Data Mining and Knowledge Discovery (2023) 37:736–766
https://doi.org/10.1007/s10618-023-00917-9

Exploring uplift modeling with high class imbalance

Otto Nyberg[1] · **Arto Klami**[1]

Received: 2 June 2022 / Accepted: 2 January 2023 / Published online: 23 January 2023

Abstract
Uplift modeling refers to individual level causal inference. Existing research on the topic ignores one prevalent and important aspect: high class imbalance. For instance in online environments uplift modeling is used to optimally target ads and discounts, but very few users ever end up clicking an ad or buying. One common approach to deal with imbalance in classification is by undersampling the dataset. In this work, we show how undersampling can be extended to uplift modeling. We propose four undersampling methods for uplift modeling. We compare the proposed methods empirically and show when some methods have a tendency to break down. One key observation is that accounting for the imbalance is particularly important for uplift random forests, which explains the poor performance of the model in earlier works. Undersampling is also crucial for class-variable transformation based models.

Keywords High class imbalance · Undersampling · Uplift modeling · Heterogeneous treatment effect

1 Introduction

High class imbalance is prevalent in e-commerce where conversion rates are typically in the range of 0.1–5% (Diemert et al. 2018; Richardson et al. 2007). The rate depends on whether the conversion event is e.g. a click, a visit, or a purchase, with the more valuable events (the purchases) being at the lower end of this spectrum. High class imbalance makes modeling difficult as observations contribute to a cost function in proportion to their number, resulting in the cost function being easily minimized

Responsible editor: Johannes Fürnkranz

✉ Otto Nyberg
otto.nyberg@helsinki.fi

Arto Klami
arto.klami@helsinki.fi

[1] Department of Computer Science, University of Helsinki, Pietari Kalmin katu 5, 00014 Helsinki, Finland

when an algorithm largely ignores the minority class. A common way to deal with this problem in classification tasks is by undersampling where observations from the majority class are dropped to better balance the number of positive and negative observations. However, models trained on undersampled data are not well-calibrated in the original task. Usually this is not a problem as a simple threshold is enough to decide which class an observation belongs to, and if needed the model can be calibrated afterwards using a calibration method (e.g. Zadrozny and Elkan 2002). However, these techniques do not directly translate to uplift modeling.

Uplift modeling is the art of modeling the causal effect of some treatment on individual observations (Rzepakowski and Jaroszewicz 2010). More formally it is defined as the difference between two probabilities

$$\tau(x) = p(y = 1|x, do(t = 1)) - p(y = 1|x, do(t = 0)) \tag{1}$$

where x are the features of an observation, y is the class-label, t is the treatment label where $t = 1$ indicates treatment and $t = 0$ no treatment, and $do(.)$ is the do-operator (Pearl 2009).

As uplift is the difference between two probabilities, we need to be more careful in accounting for the distortion of the probabilities caused by undersampling. Recently Nyberg et al. (2021) proposed a method relying on stratified undersampling for uplift estimation, but that solution relies on simplifying assumptions that are not always valid. This paper explores different undersampling and calibration methods in the context of uplift modeling, and in particular, proposes the first general method applicable for improving uplift estimates for tasks with high class imbalance that makes no simplifying assumptions on the dataset (in addition to the assumptions made by uplift modeling). We present four undersampling methods (three of which are novel) and three new calibration methods (one of which is novel) with theoretical foundations, and empirically evaluate them on the largest available uplift datasets that exhibit high class imbalance as well as on one synthetic dataset. We also demonstrate how the results depend on data characteristics, such as the amount of imbalance and the base conversion rate, and illustrate when specific methods can fail.

The main finding is that high class imbalance can be effectively addressed with undersampling. All tested uplift models improved when the class imbalance was accounted for when the datasets were large enough, and for some methods the effect is dramatic. Methods based on the class-variable transformation (Jaskowski and Jaroszewicz 2012; Lai 2006) do not work at all without undersampling but become competitive when the imbalance is corrected for, and for uplift random forests (Guelman et al. 2015) we observe a 50–60% improvement in the standard performance metric with undersampling. In some previous comparisons, such as Fernández-Loría and Provost (2022), Semenova and Temirkaeva (2019) and Belbahri et al. (2021), random forest -based methods have performed poorly and we postulate that accounting for the imbalance by undersampling would have changed some of the conclusions of these works. Another interesting observation is that we were able to reliably estimate uplift when there were as few as 200 positive observations in the minority class. Even though the exact number required naturally depends on the specific case, our result is

encouraging in terms of practical applicability of uplift modeling for industries dealing with problems with extremely small conversion rates and moderate amounts of data.

2 Related work

We build on two streams of previous work: class imbalance and uplift models. This section provides the necessary background on both aspects for understanding the rest of the paper.

2.1 Class imbalance

The term "imbalance" has been used to refer to multiple different aspects in uplift modeling: Olaya et al. (2020) used it to describe an imbalance in treatment policy usually referred to as "confounding effects" (Austin 2011), and Betlei et al. (2018) used it to describe a setting where there is a large difference in the number of treated and untreated observations. In contrast, this paper deals with the imbalance in class labels (the outcomes), following the terminology of Nyberg et al. (2021). This problem has been thoroughly studied in the context of classification and is commonly referred to as "class imbalance" (Kaur et al. 2019), but remains understudied in uplift modeling.

There are two main techniques for dealing with high class imbalance in classification: weighting and sampling, including oversamping, undersampling, and synthetic sampling. Moreover, oversampling and undersampling are sometimes combined (Chawla et al. 2002). In weighting, the minority class observations are given a larger weight in the cost function to ensure that the algorithm will account for them appropriately, whereas in oversampling the observations of the minority class are resampled so that there are multiple copies of them. Synthetic sampling generates new unique observations based on the properties of existing observations (Chawla et al. 2002). Undersampling, in turn, refers to techniques that discard some of the observations in the majority class(es).

Even though both weighting and sampling could potentially be used in the context of uplift modeling, we specifically focus on undersampling because it maps so elegantly to the typical use cases. Especially in e-commerce, one can easily collect a large number of negative observations and e.g. datasets used in this paper contain more than ten million observations. By undersampling the negative observations we can reduce the size of the training dataset and hence also the computation time. In contrast, oversampling in these cases would result in extremely large training sets.

2.2 Uplift models

In this work we consider only learning scenarios where the data has been collected in a randomized trial so that the treated and untreated observations come from the same underlying distribution $p(x)$ and the choice of the treatment has been made independently of x. With this assumption, the do-notation in Eq. (1) simplifies to conditioning on t. There are also models that do not require this assumption, that

work on observational data (e.g. Johansson et al. 2016) or on data where the treatment policy is known (e.g. Austin 2011), but these include other assumptions that are often hard to verify and including these in the experiments would complicate matters without bringing additional value.

Uplift modeling is an active research topic and numerous different principles and practical models have been proposed; see Gubela et al. (2020) for a recent overview. Our interest is in studying specifically the undersampling process as a means of accounting for high class imbalance, largely in a method-agnostic manner. Basic understanding of the modeling approaches is needed e.g. to understand which undersampling methods are compatible with what models, but we leave out the technical details of the learning algorithms as they are not central for our work. We will evaluate the different undersampling approaches in the context of a few models, selected as representative examples of popular methods belonging to different families, and these particular methods are explained briefly next.

The double classifier by Radcliffe and Surry (1999) is a classic model motivated directly by Eq. (1). It is a type of T-learner (Künzel et al. 2019) as it uses two classification models, one to model $p(y|x, do(t = 1))$ and another to model $p(y|x, do(t = 0))$ and simply estimates the uplift by computing the difference between the two probabilities. We use the double classifier with logistic regression as base classifier, denoting the model by DC-LR. Even though DC-LR has historically received little attention, in part due to critique provided by Radcliffe and Surry (1999) and Guelman et al. (2015), it performed best in a relatively recent comparison by Semenova and Temirkaeva (2019).

The class-variable transformation (CVT) was proposed by Jaskowski and Jaroszewicz (2012) and Lai (2006). In CVT, the outcome variable y and treatment label t are used to create a new variable z so that $z_i = 1$ when $y_i = 1$ and $t_i = 1$, or when $y_i = 0$ and $t_i = 0$. Otherwise $z_i = 0$. With this transformation, uplift becomes $\tau(x) = 2 \cdot p(z|x) - 1$, i.e. the uplift problem is transformed into a classification problem. This way the uplift problem can be solved with one classifier rather than two. CVT with logistic regression (CVT-LR) performed best in the comparison of Nyberg et al. (2021) on one of the datasets we will use in our evaluation.

A somewhat related approach is the revert label (RL) proposed by Athey and Imbens (2015). A similar class-transformation is performed so that the new variable is defined as

$$r_i = \frac{t_i \cdot y_i}{\pi(x_i)} - (1 - t_i)\frac{y_i}{1 - \pi(x_i)} \tag{2}$$

where $\pi(x_i)$ is the propensity score (the probability that an observation of type x_i was treated). When the training data is collected in a randomized controlled trial, this is assumed to be a constant value for all x_i. The big difference between CVT and RL is that while the former transforms the learning problem into a classification problem, the latter transforms it into a regression problem. The r in RL takes at least three values and the uplift is the expectation of r. As the expectation is continuous, this is best treated as a regression problem. This is also the same formulation later proposed by Rudaś and Jaroszewicz (2018). As a practical method building on the RL concept,

a neural network that minimizes the mean-squared error between the revert label and the output similarly to Belbahri et al. (2021) is included in the experiments. It can be shown that this formulation is equivalent to the one presented by Gutierrez and Gérardy (2017) where they showed that it is possible to minimize the mean-squared error between the uplift estimate and the actual unobservable uplift.

