Design document for uplift system

August 28, 2024

Contents

Essentials of uplift modeling system design

1.1 Business Objectives

Today, X6 Retail Group has a need to optimize costs for discounts and related advertising campaigns, such as SMS alerts, push notifications and email campaigns. The company's management set a goal as reducing promotional costs to 6% (advertising campaigns) and increasing profits by 10% (impact on the target audience).

1.2 Existing flow analysis

The existing approach (without the use of machine learning) - Customer Segmentation with RFM Analysis were not effective enough, achieving only 3% savings (advertising campaigns) and a 5% increase in profit (impact on the target audience).

1.3 Advantages of using ML

The implementation of an ML solution (in theory) should help us purposefully influence a loyal audience, which will help us reduce the costs of promotions (minimizing the impact on groups: loyal (sure things), indifferent (do not disturbs), negative (lost causes) and influence only the target audience, namely the persuadables.

1.4 Business requirements and restrictions

1.4.1 Accuracy

- The system must provide a high level of accuracy when making predictions or analyzing data, with a minimum acceptable error defined by the business (for example, a prediction accuracy of at least 95%).
- Accuracy must be consistent regardless of data volume or changes along the time.

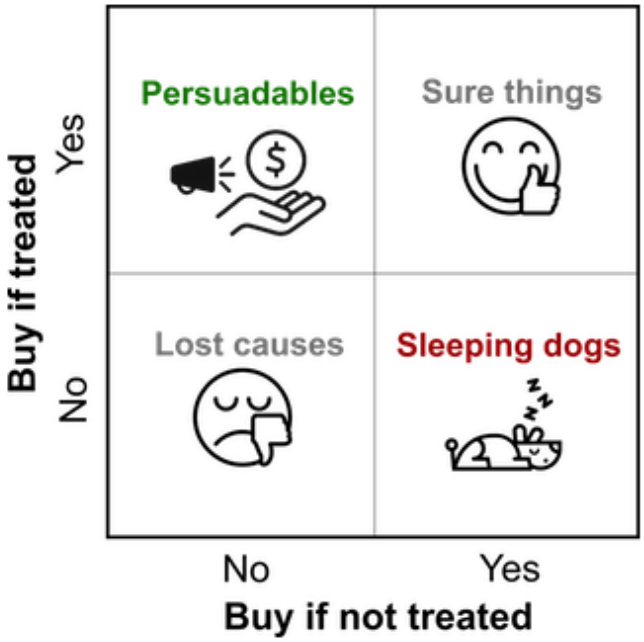

Figure 1.1: types of customers

1.4.2 Scalability

- The system must be able to efficiently process increasing volumes of data without reducing performance (with a successful MVP, the system will scale to all regions)
- Easily implement additional compute resources to support data and workload growth.
- The system must support horizontal and vertical scaling.

1.4.3 Interpretability

- The results of the system must be understandable and explainable to analysts.
- The system must provide an explanation for its decisions or predictions (for example, an explanation of the model in terms of feature significance).
- Interpretability should not reduce the accuracy of the system.

1.4.4 Processing speed

- Our system does not require real-time data processing, which allows us to optimize resources and not focus on instantaneous inference when making predictions.

1.5 Business restrictions

1.5.1 Deadlines

- The first version of the service should be launched within three months.

1.5.2 MVP integration steps

- Conducting a promotional campaign: Implementation of promotions for all segments to identify a group of persuaders.
- Data analysis: Collection and analysis of data after the completion of the promotion.
- Model Tuning: Train the Uplift model based on the collected data.
- Validation of predictions: Let's check the trained model on historical data for 1 year in a test region.
- Implementation: Integration of the model into workflows and launch of the test phase.

1.5.3 MVP Success criteria

- The MVP will be successful if promotion costs are reduced by 6% with better targeting persuadable customers.
- An additional criterion for success is a 10% increase in profits due to increased efficiency of promotions and the elimination of excess costs for ineffective audience segments.

1.6 Project scope

- Training the model on "convenient goods" (not perishable). Requires clarification from analysts.
- Validation of forecasts using historical data (promotion campaigns used for train dataset must be earlier than campaigns used for validational dataset to eliminate a possible lack of data)
- Introduction of the Uplift service to a limited number of stores in one region.

