Uplift modeling

Szymon Jaroszewicz

Institute of Computer Science
Polish Academy of Sciences
Warsaw, Poland

National Institute of Telecommunications
Warsaw, Poland

Joint work with:

- Piotr Rzepakowski
- Maciej Jaśkowski
- Łukasz Zaniewicz
- Michał Sołtys

What is uplift modeling?

A typical marketing campaign

Sample → Pilot campaign → Model $P(buy|\mathbf{X})$ → Select targets for campaign

What is uplift modeling?

A typical marketing campaign

Sample → Pilot campaign → Model $P(buy|\mathbf{X})$ → Select targets for campaign

- But this is not what we need!

- We want people who bought because of the campaign
- Not people who bought after the campaign

A typical marketing campaign

We can divide potential customers into four groups

1. Responded **because** of the action
   (**the people we want**)
2. Responded, but would have responded **anyway**
   (**unnecessary costs**)
3. Did not respond and the action had **no impact**
   (**unnecessary costs**)
4. Did not respond **because** the action had a
   (**negative impact**)

Solution: Uplift modeling

- Solution: Uplift modeling
- Two training sets:
  1. the treatment group
     on which the action was taken
  2. the control group
     on which no action was taken
     used as background
- Build a model which predicts the difference between class probabilities in the treatment and control groups

Difference with traditional classification

Notation:

- P^T probabilities in the treatment group
- P^C probabilities in the control group

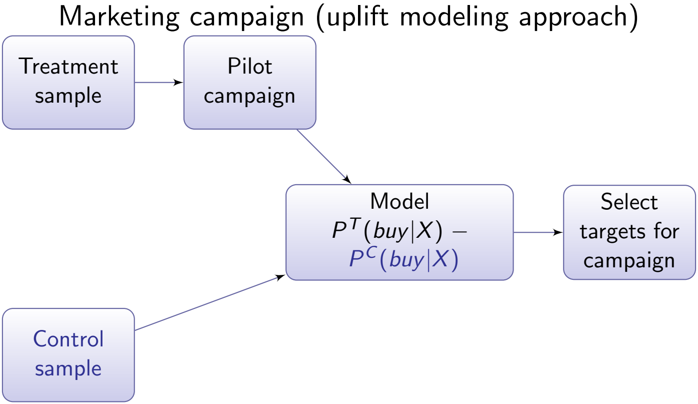
Traditional models predict the conditional probability

$$P^T(Y \mid X_1, \ldots, X_m)$$

Uplift models predict change in behaviour resulting from the action

$$P^T(Y \mid X_1, \ldots, X_m) - P^C(Y \mid X_1, \ldots, X_m)$$

Marketing campaign (uplift modeling approach)



Applications in medicine

- A typical medical trial:
  - treatment group: gets the treatment
  - control group: gets placebo (or another treatment)
  - do a statistical test to show that the treatment is better than placebo
- With uplift modeling we can find out for whom the treatment works best
- Personalized medicine

Main difficulty of uplift modeling

The fundamental problem of causal inference

- Our knowledge is always incomplete
- For each training case we know either
  - what happened after the treatment, or
  - what happened if no treatment was given
- Never both!

- This makes designing uplift algorithms challenging
- ... and the intuitions are often hard to grasp

Evaluating uplift models

Evaluating uplift models

- We have two separate test sets:
  - a treatment test set
  - a control test set

Problem

To assess the gain for a customer we need to know both treatment and control responses, but only one of them is known

Solution

Assess gains for groups of customers

Evaluating uplift classifiers

For example:

