ИКОНОМИЧЕСКИ ИЗСЛЕДВАНИЯ
ECONOMIC STUDIES
Volume 32(2), 2023

Todor Krastevich[1]

RETARGETING CUSTOMERS USING UPLIFT MODELING[2]

"Traditional" digital marketing campaigns are based primarily on a priori geotargeting, augmented with profiling of potential consumers based on language, sociodemographics, interests and preferences. A step ahead is when experimental results from A/B testing are used for more precise retargeting, in order to prove in a statistically significant way the direction and size of the effect of a potential communication marketing impact. Through the application of uplift modelling, it is possible to complement the experimental data from A/B testing by identifying the effect of specific marketing treatments (e.g. a specific message, alternative display ad design, web page layout and/or change in price offer) on specific individuals as opposed to an overall increase or decrease in conversion rates caused by the impact. This technique can help evaluate and predict their responses through supervised machine-learning classification algorithms. This nuanced analysis allows for personalized targeting of marketing communication to only leads who are likely to respond positively to an impact. This paper proposes and demonstrates a prototype model for optimal retargeting of customers based on machine learning algorithms and open-source programming.

Keywords: Uplift models; predictive modeling; retargeting; supervised machine learning

JEL: M37; C35; C55; C63

1. Introduction

Any personalized marketing communication campaign can be viewed as an analytical task whose solution supports specific decision-making. Most often, the campaign aims to exert an influence on the customer, prompting him to behave in a way that benefits the advertiser – for example, to unlock some behavioural change with beneficial consequences for the business, the environment and/or society. Achieving these objectives is done by engaging the customer (e.g. by serving information about the benefits of the product offered, offering them

[1] *Todor Krastevich, PhD, Associate Professor in Marketing Department at D. A. Tsenov Academy of Economics, e-mail: krst@uni-svishtov.bg.*

[2] This research is funded by the National Science Fund of Bulgaria (BNSF) under the project No. КП-06-Н35/7 from 12/2019.

Preliminary short version of the article was presented and discussed as a report of the 5th Jubilee International Scientific Conference `Remarketing the Reality`, 17.06.2022, organized by Department of Marketing, University of Economics – Varna, Bulgaria.

This paper should be cited as: *Krastevich, T. (2023). Retargeting Customers Using Uplift Modeling. – Economic Studies (Ikonomicheski Izsledvania), 32(2), pp. 78-99.*

special price discounts, some kind of giveaway or involving them in a cause) through direct contact in selected communication channels (e.g. phone calls, personalized emails and text messages, personalized display advertising, etc.). Such campaigns can be seen in virtually all industries, with the most common occurring in high-tech sectors of the economy, such as telecommunications, the financial sector, and commerce. However, it is possible to plan and run campaigns related to sustainable consumption (e.g. self-control and self-restraint in the use of non-renewable natural resources or products whose consumption leads to negative social consequences). Such campaigns do not aim at profit maximization, but are likely to be planned and optimized with the same analytical tools and predictive models typical of digital marketing in a business context.

In what follows, we first consider the "classical" way of applying predictive models using machine learning algorithms, revealing some of its limitations. We then attempt to justify the possibility of improving results by identifying the truly "optimal" target group of customers using incremental models and machine learning algorithms. Although the purpose of this paper is rather methodological, we would also like to focus the reader's attention on the positive financial impact of incremental modelling and the possibility of increasing the ROI of digital marketing campaigns through customer selections based on net scoring.

2. On the Limitations of "Traditional" Predictive Modelling

"Traditional" digital marketing campaigns[3] typically target potential consumers based on their attributes and recorded responses to previous campaigns. Those whose profiles most closely match those who have responded positively in the past are selected. In practice, the causal relationship between stimulus and response is not examined, but profile similarity is sought. However, in order to selectively and personalize the influence on consumer behaviour, it is interesting to evaluate precisely the mechanism of formation of the expected causal effect of the communication impact. Predictive modelling of the net effect of user-level influence could serve as a basis for machine learning algorithms for classifying, selecting, and remarketing to the right people at the right time in the right place on the Web.

