Instance-dependent cost-sensitive learning: do we really need it?

Toon Vanderschueren
KU Leuven
University of Antwerp
toon.vanderschueren@kuleuven.be

Tim Verdonck
KU Leuven
University of Antwerp
tim.verdonck@uantwerpen.be

Bart Baesens
KU Leuven
University of Southampton
bart.baesens@kuleuven.be

Wouter Verbeke
KU Leuven
wouter.verbeke@kuleuven.be

Abstract

Traditionally, classification algorithms aim to minimize the number of errors. However, this approach can lead to sub-optimal results for the common case where the actual goal is to minimize the total cost of errors and not their number. To address this issue, a variety of cost-sensitive machine learning techniques has been suggested. Methods have been developed for dealing with both class- and instance-dependent costs. In this article, we ask whether we really need instance-dependent rather than class-dependent cost-sensitive learning? To this end, we compare the effects of training cost-sensitive classifiers with instance- and class-dependent costs in an extensive empirical evaluation using real-world data from a range of application areas. We find that using instance-dependent costs instead of class-dependent costs leads to improved performance for cost-sensitive performance measures, but worse performance for cost-insensitive metrics. These results confirm that instance-dependent methods are useful for many applications where the goal is to minimize costs.

1. Introduction

The goal of a classification model is to assign the correct class label to instances based on their characteristics by learning from a set of training examples. Typically, the aim is to develop a model that minimizes the number of incorrect decisions or errors. However, such an approach implicitly assumes that all errors are equally costly - an assumption that is not realistic for many real-world applications [1]. In disease diagnosis, for example, there is a larger risk involved in wrongly predicting a sick patient to be healthy than predicting a healthy one to be sick. In this setting, costs are class-dependent. Other settings are even more complex as the costs not only depend on the predicted and actual class, but also on some properties of the instance itself (e.g. the transaction amount in credit card fraud detection). In the literature, this has also been referred to as example-dependent or observation-dependent costs.

Cost-sensitive learning is a subfield of machine learning aimed at more effectively dealing with these cases by including costs in the learning algorithms and decision-making stage. This way, the goal of the classification model becomes more aligned to the true objectives. Even though most cost-sensitive approaches aim to more effectively deal with class-dependent cost, recently techniques have been proposed for incorporating instance-dependent costs specifically. Instance-dependent costs are a critical aspect of many applications. However, taking instance-dependent costs into account also brings additional complexity to the learning problem as both the class and cost distributions need to be considered simultaneously (see Figure 1).

Despite the conceptual differences, the benefits and drawbacks of using instance- rather than class-dependent costs on the performance of learning algorithms has not yet been examined empirically. Therefore, similar in spirit to the work of [2] on class-dependent cost-sensitive boosting, we ask: Do we really need *instance-dependent* cost-sensitive learning? To this end, we present an extensive empirical evaluation comparing models trained with class-dependent and instance-dependent costs for different cost-sensitive objective functions and types of classifiers.

2. Related work

In classification, different types of costs can be formalized with the concept of a cost matrix [3]. Similar to how a confusion matrix differentiates between outcomes depending on the actual and predicted class (see Table 1a), a cost matrix associates a cost to these different outcomes. In Table 1b, a cost matrix is shown for the setting with class-dependent costs. When costs are instance-dependent, each instance will have a different cost matrix, denoted by the index i in Table 1c. Note that this framework also allows the

URI: https://hdl.handle.net/10125/79523
978-0-9981331-5-7

Table 1: Extension of the confusion matrix (1a) towards a cost matrix for class- (1b) and instance-dependent costs (1c)

(a) Confusion matrix

		Actual	
		0	1
Predicted	0	TN	FN
	1	FP	TP

(b) Class-dependent cost matrix

		Actual	
		0	1
Predicted	0	c^{TN}	c^{FN}
	1	c^{FP}	c^{TP}

(c) Instance-dependent cost matrix

		Actual	
		0	1
Predicted	0	c_i^{TN}	c_i^{FN}
	1	c_i^{FP}	c_i^{TP}

inclusion of benefits or profits in the form of negative costs. Existing work on cost-sensitive learning can be summarized based on whether the costs are class- or instance-dependent (see Table 2).

2.1. Class- and instance-dependent cost-sensitive learning

A variety of cost-sensitive machine learning techniques have been proposed for dealing with class-dependent costs. Class-dependent costs imply that one class is more important in terms of costs and, because of that, a cost-sensitive model should focus more on correctly classifying this class compared to a cost-insensitive model. In the simple case of a linear model, class-dependent costs result in a parallel shift of the decision boundary away from the more costly class (see Figure 1). Note that the literature on class-dependent cost-sensitive learning is intertwined with the literature on learning with class imbalance and, by using the appropriate costs, similar techniques can be used (see [37] for a recent survey on class imbalance).

