B2Boost: Instance-dependent profit-driven modelling of B2B churn

Bram Janssens[a], Matthias Bogaert[a, b], Astrid Bagué, Dirk Van den Poel[a, b]

[a]Ghent University, Department of Marketing, Innovation and Organisation, Tweekerkenstraat 2, 9000 Ghent, Belgium
[b]FlandersMake@UGent–corelab CVAMO
Bram.Janssens@UGent.Be, Matthias.Bogaert@UGent.Be (Corresponding author), Astrid.Bagué@UGent.Be , Dirk.VandenPoel@UGent.Be

Abstract

The purpose of this paper is to enhance current practices in business-to-business (B2B) customer churn prediction modelling. Following the recent trend from accuracy-based to profit-driven evaluation business-to-customer churn prediction, we present a novel expected maximum profit measure for B2B customer churn (EMPB), which is used to demonstrate how current practices are suboptimal due to large discrepancies in customer value. To directly incorporate the heterogeneity of customer values and profit concerns of the company, we propose an instance-dependent profit maximizing classifier based on gradient boosting, named B2Boost. The main innovation of B2Boost is the fact that it considers these differences and incorporates them into the model construction by maximizing the objective function in terms of the EMPB. The results indicate that the expected maximal profit gains made in our analyses are substantial. This study arguments towards both deploying models based on customer-specific profitability differences, as well as evaluating based on our instance-dependent EMPB measure.

Keywords: B2B Customer Churn, Cost-Sensitive Learning, Churn, Data Mining, Profit-Driven Model Evaluation, Retention Strategies

1. Introduction

Within the field of customer relationship management (CRM), it is known that attracting new customers is much more expensive than retaining existing customers (Reinartz & Kumar, 2003). As a result, both academics and practitioners have heavily researched customer churn prediction (CCP). CCP models focus on identifying the customers that are most likely to defer within a given time period (Burez & Van den Poel, 2007). Based on these CCP models, firms can then focus on pro-actively targeting those customers that are most likely to churn and convince them to retain their relationship with the company. Since it is utterly important that these predictions are as accurate as possible, researchers have heavily focused on improving the performance of CCP models (Verbraken, Verbeke, & Baesens, 2012).

Traditionally, CCP research emphasizes having prediction algorithms that are as performant as possible with regard to discriminating between churners and non-churners. However, recent studies identify the fact that this should not be the end goal of a retention campaign (Höppner et al., 2020). The final goal of a targeted marketing campaign should be to increase profit and this should also be reflected in the underlying algorithms (Verbeke et al., 2012). Therefore, several researchers have introduced algorithms that acknowledge the various costs and benefits associated to successful and unsuccessful identification of would-be churners (Stripling et al., 2018; Höppner et al., 2020; Maldonado, López & Vairetti, 2020). However, these algorithms all assume these costs and benefits to be fixed across all customers. While these assumptions hold in several popular industries which have advanced heavily with regard to CCP (e.g., telecommunication industry), they are not adapted to specificities of business-to-business (B2B) customer churn. In B2B markets, larger discrepancies exist between customer values (Jahromi et al., 2014). This puts serious strain on assumptions of fixed costs and benefits and more specifically on the assumption of a constant customer lifetime value. As a result, these profit-maximizing performance measures and algorithms are not applicable in the field of B2B CCP modelling.

This is a missed opportunity since customer retention management is even more critical in a B2B environment. In B2B markets, there are fewer customers, but they make larger and more frequent purchases (Rauyruen & Miller, 2007). As a consequence, these customers are more valuable (Rauyruen & Miller, 2007), and customer retention is considered central to developing business relationships (Eriksson &

Vaghult, 2000, Kalwani & Narayandas, 1995). Due to the large amounts of money that B2B customers typically spend, retention and the accompanying relationship development have been shown to be extremely financially rewarding for firms operating in a B2B environment (Kalwani & Narayandas, 1995). This further stresses the need for accurate profit-driven evaluation metrics and prediction models in this field. Hence, developing more performant CCP models will have an even larger impact on B2B firm profits than when compared to their B2C counterparts.

A possible solution could lay in the use of customer churn uplift (CCU) models (Ascarza, 2018; Devriendt, Berrevoets & Verbeke, 2021), as used in the first customer-specific profit-driven implementation developed by Lemmens and Gupta (2020). However, a major downside of this method is the need for a randomized controlled trial, as otherwise the models cannot clearly separate the true effect of the retention campaign. Hence, only companies with a large customer base can conduct such experiments. Many companies, however, do not have such a large customer base. This is especially true for B2B companies, which typically have a small customer base of higher value customers (Rauyruen & Miller, 2007) with large variations in customer value within one single customer base. It might be extremely wasteful to use a randomized treatment on high value customers, possibly losing many of them. On top of this, a large sample size in the randomized treatment (in order to have reliable estimates) would also leave a relatively small part of the customers in the deployment set. This would further aggravate the issue of randomized treatments on high value customers, as they are now a major part of the customer base. This entails that a profit-driven extension to the CCU B2B approach suggested by De Caigny et al. (2021) would not be suitable for all B2B firms, and that a profit-driven CCP alternative should exist next to it.

To fill this gap in literature, this study aims to develop a B2B-deployable profit-maximizing CCP framework. To do so, we first propose the expected maximum profit measure for B2B customer churn (EMPB), which takes into account the heterogeneity of customer values that exist in B2B churn. The main difference between our EMPB and the CLV-variable Expected Maximum Profit measure for customer Churn (EMPC) (Óskarsdóttir, Baesens and Vanthienen, 2018) lies in the incentive cost, which is also dependent on customer value in B2B retention campaigns (Jahromi, Stakhovych & Ewing, 2014). After having developed the EMPB, we propose a novel profit-maximizing algorithm using gradient boosting, named B2Boost, which maximizes the objective function based on our EMPB rather than a function that minimizes misclassifications (i.e., log-likelihood function). The stochastic EMPB is reformatted to its deterministic counterpart, which allows for a simple formulation of the gradient and Hessian, and fast computations. Using a real-life data set of a B2B retailer in fast-moving consumer goods (FMCG), we show how our approach improves the expected profit (EMPB) of retention campaigns on hold-out samples and that current practice in both B2B and B2C leads to suboptimal profits.

The remainder of this study is organized as follows. The next section focuses on related work in customer churn. In Section 3, we introduce a profit-driven framework for B2B churn and develop the EMPB, followed by a discussion of our proposed B2Boost algorithm in Section 4. Section 5 describes our overall methodological set-up, while Section 6 summarizes the results of this methodology. The implications of our results to managers are outlined in Section 7. The study ends with a concluding remark in Section 8, showing its limitations as well as its contributions to current literature.

2. Related Work

2.1. Customer Attrition

Customer attrition (or customer churn) is the situation where customers cease their relationship(s) towards a certain firm they are customer of. It is a complex phenomenon as the defection of customers can exist under many forms (Ascarza et al., 2018a). The most archetypical example is perhaps the complete defection of a customer, where the churner is defined as a customer which ceases any relationship with the selling firm. However, besides this traditional view, one could also model partial churn (Buckinx & Van den Poel, 2005), where a customer remains loyal towards the firm to a certain extent, but reduces the level of commitment. Such partial defection is even harder to observe in certain business settings. A typical distinction is the contractual versus non-contractual firm-customer relationship (Ascarza et al., 2018a). Defection is typically easier to detect when the customer has a contractual relationship towards the firm (Ascarza, Netzer & Hardie, 2018b), whereas the reduction in customer value has to be calculated to identify a churner in a non-contractual setting. This heavily hampers the identification of partial churners, as a reduction in customer value (e.g., yearly expenditure) can be a stochastic deviation. This is even further complicated in today’s consumer market, where certain services (e.g., freemium subscriptions) float somewhere between contractual and non-contractual settings (Ascarza et al., 2018a). The complexity of the topic has spurred a vast research field with multiple research interests. We refer the interested reader to Ascarza et al. (2018a) for a complete overview on the various research fields.

A subfield of literature which is increasingly gaining popularity are customer churn uplift (CCU) models (Ascarza, 2018; Devriendt, Berrevoets & Verbeke, 2021). In these methods, the *net effect of treatment* is modelled rather than the *propensity to churn*. For retention campaigns that are focused on generating revenue, CCU models measure the net incremental revenue due to the retention campaign (Devriendt, Moldovan, & Verbeke, 2018). To do so, practitioners first need to conduct a randomized trial which facilitates the estimation of the treatment effect (i.e., change in profitability). The initial model is deployed on the remaining part of the customers (i.e., those who did not participate in the randomized trial) and the most profitable ones are targeted. A major downside of this method is the need for a randomized controlled trial, as otherwise the models cannot clearly separate the true effect of the retention campaign. Hence, only companies with a large customer base can conduct such experiments. Many companies, however, do not have such a large customer base. This is especially true for B2B companies, which typically have a small customer base of higher value customers (Rauyruen & Miller, 2007). To the best our knowledge there is only one study which investigates CCU models in B2B churn. In their work, De Caigny et al. (2021) show how uplift modelling is feasible on a B2B dataset. They conducted a randomized trial on a large sample of 6,432 B2B customers, of which 1,399 received treatment. The completely randomized treatment is also given to any customer, regardless of CLV. The authors demonstrate how accurate treatment effects can be estimated through the methodology, thereby demonstrating the superiority of the uplift logit leaf model which outperforms several other uplift models in their case study. While their work is an extremely valuable addition to B2B customer attrition literature, not all B2B customers bases have characteristics which are suited for this approach (i.e., size large enough and limited differences in per-customer-profitability). For example in our case large discrepancies in customer value exist, and setting up such a randomized controlled experiment can potentially entail huge financial wastes. Hence, we believe the field could benefit from a CCP approach next to it, which is better suited for such customer bases.

2.2. Customer Churn Prediction Models

Our main focus is the subfield of customer churn prediction models (CCP). The problem of customer attrition has led to churn management's inauguration, whose purpose is to minimize the losses caused by leaving customers and to retain high-value customers, thereby maximizing profit (Verbraken et al., 2012). To do so, decision makers should have insights into the propensity of customers to cease their relationship with the company in a given time period. Churn prediction models are used to assign a churn propensity to each individual customer. This probability to churn is then used to target the group of customers most likely to churn with different tailored retention programs to convince them to stay with the company (Burez & Van den Poel, 2007). In order to deploy an effective customer retention program, the utilized models should be as accurate as possible (Coussement & Van den Poel, 2008), as it would be very wasteful to spend incentive budget on customers who will not churn (Tsai & Lu, 2009). Because of this, the main research field regarding CCP models is focused on improving the performance of these models (Ballings & Van den Poel, 2012; Vafeiadis et al., 2015). For an elaborate overview on CCP literature, we refer the reader to Martens et al. (2011) and De Caigny et al. (2018).

In recent years, researchers have identified that CCP models should not aim at maximizing predictive accuracy, rather they should focus on the most important business requirement: profit maximization. The end goal of a retention campaign should always be to enhance long term profit as high as possible (Verbeke et al., 2012). However, an automated CCP model would identify the customers who are most likely to leave, regardless of how this impacts firm profitability. This way, marketing actions can even have a negative effect on firm profitability (Lemmens & Gupta, 2020). This type of behaviour is clearly undesirable, yet it happens quite often due to misalignments between algorithmic and business objectives.