The causal links between communication marketing influences and consumer responses are clearly more difficult to empirically assess and model than simply predicting future customer behaviour. Classical predictive models focus on predicting events from an individual consumer's future behaviour. Most analytical approaches and tools still focus on making predictions instead of profitable targeting. Consider the fact that prediction tools based on

[3] We use the term "traditional" marketing campaigns to refer to those that, in response modeling, typically take a group of treated customers and attempt to build a predictive model that separates likely responders from non-responders using some predictive modeling technique such as decision trees, random forest, or logistic regression. In this approach, only the responses of the treated customers are considered for building the model. The traditional approach to response modeling is training a predictive model on only the treatment group (those who received the promotion). This model will separate those who are likely to respond (purchase the product) from those who are less likely to respond (did not purchase the product). In contrast, incremental modeling uses both treatment and control customer groups to build a predictive model that focuses on incremental response.

machine learning algorithms only predict momentum (i.e., the customer behaviour that is observed with the usual influencer marketing tools, which would mean keeping the usual communication mix). Such an approach relies more on the identification of correlation or hypothetical dependency, but not causality. For example, it is possible to use some form of regression to predict consumption or customer revenue, considering the influence of a range of uncontrollable factors (such as seasonality, trend, some discrete events, etc.). The objective of such an approach is to produce as accurate a forecast of the expected outcome as possible, subject to the maintenance of previously observed behaviour and/or circumstances. Such an approach to building predictive models can only inform what would happen if customers whose expected behaviour would lead to a positive effect on observed economic indicators (e.g., profit or sales revenue for the company) were not targeted. It is not particularly informative and useful for deciding whether or not a personalized impact should be exerted on a particular customer and whether or not the desired behaviour would be observed without an exerted impact. Using it in a practical context can lead to targeting the wrong people, resulting in both wasted resources and possibly negative reactions. However, what would happen if we shifted the focus from predicting expected behaviour in the usual marketing mix to assessing the difference in an individual consumer's behaviour in the future if personalized influence is or is not exerted on them? To justify a logically correct answer to this question, we advocate and share the following theses (Vittal, 2006, p. 2):

- Direct marketing impacts different customers with different effectiveness. For some of them, it can even trigger negative reactions. These are most evident in some customer retention activities falling within fixed-term contractual relationships (e.g. with telecom operators running customer retention campaigns), but can also be 'hidden' in marketing campaigns that are generally successful[4]. They both reduce revenues and increase costs for the proposing company and therefore deserve special attention.
- There is no doubt that some marketing actions have a negative impact on some customers. This is most clearly illustrated by the surprisingly common view of mobile telecommunications operators who run customer retention campaigns that apparently increase the total number of customers. These are actually quite easy to understand, as we will explain below. Less clear is how campaigns with a positive overall result often contain significant segments within which there is a negative effect. Where this not only increases campaign cost effectiveness, but actually results in a lower campaign gross result that could be achieved by targeting a lower volume.

[4] It is not difficult to understand why negative effects often occur in customer retention activities. The traditional approach of modelling customer churn first identifies those at high risk of churn. Generally, these are people who are already dissatisfied in some way. Intervention, especially over the phone, is associated with a high risk of the customer responding with a request for immediate cancellation – i.e., provoking a preemptive decision to drop out that might not otherwise have happened. due to customer apathy. Also, intrusive mechanisms for unsolicited commercial messages and communications, such as phone calls or pop-up advertisements, are often received with negativity. Niche-oriented communications designed to appeal to a specialized group can backfire on people who do not belong to that group or do not fit the stereotype of that group.

- Customers who buy or consume the most when they are the target of targeted communication are not necessarily the best targets for marketing interventions. Some of them would consume or spend in the same way, even if they were not targeted.

- Since desirable customer responses for the company may or may not be due to the direct marketing interventions (some of them will surely be observed even without a specific targeted communication impact), in order to empirically measure the net effect of the marketing impact, it is necessary to design, organize and conduct an experiment with a control group.

Can the theses put forward be empirically justified and pragmatically defended? Typically, products are promoted by communicating with the customer through various channels: SMS, push, chatbot messages on social networks and many others. The formation of segments for personalized targeting is usually solved using three types of predictive models based on machine learning: models predicting similarities based on past behaviour (Propensity modeling), models predicting certain reactions (response modelling), and models predicting the net (a.k.a. incremental) effects (uplift modelling).

