Support Vector Machines for Uplift Modeling

Łukasz Zaniewicz
Institute of Computer Science
Polish Academy of Sciences
Warsaw, Poland
Email: l.zaniewicz@phd.ipipan.waw.pl

Szymon Jaroszewicz
Institute of Computer Science
Polish Academy of Sciences
Warsaw, Poland

National Institute of Telecommunications
Warsaw, Poland
Email: s.jaroszewicz@ipipan.waw.pl

***Abstract*—Uplift modeling is a branch of Machine Learning which aims to predict not the class itself, but the difference between the class variable behavior in two groups: treatment and control. Objects in the treatment group have been subject to some action, while objects in the control group have not. By including the control group it is possible to build a model which predicts the causal effect of the action for a given individual. In this paper we present a variant of Support Vector Machines designed specifically for uplift modeling. The SVM optimization task has been reformulated to explicitly model the difference in class behavior between two datasets. The model predicts whether a given object will have a positive, neutral or negative response to a given action, and by tuning a parameter of the model the analyst is able to influence the relative proportion of neutral predictions and thus the sensitivity of the model. We adapt the dual coordinate descent method to efficiently solve our optimization task. Finally the proposed method is compared experimentally with other uplift modeling approaches.**

***Keywords*-Uplift modeling, Support Vector Machine, incremental response modeling, control group**

I. Introduction

Traditional classification methods predict the conditional class probability distribution in a given dataset. Based on those predictions an action is often taken on the classified individuals. This approach is, however, usually incorrect, especially in the case of marketing campaigns or controlled medical trials. Standard classification methods are only able to model what happens *after* the action has been taken not what happens *because* of the action. The reason is that such models do not take into account what would have happened had the action not been taken.

This is easiest to see in the context of direct marketing campaigns. Some of the customers who bought after the action would have bought anyway, the action incurred unnecessary cost. Worse, some customers who were going to buy got annoyed by the action, refrained from purchase and may even churn. The existence of such negative groups is a well known phenomenon in marketing literature [1] and detecting them is often crucial for the success of a campaign.

Uplift modeling, in contrast, allows for the use of a control dataset and aims at explicitly modeling the difference in outcome probabilities between the two groups, thus being able to identify cases for which the outcome of the action will be truly positive, neutral or negative.

In this paper we present Uplift Support Vector Machines (USVMs) which are an application of the SVM methodology to the problem of uplift modeling. The SVM optimization problem has been reformulated such that the machine accepts two training datasets: treatment and control, and models the differences in class behavior between those sets. Other uplift modeling methods return the score of an instance; USVMs are the first such method we are aware of, which aims to explicitly predict whether an outcome of an action for a given case will be positive, negative or neutral. What is especially important is that the model identifies the negative group allowing for minimizing the adverse impact of the action. Moreover, by proper choice of parameters, the analyst is able to decide on the relative proportion of neutral predictions, thus tuning model's sensitivity to positive and negative cases.

The main problem of uplift modeling is that for each data point we know only one of the outcomes, either after the action has been performed or when no action has been performed, never both. This makes the task less intuitive than standard classification, and formulating optimization tasks becomes significantly more difficult.

A. Previous work

Surprisingly, uplift modeling has received relatively little attention in the literature. The most obvious approach uses two separate probabilistic models, one built on the treatment and the other on the control dataset, and subtracts their predicted probabilities. The advantage of the two-model approach is that it can be applied with any classification model. Moreover, if uplift is strongly correlated with the class attribute itself, or if the amount of training data is sufficient for the models to predict the class probabilities accurately, the two-model approach will perform very well also in the uplift case. The disadvantage is, that when uplift follows a different pattern than the class distributions, the models will focus on predicting the class, instead of focusing on the weaker 'uplift signal'. See [2] for an illustrative example.

A few papers addressed decision tree construction for uplift modeling. See e.g. [1], [3], [4], [2]. In [5] uplift decision trees have been presented which are more in line with modern machine learning algorithms. The approach has been extended to the case of multiple treatments in [6].

Some regression techniques for uplift modeling are available. Most researchers, however, follow the two model approach either explicitly or implicitly [7], [8], [9], [10], [11]. In [12] a method has been presented which makes it possible to convert a classical logistic regression model (or in fact any other probabilistic classifier) into an uplift model. The approach is based on a class variable transformation. Recently, in [13], the approach has been extended to work in the context of online advertising, where it is necessary to not only maximize uplift (the difference between success rate in the treatment and control datasets) but also to increase advertiser's gains through maximizing response. This type of problems are beyond the scope of this paper.

Recent, thorough literature overviews on uplift modeling can be found in [5] and [2].

Support Vector Machines with parallel hyperplanes, similar to our approach, have been analyzed in the context of ordinal classification [14]; here the situation is different as two training datasets are involved.

II. Uplift Support Vector Machines

We now introduce the notation and formally define Uplift Support Vector Machines (USVMs). The class $+1$ will be considered the *positive*, or desired outcome. The scalar product of vectors $\mathbf{x}_1$, $\mathbf{x}_2$ will be denoted with $\langle \mathbf{x}_1, \mathbf{x}_2 \rangle$.

SVMs are designed primarily for classification, not probability modeling, so in order to adapt SVMs to the analyzed setting we first recast the uplift modeling problem as a three-class classification problem. This differs from the typical formulation which aims at predicting the difference in class probabilities between treatment and control groups.

Unlike standard classification, in uplift modeling we have two training samples: the *treatment group*, $\mathbf{D}^T = \{(\mathbf{x}_i, y_i) : i = 1, \ldots, n^T\}$ and the *control group* $\mathbf{D}^C = \{(\mathbf{x}_i, y_i) : i = 1, \ldots, n^C\}$, where $\mathbf{x}_i \in \mathbb{R}^m$ are the values of the predictor variables, and $y_i \in \{-1, 1\}$ is the class of the i-th data record, m is the number of attributes in the data, and n^T and n^C are the numbers of records in the treatment and control groups respectively. Objects in the treatment group have been subject to some *action* or *treatment*, while objects in the control group have not.