Compared to the class-dependent setting, much less research has looked at cost-sensitive learning when costs are instance-dependent. Conceptually, many of the techniques for dealing with class-dependent costs can and have been transferred to the instance-dependent setting. However, instance-dependent costs bring an additional degree of complexity as costs not only depend on the class but also on the characteristics of the instance itself (e.g. on the transaction's amount in fraud detection). Whereas class-dependent costs result in a parallel shift of a linear decision boundary compared to the cost-insensitive optimal decision boundary, instance-dependent costs can additionally result in a rotation of this hyperplane (see Figure 1). This toy example illustrates that when costs are instance-dependent, the learner needs to simultaneously consider both the class distribution (explicitly) and the cost distribution (implicitly).

2.2. Cost-sensitive objective functions

Cost-sensitive classification models can be obtained by incorporating costs in the objective function that is used for training. In this work, we focus specifically on this type of method for several reasons. This approach allows to directly optimize an explicitly defined measure that can be easily connected to theory or application-specific goals. Moreover, objective functions are model-independent, making it straightforward to compare performance across different types of classifiers.

In general, machine learning algorithms can be understood in terms of risk minimization [38]. In this framework, the goal of a learning algorithm is to find the classifier that minimizes the risk. Formally, for a distribution $p(\mathbf{x}, y)$ and a classifier $f_\theta : \mathbf{X} \to [0,1] : \mathbf{x} \mapsto f_\theta(\mathbf{x})$ defined by parameters $\theta \in \Theta$, the risk to be minimized is:

$$R(\theta) = \int\int \mathcal{L}(y, \mathbf{x}, \theta) p(\mathbf{x}, y) d\mathbf{x} dy,$$

where $\mathcal{L}(y, \mathbf{x}, \theta)$ represents the loss or objective function for a classifier $f_\theta(\mathbf{x})$ and data $(\mathbf{x}, y)$ [21]. In reality, the true joint probability distribution $p(\mathbf{x}, y)$ is unknown. Consequently, the learner relies on the empirical density to minimize the risk given the available training data. This is the principle of empirical risk minimization (ERM) [38]. For a dataset $(\mathbf{x}_i, y_i) \in \mathcal{D}$ with $i \in N$, the empirical risk is defined as:

$$R_{emp}(\theta) = \mathop{\mathbb{E}}_{x,y\sim D}\Big[\mathcal{L}(y_i, \mathbf{x}_i, \theta)\Big] = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(y_i, \mathbf{x}_i, \theta).$$

Clearly, it is essential to choose an appropriate loss function $\mathcal{L}$. A first and straightforward candidate is the zero-one loss comparing the actual y and predicted label $\hat{y}$: $\mathcal{L}^{0/1}(y, \hat{y}) = I(y \neq \hat{y})$, though it is common to use a convex surrogate for computational efficiency [39]. A popular choice is the **cross-entropy** loss, which is equivalent to the maximum likelihood (ML) method [40]. In binary classification, we have $\mathcal{L}^{CE}(y_i, \mathbf{x}_i, \theta)$:

$$y_i \text{log } f_\theta(\mathbf{x}_i) + (1 - y_i)\text{log}\big(1 - f_\theta(\mathbf{x}_i)\big).$$