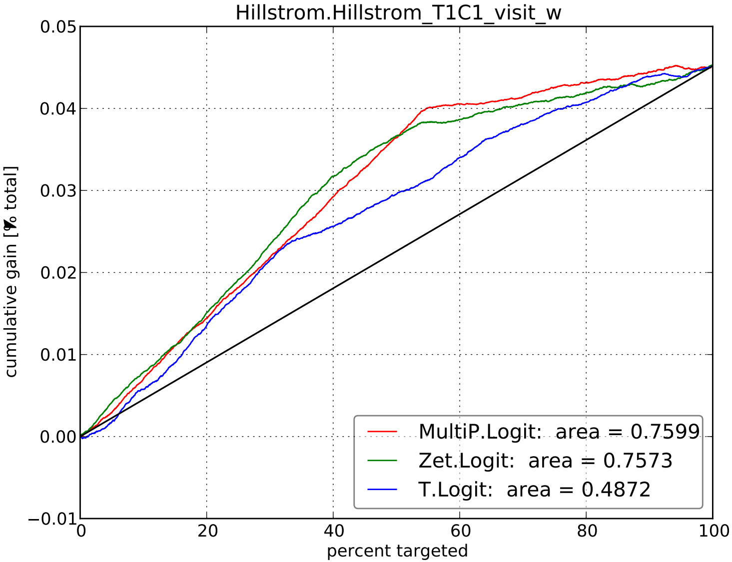
$$\begin{aligned}\text{Gain for the 10\% highest scoring customers} = &\\ &\text{\% of successes for top 10\% treated customers}\\ -\ &\text{\% of successes for top 10\% control customers}\end{aligned}$$

Uplift curves

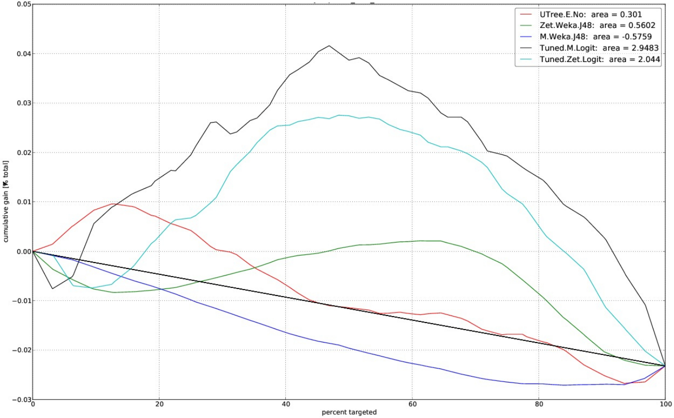
- Uplift curves are a more convenient tool:
  - Draw separate lift curves on treatment and control data (TPR on the Y axis is replaced with percentage of successes in the whole population)
  - Uplift curve =

    lift curve on treatment data – lift curve on control data
  - Interpretation: net gain in success rate if a given percentage of the population is treated
- We can of course compute the Area Under the Uplift Curve (AUUC)

An uplift curve for marketing data



An uplift curve for a medical trial dataset comparing two cancer treatments



Uplift modeling algorithms

The two model approach

An obvious approach to uplift modeling:

1. Build a classifier M^T modeling $P^T(Y|\mathbf{X})$ on the treatment sample
2. Build a classifier M^C modeling $P^C(Y|\mathbf{X})$ on the control sample
3. The uplift model subtracts probabilities predicted by both classifiers

$$M^U(Y|\mathbf{X}) = M^T(Y|\mathbf{X}) - M^C(Y|\mathbf{X})$$

Two model approach

Advantages:

- Works with existing classification models
- Good probability predictions $\Rightarrow$ good uplift prediction

Disadvantages:

- Differences between class probabilities can follow a different pattern than the probabilities themselves
  - each classifier focuses on changes in class probabilities but ignores the weaker 'uplift signal'
  - algorithms designed to focus directly on uplift can give better results