This is caused by the incorrect evaluation of the algorithms. Ideally, the winning churn model is able to correctly detect would-be churners and take into account the business requirements. Historically, popular CCP performance measures do not explicitly take into account misclassification costs and expected profits. To overcome this issue, Verbraken et al. (2012) proposed the expected maximum profit measure for customer churn (EMPC), based on the framework of Neslin et al. (2006). This performance measure scores CCP algorithms with regard to the expected maximum profit a retention campaign based on those algorithms can create. Doing so, the authors demonstrate how model selection based on the EMPC leads to superior results in terms of profits compared to the traditional AUC. One downside of the EMPC is that it assumes all costs and benefits (i.e., customer value, incentive cost, and contact cost) to be fixed across all customers. This assumption is relaxed by Óskarsdóttir et al. (2018) who extend the metric by allowing individual customer lifetime values (CLV), as the assumption of constant CLV is strongly violated in certain business situations. After developing a framework for profit-driven evaluation based on individualized values, the authors show that there are several discrepencies between their proposed metric and the original EMPC when selecting the best algorithm.

However, these studies still use these profit-driven performance measures post-hoc as they simply evaluate algorithms that are designed to distinguish churners from non-churners, rather than detect would-be churners who are most profitable. This spurred a new wave of research, where academics propose profit-maximizing algorithms that incorporate the profit aspect directly into the model construction. Stripling et al. (2018) were the first to directly integrate the EMPC as a performance metric in the objective function of a logistic model structure. Their method, called ProfLogit, uses an evolutionary algorithm (EA) to estimate the regression coefficients which optimizes an EMPC-based fitness function rather than the binomial log-likelihood. Another algorithm, called ProfTree (Höppner et al., 2020), follows a similar approach but uses a tree-based structure. The main differences lays in the fact that the goal of the EA in ProfTree is to find the optimal split rules that correspond to a maximum on the EMPC landscape, which corresponds to generations of decision trees rather than generations of coefficients. Finally, Maldonado, López and Vairetti (2020) create profit-driven extensions of (minimum error) minimax probability machines. Their benchmark study demonstrates that their extensions outperform ProfTree and ProfLogit on average and that the profit-driven algorithms statistically outperform traditional cost-insensitive machine learning approaches such as logistic regression, support vector machines, and naïve Bayes.

2.3. Customer Churn Prediction in Business-to-Business

When looking at churn in the B2B context, the used CCP models lack differentiation from their B2C counterparts and rather implement insights from the B2C field directly to the B2B field. For instance, Gordini and Veglio (2017) compare which type of hyperparameter optimization would result in optimal model performance: AUC-based or accuracy-based. However, a very similar study was performed by Coussement and Van den Poel (2008) in a B2C context. These type of repeat studies hinder specific development for the B2B field, despite the clear distinctions between both fields (Rauyruen & Miller, 2007; Jahromi et al., 2014). Given how CCP modelling can yield high financial returns in the B2B field (Kalwani & Narayandas, 1995), the lack of profit-driven modelling is a missed opportunity.

To the best of our knowledge, there is only one study in B2B churn that incorporates profits. Jahromi et al. (2014) calculate the profit of B2B churn models according to the framework outlined by Neslin et al. (2006), the same one as used in the EMPC. Nevertheless, the authors only did this after profit-insensitive model evaluation was performed, thus ignoring the profit in the actual model evaluation and selection. More recent studies, such as the one by Gordini and Veglio (2017), even leave out this postliminary profit analysis. Although the study investigates which performance measures are most-suited for parameter tuning, they ignore the EMPC which will eventually lead to sub-optimal model selection in terms of profits.

This lack of focus on profitable campaigns is further aggravated by the specifities of the B2B industry, where each customer is unique and products are often even tailored per customer. The heterogeneity in the customer base is much higher, which is also reflected in the customer-specific incentive costs associated to retention campaigns (Jahromi et al., 2014). This high heterogeneity calls for a customer-specific evaluation and training of models. A typical solution would lay in deploying the current EMPC solutions, with their derived algorithms, to B2B data and evaluate whether the outcome is similar to what is observed in the B2C field. This would once again lead to an underdevelopment of B2B churn models and, therefore, the aim of this study is to implement a B2B-specific profit-driven framework, including the development of a customized performance measure and profit-maximizing classifier.

One downside of the methodology behind current profit-driven algorithms is that it always assumes a fixed benefit-cost confusion matrix across all instances (customers). Maldonado, Domínguez, Olaya and Verbeke (2021) already identify this issue and suggest to create profit-driven metrics with different probability thresholds per customer segment, thereby acknowledging differences in customer value. Nevertheless, this methodology does not directly optimize for profit in the model construction, nor does it handle each customer individually. Instead, it requires modellers to predefine (the number of) customer segments. This while instance-dependent cost sensitive learning has already proven to work in related fields such as fraud detection (Höppner et al., 2021). Therefore, we argue that a customer-specific optimization is better suited to the heterogeneous B2B customer base and is key to our proposed implementation. Only one previous study (Lemmens & Gupta, 2020) used such a customer-specific profit-driven approach. The study, however, followed a CCU paradigm which is often not adequate for B2B applications.

As is demonstrated in our related work section, we observe that current instance-independent methodologies would yield suboptimal profits. Surprisingly, no prior study has (1) created a profit-driven B2B churn evaluation metric, and (2) used such a metric in the training phase of a B2B-specific algorithm, while this could heavily influence profitability of retention campaings. Therefore, we suggest a new version of the EMPC measure, the expected maximum profit measure for B2B customer churn (EMPB), which is altered to the specificities of the industry. The measure is used as the basis for a novel profit-maximizing algorithm using gradient boosting, named B2Boost, which optimizes the gradient and Hessian of the deterministic version of our self-defined stochastic measure. The following sections elaborate on the theory and reasoning behind our newly proposed measure and algorithm.

3. EMPB: A profit-driven evaluation measure for B2B churn

Verbraken et al. (2012) elaborate on how churn campaign profitability is driven by implicit costs and benefits associated to incorrectly and correctly identifying churners. The benefits associated to correctly identifying a non-churner (b_0) as well as the costs corresponding to misclassifying an actual churner as a non-churner (c_1) both correspond to 0, as no action is undertaken for these customers. However, contacted customers will result in an associated benefit (b_1) if classified correctly, and in an associated cost (c_0) if classified incorrectly. A false positive will accept the offered incentive d, while also inducing the contact cost f. A true positive, on the other hand, is not certain to accept the offer, as he or she may still decide to leave the firm. This acceptance (with probability of acceptance γ) decides whether the benefits associated with the customer's retention are retrieved (i.e., the customer's value CLV). Note how the incentive cost is also only activated for accepted offers, while the contact cost is lost anyhow. The number of elements in each quadrant of Table 1 then defines the overall profit of the campaign.

Table 1: Benefit-cost confusion matrix according to Verbraken et al. (2012)

	$\boldsymbol{y = 0}$	$\boldsymbol{y = 1}$
$\boldsymbol{\hat{y} = 0}$	$b_0 = 0$	$c_1 = 0$

$\boldsymbol{\hat{y}=1}$	$c_0 \ = d+f$	$b_1 = \ \gamma(CLV-d)-f$

The elements in each quadrant are determined by the accuracy of the deployed algorithm. An algorithm gives each instance (customer) i a score s_i, if this score is higher than a threshold t, the instance is assumed to be a churner and a non-churner otherwise. The optimal threshold T strongly influences the monetary outcome of the campaign, as more contacts directly translate to more costs and more income. Companies optimize t (often implicitly through the contact rate η) based upon the acceptance rate γ (percentage of contacted customers who accept the offer). This resulted in profit generated by classifier C (P_c) being defined by Eq. (1), with π_0 and π_1 the prior probabilities of classes 0 and 1 and $F_0(t)$ and $F_1(t)$ the cumulative distribution functions of the scores for those classes. Note how $\pi_0 F_0(t)$ corresponds to the fraction of true positives and $\pi_1 F_1(t)$ the fraction of false positives.

$$P_c(t;\gamma,CLV,d,f)=(\gamma(CLV-d)-f)\pi_0F_0(t)-(d+f)\pi_1F_1(t) \qquad (1)$$

Verbraken et al. (2012) assumes that this acceptance rate is uncertain and assumes that γ follows a beta distribution $u_{\alpha,\beta}$ with parameters α and β. This results in an EMPC measure which is defined by Equation (2). Note how the optimal threshold T is defined by the stochastic γ given in equation (3).

$$EMPC=\int_{\gamma} P_c(T(\gamma),\gamma,CLV,d,f)u_{\alpha,\beta}(\gamma)d\gamma \qquad (2)$$

$$T=argmax_{\forall\, t}\{P_c(t;\ \gamma,CLV,d,f)\} \quad (3)$$

Using the optimal threshold T, Verbraken et al. (2012) also determined the expected profit maximizing fraction for customer churn ($\bar{\eta}_{empc}$), which specifies the percentage of the customer base to target in a retention campaign. The authors assume CLV, d and f to be fixed and equal for all customers, making the EMPC per customer fixed as well. This assumption is relaxed by Óskarsdóttir et al. (2018) who adapt the metric by allowing customer-specific values for CLV, as the assumption of constant CLV is strongly violated in certain situations, resulting in an instance-dependent benefit-cost confusion matrix, with the main distinction being situated in b_{1i}, which is now calculated with the customer-specific CLV_i: $\gamma(CLV_i-d)-f$. This causes their EMPC per customer to become variant as well. (Eq. (4)).

$$EMPC_i=\int_{\gamma} P_c(T(\gamma),\gamma,CLV_i,d,f)u_{\alpha,\beta}(\gamma)d\gamma \quad (4)$$

Jahromi et al. (2014) elaborate on the specificities of B2B churn campaigns: B2B customers are characterized by big differences in customer lifetime value (CLV), which results in strongly varying incentive costs in churn campaigns as well ($d_i = \delta * CLV_i$). The variability in CLV makes the CLV-variable EMPC by Óskarsdóttir et al. (2018) highly suitable for B2B practice. However, the main issue is the fact that, in business-to-business practice, the incentive cost d is not fixed but dependent upon CLV (Jahromi et al., 2014). For instance, in contractual settings, it is common to offer a discount at contract renewal time (Lemmens & Gupta, 2020). This dependence on CLV makes the incentive rate δ rather than the overall incentive cost d one of the parameters in Table 2, resulting in Eq. (5), where we summate each customer i's individual $EMPB_i$. Note that the estimation of incentive cost using fixed incentive rate is a proximation of reality. While some clients may deviate from the projected incentive cost, the estimate is closer to reality than a fixed incentive cost across all customers, especially in situations with extreme heterogeneity in customer values. The metric, while designed for B2B campaigns, can also be applied to various B2C settings, where efforts made in campaigns can drastically reduce future customer lifetime value beyond the immediately offered incentive costs (Ascarza et al., 2018a).