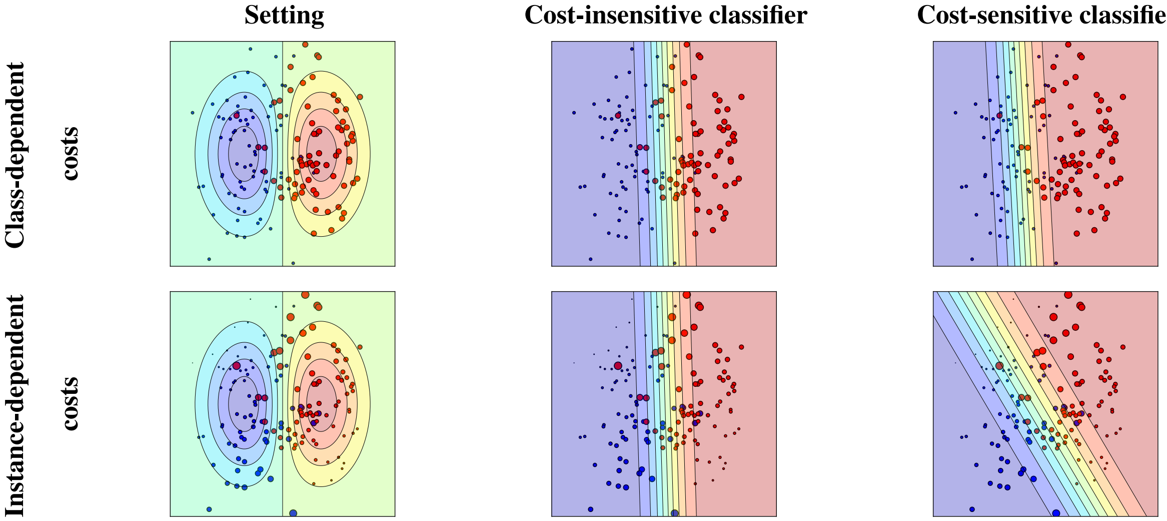

Figure 1: **Toy examples with class-dependent (top) and instance-dependent costs (bottom).** (Left) Two classes and the probability distribution are shown. The instance size is proportional to its misclassification cost. (Middle) The decision-boundary for a *cost-insensitive* classifier mimics the probability distribution. (Right) For a *cost-sensitive* classifier and class-dependent costs, the decision-boundary lies further from the more costly class. With instance-dependent costs, the decision-boundary is related to both the probability and cost distributions.

However, as argued above, a disadvantage of the maximum likelihood approach is that it does not take into account the costs of different decisions. Consequently, using this loss function, the empirical risk fails to reflect the true risk. To solve this issue, the ERM framework can be extended to include costs: given a dataset $(\mathbf{x}_i, y_i, \mathbf{c}_i) \in \mathcal{D}$ for $i \in N$ with an instance's cost matrix denoted as $\mathbf{c}_i$, a cost-sensitive loss function $\mathcal{L}(y, \mathbf{x}, \mathbf{c}, \theta)$ can be defined [21]. This way, the empirical risk can be made cost-sensitive.

A first cost-sensitive loss function is obtained by weighting the training examples by their misclassification cost [3, 29]. This can be formulated in terms of a **weighted cross-entropy** loss function $\mathcal{L}^{wCE}(y_i, \mathbf{x}_i, \mathbf{c}_i, \theta)$ [21]:

$$c_i^{FN} y_i \log f_\theta(\mathbf{x}_i) + c_i^{FP}(1 - y_i)\log\big(1 - f_\theta(\mathbf{x}_i)\big). \quad (1)$$

Note that this approach is equivalent to oversampling proportional to misclassification costs [21].

A second cost-sensitive loss function builds on the idea that the optimal cost-sensitive prediction minimizes the expected cost [3]. Using this, an alternative loss function can be defined which equals the expected cost [32, 36]. The corresponding empirical risk is the **average expected cost** $\mathcal{L}^{AEC}(y_i, \mathbf{x}_i, \mathbf{c}_i, \theta)$:

$$\begin{aligned} &y_i\Big(f_\theta(\mathbf{x}_i)c_i^{TP} + \big(1 - f_\theta(\mathbf{x}_i)\big)c_i^{FN}\Big) \\ &\quad + (1 - y_i)\Big(f_\theta(\mathbf{x}_i)c_i^{FP} + \big(1 - f_\theta(\mathbf{x}_i)\big)c_i^{TN}\Big). \end{aligned} \quad (2)$$

3. Methodology

The goal of this work is to empirically analyze the difference in training with class- and instance-dependent costs on the resulting classification model. To compare performance across a range of methodologies, we use a variety of models obtained by combining a cost-sensitive objective function with a type of classifier. To this end, we use two cost-sensitive objective functions the weighted cross-entropy $\mathcal{L}^{wCE}$ and the average expected cost $\mathcal{L}^{AEC}$ (see equations 1 and 2 respectively). These will be implemented using three different types of classifiers: logistic regression, neural network and gradient boosting. These classifiers are popular choices and are representative for the prominent families of machine learning techniques: linear and non-linear models are compared, as well as single classifiers and ensembles. This choice is also motivated by various benchmarking studies (e.g. [41, 42]). This results in a total of 6 cost-sensitive models (see Table 3). For neural networks and gradient boosting, hyperparameter selection is based on the best value of the objective function on a validation set.

3.1. Experimental procedure and evaluation metrics

For the empirical evaluation, a 2 × 5-fold stratified cross-validation procedure is used (see Algorithm 1). This is repeated for each dataset. We use a variety

Table 2: **An overview of cost-sensitive learning methodologies.** We present an overview summary of various cost-sensitive learning methods with respect to costs and (when applicable) the used classifier.

