Example-Dependent Cost-Sensitive Credit Scoring

Alejandro Correa Bahnsen
Luxembourg University

Cost-Sensitive Classification

- Classification, in the context of machine learning, deals with the problem of predicting the class (y) of set of examples given their features (x)
- Minimize the misclassification

Confusion matrix

		True Class (y_i)	
		Fraud (y_i=1)	Legitimate (y_i=0)
Predicted class (c_i)	Fraud (c_i=1)	TP	FP
	Legitimate (c_i=0)	FN	TN

Cost-Sensitive Classification

- However, it is usually assumed that all errors leads to the same cost

Cost matrix

		True Class (y_i)	
		Fraud (y_i=1)	Legitimate (y_i=0)
Predicted class (c_i)	Fraud (c_i=1)	0	1
	Legitimate (c_i=0)	1	0

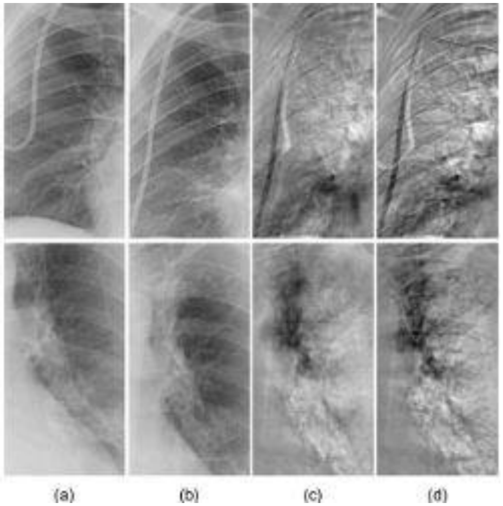
- Unrealistic in many real-world applications

Cost-Sensitive Classification

- FP = Sending a good email to the Spam folder
- FN = Failing to detect a spam email

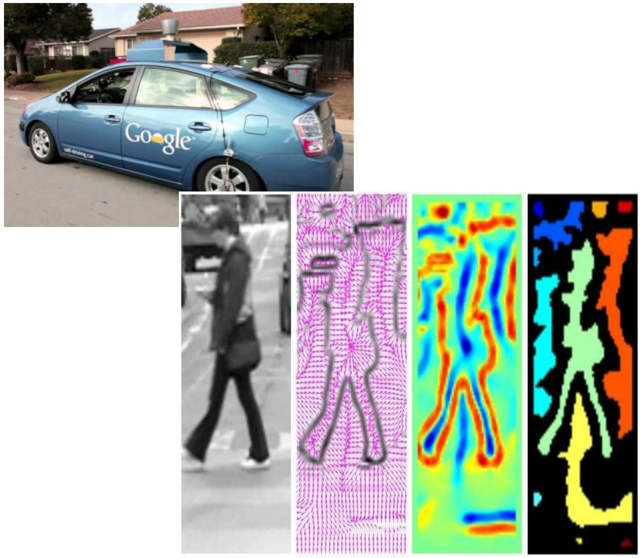
- FP = Declining a good transaction
- FN = Accepting a fraudulent transaction

Cost-Sensitive Classification



- FP = Wrongly detecting a tumor
- FN = Failing to detect a tumor

- FP = Confusing a pedestrian with the background
- FN = Failing to detecting a pedestrian



Cost-Sensitive Classification

SnT
securityandtrust.lu

Cost matrix

		True Class (y_i)	
		Fraud (y_i=1)	Legitimate (y_i=0)
Predicted class (c_i)	Fraud (c_i=1)	0	C_FP_i
	Legitimate (c_i=0)	C_FN_i	0

- In practice applications are cost-sensitive
- Furthermore, the cost varies between examples

Cost-Sensitive Classification

Cost matrix

		True Class (y_i)	
		Fraud (y_i=1)	Legitimate (y_i=0)
Predicted class (c_i)	Fraud (c_i=1)	0	C_FP_i
	Legitimate (c_i=0)	C_FN_i	0

- In practice applications are cost-sensitive
- Furthermore, the cost varies between examples