Another interesting family of models are uplift random forests. The forest proposed by Guelman et al. (2015) was included in the experiments instead of e.g. the causal random forest by (Wager and Athey 2018) as the former is better suited for binary class labels. In contrast to all the previous models, trees and forests try to directly model what makes an observation susceptible to influence. This is accomplished by applying a splitting criterion that maximizes heterogeneity in the resulting leafs, i.e. that results in leafs where the treated observations have as different positive rate as possible from untreated observations (given some constraints on leaf size etc.). Despite their recent popularity, the empirical evidence has not been entirely convincing.

The experiments in this work consider only the four models described above, each representing a common family of uplift models. The undersampling methods can also be used with various other uplift models, such as the S- and X-learners (Künzel et al. 2019) and the model proposed by Lo (2002). These models would be compatible with (some of) the proposed undersampling methods, but we leave their evaluation as future work to keep the empirical experiments manageable.

3 Methods

The main goal of our work is to establish best practices for addressing high class imbalance in uplift modeling problems using undersampling as the technical solution. As mentioned earlier, undersampling has a long history in classification problems but our recent preliminary work Nyberg et al. (2021) remains thus far the only investigation into the problem in uplift modeling. In this section we further develop the initial ideas in that work to a comprehensive formulation. We start by defining the basic concepts and notation used for addressing probabilities estimated from undersampled data, and present four alternative undersampling strategies for uplift problems, three of which are novel and one of which was previously presented in Nyberg et al. (2021). The methods differ in terms of which observations are discarded and at what rate. We then explain three methods for calibration of uplift estimates in Sect. 3.3.

3.1 The undersampling process

Undersampling refers to dropping randomly selected observations of the majority class to better balance the ratio between treated and untreated observations. For all of the proposed methods, we always keep all of the positive observations and drop some of the negative observations and all formulas in this paper are formulated assuming that $y = 0$ is the majority class. This way, the positive class $y = 1$ will have larger prevalence in the undersampled data. We define undersampling using a factor k so that

$$p^*(y=1) = k \cdot p(y=1) \tag{3}$$

where $p(y=1)$ denotes the probability of positive observations before undersampling and $p^*(y=1)$ is the corresponding probability after undersampling.[1] Here $p(y=1)$ is estimated from data and equals the fraction of positive observations. That is, k tells how much the probability of positive observations increases because of the undersampling. To improve the balance we need to have $k \geq 1$ (with equality corresponding to no undersampling) but additionally the factor has a natural upper bound $k < \frac{1}{p(y=1)}$. This corresponds to dropping all negative observations.

In practical terms, the undersampling is carried out by looping over the negative observations and independently *keeping* each one with the probability

$$s = \frac{1/k - p(y=1)}{1 - p(y=1)}. \tag{4}$$

We have chosen to formulate the undersampling process using the factor k, rather than the probability s, for several reasons: (a) it is directly interpretable as the change of probability (Eq. (3)), (b) it leads to more clear and concise equations for the stratified undersampling procedure introduced later in Sect. 3.2.3, and (c) it leads naturally to one calibration method (Sect. 3.3.2).

The factor k defines the average change. The uplift as defined in Eq. (1), however, depends on the conditional probabilities. Their distortion is characterized by

$$p^*(y=1|x) = \frac{p(y=1|x)}{p(y=1|x) + s \cdot (1 - p(y=1|x))}. \tag{5}$$

This follows directly from the undersampling process that reduces the proportion of negative observations, indicated by $(1 - p(y=1|x))$, by a factor of s while keeping all positive observations. Since s is in the denominator, this distortion is non-linear in terms of the probability $p(y=1|x)$. This means that the quantities needed for estimating uplift change because of the undersampling and this change needs to be accounted for to obtain unbiased estimates. When $p(y=1|x)$ is small the relationship is approximately linear and corresponds to multiplication with k as in the average case, but for larger probabilities this does not hold.

Figure 1 illustrates the effect of undersampling. Here we assume the probabilities $p(y=1|x)$ are estimated using maximum likelihood (ratio of positive and negative observations) in local neighborhoods of x, with the square indicating one such neighborhood. When we set $k = 2$ for this data with high class imbalance ($p(y=1) = 0.0083$), we keep negative observations with probability $s = 0.4958$ (Eq. (4)). Since the true probability is small, we have $s \approx \frac{1}{k}$. In the local neighborhood indicated by the square, the proportion of positive observations approximately doubles when dropping approximately half of the negative observations. However, if our data had a local neighborhood with high probability $p(y=1|x)$ this would not be the case. For example, for $p(y=1|x) = 0.3$ we would get $p^*(y=1|x) \approx 0.46$,

[1] We follow this convention throughout the paper; all quantities marked with an asterisk * refer to the undersampled case.

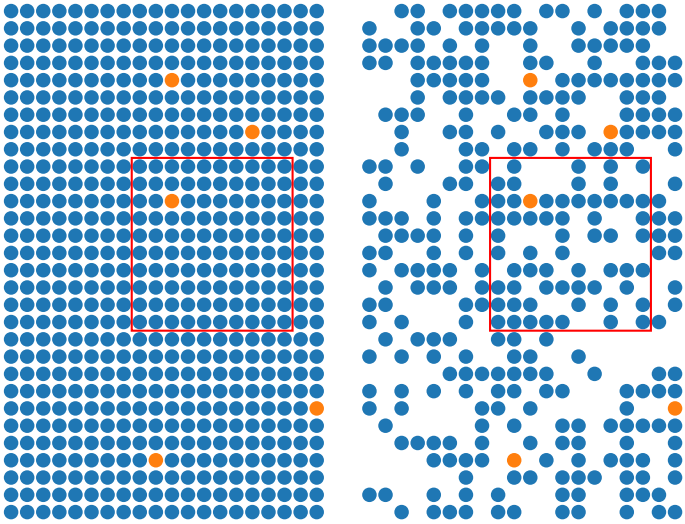

Fig. 1 Effect of undersampling. With high class imbalance very few observations are positive (orange). To improve the situation we can double the average rate of positive examples with factor $k = 2$, corresponding to dropping slightly more than half of the negative observations (blue). This changes the conditional probabilities $p(y = 1|x)$ in a non-linear way. If we estimate them using a local neighborhood (red rectangle) then the change depends on the original number of positive and negative observations in the neighborhood as characterized by Eq. (5)); see text for examples (Color figure online)

corresponding to slightly more than 50% increase in the positive rate. When k is large, this non-linear distortion becomes significant also for smaller probabilities.

3.2 Undersampling for uplift modeling

The equations above hold for any undersampling method that drops negative observations. Next, we present four different undersampling methods that can be used for improving class balance in uplift modeling. The methods differ in terms of what rate treated and untreated negative observations are discarded. To indicate this we introduce additional notation where the undersampling parameters k and s are replaced by $k_{t=1}$, $k_{t=0}$, $s_{t=1}$, and $s_{t=0}$ as needed to indicate when the undersampling is applied only to treated or untreated observations.

3.2.1 Undersampling for classification

The double classifier method (Radcliffe and Surry 1999) directly trains two models for treated and untreated observations separately, and hence standard undersampling for classification can be used to improve accuracy of these models independently. That is, we can separately perform undersampling for the treated and untreated samples, always dropping only negative observations. More formally, this is defined as

$$
\begin{aligned}
p^*(y = 1|t = 1) &= k_{t=1} \cdot p(y = 1|t = 1) \\
p^*(y = 1|t = 0) &= k_{t=0} \cdot p(y = 1|t = 0)
\end{aligned}
\tag{6}
$$

where typically $k_{t=1} \neq k_{t=0}$ since the positive rates and hence the severity of class imbalance differs in the treated and untreated observations. The factors $k_{t=1}$ and $k_{t=0}$ are chosen independently using hold-out validation on the validation set (see Sect. 4.2) and a measure of classification performance, e.g. AUC-ROC. The model estimating $p^*(y = 1|t = 1)$ is evaluated on the treated observations in the validation set and the model estimating $p^*(y = 1|t = 0)$ on the untreated observations in the validation set.

As the undersampling process distorted the probabilities, the scores output by the classifiers will not correspond to true probabilities. This distortion needs to be corrected. For classifiers this can be done by *calibration*, the process of mapping scores to empirical estimates, with several practical methods like isotonic regression (Zadrozny and Elkan 2002), Bayesian binning into quantiles (Naeini et al. 2015), or platt-scaling (Platt 1999) available. We used isotonic regression. After calibration, we can estimate the uplift using Eq. (1) directly. The obvious drawback of this conceptually simple strategy is that it is only compatible with the double classifier approach.

3.2.2 Naive undersampling

This is a method where negative observations are dropped with equal probability regardless of whether they are treated or untreated. This corresponds to *naively* doing undersampling as it has been done for classification without accounting for the differences between treated and untreated observations.

The treated and untreated observations typically have different average positive rate resulting in different severity of class imbalance. In addition, as the treated and untreated observations typically come from different underlying distributions, the optimal undersampling rate will differ. Naive undersampling ignores this and is implicitly based on the assumption that the underlying distributions and the severity of class imbalance is similar in both treated and untreated observations. We define it using

$$
p^*(y = 1) = k \cdot p(y = 1), \tag{7}
$$

and the undersampling is carried out using a single s derived using Eq. (4). The parameter k is found using hold-out validation. In contrast to the previous method, we now need to use an uplift evaluation metric for selecting the optimal parameter. We used AUUC that is also used as the main evaluation metric for uplift methods (see Sect. 4.1).