Ref	Costs		Classifier
	Class	Instance	
[4]	✓	✗	DR
[5]	✓	✗	DT
[6]	✓	✗	DR, NN
[7]	✓	✗	BO
[8]	✓	✗	DT
[9]	✓	✗	NN
[10]	✓	✗	NN
[11]	✓	✗	BO
[12]	✓	✗	-
[13]	✓	✗	SVM
[14]	✓	✗	DT
[15]	✓	✗	NB
[16]	✓	✗	DT
[17]	✓	✗	NN
[18]	✓	✗	-
[19]	✓	✗	BO
[20]	✓	✗	-
[21]	✓	✗	-
[22]	✓	✗	SVM
[23]	✓	✗	BO
[24]	✓	✗	BO
[25]	✓	✗	LR
[26]	✓	✗	DT
[27]	✗	✓	BO
[28]	✗	✓	-
[29]	✗	✓	BO
[30]	✗	✓	SVM
[31]	✗	✓	DT
[32]	✗	✓	LR
[33]	✗	✓	-
[34]	✗	✓	DT
[35]	✗	✓	BO
[36]	✗	✓	LR, BO

Costs CD: class-dependent, ID: instance-dependent
Classifier BO: boosting, DR: decision rule, DT: decision tree, LR: logistic regression, NB: Naive Bayes, NN: neural network, SVM: support vector machine

Table 3: **Overview of the different models.** We combine different objective functions and types of classifiers. Each model will be trained once with instance- and once with class-dependent costs.

	Logistic regression	**Neural network**	**Gradient boosting**
$\mathcal{L}^{wCE}$	wlogit	wnet	wboost
$\mathcal{L}^{AEC}$	cslogit	csnet	csboost

Algorithm 1: Experimental procedure

```
Result: Evaluation metrics
Load data;
Initialize cost matrix;
Split data in 5 stratified folds;
for each fold $i \in 1:5$ do
    for each repetition $j \in 1:2$ do
        Test data = fold $i$;
        Training data = $75\%$ of remaining data;
        Validation data = $25\%$ remaining data;

        # Preprocess data:
        Convert categorical features (using WoE
          encoding);
        Standardize data: $z = \frac{x-\mu}{\sigma}$;
        if training with class-dependent costs
          then
            Average cost matrix over training
              and validation set;
        end

        # Train and evaluate models:
        Train models;
        Set decision thresholds;
        Evaluate model outputs and predictions
          for different thresholds;
    end
end
Summarize evaluation metrics over all folds;
```

of metrics to evaluate the models. On the one hand, several cost-insensitive metrics will be used to assess the models' ability to accurately classify instances. First, the area under the ROC curve (AUC) and average precision (AP) are used; these respectively summarize the ROC and precision-recall curves. The latter may be more informative given the high degree of class imbalance [43] that is typically encountered in cost-sensitive applications. To evaluate the accuracy of predictions, we also use the F1-score.

On the other hand, performance will also be judged in terms of costs. First, the average expected cost (AEC, see equation 2) will be used. Second, Spearman's rank correlation coefficient ρ will be used to look at the correlation between probabilities and costs for positive instances. The aim of this metric is to analyze whether cost-sensitive models prioritize correctly classifying costlier instances. Finally, we also look at the cost savings incurred by the model. These compare the total costs incurred by the model to classifying all instances as the cheapest default class (either 0 or 1) [32]:

$$\text{Savings} = \frac{\text{Cost}\big(f_\theta(\mathbf{x})\big) - \min\{\text{Cost}\big(f_0(\mathbf{x})\big), \text{Cost}\big(f_1(\mathbf{x})\big)\}}{\text{Cost}\big(f_\theta(\mathbf{x})\big)} \tag{3}$$

The domain of this ratio is $[-\infty, 1]$ where 1 is the perfect model, but when the model does better than predicting a default class we obtain savings in $]0, 1]$.

To test the statistical significance of the results, we use the Wilcoxon Signed-Ranks Test for pairwise comparison to compare two versions of each model: one trained by incorporating class-dependent costs and one with instance-dependent costs [44]. Significance levels of 5% and 10% are used.

4. Empirical results

In this section, the empirical results of are presented. First, the data sets are presented and the cost matrices corresponding to each application area are described. Second, the experimental results are presented and these findings are used to answer the proposed research question.

4.1. Data

The data come from a diverse set of classification tasks with instance-dependent costs: fraud detection, direct marketing, customer churn and credit scoring (see Table 4). In each data set, there is class imbalance with the positive class being the minority, though some cases are more extreme than others. All data sets are publicly available (see Appendix A). The cost matrices depend on the application area and are adopted from earlier work (for an overview, see Table 5). The intuition behind these is provided below.

Fraud detection In fraud detection, a positive prediction triggers an investigation that has a fixed cost c_f, while a missed fraudulent transaction incurs a cost equal to its amount A_i (see Table 5a). For both data sets, c_f is set to 10 following [36].

