European Journal of Operational Research 284 (2020) 273–284

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Decision Support

Profit-based churn prediction based on Minimax Probability Machines

Sebastián Maldonado[a,d], Julio López[b], Carla Vairetti[c,d,*]

[a] Department of Management Control and Information Systems, School of Economics and Business, University of Chile, Santiago, Chile
[b] Facultad de Ingeniería y Ciencias, Universidad Diego Portales, Ejército 441, Santiago, Chile
[c] Facultad de Ingeniería y Ciencias Aplicadas, Universidad de los Andes Monseñor Álvaro del Portillo 12455, Las Condes, Santiago, Chile
[d] Instituto Sistemas Complejos de Ingeniería (ISCI), Chile

ARTICLE INFO

Article history:
Received 12 January 2019
Accepted 3 December 2019
Available online 6 December 2019

Keywords:
Analytics
Churn prediction
Support vector machines
Minimax probability machine
Robust optimization

ABSTRACT

In this paper, we propose three novel profit-driven strategies for churn prediction. Our proposals extend the ideas of the Minimax Probability Machine, a robust optimization approach for binary classification that maximizes sensitivity and specificity using a probabilistic setting. We adapt this method and other variants to maximize the profit of a retention campaign in the objective function, unlike most profit-based strategies that use profit metrics to choose between classifiers, and/or to define the optimal classification threshold given a probabilistic output. A first approach is developed as a learning machine that does not include a regularization term, and subsequently extended by including the LASSO and Tikhonov regularizers. Experiments on well-known churn prediction datasets show that our proposal leads to the largest profit in comparison with other binary classification techniques.

1. Introduction

Churn prediction is a well-known business analytics task, that is aimed at detecting customers that are likely to leave a company voluntarily (Baesens, 2014; Baumann, Lessmann, Coussement, & Bock, 2015). Once a company has identified potential churners, a customized retention campaign can be designed for enhancing customer loyalty. Loyalty is extremely beneficial since engaged customers generate more revenue than other customers, and it reduces operational costs and the misspending of money caused by inefficient marketing efforts (Farquad, Ravi, & Raju, 2014; Fleming & Asplund, 2007).

Customer retention has often been approached by researchers and practitioners via machine learning methods for binary classification, taking profit measures for assessing which classifier achieves the best predictive performance into account (Verbeke, Dejaeger, Martens, Hur, & Baesens, 2012). Unlike traditional evaluation metrics, such as accuracy or AUC, profit metrics focus on the actual benefits and costs of implementing the solution obtained by the classifiers, yielding better decision-making (Hand, 2009; Verbraken, Verbeke, & Baesens, 2012). In churn prediction, profit metrics estimate the average profit of a retention campaign (Baumann et al., 2015; Maldonado, Flores, Verbraken, Baesens, & Weber, 2015; Verbeke et al., 2012).

The study of robust optimization approaches have been reported widely in the machine learning literature. Robustness is an important virtue in classification since it guarantees adequate predictive performance in changing environments (Huang, Yang, King, Lyu, & Chan, 2004; López, Maldonado, & Carrasco, 2018). The use of robust optimization methods in classification usually translates to superior predictive performance (Huang et al., 2004; López et al., 2018).

Notice that predicting churners in sectors such as telecommunications or finance is a task that is constantly evolving for several reasons. First, technology evolves and the competition among the different actors varies according to the emerging technologies. Additionally, retention campaigns also change over time, based on the new technologies and the competition. Finally, the impact of a retention campaign on the customers is time-dependent (Verbraken, Baesens, & Bravo, 2017). Therefore, a robust framework for binary classification can be very useful for improving performance in such changing environments.

In this work, we extend the Minimax Probability Machine (MPM) method (Lanckriet, Ghaoui, Bhattacharyya, & Jordan, 2003), the Minimum Error Minimax Probability Machine (MEMPM) method (Huang et al., 2004), and their regularized versions (López et al., 2018) to the domain of profit measures. These techniques are adapted to maximize the profit of a retention campaign while achieving adequate classification performance. Three variants are proposed: the direct extension to the MEMPM strategy, called

[*] Corresponding author at: Facultad de Ingeniería y Ciencias Aplicadas, Universidad de los Andes Monseñor Álvaro del Portillo 12455, Las Condes, Santiago, Chile.

E-mail addresses: sebastianm@fen.uchile.cl (S. Maldonado), julio.lopez@udp.cl (J. López), cvairetti@uandes.cl (C. Vairetti).

https://doi.org/10.1016/j.ejor.2019.12.007
0377-2217/© 2019 Elsevier B.V. All rights reserved.

Profit-based MEMPM (ProfMEMPM), which does not include a regularization term; and the profit-driven, ℓ_p-regularized MPM and MEMPM extensions (ℓ_p-ProfMPM and ℓ_p-ProfMEMPM, respectively, with $p = \{1, 2\}$ being the LASSO - Least Absolute Shrinkage and Selection Operator- and the Tikhonov regularization). Our experiments demonstrate that the proposed methods perform better on average when compared with alternative classifiers in terms of expected profit.

To the best of our knowledge, reports of profit-driven classifiers that optimize profit are scarce in the machine learning literature, and limited to extensions of logistic regression (ProfLogit, Stripling, vanden Broucke, Antonio, Baesens, & Snoeck, 2018) or decision trees (ProfTree, Höppner, Stripling, Baesens, vanden Broucke, & Verdonck, 2018). Therefore, our proposals represent a valuable contribution to the state of the art on business analytics. Furthermore, the proposed strategies are novel and elegant machine learning methods based on robust optimization, which are solved using adhoc optimization strategies for fractional programming (Schaible, 1981) and second-order cone programming (Alizadeh & Goldfarb, 2003). Given this fact, they represent relevant contributions not only for the practice of analytics and Customer Relationship Management (CRM), but also novel developments in the machine learning and optimization domain.

Our proposals present a robust strategy whose goal is maximizing a profit function while guaranteeing an adequate classification, even for the worst data distribution of the churners and non-churners. This robust framework has been shown to be very effective for enhancing predictive performance in a wide variety of domains, including pattern recognition (Ma, Yang, Wen, & Sun, 2020) and credit scoring (López & Maldonado, 2019).

The remainder of the paper is organized as follows: Section 2 describes profit metrics in the context of customer retention. Section 3 presents the robust machine learning formulations that are relevant for this study. The proposed robust classification approaches for profit-driven churn prediction are presented in Section 4. Section 5 provides experimental results obtained by using real-world churn datasets. Finally, the main conclusions are provided in Section 6, which also addresses future developments.

2. Profit-driven framework for robust churn prediction

In this section, we present the profit-based frameworks proposed in Verbeke et al. (2012) and Verbraken et al. (2012), and introduce the notation for our robust framework, as well as relevant literature related to this task.

Customer attrition can be predicted either with single period future predictions, or with time-dependent strategies (Blattberg, Kim, & Neslin, 2008). The first approach is the most common one encountered in the scientific literature, and aims at predicting whether a customer will leave in the next period (Blattberg et al., 2008). Binary classification techniques, such as logistic regression (Burez & Van den Poel, 2009; Neslin, Gupta, Kamakura, Lu, & Mason, 2006), k-nearest neighbors (Datta, Masand, Mani, & Li, 2000), decision trees (Wei & Chiu, 2002), and other machine learning techniques such as random forest, artificial neural networks and support sector machines (Chen, Fan, & Sun, 2012; Farquad et al., 2014; Verbeke, Martens, & Baesens, 2014) can be used. We refer the reader to the review presented in Verbeke et al. (Verbeke, Martens, Mues, & Baesens, 2011).

The proposed profit-based framework is discussed next. A trained classifier $\mathcal{C}$ would produce a probabilistic output $s \in [0, 1]$, which can be interpreted as the probability of attrition for a new customer $\mathbf{x}$. The decision rule follows: if $s \leq t$ then $\mathbf{x}$ is classified as non-churner ($y = -1$), otherwise $\mathbf{x}$ is classified as churner ($y = 1$).

The dynamics of a retention campaign can be described as follows: A fraction of the customers with the highest risk of churning is contacted, incurring an individual cost of f. An incentive is offered to this group, which leads to a monetary cost d only if it is accepted. It is assumed that a fraction γ of the would-be churners accept this incentive and stay with the company, together with all false would-be churners since they never had the intention to churn (Verbraken et al., 2012). The benefit of retaining a would-be churner is its Customer Lifetime Value (CLV), which is usually significantly larger than d and f. There is no benefit for retaining false would-be churners, nor with the fraction $(1-\gamma)$ that effectively leaves despite the incentive, nor with the fraction of customers which is not contacted (Verbraken et al., 2012).

Formally, the average profit of a classifier follows:

$$P_{\mathcal{C}}(t;\gamma,CLV,\delta,\phi) = CLV(\gamma(1-\delta)-\phi)\pi_1 F_1(t) - CLV(\delta+\phi)\pi_{-1}F_{-1}(t), \quad (1)$$

where $\delta = \frac{d}{CLV}$ and $\phi = \frac{f}{CLV}$. The parameters π_{-1} and π_1 are the prior probabilities of a given customer to belong to class -1 (non-churner) or 1 (churner), respectively, while $F_{-1}(t)$ and $F_1(t)$ are the cumulative density functions for non-churners and churners for a given threshold t, respectively.

The *Maximum Profit Criterion* (MPC) (Verbeke et al., 2012) and the *Expected Maximum Profit Criterion* (EMPC) (Verbraken et al., 2012) measures are relevant for our study. The first metric assumes that all information in the average profit equation (cf. Eq. (1)) is known and deterministic, in contrast to the EMPC, which assumes that γ, the probability of a would-be churner accepting the incentive, is a random variable that follows a Beta distribution. The MPC measure selects the threshold that maximizes the average profit:

$$\text{MPC} = \max_t P_{\mathcal{C}}(t;\gamma,CLV,\delta,\phi). \quad (2)$$

Subsequently, the fraction of the customers to be contacted η given the optimal threshold t^* is given by:

$$\eta = \pi_{-1}F_{-1}(t^*) + \pi_1 F_1(t^*). \quad (3)$$

Alternatively, the EMPC measure is obtained as follows:

$$\text{EMPC} = \int_\gamma P_{\mathcal{C}}(t^*(\gamma);\gamma,CLV,\delta,\phi)\cdot h(\gamma)d\gamma, \quad (4)$$

with $t^*(\gamma)$ being the optimal threshold and $h(\gamma)$ the probability density function for γ. We used the values proposed by Verbraken et al. (2012) for the parameters α and β related to the Beta distribution. Finally, the fraction η of the targeted customers is given by:

$$\eta = \int_\gamma [\pi_{-1}F_{-1}(T(\gamma)) + \pi_1 F_1(T(\gamma))]\cdot h(\gamma)d\gamma. \quad (5)$$

The MPC and EMPC metrics were originally developed for selecting the best performing classification strategies among various approaches (Verbeke et al., 2012; Verbraken, Bravo, Weber, & Baesens, 2014; Verbraken et al., 2012). However, some studies go beyond that strategy. For example, Maldonado et al. (2015) proposed using these metrics for model and feature selection with SVM classifiers. The authors proposed a backward elimination strategy that removes those features whose removal maximizes the profit of the classifier. Alternatively, Höppner et al. (2018) proposed a profit-based extension for decision trees, in which the branching process is guided by profit measures using genetic algorithms (Sivanandam & Deepa, 2006). Along the same line, Stripling et al. (2018) developed ProfLogit, which follows the same reasoning behind Höppner et al. (2018) and maximizes the profit of a logistic regression model instead of a maximum likelihood function.

Inspired by the studies of Stripling et al. (2018) and Höppner et al. (2018), we aim at maximizing a profit measure during the model training. In contrast to these other approaches that use genetic algorithms, we propose three robust optimization approaches which are solved to optimality via ad-hoc techniques.

The proposed methods are inspired by the MPM and MEMPM algorithms (Huang et al., 2004; Lanckriet et al., 2003), which considers a robust framework for maximizing the two class accuracies. Our assumption is that business analytics tasks such as churn prediction can be benefited from robust optimization. These tasks usually present small changes in the distribution due to evolving retention campaign policies and the intrinsic noise that affect customer data.