1.7 Project exclusions

- Scaling to all regions and countries.
- Scaling for all products.

Preliminary research

2.1 Analysis of precious system

- Currently, the RFM system is used for advertising targeting. RFM is a method where each client has three characteristics: recency, frequency, monetary. These are integer characteristics varying from 1 to 5, obtained with BI-system currently used in the company, based on transactional data. All clients from the database are clustered (k-means++) according to these characteristics. We, with the help of a expert-team, assign an action in a future promotion for each cluster. When visualizing this clustering, two clearly defined clusters are visible. We offer a i% discount on category A products in one cluster and a j% discount on category B in other, and the latest campaign showed that this rule, specifically in this subgroup of customers (two clusters) has already saved the company more than 6% of communication costs (compared to advertising without RFM), so it's worth considering a hybrid solution and comparing it with our monolithic ML solution.

2.2 Restrictions

- Data privacy
- Communication budget = n
- Budget for discount = m
- Audience volume = k

2.3 Build or buy?

Arguments for buying:

- X6 Retail Group does not need the latest innovations and scientific breakthroughs; for them, innovation is not a goal or advertising, but a way to make money.

- For retail, the system of creating discounts/promotions and notifying consumers about them does not play a critical role. (these are not warehouses, not logistics, not retail outlets)
- Only one retailer on the market has a permanent ML-team. This retailer leads in turnover by a large margin from all others. X6 Retail Group does not have such budgets and they do not intend to directly compete with such a leader.

Arguments for building:

- We don't have any direct legacy-code, but as mentioned above, the previous solution shows very good results on a subset of clients, so ideally we need a hybrid solution.

Thus, we can conclude that the best choice in this situation would be using the open-source method. Yes, it still requires the maintenance of some ML-team, but it will clearly be cheaper than a team for developing whole solution. Additionally, this path will allow us to implement a hybrid solution.

Analysis of existing open-source solutions:
scikit-uplift

- Many out-of-box metrics, including qini coefficient and AUUC
- Compatible with Xgboost, LightGBM, Catboost
- Metrics are compatible with scikit.model_selection
- Good, built-in tools for visualizing data analysis
- There is extensive documentation

UTBoost

- Among metrics from sci-kit uplift, only the qini coefficient is built in
- There is no documentation, but the library was published along with an article where the proposed new methods from this library were extensively tested and compared with basic solutions in uplift modeling.
- All methods are based only on gradient boosting of trees, but, according to the authors of the article, these methods beat almost all basic methods in accuracy.
- Most of the library is written in C++, and it works much faster than its Python counterparts. Yes, and at the same time it is a Python package.

From the point of view of innovation, the second library looks more interesting, but as mentioned earlier, we do not need it. Plus, as mentioned in the first chapter, we don't really need speed and we have a relatively short deadline, development on the old-but-gold scikit will be much faster, plus these models will be much easier to interpret, unlike a custom Chinese solution. Using scikit-uplift is a priority.

Goals and anti-goals

Goals:

- Gradual research of the client base for a better understanding of behavioral patterns among custmers (when we launch the uplift system, we will carry out a promotion not only for clients selected by our model, but also for a random sample from the general database; this will allow us to further train more advanced models and, in principle, better understand behavioral patterns)
- Reduce customer acquisition cost (CAC)
- The return on investment (ROI) for the marketing campaign should exceed 200%, in other words, the costs should be half the profit received from the campaign.

Anti-goals:

- ROI is less than 100%, in other words, more was spent on the marketing campaign than earned.
- Customer satisfaction with a product or company is lower than before the marketing campaign.

Loss functions and metrics

Definitions of terms used:

- Control group (index C for values according to this people) - people who did not receive communication (SMS alerts, push notifications and email newsletters)
- Target group (index T for values according to this people) - those people who received the communication
- Y - 1 if client responded and 0 if he did not.
- response rate - share of customers who made a purchase

4.1 Metrics

4.1.1 Uplift by percentile

Counting algorithm:

- Sort by predicted uplift value.
- We divide the sorted data into percentiles.
- In each percentile, we separately evaluate uplift as the difference between the average response rates in the test and control groups.