Credit Scoring

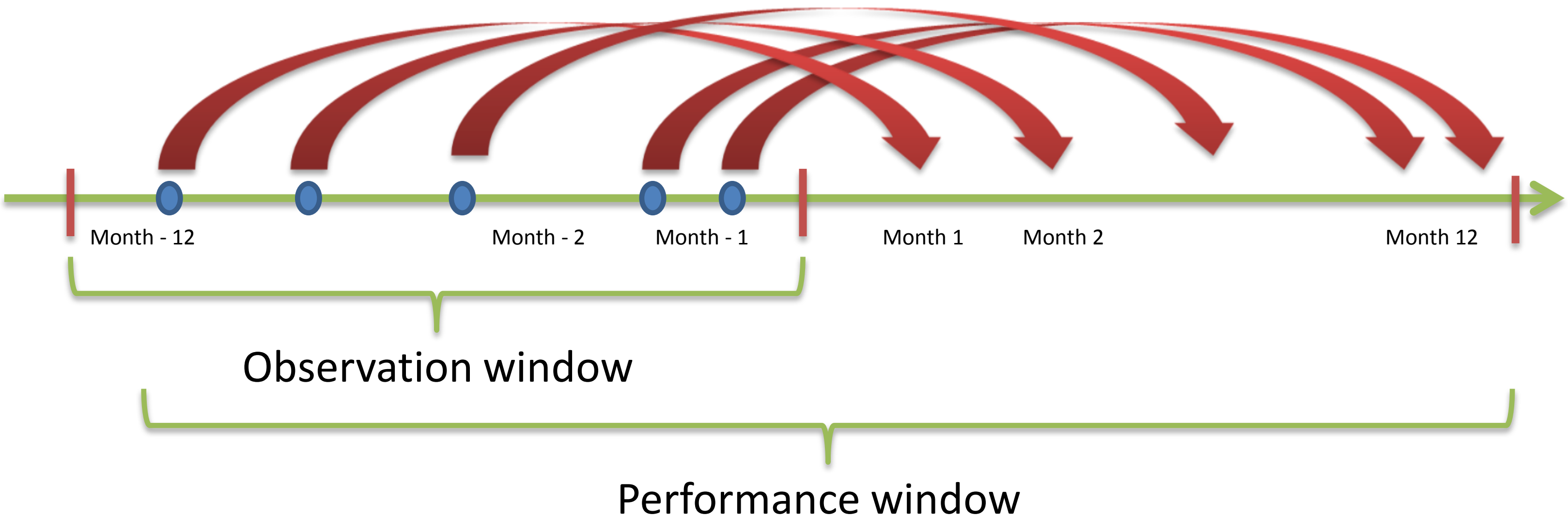
- Mitigate the impact of credit risk and make more objective and accurate decisions

- Estimate the risk of a customer defaulting his contracted financial obligation if a loan is granted, based on past experiences

- Different ML methods are used in practice, and in the literature: logistic regression, neural networks, discriminant analysis, genetic programing, decision trees, among others

Credit Scoring

- Construction of a credit score



- Applications during the observation window
- Y=1 if loan has days past due > 90 once during the 12 months after the application

Credit Scoring

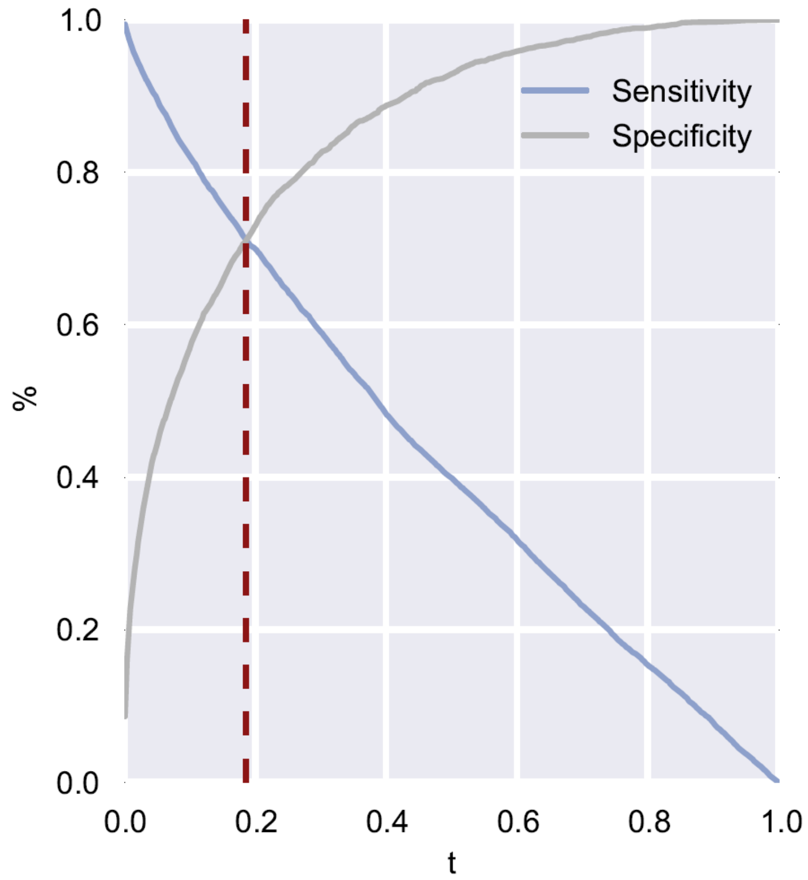
- Default probability $\hat{p} = P(y = 1|x)$.
- Classification $c(t) = 0,$ if $\hat{p} < t$
- Where t is the probability threshold

Credit Scoring

- Defining the threshold

- Where
  Sensitivity = Specificity

- Sensitivity is the true positive rate and specificity one minus the false positive rate.



