
	Rejected	15,17%	26,53%	50,70%

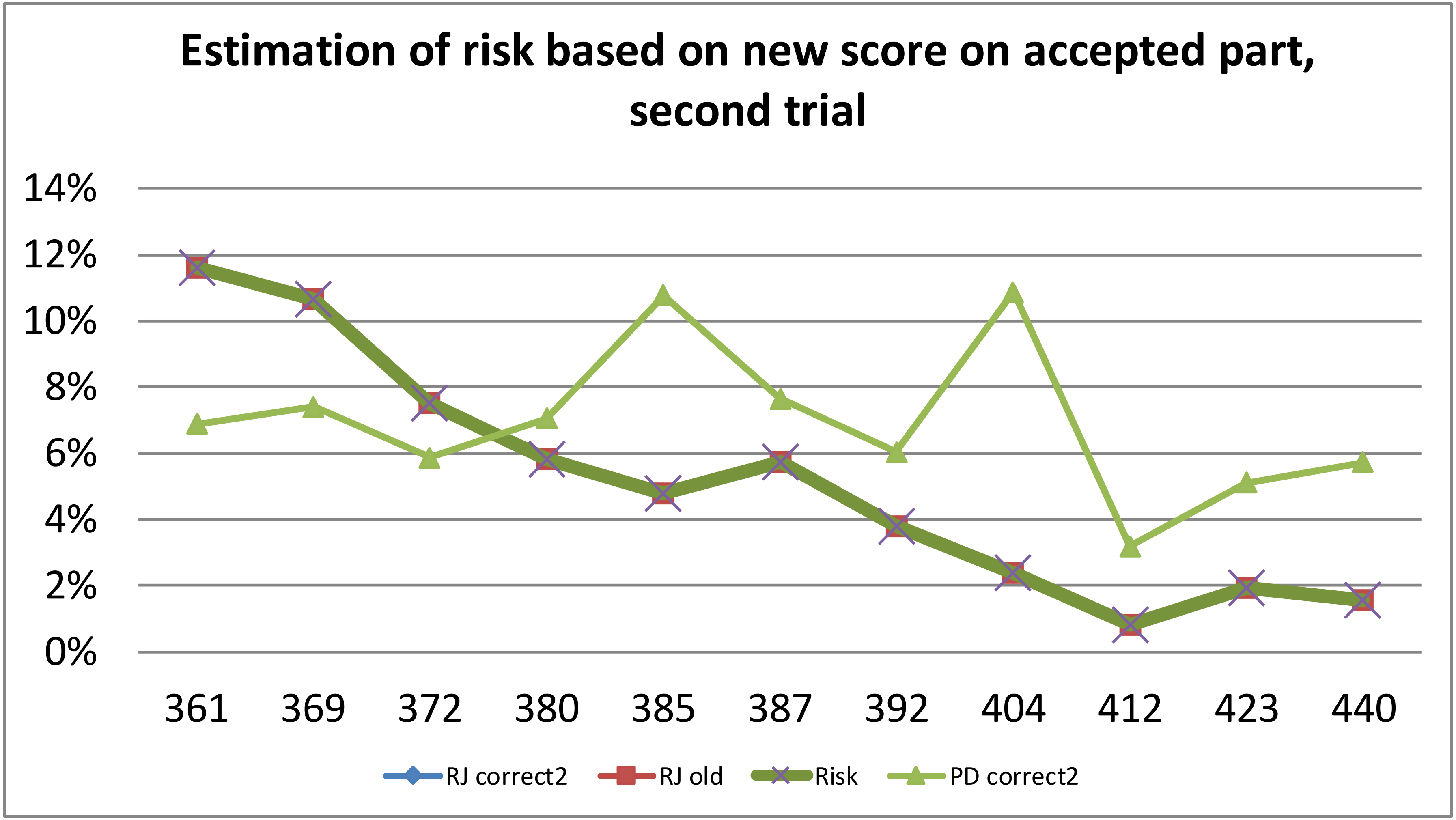
Reject Inference – druga próba



Reject Inference – druga próba



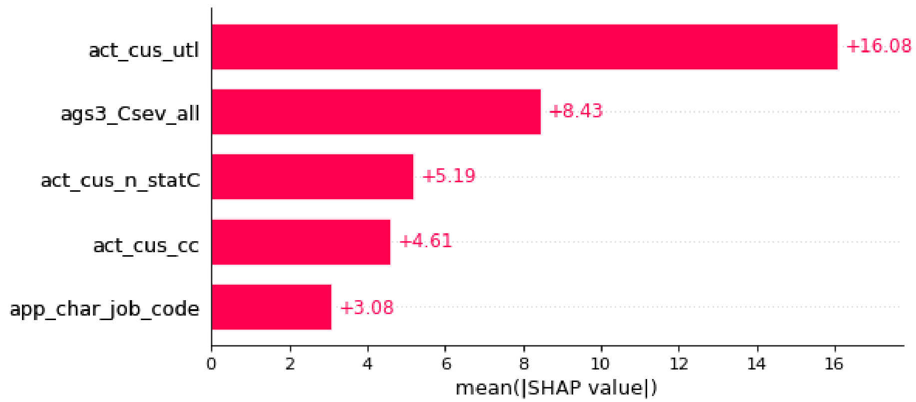
Reject Inference – druga próba

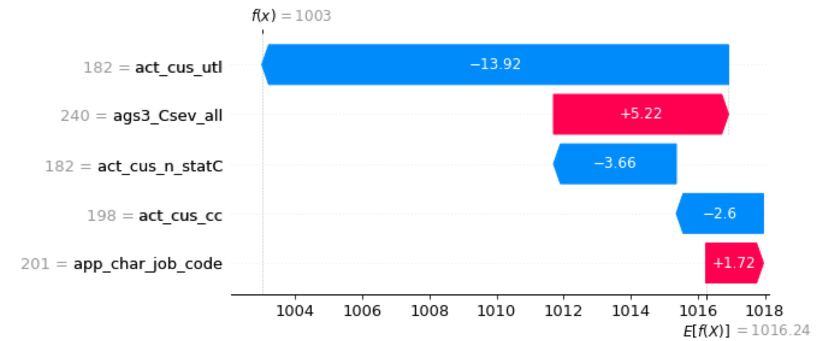


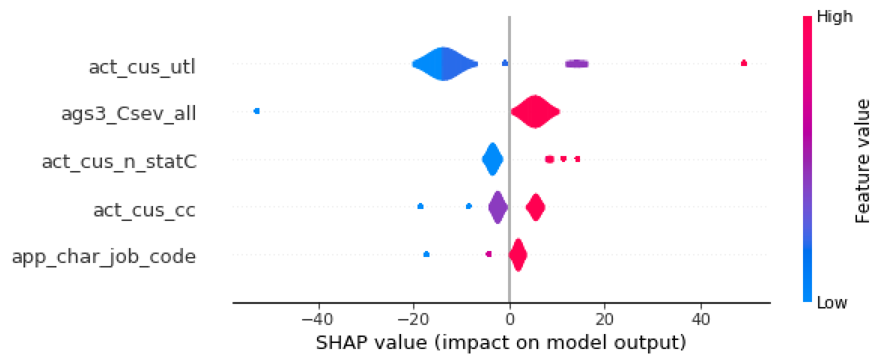
Reject Inference

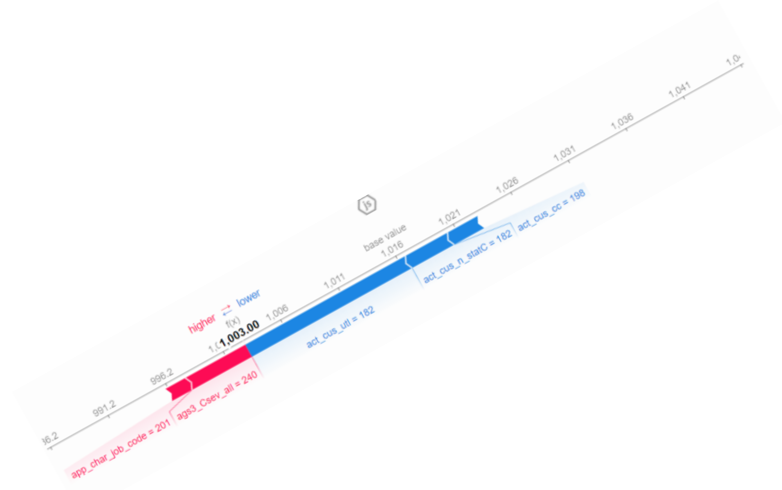
- Conclusion:
  - If you do not have a pattern of rejected customers, it is difficult to estimate risk
  - Can happen inverse event of risk profile, rejected customers can have inverse relation with the score
  - Reject Inference is always connected with huge estimation error
  - The best solutions:
    - Credit Bureau data
    - Open door strategy, not everyone under cut-off is rejected

XAI approach

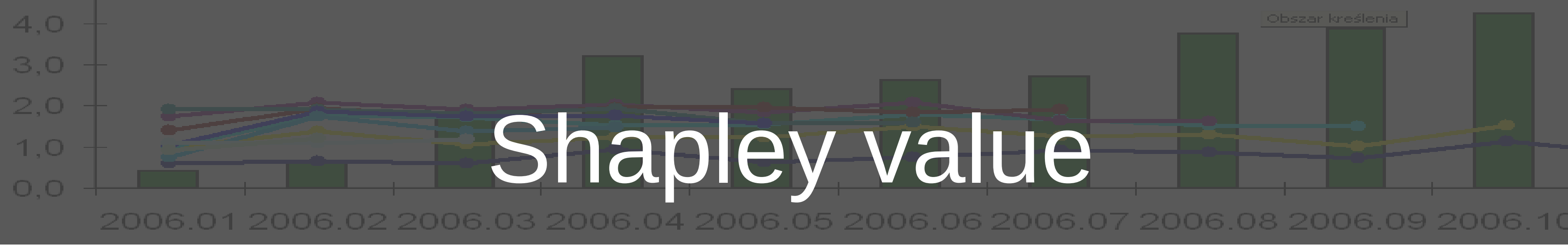