In the rest of the paper we will continue to follow the convention that all quantities related to the treatment group will be denoted with superscript ^{T} and those related to the control group with superscript ^{C}.

An *uplift model* is defined as a function

$$M(\mathbf{x}) : \mathbb{R}^m \to \{-1, 0, 1\}, \tag{1}$$

which assigns to each point in the input space one of the values $+1$, 0 and -1, interpreted, respectively, as positive, neutral and negative impact of the action. In other words, the positive prediction $+1$ means that we expect the objects class to be $+1$ if it is subject to treatment and -1 if it is not, the negative prediction means that we expect the class to be -1 after treatment and $+1$ if no action was performed, and neutral if the object's class is identical (either $+1$ or -1) regardless of whether the action was taken or not.

The proposed Uplift Support Vector Machine (USVM), which performs uplift prediction, uses two parallel hyperplanes

$$H_1 : \langle \mathbf{w}, \mathbf{x} \rangle - b_1 = 0, \qquad H_2 : \langle \mathbf{w}, \mathbf{x} \rangle - b_2 = 0,$$

where $b_1, b_2 \in \mathbb{R}$ are the intercepts. The model predictions are specified by the following equation

$$M(\mathbf{x}) = \begin{cases} +1 & \text{if } \langle \mathbf{w}, \mathbf{x} \rangle > b_1 \text{ and } \langle \mathbf{w}, \mathbf{x} \rangle > b_2, \\ 0 & \text{if } \langle \mathbf{w}, \mathbf{x} \rangle \leq b_1 \text{ and } \langle \mathbf{w}, \mathbf{x} \rangle > b_2, \\ -1 & \text{if } \langle \mathbf{w}, \mathbf{x} \rangle \leq b_1 \text{ and } \langle \mathbf{w}, \mathbf{x} \rangle \leq b_2. \end{cases} \tag{2}$$

Intuitively, the point is classified as positive if it lies on the positive side of both hyperplanes, neutral if it lies on the positive side of hyperplane H_2 only, and classified as negative if it lies on the negative side of both hyperplanes. In other words, H_1 separates positive and neutral points, and H_2 neutral and negative points. Notice that the model is valid iff $b_1 \geq b_2$; in Lemma 1 we will give sufficient conditions for this inequality to hold.

Let us now formulate the optimization task which allows for finding the model's parameters $\mathbf{w}, b_1, b_2$. We will use $\mathbf{D}^T_+ = \{(\mathbf{x}_i, y_i) \in \mathbf{D}^T : y_i = +1\}$ to denote data points belonging to the positive class in the treatment group and $\mathbf{D}^T_- = \{(\mathbf{x}_i, y_i) \in \mathbf{D}^T : y_i = -1\}$ to denote points in that group belonging to the negative class. Analogous notation is used for points in the control group. Denote $n = |\mathbf{D}^T| + |\mathbf{D}^C|$.

The parameters of an USVM can be found by solving the following optimization problem, which we call the *USVM optimization problem*.

$$\begin{aligned} \min_{\mathbf{w}, b_1, b_2 \in \mathbb{R}^{m+2}} \frac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle + C_1 \sum_{\mathbf{D}^T_+ \cup \mathbf{D}^C_-} \xi_{i,1} + C_2 \sum_{\mathbf{D}^T_- \cup \mathbf{D}^C_+} \xi_{i,1} \\ + C_2 \sum_{\mathbf{D}^T_+ \cup \mathbf{D}^C_-} \xi_{i,2} + C_1 \sum_{\mathbf{D}^T_- \cup \mathbf{D}^C_+} \xi_{i,2}, \end{aligned} \tag{3}$$

subject to the following constraints

$$\langle \mathbf{w}, \mathbf{x}_i \rangle - b_1 \geq +1 - \xi_{i,1}, \text{ forall } (\mathbf{x}_i, y_i) \in \mathbf{D}^T_+ \cup \mathbf{D}^C_-, \tag{4}$$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle - b_1 \leq -1 + \xi_{i,1}, \text{ forall } (\mathbf{x}_i, y_i) \in \mathbf{D}^T_- \cup \mathbf{D}^C_+, \tag{5}$$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle - b_2 \geq +1 - \xi_{i,2}, \text{ forall } (\mathbf{x}_i, y_i) \in \mathbf{D}^T_+ \cup \mathbf{D}^C_-, \tag{6}$$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle - b_2 \leq -1 + \xi_{i,2}, \text{ forall } (\mathbf{x}_i, y_i) \in \mathbf{D}^T_- \cup \mathbf{D}^C_+, \tag{7}$$

$$\xi_{i,j} \geq 0, \text{ forall } i = 1, \ldots, n, \ j \in \{1, 2\}, \tag{8}$$

where C_1, C_2 are penalty parameters and $\xi_{i,j}$ slack variables allowing for misclassified training cases. Note that $\xi_{i,1}$ and $\xi_{i,2}$ are slack variables related to the hyperplane H_1 and H_2 respectively. We will now give an intuitive justification for this formulation of the optimization problem.

Below, when we talk about distance of a point from a plane and point lying on a positive or negative side of the plane we implicitly assume that the width of the margin is also taken into account.

The situation is graphically depicted in Figure 1. Example points belonging to $\mathbf{D}_+^T$ are marked with T_+, points belonging to $\mathbf{D}_-^T$, respectively with T_-. Analogous notation is used for example points in the control group which are marked with C_+ and C_-.

In an ideal situation, points for which a positive $(+1)$ prediction is made include only cases in $\mathbf{D}_+^T$ and $\mathbf{D}_-^C$, that is points which do not contradict the positive effect of the action. Note that for the remaining points, which are in $\mathbf{D}_-^T$ or in $\mathbf{D}_+^C$, the effect of an action can at best be neutral. Therefore points in $\mathbf{D}_+^T$ and $\mathbf{D}_-^C$ (marked T_+ and C_- respectively in the figure) are not penalized when on the positive side of hyperplane H_1. Analogously points in $\mathbf{D}_-^T$ and $\mathbf{D}_+^C$ (marked T_- and C_+) which are on the negative side of H_2 are not penalized.