Table 2: Instance-dependent benefit-cost confusion matrix for B2B churn

	$\boldsymbol{y = 0}$	$\boldsymbol{y = 1}$
$\boldsymbol{\hat{y} = 0}$	$b_{0i} = 0$	$c_{1i} = 0$
$\boldsymbol{\hat{y} = 1}$	$c_{0i} = \delta CLV_i + f$	$b_{1i} = \gamma(1 - \delta)CLV_i - f$

$$EMPB = \sum_i EMPB_i = \sum_i \int_\gamma P_c(T(\gamma), \gamma, CLV_i, \delta, f) u_{\alpha,\beta}(\gamma) d\gamma \qquad (5)$$

Rather than taking the individual EMP value per customer and treat this vector as a whole, as is done in Óskarsdóttir et al. (2018), we use the same thresholding on the overall customer population. Hence another advantage of our framework is that we can determine the profit maximizing contact rate $\bar{\eta}$ as used in Verbraken et al. (2012), by adding the fraction of true positives true positives ($\pi_0 F_0(T)$) to the fraction of false positives ($\pi_1 F_1(T)$) at the optimal threshold T. This provides the optimal fraction of customers to target in a B2B retention campaign ($\bar{\eta}_{empb}$) and reflects the deployment of the algorithm in a more realistic way. This results in an EMPB measure which can be directly interpreted as the total expected maximal numeric profit to be obtained from the campaign. We also use the total profitability rather than the average profitability (as opposed to Verbraken et al. (2012)) as we believe the average value to be non-sensical in situations with extremely varying CLV values.

4. B2Boost: A profit-driven classifier for B2B churn

The basic idea behind our methodology is to adapt the traditional loss function (i.e., log likelihood) into a loss function that approximates our EMPB function. This requires both (1) a reformulation of the EMPB‘s profit function $P_c(s_i)$ as a cost function $C(s_i)$, with $s_i = D(\boldsymbol{x_i})$, with D the classifier which transforms the input features $\boldsymbol{x_i}$ into a score s_i for each instance i, and (2) an algorithmic implementation which allows for flexible adjustments of the objective function. We use the deterministic formulation, in which γ is a scalar, for this loss function, while the stochastic formulation $\gamma = \int_{\gamma} u_{\alpha,\beta}\, d\gamma$, is used for model evaluation on EMPB. The reason we use the deterministic γ, is the fact that this facilitates the differentiation as displayed beneath. On top of this, a stochastic γ would slow down convergence. The selected deterministic γ should reflect a value on which the stochastic γ has a high probability density. For more information, we refer to Section 5.4. Experimental Set-Up.

Gradient boosting iteratively trains weak learners in the partial residuals of the models and adds these weak learners to the ensemble such that a certain loss function is minimized (Friedman, 2001). One popular choice for these weak learners are decision trees given their instability, flexibility and speed. Whereas, in theory, gradient boosting can work with any loss function, most implementations use an accuracy measure, such as the binomial log likelihood. The XGBoost framework is easily extendible and allows for customized loss functions. Furthermore, XGBoost is a highly efficient, scalable, and performant implementation of Friedman’s gradient boosting. The key difference lies in the addition of a regularization term to control for overfitting and the use of the first and second order derivative to minimize the loss function (Chen and Guestrin, 2016). As a result the XGBoost algorithm is more performant and 10 times faster than the traditional gradient boosting implementations and the go-to algorithm for data scientists in machine learning competitions.

Gradient boosting algorithms learn to predict the error terms and iteratively adjust these towards the optimum. Computationally this means that it optimizes the loss function based on the gradient. Instead of only using first order derivates ($g(s_i)$) of the loss function, XGBoost also uses second order derivatives ($h(s_i)$) to reach the optimum. When recalculating the EMPB towards its cost equivalent $C(s_i)$, this implies that we also need to compute its first and second order derivatives $g(s_i)$ and $h(s_i)$. The XGBoost logic will iterate towards the classifier which minimizes $\frac{1}{N}\sum_i C(s_i)$, with $\boldsymbol{s_i} = D(\boldsymbol{x_i}) = P(y = 1|\boldsymbol{x_i})$ being the predicted probabilities by the algorithm. Note how the algorithm will minimize the (deterministic) averaged $\frac{1}{N}\sum_i C_{deterministic}(s_i)$, while our EMPB measure reports the (stochastic) summated opposite: $\sum_i(-C_{stochastic}(s_i))$. As we are interested in the probability of the event (churn) occurring, s_i represents the score after the use of the logistic activation function.

To calculate $C_i(s_i)$, we need to regard all benefits in Table 2 as negative costs, which results in the instance-dependent cost confusion matrix as depicted in Table 3. This cost matrix can be updated to the cost function depicted in (6). Do note that the left part of the equation ($y_i[s_iC_i(1|1) + (1 - s_i)C_i(0|1)]$) equals the cost occurred when the actual value is 1 (churn), while the right part ($(1 - y_i)[(s_iC_i(1|0) + (1 - s_i)\ C_i(0|0))]$) occurs when the actual value is 0 (no churn). The score s_i outputted by the learner then decides which cost occurs based upon the variable decision threshold *t*. For the left hand side ($y_i = 1$) this means a cost of $C_i(1|1)$ if $s_i = 1$ and $C_i(0|1)$ if $s_i = 0$ ($1 - s_i = 1$). If we then insert the values from Table 3 into the general equation Eq. (6), we get Eq. (7), which is the instance-specific cost function according to the EMPB measure. The algorithm thus updates $D(.)$, which determines s_i, which determines which cost is occurred, in order to minimize $\frac{1}{N}\sum_i C(s_i)$.

Table 3: Instance-dependent cost confusion matrix

	$\boldsymbol{y = 0}$	$\boldsymbol{y = 1}$
$\boldsymbol{\hat{y} = 0}$	$C_i(0|0) = 0$	$C_i(0|1) = 0$
$\boldsymbol{\hat{y} = 1}$	$C_i(1|0) = \delta CLV_i + f$	$C_i(1|1) = f - \gamma(1 - \delta)CLV_i$

$$C(s_i) = y_i[s_iC_i(1|1) + (1-s_i)C_i(0|1)] + (1- y_i)[(s_iC_i(1|0) + (1-s_i)\ C_i(0|0))] \quad (6)$$

$$C(s_i) = y_i[s_i(f - \gamma(1 - \delta)CLV_i)] + (1- y_i)[s_i(\delta CLV_i + f)] \quad (7)$$

The update of $D(.)$ is determined by the cost function's gradient and hessian, as second order approximation defines the loss function used at each iteration in XGBoost (Chen & Guestrin, 2016). Eq. (8) determines the minimized loss structured $Loss_t$ in the t-th iteration, with $\Omega(f_t)$ the regularization term. This thus signifies that g_i and h_i determine the update rule and should be formulated.

$$Loss_t = \sum_{i=1}^{N}[g_iD_t(x_i) + \frac{1}{2}\ h_iD_t{}^2(x_i)] + \Omega(D_t) \quad (8)$$

Note that the functions g_i and h_i are the derivatives compared to the previous iteration's predictions $\hat{y}_i(t-1)$: $g_i = \partial_{\hat{y}(t-1)}loss\left(y_i,\hat{y}_i(t-1)\right)$ & $h_i = \partial^2{}_{\hat{y}(t-1)}loss\left(y_i,\hat{y}_i(t-1)\right)$ (Chen & Guestrin, 2016). We update the predicted scores μ_i before logistic activation. These values are manipulated (updated) and outputted by the learner, after which the logistic function scales them between 0 and 1 to get $s_i = \frac{1}{1+e^{-\mu_i}}$. As the logistic function s_i's derivative is: $\frac{\partial s_i}{\partial \mu_i} = s_i(1-s_i)$, we can formulate $g(s_i)$ as Eq. (9). A further differentiation leads to the second order derivative $h(s_i)$ as $\frac{\partial^2 s_i}{\partial {\mu_i}^2} = s_i(1-s_i)(1-2s_i)$. When we compare Eq. (9) with Eq. (13), we see the similarities between both functions, resulting in Eq. (14). Equations (12) and (14) then define our XGBoost extension, which will be called B2Boost for the remainder of this study.

$$\frac{\partial C(s_i)}{\partial \mu_i} = g(s_i) = s_i(1-s_i)[y_i(C_i(1|1)\text{-}\ C_i(0|1))+(1\text{-}\ y_i)(C_i(1|0)\text{-}\ C_i(0|0))] \quad (9)$$

$$g(s_i) = s_i(1-s_i)[y_i(f - \gamma * (1 - \delta) * CLV_i)+(1\text{-}\ y_i)(f + \delta * CLV_i)] \quad (10)$$

$$g(s_i) = s_i(1-s_i)\ [f y_i - \gamma(1-\delta)CLV_i y_i + f + \delta CLV_i - f y_i + \delta CLV_i y_i] \quad (11)$$

$$g(s_i) = s_i(1-s_i)\ [f + \delta CLV_i + y_i((-\gamma + \gamma\delta + \delta)CLV_i)] \quad (12)$$

$$\frac{\partial g(s_i)}{\partial \mu_i} = h(s_i) = s_i(1-s_i)(1-2s_i)[y_i(C_i(1|1)\text{-}\ C_i(0|1))+(1\text{-}\ y_i)(C_i(1|0)\text{-}\ C_i(0|0))] \quad (13)$$

$$h(s_i) = (1-2s_i)g(s_i) \quad (14)$$

5. Methodology

5.1. Data

Data is obtained from a large North American B2B beverage retailer. Besides transactional data, we also have general customer information, equipment and interaction data from November 15, 2012 to June 13, 2015. The independent period, in which our predictors are created, ranges from November 15, 2012 to June 12, 2014. The dependent period, which is used to create our churn variable, ranges from June 13, 2014 to June 13, 2015. We define a churner as a customer who made zero purchases during the dependent period. Only customers who made a purchase during the independent period are considered (i.e., customers that made a purchase during the independent period). In total, we have 41,739 observations of which 1,573 (3.77%) are churners, while 40,166 (96.23%) customers were still active during the dependent period. This distribution is a clear case of class imbalance. Nevertheless, since we adapt the cost function of our boosting model, we actually perform a cost-sensitive learning approach and, therefore, do not decide to use any resampling approach on the profit-driven learners (Baesens, Van Vlasselaer, Verbeke, 2015, p. 200). A similar imbalance exists in the distribution of customer values. While the vast majority (41,321; 99.00%) has a value below $100,000, we also observe a limited (418; 1.00%) customer segment with extremely high values ranging between $100,000 and $10,000,000. Within the above-$100,000-segment, we observe large discrepancies, with the majority situated in the range $100,000-$1,000,000, but with some extremely valuable customers with CLVs above $5,000,000. Traditional retention campaigns neglect the special attention these high value customers deserve compared to the overall sample of customers, which is the main motivation for developing the EMPB.