https://shap.readthedocs.io/en/latest/index.html



Shapley value

Formally, a **coalitional game** is defined as: There is a set *N* (of *n* players) and a function v that maps subsets of players to the real numbers: $v\colon 2^N \to \mathbb{R}$, with $v(\emptyset) = 0$, where $\emptyset$ denotes the empty set. The function v is called a characteristic function.

The function v has the following meaning: if *S* is a coalition of players, then $v(S)$, called the worth of coalition *S*, describes the total expected sum of payoffs the members of S can obtain by cooperation.

The Shapley value is one way to distribute the total gains to the players, assuming that they all collaborate. It is a "fair" distribution in the sense that it is the only distribution with certain desirable properties listed below. According to the Shapley value,[6] the amount that player *i* is given in a coalitional game (v, N) is

$$\begin{aligned}\varphi_i(v) &= \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\;(n-|S|-1)!}{n!}(v(S\cup\{i\})-v(S))\\ &= \sum_{S \subseteq N \setminus \{i\}} \binom{n}{1, |S|, n-|S|-1}^{-1}(v(S\cup\{i\})-v(S))\end{aligned}$$

where *n* is the total number of players and the sum extends over all subsets *S* of *N* not containing player *i*. Also note that $\binom{n}{a,b,c}$ is the multinomial coefficient. The formula can be interpreted as follows: imagine the coalition being formed one actor at a time, with each actor demanding their contribution $v(S\cup\{i\}) - v(S)$ as a fair compensation, and then for each actor take the average of this contribution over the possible different permutations in which the coalition can be formed.