3. Robust classification via Minimax Probability Machines

The methods that are relevant for our proposal, namely the Minimax Probability Machine (Lanckriet et al., 2003) and the Minimum Error Minimax Probability Machine (Huang et al., 2004), are formalized in this section.

3.1. Minimax Probability Machine (MPM)

The Minimax Probability Machine (MPM) is a robust optimization approach that minimizes the worst-case probability of misclassification (Lanckriet et al., 2003). This model assumes that the samples of the two classes are generated by random variables $\mathbf{X}_1$ and $\mathbf{X}_2$, both having known mean and covariance matrices ($\boldsymbol{\mu}_k$, Σ_k) for $k = 1, 2$. Based on this assumption, a separating hyperplane of the form $\mathbf{w}^\top\mathbf{x} + b = 0$, with $\mathbf{w} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ and $b \in \mathbb{R}$, is constructed in such a way that the two classes must be classified correctly with maximal probability with respect to all distributions (Lanckriet et al., 2003).

The MPM formulation can be written as a chance-constrained problem, as follows:

$$\begin{array}{ll} \max\limits_{\mathbf{w},b,\alpha} & \alpha \\ \text{s.t.} & \inf\limits_{\mathbf{X}_1\sim(\boldsymbol{\mu}_1,\Sigma_1)} \Pr\{\mathbf{w}^\top\mathbf{X}_1 + b \geq 0\} \geq \alpha, \\ & \inf\limits_{\mathbf{X}_2\sim(\boldsymbol{\mu}_2,\Sigma_2)} \Pr\{\mathbf{w}^\top\mathbf{X}_2 + b \leq 0\} \geq \alpha, \end{array} \tag{6}$$

where $\mathbf{X} \sim (\boldsymbol{\mu}, \Sigma)$ represents the family of distributions which have a common mean and covariance matrix, and $\alpha \in (0, 1)$ is the worst-case class accuracy, i.e., the lower bound for the sensitivity and specificity.

The chance-constrained optimization model presented in Formulation (6) can be cast into a second-order cone programming (SOCP) problem (Alizadeh & Goldfarb, 2003) by using the Chebyshev inequality (Lanckriet et al., 2003, Lemma 1), which is presented next.

Lemma 3.1 (Chebyshev inequality)**.** *Let* $\mathbf{x}$ *be a n-dimensional random variable with mean* $\boldsymbol{\mu}$ *and covariance matrix* Σ*. Given a vector* $\mathbf{a} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$*, and scalars* $b \in \mathbb{R}$ *and* $\alpha \in (0, 1)$ *such that* $\mathbf{a}^\top\boldsymbol{\mu} + b \geq 0$*, the condition*

$$\inf_{\mathbf{x}\sim(\boldsymbol{\mu},\Sigma)} \Pr\{\mathbf{a}^\top\mathbf{x} + b \geq 0\} \geq \alpha$$

holds if and only if $\mathbf{a}^\top\boldsymbol{\mu} + b \geq \kappa(\alpha)\sqrt{\mathbf{a}^\top\Sigma\mathbf{a}}$[1]*, where* $\kappa(\alpha) = \sqrt{\frac{\alpha}{1-\alpha}}$*.*

Then, the use of Lemma 3.1 allows us to reformulate the MPM model as the following SOCP problem (see Lanckriet et al., 2003, Theorem 2, for details):

$$\min_{\mathbf{w}} \left\{ \sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}} + \sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}} : \ \mathbf{w}^\top(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) = 1 \right\}. \tag{7}$$

This problem can be reduced to a linear SOCP problem, which can be solved efficiently via interior point algorithms (Alizadeh & Goldfarb, 2003).

Remark 1. In practice, the mean and the covariance matrix are not available. Therefore, their respective empirical estimations are used instead. Formally, let us denote by m_1 (resp. m_2) the cardinality of the positive (resp. negative) class, by $A_1 \in \mathbb{R}^{m_1\times n}$ (resp. $A_2 \in \mathbb{R}^{m_2\times n}$) the data matrix related to the positive (resp. negative) class. The empirical estimates of the mean and covariance are given by

$$\hat{\mu}_k = \frac{1}{m_k}A_k^\top\mathbf{e}_{m_k}, \quad \hat{\Sigma}_k = \frac{1}{m_k}A_k^\top\left(I_{m_k} - \frac{1}{m_k}\mathbf{e}_{m_k}\mathbf{e}_{m_k}^\top\right)A_k,$$

where $\mathbf{e}_k$ denotes a vector de ones of dimension m_k and I_{m_k} the identity matrix of size m_k.

3.2. Minimum Error Minimax Probability Machine (MEMPM)

The main disadvantage of MPM is that it assumes that the two classes are equally important for decision-making (Huang et al., 2004), which is usually not the case in business analytics tasks (Baesens, 2014; Baumann et al., 2015; Hand, 2009; Maldonado et al., 2015; Verbraken et al., 2014). In order to overcome this issue, the Minimum Error Minimax Probability Machine (MEMPM) uses two bounds α_1 and α_2 for the worst-case accuracies instead of a single one (Huang et al., 2004). Let us define $\theta \in (0, 1)$ and $1 - \theta$ as the prior probabilities of classes 1 and 2 respectively. After applying the Chebyshev inequality, the MEMPM formulation follows:

$$\begin{array}{ll} \max\limits_{\mathbf{w},b,\alpha_1,\alpha_2} & \theta\alpha_1 + (1-\theta)\alpha_2 \\ \text{s.t.} & \mathbf{w}^\top\boldsymbol{\mu}_1 + b \geq \kappa(\alpha_1)\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}}, \\ & -(\mathbf{w}^\top\boldsymbol{\mu}_2 + b) \geq \kappa(\alpha_2)\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}, \end{array} \tag{8}$$

where $\kappa(\alpha_k) = \sqrt{\frac{\alpha_k}{1-\alpha_k}}$, for $k = 1, 2$. This formulation is a nonlinear SOCP problem, since it contains a linear objective function with two nonlinear SOC constraints[2].

Using the same reasoning as that behind MPM, Formulation (8) can be cast into a fractional programming problem Schaible (1981). In order to solve this formulation, Huang et al. (2004) proposed an iterative algorithm, which stems from the Quadratic Interpolation (QI) method (Bertsekas, 1999). First, variable α_1 is set to a fixed value. Then, the QI method is used for updating variables α_2 and $\mathbf{w}$ iteratively. Finally, α_1 can be obtained by a relation that holds for α_1, α_2, and $\mathbf{w}$ when this strategy is used (Huang et al., 2004). The optimization strategy is formalized and extended to our proposal in Section 4.

Notice that both the MPM and MEMPM methods can be extended as kernel methods (see Lanckriet et al., 2003 and Huang et al., 2004, respectively). Furthermore, some relevant extensions have been proposed. For example, the Biased Minimax Probability machine (BMPM) (Huang, H.Yang, King, & Lyu, 2006) biases the predictions towards one class, assuming a fixed value β_0 for the sensitivity. Another relevant extension is the regularized MEMPM model proposed in Maldonado, Carrasco, and López (2019), which has the following form:

$$\begin{array}{ll} \min\limits_{\mathbf{w},b,\kappa_1,\kappa_2} & \frac{1}{2}\|\mathbf{w}\|^2 + C_1\frac{1}{\kappa_1^2+1} + C_2\frac{1}{\kappa_2^2+1} \\ \text{s.t.} & \mathbf{w}^\top\boldsymbol{\mu}_1 + b \geq \kappa_1\|S_1^\top\mathbf{w}\|, \ \kappa_1 \geq 0, \\ & -\mathbf{w}^\top\boldsymbol{\mu}_2 - b \geq \kappa_2\|S_2^\top\mathbf{w}\|, \ \kappa_2 \geq 0, \\ & \mathbf{w}^\top\boldsymbol{\mu}_1 + b \geq 1, \\ & -\mathbf{w}^\top\boldsymbol{\mu}_2 - b \geq 1, \end{array} \tag{9}$$

[1] For a given $\alpha \in (0, 1)$, this expression is called linear second-order cone (SOC) constraint (Alizadeh & Goldfarb, 2003). An SOC constraint on the variable $\mathbf{x} \in \mathbb{R}^n$ has the form $\|D\mathbf{x} + \mathbf{b}\|_2 \leq \mathbf{c}^\top\mathbf{x} + d$, where $d \in \mathbb{R}$, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$, $D \in \mathbb{R}^{m\times n}$ are given.

[2] A nonlinear SOC constraint (Canelas, Carrasco, & López, 2019) has the form $\|\bar{g}(x)\|_2 \leq g_1(x)$, where $g : \mathbb{R}^n \to \mathbb{R}^m$ is a function defined by $g(x) = (g_1(x), \bar{g}(x))$, with $g_1 : \mathbb{R}^n \to \mathbb{R}$ and $\bar{g} : \mathbb{R}^n \to \mathbb{R}^{m-1}$.

where $\Sigma_k = S_k S_k^\top$, for $k = 1, 2$, $C_1, C_2 > 0$. Formulation (9) is a nonlinear smooth SOCP problem since it contains a nonconvex smooth objective function with two nonlinear SOC constraints and four linear constraints.

Another interesting approach, called Structural Minimax Probability Machine (SMPM), improves MEMPM by replacing the prior probabilities with two mixture models (Gu, Sun, & Sheng, 2017). Finally, the DR-MEMPM method (Song, Gong, Zhang, Huang, & Huang, 2017) performs embedded dimensionality reduction for multi-class learning.

4. Proposed profit-based formulations for robust churn prediction

We propose three novel robust classification approaches for profit-driven churn prediction in this section. Our proposals extend the MPM and MEMPM models (cf. formulations (6) and (8)) to the profit-based framework. The MEMPM method is a natural choice for base model of our proposal since it maximizes the two class accuracies independently in the objective function, given a robust setting that translates into two conic constraints. This model can be extended by replacing the weighted sum of the two class accuracies with a profit function. This extension, called ProfMEMPM, is presented in Section 4.1.

One issue with the MPM and MEMPM models is that they do not include any regularization term, which can boost predictive performance by reducing the complexity of robust classifiers (Saketha Nath & Bhattacharyya, 2007). Therefore, we extend our framework to regularized learning machines by considering the ℓ_1 and ℓ_2 norms. The profit-based extensions of the regularized MPM and MEMPM methods are proposed in Section 4.2 and 4.3, respectively. We refer to this models as ℓ_p-ProfMPM and ℓ_p-ProfMEMPM, respectively.

This study not only contributes with profit-driven extensions of existing robust models. The ℓ_p-ProfMPM and ℓ_p-ProfMEMPM are novel machine learning approaches, and therefore they require ad-hoc optimization strategies. Section 4.4 presents the optimization strategies used for solving the three proposals. Finally, Section 4.5 discusses the relationship between the proposed methods and existing profit-driven techniques and robust machine learning models.

4.1. Profit-based Minimum Error Minimax Probability Machine

Let us assume that X_{-1} (X_1) represents the random variable related to the non-churners (churners). It holds that $\theta = \pi_{-1}$, $1-\theta = \pi_1$, $\alpha_1 = 1 - F_{-1}(t)$, and $\alpha_2 = F_1(t)$. Based on these relations, the objective function $\theta\alpha_1 + (1-\theta)\alpha_2$ related to the MEMPM model can be replaced by the following profit measure:

$$Profit(\alpha_1, \alpha_2) = -c_{-1}\theta(1-\alpha_1) + b_1(1-\theta)\alpha_2, \tag{10}$$

where $b_1 = CLV(\gamma(1-\delta) - \phi)$ and $c_{-1} = CLV(\delta + \phi)$ (see Eq. (1)). Notice that maximizing Eq. (10) is equivalent to:

$$Profit(\alpha_1, \alpha_2) = c_{-1}\theta\alpha_1 + b_1(1-\theta)\alpha_2, \tag{11}$$

since the first term $(-c_{-1}\theta)$ is constant with respect to the decision variables. Therefore, the larger the values for $0 \le \alpha_1, \alpha_2 \le 1$, the larger the profit. Then, the proposed profit-based model follows:

$$\begin{aligned} \max_{\mathbf{w}, b, \alpha_1, \alpha_2} \quad & c_{-1}\theta\alpha_1 + b_1(1-\theta)\alpha_2 \\ \text{s.t.} \quad & \mathbf{w}^\top\boldsymbol{\mu}_1 + b \ge \kappa(\alpha_1)\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}}, \\ & -(\mathbf{w}^\top\boldsymbol{\mu}_2 + b) \ge \kappa(\alpha_2)\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}, \end{aligned} \tag{12}$$

where $\kappa(\alpha_k) = \sqrt{\frac{\alpha_k}{1-\alpha_k}}$, for $k = 1, 2$. We refer to this proposal as the Profit-Based Minimum Error Minimax Probability Machine (ProfMEMPM).