Metric Description:

- As a result, we will see how the average response rate has changed in each percentile.
- For a good model, the highest uplift value will be achieved in the first percentiles and decrease in further ones, but will not fall below zero. This will be an indicator that the model correctly estimates the effect of communication.

Why not?:

- There is no need to monitor the metric separately for each percentile at this moment. Our goal is to improve the quality in general; based on this, it will be more convenient to focus on the overall quality based on test data.

4.1.2 Weighted average uplift

Counting algorithm:

- Calculate uplift by percentile.
- Calculate the weighted average by percentiles.

Metric Description:

- This is a weighted percentile average calculated in uplift by percentile, the weight depends on the size of the target group.
- Takes values from [-1, 1] the metric is equal to 1 in a situation where customers did not buy anything before communication and bought it after, -1 is the opposite situation.

Why not?:

- It's a bit difficult to interpret this value.

4.1.3 Cumulative gain curve (uplift curve)

Counting algorithm:

- Sort clients by predicted uplift.
- Calculate following value for each p top clients:

$$CG(p) = \left(\frac{Y_p^T}{N_p^T} - \frac{Y_p^C}{N_p^C} \right) \cdot \left(N_p^T + N_p^C \right)$$

There are few metrics based on this curve:

Area under uplift curve (AUUC)

Why not?:

- Like Weighted average uplift it isn't interpretable metric.