Credit Scoring

securityandtrust.lu

- Evaluation of credit score models
  - Brier score
  - AUC
  - KS
  - F1-Score
  - Misclassification

- Nevertheless, none of these measures takes into account the business and economic realities that take place in credit scoring. Different costs that the financial institution has incur to acquire customers, or the expected profit due to a particular client, are not incorporated in the evaluation of the different models

UNIVERSITÉ DU LUXEMBOURG

Credit Scoring

- Evaluation of credit score models

Table 1. Credit scoring example-dependent cost matrix

		True Class (y_i)	
		Positive	Negative
Predicted Class (c_i)	Positive	0	$C^a_{FP_i} + C^b_{FP_i} + C^c_{FP_i}$
	Negative	$Cl_i \cdot lgd$	0

- Correct classification costs are assumed to be 0
- C_FP = losses if customer i defaults
- Cl_i is the credit line of customer i
- Lgd is the loss given default. Percentage of loss over the total credit line when the customer defaulted

Credit Scoring

- Evaluation of credit score models
- C_FN = $C^{a}_{FP_i} + C^{b}_{FP_i} + C^{c}_{FP_i}$
- $C^{a}_{FP_i} = r(Cl_i, int_{r_i}, n_i, int_{cf})$
- loss in profit by rejecting what would have been a good customer
- Where:
  - Int_r_i = interest rate of customer I
  - Int _cf = Financial intitution cost of funds
  - n_i = term of loan I
- Calculation of r in the appendix.

Credit Scoring

- Evaluation of credit score models
- C_FN = $C^{a}_{FP_i} + C^{b}_{FP_i} + C^{c}_{FP_i}$
- $C^{b}_{FP_i} = -r(Cl_{avg}, int_{r_i}, n_i, int_{cf}) \cdot (1 - \pi_1)$
- $C^{c}_{FP_i} = Cl_{avg} \cdot lgd \cdot \pi_1$
- assumption that the financial institution will not keep the money of the declined customer idle, but instead it will give a loan to an alternative customer
- Whom as an average customer has default probability equal to the prior default probability π_1

Credit Scoring

securityandtrust.lu

- Evaluation of credit score models

Table 1. Credit scoring example-dependent cost matrix

		True Class (y_i)	
		Positive	Negative
Predicted Class (c_i)	Positive	0	$C^a_{FP_i} + C^b_{FP_i} + C^c_{FP_i}$
	Negative	$Cl_i \cdot lgd$	0

$$C = \sum_{i=1}^{m} y_i(1 - c_i)C_{FN_i} + (1 - y_i)c_iC_{FP_i}.$$

UNIVERSITÉ DU LUXEMBOURG

Example-Dependent Cost-Sensitive Models

- Bayes minimum risk
  - decision model based on quantifying tradeoffs between various decisions using probabilities and the costs that accompany such decisions

- Risk of classification

$$R(c_i = 0|x_i) = C_{TN_i}(1 - \hat{p}_i) + C_{FN_i} \cdot \hat{p}_i,$$
$$R(c_i = 1|x_i) = C_{TP_i} \cdot \hat{p}_i + C_{FP_i}(1 - \hat{p}_i)$$

Example-Dependent Cost-Sensitive Models

- Bayes minimum risk

- If $R(c_i = 0|x_i) \leq R(c_i = 1|x_i)$ then $c(t) = 0$,

- Example-dependent threshold

$$t_{BMR_i} = \frac{C_{FP_i} - C_{TN_i}}{C_{FN_i} - C_{TN_i} - C_{TP_i} + C_{FP_i}}$$

Example-Dependent Cost-Sensitive Models

- Bayes minimum risk

- Calibration of probabilities
  - BMR method suffers when the estimated probabilities are not well calibrated
- Probabilities are calibrated using the ROC convex hull methodology described in the appendix

Example-Dependent Cost-Sensitive Models

- Threshold optimization

$$C = \sum_{i=1}^{m} y_i(1 - c_i)C_{FN_i} + (1 - y_i)c_i C_{FP_i}.$$

- Depends on c which depends on t
- $c(t) = 0,$ if $\hat{p} < t$

- Optimal threshold that minimizes the cost

$$t_{mc} = \underset{t}{\operatorname{argmin}}\, C(t).$$

Experiments

- Two publicly available datasets
  - Kaggle Credit dataset
  - PAKDD Credit dataset

- Contains information regarding customers income and debt from which the credit limit can be inferred, see appendix.

Table 2. Model parameters

Parameter	Kaggle Credit	PAKDD Credit
Interest rate (int_r)	4.79%	63.0%
Cost of funds (int_{cf})	2.94%	16.5%
Term (n) in months	24	24
Loss given default (lgd)	75%	75%