Direct marketing A similar reasoning applies here: any direct marketing action results in a fixed cost c_f and missing a potential success incurs an instance-dependent cost (see Table 5b). Whereas *marketing1* uses the amount A_i and $c_f = 0.68$ following both [28] and [45], *marketing2* instead uses the expected interest given A_i and $c_f = 1$, following [34].

Customer churn For customer churn prediction, c_i^{FP} and c_i^{FN} are respectively set at 2 and 12 times the monthly amount A_i for *churn1* following [45] (see Table 5c). For *churn2*, the cost matrix provided with the data set is used (not shown here, see [46]).

Credit scoring Finally, for credit scoring, the costs of a FP and FN are calculated following [32] with both a function of the loan amount A_i.

4.2. Results

To look at the effect of using instance-dependent costs during training as opposed to training with class-dependent costs, we start by measuring performance in terms of cost-insensitive metrics (see Tables 6 and A2). Although the results are fairly similar for the two settings, training with class-dependent costs gives better results for these metrics for almost all cases. Based on this observation, it can be concluded that training with instance-dependent costs may be disadvantageous in terms of errors. This effect is observed in particular for wlogit, wnet, wboost and csboost.

Next, we look at performance in terms of cost-sensitive metrics (see Tables 7 and A1). Here, training with instance-dependent costs gives comparatively better results. Using instance-dependent costs consistently leads to lower average expected costs (though the difference is not always significant). Also in terms of Spearman's ρ, it is better for all models, and this difference is significant except for csnet. In terms of savings, instance-dependent costs give better results. The only exception is csnet which has very similar performance for the two types of costs.

5. Conclusion

In this work, we presented an extensive empirical evaluation comparing different instance-dependent and class-dependent cost-sensitive learning methods. We observed that using instance-dependent instead of class-dependent costs during gives better results in terms of cost-sensitive metrics, though not for traditional accuracy metrics. These results highlight the importance of considering the right objective for an application.

Future work will look at the inclusion of cost-sensitive decision-making thresholds, which can also incorporate both instance- or class-dependent costs. Additionally, it would be interesting to investigate the influence of the characteristics of the cost distribution and cost matrix on the performance of different instance-dependent cost-sensitive learning methodologies.

Table 4: **Overview of the different datasets**: size (N), dimensionality (D) and degree of class imbalance (% Pos)

Application	**Dataset**	**Abbr.**	N	D	**% Pos**
Fraud detection	Kaggle Credit Card Fraud	*KCCF*	282982	29	0.16
	Kaggle IEEE Fraud Detection	*KIFD*	590540	431	3.50
Direct marketing	KDD Cup 1998	*KDD*	191779	22	5.07
	UCI Bank Marketing	*UBM*	45211	15	11.70
Churn prediction	Kaggle Telco Customer Churn	*KTCC*	7032	19	26.58
	TV Subscription Churn	*TSC*	9379	46	4.79
Credit scoring	Kaggle Give Me Some Credit	*GMSC*	112915	10	6.74
	UCI Default of Credit Card Clients	*DCCC*	30000	23	22.12
	VUB Credit Scoring	*VSC*	18917	16	16.95

Table 5: **Cost matrices for the different application areas.** For each application, different costs are associated with different outcomes. These are presented here. A_i and Int_i represent an instance-dependent amount; c_f is a fixed cost.

(a) Fraud detection

		y	
		0	1
$\hat{y}$	0	0	A_i
	1	c_f	c_f

(b) Direct marketing

		y	
		0	1
$\hat{y}$	0	0	A_i/Int_i
	1	c_f	c_f

(c) Customer churn

		y	
		0	1
$\hat{y}$	0	0	$12A_i$
	1	$2A_i$	0

(d) Credit scoring

		y	
		0	1
$\hat{y}$	0	0	c_i^{FN}
	1	c_i^{FP}	0

Table 6: **Instance-dependent or class-dependent costs: cost-insensitive metrics per model.** Significantly better results are denoted in **bold** (5%) and *italic* (10%).

Metric	**Costs**	**wlogit**	**cslogit**	**wnet**	**csnet**	**wboost**	**csboost**
AUC	ID	0.76	0.72	0.76	0.77	0.77	0.76
	CD	**0.77**	*0.73*	**0.78**	0.77	**0.78**	**0.79**
AP	ID	0.38	0.27	0.40	*0.38*	0.42	0.38
	CD	**0.42**	0.27	**0.45**	0.36	0.45	**0.44**
F1	ID	0.39	0.39	0.39	0.43	0.43	0.38
	CD	0.41	0.39	**0.42**	0.43	0.45	**0.44**

Significance levels: **5%**, *10%*

Table 7: **Instance-dependent or class-dependent costs: cost-sensitive metrics per model.** Significantly better results are denoted in **bold** (5%) and *italic* (10%). AEC is normalized between 0 and 1 per dataset (lower is better).