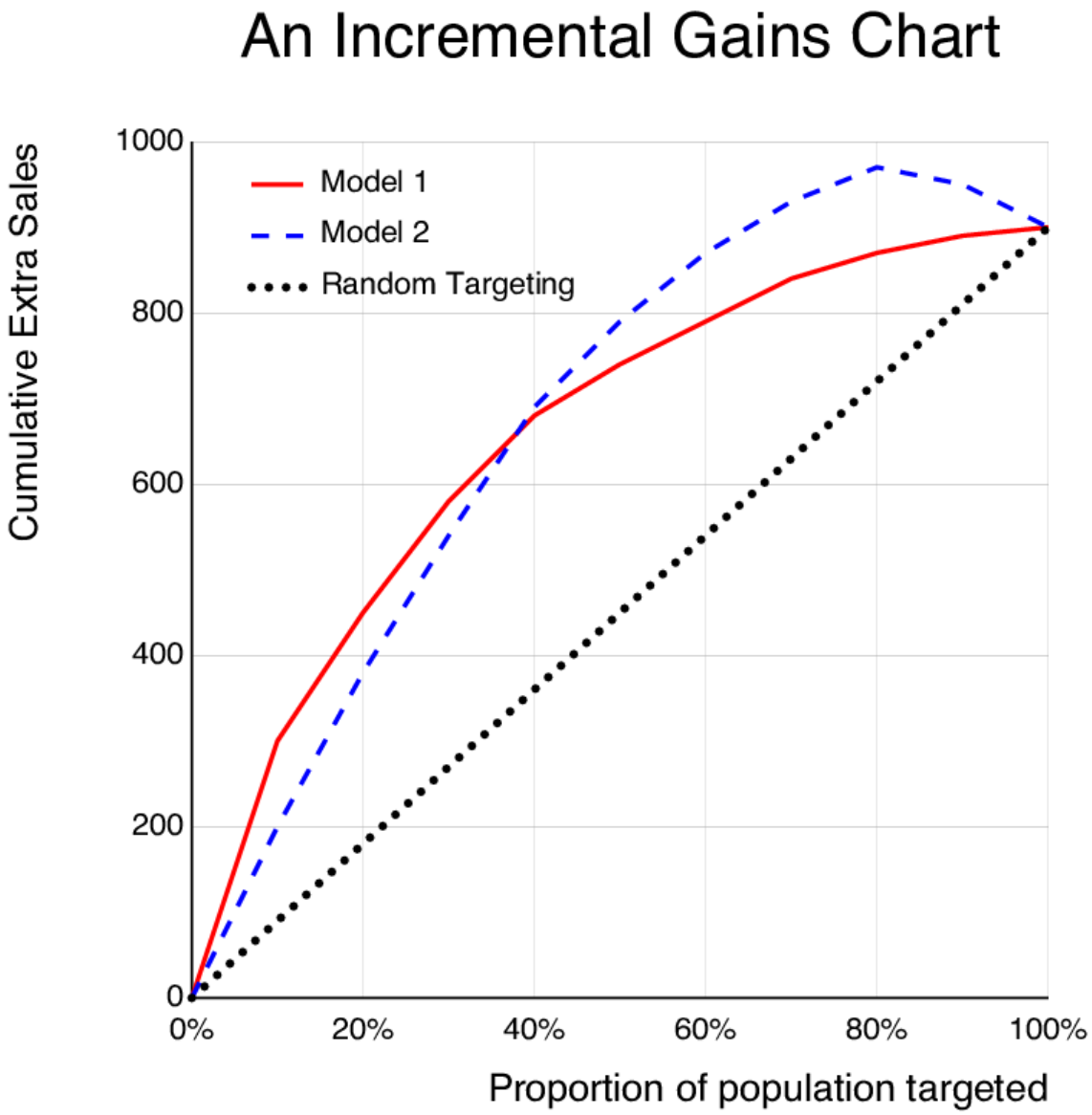

Figure 4.1: Example of CGC.

Uplift@k

It equals $CG(k)$.

Metric Description:

- Takes values from [-1, 1]. When there are no purchases in the target group, and in the control group all customers made a purchase, it equals -1, and in the opposite situation: in the target group all customers made a purchase, while in the control group no one bought anything, it equals 1.

- This metric can be calculated in two different ways: first, sort by predicted uplift and then calculate the difference in response rates of the two groups. Or, conversely, initially sort objects from the control and target groups separately.