5.2. Variables

Table 4 provides an overview of the predictors used in the baseline model. This list includes customer-specific and transactional information, as previously deployed in B2B CCP studies (e.g., Gordini & Veglio (2017)). Traditional churn predictors, such as *recency*, *frequency*, *monetary value*, and *length of relationship* are included. Because differences in purchased products may also create differences in churn behaviour (Larivière & Van den Poel, 2004), monetary variables regarding various product categories are added as well. These transactional features are supplemented with a number of firm demographics and buyer-supplier interactions as these may be indicative of customer value. For instance, we include company size through the number of employees, as larger firms may have a reduced strategic decision speed (Baum & Wally, 2003), possibly resulting in postponed reaction to dissatisfaction. Higher spending ratios could also be indicative of both induced firm loyalty as well as larger dependence (e.g., being their only supplier). Credit score is included to identify possible involuntary churners. Some firms do not wish to stop the buyer-supplier relationship but are obliged due to financial distress. A similar pattern is observed with B2C relationships (financial churn; Burez & Van den Poel, 2008). Other features are about unanticipated buyer-supplier interactions (e.g., call to after-sales services) as these may be indicative of an undesired event. The supplier's handling of these possible issues was included as well. When no interaction occurred, some of these variables are infeasible to compute. In such situations, we use K-Nearest Neighbour Imputation to estimate these values (Troyanskaya et al., 2001). The full set of used variables is listed in Table 4 and their correlations are visualized in Figure 1.

Table 4: Used variables

Variable	Description
Frequency	Number of transactions during independent period
Recency	Time (in days) since last purchase before start dependent period
Monetary Value	Total spending in dollars during independent period
Purchase Quantity	Total spending in product units during independent period
Length of relationship	Time (in days) since first registration
MV energy	Monetary value purchased from product category energy drinks
MV lemonade	Monetary value purchased from product category lemonade
MV S&F	Monetary value purchased from two famous brand lines
MV water	Monetary value purchased from product category water
MV diet	Monetary value purchased from diet products
MV main	Monetary value purchased from main brand line
MV other	Monetary value purchased from other product lines
Employee size	Number of employees at buyer firm
Spending ratio	Ratio between expenditure at supplier and annual revenue
Credit Score	Credit score of buyer firm
Interaction	Binary variable indicating whether or not a customer did start at least one interaction during the observed independent period
Interaction recency	Time (in days) since last interaction before start dependent period

Interaction frequency	Number of interactions during independent period
% First call	Percentage of issues which was resolved upon initial call
Solve time	Average time (in days) to solve issue
Deadline difference	Average time (in days) the issue was handled before or after the due date

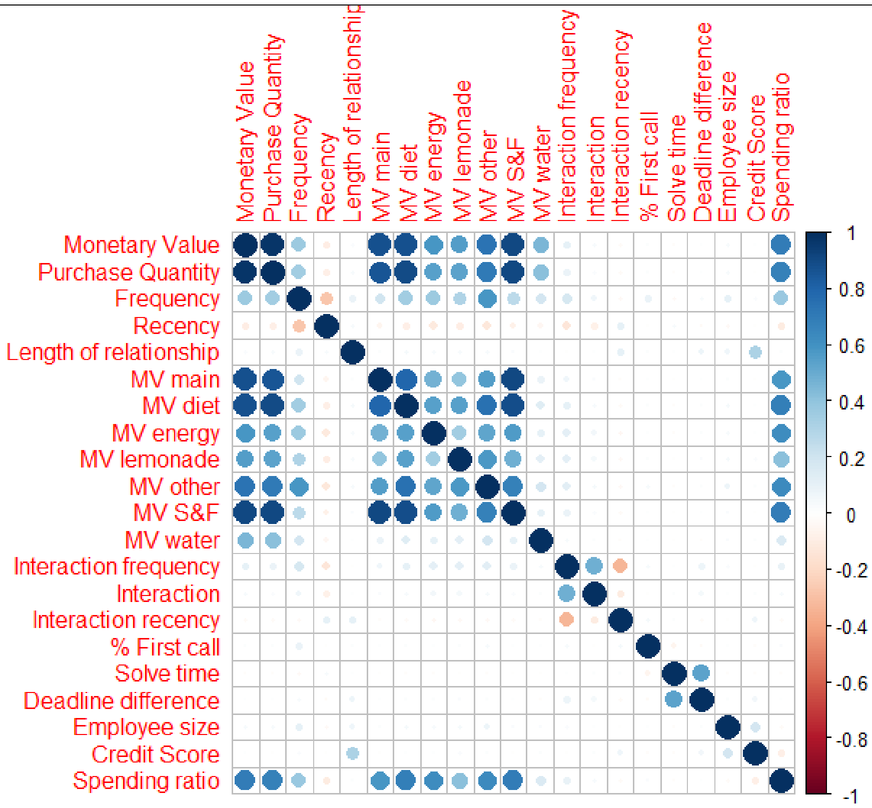

Figure 1: Correlation Heatmap Independent Features

5.3. Algorithms

We compare our proposed B2Boost method against several algorithms that are often deployed in both B2C and B2B churn prediction and yield good performance. First of all, we compare against the default implementation of XGBoost (Chen & Guestrin, 2016), which uses the binomial log likelihood as loss function. By comparing the performance of this algorithm with our model, we can perform a fair evaluation of how optimizing the MPB-based gradient enhances expected profit. Besides the base implementation, we also add an implementation which is based upon the EMPC measure as outlined by Verbraken et al. (2012), called VerbrakenBoost from now on. Typically, cost-sensitive weighting in gradient boosting is done with the parameter α (Wang, Deng & Wang, 2020), which represents the cost ratio between the false positives and the false negatives. A reformulation of the cost matrix proposed by Verbraken et al. (2012) suggests to approximate this value with the ratio $\frac{\gamma(CLV-d)+d}{\gamma(CLV-d)-f}$, with CLV set equal to the average CLV in the training sample and d set equal to 5% of this average CLV. For all three XGBoost models we tune the number of boosting rounds, which determines how many iterations are used to determine optimal tree structure. Learning rate is also optimized as this determines improvement step size, as well as the cost-complexity parameter γ.

Another important type of algorithm to include is a profit-driven classification algorithm as this allows to determine the added value of this CLV-specific method compared to CLV-invariant models. We selected ProfLogit (Stripling et al., 2018) as its implementation is publicly available while its performance is not significantly different from other profit-driven classifiers on a number of datasets (Maldonado, López & Vairetti, 2020). We follow Stripling et al.'s (2018) suggestion and employ a dense search grid for λ. However, we search at lower values than the authors suggested as initial testing revealed that the genetic algorithm set all coefficients equal to zero at higher values of λ. The genetic algorithm is allowed to search for a sufficient number of generations (1000), as long as the best-so-far solution improved during the last 100 generations. Mutation rate is set relatively high (0.50) to ensure an adequate search of the solution space. As the underlying algorithm does not support individualized CLVs and incentive costs, we set these values (CLV and d) equal to the average value in the training set, as was done for the VerbrakenBoost implementation.

Besides ProfLogit, we also add ProfTree as a profit-driven classifier (Höppner et al., 2020). ProfTree uses an evolutionary algorithm to find the optimal tree structure. Again, the parameter λ is extremely important to that regard as it influences the profit-complexity trade-off. Our examined grid only includes values below 0.50, as the algorithm is demonstrated to underperform for high values (Höppner et al., 2020).

Finally, we also add two popular algorithms that are capable of delivering excellent discriminatory power. First, we add random forest (Breiman, 2001). The algorithm's robustness and predictive performance makes it one of the most popular algorithms in CCP (e.g. Burez & Van den Poel, 2007; Burez & Van den Poel, 2009). Second, LASSO regression (Tibshirani, 1996) is also included, as this could shed light on the added value of the instance-independent learning method of ProfLogit in our setting. The models and their candidate settings are summarized in Table 5. All cost-insensitive learners (i.e., XGBoost, Random Forest, and LASSO) are trained after random over-sampling (up until 50/50 ratio) of the minority class is performed on the training samples.

Table 5: Candidate Parameter Values

Algorithm	Parameter	Candidate settings
B2Boost	# Boosting rounds	{2, 5, 10, 20, 50, 100, 200, 500}
	Learning rate	{0.001, 0.01, 0.1, 0.2, 0.5}
	γ	{0.5, 1, 1.5, 2}
VerbrakenBoost	# Boosting rounds	{2, 5, 10, 20, 50, 100, 200, 500}
	Learning rate	{0.001, 0.01, 0.1, 0.2, 0.5}
	γ	{0.5, 1, 1.5, 2}
XGBoost	# Boosting rounds	{2, 5, 10, 20, 50, 100, 200, 500}
	Learning rate	{0.001, 0.01, 0.1, 0.2, 0.5}
	γ	{0.5, 1, 1.5, 2}
ProfLogit	λ	{0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2}
ProfTree	λ	{0.01, 0.05, 0.1, 0.2, 0.5}
Random Forest	# Trees	{2, 5, 10, 20, 50, 100, 200, 500}
	# Features considered (mtry)	{2, 4, 6, 8}
LASSO	λ	{0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2}

5.4. Experimental Set-Up

To ensure the robustness of our results, we employ five times two-fold cross-validation (5×2cv) (Dietterich, 1998). First, 5×2cv randomly splits the data into two equally large folds. Next, each fold is used once as a training and once as a test set. This procedure is iterated five times, which enables us to have ten different performance measures (samples) per algorithm (Demšar 2006). Partitions are made such that the distribution of churners/non-churners and low/high value customers ($100,000 cut-off) is even across folds. A single train-validation split is used to optimize the hyperparameters as is common in predictive modelling studies (Schetgen, Bogaert & Van den Poel, 2021). Hence, the validation set is first used for hyperparameter tuning and the combined training and validation set is eventually used for fitting the final model. Performance is measured through our self-defined EMPB measure, both for hold-out evaluation as well as hyperparameter estimation. During parameter tuning, we noticed that for some fold-algorithm combinations it was not possible for the algorithms to estimate a positive EMPB on the validation set. Therefore, EMPC-based parameter optimization is performed in such cases, as parameter tuning would be infeasible in these cases otherwise.

The B2Boost algorithm and EMPB have several parameters which need to be set. The contact cost d was neglected in Jahromi et al. (2014) and set equal to \$10 in Verbraken et al. (2012). However, in our B2B setting this type of retention campaign is organized by high level account managers compared to CRM employees in B2C firms. United States account managers are reported to earn \$62,717 per year[1]. With a 40 hour work week, spread across 52 weeks, this results in an hourly wage of around \$30. If we assume the interaction with the customer to last for half an hour, we can assume a contact cost of \$15, which seems reasonable given the higher pay grade of account managers and the \$10 figure used by Verbraken et al. (2012). Following Jahromi et al. (2014), we use 0.05 as our incentive cost related to the CLV (δ). With regard to the acceptance rate (γ), both studies (Verbraken et al., 2012; Jahromi et al.; 2014) suggest a value around 0.3. However, Jahromi et al. (2014) assumes this value to be deterministic, while Verbraken et al. (2012) argue that it is not realistic to set such an unpredictable parameter to a certain value. This is why the authors suggest to use the β distribution, with α = 6 and β = 14, leading to γ ranging between 0.1 and 0.5. This β distribution is also used for the calculation of the stochastic EMPB measure on which the algorithms are evaluated, while we use the intermediate 0.3 value as the value for γ when using the value in the deterministic B2Boost implementation. With regards to the CLV, we follow Jahromi et al. (2014) and set CLV equal to the purchases made during the last year of the independent period. While this estimate is prone to be an underestimation of overall CLV, we expect this to be a good indication of the value, also for missed profit due to churners, enabling an evaluation of the overall method.

