Fonds National de la Recherche Luxembourg

SnT securityandtrust.lu

Advances in Machine Learning for Credit Card Fraud Detection

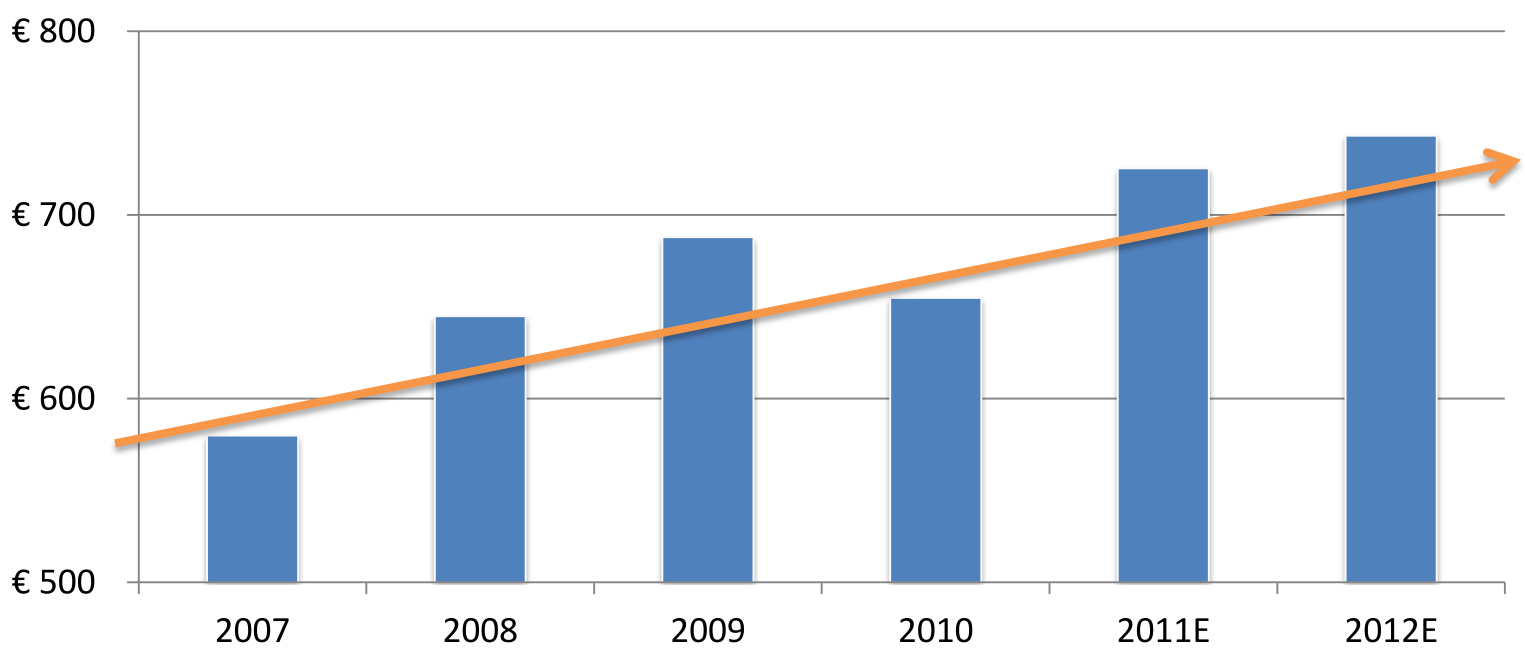
May 14, 2014

Alejandro Correa Bahnsen

uni.lu UNIVERSITÉ DU LUXEMBOURG

Introduction

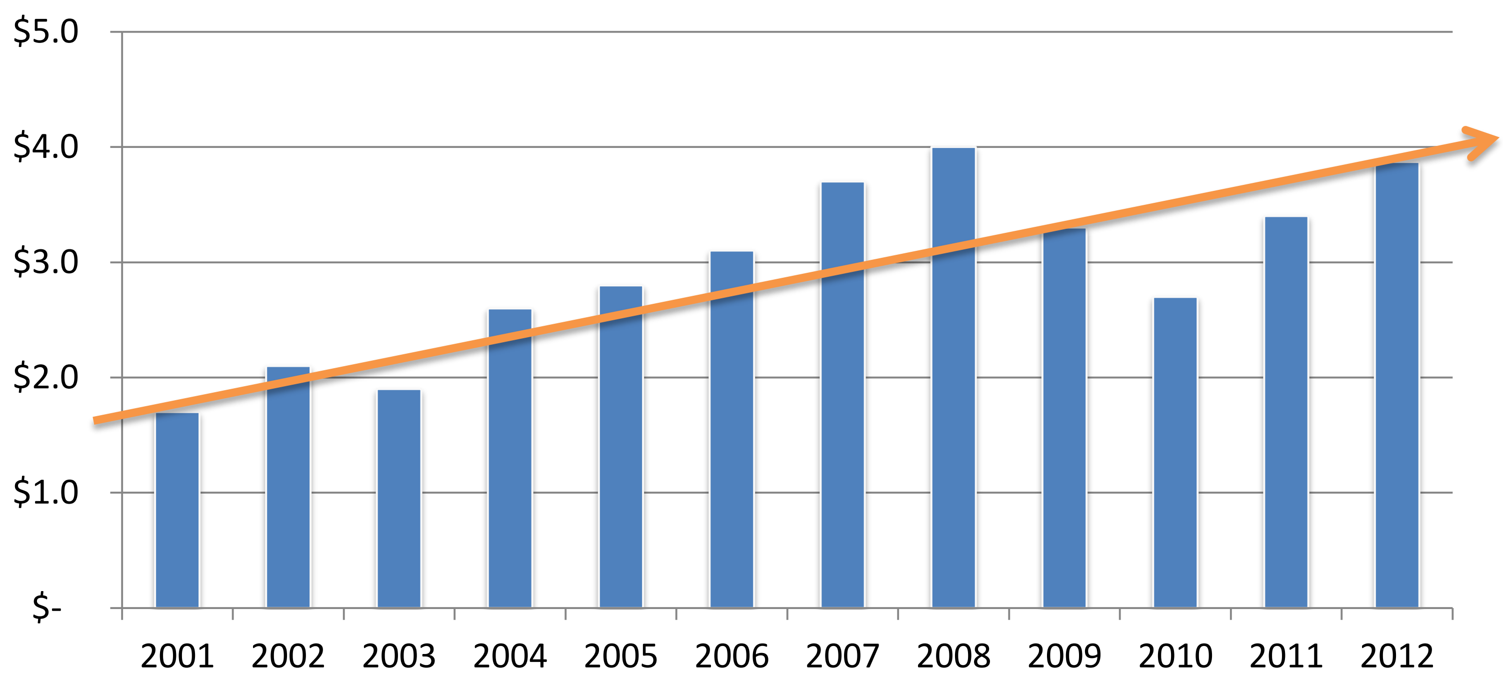
Europe fraud evolution
Internet transactions (millions of euros)



Introduction

US fraud evolution

Online revenue lost due to fraud (Billions of dollars)



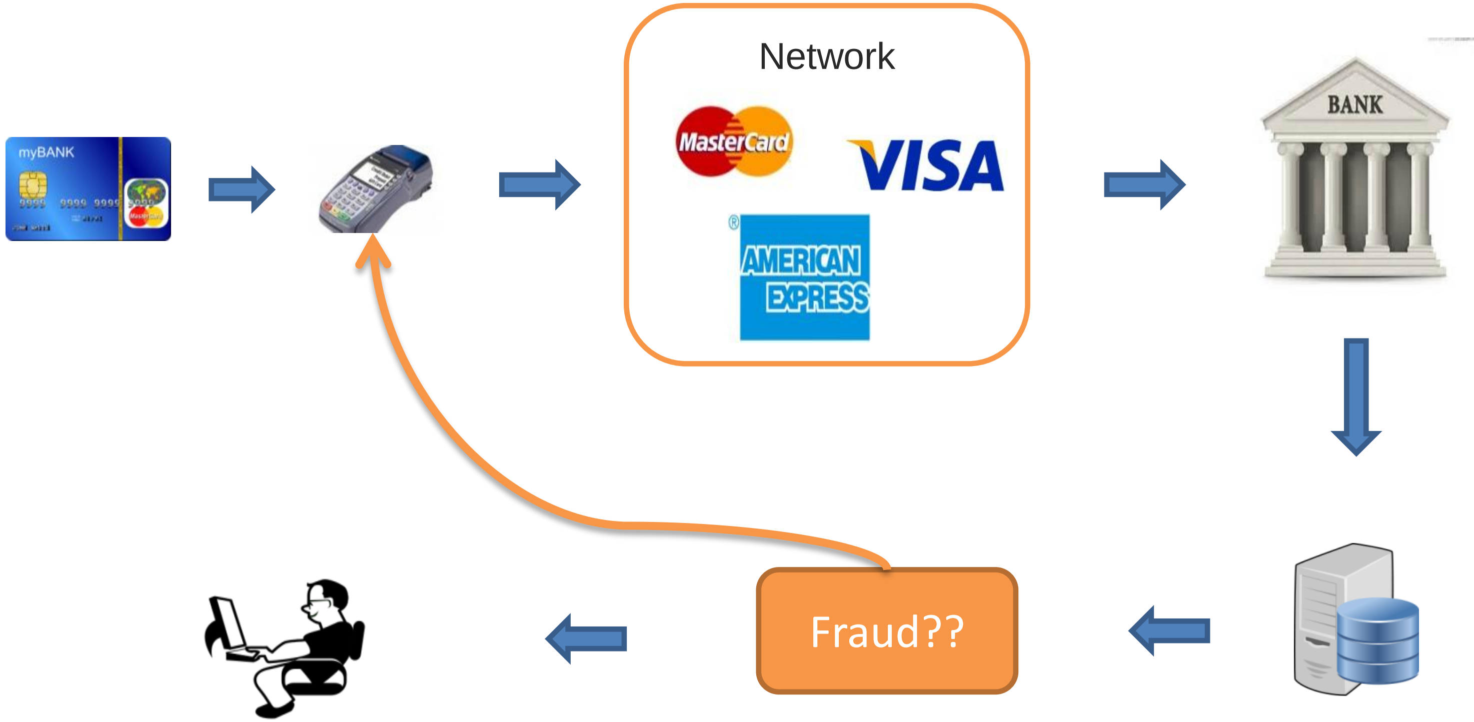
Introduction

- Increasing fraud levels around the world
- Different technologies and legal requirements makes it harder to control
- Lack of collaboration between academia and practitioners, leading to solutions that fail to incorporate practical issues of credit card fraud detection:
    - Financial comparison measures
    - Huge class imbalance
    - Low-latency response time