Decision trees for uplift modeling

Hansotia, Rukstales 2002

The $\Delta\Delta P$ criterion

$$P^T(Y=1) = 5\%$$
$$P^C(Y=1) = 3\%$$
$$\Delta P = 2\%$$

$X < a$

$$P^T(Y=1) = 8\%$$
$$P^C(Y=1) = 3.5\%$$
$$\Delta P = 4.5\%$$

$X >= a$

$$P^T(Y=1) = 3.7\%$$
$$P^C(Y=1) = 2.8\%$$
$$\Delta P = 0.9\%$$

$$\Delta\Delta P = 3.6\%$$

Pick a test with highest $\Delta\Delta P$

Hansotia, Rukstales 2002

- It is not in line with modern decision tree learning such as C4.5
  - splitting criterion directly maximizes the difference between probabilities (target criterion)
  - no pruning
- Rzepakowski, Jaroszewicz 2010, 2012
  - splitting criterion based on Information Theory, more in line with modern decision trees
  - pruning designed for uplift modeling
  - multiclass problems and multiway splits possible
  - if the control group is empty, the algorithm reduces to classical decision tree learning

KL divergence as a splitting criterion for uplift trees

- Measure difference between treatment and control groups using KL divergence

$$KL\left(P^T(Y) : P^C(Y)\right) = \sum_{y \in \mathrm{Dom}(Y)} P^T(y) \log \frac{P^T(y)}{P^C(y)}$$

KL divergence as a splitting criterion for uplift trees

- Measure difference between treatment and control groups using KL divergence

$$KL\left(P^T(Y) : P^C(Y)\right) = \sum_{y \in \mathrm{Dom}(Y)} P^T(y) \log \frac{P^T(y)}{P^C(y)}$$

- KL-divergence conditional on a test X

$$KL(P^T(Y) : P^C(Y) \mid X) =$$
$$\sum_{x \in \mathrm{Dom}(X)} \frac{N^T(X=x) + N^C(X=x)}{N^T + N^C} KL\left(P^T(Y|X=x) : P^C(Y|X=x)\right)$$

note the weighting factors
N^T and N^C denote counts in the treatment and control datasets

The KL_{gain}

How much larger does the difference between class distributions in T and C groups become after a split on X?

$$KL_{gain}(X) = KL\left(P^T(Y) : P^C(Y)|X\right) - KL\left(P^T(Y) : P^C(Y)\right)$$

Properties of the KL_{gain}

Properties:

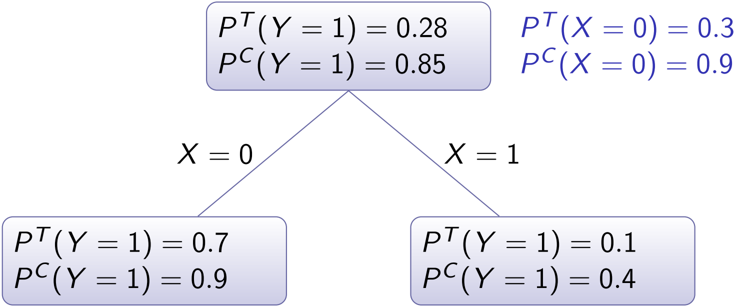
- If $Y \perp X$ then $KL_{gain}(X) = 0$
- If $P^T(Y|X) = P^C(Y|X)$ then

$$KL_{gain}(X) = \text{minimum}$$

- If the control group is empty, KL_{gain} reduces to entropy gain (Laplace correction is used on $P(Y)$)

Negative values of KL_{gain}

- Classification decision trees: $gain(X) \geq 0$
- $KL_{gain}(X)$ can be negative:



- Note the dependence of X on T/C group selection

Negative values of KL_{gain}

- Negative gain values are only possible when X depends on group selection
- This a variant of the Simpson's paradox

Theorem

If X is independent of the selection of the T and C groups then

$$KL_{gain}(X) \geq 0$$

- In practice we want X to be independent of the T/C group selection
- In medical research great care is taken to ensure this