Propensity models (also sometimes referred to as `lookalike` models) estimate the probability that a customer will perform some desired action (Radcliffe & Surry, 2011, p. 2). To train such models, sample observational users data containing past unprompted desired (or undesired) actions (or non-actions) – e.g., registered sale, subscription, installation of a trial version of an application, or other actions as a result of an inaudible or direct request (resp. refusal) by the user to the content creator. The goal of the model is to identify users similar to those who have taken the target action desired by the proposer (i.e., inclined to purchase). The marketing problems addressed by such models are related to the generation of a list of consumers with a high probability of performing the expected action (e.g., purchase), provided they are not subjected to a targeted marketing influence.

Response models aim to estimate the probability that a customer will perform some action desired by the proposer, given some targeted communication (Javaheri, Sepehri, & Teimourpour, 2014, p. 154). A training sample of data collected after some interaction with the customer is used to train such models. In contrast to the first approach, here we have the results of provoked reactions (e.g. the customer was offered a higher subscription plan and accepted it or refused it). Typical problems that can be solved with such models are related to the generation of a list of customers with a high probability of response (e.g. conversion to purchase) if they are subjected to targeted personalized marketing treatment.

In the uplift models, the goal is to estimate the probability that a customer will perform some desired action only if exposed to a targeted communication exposition. The idea is to use a model to assess the difference in customer behaviour when a personalized targeted communicational influence is exerted versus in the absence of such an influence. Formally, this means finding the difference between the probability of purchase with communication and the probability of purchase without communication (also called `net` of `incremental` effects of personalized targeted communication).

Technically, incremental or uplift modelling should produce identical results to models predicting response if the response rate in the control group tends to zero. This is a hypothetical situation in which consumers would not change their behaviour if they were not targeted by communication. Such scenarios are possible, for example, for products or services that are in the late maturity phase of their lifecycle, as well as market-new products that are completely unfamiliar to the consumer and which he would not risk buying without significant stimulation. In addition, products and services that are primarily sold by invitation (e.g. accepting a webinar invitation via email) may have very low response rates without targeted communication and incremental modelling would not bring any significant advantages over traditional modelling.

Optimizing a digital communications campaign using the first two types of models can lead to missed opportunities to increase the effectiveness of marketing spend. Why do we think this statement is reasonable? When the data used to develop a predictive model is in some way influenced or subject to change due to the interaction between the organization's business and its customers, net (incremental) effect modelling may be a more valid approach to reach unbiased conclusions. Incremental modelling allows the effect of these interactions to be extracted in a 'pure' form from the data, as well as accounting for the effects of interactions between predictors within the model (Verbeke, Baesens, & Bravo, 2018, p. 157).

The first two types of models, which we define as "traditional", predict the outcome y based on a set of variables X). In the third type of models, the goal is to determine the impact of the communication influence t on the change in the outcome Δy, i.e., to provide a metric probability estimate of the increased chance of achieving the outcome with the influence, compared to the chance of the outcome without the influence. This effect cannot be directly observed empirically as it is the result of causal dependence. Hence the key methodological problem associated with the evaluation of incremental models, which consists in the impossibility of simultaneously implementing and not implementing personalized communication with the same individual and, thus, of observing differentially customer reactions. The only adequate empirical approach for collecting the data necessary for evaluating incremental models is the experimental one, in particular, following an adequately chosen experimental design to track marketing campaign information at the customer level. The purpose of the experiment is to measure the difference in consumer response in the presence and absence of communication, respectively. This difference can be interpreted as a causal effect. If we score it with τ_i for the i-th respondent, the expression would hold:

$$\tau_i = Y_i(1) - Y_i(0) \tag{1}$$

in which with $Y_i(1)$ we express the potential reaction of the respondent if he/she was subject to a personalized communication impact (e.g. via SMS, personalized banner ad, email), with $Y_i(0)$ the potential reaction of the same respondent if he/she was not (Gutierrez & Gerardy, 2016, pp. 2-3).

In the presence of some descriptive attributes X_i associated with respondent i (e.g., gender, age, geolocation, prior behaviour, etc.), it is possible to infer a so-called conditional average treatment effect CATE[5] (Abrevaya, Hsu, & Lieli, 2014).