Agenda

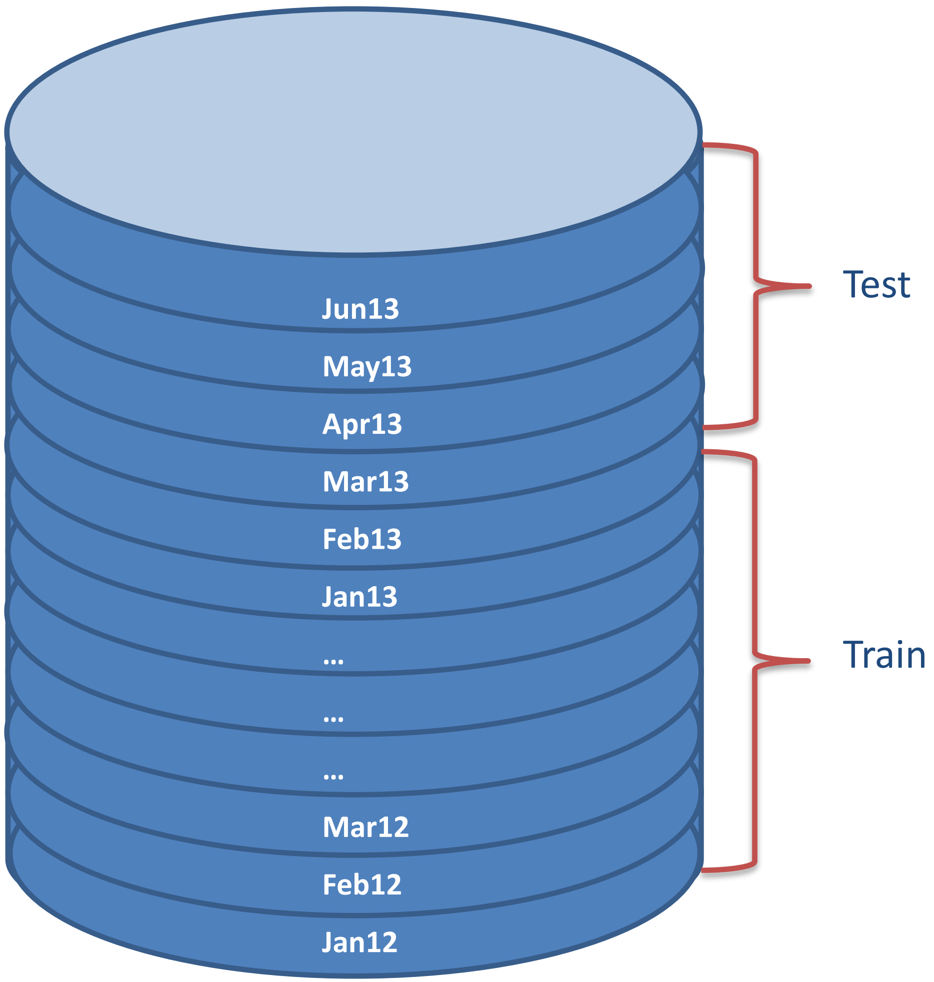
- Introduction
- Database
- Evaluation
- Algorithms
  - Cost-sensitive logistic regression
  - Bayes Minimum Risk
  - Example-dependent cost-sensitive decision tree
- Conclusions & Future Work

Simplify transaction flow



Data

- Larger European card processing company
- Jan2012 – Jun2013 card present transactions
- 1,638,772 Transactions
- 3,444 Frauds
- 0.21% Fraud rate
- 205,542 EUR lost due to fraud on test dataset



Data

Raw attributes

TRXID	Client ID	Date	Amount	Location	Type	Merchant Group	Fraud
1	1	2/1/12 6:00	580	Ger	Internet	Airlines	No
2	1	2/1/12 6:15	120	Eng	Present	Car Rent	No
3	2	2/1/12 8:20	12	Bel	Present	Hotel	Yes
4	1	3/1/12 4:15	60	Esp	ATM	ATM	No
5	2	3/1/12 9:18	8	Fra	Present	Retail	No
6	1	3/1/12 9:55	1210	Ita	Internet	Airlines	Yes

Data

Derived attributes

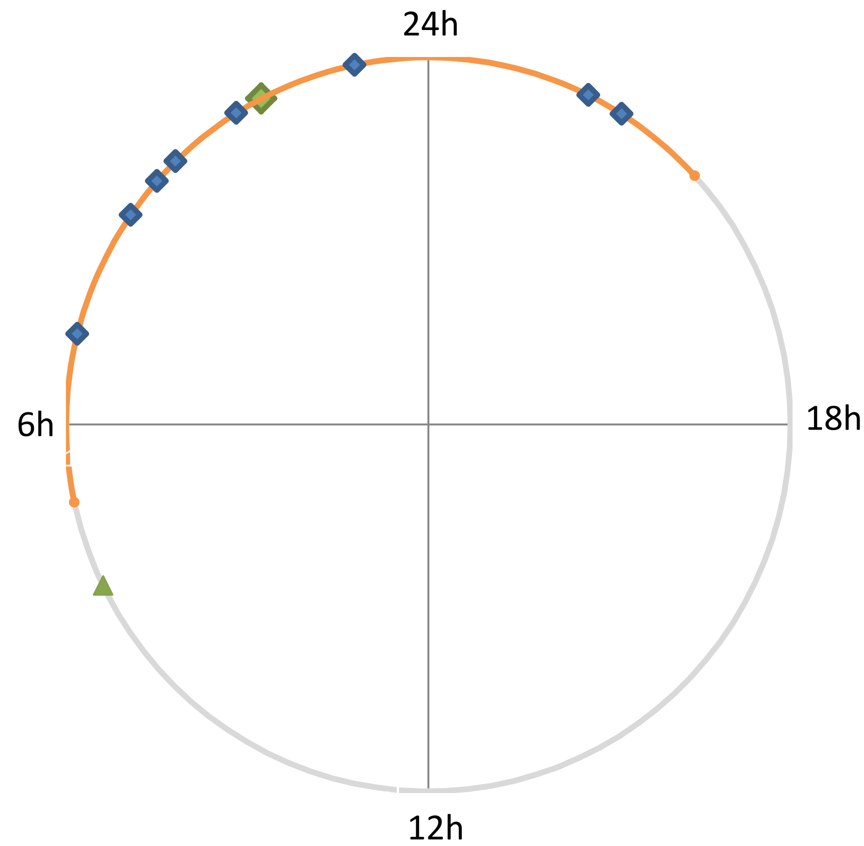
Trx ID	Client ID	Date	Amount	Location	Type	Merchant Group	Fraud	No. of Trx – same client – last 6 hour	Sum – same client – last 7 days
1	1	2/1/12 6:00	580	Ger	Internet	Airlines	No	0	0
2	1	2/1/12 6:15	120	Eng	Present	Car Renting	No	1	580
3	2	2/1/12 8:20	12	Bel	Present	Hotel	Yes	0	0
4	1	3/1/12 4:15	60	Esp	ATM	ATM	No	0	700
5	2	3/1/12 9:18	8	Fra	Present	Retail	No	0	12
6	1	3/1/12 9:55	1210	Ita	Internet	Airlines	Yes	1	760

– Combination of following criteria:

By	Group	Last	Function
Client	None	hour	Count
Credit Card	Transaction Type	day	Sum(Amount)
	Merchant	week	Avg(Amount)
	Merchant Category	month	
	Merchant Country	3 months	

Data

Date of transaction
04/03/2012 - 03:14
07/03/2012 - 00:47
07/03/2012 - 02:57
08/03/2012 - 02:08
14/03/2012 - 22:15
25/03/2012 - 05:03
26/03/2012 - 21:51
28/03/2012 - 03:41



$$Arithmetic\ Mean = \frac{1}{n}\sum t$$

$$Periodic\ Mean = \tan\_2^{-1}\left(\sum \sin(t), \sum \cos(t)\right)$$

$$Periodic\ Std = \sqrt{ln\left(1/\left(\left(\frac{1}{n}\sum \sin(t)\right)^2 + \left(\frac{1}{n}\sum \cos(t)\right)^2\right)\right)}$$

$$t \sim vonmises\left(k \approx {}^{1}/_{std}\right)$$

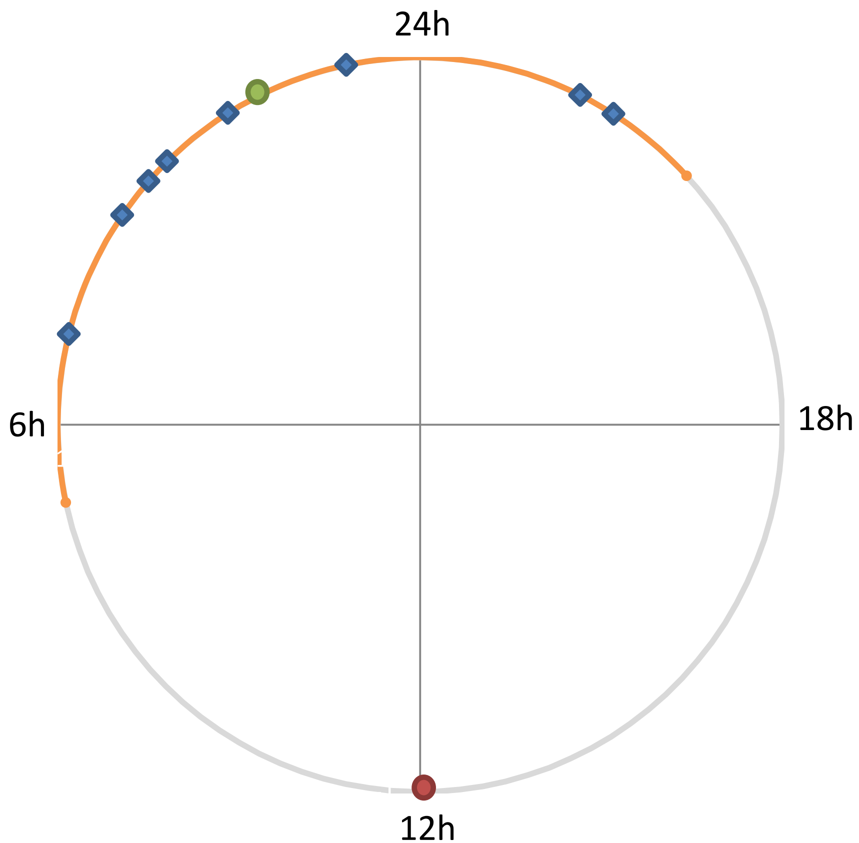
$$P(-zt < t < zt) = 0.95$$

Data

Date of transaction
04/03/2012 - 03:14
07/03/2012 - 00:47
07/03/2012 - 02:57
08/03/2012 - 02:08
14/03/2012 - 22:15
25/03/2012 - 05:03
26/03/2012 - 21:51
28/03/2012 - 03:41

02/04/2012 - 02:02

03/04/2012 - 12:10



new features

Inside CI(0.95) last 30 days
Inside CI(0.95) last 7 days
Inside CI(0.5) last 30 days
Inside CI(0.5) last 7 days

Evaluation

Confusion matrix

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

- Misclassification $= 1 - \frac{TP+TN}{TP+TN+FP+FN}$
- Recall $= \frac{TP}{TP+FN}$
- Precision $= \frac{TP}{TP+FP}$
- F-Score $= 2\frac{Precision * Recall}{Precision+Recall}$

Evaluation - Financial measure

Motivation:

TRX ID	Amount	Fraud	Algorithm 1 Prediction (Fraud?)	Algorithm 2 Prediction (Fraud?)	Algorithm 3 Prediction (Fraud?)
1	580	No	No	No	No
2	120	No	No	No	No
3	12	Yes	No	Yes	No
4	60	No	No	No	No
5	8	No	No	Yes	Yes
6	1210	Yes	No	No	Yes

	Algorithm 1	Algorithm 2	Algorithm 3
Miss-Class	2 / 6	2 / 6	2 / 6
Cost	1222	1212	14

- Equal misclassification results
- Frauds carry different cost

Evaluation

SNT
securityandtrust.lu

Cost matrix

	Actual Positive $y_i = 1$	Actual Negative $y_i = 0$
Predicted Positive $c_i = 1$	C_{TP_i}	C_{FP_i}
Predicted Negative $c_i = 0$	C_{FN_i}	C_{TN_i}

where the cost associated with two types of correct classification, true positives and true negatives, and the two types of misclassification errors, false positives and false negatives, are presented.

uni.lu UNIVERSITÉ DU LUXEMBOURG

Evaluation

- As discussed in [Elkan 2001], the cost of correct classification should always be lower than the one of misclassification. These are referred to as "reasonableness" conditions.

$$C_{FP_i} > C_{TN_i} \text{ and } C_{FN_i} > C_{TP_i}$$

- Using the "reasonableness" conditions, the cost matrix can be scaled and shifted to a simpler one with only one degree of freedom

Negative	$C^*_{FN_i} = \frac{(C_{FN_i} - C_{TN_i})}{(C_{FP_i} - C_{TN_i})}$
Positive	$C^*_{TP_i} = \frac{(C_{TP_i} - C_{TN_i})}{(C_{FP_i} - C_{TN_i})}$

Evaluation

Cost-sensitive problem definition

- Classification problem cost characteristic:

$$b_i = C^*_{FN_i} - C^*_{TP_i} - 1$$

with mean μ_b and std σ_b

- A classification problem is defined as:

cost-insensitive	$\mu_b = 0$ and $\sigma_b = 0$
class-dependent cost-sensitive	$\mu_b \neq 0$ and $\sigma_b = 0$
example-dependent cost-sensitive	$\sigma_b > 0$

Evaluation

Cost matrix: Fraud detection

	Actual Positive $y_i = 1$	Actual Negative $y_i = 0$
Predicted Positive $c_i = 1$	C_a	C_a
Predicted Negative $c_i = 0$	Amt_i	0

C_a refers to the administrative cost and Amt_i to the amount of transaction i

Evaluation

Cost-sensitive problem evaluation

- Cost of applying a classifier to a given set

$$C(S) = \sum_{i=1}^{N} \Big(y_i (c_i C_{TP_i} + (1 - c_i) C_{FN_i}) + (1 - y_i)(c_i C_{FP_i} + (1 - c_i) C_{TN_i}) \Big)$$

- Savings are:

$$C^*(S) = \frac{C_s(S) - C(S)}{C_s(S)}$$

where

$$C_s(S) = \min \Big\{ C_0(S), C_1(S) \Big\}$$

and C_0 , C_1 refers to special cases where for all the examples, c_i equals to 0 and 1 respectively.