Points in $\mathbf{D}_+^T$ and $\mathbf{D}_-^C$ which lie on the negative side of H_1 are penalized with penalty $C_1\xi_{i,1}$ where ξ_i is the distance of the point from the plane and C_1 is a penalty coefficient. Those penalties prevent the model from being overly cautious and classifying all points as neutral (see Lemmas 2 and 3 in the next section). Analogous penalty is introduced for points in $\mathbf{D}_-^T$ and $\mathbf{D}_+^C$ in the fifth term of (3). In Figure 1, those points are sandwiched between H_1 and H_2, and their penalties are marked with red arrows.

Consider now points in $\mathbf{D}_+^T$ and $\mathbf{D}_-^C$ which lie on the negative side of both hyperplanes, i.e. in the region where the model predicts a negative impact (-1). Clearly, model's predictions are wrong in this case, since if the outcome was positive in the treatment group the impact of the action can only be positive or neutral. Those data points are thus additionally penalized for being on the wrong side of the hyperplane H_2 with penalty $C_2\xi_{i,2}$. Analogous penalty is of course applied to points in $\mathbf{D}_-^T$ and $\mathbf{D}_+^C$ which lie on the positive side of both hyperplanes. Additional penalties are marked with dashed blue arrows in the figure.

To summarize, the penalty coefficient C_1 is used to punish points being on the wrong side of a single hyperplane (red arrows in Figure 1) and the coefficient C_2 controls additional penalty incurred by a point being on the wrong side of also the second hyperplane (dashed blue arrows in Figure 1). In the next section we give a more detailed analysis of how the penalties influence the model's behavior.

III. Properties of the Uplift Support Vector Machines (USVMs)

In this section we are going to analyze some mathematical properties of Uplift Support Vector Machines (USVMs), especially in the context of influence of the parameters C_1 and C_2 on model's behavior. One of the more important results is how the ratio of penalty parameters $\frac{C_2}{C_1}$ directly *influences* the number of records which are classified as neutral, or, in other words, how it influences the distance between the two separating hyperplanes. This also sheds light on the interpretation of the model.

Lemma 1: Let $\mathbf{w}^*, b_1^*, b_2^*$ be a solution to the Uplift SVM optimization problem given by Equations 3-8. If $C_2 \geq C_1$ then $b_1^* \geq b_2^*$.

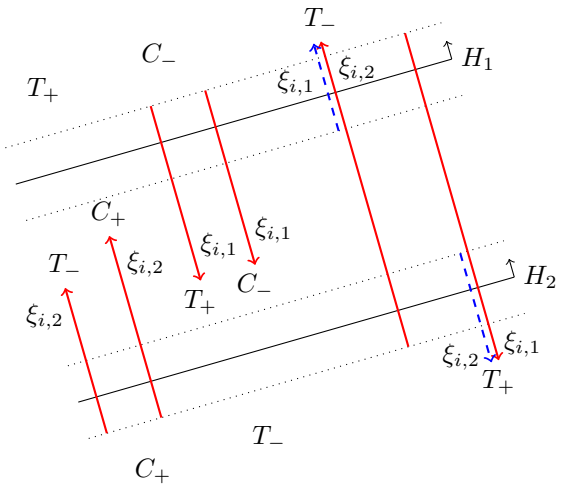

Fig. 1. The Uplift SVM optimization problem. Example points belonging to the positive class in the treatment and control groups are marked respectively with T_+ and C_+. Analogous notation is used for points in the negative class. The figure shows penalties incurred by points with respect to the two hyperplanes of the USVM. Positive sides of hyperplanes are indicated by small arrows at the right ends of lines in the image. Red arrows denote the penalties incurred by points which lie on the wrong side of a single hyperplane, blue dashed arrows denote additional penalties for being misclassified also by the second hyperplane.

The proof of this and the remaining lemmas can be found in the Appendix. The lemma guarantees that the problem possesses a well defined solution in the sense of Equation 2. Moreover it naturally constrains the penalty C_2 to be greater than or equal to C_1. From now on, instead of working with the coefficient C_2, it will be more convenient to talk about the penalty coefficient C_1 and the quotient $\frac{C_2}{C_1} \geq 1$ determining how many times is C_2 is greater than C_1.

Lemma 2: For sufficiently large value of $\frac{C_2}{C_1}$ none of the observations is penalized with a term involving the C_2 factor in the solution to the USVM optimization problem.

Equivalently the lemma states that for a large enough value of $\frac{C_2}{C_1}$, none of the points will be on the wrong side of both hyperplanes. This is possible only when the hyperplanes are maximally separated, resulting in most (often all) points classified as neutral.

Lemma 3: If $C_1 = C_2 = C$ and the solution is unique then both hyperplanes coincide: $b_1 = b_2$.

We are now ready to give an interpretation of the C_1 and $\frac{C_2}{C_1}$ parameters of the Uplift SVM. The parameter C_1 plays the role analogous to the penalty coefficient C in classical SVMs controlling the relative cost of misclassified points with respect to the margin maximization term $\frac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle$. The quotient $\frac{C_2}{C_1}$ allows the analyst to decide what proportion of points should be classified as positive or negative. In other words, it allows for controlling the size of the neutral prediction.

Note that this is *not* equivalent to selecting thresholds in data scored using a single model. For each value of $\frac{C_2}{C_1}$ a different model is built which is optimized for a specific proportion of positive and negative predictions. We believe that this property

of USVMs is very useful for practical applications, as it allows for tuning the model specifically to the desired size of the campaign.

IV. The Uplift Support Vector Machine optimization task

Let us now present the dual of the Uplift Support Vector Machine optimization task and discuss methods of solving it.

We will first introduce a class variable transformation

$$z_i = \begin{cases} y_i, & \text{if } (\mathbf{x}_i, y_i) \in \mathbf{D}^T, \\ -y_i, & \text{if } (\mathbf{x}_i, y_i) \in \mathbf{D}^C. \end{cases}$$

In other words, z_i is obtained by keeping the class variable in the treatment group and reversing it in the control. Note that this is the same transformation which has been introduced in [12] in the context of uplift modeling and logistic regression.

