Support Vector Machines for Uplift Modeling

Łukasz Zaniewicz\textsuperscript{2}  Szymon Jaroszewicz\textsuperscript{1,2}

\textsuperscript{1}National Institute of Telecommunications
Warsaw, Poland

\textsuperscript{2}Institute of Computer Science
Polish Academy of Sciences
Warsaw, Poland
What is uplift modeling?

From workshop’s description:

Traditionally, causal relationships are identified based on controlled experiments. [...] there has been an increasing interest in discovering causal relationships from observational data only.
What is uplift modeling?

From workshop’s description:

Traditionally, causal relationships are identified based on controlled experiments. [...] there has been an increasing interest in discovering causal relationships from observational data only.

- Suppose we do have data from a controlled experiment
- Question: what can Machine Learning do for us?
- Relatively little interest in the ML community
What is uplift modeling?

Given two training datasets:

1. the treatment dataset
   individuals on which an action was taken
2. the control dataset
   individuals on which no action was taken
   used as background

Build a model which predicts the causal influence of the action on a given individual
Uplift modeling

Notation:
- $P^T$ probabilities in the treatment group
- $P^C$ probabilities in the control group

Traditional classifiers predict the conditional probability

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

Uplift models predict change in behaviour resulting from the action

$$P^T(Y \mid X_1, \ldots, X_m) - P^C(Y \mid X_1, \ldots, X_m)$$
Why uplift modeling?

A typical marketing campaign

Sample → Pilot campaign → Model $P(\text{buy}|\mathbf{X})$ → 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.
Why 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 \textbf{because} of the campaign
- Not people who bought \textbf{after} the 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)*
Marketing campaign (uplift modeling approach)

Treatment sample \rightarrow \text{Pilot campaign}

Model
\[ P^T(\text{buy}|X) - P^C(\text{buy}|X) \]

Select targets for campaign

Control sample
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

- Rubin’s causal inference framework

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
The two model approach

An obvious approach to uplift modeling:

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

$$M^U(Y|X) = M^T(Y|X) - M^C(Y|X)$$
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
Support Vector Machines (SVMs) are a popular Machine Learning algorithm.

Here we adapt them to the uplift modeling problem.
Recall that the outcome of an action can be:

- positive
- negative
- neutral
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

\[ H_1 : \langle w, x \rangle + b_1 = 0 \]
\[ H_2 : \langle w, x \rangle + b_2 = 0 \]
How do we train Uplift SVMs?

Classical SVMs:

need to know if a case is classified correctly

Fundamental problem of causal inference

⇒ We never know if a point was classified correctly!

The algorithm must use only the information available
Four types of points: $T_+, T_-, C_+, C_-$

Positive area (+1):
- $T_-, C_+$ definitely misclassified
- $T_+, C_-$ may be correct, at worst neutral

Negative area (-1):
- $T_+, C_-$ definitely misclassified
- $T_-, C_+$ may be correct, at worst neutral

Neutral area (0):
- all predictions may be correct or incorrect
Penalize points separately for being on the wrong side of each hyperplane

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

\[ H_1^{+} + \xi_{i,1} + C_+^{+} \xi_{i,2} + C_-^{+} \xi_{i,1} + \xi_{i,2} + C_-^{+} \xi_{i,1} - C_-^{+} \xi_{i,2} = 0 \]

\[ H_2^{+} T_+ \]

\[ H_1^{-} T_- \]

Łukasz Zaniewicz, Szymon Jaroszewicz | Uplift SVMs
Optimization task – primal form

\[
\begin{align*}
\min_{\mathbf{w}, b_1, b_2 \in \mathbb{R}^{m+2}} & \quad \frac{1}{2} \langle \mathbf{w}, \mathbf{w} \rangle + C_1 \sum_{\mathbf{D}_T \cup \mathbf{D}_-^C} \xi_{i,1} + C_2 \sum_{\mathbf{D}_T \cup \mathbf{D}_-^C} \xi_{i,1} \\
& \quad + C_2 \sum_{\mathbf{D}_T \cup \mathbf{D}_-^C} \xi_{i,2} + C_1 \sum_{\mathbf{D}_T \cup \mathbf{D}_-^C} \xi_{i,2},
\end{align*}
\]

subject to:

\[
\begin{align*}
\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1 & \leq -1 + \xi_{i,1}, \text{ for } (\mathbf{x}_i, y_i) \in \mathbf{D}_T^+ \cup \mathbf{D}_-^C, \\
\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1 & \geq +1 - \xi_{i,1}, \text{ for } (\mathbf{x}_i, y_i) \in \mathbf{D}_T^- \cup \mathbf{D}_+^C, \\
\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2 & \leq -1 + \xi_{i,2}, \text{ for } (\mathbf{x}_i, y_i) \in \mathbf{D}_T^+ \cup \mathbf{D}_-^C, \\
\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2 & \geq +1 - \xi_{i,2}, \text{ for } (\mathbf{x}_i, y_i) \in \mathbf{D}_T^- \cup \mathbf{D}_+^C, \\
\xi_{i,j} & \geq 0, \text{ dla } i = 1, \ldots, n, j \in \{1, 2\},
\end{align*}
\]
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.)
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

![Graph showing tamoxifen data]

- **Classified Negative**
- **Classified Neutral**
- **Classified Positive**

 Łukasz Zaniewicz, Szymon Jaroszewicz | Uplift SVMs
Example: the tamoxifen drug trial data

Uplift SVMs

 Łukasz Zaniewicz, Szymon Jaroszewicz

Uplift SVMs
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.
For example:

\text{Gain for the 10\% highest scoring customers} =
\% \text{ of successes for top 10\% treated customers} -
\% \text{ of successes for top 10\% control customers}
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 UCI breast cancer data (artificially split into T/C groups)

Łukasz Zaniewicz, Szymon Jaroszewicz

Uplift SVMs
Experimental evaluation

- Used 5 datasets with real control groups
- Used additional 13 dataset artificially split into T/C groups
- Uplift SVMs compared favorably with other models
  - better than double SVM model on 13 out of 18 datasets
  - better than uplift decision trees on 12 out of 18 datasets
Thank you!