Agenda

SnT
securityandtrust.lu

- Introduction
- Database
- Evaluation
- Algorithms
  - Cost-sensitive logistic regression
  - Bayes Minimum Risk
  - Example-dependent cost-sensitive decision tree
- Conclusions & Future Work

uni.lu UNIVERSITÉ DU LUXEMBOURG

Logistic Regression

securityandtrust.lu

- Model

$$log\left(\frac{p}{1-p}\right) = \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \ldots + \theta_n x_n$$

- Cost Function

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left[-y_i log(p_\theta(x_i)) - (1-y_i)log(1-p_\theta(x_i))\right]$$

Cost Sensitive Logistic Regression

- Cost Matrix

	Actual Positive $y_i = 1$	Actual Negative $y_i = 0$
Predicted Positive $c_i = 1$	$\boxed{C_a}$	$\boxed{C_a}$
Predicted Negative $c_i = 0$	$\boxed{Amt_i}$	0

- Cost Function

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \left[y_i \left(p_\theta^*(x_i) \boxed{Ca} + (1 - p_\theta^*(x_i)) \boxed{Amt_i} \right) + (1 - y_i) p_\theta^*(x_i) \boxed{Ca} \right]$$

- Objective

  Find θ that minimized the cost function

Cost Sensitive Logistic Regression

SNT
securityandtrust.lu

- Cost Function

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left[y_i\left(p_\theta^*(x_i)Ca + (1 - p_\theta^*(x_i))Amt_i\right) + (1 - y_i)p_\theta^*(x_i)Ca\right]$$

- Gradient

$$\frac{\partial J(\theta)}{\partial \theta_{(j)}} = \frac{1}{m}\sum_{i=1}^{m}\left[\left[-y_iAmt_i + Ca - y_iCa - y_i\right]\left(\frac{\left(-e^{-\sum_{j=1}^{n}\theta_{(j)}x_{i(j)}}\right)(-x_{i(j)})}{\left(1+e^{-\sum_{j=1}^{n}\theta_{(j)}x_{i(j)}}\right)^2}\right)\right]$$

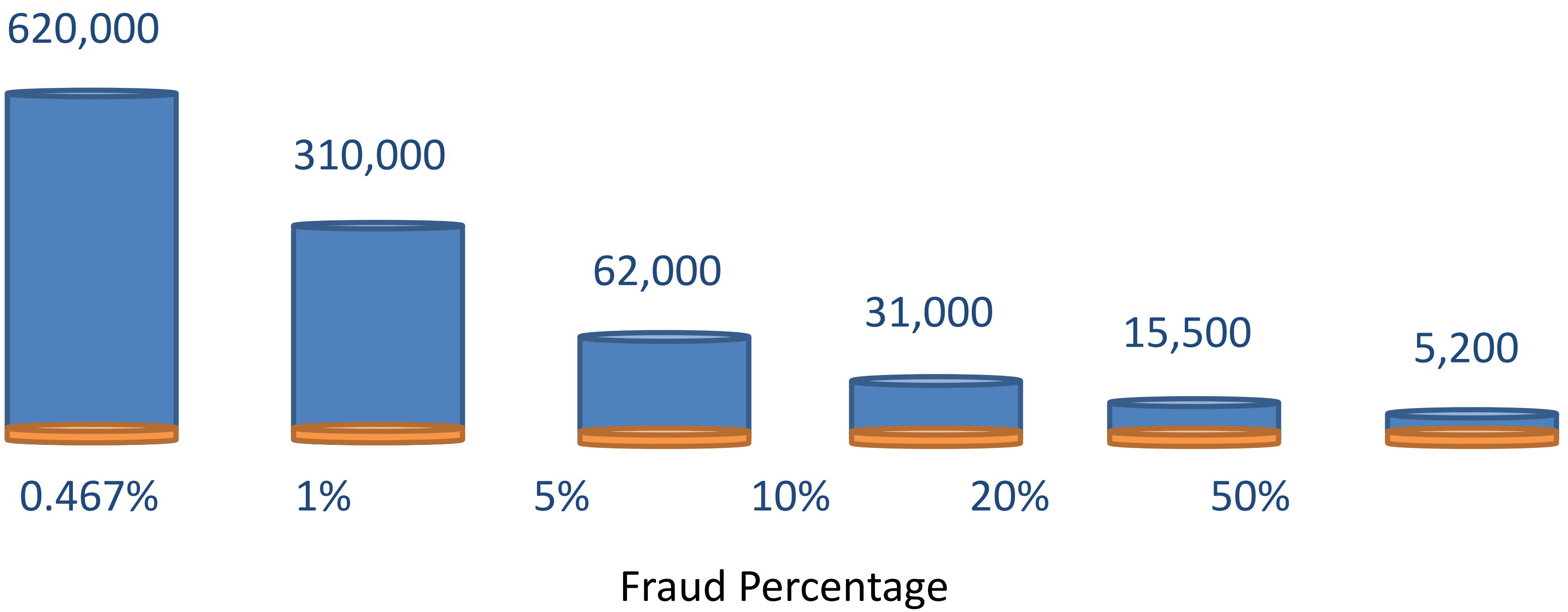
- Hessian

$$\frac{\partial^2 J(\theta)}{\partial \theta_{(j1)}\partial \theta_{(j2)}} = \frac{1}{m}\sum_{i=1}^{m}\left[\left[-y_iAmt_i + (1 - y_i)Ca\right]\left((-x_{i(j1)})(x_{i(j2)})^2(1 - p_\theta^*(x_i))^3(p_\theta^*(x_i))^3\right)\right]$$

uni.lu
UNIVERSITÉ DU LUXEMBOURG

Experiments – Logistic Regression

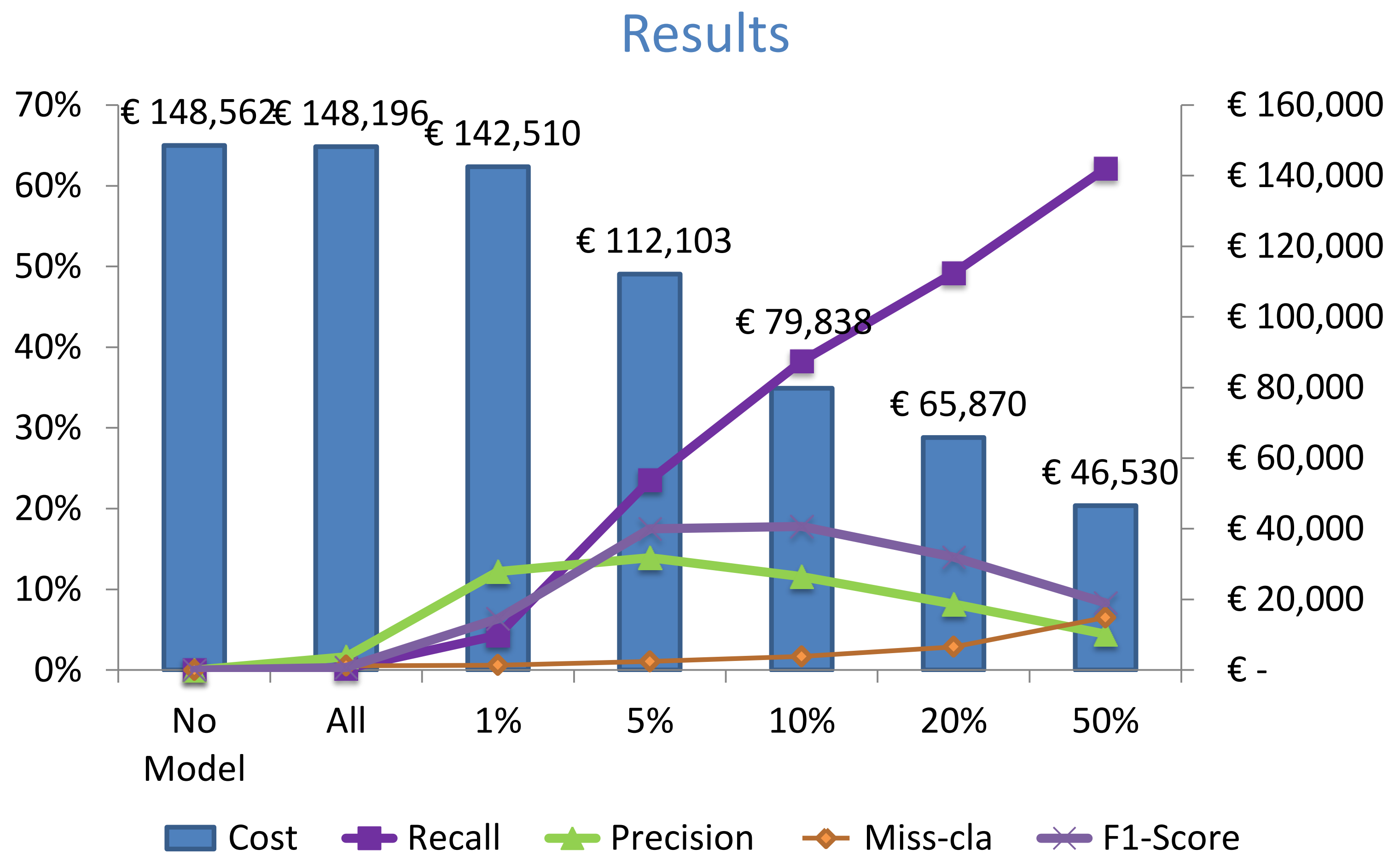
Sub-sampling procedure:



Select all the frauds and a random sample of the legitimate transactions.