This variable transformation allows us to simplify the optimization problem given in Equations 3-8 by merging (4) with (5) and (6) with (7). The simplified optimization problem is

$$\min_{\mathbf{w},b_1,b_2 \in \mathbb{R}^{m+2}} \frac{1}{2}\langle \mathbf{w}, \mathbf{w}\rangle + C_1 \sum_{\mathbf{D}_+^T \cup \mathbf{D}_-^C} \xi_{i,1} + C_2 \sum_{\mathbf{D}_-^T \cup \mathbf{D}_+^C} \xi_{i,1} + C_2 \sum_{\mathbf{D}_+^T \cup \mathbf{D}_-^C} \xi_{i,2} + C_1 \sum_{\mathbf{D}_-^T \cup \mathbf{D}_+^C} \xi_{i,2},$$

subject to constraints

$$z_i(\langle \mathbf{w}, \mathbf{x}_i\rangle - b_1) - 1 + \xi_{i,1} \geq 0 \text{ forall } i = 1, \ldots, n,$$
$$z_i(\langle \mathbf{w}, \mathbf{x}_i\rangle - b_2) - 1 + \xi_{i,2} \geq 0 \text{ forall } i = 1, \ldots, n,$$
$$\xi_{i,j} \geq 0, \text{ forall } i = 1, \ldots, n,\ j \in \{1, 2\}.$$

We will now obtain the dual form of the optimization problem. We begin by writing the following Lagrange function

$$\begin{aligned} L(\mathbf{w}, b_1, b_2, \alpha_i, \beta_i, \xi_{i,1}, \xi_{i,2}, r_i, p_i) &= \frac{1}{2}\langle \mathbf{w}, \mathbf{w}\rangle + C_1 \sum_{\mathbf{D}_+^T \cup \mathbf{D}_-^C} \xi_{i,1} + C_2 \sum_{\mathbf{D}_-^T \cup \mathbf{D}_+^C} \xi_{i,1} \\ &+ C_2 \sum_{\mathbf{D}_+^T \cup \mathbf{D}_-^C} \xi_{i,2} + C_1 \sum_{\mathbf{D}_-^T \cup \mathbf{D}_+^C} \xi_{i,2} \\ &- \sum_{i=1}^n \alpha_i \big(z_i(\langle \mathbf{w}, \mathbf{x}_i\rangle - b_1) - 1 + \xi_{i,1}\big) \\ &- \sum_{i=1}^n \beta_i \big(z_i(\langle \mathbf{w}, \mathbf{x}_i\rangle - b_2) - 1 + \xi_{i,2}\big) \\ &- \sum_{i=1}^n r_i \xi_{i,1} - \sum_{i=1}^n p_i \xi_{i,2}, \end{aligned}$$

where $\alpha_i, \beta_i \in \mathbb{R}$ are Lagrange multipliers and $r_i, p_i \geq 0$.

Now we need to calculate partial derivatives and equate them to 0 in order to satisfy Karush-Kuhn-Tucker conditions. We begin by deriving w.r.t. $\mathbf{w}$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^n \alpha_i z_i \mathbf{x}_i - \sum_{i=1}^n \beta_i z_i \mathbf{x}_i = 0,$$

from which we obtain

$$\mathbf{w} = \sum_{i=1}^n (\alpha_i + \beta_i) z_i \mathbf{x}_i. \tag{9}$$

We obtain the remaining derivatives in a similar fashion

$$\frac{\partial L}{\partial b_1} = \sum_{i=1}^n \alpha_i z_i = 0, \qquad \frac{\partial L}{\partial b_2} = \sum_{i=1}^n \beta_i z_i = 0, \tag{10}$$

$$\frac{\partial L}{\partial \xi_{i,1}} = C_1 \mathbb{1}_{[z_i=+1]} + C_2 \mathbb{1}_{[z_i=-1]} - \alpha_i - r_i = 0, \tag{11}$$

$$\frac{\partial L}{\partial \xi_{i,2}} = C_1 \mathbb{1}_{[z_i=-1]} + C_2 \mathbb{1}_{[z_i=+1]} - \beta_i - p_i = 0. \tag{12}$$

Plugging Equations 11, 12 back into the Lagrange function we obtain, after simplifications,

$$\begin{aligned} L = \frac{1}{2}\langle \mathbf{w}, \mathbf{w}\rangle &- \sum_{i=1}^n \alpha_i \big(z_i(\langle \mathbf{w}, \mathbf{x}_i\rangle - b_1) - 1\big) \\ &- \sum_{i=1}^n \beta_i \big(z_i(\langle \mathbf{w}, \mathbf{x}_i\rangle - b_2) - 1\big). \end{aligned}$$

Substituting $\mathbf{w}$ from Equation 9 and using Equation 10 we get

$$\begin{aligned} L =& \frac{1}{2} \sum_{i,j=1}^n (\alpha_i + \beta_i)(\alpha_j + \beta_j) z_i z_j \langle \mathbf{x}_i, \mathbf{x}_j\rangle \\ &- \sum_{i,j=1}^n (\alpha_i + \beta_i)(\alpha_j + \beta_j) z_i z_j \langle \mathbf{x}_j, \mathbf{x}_i\rangle \\ &+ b_1 \sum_{i=1}^n \alpha_i z_i + \sum_{i=1}^n \alpha_i + b_2 \sum_{i=1}^n \beta_i z_i + \sum_{i=1}^n \beta_i \\ =& \sum_{i=1}^n (\alpha_i + \beta_i) - \frac{1}{2} \sum_{i,j=1}^n (\alpha_i + \beta_i)(\alpha_j + \beta_j) z_i z_j \langle \mathbf{x}_i, \mathbf{x}_j\rangle, \end{aligned} \tag{13}$$

which we maximize over α_i, β_i.