The KL_{gain} ratio

- In standard decision trees, the gain is divided by test's entropy to punish tests with large number of outcomes
- In our case:
$$KL_{ratio}(X) = \frac{KL_{gain}(X)}{I(X)}$$
where
$$I(X) = H\left(\frac{N^T}{N}, \frac{N^C}{N}\right) KL(P^T(X) : P^C(X)) + \frac{N^T}{N} H(P^T(X)) + \frac{N^C}{N} H(P^C(X)) + \frac{1}{2}$$
- Tests with large numbers of outcomes are punished
- Tests for which $P^T(X)$ and $P^C(X)$ differ are punished
- This prevents splits correlated with the division into treatment and control groups

Uplift modeling through class variable transformation

Uplift modeling through class variable transformation

- Introduced in Jaśkowski, Jaroszewicz, 2012
- Allows for adapting an arbitrary classifier to uplift modeling

- Let $G \in \{T, C\}$ denote the group membership (treatment or control)
- Define an r.v.

$$Z = \begin{cases} 1 & \text{if } G = T \text{ and } Y = 1, \\ 1 & \text{if } G = C \text{ and } Y = 0, \\ 0 & \text{otherwise.} \end{cases}$$

- In plain English: flip the class in the control dataset

Uplift modeling through class variable transformation

- Now

$$
\begin{aligned}
&P(Z = 1 | X_1, \ldots, X_m) \\
&\quad = P^T(Y = 1 | X_1, \ldots, X_m) P(G = T | X_1, \ldots, X_m) \\
&\quad + P^C(Y = 0 | X_1, \ldots, X_m) P(G = C | X_1, \ldots, X_m)
\end{aligned}
$$

- Assume that G is independent of $X_1, \ldots, X_m$ (otherwise the study is badly constructed):

$$
\begin{aligned}
P(Z = 1 | X_1, \ldots, X_m) = P^T&(Y = 1 | X_1, \ldots, X_m) P(G = T) \\
&+ P^C(Y = 0 | X_1, \ldots, X_m) P(G = C)
\end{aligned}
$$

Uplift modeling through class variable transformation

- Assume $P(G = T) = P(G = C) = \frac{1}{2}$ (otherwise reweight one of the datasets):

$$
\begin{aligned}
&2P(Z = 1|X_1, \ldots, X_m) \\
&\quad = P^T(Y = 1|X_1, \ldots, X_m) + P^C(Y = 0|X_1, \ldots, X_m) \\
&\quad = P^T(Y = 1|X_1, \ldots, X_m) + 1 - P^C(Y = 1|X_1, \ldots, X_m)
\end{aligned}
$$

- Finally

$$
\begin{aligned}
P^T(Y = 1|X_1, \ldots, X_m) - P^C(Y = 1|X_1, \ldots, X_m)& \\
= 2P(Z = 1|X_1, \ldots, X_m) - 1&
\end{aligned}
$$

Uplift modeling through class variable transformation

Conclusion

Modeling $P(Z = 1|X)$ is equivalent to modeling the difference between class probabilities in the treatment and control groups

The algorithm:

1. Flip the class in $\mathbf{D}^C$
2. Concatenate $\mathbf{D} = \mathbf{D}^T \cup \mathbf{D}^C$
3. Build any classifier on $\mathbf{D}$
4. The classifier is actually an uplift model

Advantages

- Any classifier can be turned into an uplift model
- A single model is built
  - coefficients are easier to interpret than for the double model
  - the model predicts uplift directly
    (will not focus on predicting classes themselves)
  - a single model is built on a large dataset
    (double model method subtracts two models built on small datasets)
- Disadvantage: the double model may represent a more complex decision surface

Uplift Support Vector Machines

Uplift Support Vector Machines

- Introduced in Zaniewicz, Jaroszewicz, 2013
- Recall that the outcome of an action can be
  - positive
  - negative
  - neutral