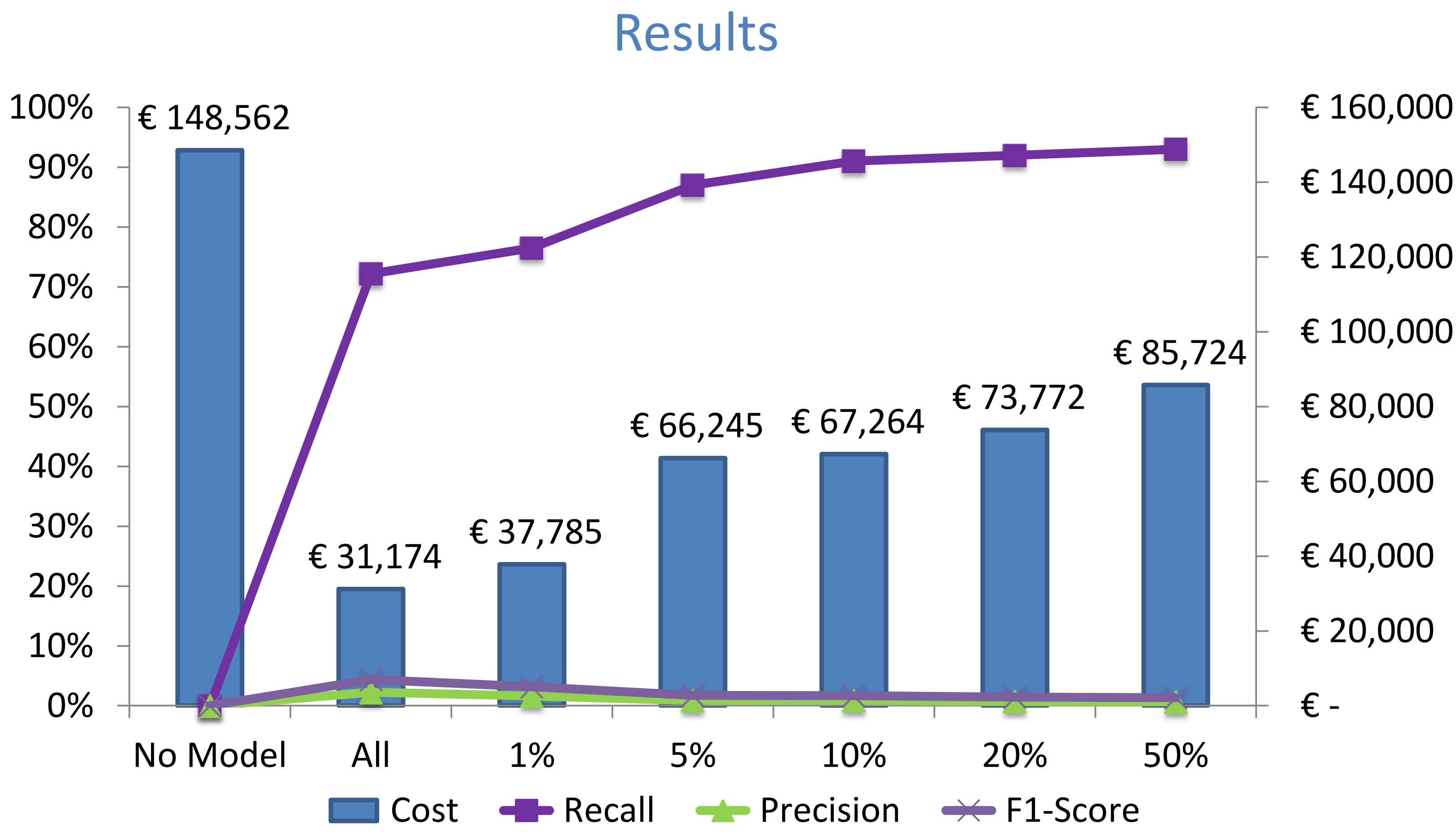
* OLD Dataset

Experiments – Logistic Regression

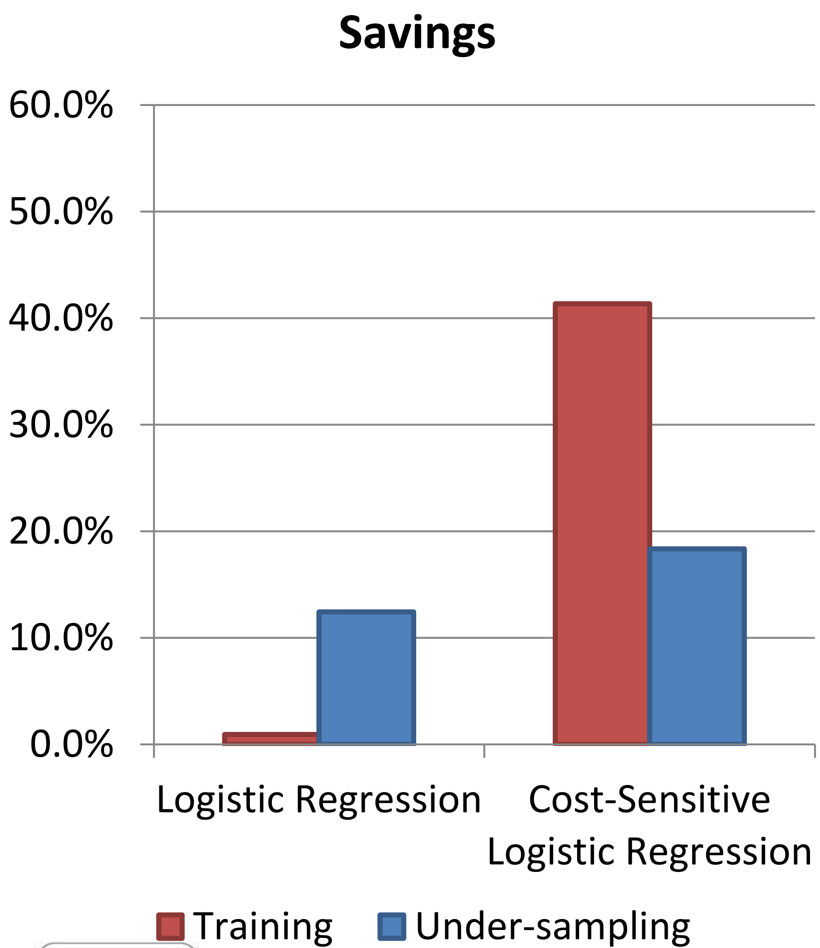


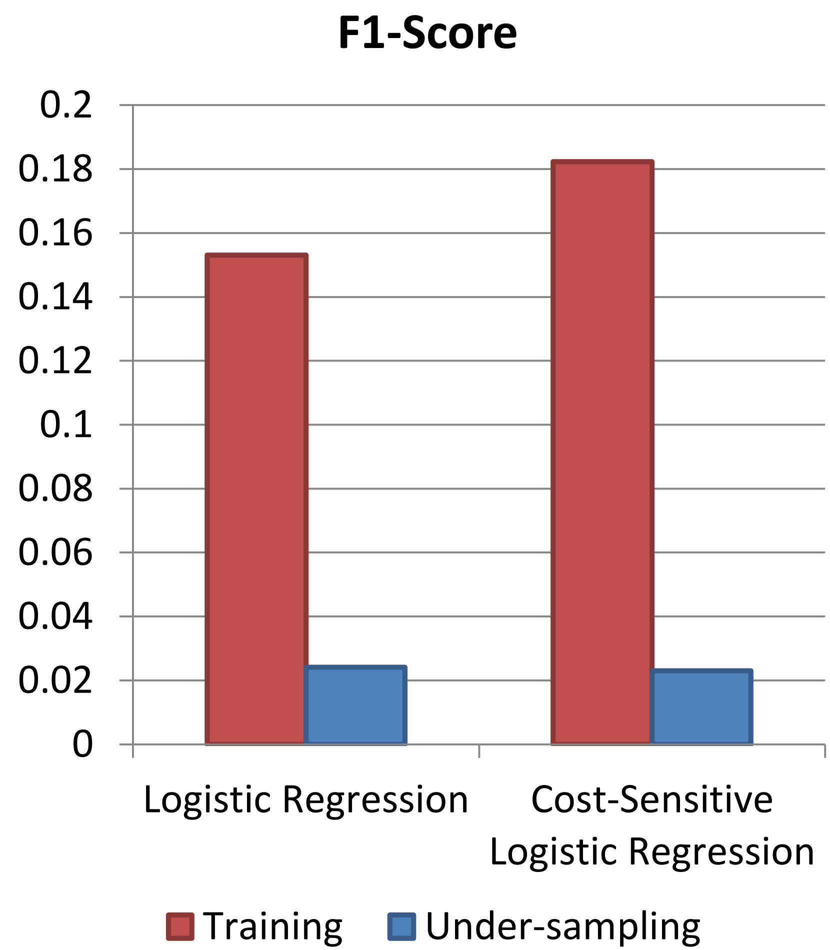
* OLD Dataset

Experiments – CS Logistic Regression



Experiments – CS Logistic Regression





Agenda

- Introduction
- Database
- Evaluation
- Algorithms
  - Cost-sensitive logistic regression
  - Bayes Minimum Risk
  - Example-dependent cost-sensitive decision tree
- Conclusions & Future Work

Bayes Minimum Risk

- Decision model based on quantifying tradeoffs between various decisions using probabilities and the costs that accompany such decisions

- Risk of classification

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

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

Bayes Minimum Risk

- Using the different risks the prediction is made based on the following condition:

$$c_i = \begin{cases} 0 & R(c_i = 0|X_i) \leq R(c_i = 1|X_i) \\ 1 & \text{otherwise} \end{cases}$$

- Example-dependent threshold

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

Is always defined taking into account the “reasonableness” conditions

Probability Calibration

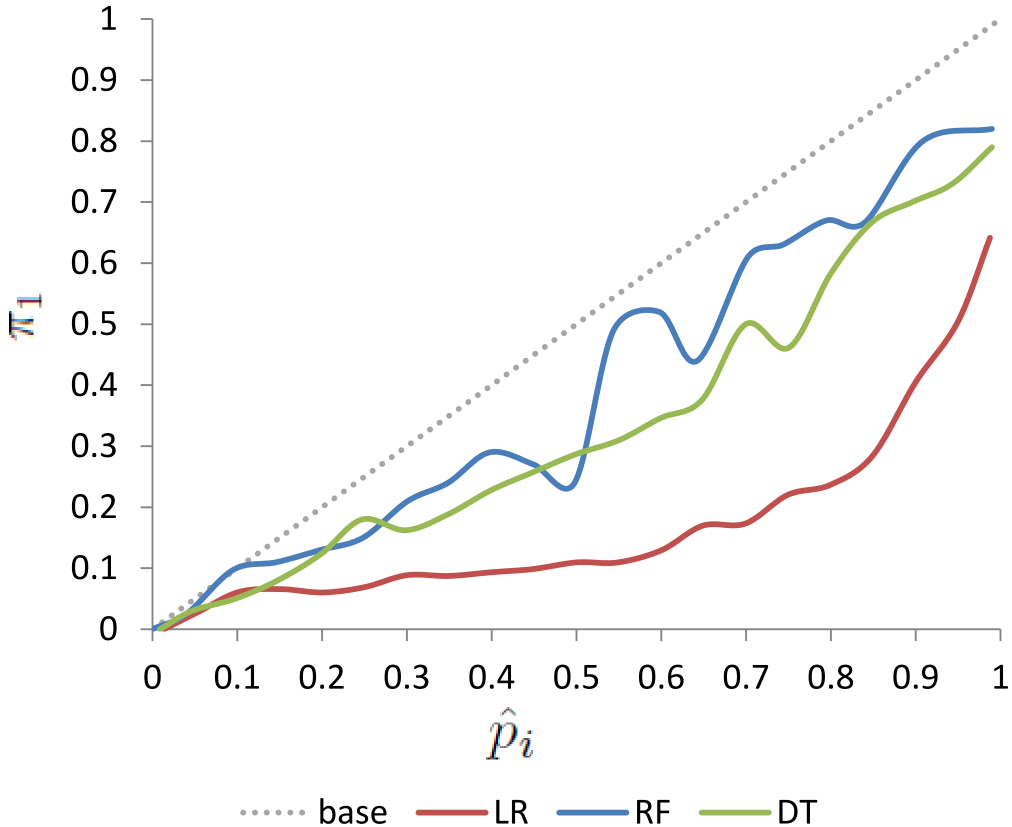
SnT
securityandtrust.lu

- When using the output of a binary classifier as a basis for decision making, there is a need for a probability that not only separates well between positive and negative examples, but that also assesses the real probability of the event [Cohen and Goldszmidt 2004]

UNIVERSITÉ DU LUXEMBOURG

Probability Calibration

- Reliability Diagram

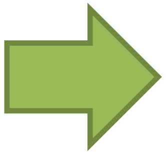


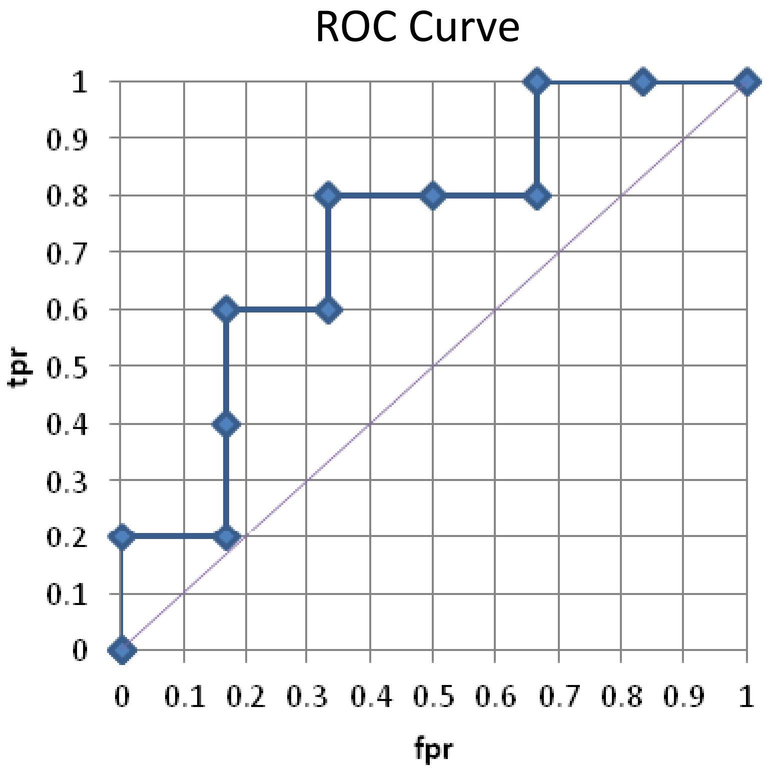
π_1 is the positive rate and $\hat{p}_i$ is the predicted probability