An alternative equivalent formula for the Shapley value is:

$$\varphi_i(v) = \frac{1}{n!}\sum_R \left[v(P_i^R \cup \{i\}) - v(P_i^R)\right]$$

where the sum ranges over all $n!$ orders R of the players and P_i^R is the set of players in N which precede i in the order R. Finally, it can also be expressed as

$$\varphi_i(v) = \frac{1}{n}\sum_{S \subseteq N \setminus \{i\}} \binom{n-1}{|S|}^{-1}(v(S\cup\{i\})-v(S))$$

which can be interpreted as

$$\varphi_i(v) = \frac{1}{\text{number of players}}\sum_{\text{coalitions excluding } i} \frac{\text{marginal contribution of } i \text{ to coalition}}{\text{number of coalitions excluding } i \text{ of this size}}$$

References

Credit scoring in the context of interpretable machine learning. Theory and practice. Edited by D. Kaszyński, B. Kamiński, T. Szapiro. Pages 51-76, Oficyna Wydawnicza SGH, Warszawa 2020 (https://ssl-kolegia.sgh.waw.pl/pl/KAE/struktura/IE/struktura/ZWiAD/publikacje/Documents/Credit_scoring_in_the_context_of_interpretable_machine_learning.pdf)

Shapley, Lloyd S. (August 21, 1951). "Notes on the n-Person Game -- II: The Value of an n-Person Game" (PDF). Santa Monica, Calif.: RAND Corporation.

Notes on the n-Person Game — II: The Value of an n-Person Game (rand.org)

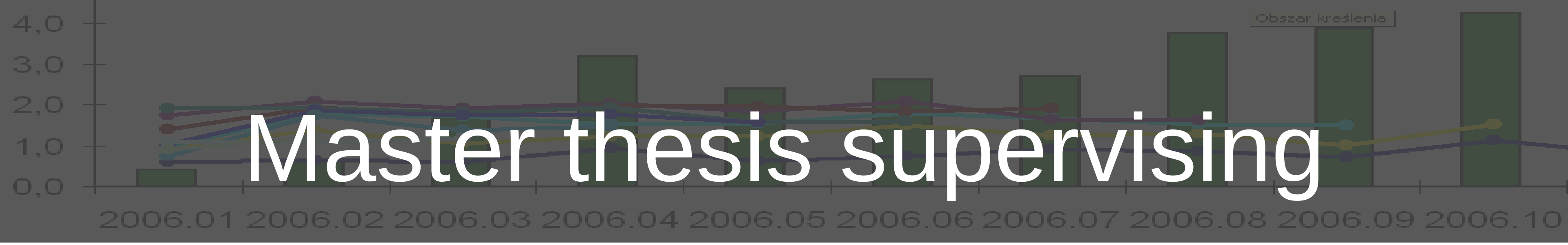
Logistic regression

- Please study it by yourself, or read the following simple document:
- https://christophm.github.io/interpretable-ml-book/logistic.html

Students for students

- We invite you to:
  - code improvements
  - developing tools and methods for automating the process
  - improving materials and updating knowledge

Master thesis supervising



- Scoring techniques and methods comparison
- Variable codding, binning
- Collinearity
- Reject Inference, MKS and MIV
- Crisis prediction and analysis, survival analysis
- Relation between predictive power and financial profit
- Model stability in the time
- Pricing management
- Variable monotonic property analysis

Statistical Methods & Business Analytics

- 2013 (International Year of Statistics 2013 www.statistics2013.org)
  - Advanced Analytics and Data Science

  www.analytics-conference.pl
- 2014
  - II Advanced Analytics and Data Science – 14.10

  http://www.sas.com/pl_pl/events/2014/advanced-analytics-and-data-science/index.html
- 2015
  - III Advanced Analytics and Data Science – 20.10

  http://www.sas.com/pl_pl/events/2015/advanced-analytics-and-data-science/speakers-and-panelists-2015.html