This approach is conceptually simple, compatible with all uplift models and only requires choosing one undersampling factor. However, it is biased whenever $p(y = 1|t = 1) \neq p(y = 1|t = 0)$, as will be explained in more detail in the next subsection. Nevertheless, it can still improve the performance in some cases as will be shown in Sect. 4.

3.2.3 Stratified undersampling

Stratified undersampling was presented in our preliminary work Nyberg et al. (2021). Similar to naive undersampling, it drops both treated and untreated majority class observations using one *common* factor k so that

$$\begin{aligned} p^*(y=1|t=1) &= k \cdot p(y=1|t=1) \\ p^*(y=1|t=0) &= k \cdot p(y=1|t=0). \end{aligned} \tag{8}$$

In contrast to the naive undersampling, however, we now use different s for the two groups: We compute $s_{t=1}$ and $s_{t=0}$ separately for the two populations using Eq. (4), now using the group-conditional probabilities $p(y=1|t=1)$ and $p(y=1|t=0)$ instead of the overall rate.

As indicated by Eq. (5), the undersampling process changes the probabilities in a non-linear manner. However, if both $p(y=1|x,t=1)$ and $p(y=1|x,t=0)$ are sufficiently small for all x, then the change is approximately linear and we have $p^*(y=1|x,t=1) \approx k \cdot p(y=1|x,t=1)$ and $p^*(y=1|x,t=0) \approx k \cdot p(y=1|x,t=0)$. Then the uplift $\tau(x)$ will also be approximately linear in k so that $\tau^*(x) \approx k \cdot \tau(x)$. Nyberg et al. (2021) explicitly relied on this linearity assumption.

In the rare case when $p(y=1|t=1) = p(y=1|t=0)$, stratified undersampling is equivalent to naive undersampling. To better understand the difference when this is not the case, we can convert the common s used in naive undersampling back to two separate factors $k_{t=1}$ and $k_{t=0}$ (using inverse of Eq. (4)). This means the naive undersampling corresponds to using different undersampling factors despite using a common s, and consequently we no longer have clear linear relation for the uplifts as both terms are modified by different factors.

3.2.4 Split undersampling

The most comprehensive undersampling method we consider is split undersampling which undersamples the treated and untreated observations with different factors $k_{t=1}$ and $k_{t=0}$. The equations are then

$$\begin{aligned} p^*(y=1|t=1) &= k_{t=1} \cdot p(y=1|t=1) \\ p^*(y=1|t=0) &= k_{t=0} \cdot p(y=1|t=0), \end{aligned} \tag{9}$$

This is equivalent to the equations of *undersampling for classification*, but now factors $k_{t=1}$ and $k_{t=0}$ are chosen *jointly*. That is, an uplift model is now trained on the undersampled dataset, and the *combination* of factors $k_{t=1}$ and $k_{t=0}$ is evaluated using hold-out validation and an uplift metric. We again used AUUC as the criterion.

This approach is general in the sense that it puts no assumptions on the positive rates or conditional probabilities present in the data. It is also general in that it includes both stratified and naive undersampling as special cases. We obtain the former when $k_{t=1} = k_{t=0}$ and the latter when $k_{t=1} = \frac{1}{s_{t=0} \cdot (1-p(y=1|t=1)) + p(y=1|t=1)}$. These equalities are the result of these two methods aiming to control the distortion in probabilities

so that they are easily manageable. In contrast, split undersampling requires no such dependence between $k_{t=1}$ and $k_{t=0}$. As the treated and untreated observations usually have different positive rates, and hence severity of class imbalance, the optimal undersampling parameters to deal with the class imbalance will also usually be different. Hence split undersampling has the potential to find undersampling parameters that better fit the problem.

As a consequence of freely choosing $k_{t=1}$ and $k_{t=0}$, the uplift estimates produced by the model trained on the undersampled data will no longer produce well-ranked predictions. The predictions might even have the wrong sign. This will need special attention later in the calibration step. We also note that even though we only consider binary uplift problems in this work, the split undersampling method directly generalizes to multi-class uplift problems (Papangelou 2021) and provides the first solution for addressing class imbalance for these. This will be elaborated on later in Sect. 3.3.3 when discussing calibration of split undersampling estimates.

3.3 Calibration methods

All of the undersampling methods distort the probabilities in a non-linear way (Eq. (5)). When rank alone is sufficient for the intended use (Verbeke et al. 2012; Devriendt et al. 2021; Gubela et al. 2020), both naive and stratified undersampling will produce adequate results without calibration. However, this is not the case for undersampling for classification and split undersampling. These two methods distort the probabilities with and without treatment so that the difference between these, the uplift estimates, will not be ranked in a meaningful way. This is further dealt with in Sect. 3.3.3.

Sometimes calibrated uplift estimates are needed for downstream processing. E.g. in the case of using *free delivery* as treatment in an online store, both a calibrated uplift estimate $\tau(x)$ and a calibrated probability estimate for $p(y = 1|x, t = 1)$ are needed for optimal targeting. Then the treatment should only be applied if $\tau(x) \cdot profit \geq cost \cdot p(y = 1|x, t = 1)$, where $profit$ refers to the profit of the sale excluding delivery costs and $cost$ refers to the cost of delivery. This is discussed in more detail by Haupt and Lessmann (2020).

In the experiments, we calibrated all uplift estimates. With *undersampling for classification* the calibration is applied after model training but before combining the two models to an uplift model. For the rest, the calibration can be performed as a separate post-processing step using the methods described next.

3.3.1 Isotonic regression and τ-isotonic regression

Isotonic regression produces a function $g(s)$ that minimizes $\sum_i (g(s_i) - y_i)^2$ under a monotonicity constraint so that $g(s_i) \leq g(s_j)$ for $s_i < s_j$. When y is binary and s_i and s_j are scores outputted by some classification algorithm, this becomes a calibration algorithm. This is commonly used as a post-processing step to transform the outputs to well-calibrated probabilities (Zadrozny and Elkan 2002). Isotonic regression is in this form used in this paper in undersampling for classification (Sect. 3.2.1).

Nyberg et al. (2021) extended calibration with isotonic regression to uplift modeling. We call this τ-*isotonic regression* to separate it from isotonic regression for calibration. In the revert-label formulation $\mathbb{E}(r|x) = \tau(x)$ (Athey and Imbens 2015), hence by replacing y_i with the revert-label r_i, $g(s)$ becomes an estimator for uplift. Using τ-isotonic regression will ensure that the uplift estimates will match empirical estimates. In the experiments, this calibration method is used together with naive undersampling to correct for the distortion introduced by undersampling. The method itself places no requirements on the uplift model or the scores, but it enforces monotonicity in the estimates.

3.3.2 Renormalization

Renormalization is a calibration method specifically for calibrating estimates obtained with stratified undersampling. For that case both of the probabilities $p^*(y = 1|x, t = 1)$ and $p^*(y = 1|x, t = 0)$ estimated from undersampled data are approximately k times as large as the actual probabilities, and consequently so are the uplift estimates $\tau^*(x)$. This distortion can be corrected easily with division by k, thus renormalizing the estimate. This correction is only applicable for stratified undersampling as it relies on use of equal k factors, and as explained in detail by Nyberg et al. (2021) it is accurate only when the conversion rates are small. For larger rates the distortions are no longer sufficiently linear.

3.3.3 Local neighborhood calibration

Local neighborhood calibration uses two input probabilities to produce one calibrated uplift estimate. Using two input probabilities enables the calibration method to change the rank of uplift estimates between observations. This is something that cannot be accomplished by τ-isotonic regression or renormalization and it is necessary to correct for the distortions introduced by split undersampling. This calibration method also extends to multi-class problems. Denoting the probability that an observation of class j is kept after undersampling by $s_{y=j,t}$, the probability of observations of that class in some local neighborhood of x is

$$p^*(y = j|x, t) = \frac{s_{y=j,t} \cdot p(y = j|x, t)}{\sum_{l \in J} s_{y=l,t} \cdot p(y = l|x, t)}. \tag{10}$$

No assumptions are made as to whether there is one class that is in majority in the multi-class case, hence the class for $s_{y,t}$ is explicitly specified. Elsewhere in the paper the majority class is assumed to be the negative class and the y is dropped from the notation. By rearranging, the original probabilities before undersampling can be calculated by solving the system of equations

$$\begin{cases} p(y = j|x, t) = \frac{p^*(y=j|x,t)}{s_{y=j,t} \cdot (1 - p^*(y=j|x,t))} \cdot \sum_{l \in J, l \neq j} s_{y=l,t} \cdot p(y = l|x, t) \\ \sum_{l \in J} p(y = l|x, t) = 1. \end{cases} \tag{11}$$

Setting $J = \{0, 1\}$ corresponds to the case with a binary class variable. Then the notation can be simplified so that $j = 0 \Rightarrow s_{y=j,t} = s_t$ (Eq. (4)) and $j = 1 \Rightarrow s_{y=j,t} = 1$ as all of the positive observations are kept. Solving the system of equations then results in the maximum likelihood estimate (see "Appendix 1" for details)

$$p(y = 1|x, t) = \frac{s_t \cdot p^*(y = 1|x, t)}{1 - p^*(y = 1|x, t) \cdot (1 - s_t)}. \tag{12}$$

Assuming that the output of a model approximates $p^*(y = 1|x, t)$, the distortion introduced by undersampling can be corrected using the equation above. Note that the equations cover the calibration of one probability. This calibration needs to be done separately for the conversion probability with $t = 1$ and $t = 0$ with appropriate parameters. Only then can a corrected uplift estimate be calculated as the difference between these two.