Following the guidelines in Huang et al. (2004), Formulation (12) can be rewritten as a fractional programming problem Schaible (1981) in order to ease the optimization process. First, variable b can be removed by combining the two nonlinear constraints in Eq. (12), leading to the following problem:

$$\begin{aligned} \max_{\alpha_1, \alpha_2, \mathbf{w} \neq \mathbf{0}} \quad & c_{-1}\theta\alpha_1 + b_1(1-\theta)\alpha_2 \\ \text{s.t.} \quad & \mathbf{w}^\top(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \ge \kappa(\alpha_1)\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}} + \kappa(\alpha_2)\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}. \end{aligned} \tag{13}$$

Let us define $\beta = \alpha_2$. Since $\alpha_1 = \frac{\kappa^2(\alpha_1)}{\kappa^2(\alpha_1)+1}$ and the maximum value of $c_{-1}\theta\alpha_1 + b_1(1-\theta)\beta$ under the constraint given in (13) is achieved when the right hand side is equal to $\mathbf{w}^\top(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$, Formulation (13) can be rewritten as follows:

$$\max_{\beta, \mathbf{w} \neq \mathbf{0}} \quad \{f_{profit}(\mathbf{w}, \beta) : \mathbf{w}^\top(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) = 1\}, \tag{14}$$

where

$$f_{profit}(\mathbf{w}, \beta) = \frac{c_{-1}\theta\gamma^2(\mathbf{w}, \beta)}{\gamma^2(\mathbf{w}, \beta) + 1} + b_1(1-\theta)\beta, \tag{15}$$

with

$$\gamma(\mathbf{w}, \beta) = \frac{1 - \kappa(\beta)\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}}{\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}}}. \tag{16}$$

In order to solve Problem (14), we propose using the Quadratic interpolation (QI) method Bertsekas (1999), which solves this formulation for a fixed β iteratively. The inner problem that results by fixing β is a relatively simple fractional problem which can be solved efficiently using gradient projection strategies, for example. The optimization scheme is discussed in Section 4.4.

4.2. Regularized profit-based Minimax Probability Machine

A novel formulation can be derived by incorporating an ℓ_p-norm regularizer for the weight vector $\mathbf{w}$ in the ProfMEMPM formulation. This inclusion, however, leads to a model that is complex to solve to optimality. In order to simplify the optimization process, we first impose that $\alpha_1 = \alpha_2 = \alpha$, extending the MPM model to profit-driven classification (the ℓ_p-ProfMPM approach). In the next section, we derive the case when $\alpha_1 \neq \alpha_2$, which results in the ℓ_p-ProfMEMPM method.

The inclusion of the ℓ_p-norm in the MPM model leads to the following problem:

$$\begin{aligned} \max_{\alpha, \mathbf{w} \neq \mathbf{0}, b} \quad & Profit(\alpha, \alpha) - \lambda\rho_p(\mathbf{w}) \\ \text{s.t.} \quad & \mathbf{w}^\top\boldsymbol{\mu}_1 + b \ge \kappa(\alpha)\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}}, \\ & -(\mathbf{w}^\top\boldsymbol{\mu}_2 + b) \ge \kappa(\alpha)\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}, \end{aligned} \tag{17}$$

where $\rho_p(\mathbf{w})$ can be either the Tikhonov ($\rho_2(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2$) or LASSO ($\rho_1(\mathbf{w}) = \|\mathbf{w}\|_1 = \sum_{i=1}^n |w_i|$) regularization, $\lambda > 0$ is a parameter designed to balance the profit and regularization, and $\kappa(\alpha) = \sqrt{\frac{\alpha}{1-\alpha}}$. The profit function becomes $Profit(\alpha, \alpha) = \Theta\alpha - c_1(1-\theta)$, with $\Theta = b_{-1}\theta + c_1(1-\theta)$.

Let us define $\beta = \kappa(\alpha)$. Since $\alpha = \frac{\kappa^2(\alpha)}{\kappa^2(\alpha)+1}$, Formulation (17) can be rewritten as

$$\begin{aligned} \max_{\beta, \mathbf{w} \neq \mathbf{0}, b} \quad & f_{profit}(\mathbf{w}, \beta) \\ \text{s.t.} \quad & \mathbf{w}^\top\boldsymbol{\mu}_1 + b \ge \beta\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}}, \\ & -(\mathbf{w}^\top\boldsymbol{\mu}_2 + b) \ge \beta\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}, \end{aligned} \tag{18}$$

where the (nonconcave) objective function is given by

$$f_{profit}(\mathbf{w}, \beta) = \Theta\frac{\beta^2}{\beta^2 + 1} - \lambda\rho_p(\mathbf{w}). \tag{19}$$

Similar to the ProfMEMPM method, we propose solving Problem (18) with the QI algorithm. However, the inner problem that results from fixing β is a very different one when compared to ProfMEMPM. A linear and a quadratic SOCP problem are derived from Formulation (18) when $p=1$ and $p=2$, respectively, and beta fixed. The optimization process is formalized in Section 4.4.

4.3. Regularized profit-based Minimum Error Minimax Probability Machine

The ℓ_p-ProfMEMPM formulation can be derived by introducing the ℓ_p-regularizer $\rho_p(\mathbf{w})$ in the ProfMEMPM model, and a trade-off parameter λ, as follows:

$$\begin{array}{ll} \max\limits_{\alpha_1,\alpha_2,\mathbf{w}\neq\mathbf{0},b} & Profit(\alpha_1,\alpha_2)-\lambda\rho_p(\mathbf{w}) \\ \text{s.t.} & \mathbf{w}^\top\boldsymbol{\mu}_1+b\geq\kappa(\alpha_1)\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}}, \\ & -(\mathbf{w}^\top\boldsymbol{\mu}_2+b)\geq\kappa(\alpha_2)\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}, \end{array} \quad (20)$$

where *Profit*(α_1, α_2) is defined in Eq. (11). Formulation (20) can be rewritten in order to ease the optimization process. Since $\alpha_i=\frac{\kappa^2(\alpha_i)}{\kappa^2(\alpha_i)+1}$, for $i=1,2,$ the objective function in Eq. (20) can be rewritten as

$$\begin{aligned} f(\alpha_1,\alpha_2,\mathbf{w},b) &= c_{-1}\theta\frac{\kappa^2(\alpha_1)}{\kappa^2(\alpha_1)+1} \\ &\quad +b_1(1-\theta)\frac{\kappa^2(\alpha_2)}{\kappa^2(\alpha_2)+1}-\lambda\rho_p(\mathbf{w}) \\ &= c_{-1}\theta\left(1-\frac{1}{\kappa^2(\alpha_1)+1}\right) \\ &\quad +b_1(1-\theta)\left(1-\frac{1}{\kappa^2(\alpha_2)+1}\right)-\lambda\rho_p(\mathbf{w}). \end{aligned}$$

Let us denote by $\beta_i=\kappa^2(\alpha_i),$ for $i=1,2.$ Then, Problem (20) can be rewritten as

$$\begin{array}{ll} \min\limits_{\beta_1,\beta_2,\mathbf{w}\neq\mathbf{0},b} & \lambda\rho_p(\mathbf{w})+\frac{c_{-1}\theta}{\beta_1+1}+\frac{b_1(1-\theta)}{\beta_2+1} \\ \text{s.t.} & \mathbf{w}^\top\boldsymbol{\mu}_1+b\geq\sqrt{\beta_1\mathbf{w}^\top\Sigma_1\mathbf{w}},\ \beta_1>0, \\ & -(\mathbf{w}^\top\boldsymbol{\mu}_2+b)\geq\sqrt{\beta_2\mathbf{w}^\top\Sigma_2\mathbf{w}},\ \beta_2>0, \end{array} \quad (21)$$

which has a convex objective function. Problem (21) adapted further by considering the the arithmetic mean-geometric mean inequality:

$$\sqrt{\beta_i\mathbf{w}^\top\Sigma_i\mathbf{w}}\leq\frac{1}{2}\left(\beta_i t_i+\frac{\mathbf{w}^\top\Sigma_i\mathbf{w}}{t_i}\right),\quad i=1,2, \quad (22)$$

leading to the following minimization problem:

$$\begin{array}{ll} \min\limits_{\substack{\beta_1,\beta_2,\mathbf{w}\neq\mathbf{0}\\ b,t_1,t_2}} & \lambda\rho_p(\mathbf{w})+\frac{c_{-1}\theta}{\beta_1+1}+\frac{b_1(1-\theta)}{\beta_2+1} \\ \text{s.t.} & \frac{1}{2}\left(\beta_1 t_1+\frac{\mathbf{w}^\top\Sigma_1\mathbf{w}}{t_1}\right)\leq\mathbf{w}^\top\boldsymbol{\mu}_1+b,\ \beta_1,t_1>0, \\ & \frac{1}{2}\left(\beta_2 t_2+\frac{\mathbf{w}^\top\Sigma_2\mathbf{w}}{t_2}\right)\leq-(\mathbf{w}^\top\boldsymbol{\mu}_2+b),\ \beta_2,t_2>0. \end{array} \quad (23)$$

It is easy to prove that problems (21) and (23) are equivalents.

As previously mentioned, solving this formulation directly is complex. Hence, a two-step iterative method is proposed in the next section. This strategy is tailored for this particular approach, and differs from the QI algorithm considered by the MEMPM, ProfMEMPM, and ℓ_p-ProfMPM methods.

4.4. Optimization approach for solving the proposed methods

First, we propose an optimization framework based on the QI algorithm used for solving the ProfMEMPM and ℓ_p-ProfMPM methods. If β remains fixed within (0,1), the inner problem solved by the QI algorithm is derived in Remarks 2 and 3 for the ProfMEMPM and ℓ_p-ProfMPM models, respectively.

Remark 2. Formulation (14) becomes the following concave-convex fractional programming problem Schaible (1981) with β fixed:

$$\max_{\mathbf{w}\neq\mathbf{0}}\left\{\frac{1-\kappa(\beta)\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}}{\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}}}:\ \mathbf{w}^\top(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)=1\right\}. \quad (24)$$

Following the MEMPM approach, we propose solving this problem via the Rosen's gradient projection method Bertsekas (1999).

Remark 3. For β fixed within (0,1), Problem (18) becomes:

$$\begin{array}{ll} \min\limits_{\mathbf{w}\neq\mathbf{0},b} & \rho_p(\mathbf{w}) \\ & \mathbf{w}^\top\boldsymbol{\mu}_1+b\geq\beta\sqrt{\mathbf{w}^\top\Sigma_1\mathbf{w}}, \\ & -(\mathbf{w}^\top\boldsymbol{\mu}_2+b)\geq\beta\sqrt{\mathbf{w}^\top\Sigma_2\mathbf{w}}. \end{array} \quad (25)$$

Note that this formulation reduces to a quadratic SOCP problem with two linear SOC constraints when $p=2$ (ℓ_2-ProfMPM), while the ℓ_1-ProfMPM approach ($p=1$) reduces to a non-smooth SOCP problem with two linear SOC constraints because of the inclusion of absolute values. The ℓ_1-ProfMPM formulation can be cast into a linear SOCP problem with two linear SOC and $2n$ linear constraints. Both models can be solved efficiently via interior point algorithms (Alizadeh & Goldfarb, 2003) using, for instance, the SeDuMi toolbox (Sturm, 1999).

As mentioned before, sequential optimization strategy is applied for solving the ProfMEMPM and ℓ_p-ProfMPM methods. The idea of this approach is to set β to a specific value within interval (0,1), and then solve the resulting inner problem (Formulation (24) or (25)) for obtaining $\mathbf{w}$ and the profit function $f_{profit}(\mathbf{w},\beta)$.

