Cost-sensitive Classification Tree Induction as a Bi-level Optimization Problem

Rihab Said
SMART Lab, ISG
University of Tunis, Tunisia

Maha Elarbi
SMART Lab, ISG
University of Tunis, Tunisia

Slim Bechikh
SMART Lab, ISG
University of Tunis, Tunisia

Carlos A. Coello Coello
CINVESTAV-IPN
Mexico

Lamjed Ben Said
SMART Lab, ISG
University of Tunis, Tunisia

ABSTRACT

Data imbalance is still so far a challenging issue in data classification. Recent works suggest the use of cost sensitive learning with genetic programming as an effective tool to design classification trees with automatically learned costs. Although promising results were obtained, evaluating a classification tree with a single cost matrix is not a wise choice. Indeed, the tree quality evaluation requires trying several misclassification cost matrices to be more precise and fair. Motivated by this observation, we propose in this paper a bi-level modeling of the cost-sensitive classification tree induction problem where the upper level evolves the classification trees, while the cost matrix of each tree is optimized at the lower level. Our bi-level modeling is solved using an improved version of an existing co-evolutionary algorithm, and the resulting method is named Bi-COS (Bi-level COst Sensitive). The obtained comparative experimental results on several imbalanced benchmark datasets show the merits of Bi-COS with respect to the state-of-the art.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning; Genetic programming**.

KEYWORDS

Imbalanced data classification, cost-sensitive learning, bi-level optimization, classification tree, optimized misclassification costs

ACM Reference Format:
Rihab Said, Maha Elarbi, Slim Bechikh, Carlos A. Coello Coello, and Lamjed Ben Said. 2022. Cost-sensitive Classification Tree Induction as a Bi-level Optimization Problem . In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM. . . $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Several classification problems face the class imbalance issue which means having a majority class with high accuracy and a minority class with poor accuracy. It is challenging to address this issue because in many practical situations, the minority class must gain core atttention. For instance, in oil spills detection and medical diagnosis, the minority class is more important than the majority one. To address the class imbalance issue, cost sensitive learning represents an important method that treats classification errors in different manners to make the classification algorithms sensitive to them. However, cost information requires domain experts which makes the misclassification cost unknown. Recent works have considered the use of Genetic Programming (GP) and cost sensitive learning to design classification trees with automatically learned misclassification costs. This combination has shown interesting results. However, existing methods share the same incovenient which manifests in generating a single cost information for each classifier. Since the classifier performance depends on the cost quality, the classifier evaluation requires trying several misclassification cost matrices to be more precise and fair. To address this issue, we model the cost-sensitive classification tree induction problem as a bi-level optimization problem that is solved using the baseline of an existing bi-level co-evolutionary algorithm called CEMBA (Co-Evolutionary Migration-Based Algorithm) [8] while using a novel migration process. The overall algorithm is named Bi-COS and performs classifier construction at the upper level, while the learning costs task is performed at the lower level. The main goal is to generate several costs for each classifier at the lower level and the obtained optimal cost will be sent to the upper level to evaluate the corresponding classifier. In fact, we focus on the binary classification because many real-world imbalanced data situations are binary and even multi-class classification could be seen as multiple binary classification tasks. The main contributions are stated as follows:

(1) Proposing a bi-level modeling of the cost-sensitive classification tree induction problem that evolves the classification trees at the upper level and optimizes the misclassification costs for each classifier at the lower level to allow more precise and fair evaluation of any classification tree.
(2) Designing an enhanced and adapted version of an existing algorithm to solve the proposed bi-level modeling by

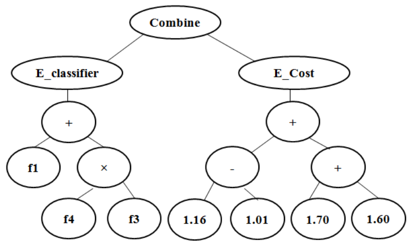

Figure 1: Single-level modeling of existing approaches [6].

modifying its migration strategy with the aim of ensuring efficacious variation and diversification of the classification.

(3) Assessing the performance of Bi-COS on ten commonly-used imbalanced datasets with up to 12600 features and 336 instances.