Uplift Support Vector Machines

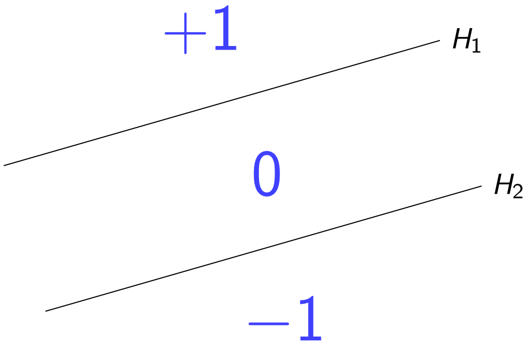
- Introduced in Zaniewicz, Jaroszewicz, 2013
- Recall that the outcome of an action can be
  - positive
  - negative
  - neutral

Main idea

Use two parallel hyperplanes dividing the sample space into three areas:

- positive (+1)
- neutral (0)
- negative (-1)

Uplift Support Vector Machines



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

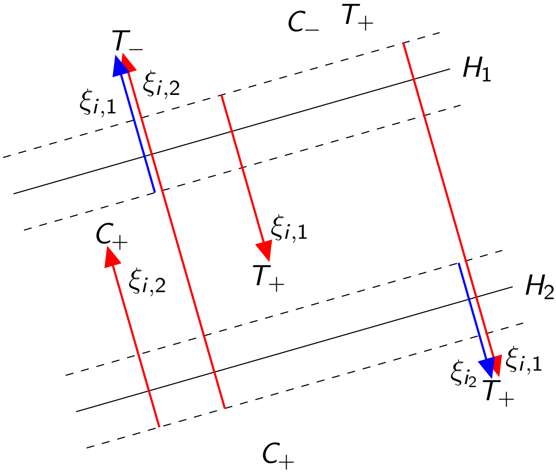
Uplift Support Vector Machines

- Fundamental problem of causal inference
  $\Rightarrow$ We never know if a point was classified correctly!
- Need to use as much information as possible
- Four types of points: T_+, T_-, C_+, C_-
- Positive area $(+1)$:
  - T_-, C_+ definitely misclassified
  - T_+, C_- may be correct and definitely not a loss
    (true outcome may only be neutral)
- Negative area (-1):
  - T_+, C_- definitely misclassified
  - T_-, C_+ may be correct and definitely not a loss
    (true outcome may only be neutral)
- Neutral area (0):
  - all predictions may be correct or incorrect

Uplift Support Vector Machines – problem formulation

- Penalize points for being on the wrong side of each hyperplane separately
- Points in the neutral area are penalized for crossing one hyperplane
  - this prevents all points from being classified as neutral
- Points which are definitely misclassified are penalized for crossing two hyperplanes
  - such points should be avoided, thus the higher penalty
- Other points are not penalized

Uplift Support Vector Machines – problem formulation



Optimization task – primal form

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

subject to:

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1 \leq -1 + \xi_{i,1}, \text{ for } (\mathbf{x}_i, y_i) \in \mathbf{D}_+^T \cup \mathbf{D}_-^C,$$
$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1 \geq +1 - \xi_{i,1}, \text{ for } (\mathbf{x}_i, y_i) \in \mathbf{D}_-^T \cup \mathbf{D}_+^C,$$
$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2 \leq -1 + \xi_{i,2}, \text{ for } (\mathbf{x}_i, y_i) \in \mathbf{D}_+^T \cup \mathbf{D}_-^C,$$
$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2 \geq +1 - \xi_{i,2}, \text{ for } (\mathbf{x}_i, y_i) \in \mathbf{D}_-^T \cup \mathbf{D}_+^C,$$
$$\xi_{i,j} \geq 0, \text{ dla } i = 1, \ldots, n, \; j \in \{1, 2\},$$

Optimization task – primal form

We have two penalty parameters:

C_1 penalty coefficient for being on the wrong side of one hyperplane

C_2 coefficient of additional penalty for crossing also the second hyperplane

- All points classified as neutral are penalized with $C_1\xi$
- All definitely misclassified points are penalized with $C_1\xi$ and $C_2\xi$

How do C_1 and C_2 influence the model?