In the next step, β is updated via the Quadratic Interpolation (QI) method (Bertsekas, 1999). The idea of the QI method is to find the maximum point by updating a three-point pattern (β_1, β_2, β_3) repeatedly. The new β denoted by $\bar{\beta}^k$ is given by the quadratic interpolation from the three-point pattern. Then, the whole process is repeated until convergence. This process is presented in Algorithm 1.

Remark 4. It follows from (Sun & Yuan, 2006, Theorem 2.4.3) that the sequence $\{\bar{\beta}^k\}$ generated from Algorithm 1 converges to the solution of formulation (14) (resp. formulation (18)). Moreover, this convergence is superlinear.

Once the modified QI algorithm reaches convergence, the variable α_1 of the formulation (13) is computed as

$$\alpha_1^*=\frac{[\gamma(\mathbf{w}^*,\beta^*)]^2}{[\gamma(\mathbf{w}^*,\beta^*)]^2+1}, \quad (26)$$

where $\gamma(\mathbf{w}^*, \beta^*)$ is given by Eq. (16), and $(\mathbf{w}^*, \beta^*)$ is the solution tuple obtained by the QI algorithm. Similarly, the variable α of the formulation (17) is computed as

$$\alpha^*=\frac{{\beta^*}^2}{{\beta^*}^2+1}. \quad (27)$$

Finally, the optimal intercept b^* associated to the ProfMEMPM (Formulation (13)) and ℓ_p-ProfMPM (Formulation (17)) methods is given by

$$\begin{aligned} b^* &= -{\mathbf{w}^*}^\top\boldsymbol{\mu}_1+\kappa(\alpha_1^*)\sqrt{{\mathbf{w}^*}^\top\Sigma_1\mathbf{w}^*} \\ &= -{\mathbf{w}^*}^\top\boldsymbol{\mu}_2-\kappa(\beta^*)\sqrt{{\mathbf{w}^*}^\top\Sigma_2\mathbf{w}^*} \end{aligned} \quad (28)$$

and

$$\begin{aligned} b^* &= -{\mathbf{w}^*}^\top\boldsymbol{\mu}_1+\beta^*\sqrt{{\mathbf{w}^*}^\top\Sigma_1\mathbf{w}^*} \\ &= -{\mathbf{w}^*}^\top\boldsymbol{\mu}_2-\beta^*\sqrt{{\mathbf{w}^*}^\top\Sigma_2\mathbf{w}^*}, \end{aligned} \quad (29)$$

respectively.

Algorithm 1 Modified Quadratic Interpolation for solving ProfMEMPM and ℓ_p-ProfMPM.

```
Input: Let $\theta \in (0,1)$, $\lambda > 0$, $\varepsilon > 0$ be a tolerance sufficiently small, $\beta_1, \beta_2, \beta_3 \in (0,1)$ with $\beta_1 < \beta_2 < \beta_3$, $\hat{\beta} = 10^{99}$, $c_{-1}, b_1$, dataset and labels, and $p \in \{1,2\}$. Set $k = 0$.
1: repeat
2:     Find w by solving Formulation (24) (resp. Formulation (25)) for $\beta = \beta_i$, and compute $f_{profit}(\mathbf{w}, \beta_i)$ via Eq. (15) (resp. Eq. (16)), for $i = 1, 2, 3$.
3: until $f_{profit}(\mathbf{w}, \beta_1) < f_{profit}(\mathbf{w}, \beta_2)$ and $f_{profit}(\mathbf{w}, \beta_2) > f_{profit}(\mathbf{w}, \beta_3)$.
4: Compute
   $$\bar{\beta}^k = \frac{1}{2}\frac{(\beta_2^2 - \beta_3^2) f_{profit}(\mathbf{w}, \beta_1) + (\beta_3^2 - \beta_1^2) f_{profit}(\mathbf{w}, \beta_2) + (\beta_1^2 - \beta_2^2) f_{profit}(\mathbf{w}, \beta_3)}{(\beta_2 - \beta_3) f_{profit}(\mathbf{w}, \beta_1) + (\beta_3 - \beta_1) f_{profit}(\mathbf{w}, \beta_2) + (\beta_1 - \beta_2) f_{profit}(\mathbf{w}, \beta_3)}.$$
5: Find w by solving Formulation (24) (resp. Formulation (25)) for $\beta = \bar{\beta}^k$, and compute $f_{profit}(\mathbf{w}, \bar{\beta}^k)$ via Eq. (15) (resp. Eq. (19)).
6: if $|\bar{\beta}^k - \hat{\beta}| < \varepsilon$ then
7:     return $(\bar{\beta}^k, \mathbf{w})$ and stop.
8: end if
9: if $\bar{\beta}^k < \beta_2$ then
10:    if $f_{profit}(\mathbf{w}, \bar{\beta}^k) \le f_{profit}(\mathbf{w}, \beta_2)$ then
11:        $\beta_1 \leftarrow \bar{\beta}^k$, $f_{profit}(\mathbf{w}, \beta_1) \leftarrow f_{profit}(\mathbf{w}, \bar{\beta}^k)$.
12:    else
13:        $(\beta_2, \beta_3) \leftarrow (\bar{\beta}^k, \beta_2)$,
           $(f_{profit}(\mathbf{w}, \beta_2), f_{profit}(\mathbf{w}, \beta_3)) \leftarrow (f_{profit}(\mathbf{w}, \bar{\beta}^k), f_{profit}(\mathbf{w}, \beta_2))$.
14:    end if
15: end if
16: if $\bar{\beta}^k > \beta_2$ then
17:    if $f_{profit}(\mathbf{w}, \bar{\beta}^k) \le f_{profit}(\mathbf{w}, \beta_2)$ then
18:        $\beta_3 \leftarrow \bar{\beta}^k$, $f_{profit}(\mathbf{w}, \beta_3) \leftarrow f_{profit}(\mathbf{w}, \bar{\beta}^k)$.
19:    else
20:        $(\beta_1, \beta_2) \leftarrow (\beta_2, \bar{\beta}^k)$,
           $(f_{profit}(\mathbf{w}, \beta_1), f_{profit}(\mathbf{w}, \beta_2)) \leftarrow (f_{profit}(\mathbf{w}, \beta_2), f_{profit}(\mathbf{w}, \bar{\beta}^k))$.
21:    end if
22: end if
23: $\hat{\beta} = \bar{\beta}^k$, $k = k + 1$, and repeat from Step 4.
```

Next, the optimization procedure for solving ℓ_p-ProfMEMPM is formalized. We propose two-step coordinate descent approach for solving Formulation (23) (see Bertsekas, 2015, section 6.5 for details). The first step consists in fixing t_1, t_2 and obtaining the optimal values for $\{\mathbf{w}, b, \beta_1, \beta_2\}$ by solving the following inner problem:

$$\begin{array}{ll} \min\limits_{\beta_1, \beta_2, \mathbf{w} \neq \mathbf{0}, b} & \lambda \rho_p(\mathbf{w}) + \frac{c_{-1}\theta}{\beta_1 + 1} + \frac{b_1(1-\theta)}{\beta_2 + 1} \\ \text{s.t.} & \frac{1}{2}\left(\beta_1 t_1 + \frac{\mathbf{w}^\top \Sigma_1 \mathbf{w}}{t_1}\right) \le \mathbf{w}^\top \boldsymbol{\mu}_1 + b, \;\; \beta_1 > 0, \\ & \frac{1}{2}\left(\beta_2 t_2 + \frac{\mathbf{w}^\top \Sigma_2 \mathbf{w}}{t_2}\right) \le -(\mathbf{w}^\top \boldsymbol{\mu}_2 + b), \;\; \beta_2 > 0. \end{array} \tag{30}$$

Formulation (30) is a convex optimization problem given the convexity of both the objective function and the feasible region. We solve this problem using the CVX solver (Grant & Boyd, 2014) for convex optimization.

The next step of the coordinate descent algorithm consist in obtaining t_1 and t_2 for $\{\mathbf{w}, b, \beta_1, \beta_2\}$ fixed. In order to do this, the right-hand side of the mean inequality discussed in the previous section (see Eq. (22)) should be equal to the left-hand side. This equality leads to the following result:

$$t_i = \sqrt{\frac{\mathbf{w}^\top \Sigma_i \mathbf{w}}{\beta_i}}, \quad i = 1, 2. \tag{31}$$

The two-step iterative process is summarized in Algorithm 2.

Proposition 4.1. *The coordinate descent method proposed in Algorithm 2 converges to the optimal solution of Problem (23).*

The proof for Proposition 4.1 is provided in Appendix A.

Algorithm 2 Two-step algorithm for solving ℓ_p-ProfMEMPM.

```
Input: Let $\theta \in (0,1)$, $\lambda > 0$, $\varepsilon > 0$ be a tolerance sufficiently small, $c_{-1}, b_1$, dataset and labels, and $p \in \{1,2\}$. Let $t_1^0, t_2^0 > 0$ be an initial point. Set $k = 0$.
repeat
    Compute $(\mathbf{w}^k, b^k, \beta_1^k, \beta_2^k)$ by solving problem (30).
    Compute $t_1^{k+1}, t_2^{k+1}$ by equation (31).
    $k = k + 1$.
until Stopping criterion is reached
return $(\mathbf{w}^k, b^k, \beta_1^k, \beta_2^k)$ and stop.
```

4.5. Comparative analysis of related methods

As profit maximization techniques for churn prediction, our proposals are strongly related with ProfLogit (Stripling et al., 2018) and ProfTree (Höppner et al., 2018). These two methods consider complex nonlinear optimization problems in the sense that achieving global optimality requires intractable computational effort. As a consequence, the search space is usually limited by heuristics, such as evolutionary algorithms. ProfLogit and ProfTree are solved using genetic algorithms (GA), aimed at finding a good solution in a fixed number of iterations rather than focusing on optimality (Sivanandam & Deepa, 2006). Our proposals, in contrast, solve optimization problems that converge to the optimal solution (see Sun & Yuan, 2006 for details). Therefore, they are of a completely different nature when compared to our proposals, which do not overlap with the existing in the literature on profit metrics.

Next, we would like to clarify the relationship between our proposals and the robust model proposed in López and Maldonado (2019). This approach was especially tailored for credit scoring and, in particular, for a project that with expensive variable collection costs.

The first model proposed in López and Maldonado (2019), called ℓ_p-PSOCP, is inspired by the work by Saketha Nath and Bhattacharyya (2007), which considers variables α_1 and α_2 as previously-specified parameters to be tuned via crossvalidation. Therefore, the profit is not directly optimized in the objective function, in contrast to our three proposals. It also defines a three-term profit metric that consider the benefits and costs of granting credit, but also the variable acquisition costs. Formally, this model solves the following second-order cone programming (SOCP) problem:

$$\begin{array}{ll} \min\limits_{\mathbf{w}\neq\mathbf{0},b,\mathbf{z}} & \lambda_1 \sum_{j=1}^{J} c_j^* z_j + \lambda_2 \rho_p(\mathbf{w}) \\ & \mathbf{w}^\top \boldsymbol{\mu}_0 - b \geq \kappa(\eta_0)\sqrt{\mathbf{w}^\top \Sigma_0 \mathbf{w}}, \\ & -\mathbf{w}^\top \boldsymbol{\mu}_1 + b \geq \kappa(\eta_1)\sqrt{\mathbf{w}^\top \Sigma_1 \mathbf{w}}, \\ & -z_j \leq w_l \leq z_j, \;\; l \in \mathcal{I}_j, \;\; j = 1, \ldots, J. \end{array} \tag{32}$$

with c_j^* denoting the variable acquisition cost of group j estimated at a borrower level. The first term in the objective function in Eq. (32) and the last set of constraints are related with the ℓ_∞-norm penalty (Zou & Yuan, 2008), which is defined by

$$\Gamma(\mathbf{w}) = \sum_{j=1}^{J} ||\mathbf{w}^{(j)}||_\infty, \tag{33}$$

where $||\mathbf{w}^{(j)}||_\infty = \max_{l\in\mathcal{I}_j}\{|w_l|\}$, i.e., the greatest weight (in magnitude) for each group of attributes $j = 1, \ldots, J$ is minimized. The optimization strategy is also very different when compared to the three proposals.