Experiments

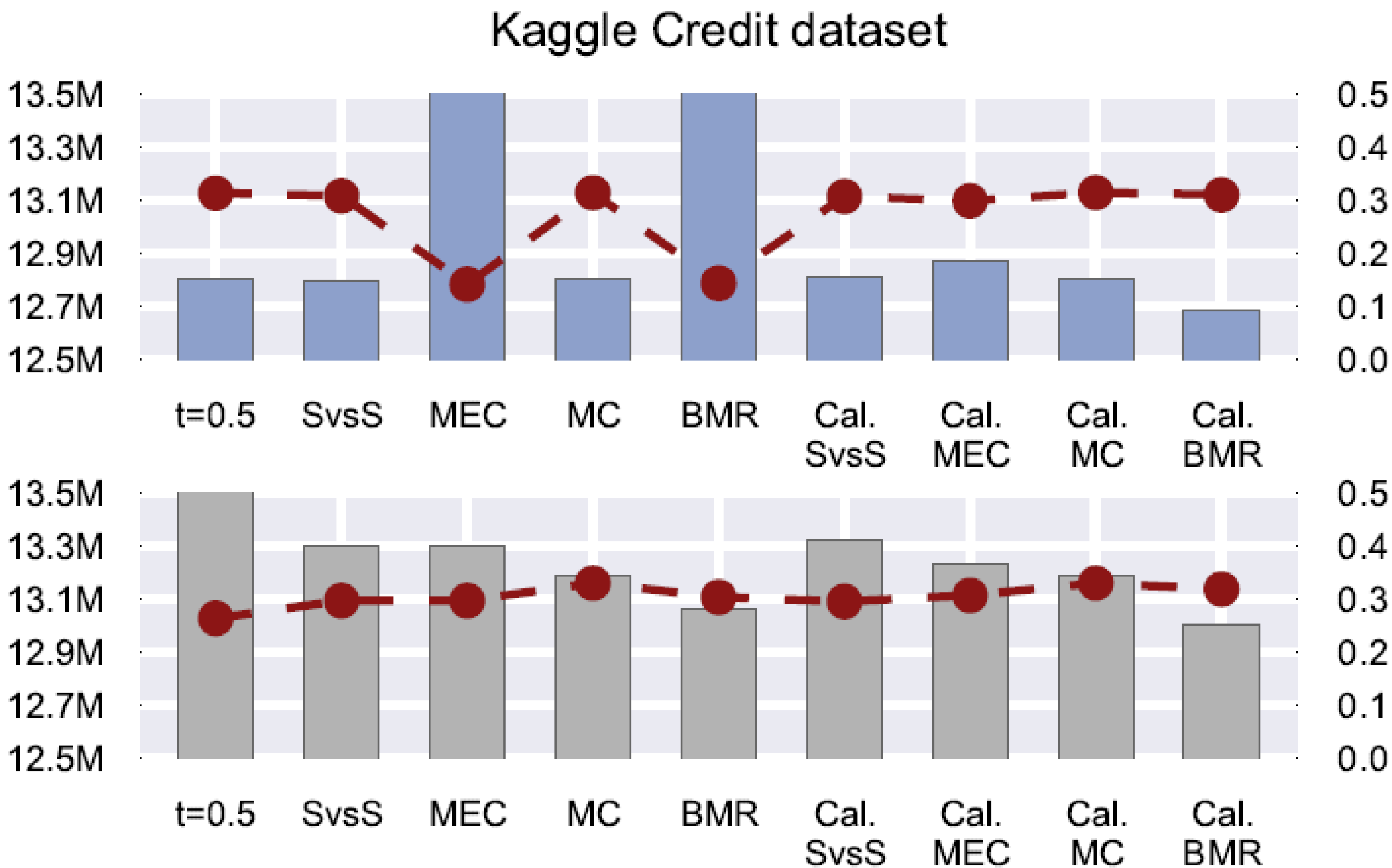
- Using Random Forest to estimate the probabilities

- Databases partitioned in training, validation and testing
- Each of them contain 50%, 25% and 25% of the total examples, respectively

- Under-sampled dataset
- Under-sampling of the negative examples is made in order to have a balanced class distribution on the training set