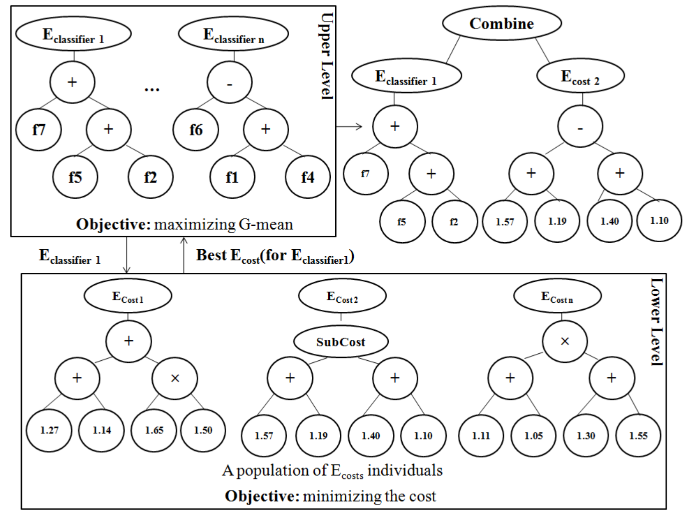

Figure 2: Main idea of the proposed Bi-COS.

2 BACKROUND AND RELATED WORKS

Several applications face an unequality issue among classes [4]. For instance, in clinical databases, a large amount of medical patients' information is stored. However, compared with normal cases, the disease cases are fairly rare. This situation presents an imbalanced data issue in which the used dataset does not have an equal distribution among classes. The imbalanced classification faces many challenges such as the difficulty of the imbalance ratio learning and noisy class label. In the literature, several approaches have been proposed for imbalanced data classification by using three main strategies which are [4]: (1) resampling techniques, (2) developing new fitness functions, and (3) cost sensitive learning. Cost sensitive learning consists in considering the information cost in order to minimize the total cost of misclassification caused by several mistakes. Indeed, there are two types of misclassification costs which are (1) class-dependent cost in which instances of the same class have the same misclassification cost, and (2) class-independent cost in which, for each instance, a different misclassification cost is assigned. The investigation of cost sensitive learning in GP has gained the attention of researchers. For instance, Li et al. [5] proposed two GP methods for cost sensitive classification. These two methods are applied for the manipulation of training data and the modification of the learning algorithm. However, the adopted cost matrix is manually designed. More recently, Pei et al. [6] designed the Genetic-Programming-Cost Sensitive (GP-CS) for the construction of classifiers when the cost information is unknown. The goal was to automatically learn cost values. However, a single cost value is generated which makes the classifier evaluation not precise.

3 PROPOSED APPROACH

3.1 Main idea and motivations

Existing methods have a single-level modeling characterized by generating a single cost value for each classifier [6] (cf. Figure 1). The left sub-tree represents a classifier $E_{classsifier}$ that selects features $f1$, $f3$, and $f4$ and the right sub-tree is its corresponding cost value E_{cost}. By generating a single E_{cost}, the $E_{classsifier}$ cannot be precisely evaluated since the performance of $E_{classsifier}$ depends on the quality of its E_{cost}. In other words, the $E_{classsifier}$ evaluation requires trying several misclassification cost matrices to be more fair and precise. To address this issue, we propose to model the cost-sensitive classification tree induction problem as a bi-level optimization problem. The classifiers construction is performed at the upper level problem, while their cost values are learned at the lower level problem. Figure 2 illustrates an example for the evaluation of a single upper level individual (i.e., classifier) using the proposed Bi-COS. Indeed, the left sub-tree is generated at the upper level problem and then it is sent to the lower level problem. By using $E_{classsifier}$ as a fixed parameter, the lower level generates a population of cost values (i.e. right sub-trees) and then, the best E_{cost} for the corresponding $E_{classsifier}$ is determined and passed to the upper level. Based on the best E_{cost} received from the lower level, $E_{classsifier}$ is evaluated. By using tournament selection, the best individuals are selected, then, new populations are created by applying mutation, crossover, and elitism. In a bi-level optimization problem, the evaluation of each upper level solution requires the use of its corresponding optimal lower level solution which leads to a high number of evaluations [7], [2], [1]. For this reason, we have used an existing algorithm called CEMBA [8] as an effective and efficient evolutionary bi-level algorithm. To ensure the diversification of classification trees, we have proposed an improved migration strategy as follows. First, if a Left Sub-Tree (LST) is selected at the upper level while having a Right Sub-Tree (RST) at the lower level, then this LST remains selected; and if the LST is discarded and its corresponding RST is also discarded then this LST remains discarded. Second, when a LST exists without its RST, then, a random variable V is generated in the range [0,

Table 1: Terminal and function sets.