4 Experiments and results

We illustrate and evaluate the new methods using three experiments where each experiment addresses a separate research question. Before presenting the experiments and the results, we describe the metrics and datasets.

4.1 Metrics and hold-out validation

The main evaluation metric used is the *area under the uplift curve (AUUC)* (Jaskowski and Jaroszewicz 2012) commonly used for evaluating uplift models. It measures the expected increase in positive rate due to targeting treatments rather than randomizing them, averaged over all treatment rates. Hence it is the expected increase in positive rate due to your model if you have no preference on treatment rate. AUUC is a general purpose metric for goodness of fit and is particularly suitable in academic contexts where the use case is undefined. The absolute values of AUUC are often small even when the relative improvements are large. For legibility, we will report results as mAUUC ($1000 \cdot AUUC$) but will additionally clarify in the text the relative improvement.

As AUUC depends only on the rank of the observations and not the magnitude, we additionally used the *expected uplift calibration error (EUCE)* (Nyberg et al. 2021) as a metric to estimate how well the predictions match empirical rates. To estimate EUCE, all observations are first sorted based on the uplift predictions into m bins so that each bin contains $C = N/m$ observations with the first bin containing the observations with smallest predictions etc. N is the number of observations. For each bin j, the empirical uplift is estimated as

$$b_j = \frac{\sum y_{i,t=1}}{N_{j,t=1}} - \frac{\sum y_{i,t=0}}{N_{j,t=0}} \tag{13}$$

where the sum is over all observations i in bin j. $N_{j,t=1}$ and $N_{j,t=0}$ refer to the number of treated and untreated observations in bin j, whereas $y_{i,t=1}$ and $y_{i,t=0}$ refer to the labels of the treated and untreated observations in the bin. Further, denoting the average uplift estimate for the observations in one bin $u_j = \frac{\sum \tau(x_i)}{C}$, EUCE can be expressed as

$$EUCE = \frac{1}{m} \sum_j |u_j - b_j|. \tag{14}$$

Following the original formulation, the number of bins m used for estimating EUCE was set to 100 in the experiments.

For all methods we select the optimal undersampling factors k (or $k_{t=1}$ and $k_{t=0}$) using simple hold-out validation. The datasets were randomly split into training sets (50%), validation sets (25%), and testing sets (25%), where the training data is used for learning the models, the validation data for selecting the undersampling factor, and the final metrics are evaluated on the test data. This setup was deemed sufficient for the tree largest datasets. For the two smaller ones, this procedure was repeated 10 times so that the observations were randomly re-asssigned to the sets for each run, and the testing set metrics were averaged for the result tables.

In the experiments the values for $k_{t=1}$ and $k_{t=0}$ tested were {1, 2, 4, 8, 16, 32, 64, 128, 256}. In addition, in Experiment 1 the $k_{t=0}$ values included all values for $k_{t=1} \cdot 1.55$. These choice were made because this includes stratified undersampling and cases where $p^*(y = 1|t = 1) = p^*(y = 1|t = 0)$. The second choice captures the intuition that if both the treated and untreated observations have the same conversion rate, the issues caused by undersampling should be of similar magnitude in both cases. The best parameters were chosen using AUC-ROC on the validation set for classic undersampling and AUUC on the validation set for all other models.

4.2 Datasets

Evaluating methods for correcting class imbalance requires data that exhibits high class imbalance and is sufficiently large for evaluating the uplift reliably. We evaluate the methods on the three largest publicly available datasets, and additionally on two smaller datasets to illustrate the limitations of the methods. Details on the datasets are provided in Table 1.

We used `Criteo-uplift 2` (Diemert et al. 2018) as the main data, using it in all experiments. The data originally has 13,979,592 observations, but for Experiments 1 and 2 we downsampled the data so that the ratio between treated and untreated observations was 1:1.

We used `Criteo-uplift 1` (Diemert et al. 2018) for the main experiment. The dataset originally has 25,309,482 observations, but was downsampled for Experiment 1 so that the ratio between treated and untreated observations was 1:1. This data combines multiple ad campaigns (randomized experiments) with varying conversion rates, and hence a model that is able to identify which campaign an observation received can obtain high uplift but such a model would not be useful for future campaigns. For

Table 1 Statistics of the datasets as used in the experiments

Dataset	Observations	$p(y = 1\|t = 1)$ (%)	$p(y = 1\|t = 0)$ (%)
`Criteo-uplift 1`	7,801,310	0.239	0.174
`Criteo-uplift 2`	4,193,874	0.309	0.194
`Zhao`	642,531	0.51	0.29
`Starbucks`	126,184	1.68	0.73
`Hillstrom`	42,613	1.25	0.57

With the exception of Starbucks, the datasets were modified from the original releases for the purposes of our experiments as explained in the main text. All datasets have an approximate 1:1 ratio between the number of treated and untreated observations

the purpose of comparing different undersampling approaches (or even uplift models) this property is not important, but need to be kept in mind when interpreting the absolute uplift estimates.[2] Both datasets by `Criteo` comprise of click-stream data for online marketing and they hence have high class imbalance as expected in this common use case. Even though both datasets are provided by the same organization, they are two independent datasets.

We used the synthetic dataset by `Zhao` (Zhao et al. 2022) for the main experiment.[3] The dataset originally has 1,000,000 observations, but it does not have high class imbalance. We modified the data by dropping positive observations resulting in a dataset with high class imbalance. The resulting data has 642,531 observations, which of 321,194 are treated observations and 321,337 untreated observations.

The `Starbucks` dataset (Rössler et al. 2021) was used for the main experiment as is. It naturally exhibits high class imbalance.

The `Hillstrom` dataset (Radcliffe 2008) was used for the main experiment. The dataset originally has 64,000 observations, but we discarded the observations with treatment label *Womens E-Mail*. We used the treatment *Mens E-Mail* for $t = 1$ and the *No E-Mail* for $t = 0$. We used the conversion label as it exhibits high class imbalance.

4.3 Experiment 1: comparing methods and models

Our main experiment evaluates and quantifies the effect of using different undersampling approaches together with four different uplift models. As models we use:

1. DC-LR: double-classifier with logistic regression as the base classifier (Radcliffe and Surry 1999), using the `scikit-learn` (Pedregosa et al. 2011) implementation for logistic regrssion with default parameters;
2. CVT-LR: class-variable transformation with logistic regression (Jaskowski and Jaroszewicz 2012), again using `scikit-learn` for the base classifier;
3. Uplift RF: Uplift random forest by Guelman et al. (2015), using Kullback-Leibler divergence as the split criterion; and

[2] For more information, see https://ailab.criteo.com/criteo-uplift-prediction-dataset/.

[3] Dataset available at https://doi.org/10.5281/zenodo.3653141.

Table 2 Theoretically sound combinations of models, undersampling methods, and calibration methods (checkmarks)

	DC-LR	CVT-LR	Uplift RF	Uplift NN
Baseline (no undersampling)	✓	✓	✓	✓
Undersampling for classification	✓			
Naive und. + renormalization				
Naive und. + τ-isotonic regression	✓	✓	✓	✓
Naive und. + local neighborhood	(✓)	(✓)	(✓)	(✓)
Stratified und. + renormalization	✓	✓	✓	✓
Stratified und. + τ-isotonic regression	(✓)	(✓)	(✓)	(✓)
Stratified und. + local neighborhood	(✓)		(✓)	
Split und. + renormalization				
Split und. + τ-isotonic regression	(✓)		(✓)	
Split und. + local neighborhood	✓		✓	

Combinations not studied experimentally are in parentheses

4. Uplift NN: Neural network with four hidden layers, each with 128 units, optimized to minimize mean-squared error against revert label targets similar to Belbahri et al. (2020).

The exact implementations and details are available as open software at https://github.com/Trinli/uplift_modeling.

We combine all four uplift models with three undersampling approaches and the most appropriate choices of calibration methods:

1. Baseline with no undersampling
2. Naive undersampling with τ-isotonic regression for calibration
3. Stratified undersampling with renormalization for calibration

In addition, we evaluate the results for the following combinations that can only be applied in the context of specific uplift models:

1. DC-LR with undersampling for classification, which can only be used with DC,
2. DC-LR and Uplift RF with split undersampling and local neighborhood calibration.

Table 2 presents all theoretically sound combinations, including ones that were left out since they are—in our opinion—not interesting in practice. E.g. stratified undersampling with τ-isotonic regression would blatantly ignore the entire point with stratified undersampling - to change the positive rates in a way that is easy to work with. It also did not work particularly well in Nyberg et al. (2021) and is hence left out. Similar considerations apply to the other methods left out.