Metric	**Costs**	**wlogit**	**cslogit**	**wnet**	**csnet**	**wboost**	**csboost**
AEC	ID	*0.56*	**0.07**	*0.47*	0.18	*0.41*	**0.06**
	CD	0.68	0.25	0.59	0.18	0.48	0.21
Spearman's ρ	ID	**0.09**	**0.11**	**0.16**	-0.06	**0.13**	**0.23**
	CD	-0.10	-0.05	-0.10	-0.07	-0.07	-0.10
Savings	ID	**0.37**	**0.38**	**0.40**	0.34	0.32	*0.38*
	CD	0.33	0.31	0.35	0.34	0.29	0.34

Significance levels: **5%**, *10%*

Acknowledgements This work was supported by the BNP Paribas Fortis Chair in Fraud Analytics and FWO research project G015020N. The computational resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government - department EWI.

References

[1] C. X. Ling and V. S. Sheng, "Cost-sensitive learning and the class imbalance problem," *Encyclopedia of machine learning*, vol. 2011, pp. 231–235, 2008.

[2] N. Nikolaou, N. Edakunni, M. Kull, P. Flach, and G. Brown, "Cost-sensitive boosting algorithms: Do we really need them?," *Machine Learning*, vol. 104, no. 2, pp. 359–384, 2016.

[3] C. Elkan, "The foundations of cost-sensitive learning," in *International joint conference on artificial intelligence*, vol. 17, pp. 973–978, Lawrence Erlbaum Associates Ltd, 2001.

[4] M. Pazzani, C. Merz, P. Murphy, K. Ali, T. Hume, and C. Brunk, "Reducing misclassification costs," in *Machine Learning Proceedings 1994*, pp. 217–225, Elsevier, 1994.

[5] P. D. Turney, "Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm," *Journal of artificial intelligence research*, vol. 2, pp. 369–409, 1994.

[6] T. Fawcett and F. Provost, "Adaptive fraud detection," *Data mining and knowledge discovery*, vol. 1, no. 3, pp. 291–316, 1997.

[7] K. M. Ting and Z. Zheng, "Boosting trees for cost-sensitive classifications," in *European conference on machine learning*, pp. 190–195, Springer, 1998.

[8] K. M. Ting, "Inducing cost-sensitive trees via instance weighting," in *European symposium on principles of data mining and knowledge discovery*, pp. 139–147, Springer, 1998.

[9] M. Kukar, I. Kononenko, *et al.*, "Cost-sensitive learning with neural networks.," in *ECAI*, vol. 15, pp. 88–94, Citeseer, 1998.

[10] S. Lawrence, I. Burns, A. Back, A. C. Tsoi, and C. L. Giles, "Neural network classification and prior class probabilities," in *Neural networks: tricks of the trade*, pp. 299–313, Springer, 1998.

[11] G. K. J. Shawe-Taylor and G. Karakoulas, "Optimizing classifiers for imbalanced training sets," *Advances in neural information processing systems*, vol. 11, no. 11, p. 253, 1999.

[12] P. Domingos, "Metacost: A general method for making classifiers cost-sensitive," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 155–164, 1999.

[13] K. Veropoulos, C. Campbell, N. Cristianini, *et al.*, "Controlling the sensitivity of support vector machines," in *Proceedings of the international joint conference on AI*, vol. 55, p. 60, Stockholm, 1999.

[14] C. Drummond and R. C. Holte, "Exploiting the cost (in) sensitivity of decision tree splitting criteria," in *ICML*, vol. 1, 2000.

[15] X. Chai, L. Deng, Q. Yang, and C. X. Ling, "Test-cost sensitive naive bayes classification," in *Fourth IEEE International Conference on Data Mining (ICDM'04)*, pp. 51–58, IEEE, 2004.

[16] K. M. Ting, "An instance-weighting method to induce cost-sensitive trees," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 3, pp. 659–665, 2002.

[17] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Transactions on knowledge and data engineering*, vol. 18, no. 1, pp. 63–77, 2005.

[18] V. S. Sheng and C. X. Ling, "Thresholding for making classifiers cost-sensitive," in *AAAI*, vol. 6, pp. 476–481, 2006.

[19] Y. Sun, M. S. Kamel, A. K. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognition*, vol. 40, no. 12, pp. 3358–3378, 2007.

[20] N. V. Chawla, D. A. Cieslak, L. O. Hall, and A. Joshi, "Automatically countering imbalance and its empirical relationship to cost," *Data Mining and Knowledge Discovery*, vol. 17, no. 2, pp. 225–252, 2008.