Experiments – Results Kaggle Credit

- Using Random Forest to estimate the probabilities



Experiments – Results Kaggle Credit

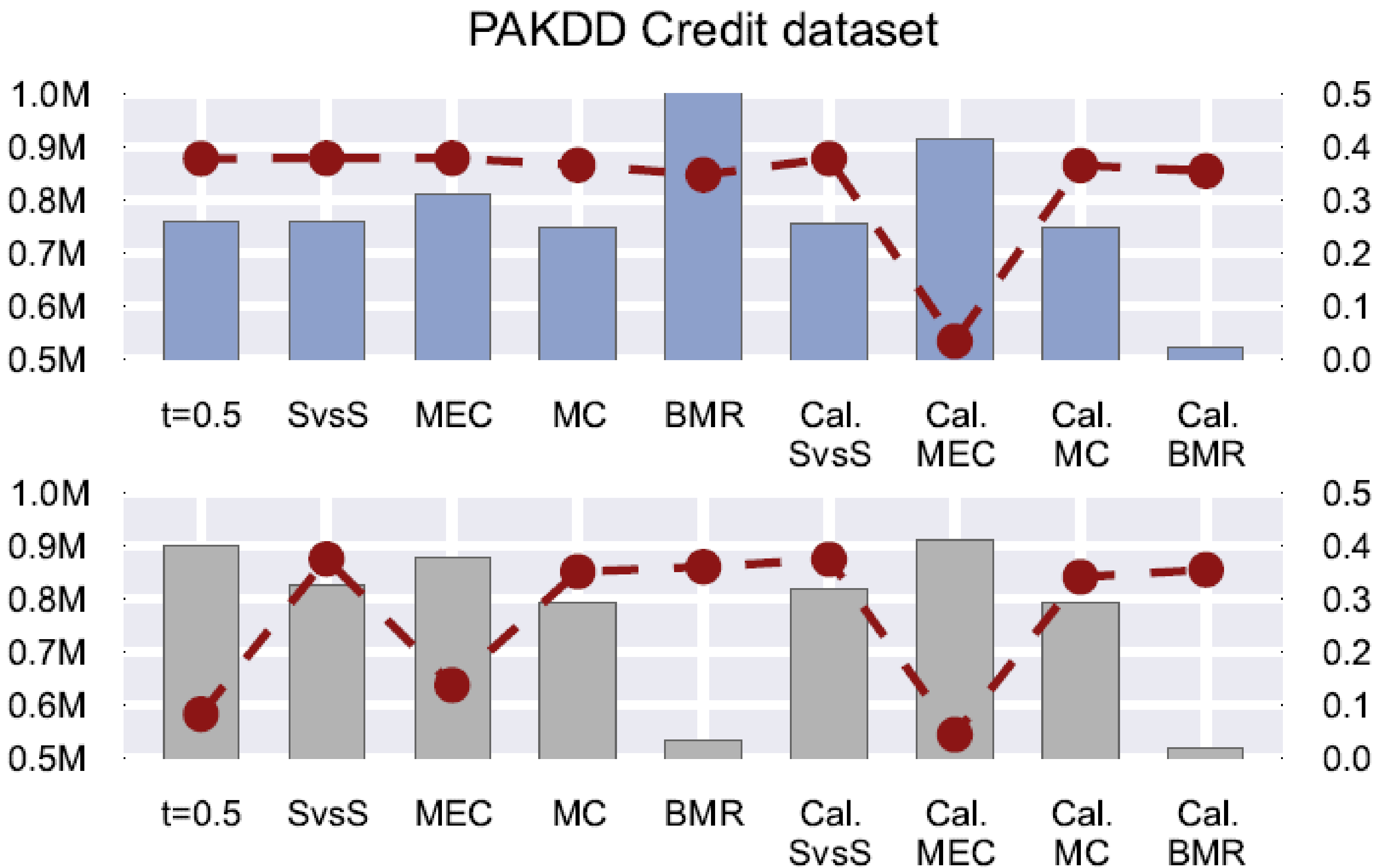
- Using Random Forest to estimate the probabilities

Kaggle Credit dataset

	Under-sampled				Training			
Method	Brier	Miscla	F_1-Score	Cost	Brier	Miscla	F_1-Score	Cost
$t = 0.5$	0.1596	0.2240	0.3136	12,805,180	0.0523	**0.0657**	0.264	21,465,633
t_{SvsS}	0.1596	0.2325	0.3085	12,798,294	0.0523	0.2420	0.2969	13,299,471
t_{ec}	0.1596	0.8062	0.1421	23,934,682	0.0523	0.2420	0.2969	13,299,471
t_{mc}	0.1596	0.2234	0.3144	12,805,749	0.0523	0.1941	**0.3301**	13,191,213
t_{BMR_t}	0.1596	0.7897	0.1440	23,680,770	0.0523	0.2239	0.3036	13,061,081
$Cal \cdot t_{SvsS}$	0.0528	0.2339	0.3074	12,815,640	0.0519	0.2447	0.2954	13,319,945
$Cal \cdot t_{ec}$	0.0528	0.2484	0.2991	12,874,154	0.0519	0.2281	0.3069	13,234,844
$Cal \cdot t_{mc}$	0.0528	0.2228	0.3147	12,803,906	0.0519	0.1941	**0.3301**	13,191,213
$Cal \cdot t_{BMR_t}$	0.0528	0.2158	0.3103	**12,687,521**	0.0519	0.1989	0.3187	13,004,645

Experiments – Results PAKDD Credit

- Using Random Forest to estimate the probabilities



Experiments – Results PAKDD Credit

- Using Random Forest to estimate the probabilities

PAKDD Credit dataset

	Under-sampled				Training			
Method	Brier	Miscla	F_1-Score	Cost	Brier	Miscla	F_1-Score	Cost
$t = 0.5$	0.2359	0.3955	0.3781	759,720	0.1541	0.2006	0.0830	898,871
t_{SvsS}	0.2359	0.3969	0.3786	761,215	0.1541	0.4013	0.3750	827,266
t_{ec}	0.2359	0.4663	**0.3787**	810,523	0.1541	0.2049	0.1376	878,585
t_{mc}	0.2359	0.3398	0.3658	746,866	0.1541	0.3102	0.3511	793,888
t_{BMR_t}	0.2359	0.7154	0.3475	1,026,159	0.1541	0.5175	0.3600	534,485
$Cal \cdot t_{SvsS}$	0.1527	0.3913	0.3781	756,714	0.1528	0.3846	0.3744	817,707
$Cal \cdot t_{ec}$	0.1527	0.1994	0.0345	915,892	0.1528	**0.1990**	0.0444	911,598
$Cal \cdot t_{mc}$	0.1527	0.3405	0.3652	747,720	0.1528	0.2939	0.3411	794,263
$Cal \cdot t_{BMR_t}$	0.1527	0.5142	0.3546	523,276	0.1528	0.5126	0.3547	**520,461**