Tables 3, 4 and 5 report the results for all models on `Criteo-uplift 1`, Tables 6, 7 and 8 report the results on `Criteo-uplift 2`, and Tables 9, 10 and 11 report the results on `Zhao`. Tables 12, 13 and 14 report the results on the `Starbucks` with measures of variability over 10 runs as the dataset is small. Similarly, Tables 15, 16

Table 3 mAUUC on `Criteo-uplift 1`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	0.403	−0.006	0.289	0.226
Classic undersampling	0.257	n/a	n/a	n/a
Naive undersampling	0.256	0.385	0.271	0.231
Stratified undersampling	0.387	**0.391**	0.425	**0.406**
Split undersampling	**0.446**	n/a	**0.465**	n/a

A larger value is better and the best results for every uplift model is highlighted with bold font

Table 4 EUCE on `Criteo-uplift 1`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	0.00060	0.13441	0.00100	**0.00136**
Classic undersampling	0.00086	n/a	n/a	n/a
Naive undersampling	0.00177	0.00142	0.00140	0.00167
Stratified undersampling	0.00050	**0.00080**	**0.00087**	0.00575
Split undersampling	**0.00047**	n/a	0.00095	n/a

A smaller value is better. The best values for every uplift model is highlighted with bold font

Table 5 Optimal k-values on `Criteo-uplift 1`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	1	1	1	1
Classic undersampling	[32, 1]	n/a	n/a	n/a
Naive undersampling	16	128	256	32
Stratified undersampling	4	256	16	2
Split undersampling	[4, 12.4]	n/a	[64, 64]	n/a

Whenever two separate values for $k_{t=1}$ and $k_{t=0}$ are needed, they are presented in brackets as $[k_{t=1}, k_{t=0}]$

and 17 report the results on `Hillstrom` over 10 runs. The Tables 3, 6, 9, 12, and 15 report the main metric of mAUUC.

The best mAUUC scores are small for the three first datasets, but still correspond to significant practical improvements since the positive rates in these datasets are small. To help interpretation, the mAUUC scores can be converted to expected increase in the positive rate compared to the average positive rates: For `Criteo-uplift 1` the best mAUUC of 0.465 corresponds to a 23% improvement, for `Criteo-uplift 2` the best mAUUC of 0.482 equals a 19% increase, and for `Zhao` the mAUUC of 0.908 corresponds to a 35% increase. On the smaller datasets the best mAUUC of 2.377 of `Starbucks` corresponds to a 20% increase, and on `Hillstrom` the change is virtually zero.

To summarize the results, we next explain the most important observations supported by these results.

Table 6 mAUUC on `Criteo-uplift 2`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	**0.482**	− 0.246	0.278	0.288
Classic undersampling	0.442	n/a	n/a	n/a
Naive undersampling	0.481	0.443	0.360	0.438
Stratified undersampling	0.460	**0.445**	0.300	**0.468**
Split undersampling	0.422	n/a	**0.417**	n/a

A larger value is better. The best result for every uplift model is highlighted with bold font

Table 7 EUCE on `Criteo-uplift 2`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	**0.00053**	0.01707	0.00082	0.00865
Classic undersampling	0.00060	n/a	n/a	n/a
Naive undersampling	0.00183	0.00170	0.00177	0.00193
Stratified undersampling	0.00084	**0.00102**	0.00070	**0.00118**
Split undersampling	0.00064	n/a	**0.00061**	n/a

A smaller value is better. The best value for every uplift model is highlighted with bold font

Table 8 Optimal k-values on `Criteo-uplift 2`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	1	1	1	1
Classic undersampling	[128, 256]	n/a	n/a	n/a
Naive undersampling	1	32	8	32
Stratified undersampling	4	32	2	32
Split undersampling	[8, 16]	n/a	[32, 99.1]	n/a

Where both $k_{t=1}$ and $k_{t=0}$ are needed, they are presented in brackets as $[k_{t=1}, k_{t=0}]$

Undersampling helps and there is a preferred undersampling method for every uplift model. All four uplift models on the three larger datasets benefit notably from addressing the class imbalance using undersampling, both in terms of AUUC and EUCE. The more advanced stratified and split undersampling approaches provide the best performance. Even though classic and naive undersampling also sometimes improve the accuracy, most notably for CVT-LR, the two more advanced methods are to be preferred in practice as they reliably provide good performance. For CVT-LR and Uplift NN the recommendation is to always use stratified undersampling, whereas for the other two methods the accuracy can often be improved further by considering the computationally heavier split undersampling.

The methods differ in sensitivity to class imbalance. Correcting for class imbalance is extremely important for CVT-LR, Uplift RF and Uplift NN. The mAUUC on the original data is below 0.3 for both Criteo datasets and CVT-LR fails to obtain even a positive score, whereas with undersampling all reach an mAUUC in the range of 0.39–0.47.

Table 9 mAUUC on `Zhao`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	**0.908**	0.708	0.588	0.243
Classic undersampling	0.899	n/a	n/a	n/a
Naive undersampling	0.435	0.670	0.516	0.467
Stratified undersampling	**0.908**	**0.782**	**0.604**	**0.733**
Split undersampling	0.888	n/a	0.557	n/a

A larger value is better and the best results for every uplift model is highlighted with bold font

Table 10 EUCE on `Zhao`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	**0.00179**	0.13820	0.00202	0.00268
Classic undersampling	0.00232	n/a	n/a	n/a
Naive undersampling	0.00378	0.00366	0.00381	0.00415
Stratified undersampling	0.00336	**0.00294**	**0.00188**	**0.00214**
Split undersampling	0.00354	n/a	0.00321	n/a

A smaller value is better. The best values for every uplift model is highlighted with bold font

Table 11 Optimal k-values on `Zhao`

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	1	1	1	1
Classic undersampling	[4, 2]	n/a	n/a	n/a
Naive undersampling	64	16	128	128
Stratified undersampling	1	32	8	128
Split undersampling	[1, 32]	n/a	[8, 16]	n/a

Whenever two separate values for $k_{t=1}$ and $k_{t=0}$ are needed, they are presented in brackets as $[k_{t=1}, k_{t=0}]$

Table 12 Mean mAUUC on `Starbucks` of 10 runs, standard deviation in parenthesis

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	1.973	1.557	2.147	0.579
	(0.241)	(0.422)	(0.157)	(0.576)
Classic undersampling	1.901	n/a	n/a	n/a
	(0.250)	n/a	n/a	n/a
Naive undersampling	1.919	**1.847**	2.043	1.902
	(0.255)	(0.215)	(0.233)	(0.272)
Stratified undersampling	**1.988**	1.836	2.217	**1.773**
	(0.220)	(0.205)	(0.175)	(0.405)
Split undersampling	1.954	n/a	**2.377**	n/a
	(0.348)	n/a	(0.254)	n/a

A larger value is better. The best result for every uplift model is highlighted with bold font

Table 13 Mean EUCE on `Starbucks` of 10 runs, standard deviation in parenthesis

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	**0.00928**	0.01243	**0.01007**	0.03789
	(0.00084)	(0.00157)	(0.00060)	(0.03177)
Classic undersampling	0.01001	n/a	n/a	n/a
	(0.00081)	n/a	n/a	n/a
Naive undersampling	0.01055	**0.01006**	0.01013	**0.01004**
	(0.00055)	(0.00065)	(0.00076)	(0.00058)
K-undersampling	0.01255	0.01228	0.01022	0.01063
	(0.00079)	(0.00080)	(0.00083)	(0.00072)
Split-undersampling	0.00938	n/a	0.01015	n/a
	(0.00094)	n/a	(0.00067)	n/a

A smaller value is better. The best value for every uplift model is highlighted with bold font

Even the smallest increase in mAUUC (Uplift RF on `Criteo-uplift 2`) corresponds to a 50% relative improvement. On `Zhao` the improvements were less striking, but also here these methods improved by undersampling with the largest improvement seen on the Uplift NN. DC-LR, however, is very robust to the class imbalance, reaching similar mAUUC for the three larger datasets already without undersampling. Importantly, undersampling does not seem to hurt either—for `Criteo-uplift 1` we observe a small improvement and for `Criteo-uplift 2` a small decrease in mAUUC, but for the three larger datasets, the method mostly remains competitive also with undersampling.

Split undersampling is presented in a bit more detail in Fig. 2. DC-LR and Uplift RF were tested on `Criteo-uplift 2` for these plots. The baseline with no undersampling is at the bottom left corner (indicated with an x). As can be seen in the plots, DC-LR is better than Uplift RF on most selections of $k_{t=0}$ and $k_{t=1}$. The best k-values were off-diagonal for both models (marked with a star). The plots show that split undersampling might be better than stratified undersampling with these models, although if computational complexity is an issue, the results on the diagonal show that stratified undersampling might be a good compromise.

Datasets need to be large enough to benefit from undersampling. The improvements in mAUUC were largest on the largest datasets and decreased with dataset size. On `Criteo-uplift 1` and `Criteo-uplift 2` we saw sizeable improvements, whereas on `Zhao` the improvements were more modest. On the second smallest dataset, `Starbucks`, we see clear improvements in mAUUC only for the neural net, but even then the metrics do not exceed that of the basic benchmark of DC-LR. On the smallest dataset, `Hillstrom`, all mAUUC-metrics are essentially zero and slightly below the baselines. This seems to indicate slight overfitting. These results on `Hillstrom` are in line with Rössler et al. (2021) who ran repeated experiments with a double classifier on the same dataset and found no uplift. Further, while comparing EUCE-values for models with no uplift is pointless, Table 16 with EUCE for `Hillstrom` is included for completeness. The effect of dataset size is investigated in more detail in Sect. 4.4.