[21] J. P. Dmochowski, P. Sajda, and L. C. Parra, "Maximum likelihood in cost-sensitive learning: Model specification, approximations, and upper bounds.," *Journal of Machine Learning Research*, vol. 11, no. 12, 2010.

[22] Y.-F. Li, J. Kwok, and Z.-H. Zhou, "Cost-sensitive semi-supervised support vector machine," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 24, 2010.

[23] J. Zheng, "Cost-sensitive boosting neural networks for software defect prediction," *Expert Systems with Applications*, vol. 37, no. 6, pp. 4537–4543, 2010.

[24] B. Krawczyk, M. Woźniak, and G. Schaefer, "Cost-sensitive decision tree ensembles for effective imbalanced classification," *Applied Soft Computing*, vol. 14, pp. 554–562, 2014.

[25] E. Stripling, S. vanden Broucke, K. Antonio, B. Baesens, and M. Snoeck, "Profit maximizing logistic model for customer churn prediction using genetic algorithms," *Swarm and Evolutionary Computation*, vol. 40, pp. 116–130, 2018.

[26] S. Höppner, E. Stripling, B. Baesens, S. vanden Broucke, and T. Verdonck, "Profit driven decision trees for churn prediction," *European Journal of Operational Research*, vol. 284, no. 3, pp. 920–933, 2020.

[27] W. Fan, S. J. Stolfo, J. Zhang, and P. K. Chan, "Adacost: misclassification cost-sensitive boosting," in *Icml*, vol. 99, pp. 97–105, Citeseer, 1999.

[28] B. Zadrozny and C. Elkan, "Learning and making decisions when costs and probabilities are both unknown," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 204–213, 2001.

[29] B. Zadrozny, J. Langford, and N. Abe, "Cost-sensitive learning by cost-proportionate example weighting," in *Third IEEE international conference on data mining*, pp. 435–442, IEEE, 2003.

[30] U. Brefeld, P. Geibel, and F. Wysotzki, "Support vector machines with example dependent costs," in *European Conference on Machine Learning*, pp. 23–34, Springer, 2003.

[31] Y. Sahin, S. Bulkan, and E. Duman, “A cost-sensitive decision tree approach for fraud detection,” *Expert Systems with Applications*, vol. 40, no. 15, pp. 5916–5923, 2013.

[32] A. C. Bahnsen, D. Aouada, and B. Ottersten, “Example-dependent cost-sensitive logistic regression for credit scoring,” in *2014 13th International Conference on Machine Learning and Applications*, pp. 263–269, IEEE, 2014.

[33] A. C. Bahnsen, A. Stojanovic, D. Aouada, and B. Ottersten, “Improving credit card fraud detection with calibrated probabilities,” in *Proceedings of the 2014 SIAM International Conference on Data Mining*, pp. 677–685, SIAM, 2014.

[34] A. C. Bahnsen, D. Aouada, and B. Ottersten, “Example-dependent cost-sensitive decision trees,” *Expert Systems with Applications*, vol. 42, no. 19, pp. 6609–6619, 2015.

[35] Y. Zelenkov, “Example-dependent cost-sensitive adaptive boosting,” *Expert Systems with Applications*, vol. 135, pp. 71–82, 2019.

[36] S. Höppner, B. Baesens, W. Verbeke, and T. Verdonck, “Instance-dependent cost-sensitive learning for detecting transfer fraud,” *European Journal of Operational Research*, 2021.

[37] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, “A survey on addressing high-class imbalance in big data,” *Journal of Big Data*, vol. 5, no. 1, pp. 1–30, 2018.

[38] V. Vapnik, “Principles of risk minimization for learning theory,” in *Advances in neural information processing systems*, pp. 831–838, 1992.

[39] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe, “Convexity, classification, and risk bounds,” *Journal of the American Statistical Association*, vol. 101, no. 473, pp. 138–156, 2006.

[40] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.

[41] S. Lessmann, B. Baesens, H.-V. Seow, and L. C. Thomas, “Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research,” *European Journal of Operational Research*, vol. 247, no. 1, pp. 124–136, 2015.

[42] B. R. Gunnarsson, S. vanden Broucke, B. Baesens, M. Óskarsdóttir, and W. Lemahieu, “Deep learning for credit scoring: Do or don’t?,” *European Journal of Operational Research*, 2021.

[43] J. Davis and M. Goadrich, “The relationship between precision-recall and roc curves,” in *Proceedings of the 23rd international conference on Machine learning*, pp. 233–240, 2006.

[44] J. Demšar, “Statistical comparisons of classifiers over multiple data sets,” *The Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[45] G. Petrides and W. Verbeke, “Misclassification cost-sensitive ensemble learning: A unifying framework,” *arXiv preprint arXiv:2007.07361*, 2020.