We also report the profit-maximizing fraction of customer to target based on the EMPB ($\bar{\eta}_{empb}$) and EMPC ($\bar{\eta}_{empc}$). For each algorithm-fold combination, we also compute the AUC and EMPC as defined by Verbraken et al. (2012) on the hold-out test sample, to compare how the algorithms perform with regard to cost-insensitive binary predictive performance and with regard to profit-driven measures that do not

[1] https://www.indeed.com/career/account-manager/salaries

account for the individualized values of customers. The EMPC measure is calculated with the settings which are deemed appropriate for a B2B use case. Specifically, we set these values (CLV and d) equal to the average value in the test set, similar to the learning approaches used for the EMPC-driven cost-sensitive learners (e.g., VerbrakenBoost), while the contact cost is set equal to the one used for EMPB evaluation (i.e., \$15). The optimal contact rate according to the EMPC measure is reported as well. Contrary to the traditional implementation ($EMPC_{traditional}$), we do not average the EMPC measure across customers, but rather report the overall profitability ($N_{test} * EMPC_{traditional}$) to make its outcome more directly comparable to the EMPB measure, which also reports the overall profitability.

The functions to implement the EMPB, B2Boost, and VerbrakenBoost are all coded in Python and are made publicly available via https://github.com/bram-janssens/B2Boost.

6. Results

6.1. Classifier Comparison

Table 6 summarizes the results for the seven algorithms across all ten folds. Average values across the folds are reported, with the standard deviation reported in brackets. The best performer per performance measure is indicated in bold. The results clearly indicate how difficult it is for CCP models to create an actual profit under high heterogeneity (i.e., extreme variation in CLV and incentive cost linked to CLV). While most algorithms are capable of scoring competitive values on AUC (i.e., ranging around 0.80), we observe that this does not translate to profitable retention campaigns (as measured through EMPB). The enormous costs associated with incorrect estimations of the behaviour of high value customers eradicate all gains made on lower value actual would-be churners. Algorithms that do not take this into account during the training phase will only coincidently generate an actual profit (i.e., false positives are coincidently low value customers), while our algorithm acknowledges the fact that high value customers should be regarded differently, given the high impact decisions on customer interaction have. Therefore, B2Boost is the only algorithm to create a substantial profit. By contacting only a small portion of customers (i.e., 0.55% across folds), the algorithm is able to detect a sufficient amount of high value would-be churners.

Interestingly, we observe two out of three EMPC-based learners to score competitively on the EMPC measure (i.e., VerbrakenBoost, and ProfLogit), with ProfLogit having the best performance on EMPC from all classifiers. ProfTree, however, scores remarkable low with the lowest EMPC score of all classifiers. An explanation may lay in the underlying decision tree structure, which may be overly simplistic for the task at hand. This is also reflected in the extremely low score on the AUC metric. Nonetheless, distinction based on EMPC is relatively limited, with all classifiers having similar scores. The large average value (induced by the high value customers) which is used as basis for the EMPC measure tends to overvalue customers, which makes the uncertainty surrounding their potential churn behaviour diminish compared to the extremely large financial reward by contacting them if they would churn. This is also reflected in the extremely large contact rates.

Table 6: Average performance of algorithms

	EMPB	$\bar{\eta}_{empb}$	EMPC	$\bar{\eta}_{empc}$	AUC
B2Boost	**$68,455.86**	0.0055	$8,760,645.39	0.9885	0.7538
	(± $106,331.13)	(± 0.0060)	(± $188,712.88)	(± 0.0113)	(± 0.0893)
VerbrakenBoost	$1,133.43	0.0005	$8,790,515.66	0.9840	**0.8240**
	(± $3,584.24)	(± 0.0016)	(± $190,655.48)	(± 0.0176)	**(± 0.0142)**
XGBoost	$0.00	0.0000	$8,791,320.55	0.9905	0.7758
	(± $0.00)	(± 0.0000)	(± $192,307.13)	(± 0.0069)	(± 0.0049)
ProfLogit	$0.00	0.0000	**$8,792,495.90**	0.9905	0.7723
	(± $0.00)	(± 0.0000)	**(± $192,755.30)**	(± 0.0044)	(± 0.0152)
ProfTree	$0.00	0.0000	$8,756,075.92	0.9945	0.6690
	(± $0.00)	(± 0.0000)	(± $337,855.30)	(± 0.0016)	(± 0.0099)
Random Forest	$0.00	0.0000	$8,791,711.79	0.9785	0.8040
	(± $0.00)	(± 0.0000)	(± $187,287.89)	(± 0.0293)	(± 0.0675)
LASSO	$0.00	0.0000	$8,791,619.54	0.9845	0.7811
	(± $0.00)	(± 0.0000)	(± $195,072.71)	(± 0.0055)	(± 0.0059)

Besides B2Boost, only VerbrakenBoost was able to create a positive result. A similar observation can be made when looking at the hyperparameter optimization. B2Boost optimizes on EMPB in each unique fold (i.e., 10 EMPB-based hyperparameter optimizations), while other algorithms do this much more occasionally. Only VerbrakenBoost does this in 50% of folds (i.e., 5 EMPB-based hyperparameter optimizations), while all other classifiers have maximally one EMPB-based optimization (i.e., XGBoost and random forest; other learners have no EMPB-based hyperparameter optimizations).

Other algorithms focus on instances they predict to be certain to churn, but do not acknowledge the differences in value and associated costs and benefits, resulting in unprofitable campaigns (i.e., optimal EMPB contact rate equal to zero). Overall, the results clearly indicate that our algorithm is capable of deriving much more profitable campaigns in B2B settings than traditional CCP models. Traditional methods, including recent algorithm-based enhancements, seem unsuited for the specificities of B2B retention campaigns.

The results in Table 6 also highlight how currently used measures lead to incorrect decision making. When applying the EMPC measure with the default values, as could often be the case for firms operating without decent estimates of their customer values or benefit and incentive costs, one would assume all algorithms to effectively generate a profit. Managers would then deploy the preferred algorithm and are actually expected to lose money doing so, as the ideal contact rate is zero, which means that higher values lead to negative EMPB values. Depending on the used metric, either VerbrakenBoost or ProfLogit would be identified as most performant in our case, while in reality these would be suboptimal decision options.

6.2. Classifier Interpretation

We established the fact that our algorithm is capable of learning value-based differences between customers and how these affect retention campaigns. However, future practitioners also need to know which information is vital to build such models. Therefore, we will compare the variable importances of the profit-based algorithm with those of the default XGBoost implementation. Several methodologies exist to do so. However, many of them have the underlying issue that they have no theoretical foundation. An exception is SHAP (SHapley Additive exPlanations; Lundberg & Lee, 2017). SHAP combines strengths of Shapley values (Shapley, 1953) and LIME (Ribeiro, Singh & Guestrin, 2016) by creating Shapley values of the conditional expectation function of the original model. The theoretical foundation is gained from the usage of Shapley values, which are based on game theory, with features acting as players to influence the outcome (i.e., the deviation from the average predicted value). The importance is then defined as the average contribution to this deviation when a feature is included in the 'coalition'. Variable importances are calculated by aggregating the (absolute) SHAP values across all observations.

Figure 2 displays the SHAP feature importances for the B2Boost algorithm, and the default XGBoost implementation. The traditional transactional RFM variables are the most important ones, both for the

B2Boost implantation and the XGBoost implementation. It seems that while the B2Boost algorithm learns something slightly deviant to what traditional churn models learn (i.e., which customers to prioritize rather than which customers are most probable to churn), it values similar input features as traditional churn models. Interesting to note is the larger emphasis on monetary value in B2Boost, which seems logical given the fact that more profitable customers should be targeted. A somewhat surprising feature is *deadline difference.* The feature is much more important in the B2Boost implementation than in the XGBoost counterpart. Interestingly, one could regard the time the issue was handled before or after the due date, as an outcome of the effort being put in by the selling firm. Firms are likely to put more effort into important customers, making *deadline difference* an indicator of relative customer importance based on the selling firms' own behaviour. *Interaction recency* is the only interaction-based variable to play a significant role in both classifiers. This could indicate that only recent interactions (e.g., complaints) are indicative of churn behaviour, signifying a forgive-and-forget mentality over more long-ago issues in B2B relationships.

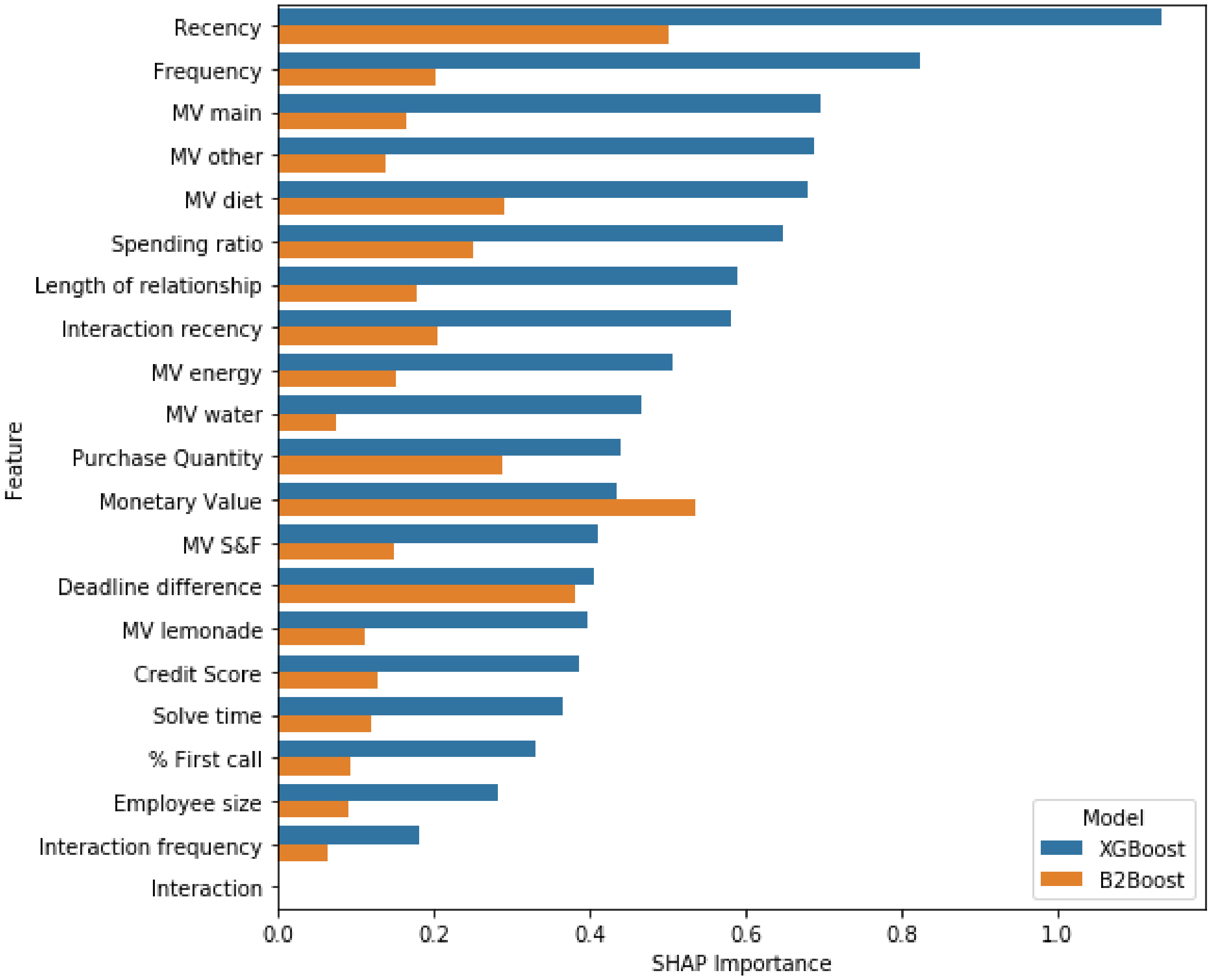

Figure 2: Feature Importances as defined by average SHAP value. Clear larger importance Monetary Value in B2Boost.