Experiments

- Using:
  - Random Forest
  - logistic regression
  - gradient boosting
  - Gaussian naive Bayes
  - extra trees classifiers

- 10-fold cross-validation

Experiments – Results Kaggle Credit

securityandtrust.lu

Kaggle Credit dataset

		Decrease in cost (%)			Misclassification (%)			
Algorithm	Data	t_{mc}	$Cal \cdot t_{ec}$	$Cal \cdot t_{BMR_t}$	t_{SvsS}	t_{mc}	$Cal \cdot t_{ec}$	$Cal \cdot t_{BMR_t}$
Random forest	u	-0.06±0.17	-0.6±0.35	0.86±0.46	23.25±0.15	22.34±0.69	24.84±0.74	21.58±0.46
Logistic reg.	u	1.24±0.69	0.96±0.71	4.28±0.54	29.73±0.98	21.69±2.05	26.78±1.92	21.26±1.43
Gradient boost.	u	-0.48±0.39	-0.57±0.72	2.22±0.62	22.59±0.12	23.03±1.01	24.16±1.09	20.32±0.57
Naive Bayes	u	0.34±0.38	-0.68±0.49	1.76±0.51	34.49±0.58	29.43±2.21	36.06±2.52	29.72±1.26
Extra trees	u	0.02±0.36	0.06±0.26	1.03±0.44	23.97±0.1	21.96±0.68	23.69±1.04	21.78±0.53
Random forest	t	0.81±0.36	0.48±0.38	2.22±0.35	24.2±0.18	19.41±0.89	22.81±1.19	19.89±0.46
Logistic reg.	t	2.23±0.92	1.13±0.54	3.99±1.03	36.68±0.62	26.78±6.58	36.42±4.59	28.92±4.92
Gradient boost.	t	-0.47±0.48	-0.75±0.56	1.91±0.7	22.41±0.11	21.61±0.91	23.74±1.07	19.89±0.47
Naive Bayes	t	2.39±0.51	0.59±0.5	2.17±0.72	34.47±0.91	25.55±1.01	32.65±1.11	28.05±0.79
Extra trees	t	1.05±0.54	0.25±0.79	1.4±0.44	25.3±0.22	19.64±0.78	24.17±1.51	21.29±0.55