*	Terminal set		Function set		
	Name	Type	Name	Input type	Output type
Upper level (Left sub-tree)	A random constant	*Iput* (float)	+, −, ÷ (protected), and ×	[*Iput*, *Iput*]	*Iput*
	Features dataset	*Iput* (float)	If	[*Iput*, *Iput*, *Iput*]	*Iput*
			$E_{classsifier}$	[*Iput*]	*Predi* (float)
Lower level (Right sub-tree)	Random numbers which are uniformally distributed in the range [1, 2]	*Icost* (float)	×, +, $SubCost$, and $DivCost$	[*Icost*, *Icost*]	*Icost*
			$E_{classsifier}$	[*Icost*]	*Ocost* (float)

Table 2: Default parameters settings.

Upper and Lower levels Populations	$UP_1 = UP_2 = LP_1 = LP_2 = 25$
Upper and Lower level Generations	$UG = 25$, $LG = 25$
Stopping criterion	781250 evaluations
Initialization	Ramped half-and-half
Crossover type and probability	Sub-tree crossover, probability = 0.8
Mutation type and probability	Sub-tree mutation, probability = 0.2
Selection	Tournament selection (size = 6)
Maximum tree depth	10
Elitism	1
Runs	30

1]. If $V < 0.5$, then, the LST will be discarded; otherwise, it remains selected and we choose a random RST from the lower level population. Third, when the LST is discarded while its corresponding RST exists at the lower level; if $V < 0.5$, the LST will be selected and the RST will be used to compute the cost value; otherwise, the LST remains discarded and RST will be discarded.

3.2 Detailed description of Bi-COS

Bi-COS is based on Strongly-Typed GP. Details about terminal and function sets are given by Table 1. We mention that the root node of a tree is a function termed *Combine* that takes two arguments: (1) $E_{classifier}$ output and (2) E_{cost} output. Details about Bi-COS structure are given as follows.

3.2.1 Upper level.

- **Classifier construction:** The left sub-tree represents the classifier that selects several features.
- **Classification predictions:** The threshold-moving idea [6] is used to predict classification decisions. First, values of the left sub-tree output are normalized into [0, 1] using the min-max normalization. Based on the learned cost value C received from the lower level, a threshold is determined as follows: $TH = \frac{C}{C+1}$ [6]. Therefore, if the prediction is $\geq TH$, then the instance is classified into the majority class; otherwise it is classified into the minority class.
- **Variation operators:** To create new upper level populations, crossover, mutation and elitism are applied.
- **Fitness function:** After taking the classification decisions, individuals are evaluated using G-Mean.

3.2.2 Lower level.

- **Cost learning:** The right sub-tree represents the cost learning task. For each $E_{classifier}$, several E_{cost} values are generated. The use of the range of [1, 2] is explained by the fact that the misclassification cost for the minority class should be greater than or equal to the misclassification cost for the majority class (i.e. 1) [6]. *SubCost* and *DivCost* are slightly different from the original - and ÷ to avoid producing risky cost values for the minority class [6].
- **Variation operators:** To search for optimal learned costs, the lower level search space must be varied through the use of crossover, mutation, and elitism.
- **Evaluation:** Each lower level sub-tree generates a misclassification cost using the corresponding E_{cost} function and the individual with the minimum misclassification cost will be passed to the upper level.

4 EXPERIMENTAL STUDY

Experiments were performed using ten datasets [1]. Bi-COS was compared with two Cost-Sensitive GP methods (i.e. CS-GP [6], GP_{rcw} [5]) and six GP-methods utilizing several fitness functions (i.e. GP_{ave} [9], GP_{G-mean}, GP_{amse}, GP_{corr}, G_{Dist} [3], GP_{aucw} [10]). A trial-and-error method was adopted for tuning Bi-COS parameters (cf. Table 2) while the Friedman and holm statistical tests were also used.