Of special interest are the variables *MV diet* and *MV main.* The *MV main* variable corresponds to the monetary value purchased of the main brand line, which seems indicative to the overall commitment to the firm. This while the *MV diet* variable is the monetary value purchased of the diet (reduced sugar) products. These can be seen as more 'specialty' products. It seems reasonable to assume that firms who also purchase these products have a wider product offering of our partner to their own customers, which induces their switching costs. Nevertheless, caution is in place, as the *MV diet* and *MV main* variables are relatively highly correlated (ρ = 0.79). Nonetheless, the fact that both variables are given a relatively high SHAP-based feature importance in both the B2Boost and XGBoost implementation, does indicate that both features have important predictive power despite the correlation bias and that a combined uncorrelated feature (e.g., MV main + diet) would have an even higher feature importance. Since these variables are already ranked high on the importance ranking, the only change would be that they become even more important. Overall, future practitioners of the model are advised to deploy both traditional transactional variables as well as information on customer switching costs and indicators of customer importance.

Table 7 displays the validated hyperparameter settings for each repartition. The parameter settings are remarkably dispersed. Combined with the B2Boost performance, which outperforms all the other classifiers with regard to EMPB, we can state that the algorithm seems relatively robust with regard to the selected parameter settings. Only the gamma parameter seems to be consistently set at 0.5. In general, practitioners are advised to use large enough grid searches.

Table 7: Optimized Hyperparameter Settings

Fold	# Boosting rounds	Learning rate	γ
1	100	0.01	0.5
2	500	0.001	0.5
3	500	0.2	0.5
4	50	0.001	0.5
5	5	0.5	0.5
6	2	0.2	0.5
7	2	0.1	0.5
8	20	0.2	0.5
9	200	0.01	1.5
10	20	0.01	0.5

To gain further insight into the profit-driven classifications suggested by B2Boost compared to the traditional profit-insensitive suggestions, we compare the CLV distributions of the top 1% predicted probabilities of the random forest algorithm with top 1% suggestions of the B2Boost algorithm. These algorithms are selected as they are (1) the profit-unaware algorithm with the highest discriminatory power (random forest) and (2) the best classifier on the EMPB measure (B2Boost). The analysis is performed across two previously unseen folds to ensure robust outcomes.

The results are visualized in Figure 3, which displays the cumulative distribution of CLVs in the suggested top percentile of both B2Boost (blue line) and random forest (orange line). The reader can observe that the random forest algorithm suggests mostly small value customers, as these are most dominantly present in the customer base. The B2Boost algorithm, on the other hand, acknowledges these unique customer values and, as a consequence, suggests higher valued customers to be contacted, resulting in the better EMPB values scored by the B2Boost model.

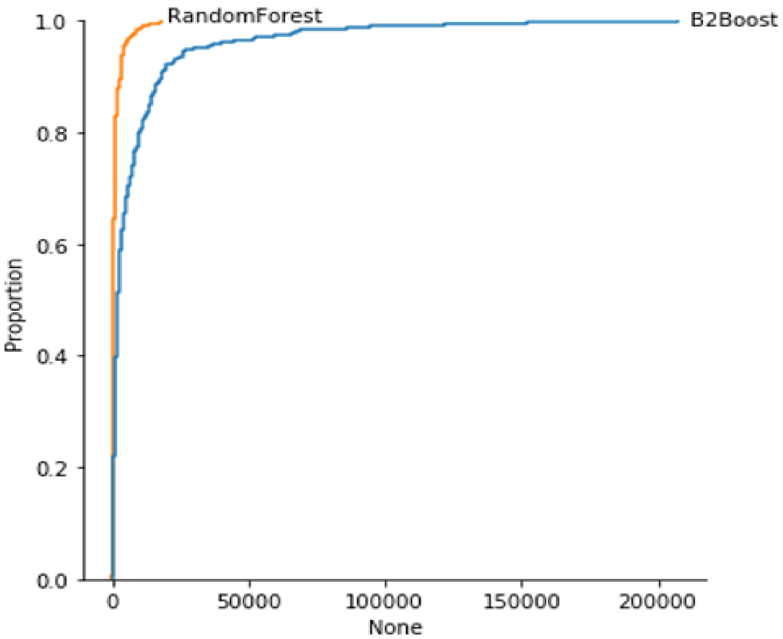

Figure 3: Cumulative distribution CLVs top percentile B2Boost vs random forest

6.3. Sensitivity Analysis

Initially, we wanted to ensure whether our results are representative by using multiple benchmark datasets. However, popular datasets used in churn literature (e.g., the ones used in (Stripling et al., 2018) and (Maldonado, López & Vairetti, 2020)) do not feature such large discrepancies in customer lifetime value. Therefore, we decided to change small details to our methodology to control whether all methods consistently resulted in the competitive performance of B2Boost. This offers the additional benefit of evaluating the robustness and importance of our assumptions. Specifically, we decided to change three specific implementations independently. First, the value of δ is adapted, as one could argue that 5% of CLV offered as incentive is too high, resulting in an over-valued cost associated to misclassification. Both a slightly more conservative view of 0.01 and an extremely low incentive rate of 0.00001, more closely aligned to the B2C values suggested in Verbraken et al. (2012) were tested. Second, we also re-evaluate the used CLV values. Their values in the main analysis are based on a one-year period, which can be too short to comprehend true customer value, which may result in undervalued customers, which also heavily influences the cost-benefit ratio. Therefore, we tested the outcomes for the situations where CLV values were multiplied by factors three and five, which would be more representative of three and five year periods with low discounting rates and stable expenditure patterns. Finally, we also alter the contact cost to the value of $10 as used in Verbraken et al. (2012). While this adaptation is smaller than the first and second adaptations, it also represents a return to the values as proposed in current EMPC literature, which is why we also control for how this assumption affects retention campaign profitability. In each implementation, we adapted these changes accordingly in all possible settings, both with regard to algorithm settings (i.e., profit-driven learners), as well as to the evaluation step (i.e., EMPC and EMPB).

Table 8: Average performance of algorithms with δ = 0.01

	EMPB	$\bar{\eta}_{empb}$	EMPC	$\bar{\eta}_{empc}$	AUC
B2Boost	**\$60,629.28**	0.0045	\$9,164,010.42	0.9855	0.7700
	(± \$50,389.14)	(± 0.0037)	(± \$203,297.56)	(± 0.0233)	(± 0.0177)
VerbrakenBoost	\$22,711.3217	0.0005	\$9,166,766.57	0.9790	**0.8135**
	(± \$71,819.51)	(± 0.0016)	(± \$203,042.54)	(± 0.0371)	**(± 0.0212)**
XGBoost	\$0.00	0.0000	\$9,171,187.93	0.9905	0.7758
	(± \$0.00)	(± 0.0000)	(± \$200,391.27)	(± 0.0069)	(± 0.0049)
ProfLogit	\$0.00	0.0000	**\$9,172,265.47**	0.9860	0.7888
	(± \$0.00)	(± 0.0000)	**(± \$201,087.39)**	(± 0.0094)	(± 0.008)
ProfTree	\$0.00	0.0000	\$9,134,497.53	0.9945	0.6690
	(± \$0.00)	(± 0.0000)	(± 352,074.51)	(± 0.0016)	(± 0.0099)
Random Forest	\$0.00	0.0000	\$9,165,385.01	0.9855	0.7961
	(± \$0.00)	(± 0.0000)	(± \$214,580.83)	(± 0.0128)	(± 0.0754)
LASSO	\$125.48	0.0005	9,171,441.37	0.9845	0.781
	(± \$396.80)	(± 0.0016)	(± \$203,290.51)	(± 0.0055)	(± 0.0059)

Table 9: Average performance of algorithms with δ = 0.00001

	EMPB	$\bar{\eta}_{empb}$	EMPC	$\bar{\eta}_{empc}$	AUC
B2Boost	**\$389,440.95**	0.0555	\$9,267,993.88	0.9715	0.7154
	(± \$90,953.78)	(± 0.0306)	(± \$200,782.53)	(± 0.0187)	(± 0.0239)
VerbrakenBoost	\$23,427.956	0.0005	\$9,252,492.66	0.9935	**0.8069**
	(± \$74,085.70)	(± 0.0016)	(± \$225,180.43)	(± 0.0024)	**(± 0.0234)**
XGBoost	\$0.00	0.0000	\$9,266,059.81	0.9905	0.7758
	(± \$0.00)	(± 0.0000)	(± \$202,410.28)	(± 0.0069)	(± 0.0049)
ProfLogit	\$13,893.76	0.0050	**\$9,269,849.43**	0.9690	0.7346
	(± \$25,711.08)	(± 0.0111)	**(± \$204,939.72)**	(± 0.0240)	(± 0.0132)
ProfTree	\$0.00	0.0000	\$9,229,008.33	0.9945	0.6690
	(± \$0.00)	(± 0.0000)	(± \$355,625.73)	(± 0.0016)	(± 0.0099)
Random Forest	\$7,881.08	0.0035	\$9,247,765.69	0.9820	0.7423
	(± \$24,922.16)	(± 0.0111)	(± \$216,936.20)	(± 0.0247)	(± 0.1095)
LASSO	\$3,574.31	0.0020	\$9,266,301.87	0.9845	0.7810
	(± 5,845.70)	(± 0.0026)	(± \$205,342.90)	(± 0.0055)	(± 0.0059)

Tables 8 and 9 display the effect of diminishing incentive rates. Interestingly, the same algorithms are identified as top performers per performance measure compared to the main analysis with δ = 0.05.