Table 14 Medians of optimal k-values on `Starbucks` of 10 runs, minimum and maximum in parenthesis

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	1	1	1	1
	(1, 1)	(1, 1)	(1, 1)	(1, 1)
Classic undersampling	[16, 8]	n/a	n/a	n/a
	[(1, 32), (1, 128)]	n/a	n/a	n/a
Naive undersampling	8	48	6	16
	(1, 32)	(2, 64)	(1, 64)	(8, 64)
Stratified undersampling	8	32	3	32
	(2, 32)	(1, 32)	(1, 32)	(1, 32)
Split undersampling	[24, 64]	n/a	[3, 32]	n/a
	[(4, 32), (16, 128)]	n/a	[(1, 16), (1, 64)]	n/a

For classic and split underesampling, the median values are reported in square brackets as $[k_{t=1}, k_{t=0}]$, the minimum and maximum values for $k_{t=1}$ on the next line, and the minimum and maximum values for $k_{t=0}$ on the third line

Table 15 Mean mAUUC on `Hillstrom` of 10 runs, standard deviation in parenthesis

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	**0.051**	**0.042**	**0.142**	**0.007**
	(0.446)	(0.473)	(0.431)	(0.391)
Classic undersampling	−0.181	n/a	n/a	n/a
	(0.393)	n/a	n/a	n/a
Naive undersampling	−0.064	−0.240	0.010	−0.076
	(0.444)	(0.454)	(0.421)	(0.453)
Stratified undersampling	−0.162	−0.258	0.047	−0.205
	(0.371)	(0.221)	(0.472)	(0.513)
Split undersampling	−0.222	n/a	−0.064	n/a
	(0.454)	n/a	(0.664)	n/a

A larger value is better. The best result for every uplift model is highlighted with bold font

Table 16 Mean EUCE on `Hillstrom` of 10 runs, standard deviation in parenthesis

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	0.01542	0.02503	**0.01464**	0.04051
	(0.00109)	(0.00199)	(0.00140)	(0.02624)
Classic undersampling	0.01547	n/a	n/a	n/a
	(0.00105)	n/a	n/a	n/a
Naive undersampling	0.01531	0.01532	0.01535	**0.01538**
	(0.00096)	(0.00127)	(0.00109)	(0.00091)
K-undersampling	**0.01347**	**0.01292**	0.01469	0.02006
	(0.00102)	(0.00084)	(0.00129)	(0.00772)
Split-undersampling	0.01550	n/a	0.01519	n/a
	(0.00086)	n/a	(0.00181)	n/a

A smaller value is better. The best value for every uplift model is highlighted with bold font

4.4 Experiment 2: reducing dataset size

The previous experiment indicates that improvement in AUUC decreased with dataset size. In this experiment we inspect in more detail how the approaches work on smaller datasets, which in class-imbalanced problems necessarily means having only a few positive observations. For this study we retain only the better stratified and split undersampling methods. To create the smaller training datasets in a controlled manner we randomly took subsamples of 100%, 50%, 25%, 10%, 5%, 2.5%, and 1.25% observations of the `Criteo-uplift 2` data, but still use the large test set with 25% of the 4.2 million observations. Using the full test set ensures that we can reliably estimate the final accuracy, and hence the results provide more direct evidence on how the methods themselves can account for smaller sample size. As the dataset has 4063 positive untreated observations and we used the 50/25/25 split (training/validation/test), roughly 1000 of those were in the testing set, 2000 in the training set, and the remaining 1000 in the validation set.

Table 17 Medians of optimal k-values on `Hillstrom` of 10 runs, minimum and maximum values are in parenthesis

	DC-LR	CVT-LR	Uplift RF	Uplift NN
No undersampling	1	1	1	1
	(1, 1)	(1, 1)	(1, 1)	(1, 1)
Classic undersampling	[4, 12]	n/a	n/a	n/a
	[(1, 64), (1, 128)]	n/a	n/a	n/a
Naive undersampling	16	16	6	8
	(1, 64)	(1, 32)	(1, 32)	(1, 32)
Stratified undersampling	16	12	1.5	4
	(2, 64)	(1, 64)	(1, 64)	(1, 64)
Split undersampling	[16, 64]	n/a	[8, 16]	n/a
	[(2, 64), (1, 128)]	n/a	[(1, 64), (2, 128)]	n/a

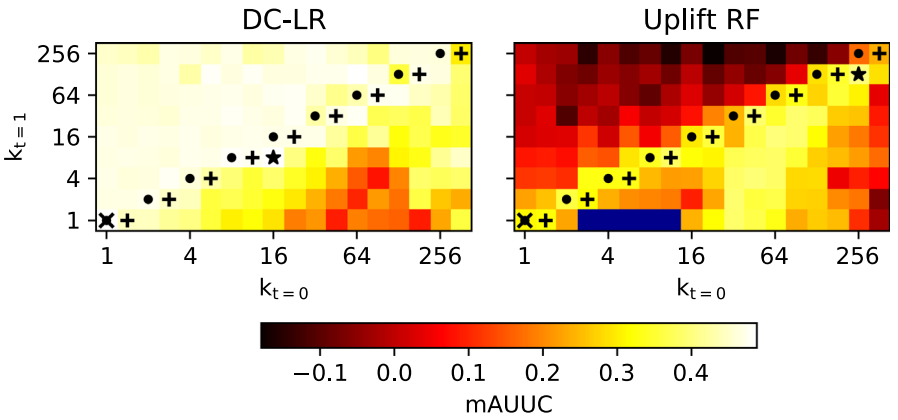

Fig. 2 mAUUC for DC-LR and Uplift RF with split undersampling and different values of $k_{t=1}$ and $k_{t=0}$. The best values are marked with a star. In the bottom left corner marked with an x is the baseline with no undersampling ($k_{t=1} = k_{t=0} = 1$), on the diagonal marked with dots are cases where $k_{t=1} = k_{t=0}$ (equivalent to stratified undersampling). On the off-diagonal marked with plus-signs are cases where $k_{t=1}$ was selected so that $p(y = 1|t = 1) = p(y = 1|t = 0)$. The squares in blue could not be trained within 24 hours (Color figure online)

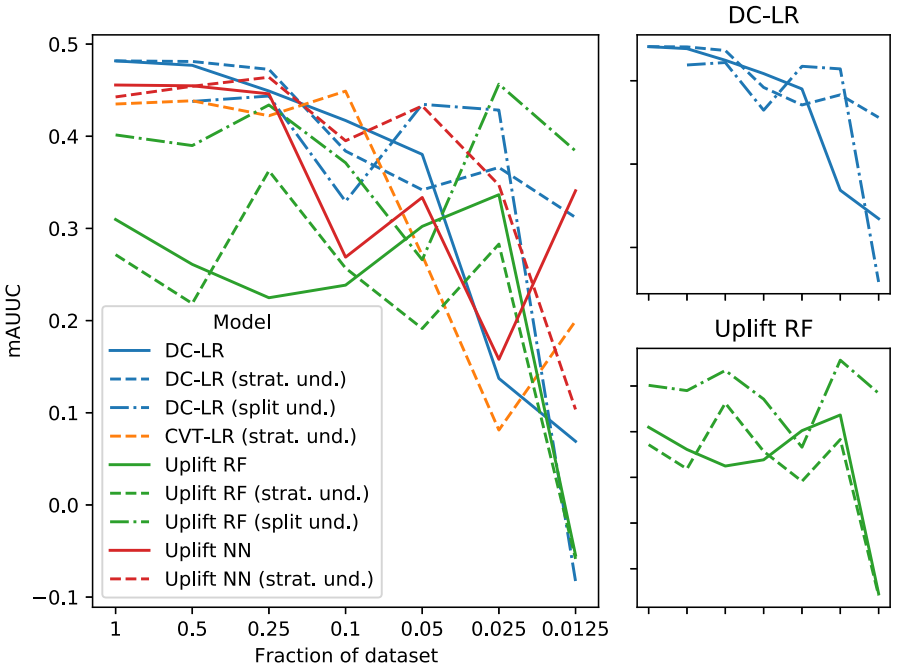

Fig. 3 mAUUC on the tested models on `Criteo-uplift 2` with reduced dataset size. CVT-LR was left out as it was negative at all points. Both RF without undersampling and with stratified undersampling performed particularly poorly, but this was largely corrected with split undersampling and local neighborhood calibration. Most models performed well with 10% of the dataset (fraction 0.1). DC-LR and Uplift RF plots are reproduced to the right for legibility

Figure 3 plots mAUUC for the different models as a function of the data size, dropping CVT-LR without undersampling as it always had an mAUUC below zero. The plot is slightly cluttered, and hence we summarize here the main observations. The main trend is that the accuracies decrease for smaller datasets, but importantly the best methods retain very high mAUUC even when trained on extremely small data. The performance of all models was still good with one tenth of the data (fraction 0.1) where there was only 200 positive untreated observations in the training data. One tenth approximately corresponds to the size of `Zhao`. The fraction 0.025 of the dataset roughly corresponds to the size of `Starbucks`. At this fraction, there was already more variability, although the models still produce positive results. The performance of some models were decent even with the smallest of the tested fractions (0.0125), although at this point the models were quite unstable and some models even produced negative mAUUC. This fraction is roughly equivalent to the size of `Hillstrom`. Essentially the same pattern is seen here as in Experiment 1: large datasets clearly benefit from undersampling, and the benefits of undersampling decreases with decreasing dataset size.

A different observations was that even though DC-LR was in the previous experiment found to be robust for class imbalance, using undersampling becomes important also for that when trained on very small datasets.

4.5 Experiment 3: when is $p(y = 1|t)$ *small*?

Stratified undersampling and renormalization calibration rely on the assumption that $p(y = 1|t = 1)$ and $p(y = 1|t = 0)$ are both small and hence also similar. This experiment investigates what *small* might actually mean in practice by changing the positive rate $p(y = 1|t = 1)$ in the dataset, building on the expectation that for higher $p(y = 1|t = 1)$ stratified undersampling might break down and reveal larger advantage for split undersampling. To directly measure this, we only use the models compatible with split undersampling.