[46] A. C. Bahnsen, D. Aouada, and B. Ottersten, “A novel cost-sensitive framework for customer churn predictive modeling,” *Decision Analytics*, vol. 2, no. 1, pp. 1–15, 2015.

[47] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, “Calibrating probability with undersampling for unbalanced classification,” in *2015 IEEE Symposium Series on Computational Intelligence*, pp. 159–166, IEEE, 2015.

[48] S. Moro, P. Cortez, and P. Rita, “A data-driven approach to predict the success of bank telemarketing,” *Decision Support Systems*, vol. 62, pp. 22–31, 2014.

[49] I.-C. Yeh and C.-h. Lien, “The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients,” *Expert Systems with Applications*, vol. 36, no. 2, pp. 2473–2480, 2009.

[50] G. Petrides, D. Moldovan, L. Coenen, T. Guns, and W. Verbeke, “Cost-sensitive learning for profit-driven credit scoring,” *Journal of the Operational Research Society*, pp. 1–13, 2020.

A. Data

The data sets that are used in the experiments presented in this paper are publicly available online.

- **Kaggle Credit Card Fraud (KCCF)** [47]
  https://www.kaggle.com/mlg-ulb/creditcardfraud
- **Kaggle IEEE Fraud Detection (KIFD)**
  https://www.kaggle.com/c/ieee-fraud-detection
- **UCI KDD98 Direct Mailing (KDD)**
  http://kdd.ics.uci.edu/databases/kddcup98/kddcup98.html
- **UCI Bank Marketing (UBM)** [48]
  https://archive.ics.uci.edu/ml/datasets/Bank+Marketing
- **Kaggle Telco Customer Churn (KTCC)**
  https://www.kaggle.com/blastchar/telco-customer-churn
- **TV Subscription Churn (TSC)** [46]
  https://github.com/albahnsen/CostSensitiveClassification/blob/master/costcla/datasets/data/churn_tv_subscriptions.csv.gz
- **Kaggle Give Me Some Credit (GMSC)**
  https://www.kaggle.com/c/GiveMeSomeCredit
- **UCI Default of Credit Card Clients (DCCC)** [49]
  https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
- **VUB Credit Scoring (VSC)** [50]
  https://github.com/vub-dl/data-csl-pdcs

B. Additional results

Here we present the average results per dataset for the models trained with instance- and class-dependent (see Table A1 and A2) to complement the results averaged per model in the main body. These results are in line with earlier findings: training with instance-dependent costs gives better results in terms of cost-sensitive metrics compared to training with class-dependent costs, but worse in terms of cost-insensitive metrics.

Table A1: **Instance-dependent or class-dependent costs: cost-sensitive metrics per dataset.** Significantly better results are denoted in **bold** (5%) and *italic* (10%).

Metric	Costs	KCCF	GMSC	KIFD	KTCC	KDD	TSC	UBM	DCCC	VSC
AEC	ID	0.08	458.90	*2.53*	82.05	*0.72*	60.22	**0.52**	**15674.65**	*0.08*
	CD	0.08	460.81	3.05	81.32	0.72	60.42	0.67	16724.90	0.09
Spearman's ρ	ID	**0.17**	**-0.04**	*0.09*	0.12	**0.03**	-0.35	*0.55*	*0.05*	**0.36**
	CD	-0.07	-0.15	-0.17	0.12	-0.14	**-0.30**	0.18	-0.30	0.11
Savings	ID	0.66	*0.47*	*0.59*	0.20	*-0.03*	0.06	**0.55**	**0.35**	**0.41**
	CD	0.66	0.47	0.49	0.21	-0.05	0.06	0.43	0.30	0.37

Significance levels: **5%**, *10%*

Table A2: **Instance-dependent or class-dependent costs: cost-insensitive metrics per dataset.** Significantly better results are denoted in **bold** (5%) and *italic* (10%).

Metric	Costs	KCCF	GMSC	KIFD	KTCC	KDD	TSC	UBM	DCCC	VSC
AUC	ID	0.96	0.81	0.89	0.82	0.51	0.61	0.73	0.72	0.76
	CD	0.96	**0.81**	*0.90*	0.82	**0.53**	0.62	**0.76**	*0.75*	*0.77*
AP	ID	0.72	0.30	0.45	0.61	0.05	0.08	0.29	0.46	0.38
	CD	0.77	0.31	*0.51*	0.60	**0.06**	0.08	*0.37*	0.49	0.39
F1 0.5	ID	0.74	0.33	0.43	0.59	0.10	0.13	0.37	0.49	0.44
	CD	0.81	0.33	0.47	0.59	*0.10*	0.14	*0.41*	*0.51*	*0.45*

Significance levels: **5%**, *10%*