[5] The acronym CATE was first introduced by Hahn (1998) and popularized by Heckman, Ichimura, Todd (1997; 1998).

$$CATE = E[Y_i(1) \mid X_i] - E[Y_i(0) \mid X_i] \quad (2)$$

However, such a model does not lend itself to optimization because either the causal effect τ_i nor the conditional mean effect of the impact on individual respondents can be the objects of empirical observation. For this reason, it is necessary to acquire the data in a controlled experimental order in which respondents are randomly subdivided into a target (experimental) group on which personalized communication is exerted and a control group whose behaviour is only subject to observation but not to communication influence. If we introduce a dichotomous variable D taking values 1 when communication is exercised and 0 when it is not, the observed response Y_i of the i-th respondent can be expressed as:

$$Y_i = DY_i(1) + (1 - D_i)Y_i(0) = \begin{cases} Y_i(1), \text{if } D_i = 1 \\ Y_i(0), \text{if } D_i = 1 \end{cases} \quad (3)$$

Hence, the estimate of the average conditional effect of the communication impact could be expressed as:

$$\widehat{CATE} = E[Y_i|X_i = x, D_i = 1] - E[Y_i|X_i = x, D_i = 0] \quad (4)$$

The latter expression represents the general form of an incremental predictive model if conditional independence holds. Conditional independence can only be ensured if respondents from the target audience are randomly assigned to experimental and control groups and this assignment does not depend on the values of any of the observed attributes X. This means that the possible response of a given user $\{Y_i\ (0/1\}$ is solely and exclusively a consequence of the characteristics of user X_i, but not of its membership in an a priori distribution across the groups observed during the experiment. In general, this condition can be expressed as (Hahn, 1998, p. 322):

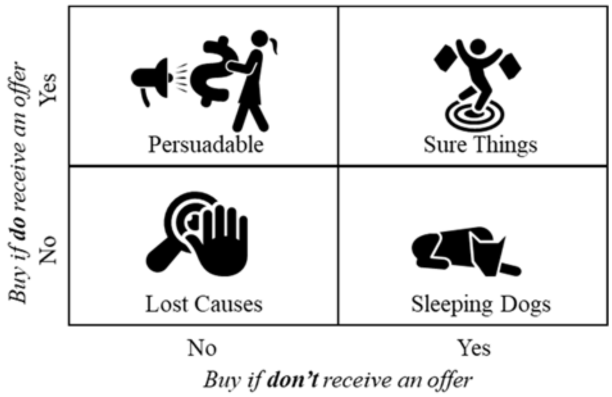
$$\{Y_i(0), Y_i(1)\} \perp D_i|X_i \quad (5)$$

If this condition is met, then we can assume that the observable response of respondent Y_i over the course of the experiment will depend solely on whether or not a personalized communication influence is exerted on the user.

3. Uplift Modelling Framework

As already mentioned, the purpose of the modelling is to distinguish responders from non-responders, and additionally to distinguish within the experimental and control groups those who respond as a result of the impact from those who respond without being impacted. In fact, within the group of non-responders, a further similar segmentation needs to be made based on the observed responses in the presence and absence of a targeted impact. Vittal (2006), Radcliffe (2007), Siegel (2011, p. 8; 2016, p. 270), and Kane et al. (2014) assume that four non-overlapping customer types are likely to exist in any personalized marketing campaign (see Figure 1).

Figure 1. Four customer types identified as a purchasing behaviour function when treated or not treated



Source: Adapted from Sigel (2016, p. 270) and Radcliffe (2007, p. 3).

The first supposed type of potential consumers are individuals who react negatively when they are the target of a marketing campaign. They will not buy if they are treated unless they react positively or do nothing. Some authors call this type of consumers “sleeping dogs” (Radcliffe, Generating incremental sales: Maximizing the incremental impact of cross-selling, up-selling and deep-selling through uplift modelling, 2007, p. 2). Obviously, customized targeting of such individuals would provoke the opposite of the desired effect and instead of the campaign generating additional revenue, it would provoke unjustified costs. Where the relative proportion of such individuals is relatively high in the target audience, it would be prudent from a cost-effectiveness perspective not to run a campaign as it would result in a net loss.