Finally, from the assumption that $r_i, p_i \geq 0$ and (11), (12) combined with the KKT condition on nonnegativity of α_i, β_i and from (10) we obtain the following constraints for the dual optimization problem

$$0 \leq \alpha_i \leq C_1 \mathbb{1}_{[z_i=+1]} + C_2 \mathbb{1}_{[z_i=-1]}, \tag{14}$$

$$0 \leq \beta_i \leq C_1 \mathbb{1}_{[z_i=-1]} + C_2 \mathbb{1}_{[z_i=+1]}, \tag{15}$$

$$\sum_{i=1}^n \alpha_i z_i = \sum_{i=1}^n \beta_i z_i = 0. \tag{16}$$

A. The optimization algorithm

The optimization problem presented above can be solved using off the shelf constrained optimization software or using methods designed specifically for Support Vector Machines. We have adapted to our problem the dual coordinate descent method [15] used in the LIBLINEAR package which is currently the most popular method of solving SVM-type optimization problems. Details are omitted due to lack of space and the fact that the algorithm is very similar to the one in [15]. Our final approach uses the CVXOPT convex solver [16] for

problems of up to 500 records, and for larger problems the dual coordinate descent method.

V. Experimental evaluation

In this section we present an experimental evaluation of the proposed Uplift Support Vector Machines. We begin with an illustrative example showing the approach applied to two datasets. Later, we present an experimental comparison with other uplift modeling methods on several benchmark datasets.

While testing uplift modeling algorithms one encounters the problem of the lack of publicly available datasets. Even though control groups are ubiquitous in medicine and become common in marketing, there are very few publicly available datasets which include a control group as well as a reasonable number of predictive attributes. In this paper we will use the few publicly available datasets we are aware of, as well as some artificially generated examples based on datasets from the UCI repository. We describe the two approaches in turn.

The first publicly available dataset, provided on Kevin Hillstrom's MineThatData blog, contains results of an e-mail campaign for an Internet based retailer [17]. The dataset contains information about 64 000 customers. The customers have been randomly split into three groups: the first received an e-mail campaign advertising men's merchandise, the second, a campaign advertising women's merchandise, and the third was kept as control. Data is available on whether a person visited the website and/or made a purchase (conversion). We only focus on visits since very few conversions actually occurred. In this paper we use the dataset in two ways: combining both e-mailed groups into a single treatment group (`Hillstrom-visit`) and using only the women's merchandise group (`Hillstrom-visit-w`).

Additionally, we found two suitable publicly available clinical trial datasets which accompany a book on survival analysis [18]. Unfortunately, very few predictive attributes are present which limits their usefulness.

The first dataset is the Bone Marrow Transplant (`BMT`) data which covers patients who received two types of bone marrow transplant: taken from the pelvic bone (which we used as the control group since this is the procedure commonly used at the time the data was created) or from the peripheral blood (a novel approach, used as the treatment group in this paper). The peripheral blood transplant is generally the preferred treatment, so minimizing its side effects is highly desirable. There are only three randomization time variables available: the type and extent of the disease, as well as patients age. There are two target variables representing the occurrence of the chronic (`cgvh`) and acute (`agvh`) graft versus host disease. We ignore the survival analysis nature of the data and simply treat nonoccurrence as the successful outcome.

Note that even though the `BMT` dataset does not, strictly speaking, include a control group, uplift modeling can still applied. The role of the control group is played by one of the treatments, and the method allows for selection of patients to whom an alternative treatment should be applied.

The second clinical trial dataset (`tamoxifen`) we analyze comes from the study of treatment of breast cancer with tamoxifen. The control group received tamoxifen alone and the treatment group tamoxifen combined with radio therapy. We attempt to model the variable `stat` describing whether the patient was alive at the time of the last follow-up. The dataset contains six variables. Here we again ignore the survival character of the data.

As can be seen, there are very few real uplift datasets available, moreover, they all have a limited number of attributes (up to 10) and/or data. In [5] an approach has been proposed to split standard UCI datasets into treatment and control groups suitable for uplift modeling. The conversion is performed by first picking one of the data attributes which either has a causal meaning or splits the data evenly into two groups. As a postprocessing step, attributes strongly correlated with the split are removed (ideally, the division into treatment and control groups should be independent from all predictive attributes, but this is possible only in a controlled experiment). Multiclass problems are converted to binary problems with the majority class considered to be $+1$ and remaining classes -1. The procedure is described in detail in [5], where a table is given with the exact conditions used to split the data.

A. An illustrative example

We will first illustrate how the method behaves on two example datasets: the tamoxifen trial data (`tamoxifen`) and the `credit-a` dataset from the UCI repository. More specifically, we are going to show how the choice of the parameter $\frac{C_2}{C_1}$ affects model behavior. Since this section has mainly illustrative purpose, all curves are drawn based on the full dataset; more rigorous experiments involving test sets are given in Section V-B.

Figure 2 shows the number of cases classified as positive, neutral and negative depending on the quotient $\frac{C_2}{C_1}$ for the two datasets. The numbers shown were obtained on the full dataset and are averages of respective numbers of cases in treatment and control groups. The parameter C_1 was set to 0.1, but for other values we obtained very similar results.

It can clearly be seen that for low values of the quotient, the neutral class is practically empty, but as the quotient increases, more and more cases are classified as neutral. Finally, almost no cases are classified as positive or negative. The figure validates our interpretation presented earlier in Lemmas 1-3. The analyst can use the parameter $\frac{C_2}{C_1}$ to control the proportion of negative and positive predictions and tune the model to those values.

It is worth noting that the model is quite sensitive to the value of $\frac{C_2}{C_1}$, and the interval between the extreme cases of no and almost all predictions being neutral can be quite narrow. Currently this issue is solved by checking model behavior for various values of the parameter for a single value of $C_1 = 0.1$ and, after finding good boundary points, selecting values only from that interval (see also the discussion on parameter tuning below). Finding a more convenient solution is left as future research.

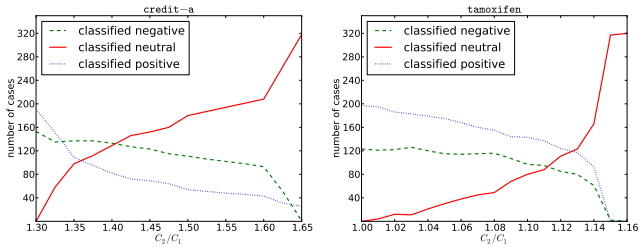

Fig. 2. Number of cases classified as positive, neutral and negative as a function of the quotient $\frac{C_2}{C_1}$ of USVM penalty coefficients for the `credit-a` and `tamoxifen` datasets.