Reducing incentive rates results into more and more customers being identified or worthy of targeting, which is reflected in increased contact rates for both the EMPB and EMPC measure. The lower incentive rates also result into more and more classifiers being capable of creating profitable campaigns, as the cost of contacting a customer which would not churn in the first place becomes much less severe. B2Boost learns this by becoming more focused towards contacting many high value customers and their propensity becomes of comparatively lower importance. However, this does not yet translate to a better performance on the hold-out test set as EMPB performance actually drops for $\delta = 0.01$ when compared to the application where $\delta = 0.05$. The beneficial effect of this adaptive learning, however, becomes clear when looking at the more extreme case of $\delta = 0.00001$. This extremely low incentive rate translates to a significant drop in AUC, becoming strongly outperformed by most learners. However, the focus on valuable customers, makes campaigns based on this algorithm much more profitable than campaigns based on other algorithms. Overall, lower incentive rates translate to more profitable campaigns. Note how the EMPC measure signifies a similar rise in profitability in absolute terms. As the metric, however, overestimates profitability by almost $9,000,000 when compared to the EMPB metric, is this rise much smaller when looking at it from a relative perspective.

Table 10: Average performance of algorithms with $CLVx3$

	EMPB	$\bar{\eta}_{empb}$	EMPC	$\bar{\eta}_{empc}$	AUC
B2Boost	$142,931.93	0.0210	$26,792,474.51	0.9845	**0.7970**
	(± $195,830.32)	(± 0.0311)	(± $619,501.48)	(± 0.0215)	**(± 0.0229)**
VerbrakenBoost	**$239,897.80**	0.0235	$26,624,707.51	0.9910	0.7969
	(± $262,626.82)	(± 0.0257)	(± $758,207.53)	(± 0.0066)	(± 0.0299)
XGBoost	$41,232.35	0.0085	$26,835,021.24	0.9905	0.7758
	(± $78,139.54)	(± 0.0145)	(± $576,311.63)	(± 0.0069)	(± 0.0049)
ProfLogit	$0.00	0.0000	**$26,845,420.62**	0.9905	0.7659
	(± $0.00)	(± 0.0000)	**(± $579,222.60)**	(± 0.0055)	(± 0.0147)
ProfTree	$0.00	0.0000	$8,756,075.92	0.9945	0.6690
	(± $0.00)	(± 0.0000)	(± $337,855.39)	(± 0.0016)	(± 0.0099)
Random Forest	$22,208.33	0.0360	$26,769,198.63	0.9940	0.7250
	(± $46,502.52)	(± 0.0583)	(± $599,656.84)	(± 0.0021)	(± 0.0882)
LASSO	$39,586.8547	0.0010	$26,833,156.43	0.9845	0.7811
	(± 84,003.30)	(± 0.0021)	(± 585,423.07)	(± 0.0055)	(± 0.0059)

Tables 10 and 11 indicate the average performance when customer values are adapted with a factor 3 or 5, respectively. Higher valued customers also heavily influence retention campaigns, with all algorithms receiving positive EMPB scores when CLV values are five times their value of the main analysis. When customer values increase, we observe the algorithm which is best at distinguishing churners from non-churners (i.e., VerbrakenBoost) also to become the most profitable algorithm. This is caused by the lesser influence of costs compared to benefits. A similar observation can be made when inspecting how the B2Boost algorithm copes with these changed customer values. As all customers become more valuable, discriminatory power becomes more important. This causes the B2Boost algorithm to shift its focus towards more discriminatory power. As a result, we see increases in the algorithm's performance as measured by AUC. Overall, when retention campaign costs become less important in relation to the campaign benefits, the B2Boost algorithm's behaviour becomes more reminiscent of traditional discriminative algorithms. This also causes instance-insensitive algorithms (such as random forest in this case) to become competitive with the algorithm.

Table 11: Average performance of algorithms with $CLVx5$

	EMPB	$\bar{\eta}_{empb}$	EMPC	$\bar{\eta}_{empc}$	AUC
B2Boost	\$433,400.21	0.0500	\$44,771,208.95	0.9835	0.7856
	(± 439852.68)	(± 0.0589)	(± \$964,779.88)	(± 0.0204)	(± 0.0574)
VerbrakenBoost	**\$939,427.40**	0.0545	\$44,675,304.62	0.9905	**0.7974**
	(± \$659,471.53)	(± 0.0472)	(± \$1,137,180.22)	(± 0.0083)	**(± 0.0402)**
XGBoost	\$273,001.15	0.0360	\$44,878,721.93	0.9905	0.7758
	(± \$273,847.56)	(± 0.0314)	(± \$960,319.47)	(± 0.0069)	(± 0.0049)
ProfLogit	\$11,917.447	0.0115	**\$44,896,592.23**	0.9885	0.7544
	(± \$14,954.221)	(± 0.0147)	**(± \$969,292.53)**	(± 0.0071)	(± 0.0107)
ProfTree	\$0.00	0.0000	\$8,756,075.92	0.9945	0.6690
	(± \$0.00)	(± 0.0000)	(± \$337,855.39)	(± 0.0016)	(± 0.0099)
Random Forest	\$554,590.28	0.1325	\$44,821,480.85	0.9905	0.7811
	(± \$519,886.87)	(± 0.0943)	(± \$968,813.39)	(± 0.0080)	(± 0.0542)
LASSO	\$90,672.92	0.0085	\$44,874,807.71	0.9840	0.7809
	(± \$179,132.41)	(± 0.0201)	(± 975,900.98)	(± 0.0066)	(± 0.0060)

The effect of a smaller contact cost of \$10 is relatively small, with the conclusions from Table 12 (i.e., results with f = \$10) being not too deviant from the ones in Table 6 (i.e., results with f = \$10). This is not surprising, given the relatively small deviation in contact cost and the relatively small importance of contact costs when compared to incentive costs and campaign benefits in the B2B industry.

Table 12: Average performance of algorithms with f = $10

	EMPB	$\bar{\eta}_{empb}$	EMPC	$\bar{\eta}_{empc}$	AUC
B2Boost	**$35,910.33**	0.0040	$8,765,532.91	0.9910	0.7857
	(± $74,531.63)	(± 0.0077)	(± $206,720.83)	(± 0.0070)	(± 0.0345)
VerbrakenBoost	$8,706.85	0.0020	$8,771,343.08	0.9920	**0.8147**
	(± $19,174.86)	(± 0.0035)	(± $224,926.50)	(± 0.0035)	**(± 0.0194)**
XGBoost	$0.00	0.0000	$8,795,251.05	0.9905	0.7758
	(± $0.00)	(± 0.0000)	(± $192,305.57)	(± 0.0069)	(± 0.0049)
ProfLogit	$0.00	0.0000	**$8,798,275.99**	0.9875	0.7634
	(± $0.00)	(± 0.0000)	**(± $194,166.51)**	(± 0.0072)	(± 0.0154)
ProfTree	$0.00	0.0000	$8,759,991.42	0.9945	0.6690
	(± $0.00)	(± 0.0000)	(± $337,855.79)	(± 0.0016)	(± 0.0099)
Random Forest	$0.00	0.0000	$8,785,024.30	0.9860	0.7507
	(± $0.00)	(± 0.0000)	(± $202,389.34)	(± 0.0137)	(± 0.0884)
LASSO	$4,376.62	0.0010	$8,795,549.54	0.9845	0.7811
	(± $9,444.42)	(± 0.0021)	(± 195,072.54)	(± 0.0055)	(± 0.0059)

An important aspect which has to be considered when dealing with point estimates such as the estimated CLV, is the fact that these estimates are uncertain and that, in reality, actual customer value might deviate from these estimates. To account for this uncertainty, we add some random noise around a multitude of the currently predicted CLV values to account for the uncertainty about our CLV estimates. Specifically, we include random noise to each unique CLV estimate in the final evaluation step. This noise was normally distributed with mean 0 and standard deviation equal to the mean value of the CLV estimates. This process is repeated 100 times, resulting in 100 unique CLV estimate vectors used for evaluation. B2Boost and random forest are evaluated using this method in a two-fold cross-validation (i.e., two unseen folds which were not included in main analysis), resulting in 200 unique EMPB evaluations per algorithm. Similar to the algorithm selection used for the analysis which is depicted in Figure 3, these algorithms are selected as they are (1) the profit-unaware algorithm with the highest discriminatory power (random forest) and (2) the best classifier on the EMPB measure (B2Boost). Hyperparameters were optimized using the same method as in the main analysis.

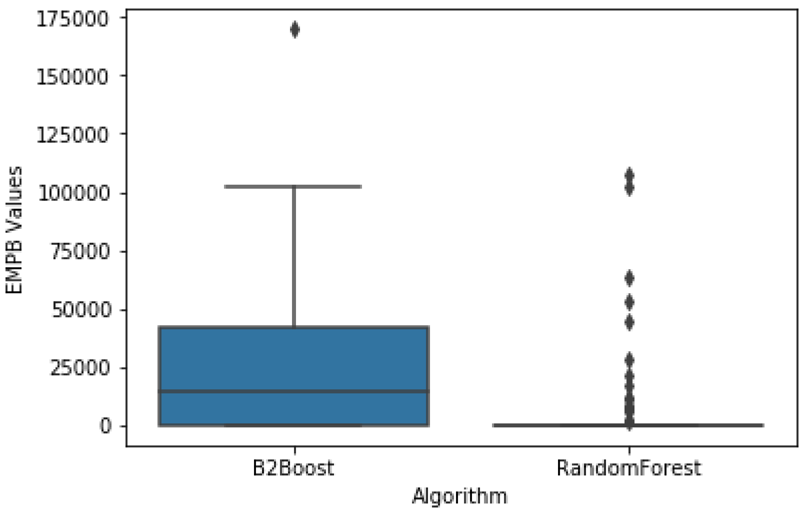

Figure 4: EMPB distributions of B2Boost vs. Random Forest

Results are compared by visualizing the 200 unique EMPB scores per algorithm into a boxplot in Figure 4. We clearly observe that the EMPB values reached by the B2Boost algorithm are much more often positive than the EMPB values of the random forest implementation. While the random forest has ‘accidental’ positive values (which may be quite high), the B2Boost algorithm reaches these values much more consistently, as this is the primary learning objective of the algorithm.

7. Managerial Implications

Our results indicate that firms which operate under specific conditions (i.e., relatively small customer base, highly variant customer values, and high incentive costs), may have extreme difficulties in establishing profitable retention campaigns in the B2B industry. The EMPB measure provides aid to managers who want to see which algorithms are most profitable in such situations, accounting for the deviations in customer values and associated deviant costs and benefits.

We demonstrate how our algorithm is the only one capable of generating substantial profits across all different situations, which is why we strongly advise CCP modellers to incorporate this algorithm when creating CCP models. It should, however, be supplemented with other (ensemble) learners as we demonstrate how random forest is capable of outperforming B2Boost in some alterations. Overall, practitioners are advised to evaluate different models, including B2Boost, with regard to our EMPB measure, as this gives a more fair representation of campaign profitability. Traditional measures, such as

EMPC and AUC, are demonstrated to positively evaluate unprofitable campaigns, which is highly undesirable.

Interestingly, the optimal campaign profit was each time created by a classifier which was not yet used in academic literature (i.e., VerbrakenBoost or B2Boost). This signifies that currently used algorithms are not yet developed to their true potential and that practitioners should always think about how to adapt certain algorithms (e.g., XGBoost) in order to have marketing campaigns which are as profitable as possible.