Table 3 gives the obtained results of the Area Under a Curve (AUC) that measures the classifier ability to distinguish between classes. It is observed that Bi-COS outperforms existing methods. Compared to CS-GP, Bi-COS achieves the best performance on 6 datasets. For the Colon dataset, Bi-COS achieves 10.06 % higher AUC compared to CS-GP. On the other datasets, Bi-COS achieves slightly better performance (eg. 1.92% higher on yeoh-2002-v1). Compared to GP_{rcw}, Bi-COS achieves significantly better performance on all datasets (e.g. 16.01% higher on colon).

Compared to GP_{ave}, Bi-COS achieves significantly better performance in 8 datasets. For the 2 other datasets, GP_{ave} achieves slightly wost performance on new-thyroid1 (1.72% lower) and leukemia (1.57% lower). It is observed that the superiority of Bi-COS appears on lung (15.65% higher) and Yeoh-2002-v1 (15.74% higher). Furthermore, Bi-COS outperforms GP_{amse} on the majority of datasets (e.g. 15.19% higher in colon and 17.36% higher in lung).

Compared to GP_{G-mean}, Bi-COS achieves significantly better performance in 9 out of the 10 datasets. However, for the new-thyroid1 dataset, Bi-COS is slightly better with 1.58% higher result. Compared to GP methods which applies AUC approximation measures as a fitness function (i.e. GP_{corr}, GP_{dist} , and GP_{aucw}), Bi-COS achieves better results in 8 datasets. On the 2 other datasets, Bi-COS obtains 0.18% and 0.14% lower result than GP_{aucw}.

The previous observations are explained by the ability of Bi-COS in optimizing the misclassification cost for each classifier. Indeed, the best cost will be chosen to be used for the classifier evaluation. By following the bi-level scheme, we guarantee a fair and precise evaluation for any classifier.

[1] https://schlieplab.org/Static/Supplements/CompCancer/datasets.htm and https://sci2s.ugr.es/keel/imbalanced.php

Table 3: The obtained best, median, and standard (std) AUC results (%) (*IR* is the class imbalance ratio).

Dataset	Method	Best	Median	Std
Lung (IR =8)	Bi-COS	100	**98.55**	**2.10**
	CS-GP	100($\approx$)	96.80(-)	2.70(-)
	GP_{rcw}	100($\approx$)	79.95(-)	15.70(-)
	GP_{ave}	100($\approx$)	82.90(-)	15.00(-)
	GP_{G-mean}	98.61(-)	80.25(-)	18.60(-)
	GP_{amse}	100($\approx$)	81.16(-)	16.88(-)
	GP_{corr}	100($\approx$)	80.05(-)	17.81(-)
	GP_{dist}	100($\approx$)	83.79(-)	15.05(-)
*	GP_{aucw}	100($\approx$)	91.99(-)	3.70(-)
tomlins-2006-v1 (IR =8)	Bi-COS	100	**99.25**	**1.15**
	CS-GP	100($\approx$)	96.25(-)	4.77(-)
	GP_{rcw}	100($\approx$)	85.97(-)	13.20(-)
	GP_{ave}	100($\approx$)	88.01(-)	13.78(-)
	GP_{G-mean}	100($\approx$)	83.80(-)	15.00(-)
	GP_{amse}	100($\approx$)	92.40(-)	8.55(-)
	GP_{corr}	100($\approx$)	90.02(-)	14.20(-)
	GP_{dist}	100($\approx$)	95.13(-)	6.90(-)
*	GP_{aucw}	100($\approx$)	90.76(-)	10.11(-)
Yeoh-2002-v1(IR =5)	Bi-COS	100	98.94	2.32
	CS-GP	100($\approx$)	97.02(-)	4.35(-)
	GP_{rcw}	100($\approx$)	89.03(-)	10.01(-)
	GP_{ave}	100($\approx$)	83.20(-)	12.09(-)
	GP_{G-mean}	95.29(-)	65.99(-)	16.50(-)
	GP_{amse}	91.89(-)	63.17(-)	12.59(-)
	GP_{corr}	100($\approx$)	92.72(-)	8.10(-)
	GP_{dist}	100($\approx$)	90.88(-)	8.67(-)
*	GP_{aucw}	100($\approx$)	**99.12(+)**	**2.30(+)**
New-thyroid1 (IR =5)	Bi-COS	100	99.51	2.06
	CS-GP	100($\approx$)	96.29(-)	4.60(-)
	GP_{rcw}	100($\approx$)	96.10(-)	4.31(-)
	GP_{ave}	100($\approx$)	97.79(-)	3.00(-)
	GP_{G-mean}	100($\approx$)	97.93(-)	3.42(-)
	GP_{amse}	100($\approx$)	96.04(-)	4.56(-)
	GP_{corr}	100($\approx$)	96.22(-)	6.50(-)
	GP_{dist}	100($\approx$)	98.69(-)	3.77(-)
*	GP_{aucw}	100($\approx$)	**99.65(+)**	**2.05(+)**
DLBCL (IR =3)	Bi-COS	100	**86.50**	**5.10**
	CS-GP	94.50(-)	80.15(-)	9.24(-)
	GP_{rcw}	97.10(-)	79.15(-)	13.02(-)
	GP_{ave}	98.20(-)	75.03(-)	15.71(-)
	GP_{G-mean}	100($\approx$)	76.91(-)	15.82(-)
	GP_{amse}	100($\approx$)	76.88(-)	13.53(-)
	GP_{corr}	98.20(-)	80.76(-)	11.87(-)
	GP_{dist}	99.10(-)	83.98(-)	10.03(-)
	GP_{aucw}	100($\approx$)	85.40(-)	10.95(-)