Probability Calibration

- ROC Convex Hull calibration [Hernandez-Orallo et al. 2012]

Class (y)	Prob (p)
0	0.0
1	0.1
0	0.2
0	0.3
1	0.4
0	0.5
1	0.6
1	0.7
0	0.8
1	0.9
1	1.0

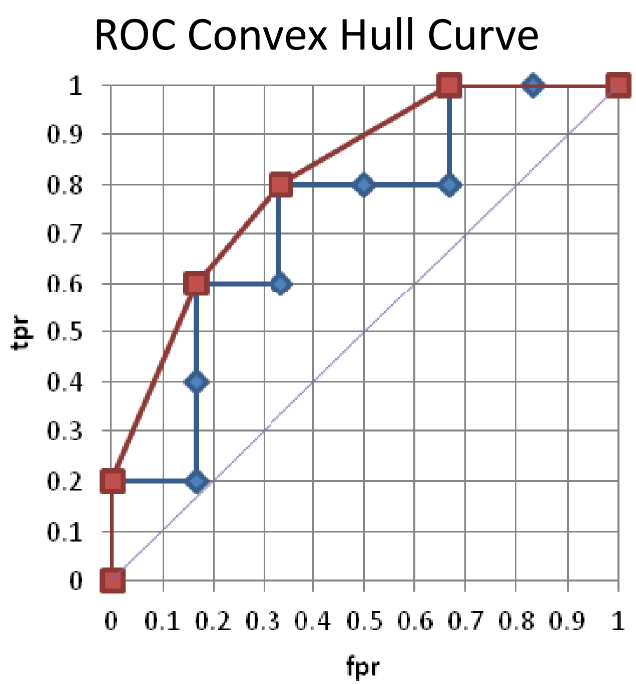




Probability Calibration

securityandtrust.lu

- ROC Convex Hull calibration

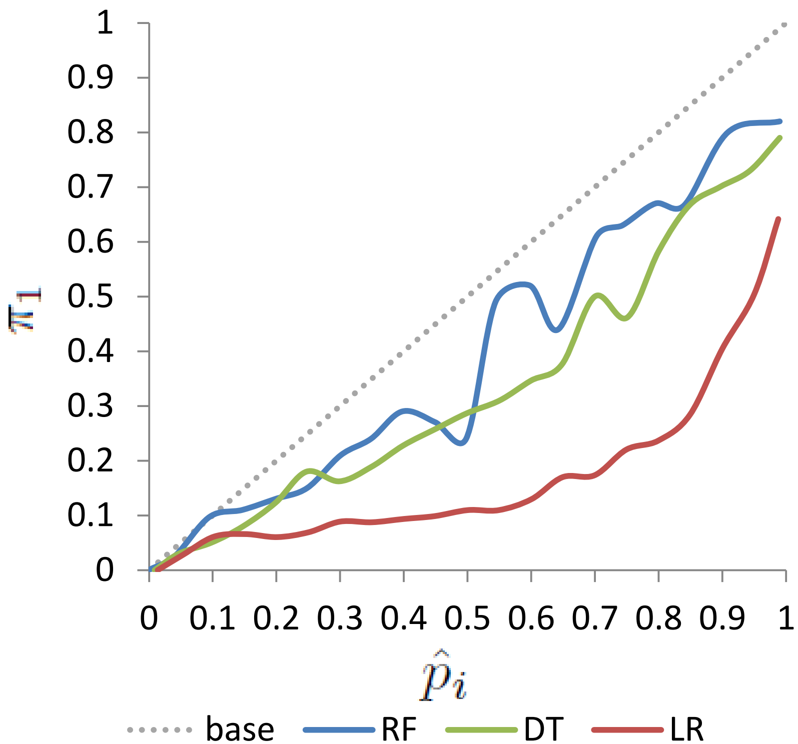


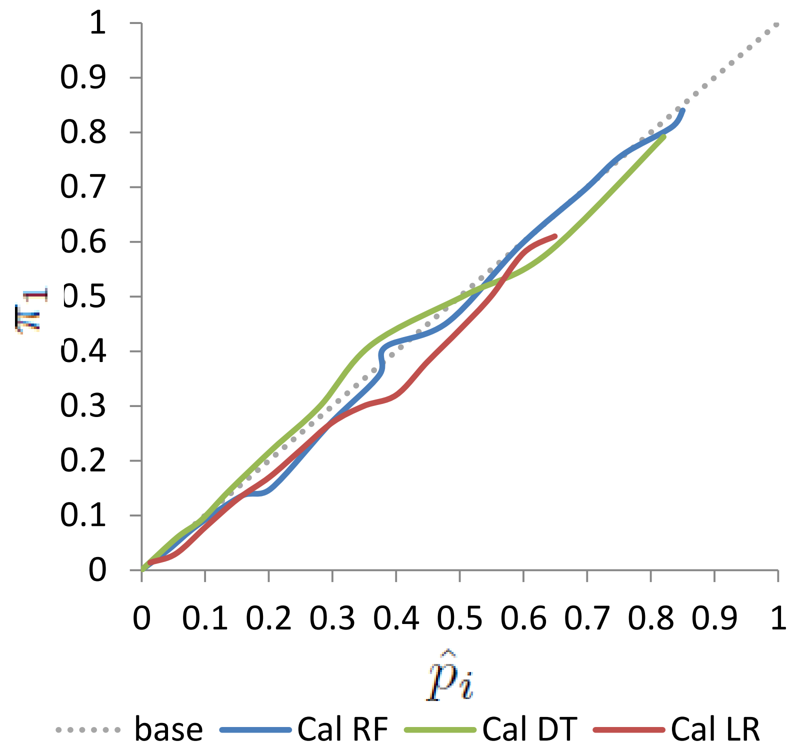
Class (y)	Prob (p)	Cal Prob
0.0	0	0
0.1	1	0.333
0.2	0	0.333
0.3	0	0.333
0.4	1	0.5
0.5	0	0.5
0.6	1	0.666
0.7	1	0.666
0.8	0	0.666
0.9	1	1
1.0	1	1

the calibrated probabilities are extracted by first grouping the probabilities according to the points in the ROCCH curve, and then the calibrated probabilities are equal to the slope for each group.

Probability Calibration

- Reliability Diagram





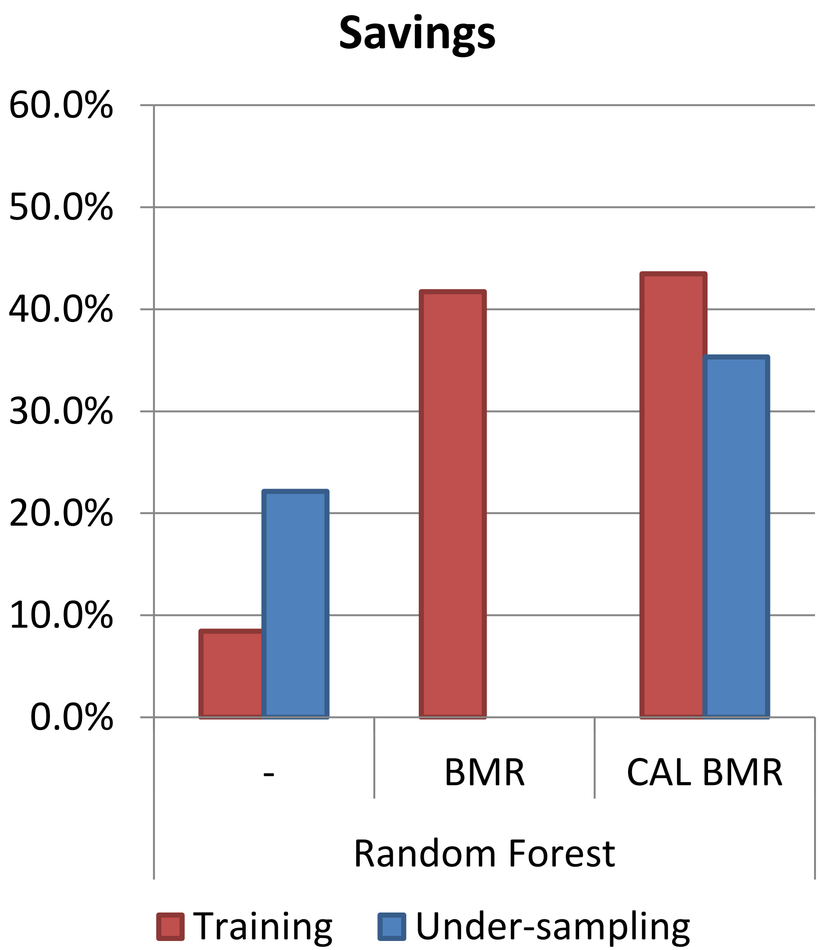
Experiments – Bayes Minimum Risk

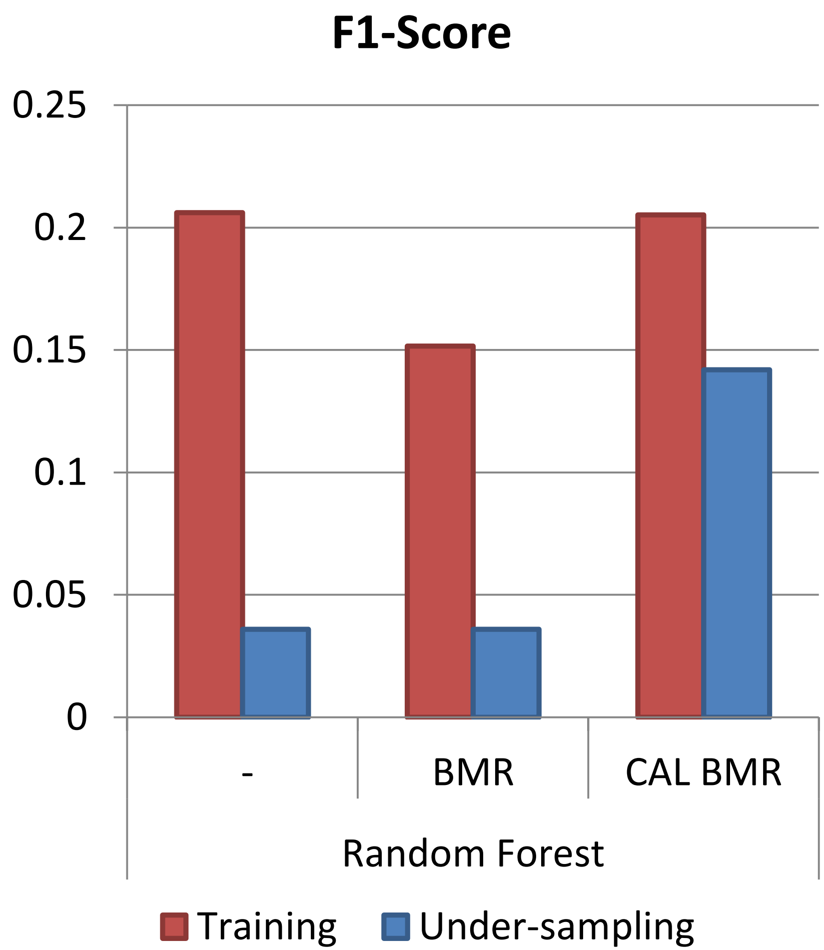
- Estimation of the fraud probabilities using one of the following algorithms:
  1. Random Forest
  2. Decision Trees
  3. Logistic Regression