The second model studied in López and Maldonado (2019) is the following optimization problem:

$$\begin{array}{ll} \max\limits_{\mathbf{w}\neq\mathbf{0},b,\eta_1,\mathbf{z}} & \Theta\eta_1 - \lambda_1 \sum_{j=1}^{J} c_j^* z_j - \lambda_2 \rho_p(\mathbf{w}) \\ & \mathbf{w}^\top \boldsymbol{\mu}_0 - b \geq \kappa(\eta_0)\sqrt{\mathbf{w}^\top \Sigma_0 \mathbf{w}}, \\ & -\mathbf{w}^\top \boldsymbol{\mu}_1 + b \geq \kappa(\eta_1)\sqrt{\mathbf{w}^\top \Sigma_1 \mathbf{w}}, \\ & -z_j \leq w_l \leq z_j, \;\; l \in \mathcal{I}_j, j = 1, \ldots, J, \end{array} \tag{34}$$

where we assume that only the parameter $\eta_0 \in (0,\ 1)$ is fixed. Here, the profit metric is optimized which only depends on the variable η_1, that is, $Profit(\eta_1) = \Theta\eta_1 + c$, with $\Theta = c_1\pi_1$ and $c = b_0\pi_0\eta_0 - c_1\pi_1$. This model contains a concave objective function with a linear SOC, a nonlinear SOC and $2n$ linear constraints.

Notice that these models are very different when compared with our regularized proposals because (1) these formulations consider a very different profit measure that includes the variable acquisition costs, (2) they are tailored for credit scoring, and (3) they fix either one or the two class recall variables η, treating them as tuning parameters. Notice that none of our proposals fix the class recalls.

Finally, the regularized MEMPM model proposed in Maldonado et al. (2019) (cf. Eq. (9)) is a smooth nonlinear SOCP problem, which was designed to be optimized via an interior point algorithm called FDIPA (Canelas et al., 2019). This approach is completely different when compared to our proposals. Additionally, the contribution of the model proposed in Maldonado et al. (2019) is purely methodological in binary classification, in contrast with the current study, which has an hybrid positioning in business analytics.

5. Experimental results

In this section, we report experiments on nine churn prediction datasets previously used for benchmarking in similar studies (see e.g. Stripling et al., 2018; Verbeke et al., 2012; Zhu, Baesens, & vanden Broucke, 2017). The relevant information for each benchmark dataset is summarized in Table 1.

Table 1
Number of observations, variables, and churn rate for all datasets.

ID	Name	Region	# Obs.	# Att.	% Churn
K1	Korean1	East Asia	14490	20	23.11
K2	Korean2	East Asia	3283	18	40.6
K3	Korean3	East Asia	4574	14	38.54
K4	Korean4	East Asia	5327	47	46.72
K5	Korean5	East Asia	3441	14	38.68
K6	Korean6	East Asia	44942	12	43.63
D1	Duke1	North America	93893	50	49.75
D2	Duke2	North America	20406	73	1.99
O1	Operator1	North America	47761	47	3.69

The experimental setting used in Zhu et al. (2017) was applied for all datasets and methods. Two-fold crossvalidation was performed five times (5x2 CV), and the average value for the AUC, MPC, and EMPC metrics were computed. The EMPC is the main metric for this study since it incorporates the distribution of the random variable gamma (the fraction of the would-be churners that accept the incentive) in the modeling process, and, is therefore more complete than MPC and AUC (Verbraken et al., 2012; Zhu et al., 2017).

The following classification methods were studied:

- k-nearest neighbors (k-NN). Parameter k was set to 5.
- Standard logistic regression (Logit). No parameter tuning is required for this approach.
- Naïve Bayes (N. Bayes). No parameter tuning is required for this approach.
- Soft-margin SVM (ℓ_2-SVM). The following values were explored for the parameter C: $\{2^{-7}, 2^{-6}, \ldots, 2^{6}, 2^{7}\}$.
- The ProfLogit and ProfTree strategies. For ProfTree, the maximal depth was set to 3. For ProfLogit, the number of iterations for the GA was set to 50. The default configurations of the implementations made by the authors of these methods were considered for the regularization parameters included in both methods.
- Minimax Probability Machine (MPM). No parameter tuning is required for this approach.
- Biased Minimax Probability Machine (BMPM). The following values were explored for the parameter β_0 (fixed value for the sensitivity): {0.2, 0.4, 0.6, 0.8}.
- Minimum Error Minimax Probability Machine (MEMPM). No parameter tuning is required for this approach since θ is the prior probability for class -1, as suggested in Huang et al. (2004).
- The proposed profit-driven approaches. The following values were explored for the parameters θ and λ: $\theta \in \{2^{-7}, 2^{-6}, \ldots,\ 2^{-1}, 1-2^{-1}, \ldots,\ 1-2^{-6},\ 1-2^{-7}\}$, $\lambda \in \{2^{-8},\ 2^{-6}, 2^{-4}, \ldots, 2^{4}, 2^{6}\}$.

Following the studies by Verbeke et al. (2012) and Zhu et al. (2017), dummy encoding was considered for categorical variables. Data resampling was performed for the datasets that exhibit the class-imbalance problem. In particular, random undersampling was applied on the datasets with more than 20,000 samples. The Fisher score was used to filter out irrelevant variables for those datasets that have more than 30 features. The dimensionality was reduced to 30 variables for those datasets, as suggested in Zhu et al. (2017). For a given variable j, the Fisher score has the following formula (Duda, Hard, & Stork, 2001):

$$Fisher(j) = \frac{|\mu_j^+ - \mu_j^-|}{(\sigma_j^+)^2 + (\sigma_j^-)^2}, \tag{35}$$

Table 2
Predictive performance summary for the various classification methods. EMPC measure.

EMPC(€)	K1		K2		K3		K4		K5	
method	mean	std	mean	std	mean	std	mean	std	mean	std
k-NN	4.89	0.03	16.3	0.02	15.1	0.04	21.0	0.07	16.8	0.15
Logit	2.68	0.20	14.7	0.29	14.3	0.21	20.9	0.09	10.0	0.23
N. Bayes	2.73	0.14	12.7	0.22	14.2	0.39	20.3	0.15	13.9	0.46
ℓ_2-SVM	2.38	0.28	14.0	0.58	14.5	0.20	20.5	0.17	9.6	0.23
ProfLogit	4.02	0.23	17.3	5.42	16.7	2.37	21.6	2.29	28.0	0.49
ProfTree	4.18	0.19	**18.6**	0.20	**23.1**	0.12	**23.9**	0.05	**65.8**	0.19
MPM	2.73	0.22	12.8	0.47	14.0	0.24	20.9	0.12	9.8	0.31
BMPM	1.66	0.21	11.5	0.21	14.2	0.23	21.0	0.11	9.7	0.24
MEMPM	4.99	0.31	16.4	0.29	18.0	0.29	21.8	0.18	15.2	0.15
ProfMEMPM	4.99	0.31	16.4	3.27	18.0	0.35	21.9	0.14	15.2	0.13
ℓ_2-ProfMPM	5.10	0.18	16.5	0.37	17.9	0.36	22.0	0.19	15.2	0.19
ℓ_1-ProfMPM	5.02	0.25	16.4	0.39	18.0	0.34	21.2	0.29	15.2	0.11
ℓ_2-ProfMEMPM	**5.23**	0.05	16.56	0.09	18.03	0.22	21.39	0.15	15.19	0.05
ℓ_1-ProfMEMPM	5.21	0.04	16.56	0.11	18.12	0.21	21.20	0.17	15.17	0.06

Table 3
Predictive performance summary for the various classification methods. EMPC measure.

EMPC(€)	K6		D1		D2		O1	
method	mean	std	mean	std	mean	std	mean	std
k-NN	18.30	0.01	22.34	0.00	0.000	0.00	0.000	0.00
Logit	18.94	0.03	22.34	0.00	0.000	0.02	0.068	0.07
N. Bayes	18.45	0.04	22.34	0.00	0.000	2.74	0.005	0.97
ℓ_2-SVM	15.30	0.05	22.34	0.14	−0.066	0.02	0.010	0.03
Proflogit	13.76	6.23	21.46	0.58	0.000	0.17	0.006	0.54
Proftree	8.86	0.06	22.30	0.00	0.000	0.00	0.002	0.00
MPM	15.18	0.06	13.85	0.06	−0.191	0.03	0.001	0.01
BMPM	15.15	0.04	11.61	0.18	−0.191	0.03	−0.005	0.01
MEMPM	18.65	1.50	22.34	0.16	0.000	0.09	0.059	0.01
ProfMEMPM	18.71	0.13	22.34	0.03	0.003	0.09	0.058	0.01
ℓ_2-ProfMPM	18.30	0.10	22.34	0.06	0.003	0.05	0.031	0.26
ℓ_1-ProfMPM	18.56	0.07	22.34	0.04	0.004	0.03	0.070	0.27
ℓ_2-ProfMEMPM	20.64	2.17	**22.42**	0.09	**11.28**	11.89	11.35	11.88
ℓ_1-ProfMEMPM	**20.76**	2.13	**22.42**	0.09	11.27	11.88	**11.37**	11.91

Table 4
Predictive performance summary for the various classification methods. AUC measure.

AUC	K1		K2		K3		K4		K5	
method	mean	std	mean	std	mean	std	mean	std	mean	std
k-NN	0.60	0.00	0.65	0.01	0.72	0.01	0.85	0.00	0.84	0.00
Logit	0.62	0.01	**0.83**	0.01	0.85	0.01	0.90	0.00	0.68	0.01
N. Bayes	0.60	0.01	0.77	0.01	0.84	0.01	0.87	0.01	0.81	0.01
ℓ_2-SVM	0.58	0.02	0.79	0.01	0.85	0.01	0.88	0.01	0.67	0.01
ProfLogit	0.62	0.01	0.80	0.01	0.81	0.02	0.77	0.07	0.64	0.01
ProfTree	**0.64**	0.01	0.81	0.02	**0.89**	0.01	0.82	0.02	**0.94**	0.00
MPM	0.62	0.01	0.75	0.01	0.85	0.01	0.90	0.00	0.67	0.01
BMPM	0.55	0.01	0.70	0.01	0.85	0.01	0.90	0.00	0.66	0.01
MEMPM	0.55	0.01	0.68	0.01	0.85	0.01	0.90	0.00	0.65	0.01
ProfMEMPM	0.55	0.01	0.68	0.12	0.85	0.01	0.90	0.00	0.65	0.01
ℓ_2-ProfMPM	0.58	0.01	0.71	0.01	0.85	0.01	**0.90**	0.00	0.66	0.01
ℓ_1-ProfMPM	0.57	0.01	0.69	0.01	0.84	0.01	0.87	0.00	0.66	0.01
ℓ_2-ProfMEMPM	0.61	0.00	0.70	0.01	0.86	0.01	0.88	0.01	0.66	0.01
ℓ_1-ProfMEMPM	0.61	0.01	0.70	0.01	0.86	0.01	0.87	0.01	0.66	0.01

Table 5
Predictive performance summary for the various classification methods. AUC measure.

AUC	K6		D1		D2		O1	
method	mean	std	mean	std	mean	std	mean	std
k-NN	0.68	0.00	0.54	0.00	0.53	0.01	0.60	0.01
Logit	0.79	0.00	0.60	0.00	0.55	0.01	0.73	0.02
N. Bayes	0.72	0.01	0.57	0.00	0.53	0.01	0.69	0.04
ℓ_2-SVM	**0.80**	0.00	0.59	0.00	0.52	0.01	0.71	0.02
ProfLogit	0.75	0.21	0.56	0.02	0.54	0.05	0.56	0.03
ProfTree	0.68	0.04	0.59	0.01	0.53	0.02	0.58	0.02
MPM	0.78	0.00	**0.61**	0.00	0.50	0.01	**0.73**	0.05
BMPM	0.78	0.00	0.53	0.01	0.51	0.01	0.72	0.03
MEMPM	0.75	0.06	0.51	0.01	0.51	0.05	0.70	0.01
ProfMEMPM	0.77	0.01	0.57	0.00	0.53	0.04	0.70	0.02
ℓ_2-ProfMPM	0.64	0.00	0.57	0.00	0.55	0.01	0.68	0.01
ℓ_1-ProfMPM	0.77	0.00	0.57	0.00	**0.56**	0.01	0.70	0.03
ℓ_2-ProfMEMPM	0.75	0.00	0.58	0.00	**0.56**	0.05	0.71	0.01
ℓ_1-ProfMEMPM	0.77	0.00	0.59	0.00	0.53	0.05	0.72	0.01

where μ_j^+ and μ_j^- are the means of the two classes, and σ_j^+ and σ_j^- are their respective standard deviations.