Dataset	Method	Best	Median	Std
Ecoli1 (IR =3)	Bi-COS	100	**90.01**	**3.17**
	CS-GP	95.10(-)	83.40(-)	8.30(-)
	GP_{rcw}	97.01(-)	82.50(-)	11.12(-)
	GP_{ave}	99.00(-)	82.80(-)	9.79(-)
	GP_{G-mean}	96.10(-)	83.20(-)	9.11(-)
	GP_{amse}	96.10(-)	82.90(-)	13.21(-)
	GP_{corr}	99.01(-)	80.76(-)	11.87(-)
	GP_{dist}	99.01(-)	84.09(-)	9.09(-)
*	GP_{aucw}	100($\approx$)	82.12(-)	9.50(-)
Leukemia (IR =2)	Bi-COS	100	**89.97**	**5.11**
	CS-GP	95.90(-)	84.80(-)	9.30(-)
	GP_{rcw}	99.20(-)	81.14(-)	14.53(-)
	GP_{ave}	97.92(-)	88.40(-)	7.98(-)
	GP_{G-mean}	100($\approx$)	81.42(-)	15.60(-)
	GP_{amse}	100($\approx$)	81.33(-)	11.95(-)
	GP_{corr}	100($\approx$)	85.90(-)	10.97(-)
	GP_{dist}	97.01(-)	86.03(-)	9.00(-)
*	GP_{aucw}	100($\approx$)	85.98(-)	9.77(-)
Colon (IR =2)	Bi-COS	97.88	**89.51**	**5.20**
	CS-GP	90.18(-)	78.91(-)	7.20(-)
	GP_{rcw}	87.89(-)	73.50(-)	10.51(-)
	GP_{ave}	91.36(-)	75.21(-)	10.40(-)
	GP_{G-mean}	92.50(-)	71.14(-)	13.05(-)
	GP_{amse}	94.99(-)	74.32(-)	10.88(-)
	GP_{corr}	96.03(-)	74.99(-)	10.34(-)
	GP_{dist}	92.50(-)	76.25(-)	9.76(-)
*	GP_{aucw}	91.36(-)	78.50(-)	7.43(-)
Glass0 (IR =2)	Bi-COS	100	**99.21**	**1.52**
	CS-GP	100($\approx$)	98.67(-)	2.92(-)
	GP_{rcw}	100($\approx$)	88.02(-)	12.99(-)
	GP_{ave}	100($\approx$)	91.71(-)	10.15(-)
	GP_{G-mean}	100($\approx$)	90.00(-)	11.70(-)
	GP_{amse}	100($\approx$)	82.37(-)	11.85(-)
	GP_{corr}	100($\approx$)	95.96(-)	6.40(-)
	GP_{dist}	100($\approx$)	96.41(-)	5.55(-)
*	GP_{aucw}	100($\approx$)	97.90(-)	3.53(-)
Iris0 (IR =2)	Bi-COS	100	**98.98**	**2.79**
	CS-GP	100($\approx$)	98.00(-)	3.00(-)
	GP_{rcw}	100($\approx$)	95.55(-)	5.51(-)
	GP_{ave}	100($\approx$)	93.97(-)	8.00(-)
	GP_{G-mean}	100($\approx$)	91.90(-)	8.30(-)
	GP_{amse}	100($\approx$)	89.99(-)	7.20(-)
	GP_{corr}	100($\approx$)	94.01(-)	7.11(-)
	GP_{dist}	100($\approx$)	95.90(-)	3.55(-)
	GP_{aucw}	100($\approx$)	93.91(-)	5.01(-)