- For each algorithm comparison of
  - Raw prediction
  - Bayes Minimum Risk
  - Probability Calibration and Bayes Minimum Risk

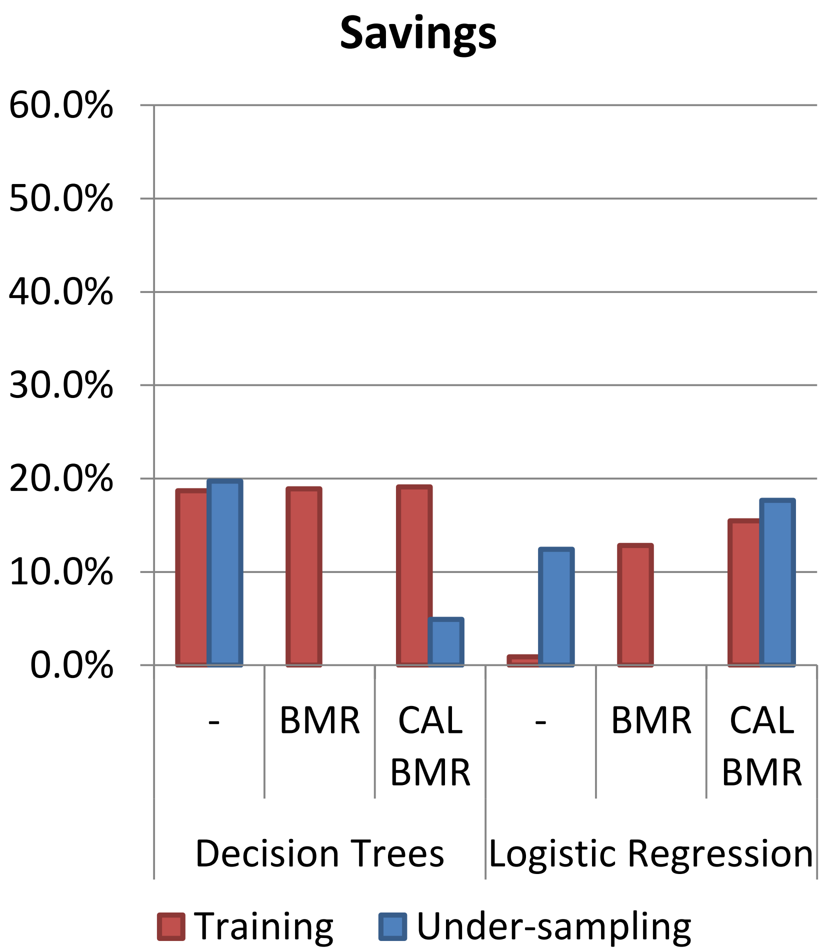
- Trained using the different sets
  - Training
  - Under-sampling

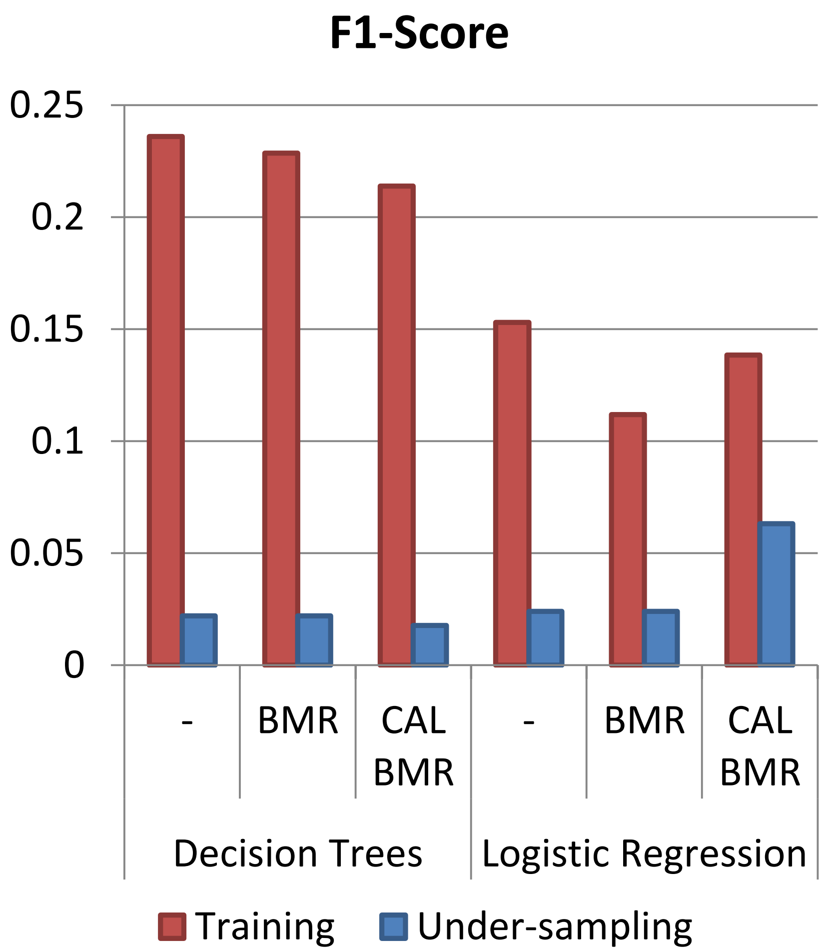
Experiments – Bayes Minimum Risk





Experiments – Bayes Minimum Risk





Agenda

- Introduction
- Database
- Evaluation
- Algorithms
  - Cost-sensitive logistic regression
  - Bayes Minimum Risk
  - Example-dependent cost-sensitive decision tree
- Conclusions & Future Work

EDCS – Decision trees

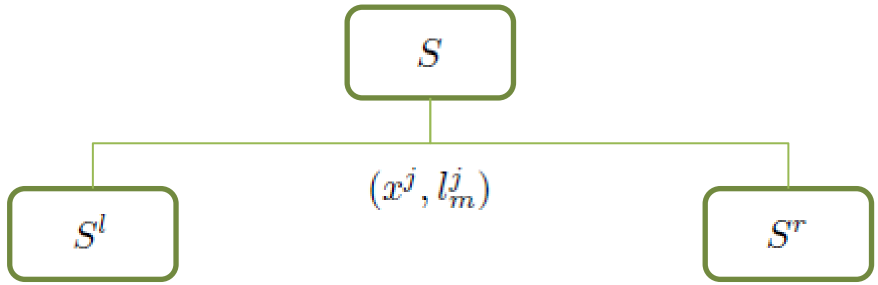
Decision trees

Classification model that iteratively creates binary decision rules ((x^j, l^j_m)) that maximize certain criteria

Where ((x^j, l^j_m)) refers to making a rule using feature j on value m

EDCS – Decision trees

Decision trees - Construction



$$\pi_1 = \frac{|S_1|}{|S|}$$
$$\pi_1^l = \frac{|S_1^l|}{|S^l|}$$
$$\pi_1^r = \frac{|S_1^r|}{|S^r|}$$

$$S^l = \{X | X_i \in S \wedge x_i^j \leq l^j_m\} \qquad S^r = \{X | X_i \in S \wedge x_i^j > l^j_m\}$$

- Then the impurity of each leaf is calculated using:

$$\begin{array}{lcl} \text{Misclassification} & : & I_m(\pi_1) = 1 - \max\{\pi_1, (1-\pi_1)\} \\ \text{Entropy} & : & I_e(\pi_1) = -\pi_1 \log \pi_1 - (1-\pi_1)\log(1-\pi_1) \\ \text{Gini} & : & I_g(\pi_1) = 2\pi_1(1-\pi_1) \end{array}$$

- Afterwards the gain of applying a given rule to the set S is:

$$Gain((x^j, l^j_m)) = I(\pi_1) - \frac{|S^l|}{|S|} I(\pi_1^l) - \frac{|S^r|}{|S|} I(\pi_1^r)$$

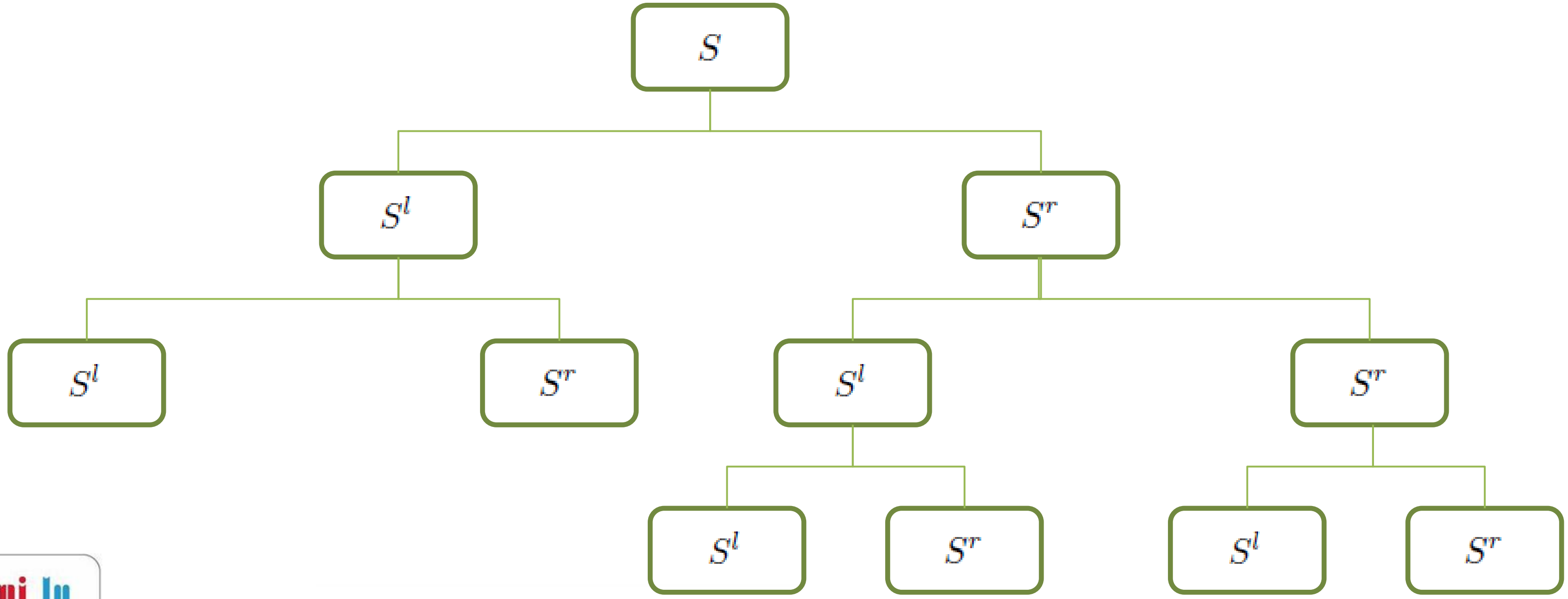
EDCS – Decision trees

Decision trees - Construction

- The rule that maximizes the gain is selected

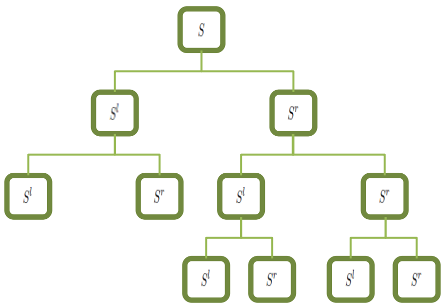
$$(best_x, best_l) = \text{argmax}_{(j,m)}\, Gain((x^j, l^j_m))$$

- The process is repeated until a stopping criteria is met:

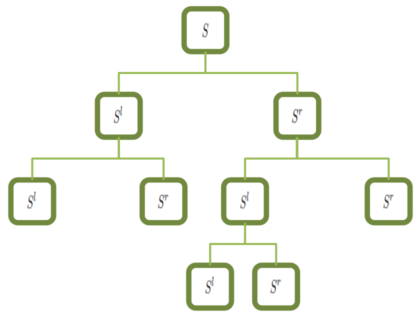


Decision trees - Pruning

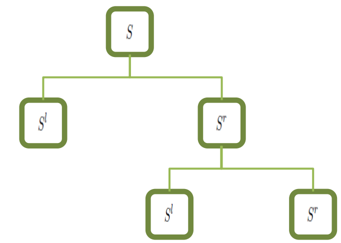
- Calculation of the Tree error and pruned Tree error



$$\epsilon(Tree, S)$$



$$\frac{\epsilon(EB(Tree, branch), S) - \epsilon(Tree, S)}{|Tree| - |EB(Tree, branch)|}$$



$$\frac{\epsilon(EB(Tree, branch), S) - \epsilon(Tree, S)}{|Tree| - |EB(Tree, branch)|}$$

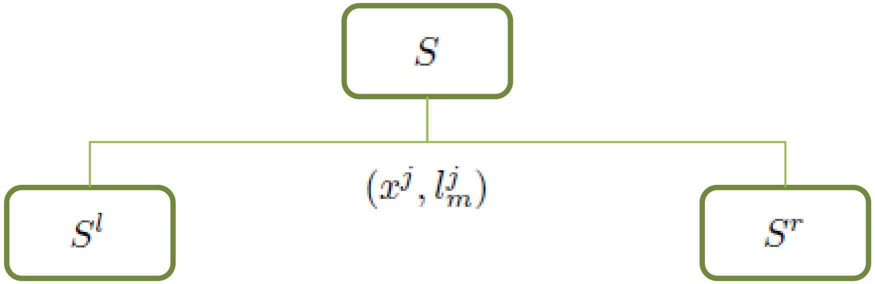
- After calculating the pruning criteria for all possible trees. The maximum improvement is selected and the Tree is pruned.
- Later the process is repeated until there is no further improvement.