Following the framework of Verbeke et al. (2012), the following values were used for the parameters related to the MPC and EMPC measures: $CLV = 200$, $\delta = \frac{d}{CLV} = \frac{10}{200}$, and $\phi = \frac{f}{CLV} = \frac{1}{200}$. These metrics are reported in Euro per customer. Parameter γ is set to 0.3 for the MPC measure, while it is assumed to follow a Beta distribution for the EMPC metric, with $\alpha = 6$ and $\beta = 14$ as shape parameters.

Notice that the use of a single CLV value instead of individual CLVs is an important limitation of the framework by Verbeke et al. (2012). The use of this framework is due mainly to lack of data availability since the datasets used in this study do not include the information required to estimate individual CLVs. However, there are several studies that overcome this issue by taking individual CLVs into account; see Bahnsen, Aouada, and Ottersten (2015), and Oskarsdottir, Baesens, and Vanthienen (2018).

The average EMPC and AUC are reported inTables 2–5 for all the methods and datasets. The largest value for these metrics among the eleven methods is highlighted in bold type for all the datasets. The performances in terms of the MPC, F1, precision, and recall metrics are presented in the Appendix A.

On Tables 2 and 3, we observe that the proposed methods achieve very good performance in general, being the best method or close to it in most cases, when considering EMPC as performance metrics. ProfTree achieves excellent performance on some datasets (K2 to K5), but it is not as consistent as our proposals in terms of performance. In contrast, there is no clear best approach when the performance is studied using AUC (Tables 4 and 5), and the choice of the best method for each dataset is clearly not consistent with the results in Tables 2 and 3. A similar conclusion can be drawn from the metrics MPC, F1, precision, and recall, which are presented in the tables in Appendix A.

In order to confirm that our approach has the best overall performance, the Friedman test and Holm test were used to assess statistical significance. This approach was suggested in Demšar (2006) for comparing classification performance among various machine learning methods, and it was used in Zhu et al. (2017) in the context of churn prediction. The first step consists of computing the average rank for each method based on EMPC. Next, the Friedman test with Iman-Davenport correction is applied for assessing whether or not all the average ranks are statistically similar (Demšar, 2006). The Holm post-hoc test is used in case the null hypothesis of equal ranks is rejected. This test performs pairwise comparisons between each method and the one with the best performance (Demšar, 2006).

The result for the F statistic obtained with the Friedman test is $F = 63.61$, with a p value below 0.001, rejecting the null hypothesis of equal ranks. The results for the Holm test are presented in Table 6. For each model, we present the average rank, the average EMPC, the p value obtained with the Holm test, the significance threshold defined for the test, and the result of the pairwise test. This result is ‘reject’ when the p value is below the significance threshold $\alpha/(j-1)$, with $\alpha = 5\%$ and $j = 2, \ldots, 14$ being the overall ranking for a given technique. This outcome implies that

Table 6
Holm post-hoc test for pairwise comparisons.

Method	Mean Rank	Avg. EMPC	p value	$\alpha/(j-1)$	Action
ℓ_1-ProfMEMPM	3.0000	15.7878	–	–	Not reject
ℓ_2-ProfMEMPM	3.1667	15.7867	0.9326	0.0500	Not reject
ProfMEMPM	5.3889	13.0701	0.2257	0.0250	Not reject
MEMPM	5.7222	13.0521	0.1675	0.0167	Not reject
ℓ_1-ProfMPM	6.0000	12.9816	0.1282	0.0125	Not reject
ℓ_2-ProfMPM	6.0000	13.0349	0.1282	0.0100	Not reject
ProfTree	6.0000	18.5269	0.1282	0.0083	Not reject
ProfLogit	7.2222	13.6496	0.0323	0.0071	Not reject
k-NN	8.3333	12.7333	0.0068	0.0062	Not reject
Logit	8.5000	11.5520	0.0053	0.0056	Reject
N. Bayes	10.0000	11.6206	0.0004	0.0050	Reject
ℓ_2-SVM	10.6111	10.9538	0.0001	0.0045	Reject
MPM	12.2778	9.8945	0.0000	0.0042	Reject
BMPM	12.7778	9.4038	0.0000	0.0038	Reject

this method is statistically outperformed by the one with the best rank.

On Table 6, it can be seen that ℓ_1-ProfMEMPM achieves the best average performance (best average ranking considering the nine datasets), outperforming BMPM, MPM, logistic regression, ℓ_2-SVM, and Naïve Bayes statistically. However, there are no significant differences among ℓ_1-ProfMEMPM and the remaining methods. The average ranks for our proposals are 3.00 (ℓ_1-ProfMEMPM), 3.16 (ℓ_2-ProfMEMPM), 5.39 (ProfMEMPM), 6.00 (ℓ_1-ProfMPM), and 6.00 (ℓ_2-ProfMPM), being in the top six performances together with MEMPM (mean rank = 5.72). The most sophisticated approaches in our proposal (ℓ_2-ProfMEMPM and ℓ_1-ProfMEMPM) achieve clearly the best mean performances. The ProfTree and ProfLogit methods achieved very good performance in terms of average EMPC, but they also showed a high variance and therefore, they are not well-ranked consistently on all the datasets, unlike our proposed methods.

It can be concluded from these experiments that it is of utmost importance to select the best method using profit metrics. Using statistical measures such as AUC may lead to more accurate classifiers, but those are not as profitable as the one chosen by a profit measure. Additionally, profit metrics can be optimized directly in the model training, and those models can be more profitable than standard classification approaches evaluated with profit measures. In our case, the proposed profit-driven approaches based on robust optimization achieved the best overall performance among the eleven methods.

Finally, the average training times for all methods and datasets are presented in Table 7. These experiments were performed on an HP Envy dv6 with 16 gigabyte RAM (750 GB SSD), and an i7-2620M processor with 2.70 gigahertz. All classification strategies were implemented on Matlab R2014a and Microsoft Windows 8.1 Operating System (64-bits), with the exception of ProfTree and ProfLogit that were implemented in R and Python, respectively.

It can be seen in Table 7 that all running times are tractable and under one minute for most methods and datasets. The proposed ProfMEMPM method shows similar running times when compared to the alternative robust approaches, being also comparable with the standard classification approaches. The proposed ℓ_p-ProfMPM and ℓ_p-ProfMEMPM methods are relatively slow when compared with ProfMEMPM, being similar to the ProfLogit method in terms of running times. This is because ProfMEMPM solves a concave-convex fractional problem (cf. Formulation (24)) as the inner model of the QI algorithm using the Rosen's gradient projection method, while ℓ_p-ProfMPM and ℓ_p-ProfMEMPM deal with more complex inner formulations. On the one hand, ℓ_p-ProfMPM solves a SOCP problem with two second-order cone constraints, which requires larger training times since a generic SOCP solver such as SeDuMi is used. On the other hand, ℓ_p-ProfMEMPM applies a solves a convex optimization problem iteratively. A generic solver is also used for solving this problem; in this case the CVX solver (Grant & Boyd, 2014).

Regarding the alternative approaches, ℓ_2-SVM shows very large running times for Korea 6 and Telecom 2, while the profit-based strategies ProfTree and ProfLogit tend to be slower than the remaining methods, mostly due to the optimization strategy used for training (Genetic Algorithms).

6. Conclusions and future research

Three novel approaches for profit-driven classification is presented in this work. The robust framework presented in Huang et al. (2004) was adapted for direct profit maximization via mathematical programming. A pessimistic approach is assumed in this framework, in which each training pattern needs to be classified correctly even for the worst data distribution for a given mean and covariance matrix. The robustness conferred by this pessimistic approach has proven to be very effective in improving predictive performance in classification tasks (Gu et al., 2017; Huang et al., 2006; López, Maldonado, & Carrasco, 2017).

Our proposal is tailored for the churn prediction task, using the expected average profit per customer given a retention campaign as objective function. This strategy is inspired in previous studies that evaluate models using profit measures instead of the traditional statistical metrics, such as accuracy or AUC (see e.g. Hand, 2009; Maldonado et al., 2015; Neslin et al., 2006; Verbeke et al., 2012). The proposed method goes one step further and aims at optimizing the profit during the model training. This approach has

Table 7
Running times, in seconds, for all datasets and methods.

Method	K1	K2	K3	K4	K5	K6	D1	D2	O1
k-NN	0".022	0".044	0".028	0".003	0".028	0".009	0".019	0".012	0".000
Logit	0".141	0".191	0".028	0".356	0".156	0".234	2".791	0".348	0".069
N. Bayes	0".025	0".013	0".013	0".025	0".013	0".053	1".013	0".020	0".038
ℓ_2-SVM	1".922	0".497	0".909	1".538	0".703	13".12	378".4	4".918	1".250
ProfLogit	30".17	17".78	15".04	55".07	14".92	31".01	295".1	88".38	44".82
ProfTree	680".2	170".0	228".1	254".5	177".2	1610".8	4804".0	75".16	197".8
MPM	2".438	1".053	1".094	1".519	1".022	2".297	8".747	1".278	1".253
BMPM	0".784	0".188	0".047	0".097	0".288	0".469	0".016	0".219	0".503
MEMPM	0".066	1".122	0".494	0".669	2".778	3".522	0".422	0".013	1".491
ProfMEMPM	0".028	6".294	0".381	0".309	0".138	3".084	0".450	0".000	1".194
ℓ_2-ProfMPM	31".84	16".88	22".31	26".67	20".87	37".56	142".3	20".17	16".32
ℓ_1-ProfMPM	31".49	20".91	20".10	28".38	21".61	53".82	139".8	23".22	24".90
ℓ_2-ProfMEMPM	29".62	30".71	9".93	38".94	10".21	7".45	15".62	21".62	42".54
ℓ_1-ProfMEMPM	13".25	15".30	14".70	61".57	14".34	2".84	16".93	31".0	21".62

shown very positive results (Höppner et al., 2018; Stripling et al., 2018).

Experiments were performed on churn prediction datasets, and the proposed profit-based methods achieved superior performance in average compared to well-known classification techniques. This result confirms the importance of using profit metrics both for evaluating various classification approaches and for calibrating models. Since ProfMEMPM also performs better than MEMPM in terms of profit, we also confirm that our strategy for profit-based parameter selection is a better alternative than the strategy suggested in Huang et al. (2004) in business analytics tasks, such as churn prediction. Finally, the complexity analysis shows that ProfMEMPM is usually faster than ProfLogit and ProfTree, having similar running times when compared to fast standard classification approaches.

Regarding future developments, several opportunities can be identified from this work. First, we would like to assess the hypothesis that robust optimization schemes are able to account for small changes in the data distribution, i.e., are able to construct robust predictors considering an adequate out-of-time validation framework. Unfortunately, the datasets used in this study do not include timestamps or information about retention campaigns, and therefore proving this hypothesis requires a completely new study based on simulated and real-world data. Additionally, the proposed robust framework can be used in multiclass classification tasks. The MPM method was extended to multiclass learning in Hoi and Lyu (2004), providing a good starting point for this avenue of future work. Finally, churn prediction and other business analytics tasks usually face the class-imbalance issue, and this problem can be tackled via cost-sensitive learning. In this work, we deal with the class-imbalance problem via random undersampling, but there are several alternatives that can be explored (see Zhu et al. (2017) for a very detailed benchmark and discussion). Cost-sensitive classification based on a different robust framework was studied in Maldonado and López (2014), providing a starting point for this research opportunity.

Acknowledgments

The authors gratefully acknowledge financial support from CONICYT PIA/BASAL AFB180003 and FONDECYT, grants 1160738 and 1160894. The authors are grateful to the anonymous reviewers who contributed to improving the quality of the original paper.