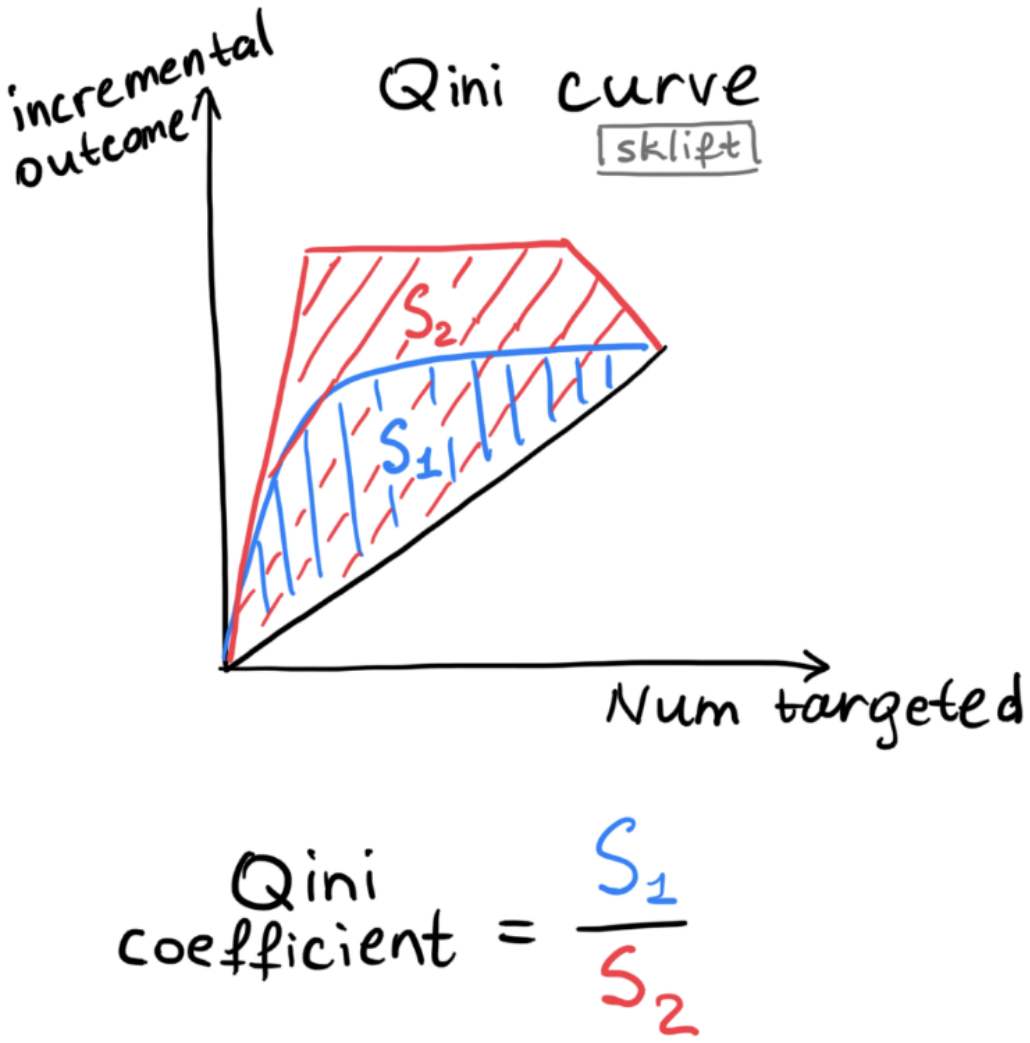

Figure 4.2: Scheme of AUUC and qini cof. calculation.

Why not?:

- We have no obvious restrictions on the number of clients with whom we want to communicate; there is no point in limiting ourselves to just the top k clients.

Uplift_max

It equals $\max_p CG(p)$

It's perfect metric for our problem. Unlike Uplift@k it hasn't any restrictions on the number of clients with whom we want to communicate and unlike AUUC it's quite interpretable.

4.1.4 Qini coefficient

Counting algorithm:

- Sort clients by predicted uplift.
- Calculate following value for each t top clients:

$$\text{qini curve}(t) = Y_t^T - \frac{Y_t^C N_t^T}{N_t^C}$$

Area under qini coeficient (AUQC)

Calculation and disadvantages are the same as for AUUC.

But advantage of this metric is a physical interpretation. The physical meaning of the qini curve is to prevent the model from raising only the target (treatment) group to the top in the ranking, penalizing it for this with the $\frac{N_t}{N_c}$ multiplier, which reduces the final value if N_t is much greater than N_c.

In conclusion, uplift_max - is a perfect metric for us, but additionally we should try AUQC.

4.2 Loss-functions

Choosing loss already depends on chosen model for our system, so further we will talk a bit about on which uplift-model we should stop in our case.

Simple approaches such as one model or T-learner aren't our case because it doesn't predict uplift directly and chosen metrics (upliftmax and AUQC) are based on this value.

Reinforcement learning is also not for us because it seems quite complicated and innovative, and as we mentioned in previous chapters it isn't our goal.

So, for us the most suitable methods are transformation of targets and tree-based. Let's discuss about each in detail.

4.2.1 Transformation of targets

We transform our values into new targets:

$$Y^* = Y\frac{T}{P(T=1 \mid X)} - Y\frac{1-T}{P(T=0 \mid X)}$$

An advantage of this value is very probable convergence to uplift value. Next, we just train any generic ML-model to predict it. The loss may be st simple like MSE or MAE, depends on distribution of a new target (For example: if there are a lot of outliers in this distribution we'll take MAE)

4.2.2 Decision tree

We split each node into leaves to maximize a difference between uplifts among leaves. Estimator for uplift in node:

$$\widehat{\tau}_{\text{node}} = \frac{\sum_{i\in \text{ node}} Y_i T_i}{\sum_{i\in \text{ node}} T_i} - \frac{\sum_{i\in \text{ node}} Y_i \left(1 - T_i\right)}{\sum_{i\in \text{ node}} \left(1 - T_i\right)}$$

In this case optimization criteria will be just a maximization of difference: $|\widehat{\tau}_{\text{left}} - \widehat{\tau}_{\text{right}}|$.

There are more complicated optimization criteria based on distances between distributions (distances not in direct mathematical sense!) such as Kullback-Leibler divergence and Energy distance.

Tree-based method trains very slowly. We may test this method in our system but give preference to transformation of targets.

Gathering datasets

5.1 Data sources

5.1.1 Internal data sources

- Calendar of past and future promotions. Information about promotions, the point at which the promotion was held, the dates on which the promotions were held and the communication channels. More details in table 1

- Information about the purchase history of a particular user. Data about the customer, product, purchase date, and amount. More details in table 2.