The outcome of our methodology, however, strongly depends on the customer values and offered incentive rates. This heavily influences both model training and model evaluation. An incorrect estimation of these parameters would have resulted in the incorrect selection of either VerbrakenBoost or B2Boost in our case study, dependent on the true customer values. As this incorrect selection could lead to suboptimal profits, it is extremely important to have correct estimates of these customer values. Many firms currently still operate without a clear view on the individual customer values or ideally offered incentives, and should have accurate estimates of these values before starting to deploy retention campaigns, as these may be unprofitable otherwise.

8. Conclusion and Future Research

This study contributes to literature in several ways. First, we develop an B2B-specific profit-driven metric, called the EMPB. The measure's framework is based on the framework of Óskarsdóttir et al. (2018), which is customized to the specifities of the business-to-business industry. This metric is capable of identifying issues with current practices in CCP modelling by incorporating the high variation in customer values associated to this industry. The main issue being the neglectence of these differences when deploying customer retention campaigns, by not incorporating the higher costs and benefits associated to correct and incorrect classifications associated to high value customers. This results into suboptimal profits or, as it the case for many tested situations, even unprofitable campaigns.

To overcome this, we propose a novel profit-maximizing classifier, based on extreme gradient boosting, which uses the gradient of the self-defined CLV-varying EMPB rather than the default log-likelihood loss function. Our results suggest that this B2Boost algorithm increases profits strongly. The relationship establishes itself over multiple experimental configurations, with the B2Boost algorithm being the only algorithm competitive in all parameter configurations. Only when overall discriminatory power regains its status as most important determinant, we observe one traditional learner (i.e., the random forest algorithm) to become competitive with B2Boost in terms of campaign profitability, and one cost-sensitive learner adapted to the EMPC metric as outlined by Verbraken et al. (2012).

The B2Boost algorithm consequently outperforms its base XGBoost variant with regard to EMPB performance. This means that incorporating EMPB in the objective function is a successful strategy for creating instance-sensitive churn models. Random forest, however, is successful in defeating B2Boost when customer values increase. It might be interesting to see how an adaptation to random forest, accounting for individual campaign benefits and costs in the objective function, may perform with regard to EMPB. If it would consequently outperform random forest, as B2Boost consequently outperforms XGBoost, this may result in an overall competitive CCP model.

Another addition to literature, is the first implementation of a cost-sensitive boosting algorithm based on the EMPC metric as defined by Verbraken et al. (2012). The algorithm is the best performer with regard to AUC, and proves to be highly competitive in terms of EMPB when discriminatory power is of main importance. Profit-driven boosting algorithms show clear potential for more profitable marketing campaigns and are an interesting avenue for future research in the field of business analytics.

To further establish our findings, further research is required. As we only have access to one dataset which showed such large variation, we solely test the profit-enhancing effect of our methodology onto one specific setting, thereby limiting its generalizability. However, results clearly indicate that our methodology can enhance retention campaign profitability, resulting in suboptimal campaigns when unaccounted for. Future research should deploy the algorithm on different datasets with large variation in customer values and see if current practices also lead to suboptimal outcomes there, as well as investigate how the degree of variation influences the magnitude of the profit gains the methodology creates.

Central to our framework is the incentive rate δ, which is assumed fixed across all customers. However, it is quite feasible that some customers are offered incentives with δ = 0.05, while others accept smaller offers at δ = 0.01. While this would still signify a value-dependent incentive, the incentive rate would vary per customer. This would alter our framework to work with individualized $\delta_i's$, rather than a fixed δ. This would further increase the individualized focus of the derived metrics and learners, and could theoretically further enhance campaign profitability. We did not test the potential impact of such alternations to the EMPB framework in this study, as the estimation of such desired incentive rates is difficult, with the need for a more uplift-based approach. Nevertheless, we believe that such alterations to the framework could prove interesting towards future research.

Despite its limitations, we feel confident that this study contributes to current literature, as the results are promising and would lead to possibly large gains in business-to-business industries, where data-driven retention management may not have an as established reputation as in other industries (e.g., telecommunication industry), despite its gains above common managerial heuristics being clearly established (Jahromi et al., 2014).

Acknowledgements

The authors are thankful to Dr. Michel Ballings for providing the data and to the three anonymous reviewers whose comments have helped significantly improve an earlier versions of this paper. The authors are also grateful to the Guest Editor of the Data Science & Decision Analytics Special Issue, Dr. Asil Oztekin, for his guidance and timely management of this manuscript.

Reference list

Ascarza, E. (2018). Retention futility: Targeting high-risk customers might be ineffective. *Journal of Marketing Research*, *55*(1), 80-98.

Ascarza, E., Neslin, S. A., Netzer, O., Anderson, Z., Fader, P. S., Gupta, S., ... & Schrift, R. (2018a). In pursuit of enhanced customer retention management: Review, key issues, and future directions. *Customer Needs and Solutions*, *5*(1), 65-81.

Ascarza, E., Netzer, O., & Hardie, B. G. (2018b). Some customers would rather leave without saying goodbye. *Marketing Science*, *37*(1), 54-77.

Baesens, B., Van Vlasselaer, V., & Verbeke, W. (2015). *Fraud analytics using descriptive, predictive, and social network techniques: a guide to data science for fraud detection*. John Wiley & Sons.

Ballings, M., & Van den Poel, D. (2012). Customer event history for churn prediction: How long is long enough?. *Expert Systems with Applications*, *39*(18), 13517-13522.

Baum, R. J., & Wally, S. (2003). Strategic decision speed and firm performance. *Strategic management journal*, *24*(11), 1107-1129.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

Buckinx, W., & Van den Poel, D. (2005). Customer base analysis: partial defection of behaviourally loyal clients in a non-contractual FMCG retail setting. *European Journal of Operational Research*, *164*(1), 252-268.

Burez, J., & Van den Poel, D. (2007). CRM at a pay-TV company: Using analytical models to reduce customer attrition by targeted marketing for subscription services. *Expert Systems with Applications*, *32*(2), 277-288.

Burez, J., & Van den Poel, D. (2008). Separating financial from commercial customer churn: A modeling step towards resolving the conflict between the sales and credit department. *Expert Systems with Applications*, *35*(1-2), 497-514.

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

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794). ACM.

Coussement, K., & Van den Poel, D. (2008). Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. *Expert systems with applications*, *34*(1), 313-327.

De Caigny, A., Coussement, K., & De Bock, K. W. (2018). A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. *European Journal of Operational Research*, *269*(2), 760-772.

De Caigny, A., Coussement, K., Verbeke, W., Idbenjra, K., & Phan, M. (2021). Uplift modeling and its implications for B2B customer churn prediction: A segmentation-based modeling approach. *Industrial Marketing Management*, *99*, 28-39.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, *7*(Jan), 1-30

Devriendt, F., Moldovan, D., & Verbeke, W. (2018). A literature survey and experimental evaluation of the state-of-the-art in uplift modeling: A stepping stone toward the development of prescriptive analytics. *Big data*, *6*(1), 13-41.

Devriendt, F., Berrevoets, J., & Verbeke, W. (2021). Why you should stop predicting customer churn and start using uplift models. *Information Sciences*, *548*, 497-515.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, *10*(7), 1895-1923.

Eriksson, K., & Vaghult, A. L. (2000). Customer Retention, Purchasing Behavior and Relationship Substance in Professional Services. *Industrial Marketing Management*, *29*(4), 363–372.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 1189–1232.

Gordini, N., & Veglio, V. (2017). Customers churn prediction and marketing retention strategies. An application of support vector machines based on the AUC parameter-selection technique in B2B e-commerce industry. *Industrial Marketing Management*, *62*, 100–107.

Höppner, S., Baesens, B., Verbeke, W., & Verdonck, T. (2021). Instance-Dependent Cost-Sensitive Learning for Detecting Transfer Fraud. *European Journal of Operational Research.*

Höppner, S., Stripling, E., Baesens, B., vanden Broucke, S., & Verdonck, T. (2020). Profit driven decision trees for churn prediction. *European Journal of Operational Research.*, *284*(3), 920-933.

Jahromi, A. T., Stakhovych, S., & Ewing, M. (2014). Managing B2B customer churn, retention and profitability. *Industrial Marketing Management*, *43*(7), 1258–1268.

Kalwani, M. U., & Narayandas, N. (1995). Long-Term Manufacturer-Supplier Relationships: Do They Pay off for Supplier Firms? *Journal of Marketing*, *59*(1), 1-16.

Larivière, B., & Van den Poel, D. (2004). Investigating the role of product features in preventing customer churn, by using survival analysis and choice modeling: The case of financial services. *Expert Systems with Applications*, *27*(2), 277-285.

Lemmens, A., & Gupta, S. (2020). Managing churn to maximize profits. *Marketing Science*, *39*(5), 956-973.

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In *Advances in neural information processing systems* (pp. 4765-4774).

Maldonado, S., Domínguez, G., Olaya, D., & Verbeke, W. (2021). Profit-driven churn prediction for the mutual fund industry: A multisegment approach. *Omega*, *100*, 102380.

Maldonado, S., López, J., & Vairetti, C. (2020). Profit-based churn prediction based on Minimax Probability Machines. *European Journal of Operational Research*, *284*(1), 273-284.

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

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

Rauyruen, P., & Miller, K. E. (2007). Relationship quality as a predictor of B2B customer loyalty. *Journal of Business Research*, *60*(1), 21–31.

Reinartz, W. J., & Kumar, V. (2003). The impact of customer relationship characteristics on profitable lifetime duration. *Journal of Marketing*, *67*(1), 77-99.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135-1144).

Schetgen, L., Bogaert, M., & Van den Poel, D. (2021). Predicting donation behavior: Acquisition modeling in the nonprofit sector using Facebook data. *Decision Support Systems*, *141*, 113446.

Shapley, L. S. (1953). A value for n-person games. *Contributions to the Theory of Games*, *2*(28), 307-317.

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.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267-288.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., ... & Altman, R. B. (2001). Missing value estimation methods for DNA microarrays. *Bioinformatics*, *17*(6), 520-525.

Tsai, C.-F., & Lu, Y.-H. (2009). Customer churn prediction by hybrid neural networks. *Expert Systems with Applications*, *36*(10), 12547–12553.

Vafeiadis, T., Diamantaras, K. I., Sarigiannidis, G., & Chatzisavvas, K. C. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory*, *55*, 1-9.

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., Mues, C., & Baesens, B. (2011). Building comprehensible customer churn prediction models with advanced rule induction techniques. *Expert Systems with Applications*, *38*(3), 2354-2364.

Verbraken, T., Verbeke, W., & Baesens, B. (2012). A novel profit maximizing metric for measuring classification performance of customer churn prediction models. I*EEE transactions on knowledge and data engineering*, *25*(5), 961-973.

Wang, C., Deng, C., & Wang, S. (2020). Imbalance-XGBoost: leveraging weighted and focal losses for binary label-imbalanced classification with XGBoost. *Pattern Recognition Letters*, *136*, 190-197.