Appendix A. Proof of Proposition 4.1

Proof. We denote by $J_k = J_k(\mathbf{w}^k, b^k, \beta_1^k, \beta_2^k)$ the optimal value of the objective function evaluated at the optimal solution $(\mathbf{w}^k, b^k, \beta_1^k, \beta_2^k)$ in the k-th iteration. Let us define

$$h_i(t_i, \mathbf{w}, b, \beta_i) = \frac{1}{2}\left(\beta_i t_i + \frac{\mathbf{w}^\top \Sigma_i \mathbf{w}}{t_i}\right) - (-1)^{i+1}(\mathbf{w}^\top \boldsymbol{\mu}_i + b), \quad i = 1, 2.$$

It holds that $h_i(t_i^k, \mathbf{w}, b, \beta_i) \le 0$ and $h_i(t_i^k, \mathbf{w}^k, b^k, \beta_i^k) \le 0$, for $i = 1, 2$. It also holds that $h_i(t_i^{k+1}, \mathbf{w}, b, \beta_i) \le 0$ at iteration $k+1$, for $i = 1, 2$, where

$$t_i^{k+1} = \sqrt{\frac{{\mathbf{w}^k}^\top \Sigma_i \mathbf{w}^k}{\beta_i^k}}, \quad i = 1, 2.$$

Subsequently, the use of this equality leads to the following relation:

$$\beta_i^k t_i^{k+1} + \frac{{\mathbf{w}^k}^\top \Sigma_i \mathbf{w}^k}{t_i^{k+1}} = 2\sqrt{\beta_i^k {\mathbf{w}^k}^\top \Sigma_i \mathbf{w}^k} \le \beta_i^k t_i^k + \frac{{\mathbf{w}^k}^\top \Sigma_i \mathbf{w}^k}{t_i^k},$$

where the above relation is derived from the arithmetic mean-geometric mean inequality. For each class $i = 1, 2,$ it follows from this relation that $h_i(t_i^{k+1}, \mathbf{w}^k, b^k, \beta_i^k) \le h_i(t_i^k, \mathbf{w}^k, b^k, \beta_i^k)$. This implies that the optimal solution $(\mathbf{w}^k, b^k, \beta_1^k, \beta_2^k)$ is a feasible solution of Problem (30) at iteration $k+1$. Since J_{k+1} is the optimal value of the objective function at iteration $k+1$, it holds that $J_{k+1} \le J_k$. Moreover, since the objective function is always positive, Algorithm 2 converges to the solution of problem (23). □

Appendix B. Performance summary in terms of various classification measures

Tables B.1–B.8.

Table B.1
Predictive performance summary for the various classification methods. MPC measure.

MPC	K1		K2		K3		K4		K5	
method	mean	std	mean	std	mean	std	mean	std	mean	std
k-NN	4.49	0.00	16.20	0.01	14.82	0.02	20.31	0.03	16.68	0.19
Logit	2.68	0.21	14.67	0.29	14.32	0.21	20.93	0.09	9.99	0.23
N. Bayes	2.73	0.15	12.70	0.22	14.23	0.39	20.25	0.15	13.88	0.46
ℓ_2-SVM	2.38	0.28	13.97	0.58	14.48	0.20	20.52	0.17	9.62	0.23
ProfLogit	4.31	0.18	17.27	5.40	16.74	2.37	21.56	2.29	28.00	0.30
ProfTree	4.49	0.39	**18.58**	0.23	**23.13**	0.13	**23.86**	0.10	**65.82**	0.35
MPM	2.74	0.22	12.84	0.47	13.97	0.24	20.86	0.12	9.85	0.31
BMPM	1.66	0.21	11.53	0.21	14.21	0.23	20.97	0.11	9.70	0.24
MEMPM	4.64	0.31	16.24	0.29	18.00	0.29	21.66	0.18	15.07	0.15
ProfMEMPM	4.64	0.31	16.26	3.27	18.00	0.35	21.78	0.14	15.07	0.13
ℓ_2-ProfMPM	4.76	0.18	16.38	0.37	17.87	0.36	21.80	0.19	15.11	0.19
ℓ_1-ProfMPM	4.64	0.25	16.30	0.39	18.03	0.34	20.91	0.29	15.07	0.11
ℓ_2-ProfMEMPM	**4.86**	0.05	16.46	0.11	18.02	0.22	21.11	0.20	15.06	0.05
ℓ_1-ProfMEMPM	4.83	0.04	16.46	0.13	18.10	0.21	20.82	0.33	15.03	0.07

Table B.2
Predictive performance summary for the various classification methods. MPC measure.

MPC	K6		D1		D2		O1	
method	mean	std	mean	std	mean	std	mean	std
k-NN	18.23	0.00	22.34	0.00	0.00	0.00	0.00	0.00
Logit	18.83	0.04	22.34	0.00	0.00	0.02	0.04	0.07
N. Bayes	18.32	0.06	22.34	0.00	0.00	3.45	0.00	1.86
ℓ_2-SVM	15.30	0.05	22.34	0.14	−0.09	0.02	0.01	0.04
ProfLogit	17.66	8.20	21.49	0.56	0.00	1.65	0.01	2.11
ProfTree	11.37	0.39	22.34	0.00	0.00	0.00	0.00	0.00
MPM	15.18	0.06	13.85	0.06	−0.19	0.03	0.00	0.01
BMPM	15.15	0.04	11.61	0.18	−0.19	0.03	−0.01	0.01
MEMPM	18.54	1.50	22.34	0.16	0.00	0.09	0.04	0.01
ProfMEMPM	18.58	0.13	22.34	0.03	0.00	0.09	0.04	0.01
ℓ_2-ProfMPM	18.27	0.10	22.34	0.06	0.00	0.05	0.03	0.27
ℓ_1-ProfMPM	18.36	0.07	22.34	0.04	0.00	0.03	0.05	0.35
ℓ_2-ProfMEMPM	20.54	2.19	**22.42**	0.09	**11.28**	11.89	11.31	11.87
ℓ_1-ProfMEMPM	**20.64**	2.15	**22.42**	0.09	11.26	11.87	**11.33**	11.90

Table B.3
Predictive performance summary for the various classification methods. F1 measure.

F1	K1		K2		K3		K4		K5	
method	mean	std	mean	std	mean	std	mean	std	mean	std
k-NN	**0.27**	0.11	0.49	0.10	0.52	0.14	0.63	0.21	0.59	0.22
Logit	0.04	0.03	0.57	0.14	0.58	0.17	0.63	0.22	0.37	0.09
N. Bayes	0.14	0.07	0.44	0.12	0.46	0.13	0.58	0.21	0.49	0.22
ℓ_2-SVM	0.04	0.01	0.66	0.02	0.74	0.02	0.84	0.00	0.13	0.02
ProfLogit	0.01	0.02	0.62	0.26	0.70	0.04	0.67	0.09	0.00	0.00
ProfTree	0.00	0.00	**0.77**	0.02	**0.84**	0.01	0.73	0.04	**0.91**	0.01
MPM	0.05	0.00	0.60	0.03	0.74	0.01	0.84	0.00	0.43	0.02
BMPM	0.00	0.00	0.46	0.02	0.60	0.04	0.84	0.00	0.42	0.03
MEMPM	0.00	0.00	0.24	0.04	0.72	0.02	0.84	0.00	0.39	0.03
ProfMEMPM	0.00	0.00	0.37	0.07	0.72	0.02	**0.84**	0.00	0.39	0.03
ℓ_2-ProfMPM	0.00	0.00	0.51	0.02	0.71	0.01	0.81	0.01	0.46	0.04
ℓ_1-ProfMPM	0.00	0.00	0.49	0.02	0.71	0.01	0.68	0.01	0.00	0.00
ℓ_2-ProfMEMPM	0.04	0.01	0.26	0.06	0.73	0.02	0.77	0.01	0.43	0.03
ℓ_1-ProfMEMPM	0.05	0.01	0.26	0.11	0.71	0.02	0.77	0.01	0.40	0.05

Table B.4
Predictive performance summary for the various classification methods. F1 measure.

F1	K6		D1		D2		O1	
method	mean	std	mean	std	mean	std	mean	std
k-NN	0.65	0.00	0.54	0.00	0.11	0.00	0.03	0.01
Logit	0.70	0.00	0.56	0.00	0.13	0.00	0.04	0.00
N. Bayes	0.64	0.01	**0.63**	0.00	0.07	0.00	0.04	0.01
ℓ_2-SVM	**0.71**	0.00	0.54	0.00	0.13	0.00	0.04	0.01
ProfLogit	0.26	0.27	0.51	0.06	0.00	0.00	0.00	0.00
ProfTree	0.68	0.02	0.61	0.03	0.09	0.01	0.04	0.00
MPM	0.69	0.00	0.58	0.00	0.01	0.00	0.00	0.00
BMPM	0.68	0.00	0.38	0.00	0.00	0.00	0.00	0.00
MEMPM	0.63	0.08	0.38	0.00	0.00	0.00	0.00	0.00
ProfMEMPM	0.66	0.08	0.48	0.00	0.00	0.00	0.00	0.00
ℓ_2-ProfMPM	0.65	0.00	0.53	0.00	0.00	0.00	0.00	0.00
ℓ_1-ProfMPM	0.67	0.00	0.48	0.00	0.00	0.00	0.00	0.00
ℓ_2-ProfMEMPM	0.68	0.03	0.52	0.01	**0.27**	0.30	**0.32**	0.34
ℓ_1-ProfMEMPM	0.69	0.03	0.54	0.01	0.26	0.34	0.26	0.33

Table B.5
Predictive performance summary for the various classification methods. Recall measure.

Recall	K1		K2		K3		K4		K5	
method	mean	std	mean	std	mean	std	mean	std	mean	std
k-NN	**0.21**	0.08	0.49	0.10	0.51	0.13	0.60	0.20	0.60	0.21
Logit	0.02	0.01	0.58	0.14	0.61	0.17	0.59	0.21	0.30	0.08
N. Bayes	0.10	0.05	0.37	0.15	0.38	0.11	0.51	0.18	0.46	0.25
ℓ_2-SVM	0.02	0.01	0.68	0.06	0.82	0.05	**0.79**	0.00	0.07	0.01
ProfLogit	0.00	0.01	0.79	0.38	0.79	0.10	0.56	0.10	0.00	0.00
ProfTree	0.00	0.00	**0.92**	0.07	**0.98**	0.00	0.63	0.08	**0.88**	0.01
MPM	0.03	0.00	0.56	0.06	0.82	0.01	0.78	0.01	0.35	0.03
BMPM	0.00	0.00	0.36	0.03	0.51	0.05	0.78	0.01	0.34	0.04
MEMPM	0.00	0.00	0.15	0.03	0.69	0.02	0.78	0.00	0.31	0.04
ProfMEMPM	0.00	0.00	0.29	0.11	0.69	0.02	0.78	0.01	0.31	0.03
ℓ_2-ProfMPM	0.00	0.00	0.45	0.02	0.81	0.00	0.74	0.01	0.40	0.06
ℓ_1-ProfMPM	0.00	0.00	0.41	0.04	0.81	0.00	0.54	0.02	0.00	0.00
ℓ_2-ProfMEMPM	0.02	0.00	0.16	0.05	0.72	0.03	0.65	0.01	0.35	0.03
ℓ_1-ProfMEMPM	0.03	0.01	0.17	0.09	0.68	0.03	0.65	0.01	0.32	0.06

Table B.6
Predictive performance summary for the various classification methods. Recall measure.