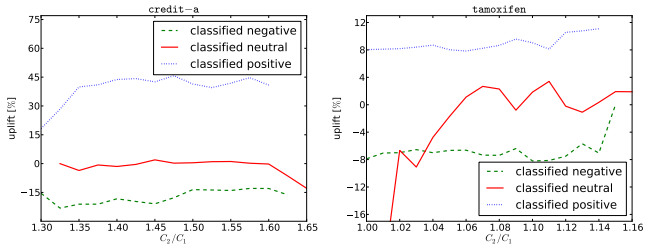

Fig. 3. The difference between success probabilities in treatment and control groups within cases predicted respectively as neutral, positive and negative as a function of $\frac{C_2}{C_1}$.

Figure 3 shows the uplift, i.e. the difference between success probabilities (the $+1$ class) in the treatment and control groups for those two datasets as a function of the quotient $\frac{C_2}{C_1}$.

It can be seen that, indeed, within the records predicted as positive, the probability of success (the $+1$ class) is larger in the treatment group than in the control group. The reverse is true for negative predictions. For cases predicted to be neutral, the difference in class probability between the groups indeed oscillates around zero. Note that towards the ends of the charts the estimates were based on very little data (see Figure 2) resulting in sudden jumps or dips in the values of uplift, such as the one visible in the neutral group for the `tamoxifen` dataset. The figure demonstrates that our method does indeed make correct uplift predictions (in the sense of Equation 1).

B. Comparison on benchmark datasets

Let us now discuss evaluation of uplift models using so called uplift curves. One of the tools for assessing performance of standard classification models are lift curves (also known as cumulative gains curves or cumulative accuracy profiles). In a lift curve, the x axis corresponds to the number of cases targeted and the y axis to the number of successes captured by the model. In our case both numbers will be expressed as percentage of the total population.

The *uplift curve* is computed by subtracting the lift curve obtained on the control test set from the lift curve obtained on the treatment test set. Both curves are generated using the same uplift model. Recall the number of successes on the y axis is expressed as a percentage of the total population which guarantees that the curves can be meaningfully subtracted. The interpretation of the uplift curve is as follows: on the x axis

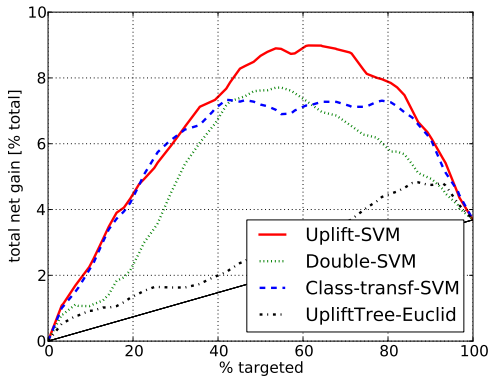

Fig. 4. Uplift curves for the `breast-cancer` dataset for Uplift SVM, the double SVM approach, SVM uplift model based on class variable transformation (`Class-transf-SVM`), and Euclidean distance based uplift decision trees (`UpliftTree-Euclid`). The x-axis represents the percentage of the population to which the action has been applied and the y-axis the net gain from performing the action. It can be seen that targeting about 50% of the population gives significant gains over targeting nobody or the whole population. The proposed Uplift SVM model achieves the best performance over the whole range of the plot.

we select the percentage of the population on which an action is performed and on the y axis we read the difference between the success rates in the treatment and control groups. A point at $x = 100\%$ gives the gain in success probability we would obtain if the action was performed on the whole population. A diagonal line corresponds random selection. The Area Under the Uplift Curve (AUUC) can be used as a single number summarizing model performance. In this paper we subtract the area under the diagonal line from this value in order to obtain more meaningful numbers. More details on evaluating uplift models and on uplift curves can be found in [5], [2].

Figure 4 shows uplift curves for the `breast-cancer` dataset for four different uplift models used in the comparison (see below). It can be seen that applying the action only to some proportion of the population leads to significant gains in net success rate. The curves in the figure have been generated by averaging over 128 random train test splits; the same method has been used for other experiments in this section and is described in detail below.

We will now compare the performance of Uplift Support Vector Machines (`Uplift-SVM`) and three other uplift modeling methods on several benchmark datasets. Two of the approaches are also based on Support Vector Machines: the method based on building two separate SVM models (`Double-SVM`) on treatment and control groups and subtracting their predicted probabilities as well as a single Support Vector Machine adapted to uplift modeling using the class variable transformation proposed in [12] (`Class-transf-SVM`). Since both those methods require probabilities to be predicted, the SVMs have been calibrated by training logistic regression models on their outputs. The fourth method used in comparison are uplift decision trees based on the Euclidean distance splitting criterion introduced

in [5], referred to as `UpliftTree-Euclid`.

The parameters of all SVM based models have been chosen using 5-fold cross-validation. The parameter C for classical SVMs was chosen from the set $\{10^{-2}, 10^{-1}, \ldots, 10^{5}\}$.

For Uplift Support Vector Machines the parameter choice was more difficult due to high sensitivity to the value of parameter $\frac{C_2}{C_1}$. We used a pre-tuning step to select the value range for this parameter for a fixed value of $C_1 = 0.1$; the range was later used also for other values of C_1. The endpoints were the two extreme values of $\frac{C_2}{C_1}$ for which almost all and almost no predictions become neutral. Those values were then used endpoints of the interval on which grid search over both parameters was performed. The parameter C_1 was selected from the set $\{10^{-2}, 10^{-1}, \ldots, 10^{3}\}$. For each grid point 5-fold cross-validation was used to measure model performance.

Table I compares Areas under the Uplift Curve for Uplift SVMs against the remaining three uplift modeling approaches on all benchmark datasets. The areas are given in terms of percentages of the total population (used also on the y-axis). Testing was performed by repeating 128 times a random train/test split with 80% of data used for training (and cross-validation based parameter tuning). The remaining 20% were used for testing. Cases when a given method is better than the proposed Uplift SVM are marked in bold. The last row of the table lists the number of times Uplift-SVM was better than each respective method.