UNIVERSITÉ DU LUXEMBOURG

Experiments – Results PAKDD Credit

securityandtrust.lu

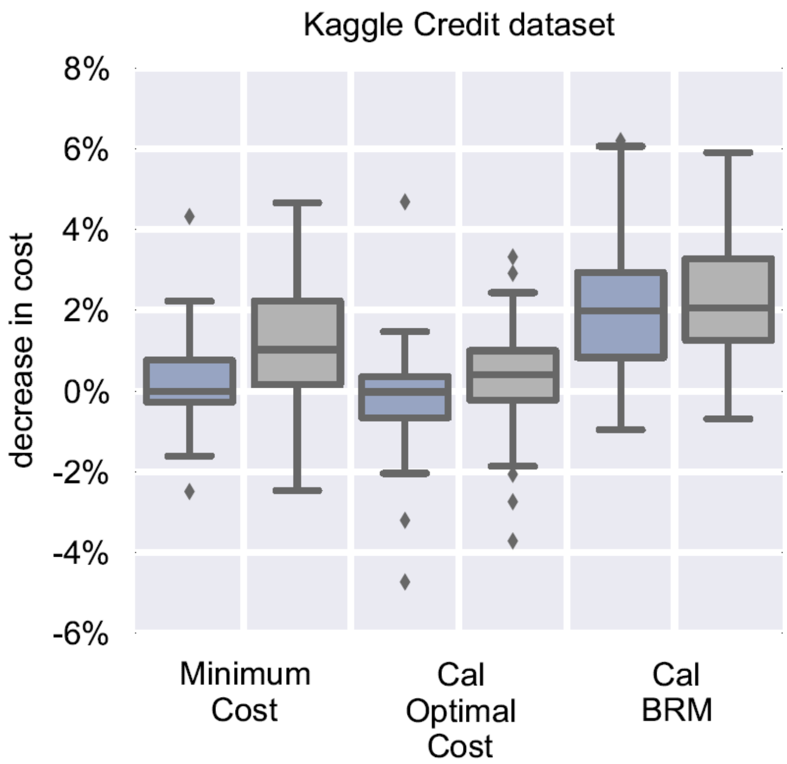
PAKDD Credit dataset

		Decrease in cost (%)			Misclassification (%)			
Algorithm	Data	t_{mc}	$Cal \cdot t_{ec}$	$Cal \cdot t_{BMR_t}$	t_{SvsS}	t_{mc}	$Cal \cdot t_{ec}$	$Cal \cdot t_{BMR_t}$
Random forest	u	1.84±0.75	-20.5±2.8	31.1±1.96	39.69±0.27	33.98±1.15	19.94±0.14	51.42±0.26
Logistic reg.	u	38.8±0.87	38.8±0.87	63.6±0.78	77.67±0.18	19.9±0.16	19.9±0.16	53.67±0.21
Gradient boost.	u	0.17±0.6	-26.6±1.92	27.4±1.56	38.32±0.16	36.73±1.39	20.04±0.18	50.8±0.22
Naive Bayes	u	0.07±0.9	-39.2±1.88	20.5±1.23	40.15±0.29	44.77±1.07	19.9±0.16	52.16±0.24
Extra trees	u	5.14±1.36	-8.44±2.31	36.6±1.53	41.44±0.32	30.17±1.15	19.98±0.17	51.96±0.27
Random forest	t	4.0±1.05	-10.3±2.21	37.0±1.22	40.13±0.46	31.02±1.62	19.9±0.17	51.26±0.23
Logistic reg.	t	38.8±0.87	38.8±0.87	63.6±0.78	77.67±0.18	19.9±0.16	19.9±0.16	53.67±0.21
Gradient boost.	t	-0.24±0.32	-28.1±2.7	26.6±1.54	37.96±0.31	35.21±1.48	19.96±0.17	50.73±0.22
Naive Bayes	t	-0.34±0.83	-39.3±2.19	20.3±0.81	40.23±0.29	44.9±0.92	19.92±0.16	52.3±0.28
Extra trees	t	7.62±0.93	-0.44±1.3	41.4±0.87	41.73±0.25	28.61±1.17	19.92±0.15	52.09±0.27

UNIVERSITÉ DU LUXEMBOURG

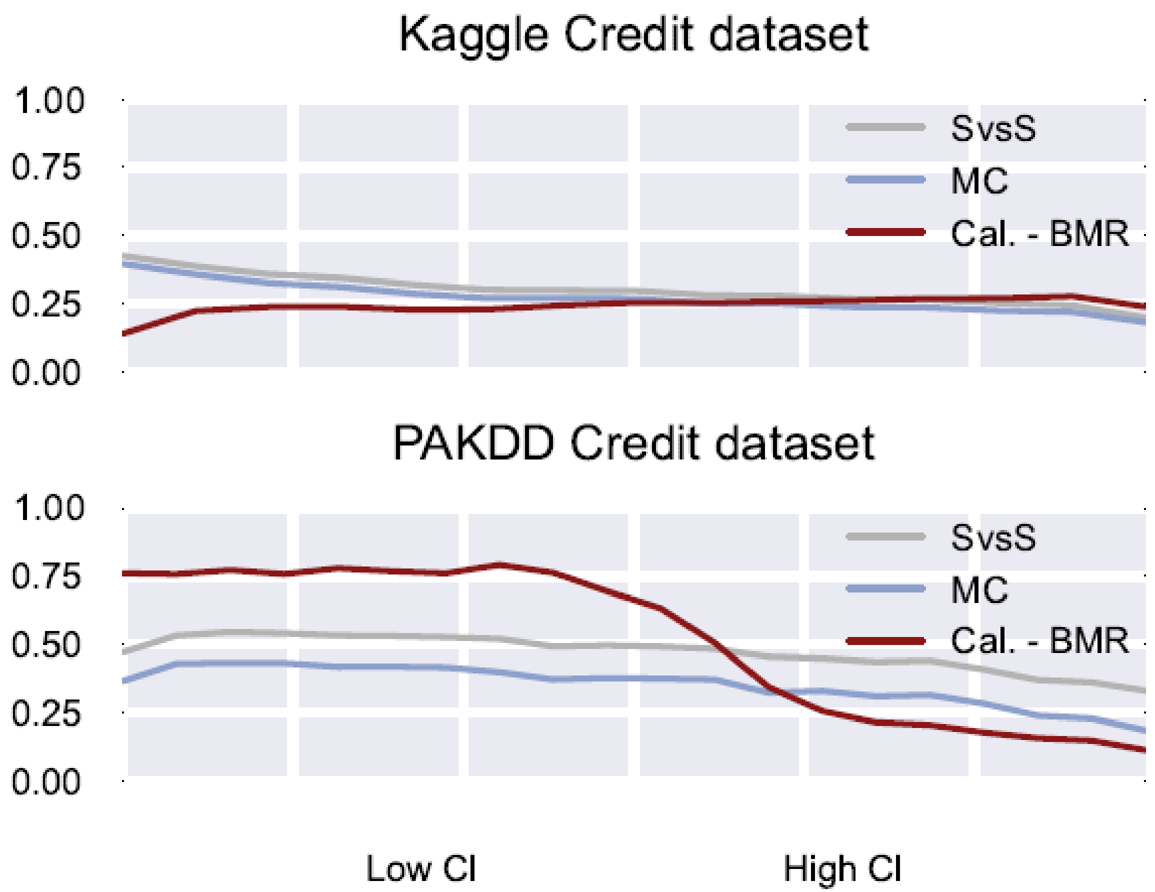
Experiments – Results

- Comparison of average decrease in cost between algorithms



Experiments – Results PAKDD Credit

- Comparison of the misclassification of the different models against different percentiles the credit limit Cl



Conclusion

- Selecting models based on traditional statistics does not give the best results in terms of cost
- Models should be evaluated taking into account real financial costs of the application
- Algorithms should be developed to incorporate those real financial costs

Thank you!