We again use a semi-synthetic datasets, constructing a data of 734,000 observations by sampling observations from `Criteo-uplift 2` randomly to produce data with the following properties: Half of the observations were treated, half untreated. The positive rate among the untreated observations was kept constant at 0.19% (this is the natural rate in the dataset) while the positive rate among treated observations was first reduced to match 0.19%, and then doubled from that five times to get rates 0.39%, 0.78%, 1.55%, 3.1%, and 6.2%. The last one is roughly the upper bound for the conversion rate for treated observations that could be generated from `Criteo-uplift 2` without resampling.

The results are reported in Fig. 4 as relative to the metrics of DC-LR to make the figure legible, and as absolute values in Table 18. The results confirm some of the earlier findings: (a) DC-LR is robust to class imbalance as seen by different variants having near identical performance except for at the smallest conversion rate, and (b) correcting for class imbalance is crucial for Uplift RF. We also confirm the basic hypothesis that stratified undersampling works well when the conversion rates are small—for $p(y = 1|t = 1) = 0.0019$ we can correct Uplift RF also with that method—but when

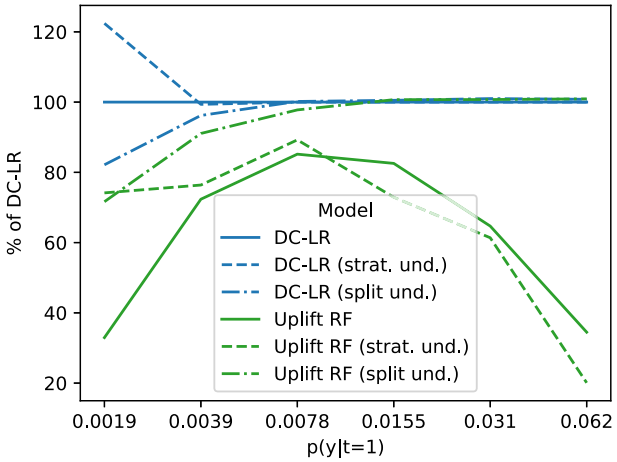

Fig. 4 The performance of DC-LR and Uplift RF with different undersampling strategies over semi-simulated datasets where $p(y = 1|t = 0)$ is kept constant and $p(y = 1|t = 1)$ is adjusted (horizontal axis). DC-LR with no undersampling is used as baseline (100%) and all other results are normalized by the AUUC of this baseline

Table 18 AUUC on `Criteo-uplift 2` for different $p(y = 1|t = 1)$. $p(y = 1|t = 1)$ is first dropped to 0.0019, which is equal to $p(y = 1|t = 0)$, and then increased by a factor of two up til 0.062 (6.2%)

$p(y = 1\|t = 1)$	0.0019	0.0039	0.0078	0.0155	0.031	0.062
DC-LR	0.00018	0.00081	0.00258	0.00601	0.01171	0.02202
DC-LR (strat. und.)	0.00022	0.00080	0.00258	0.00601	0.01171	0.02202
DC-LR (split und.)	0.00015	0.00078	0.00258	0.00604	0.01183	0.02220
Uplift RF	0.00006	0.00058	0.00219	0.00496	0.00758	0.00760
Uplift RF (strat. und.)	0.00013	0.00062	0.00230	0.00438	0.00719	0.00442
Uplift RF (split und.)	0.00013	0.00073	0.00252	0.00605	0.01180	0.02222

the conversion rate grows we indeed need to use split undersampling. This is best seen in Uplift RF that is more sensitive to the correction method, but also for DC-LR we observe very good performance for stratified undersampling for low conversion rates.

Table 18 provides the exact AUUC numbers and shows they are consistent with what we should expect when modifying the underlying problem like this. Roughly speaking, the uplift (AUUC) present in the data should approximately double as $p(y = 1|t = 1)$ doubles, and from the rate 0.0039 this indeed holds for all DC-LR variants tested and Uplift RF with split undersampling. For the smallest conversion rate the AUUC is lower than expected, possibly suggesting none of the models is finding a very good solution.

5 Discussion

Even though the problem of high class imbalance is prevalent in typical uplift modeling problems, especially in e-commerce, the aspect has been ignored in the literature. In addition, the largest available datasets for training and evaluating uplift models, `Criteo-uplift 1` and `2`, are known for being hard to reach any meaningful improvements on. We believe this is precisely because the datasets exhibit high class imbalance and there has not existed techniques for uplift modeling to deal with this. Many of the methods fail miserably on these datasets if the imbalance is not corrected for, and the same happens on `Zhao` as reported here. This implies that some of the conclusions made in earlier works may be misguided—specific methods are observed to perform poorly but could have been fixed fairly easily by the undersampling techniques proposed here. For instance, Fernández-Loría and Provost (2022) compared a classification model to an uplift random forest and claimed that a simple classification model performed better on uplift metrics based on empirical experiments on `Criteo-uplift 2`. As shown here, uplift random forests are a weak baseline for imbalanced datasets and for fair comparison the proposed method would need to be contrasted either against a double classifier method robust to class imbalance, or against uplift RF with undersampling. Even though the specific method and experimental details differ, the 70% improvement in AUUC observed in our case strongly suggests that the baseline in their case could have been improved easily. Similarly, Semenova and Temirkaeva (2019) report poor performance for uplift random forests, again on `Criteo-uplift 2`, and proceed to suggest DC-LR as the method of choice. We believe this result is also because of not accounting for the class imbalance and hence as such not a fault of the uplift random forest method itself. In addition to uplift random forests, we highlight the importance of correcting for the imbalance also for CVT-LR. In our experiments CVT-LR did not work at all when applied as is, even though it is a useful method on more balanced datasets (Jaskowski and Jaroszewicz 2012).

Another important observation is that the benefit of undersampling was dependent on dataset size. While we could see clear improvements for the three large datasets, on the smaller `Starbucks` and `Hillstrom` we could not observe any reliable differences. The key reason is that the mAUUC estimates themselves are very noisy for datasets this small. It means that potential improvements are difficult to differentiate from the variability across runs, but more importantly it implies we lose the ability to select the undersampling factor k well. All of the proposed methods rely on validation set accuracy for the choice of k, and with small validation sets the choice becomes largely random. Our main motivation for the work was in improving accuracy for large-scale uplift problems where undersampling was shown to work well and has the additional advantage of improving computational speed, and the results show that it indeed is limited to these setups. For small datasets we would need alternative solutions, since even the notion of discarding any of the limited samples (by dropping them, or even by dedicating them to be used only for validation) is not a sensible starting point. Methods based on oversampling, reweighting or synthetic sampling might provide a better starting point, but would require dedicated effort to adjust for

the needs of uplift modeling and with extremely small datasets the question of not being able to accurately estimate the uplift is likely to remain a major challenge.

The observation that DC-LR is highly resilient to class imbalance is also important. Even though Radcliffe and Surry (1999) and Guelman et al. (2015) discouraged use of DC-LR based on theoretical arguments and many follow-up works refer to their recommendations, the recommendations were not backed up by empirical evidence. In light of the results of Semenova and Temirkaeva (2019) and our observation of robustness to class imbalance, we explicitly recommend including DC-LR, one of the easiest uplift models to use, as a baseline in method comparisons for tasks with strong imbalance. We do not a have clear explanation of the particularly good performance of DC-LR on the `Criteo` datasets, but speculate that it may relate to the input features that are projections from real inputs to preserve anononymity; this may remove non-linearities in the actual modeling problem that only more advanced models would have thrived on. Regardless of whether DC-LR is particularly accurate on other datasets, the robustness to class imbalance makes it an important baseline.

Considering the imbalance of Uplift RF, the results stem from some leafs often containing just a handful of positive observations. In cases where we are interested in conditional probabilities that are a fraction of a percent, changing the number of positive treated or untreated observations by just one will already cause a sizeable change in ranks between predictions. Undersampling makes the leaves more balanced and removes this instability. A similar result might be achievable by some form of regularization, but this is something that has not been deal with in the uplift random forest literature. We leave this for future work.

Our results focus on showing the accuracy of the uplift models, not paying attention to the computational cost as we do not believe it to be a major factor in practical use. The undersampling methods require selection of the undersampling factors by cross-validation and split undersampling requires performing a sweep over two factors jointly, but the computation of the alternative solutions parallelizes trivially. Furthermore, for larger undersampling factors the datasets become extremely small compared to the original data and hence many of the alternatives will be fast to evaluate. Finally, after selecting the undersampling factor it is faster to re-train the uplift model (e.g. for newly arriving data), eventually compensating for the increased cost in initial modeling.

We also want to highlight an observation that is valuable for practical use of uplift models e.g. in industry. In Experiment 2 we showed that we were able to estimate uplift accurately based on just a few hundred positive training observations. The cost of obtaining positive observations may sometimes be high and knowing that already a number this small may be enough will help in designing the data collection experiment. Even though the exact number of required observations naturally depends on the specific case, our results already provide a rough order of magnitude as a target.

Finally, we observe that only three of the datasets were large-scale datasets and hence e.g. general conclusions on relative accuracy of specific methods cannot be made based on these results. Instead, our main point is to highlight the importance of accounting for the class imbalance and demonstrate that undersampling provides a general solution to the problem. For further investigation of the methods on additional

datasets, we refer you to our code at https://github.com/Trinli/uplift_modeling that allows easily re-running the experiments with any data.