Uplift SVM consistently outperforms the method based on two separate SVM models (better performance on 13 out of 18 datasets) as well as the uplift decision trees (on 12 of 18 datasets). It's performance is on par with the method using SVMs with class variable transformation proposed in [12] which it outperforms on roughly half of the datasets. That is, there seems to be roughly equal chance than either of the methods will perform better on a new dataset. We believe that those results clearly demonstrate that USVMs are a useful addition to the uplift modeling toolbox.

Overall, our method performs comparably to or better than current state-of-the-art uplift modeling methods. We also believe, that other advantages of the proposed Uplift SVMs are equally important. For example, it allows for natural prediction of cases with positive, negative and neutral outcomes (as shown in Section V-A) which is very useful in practice. Especially the negative group is important from the point of view of practical applications. Being able to detect this group and refraining from targeting it was crucial for many successful marketing campaigns. Additionally, through the choice of the parameter $\frac{C_2}{C_1}$ the analyst is able to decide how conservative should the model be when selecting those groups.

VI. Conclusions and future research

We have presented Uplift Support Vector Machines, an adaptation of the SVM methodology to the uplift modeling problem. The proposed method has been analyzed theoretically, whence it has been demonstrated that by an appropriate choice of model parameters, one is able to tune how conservative the model is in declaring a positive or negative impact of an action. The dual coordinate descent optimization method has been adapted to solve the corresponding optimization task. Future research will include adapting further SVM optimization methods to the problem as well as a theoretical analysis of the generalization properties.

VII. Acknowledgements

This work was supported by Research Grant no. N N516 414938 of the Polish Ministry of Science and Higher Education (Ministerstwo Nauki i Szkolnictwa Wyższego) from research funds for the period 2010–2013. Ł.Z. was co-funded by the European Union from resources of the European Social Fund. Project POKL 'Information technologies: Research and their interdisciplinary applications', Agreement UDA-POKL.04.01.01-00-051/10-00.

References

[1] B. Hansotia and B. Rukstales, "Incremental value modeling," *Journal of Interactive Marketing*, vol. 16, no. 3, pp. 35–46, 2002.

[2] N. Radcliffe and P. Surry, "Real-world uplift modelling with significance-based uplift trees," Stochastic Solutions, Portrait Technical Report TR-2011-1, 2011.

[3] D. M. Chickering and D. Heckerman, "A decision theoretic approach to targeted advertising," in *UAI*, Stanford, CA, 2000, pp. 82–88.

[4] N. J. Radcliffe and P. D. Surry, "Differential response analysis: Modeling true response by isolating the effect of a single action," in *Proceedings of Credit Scoring and Credit Control VI*. Credit Research Centre, University of Edinburgh Management School, 1999.

[5] P. Rzepakowski and S. Jaroszewicz, "Decision trees for uplift modeling," in *Proc. of the 10th IEEE International Conference on Data Mining (ICDM)*, Sydney, Australia, Dec. 2010, pp. 441–450.

[6] ——, "Decision trees for uplift modeling with single and multiple treatments," *Knowledge and Information Systems*, 2011.

[7] J. Robins, "Correcting for non-compliance in randomized trials using structural nested mean models," *Communications in Statistics - Theory and Methods*, vol. 23, no. 8, pp. 2379–2412, 1994.

[8] J. Robins and A. Rotnitzky, "Estimation of treatment effects in randomised trials with non-compliance and a dichotomous outcome using structural mean models," *Biometrika*, vol. 91, no. 4, pp. 763–783, 2004.

[9] S. Vansteelandt and E. Goetghebeur, "Causal inference with generalized structural mean models," *Journal of the Royal Statistical Society B*, vol. 65, no. 4, pp. 817–835, 2003.

[10] V. S. Y. Lo, "The true lift model - a novel data mining approach to response modeling in database marketing," *SIGKDD Explorations*, vol. 4, no. 2, pp. 78–86, 2002.

[11] K. Larsen, "Net lift models: Optimizing the impact of your marketing," in *Predictive Analytics World*, 2011, workshop presentation.

[12] M. Jaśkowski and S. Jaroszewicz, "Uplift modeling for clinical trial data," in *ICML 2012 Workshop on Machine Learning for Clinical Data Analysis*, Edinburgh, Scotland, Jun. 2012.

[13] D. Pechyony, R. Jones, and X. Li, "A joint optimization of incrementality and revenue to satisfy both advertiser and publisher," in *WWW 2013 Companion*, 2013.

[14] A. Shashua and A. Levin, "Ranking with large margin principle: Two approaches," *Advances in neural information processing systems*, vol. 15, pp. 937–944, 2002.

[15] C.-J. Hsieh, K.-W. Chang, C.-J. Lin, S. Keerthi, and S. Sundararajan, "A dual coordinate descent method for large-scale linear SVM," in *Proc. of the 25-th International Conference on Machine Learning (ICML)*, Helsinki, Finland, 2008.

[16] M. S. Andersen, J. Dahl, Z. Liu, and L. Vandenberghe, "Interior-point methods for large-scale cone programming," in *Optimization for Machine Learning*. MIT Press, 2012, pp. 55–83.

[17] K. Hillstrom, "The MineThatData e-mail analytics and data mining challenge," MineThatData blog, http://blog.minethatdata.com/2008/03/minethatdata-e-mail-analytics-and-data.html, 2008, retrieved on 02.04.2012.

[18] M. Pintilie, *Competing risks : a practical perspective*. John Wiley & Sons Inc., 2006.