EDCS – Decision trees

- Maximize the accuracy is different than maximizing the cost.
- To solve this, some studies had been proposed method that aim to introduce the cost-sensitivity into the algorithms [Lomax and Vadera 2013].
- However, research have been focused on class-dependent methods [Draper et al. 1994; Ting 2002; Ling et al. 2004; Li et al. 2005; Kretowski and Grzes 2006; Vadera 2010]
- We propose:
  - Example-dependent cost based impurity measure
  - Example-dependent cost based pruning criteria

EDCS – Decision trees

Cost based impurity measure



$S^l = \{X | X_i \in S \wedge x^j_i \leq l^j_m\}$ $\qquad$ $S^r = \{X | X_i \in S \wedge x^j_i > l^j_m\}$

- The impurity of each leaf is calculated using:

$$I_c(S) = C_s(S) = \min \left\{ C_0(S), C_1(S) \right\}$$

$$f(S) = \begin{cases} 0 & \text{if } C_0(S) \leq C_1(S) \\ 1 & \text{otherwise} \end{cases}$$

- Afterwards the gain of applying a given rule to the set S is:

$$Gain_c((x^j, l^j_m)) = I_c(S) - (I_c(S^l) + I_c(S^r))$$

EDCS – Decision trees

Weighted vs. not weighted gain

$$Gain((x^j, l_m^j)) = I(\pi_1) - \frac{|S^l|}{|S|} I(\pi_1^l) - \frac{|S^r|}{|S|} I(\pi_1^r)$$

$$Gain_c((x^j, l_m^j), S) = I_c(S) - (I_c(S^l) + I_c(S^r))$$

- Using the not weighted gain, when booths left and right leafs have the same prediction, the gain is equal 0

if

$$f(S^l) = f(S^r)$$

then

$$I_c(S) = (I_c(S^l) + I_c(S^r))$$

EDCS – Decision trees

Cost sensitive pruning

$$PC_c = \frac{C(S,f(S,Tree)) - C(S,f(S,EB(Tree,branch)))}{|Tree| - |EB(Tree,branch)|}$$

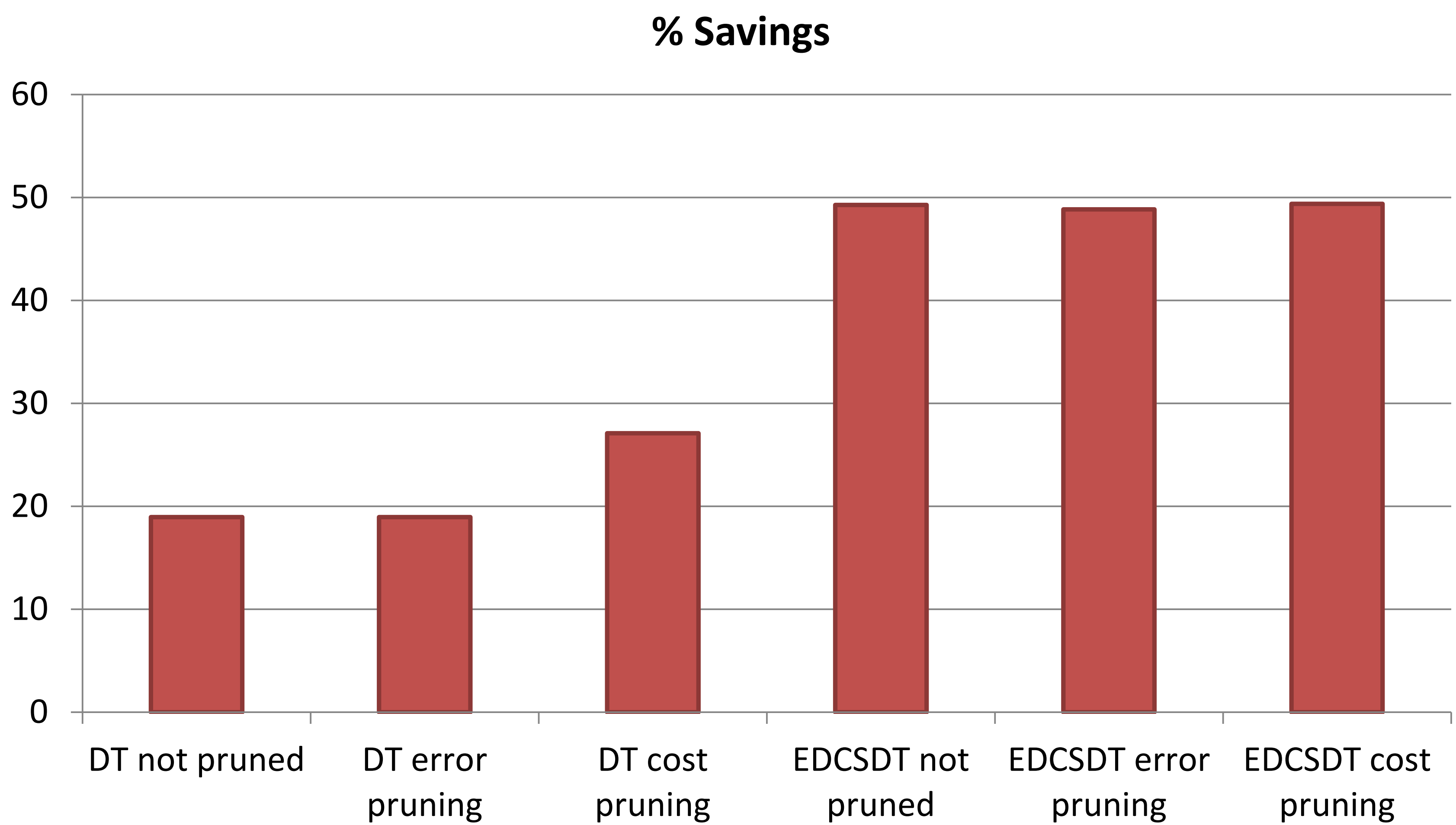
- New pruning criteria that evaluates the improvement in cost of eliminating a particular branch

Experiments - EDCS – Decision trees

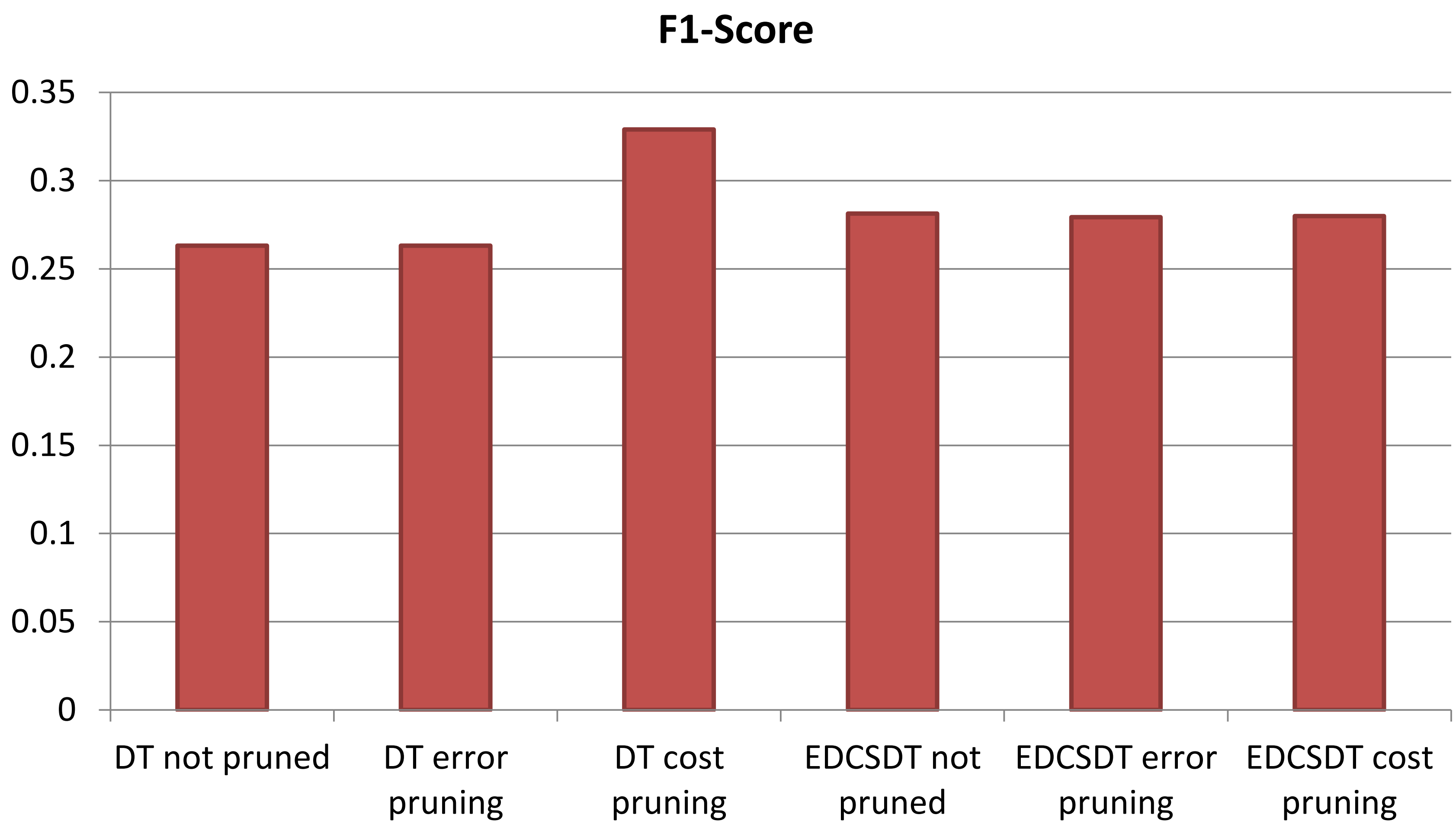
- Comparison of the following algorithms:
  - Decision Tree – not pruned
  - Decision Tree – error based pruning
  - Decision Tree – cost based pruning
  - EDCS-Decision Tree – not pruned
  - EDCS-Decision Tree – error based pruning
  - EDCS-Decision Tree – cost based pruning

- Trained using the different sets:
  - Training
  - Under-sampling
  - Cost-proportionate Rejecting-sampling
  - Cost-proportionate Over-sampling