6 Conclusion

In this work we thoroughly investigated undersampling as pre-processing and calibration as post-processing for uplift modeling, considerably extending our preliminary work Nyberg et al. (2021) providing the first practical solutions for addressing class imbalance in uplift modeling. We showed how probabilities are distorted as a consequence of undersampling, and provided alternative undersampling approaches and calibration methods for addressing this distortion to produce valid uplift estimates from undersampled data.

We demonstrated the different undersampling methods in context of several uplift models on the largest available datasets with clear results: most uplift models need undersampling to perform well if the data exhibits high class imbalance, and in particular uplift random forests and methods based on the class-variable transformation are extremely sensitive to class imbalance. However, undersampling mitigates the problem well. The proposed methods work reliably for sufficiently large datasets (approximately 500,000 samples or more), but accounting for class imbalance based on very small data sets requires further work. Based on our findings, we conclude by making four concrete recommendations for both the research community and the industry using uplift models:

1. If the data exhibits class imbalance, you need to account for it.
2. Accounting for the imbalance is particularly important for uplift models based on random forests and class variable transformations.
3. The double classifier with logistic regression is robust to class imbalance and should be included as a benchmark in method comparisons.
4. The best methods to account for the class imbalance are stratified and split undersampling.

Acknowledgements This project was supported by Business Finland (project MINERAL) and the Academy of Finland (Flagship programme: Finnish Center for Artificial Intelligence, FCAI).

Funding Open Access funding provided by University of Helsinki including Helsinki University Central Hospital.

Declarations

Conflict of interest Neither author has any conflicts of interest to disclose.

A Appendix: Local neighborhood correction

Equation (12) in Sect. 3.3.3 shows how estimates of $p^*(y = 1|x, t)$ can be converted into estimates of $p(y = 1|x, t)$. Here we justify the expression.

Consider a case where the conditional probabilities are estimated from observations in local neighborhood of x, denoting the counts of the original observations in the neighborhood by N_{pos} and N_{neg} and the corresponding counts after undersampling by N^*_{pos} and N^*_{neg}. From the undersampled data we hence get the maximum likelihood estimate $p^*_{ML}(y = 1|x, t) = \frac{N^*_{pos}}{N^*_{pos}+N^*_{neg}}$. For the original data we have

$$p_{ML}(y = 1|x, t) = \frac{N_{pos}}{N_{pos} + N_{neg}} = \frac{p^*_{ML}(y = 1|x, t)}{p^*_{ML}(y = 1|x, t) + \frac{N_{neg}}{N^*}},$$

where we are able to express some terms as a function of $p^*_{ML}(y = 1|x, t)$ by dividing the factors with $N^* = N^*_{pos} + N^*_{neg}$ because N_{pos} does not change in the undersampling.

Here N_{neg} is a random variable as it cannot be directly observed in the undersampled data. The number of discarded observations $N_{neg} - N^*_{neg}$ follows a negative binomial distribution with parameters N^*_{neg} and $1 - s_t$ and hence the expectation of N_{neg} is $\frac{N^*_{neg}}{s_t}$ and the mode is $\frac{N^*_{neg}-(1-s_t)}{s_t}$ rounded down to an integer. By plugging in the expectation, writing $\frac{N^*_{neg}}{N^*} = 1 - p^*_{ML}(y = 1|x, y)$, and performing simple algebraic manipulation we get

$$p_{ML}(y = 1|x, t) = \frac{p^*_{ML}(y = 1|x, t)}{p^*_{ML}(y = 1|x, t) + \frac{N^*_{neg}}{s_t N^*}} = \frac{s_t \cdot p^*_{ML}(y = 1|x, t)}{1 - p^*_{ML}(y = 1|x, t) \cdot (1 - s_t)},$$

matching Eq. (12).

For the mode the rounding operation complicates the derivation, but in practical terms the only difference is that in the denominator we need to subtract $\frac{1}{N^*}$ from $p^*_{ML}(y = 1|x, t)$. Since we expect the estimators to be computed from sufficiently large total N^*, this bias is negligible in practice and hence the relationship holds also for the most likely estimators.

References

Athey S, Imbens G (2015) Recursive partitioning for heterogeneous causal effects. arXiv arXiv:1504.01132

Austin PC (2011) An introduction to propensity score methods for reducing the effects of confounding in observational studies. Multivar Behav Res 46(3):399–424

Belbahri M, Gandouet O, Kazma G (2020) Adapting neural networks for uplift models. arXiv:2011.00041

Belbahri M, Gandouet O, Murua A et al (2021) A twin neural model for uplift. arxiv:2105.05146

Betlei A, Diemert E, Amini MR (2018) Uplift prediction with dependent feature representation in imbalanced treatment and control conditions. Lecture notes in computer science, vol V. Springer, Cham, pp 47–55

Chawla NV, Bowyer KW, Hall LO et al (2002) Smote: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357

Devriendt F, Berrevoets J, Verbeke W (2021) Why you should stop predicting customer churn and start using uplift models. Inf Sci 548:497–515
Diemert E, Betlei A, Renaudin C et al (2018) A large scale benchmark for uplift modeling. In: Proceedings of the AdKDD and TargetAd workshop, KDD, London, United Kingdom, August, 20, 2018
Fernández-Loría C, Provost F (2022) Causal classification: treatment effect vs. outcome prediction. J Mach Learn Res 23:1–35
Gubela RM, Lessmann S, Jaroszewicz S (2020) Response transformation and profit decomposition for revenue uplift modeling. Eur J Oper Res 283(2):647–661
Guelman L, Guillén M, Pérez-Marín AM (2015) Uplift random forests. Cybern Syst 46(3–4):230–248
Gutierrez P, Gérardy JY (2017) Causal inference and uplift modelling: a review of the literature. In: Proceedings of the 3rd international conference on predictive applications and APIs, vol 67, pp 1–13
Haupt J, Lessmann S (2020) Targeting customers under response-dependent costs. arxiv:2003.06271. https://doi.org/10.1016/j.ejor.2021.05.045
Jaskowski M, Jaroszewicz S (2012) Uplift modeling for clinical trial data. ICML workshop on clinical data analysis
Johansson FD, Shalit U, Sontag D (2016) Learning representations for counterfactual inference. In: Proceedings of the 33rd international conference on machine learning
Kaur H, Pannu HS, Malhi AK (2019) A systematic review on imbalanced data challenges in machine learning: applications and solutions. ACM Comput Surv 52(4):1–36
Künzel SR, Sekhon JS, Bickel PJ et al (2019) Metalearners for estimating heterogeneous treatment effects using machine learning. Proc Natl Acad Sci USA 116:4156–4165
Lai LYT (2006) Influential marketing: a new direct marketing strategy addressing the existence of voluntary buyers. Ph.D. thesis, University of British Columbia
Lo VS (2002) The true lift model—a novel data mining approach to response modeling in database marketing. SIGKDD Explor 4:78–86
Naeini MP, Cooper GF, Hauskrecht M (2015) Obtaining well calibrated probabilities using Bayesian binning. In: Proceedings of the twenty-ninth AAAI conference on artificial intelligence, vol 2015, pp 2901–2907
Nyberg O, Kuśmierczyk T, Klami A (2021) Uplift modeling with high class imbalance. In: Proceedings of the 13th Asian conference on machine learning, pp 315–330
Olaya D, Coussement K, Verbeke W (2020) A survey and benchmarking study of multitreatment uplift modeling. Data Min Knowl Disc 34(2):273–308
Papangelou K (2021) Assessing treatment effect heterogeneity: predictive covariate selection and subgroup identification. Ph.D. thesis, University of Manchester
Pearl J (2009) Causal inference in statistics: an overview. Stat Surv 3:96–146
Pedregosa F, Varoquaux G, Gramfort A et al (2011) Scikit-learn: machine learning in python. J Mach Learn Res 12:2825–2830
Platt J (1999) Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Adv Large Margin Classif 10:61–74
Radcliffe NJ (2008) Hillstrom's MineThatData email analytics challenge: an approach using uplift modelling. Response, pp 1–19. http://stochasticsolutions.com/
Radcliffe NJ, Surry PD (1999) Differential response analysis: modelling true response by isolating the effect of a single action. Credit scoring and credit control VI
Richardson M, Ragno R, Dominowska E (2007) Predicting clicks: estimating the click-through rate for new ads. In: Proceedings of the 16th international conference on World Wide Web, pp 521–529
Rössler J, Tilly R, Schoder D (2021) To treat, or not to treat: reducing volatility in uplift modeling through weighted ensembles. In: Proceedings of the 54th Hawaii international conference on system sciences
Rudaś K, Jaroszewicz S (2018) Linear regression for uplift modeling. Data Min Knowl Discov 32:1–31
Rzepakowski P, Jaroszewicz S (2010) Decision trees for uplift modeling. In: Proceedings—IEEE international conference on data mining, ICDM pp 441–450
Semenova D, Temirkaeva M (2019) The comparison of methods for individual treatment effect detection. In: CEUR workshop proceedings, pp 46–56
Verbeke W, Dejaeger K, Martens D et al (2012) New insights into churn prediction in the telecommunication sector: a profit driven data mining approach. Eur J Oper Res 218:211–229
Wager S, Athey S (2018) Estimation and inference of heterogeneous treatment effects using random forests. J Am Stat Assoc 113(523):1228–1242

Zadrozny B, Elkan C (2002) Transforming classifier scores into accurate multiclass probability estimates. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining KDD, vol 02, pp 694–699

Zhao Z, Zhang Y, Harinen T, et al (2022) Feature selection methods for uplift modeling and heterogeneous treatment effect. In: IFIP advances in information and communication technology, pp 217–230