TABLE I
AREAS UNDER THE UPLIFT CURVE FOR FOUR UPLIFT MODELS ON REAL AND ARTIFICIAL DATASETS.

dataset	Uplift-SVM	Double-SVM	Class-transf-SVM	UpliftTree-Euclid
hepatitis	1.68	0.04	**2.2**	-0.64
breast cancer	4.48	3.13	3.87	0.86
australian	0.14	**0.65**	**1.28**	-0.57
diabetes	-0.28	**0.78**	**0.89**	**1.14**
dermatology	7.82	5.51	**8.02**	7.02
credit-a	6.46	1.14	5.71	-1.51
heart-c	2.57	0.15	2.51	1.78
labor	-0.43	-0.84	-5.52	-1.84
liver disorders	0.9	**3.84**	**2.4**	**2.28**
splice	2.17	2.16	**2.19**	**3.73**
hypothyroid	2.84	2.51	0.06	**2.88**
diagnosis	14.83	-0.87	13.93	13.61
primary-tumor	4.17	0.92	**4.52**	-1.07
BMT-agvh	-2.99	-2.03	**-1.84**	**-0.94**
BMT-cgvh	4.57	**5.61**	3.56	2.32
tamoxifen	1.41	**1.80**	**1.75**	0.32
Hillstrom-visit	0.32	0.21	**0.39**	**0.40**
Hillstrom-visit-w	0.69	0.66	**0.73**	0.66
Uplift SVM win/total		13/18	7/18	12/18

APPENDIX

Let us begin with an observation which will be used in the proofs. Consider the Uplift SVM optimization problem given by Equations 3-8. Notice that when $\mathbf{w}, b_1, b_2$ are fixed, the optimal values of slack variables $\xi_{i,j}$ are uniquely determined. Optimal values for slack variables present in Equation 4 are $\xi^*_{i,1} = \max\{0, -\langle \mathbf{w}, \mathbf{x}_i\rangle + b_1 + 1\}$, and for those present in Equation 5, $\xi^*_{i,1} = \max\{0, \langle \mathbf{w}, \mathbf{x}_i\rangle - b_1 + 1\}$. Analogous formulas can be given for $\xi^*_{i,2}$ and Equations 7-8.

Proof of Lemma 1: Let $S^* = \langle \mathbf{w}^*, b^*_1, b^*_2\rangle$ be an optimal solution with $b^*_1 < b^*_2$. Consider also a set of parameters $S' = \langle \mathbf{w}^*, b^*_2, b^*_1\rangle$ with the values of b^*_1, b^*_2 interchanged and look at the target function (3) for both sets of parameters.

Take a point $(\mathbf{x}_i, y_i) \in \mathbf{D}^T_+$ for which, under the set of parameters S', $\xi'_{i,1} > 0$ and $\xi'_{i,2} = 0$, that is the point is penalized only for crossing the hyperplane H_1. Under the parameters S^* the point will be penalized not with $C_1\xi^*_{i,1}$ for crossing H_1 but, instead, with $C_2\xi'_{i,2}$ for crossing H_2. Since, by switching from S^* to S' the hyperplanes simply exchange intercepts, we have $\xi^*_{i,1} = \xi'_{i,2}$ and, from the assumption, $C_2\xi^*_{i,1} > C_1\xi'_{i,2}$. Thus the amount every point $(\mathbf{x}_i, y_i) \in \mathbf{D}^T_+$ contributes to the target function (3) is lower in S' than in S^*.

By a similar argument one can see that for a point $(\mathbf{x}_i, y_i) \in \mathbf{D}^T_+$ for which, under S', $\xi'_{i,1}, \xi'_{i,2} > 0$ (i.e. it is penalized for crossing both hyperplanes) a switch to parameter set S^* increases the target function by $(b^*_2 - b^*_1)(C_2 - C_1)$.

Analogous arguments hold for points in $\mathbf{D}^T_-$, $\mathbf{D}^C_+$ and $\mathbf{D}^C_-$ contradicting the optimality of S^*. ∎

Proof of Lemma 2: Let us first consider only the hyperplane H_1. Assume that there exists at least one point in $\mathbf{D}^T_- \cup \mathbf{D}^C_+$ which is punished with a term involving the C_2 penalty coefficient, and therefore lies on the wrong side of H_1. Out of all such points choose the one $(\tilde{\mathbf{x}}_i, \tilde{y}_i)$ which is furthest from H_1 and denote by $\tilde{\xi}_{i,1}$, $\tilde{\xi}_{i,2}$ its slack variables w.r.t. H_1 and H_2 respectively. The penalty incurred by $(\tilde{\mathbf{x}}_i, \tilde{y}_i)$ equals

$$C_2\tilde{\xi}_{i,1} + C_1\tilde{\xi}_{i,2}.$$

Let us now shift the hyperplane H_1 by exactly $\tilde{\xi}_{i,1}$; as a result, the point is only penalized by $C_1\tilde{\xi}_{i,2}$. The same is true for all other points from $\mathbf{D}^T_- \cup \mathbf{D}^C_+$. On the other hand, after shifting H_1, penalties w.r.t. H_1 of points in $\mathbf{D}^T_+ \cup \mathbf{D}^C_-$ could have increased, but the increase is bounded by $C_1\tilde{\xi}_{i,1}$ per point.

Denote $n_1 = |\mathbf{D}^T_- \cup \mathbf{D}^C_+|$, $n_2 = |\mathbf{D}^T_+ \cup \mathbf{D}^C_-|$. The change in penalties caused by shifting H_1 is bounded from above by

$$C_1\tilde{\xi}_{i,2} - (C_2\tilde{\xi}_{i,1} + C_1\tilde{\xi}_{i,2}) + n_2C_1\tilde{\xi}_{i,1} = \tilde{\xi}_{i,1}(n_2C_1 - C_2),$$

which is negative for sufficiently large value of C_2, such that shifting H_1 is guaranteed to decrease the target function. ∎

Proof of Lemma 3: Let us fix any $\mathbf{w}$ and optimize with respect to b_1, b_2. Under the assumption of the lemma, the target function (3) can be rewritten as

$$\frac{1}{2}\langle \mathbf{w}, \mathbf{w}\rangle + C \sum_{\mathbf{D}^T \cup \mathbf{D}^C} \xi_{i,1} + C \sum_{\mathbf{D}^T \cup \mathbf{D}^C} \xi_{i,2}.$$

Note, that the first term is constant, the second is a function of b_1 and the third of b_2. Moreover the second and third term are fully symmetric so the target function can be rewritten as $const. + f(b_1) + f(b_2)$, where f is some function of b_1 or b_2. Notice that optimization over b_1 is done independently of optimization over b_2 and since the optimized functions f are identical, the resulting optima for b_1 and b_2 must be identical if the solution is unique. ∎