Contact information

Alejandro Correa Bahnsen

University of Luxembourg

Luxembourg

al.bahnsen@gmail.com

http://www.linkedin.com/in/albahnsen

http://www.slideshare.net/albahnsen

Appendix

A Calculation of a loan profit

The profit r is calculated as the present valaue of the difference between the financial institution gains and expenses, given the credit line Cl_i, the term n_i and the financial institution lending rate int_{r_i} for customer i, and the financial institution of cost funds int_{cf}.

$$r(Cl, int_r, n, int_{cf}) = PV(A(Cl, int_r, n), int_{cf}, n) - Cl, \quad (9)$$

with A being the customer monthly payment and PV the present value of the monthly payments, which are calculated using the time value of money equations [15],

$$A(Cl, int, n) = Cl\frac{int(1+int)^n}{(1+int)^n - 1}, \quad (10)$$

$$PV(a, int, n) = \frac{a}{int}\left(1 - \frac{1}{(1+int)^n}\right). \quad (11)$$

Appendix

B Calculation of the credit limit

There exist several strategies to calculate the Cl_i depending on the type of loans, the state of the economy, the current portfolio, among others [1, 15]. Nevertheless, given out lack of information regarding the specific business environment of both datasets, we simply define Cl_i as

$$Cl_i = \min\left(k \cdot Inc_i, Cl_{max}, Cl_{max}(debt_i)\right). \tag{12}$$

We fix $k = 3$ since it is the average personal loans request related to monthly income, and Cl_{max} to 25,000 Euros, which is the maximum amount for personal loans without collateral as reported by several financial institutions. Lastly, the maximum credit line given the current debt is calculated as the maximum credit limit such that the current debt ratio plus the new monthly payment does not surpass the customer monthly income. It is calculated as

$$Cl_{max}(debt_i) = PV\left(Inc_i \cdot MP_{min}(debt_i), int_r, n\right), \tag{13}$$

and

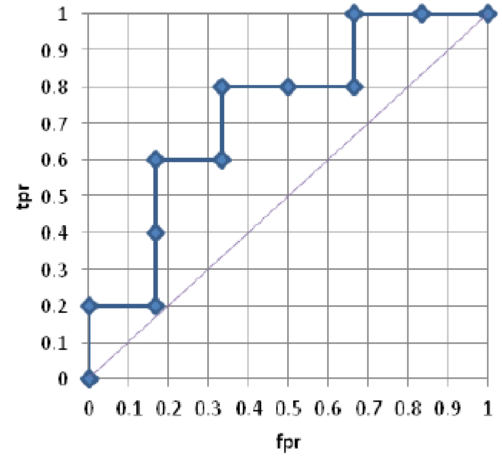
$$MP_{min}(debt_i) = \min\left(\frac{A(k \cdot Inc_i, int_r, n)}{Inc_i}, 1 - debt_i\right). \tag{14}$$

Appendix

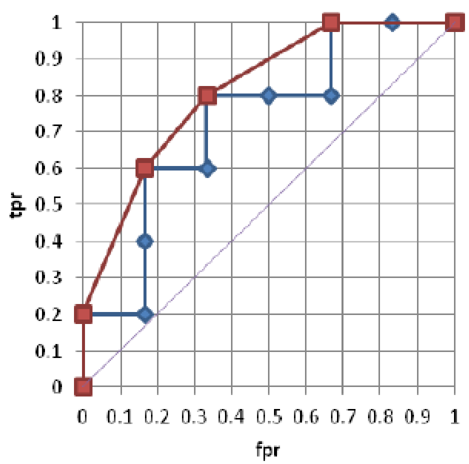
Probability Calibration

Probability	Label
0.0	0
0.1	1
0.2	0
0.3	0
0.4	1
0.5	0
0.6	1
0.7	1
0.8	0
0.9	1
1.0	1

(a) Set of probabilities and their respective class label



(b) ROC curve of the set of probabilities



(c) Convex hull of the ROC curve

Probability	Cal Probability
0.0	0
0.1	0.333
0.2	0.333
0.3	0.333
0.4	0.5
0.5	0.5
0.6	0.666
0.7	0.666
0.8	0.666
0.9	1
1.0	1

(d) Calibrated probabilities

Figure 1: Estimation of calibrated probabilities using the ROC convex hull [9].