Possible data: If we have an app. It is easier for us to display dependencies in customer purchases. Since information about clicks, category views and add to cart gives implicit information about the customer.

5.1.2 External data sources

- Weather data. Buy information from the weather service. Weather information is an important factor in going out to get groceries at the store. See Table 3 for more details

- Data on promotions in competitors' stores. Information about prices and relevance of the promo action. More details in table 4.

5.2 Meta-data

- Store information. Store Id, geographical location, code of product category presented in stores. More details in table 5

- Product. Product code. product category code. More details in table 6

- Transactions. Customer identifier, point of sale identifier, product category identifier, product identifier. More details in table 6.

5.3 Data labeling

In order to obtain a training sample for uplift modeling, an experiment must be conducted:

- Randomly split a representative portion of the customer base into target and control groups;
- Launch a test marketing campaign on a target group.

Before conducting the main promotional campaign, it is recommended to randomly select a small portion of the customer base and divide it into control and target groups. With the help of this data it will be possible not only to adequately assess the effectiveness of the campaign, but also to collect additional data for further refinement of the model.

Table 5.1: Calendar of promotional events held

Field	Description
Stock ID	Unique share identifier
Holding point	Unique share identifier
Start date	Start date of the action
End date	Campaign end date
Communication channel	Channel through which the action was conducted

Table 5.2: Purchase history of specific users

Field	Description
Purchase ID	Unique purchase identifier
Client ID	Unique client identifier
Product code	The code of the item that was purchased
Date of purchase	Date of purchase
Amount	Purchase amount

Table 5.3: Weather data

Field	Description
Date	Date of weather data
Location	Geographical location
Temperature	Temperature at the specified time
Precipitation	Information on the presence of precipitation
Humidity	Humidity level

Table 5.4: Data on promotions in competitors' stores

Field	Description
Date	Date of the action
Stock ID	Unique share identifier
Competitor ID	Competitor's unique identifier
Location	Geographical location
Item ID	Unique product identifier
ID product category	Unique product category identifier
Price during the promotion period	Price of goods during the promotion period
Price on regular days	Price of goods on normal days

Table 5.5: Information on stores

Field	Description
Store ID	Promotion Date
Location	Geographic location
Product Category ID	Unique product category ID

Table 5.6: Product Information

Field	Description
Product ID	Unique product ID
Product Category ID	Unique product category ID
Store ID	Unique store ID
Customer ID	Unique identifier of the client

Validation schemes

At the initial stage, when we have a limited amount of data from previous promotional campaign, we may encounter the problem of insufficient representativeness and overfitting. To minimize these risks, we will use stratified cross-validation on data from one launch of a promotional campaign. However, it is important to understand that this is not a perfect approach and results may be limited.

6.1 Standard Schema

We use stratified cross-validation by groups with parameters:

- K = 5.
- Groups: control and target.
- Data splitting: train - 60%, validation - 30%, test - 10%.
- Metric: uplift_max

Each time the model is trained, the random_state parameter is changed to provide variety.

Disadvantages of this approach:

- Possible overfitting on a limited data set.
- Validation and test samples may not be relevant for real cases. This is due to the fact that the metrics are calculated on the data from the same promotional campaign as the model was trained, while in production there will be completely new data.

The advantages of this approach:

- We apply for training the whole dataset.

6.2 Non-trivial Schema

With the increase of the number of launches of promotional campaigns, we plan to introduce more complex validation schemes. This will improve the representativeness and reliability of the model assessment:

- Training on historical data: we train on data from the last 4 campaign launches, excepting one last. It allows us to take into account temporary changes and seasonality.
- Validation on data from the last run: we use data from the last run to evaluate the model to check its relevance and adaptability to new conditions.

This approach will help create more relevant and reliable model evaluation under real-world conditions.

Baseline solution

7.1 Basic model

We will use linear regression to predict the likelihood of a response to a promotion. it helps assess baseline performance and set an initial uplift metric.

7.2 Base metric

As was mentioned in chapter devoted to metrics, base metrics are uplift_max and AUQC.