Recall	K6		D1		D2		O1	
method	mean	std	mean	std	mean	std	mean	std
k-NN	0.69	0.01	0.54	0.00	0.58	0.01	0.15	0.05
Logit	0.79	0.00	0.55	0.01	0.63	0.01	0.54	0.05
N. Bayes	0.81	0.01	0.82	0.02	1.00	0.01	0.48	0.31
ℓ_2-SVM	0.79	0.00	0.51	0.01	0.61	0.02	0.38	0.16
ProfLogit	0.22	0.24	0.51	0.14	0.00	0.00	0.00	0.00
ProfTree	**0.88**	0.06	**0.67**	0.08	**0.75**	0.21	**0.89**	0.05
MPM	0.70	0.00	0.57	0.00	0.01	0.00	0.00	0.00
BMPM	0.70	0.00	0.28	0.00	0.00	0.00	0.00	0.00
MEMPM	0.63	0.11	0.28	0.00	0.00	0.00	0.00	0.00
ProfMEMPM	0.68	0.11	0.42	0.00	0.00	0.00	0.00	0.00
ℓ_2-ProfMPM	0.65	0.01	0.50	0.00	0.00	0.00	0.00	0.00
ℓ_1-ProfMPM	0.69	0.00	0.43	0.00	0.00	0.00	0.00	0.00
ℓ_2-ProfMEMPM	0.70	0.05	0.47	0.02	0.30	0.35	0.31	0.32
ℓ_1-ProfMEMPM	0.71	0.05	0.51	0.01	0.37	0.48	0.25	0.32

Table B.7
Predictive performance summary for the various classification methods. Precision measure.

Precision	K1		K2		K3		K4		K5	
method	mean	std	mean	std	mean	std	mean	std	mean	std
k-NN	0.36	0.16	0.50	0.09	0.53	0.15	0.67	0.22	0.58	0.22
Logit	0.39	0.23	0.56	0.13	0.56	0.16	0.68	0.24	0.47	0.10
N. Bayes	0.31	0.10	0.55	0.14	0.58	0.17	0.68	0.25	0.54	0.17
ℓ_2-SVM	**0.64**	0.23	0.65	0.03	0.67	0.05	0.90	0.00	0.64	0.05
ProfLogit	0.07	0.21	0.61	0.05	0.64	0.04	0.85	0.12	0.00	0.00
ProfTree	0.09	0.27	0.67	0.03	0.74	0.02	0.88	0.07	**0.95**	0.01
MPM	0.52	0.03	0.65	0.01	0.68	0.01	0.91	0.00	0.56	0.02
BMPM	0.00	0.00	0.63	0.03	**0.75**	0.01	0.91	0.00	0.55	0.03
MEMPM	0.00	0.00	0.80	0.08	0.74	0.01	0.91	0.00	0.52	0.03
ProfMEMPM	0.00	0.00	0.80	0.08	0.74	0.01	0.91	0.00	0.53	0.03
ℓ_2-ProfMPM	0.00	0.00	0.60	0.01	0.63	0.01	0.89	0.01	0.57	0.02
ℓ_1-ProfMPM	0.00	0.00	0.60	0.01	0.62	0.01	0.94	0.01	0.00	0.00
ℓ_2-ProfMEMPM	0.64	0.07	0.78	0.10	0.74	0.01	**0.95**	0.00	0.55	0.02
ℓ_1-ProfMEMPM	0.58	0.05	**0.81**	0.14	0.74	0.01	**0.95**	0.00	0.55	0.02

Table B.8
Predictive performance summary for the various classification methods. Precision measure.

Precision	K6		D1		D2		O1	
method	mean	std	mean	std	mean	std	mean	std
k-NN	0.62	0.00	0.54	0.00	0.06	0.00	0.02	0.00
Logit	0.63	0.00	0.57	0.00	0.07	0.00	0.02	0.00
N. Bayes	0.54	0.00	0.51	0.00	0.04	0.00	0.02	0.00
ℓ_2-SVM	0.64	0.00	0.56	0.00	0.08	0.00	0.02	0.01
ProfLogit	0.44	0.26	0.53	0.02	0.00	0.00	0.00	0.00
ProfTree	0.55	0.04	0.56	0.01	0.05	0.01	0.02	0.00
MPM	0.67	0.00	0.58	0.00	0.34	0.12	0.00	0.00
BMPM	**0.67**	0.00	0.58	0.00	**0.50**	0.36	0.00	0.00
MEMPM	0.63	0.05	0.58	0.00	0.25	0.35	0.00	0.00
ProfMEMPM	0.65	0.04	**0.58**	0.00	0.18	0.24	0.00	0.00
ℓ_2-ProfMPM	0.64	0.00	0.56	0.00	0.00	0.00	0.00	0.00
ℓ_1-ProfMPM	0.66	0.00	0.55	0.00	0.10	0.32	0.00	0.00
ℓ_2-ProfMEMPM	0.66	0.01	0.57	0.00	0.27	0.28	**0.55**	0.29
ℓ_1-ProfMEMPM	**0.67**	0.01	0.57	0.00	0.23	0.25	0.42	0.29

References

Alizadeh, F., & Goldfarb, D. (2003). Second-order cone programming. *Mathematical Programming, 95*, 3–51.

Baesens, B. (2014). *Analytics in a big data world*. John Wiley and Sons.

Bahnsen, A. C., Aouada, D., & Ottersten, B. (2015). Example-dependent cost-sensitive decision trees. *Expert Systems with Applications, 42*(19), 6609–6619.

Baumann, A., Lessmann, S., Coussement, K., & Bock, K. W. D. (2015). Maximize what matters: Predicting customer churn with decision-centric ensemble selection. In *Proceedings of the 23rd European conference on information systems (ECIS'15), Munster, Germany, May 26–29.*.

Bertsekas, D. P. (1999). *Nonlinear programming* (2nd). Athena Scientific.

Bertsekas, D. P. (2015). *Convex optimization algorithms*. Athena Scientific.

Blattberg, R., Kim, B., & Neslin, S. (2008). *Database marketing: Analyzing and managing customers*. New York: Springer Science+Business Meida, LLC.

Burez, J., & Van den Poel, D. (2009). Handling class imbalance in customer churn prediction. *Expert Systems with Applications, 36*(3), 4626–4636.

Canelas, A., Carrasco, M., & López, J. (2019). A feasible direction algorithm for nonlinear second-order cone programs. *Optimization Methods and Software, 34*(6), 1322–1341.

Chen, Z.-Y., Fan, Z.-P., & Sun, M. (2012). A hierarchical multiple kernel support vector machine for customer churn prediction using longitudinal behavioral data. *European Journal of Operational Research, 223*(2), 461–472.

Datta, P., Masand, B., Mani, D., & Li, B. (2000). Automated cellular modeling and prediction on a large scale. *Artificial Intelligence Review, 14*, 485–502.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data set. *J. Mach. Learn. Res.*, 1–30.

Duda, R., Hard, P., & Stork, D. (2001). *Pattern classification*. Wiley-Interscience Publication.

Farquad, M., Ravi, V., & Raju, S. B. (2014). Churn prediction using comprehensible support vector machine: An analytical CRM application. *Applied Soft Computing, 19*, 31–40.

Fleming, J., & Asplund, J. (2007). *Human sigma: Managing the employee-customer encounter*. New York: Gallup Press.

Grant, M., & Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx.

Gu, B., Sun, X., & Sheng, V. S. (2017). Structural minimax probability machine. *IEEE Transactions on Neural Networks and Learning Systems, 28*(7), 1646–1656.

Hand, D. (2009). Measuring classifier performance: A coherent alternative to the area under the ROC curve. *Machine Learning, 77*(1), 103–123.

Hoi, C., & Lyu, M. (2004). Robust face recognition using minimax probability machine. In *Proceedings of the IEEE international conference on multimedia & expo (ICME)* (pp. 1175–1178).

Höppner, S., Stripling, E., Baesens, B., vanden Broucke, S., & Verdonck, T. (2018). Profit driven decision trees for churn prediction. *European Journal of Operational Research*. doi:10.1016/j.ejor.2018.11.072.

Huang, K., Yang, H., King, I., & Lyu, M. R. (2006). Maximizing sensitivity in medical diagnosis using biased minimax probability machine. *IEEE Transactions on Biomedical Engineering, 53*(5), 821–831.

Huang, K., Yang, H., King, I., Lyu, M., & Chan, L. (2004). The minimum error minimax probability machine. *Journal of Machine Learning Research, 5*, 1253–1286.

Lanckriet, G., Ghaoui, L., Bhattacharyya, C., & Jordan, M. (2003). A robust minimax approach to classification. *Journal of Machine Learning Research, 3*, 555–582.

López, J., & Maldonado, S. (2019). Profit-based credit scoring based on robust optimization and feature selection. *Information Sciences, 500*, 190–202.

López, J., Maldonado, S., & Carrasco, M. (2017). A robust formulation for twin multiclass support vector machine. *Applied Intelligence, 47*(4), 1031–1043.

López, J., Maldonado, S., & Carrasco, M. (2018). Double regularization methods for robust feature selection and SVM classification via dc programming. *Information Sciences, 429*, 377–389.

Ma, J., Yang, L., Wen, Y., & Sun, Q. (2020). Twin minimax probability extreme learning machine for pattern recognition. *Knowledge-Based Systems, 187*, 104806. doi:10.1016/j.knosys.2019.06.014.

Maldonado, S., Carrasco, M., & López, J. (2019). Regularized minimax probability machine. *Knowledge-Based Systems, 177*, 127–135.

Maldonado, S., Flores, A., Verbraken, T., Baesens, B., & Weber, R. (2015). Profit-based feature selection using support vector machines - general framework and an application for customer churn prediction. *Applied Soft Computing, 35*, 740–748.

Maldonado, S., & López, J. (2014). Imbalanced data classification using second-order cone programming support vector machines. *Pattern Recognition, 47*, 2070–2079.

Neslin, S., Gupta, S., Kamakura, W., Lu, J., & Mason, C. (2006). Defection detection: Measuring and understanding the predictive accuracy of customer churn models. *Journal of Marketing Research, 43*(2), 204–211.

Oskarsdottir, M., Baesens, B., & Vanthienen, J. (2018). Profit based model selection for customer retention using individual customer lifetime values. *Big Data, 6*(1), 53–65.

Saketha Nath, J., & Bhattacharyya, C. (2007). Maximum margin classifiers with specified false positive and false negative error rates. In *Proceedings of the SIAM international conference on data mining.*

Schaible, S. (1981). Factional programming: Applications and algorithms. *European Journal of Operational Research, 7*(2), 111–120.

Sivanandam, S., & Deepa, S. (2006). *Introduction to genetic algorithms*. Springer.

Song, S., Gong, Y., Zhang, Y., Huang, G., & Huang, G.-B. (2017). Dimension reduction by minimum error minimax probability machine. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 47*(1), 58–69.

Stripling, E., vanden Broucke, S., Antonio, K., Baesens, B., & Snoeck, M. (2018). Profit maximizing logistic model for customer churn prediction using genetic algorithms. *Swarm and Evolutionary Computation, 40*, 116–130.

Sturm, J. (1999). Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software, 11*(12), 625–653. Special issue on Interior Point Methods (CD supplement with software).

Sun, W., & Yuan, Y.-X. (2006). *Optimization theory and methods: Nonlinear programming*. Springer.

Verbeke, W., Dejaeger, K., Martens, D., Hur, J., & Baesens, B. (2012). New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. *European Journal of Operational Research, 218*(1), 211–229.

Verbeke, W., Martens, D., & Baesens, B. (2014). Social network analysis for customer churn prediction. *Applied Soft Computing, 14*, 431–446.

Verbeke, W., Martens, D., Mues, C., & Baesens, B. (2011). Building comprehensible customer churn prediction models with advanced rule induction techniques. *Expert Systems with Applications, 38*, 2354–2364.

Verbraken, T., Baesens, B., & Bravo, C. (2017). *Profit driven business analytics*. Wiley.

Verbraken, T., Bravo, C., Weber, R., & Baesens, B. (2014). Development and application of consumer credit scoring models using profit-based classification measures. *European Journal of Operational Research, 238*(2), 505–513.

Verbraken, T., Verbeke, W., & Baesens, B. (2012). A novel profit maximizing metric for measuring classification performance of customer churn prediction models. *IEEE Transactions on Knowledge and Data Engineering, 25*(5), 961–973.

Wei, C., & Chiu, I. (2002). Turning telecommunications call details to churn prediction: A data mining approach. *Expert Systems with Applications, 23*, 103–112.

Zhu, B., Baesens, B., & vanden Broucke, S. (2017). An empirical comparison of techniques for the class imbalance problem in churn prediction. *Information Sciences, 408*, 84–99.

Zou, H., & Yuan, M. (2008). The f-infinite norm support vector machine. *Statistica Sinica, 18*, 379–398.