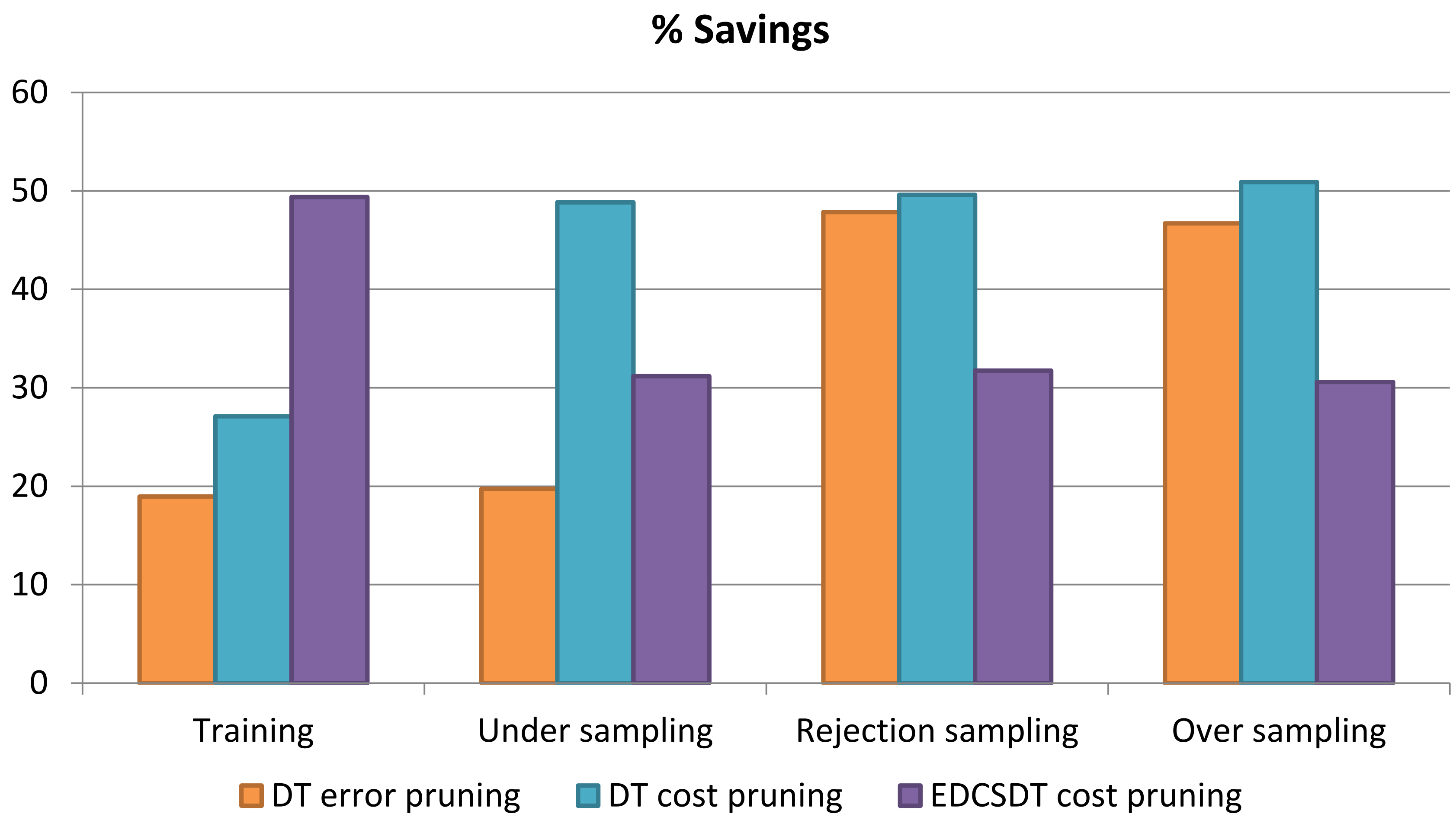
Experiments - EDCS – Decision trees



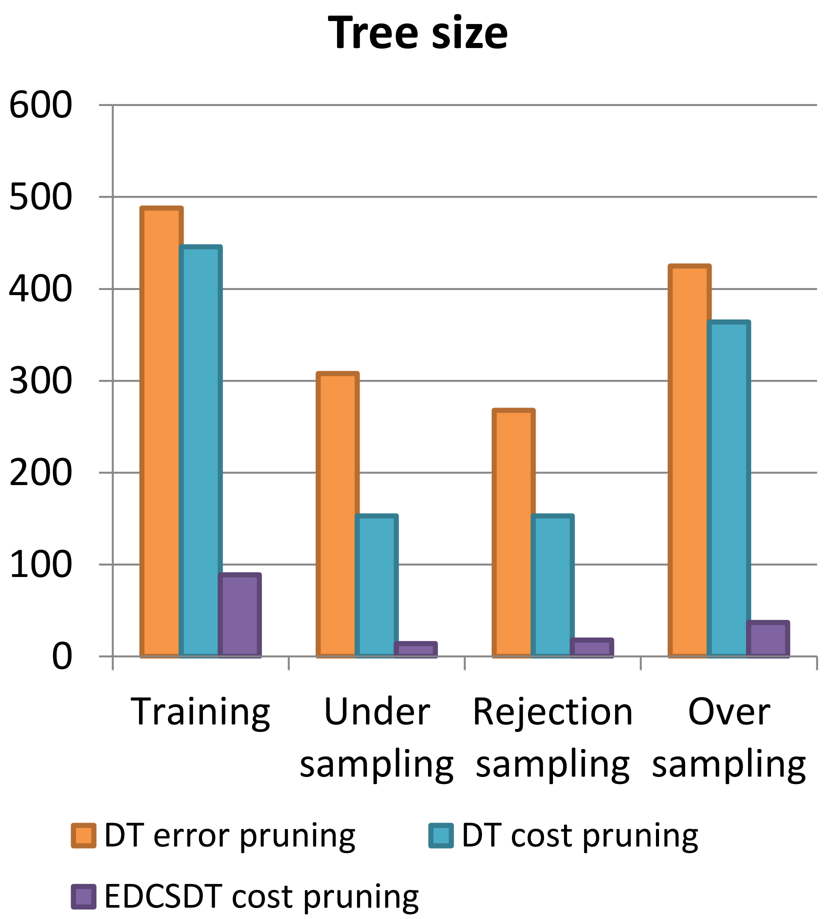
Experiments - EDCS – Decision trees



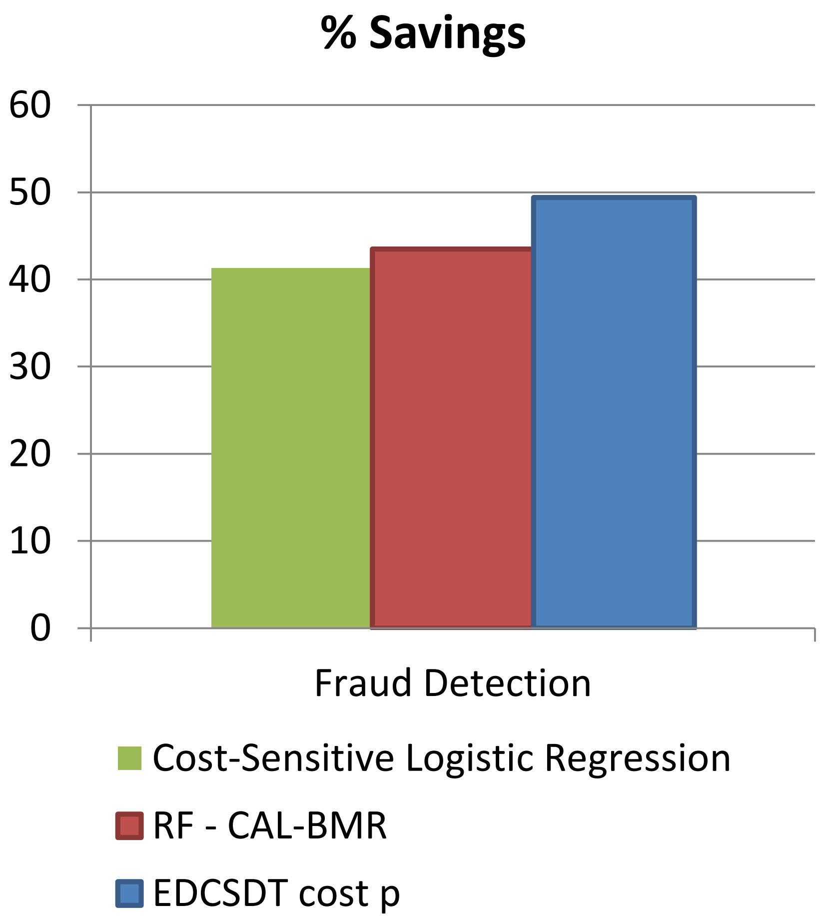
Experiments - EDCS – Decision trees



Experiments - EDCS – Decision trees



Experiments – Comparison



Conclusions

- New framework for defining cost-sensitive problems
- Including the cost into Logistic Regression increases the savings
- Bayes minimum risk model arise to better results measure by savings and results are independent of the base algorithm used
- Calibration of probabilities help to achieve further savings
- Example-dependent cost-sensitive decision trees improves the savings and have a much lower training time than traditional decision trees

Future work

- Boosted Example Dependent Cost Sensitive Decision Trees

- Example-Dependent Cost-Sensitive Calibration Method

- Reinforced Learning (Asynchronous feedback)

Contact information

Alejandro Correa Bahnsen

University of Luxembourg

Luxembourg

al.bahnsen@gmail.com

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

http://www.slideshare.net/albahnsen

References

- Correa Bahnsen, A., Stojanovic, A., Aouada, D., & Ottersten, B. (2013). Cost Sensitive Credit Card Fraud Detection using Bayes Minimum Risk. In International Conference on Machine Learning and Applications. Miami, USA: IEEE.

- Correa Bahnsen, A., Stojanovic, A., Aouada, D., & Ottersten, B. (2014). Improving Credit Card Fraud Detection with Calibrated Probabilities. In SIAM International Conference on Data Mining. Philadelphia, USA: SIAM.

- Correa Bahnsen, A., Aouada, D., & Ottersten, B. (2014). Example-Dependent Cost-Sensitive Credit Scoring using Bayes Minimum Risk. Submitted to ECAI 2014.

- Correa Bahnsen, A., Aouada, D., & Ottersten, B. (2014). Example-Dependent Cost-Sensitive Decision Tress. Submitted to ACM TIST 2014.

References

securityandtrust.lu

- Charles Elkan. 2001. The Foundations of Cost-Sensitive Learning. In Seventeenth International Joint Conference on Artificial Intelligence. 973–978.
- Bianca Zadrozny, John Langford, and Naoki Abe. 2003. Cost-sensitive learning by cost-proportionate example weighting. In Third IEEE International Conference on Data Mining. IEEE Comput. Soc, 435–442.
- Mac Aodha, O., & Brostow, G. J. (2013). Revisiting Example Dependent Cost-Sensitive Learning with Decision Trees. In The IEEE International Conference on Computer Vision (ICCV).
- Cohen, I., & Goldszmidt, M. (2004). Properties and Benefits of Calibrated Classifiers. In Knowledge Discovery in Databases: PKDD 2004 (Vol. 3202, pp. 125–136). Springer Berlin Heidelberg.
- Hernandez-Orallo, J., Flach, P., & Ferri, C. (2012). A Unified View of Performance Metrics : Translating Threshold Choice into Expected Classification Loss. Journal of Machine Learning Research, 13, 2813–2869.
- Susan Lomax and Sunil Vadera. 2013. A survey of cost-sensitive decision tree induction algorithms. Comput. Surveys 45, 2 (Feb. 2013), 1–35.
- BA Draper, CE Brodley, and PE Utgoff. 1994. Goal-directed classification using linear machine decision trees. IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (1994), 888–893.
- KM Ting. 2002. An instance-weighting method to induce cost-sensitive trees. IEEE Transactions on Knowledge and Data Engineering 14, 3 (2002), 659–665.
- J Li, Xiaoli Li, and Xin Yao. 2005. Cost-Sensitive Classification with Genetic Programming. In 2005 IEEE Congress on Evolutionary Computation, Vol. 3. IEEE, 2114–2121.
- Charles X. Ling, Qiang Yang, Jianning Wang, and Shichao Zhang. 2004. Decision trees with minimal costs. In Twenty-first international conference on Machine learning - ICML '04.
- M Kretowski and M Grzes. 2006. Evolutionary induction of cost-sensitive decision trees. In Foundations of Intelligent Systems. Springer Berlin Heidelberg, 121–126.