5 CONCLUSION AND FUTURE WORKS

The aim of this paper was to propose a bi-level modeling for the cost-sensitive classification tree induction problem while performing classifiers construction at the upper level and cost learning at the lower level. Indeed, cost values are automatically learned and the best cost information is determined from a population of generated cost values. Compared to recent and pertinent works, the proposed Bi-COS is able to achieve better classification performance while ensuring a precise and fair evaluation.

Several perspectives could be extended from this work. First, we attempt to investigate the performance of Bi-COS for the multi-class classification case. Second, it would be interesting to apply other types of cost information such as the class-independent cost in which a different misclassification cost is assigned for each instance.

ACKNOWLEDGMENTS

Carlos A. Coello Coello gratefully acknowledges support from CONACyT grant no. 2016-01-1920 (Investigación en Fronteras de la Ciencia 2016).

REFERENCES

[1] Radhia Azzouz, Slim Bechikh, and Lamjed Ben Said. 2014. A multiple reference point-based evolutionary algorithm for dynamic multi-objective optimization with undetectable changes. In *2014 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 3168–3175.

[2] Slim Bechikh, Lamjed Ben Said, and Khaled Ghédira. 2011. Negotiating decision makers' reference points for group preference-based Evolutionary Multi-objective Optimization. In *2011 11th International Conference on Hybrid Intelligent Systems (HIS)*. IEEE, 377–382.

[3] Urvesh Bhowan, Mark Johnston, and Mengjie Zhang. 2011. Developing new fitness functions in genetic programming for classification with unbalanced data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 42, 2 (2011), 406–421.

[4] Guo Haixiang, Li Yijing, Jennifer Shang, Gu Mingyun, Huang Yuanyue, and Gong Bing. 2017. Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications* 73 (2017), 220–239.

[5] Jin 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.

[6] Wenbin Pei, Bing Xue, Lin Shang, and Mengjie Zhang. 2021. Genetic programming for development of cost-sensitive classifiers for binary high-dimensional unbalanced classification. *Applied Soft Computing* 101 (2021), 106989.

[7] Rihab Said, Slim Bechikh, Ali Louati, Abdulaziz Aldaej, and Lamjed Ben Said. 2020. Solving combinatorial multi-objective bi-level optimization problems using multiple populations and migration schemes. *IEEE Access* 8 (2020), 141674–141695.

[8] Rihab Said, Maha Elarbi, Slim Bechikh, and Lamjed Ben Said. 2021. Solving combinatorial bi-level optimization problems using multiple populations and migration schemes. *Operational Research* (2021), 1–39.

[9] Binh Tran, Bing Xue, and Mengjie Zhang. 2016. Genetic programming for feature construction and selection in classification on high-dimensional data. *Memetic Computing* 8, 1 (2016), 3–15.

[10] Lian Yan, Robert H Dodier, Michael Mozer, and Richard H Wolniewicz. 2003. Optimizing classifier performance via an approximation to the Wilcoxon-Mann-Whitney statistic. In *Proceedings of the 20th international conference on machine learning (icml-03)*.

848–855.