Influence of penalty coefficients C_1 and C_2 on the model

Lemma

For a well defined model $C_2 \geq C_1$. Otherwise the order of the hyperplanes would be reversed.

Lemma

If $C_2 = C_1$ then no points are classified as neutral.

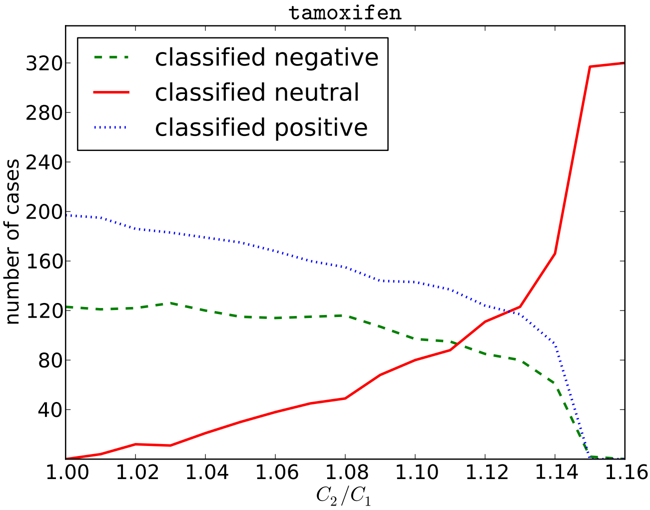
Lemma

For sufficiently large ratio C_2/C_1 no point is penalized for crossing both hyperplanes. (Almost all points are classified as neutral.)

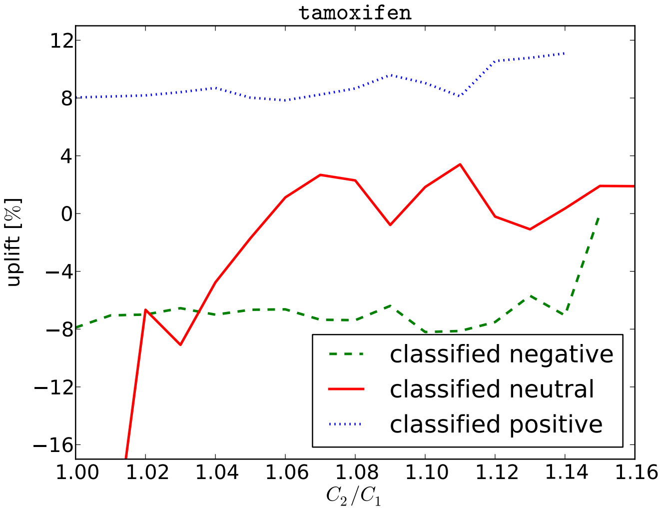
Influence of penalty coefficients C_1 and C_2 on the model

- The C_1 coefficient plays the role of the penalty in classical SVMs
- The ratio C_2/C_1 decides on the proportion of cases classified as neutral

Example: the tamoxifen drug trial data



Example: the tamoxifen drug trial data



Literature

- Nicholas J. Radcliffe and Patrick D. Surry. *Real-World Uplift Modelling with Significance-Based Uplift Trees*, White paper, Stochastic solution ltd, 2011 (PDF) [a good overview paper]
- P. Rzepakowski, S. Jaroszewicz. *Decision Trees for Uplift Modeling*, ICDM'2010 (PDF)
- M. Jaśkowski, S. Jaroszewicz. *Uplift Modeling for Clinical Trial Data*, In ICML 2012 Workshop on Machine Learning for Clinical Data Analysis (PDF)
- Ł. Zaniewicz, S. Jaroszewicz. *Support Vector Machines for Uplift Modeling*, IEEE ICDM Workshop on Causal Discovery (CD 2013) at ICDM 2013