The second presumed type of potential customers are individuals who respond positively or buy, regardless of whether they have been the target of personalized communication influence. Quite conventionally, this type of consumer could be defined as “loyal” or “sure things”. Marketing to such individuals does not generate additional revenue, but it does generate additional costs (these are the fixed costs of contacting the potential customer). These additional costs can of course have a secondary communication effect of strengthening the relationship with the customer and hypothetically reducing the likelihood of churn. However, such effects are difficult to monetize and are ignored in incremental modelling.

The third potential customer type is non-responders (i.e. non-buyers), regardless of whether or not any personalized communication influence is exerted on them. They can be conventionally referred to as “indifferent”, although some authors define them as “lost causes” (Verbeke, Baesens, & Bravo, 2018, p. 159) or “invulnerable” (Radcliffe, Generating incremental sales: Maximizing the incremental impact of cross-selling, up-selling and deep-selling through uplift modelling, 2007, p. 2). Like “loyalists”, targeting “indifferents” does not generate additional revenue, only additional costs. However, these additional costs are generally lower compared to the additional costs of targeting ‘loyal’ customers, as “indifferents” do not respond and take advantage of the incentive offered, whereas ‘loyal’ do. Therefore, as objects of the campaign, the “loyals” are ‘more expensive’ than the “indifferents”.

The fourth possible type of potential customers are those who react positively (i.e. buy) when they are influenced by personalized communication and do not react if they are not influenced. Such ‘persuadable’ respondents are essentially those who should be the target segment of the campaign. They only buy (either more or earlier, depending on the context) if they are contacted. Targeted communication to those susceptible to persuasion generates additional revenue and once the costs of targeting other types of customers are deducted makes the campaign profitable. The incremental modelling aims to generate a list of exactly this type of customer to target for the campaign.

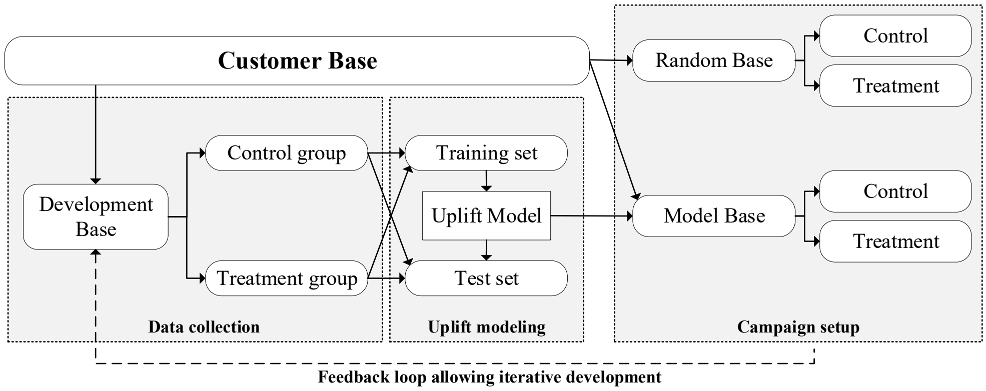
It goes with no doubt that the actual behaviour of the consumer in the presence or absence of a communication impact depends on the characteristics, channels, tools and incentives used in the marketing campaign itself, as well as on their personal characteristics. While the latter are independent variables that cannot be controlled, the tools used for the campaign can be optimized and personalized (even at a customer level) in order to maximize financial returns.

We also draw attention to the fact that the availability of the four described types of leads is not guaranteed in every available customer database used to model and optimize personalized marketing campaigns. In other words, any combination of proportions of the four customer types is possible within a given customer base. The exact proportions of these combinations depend on both the characteristics of the target audience and the characteristics of the campaign type. It is possible, for example, that there are no individuals described as ‘sleeping dogs’. In such a constellation, there is no risk of adverse impact. However, there may be no ‘persuadable’ customers. In such a scenario, a ‘red’ light should be given before launching a campaign, as no net revenue would be generated. Although these are hypothetically extreme scenarios, it should be borne in mind that, in general, the relative proportion of persuadable customers is almost always very small. This implies a very precise ex-ante analysis of the expected benefits and costs associated with the campaign (Verbeke, Baesens, & Bravo, 2018, p. 161).

But how do we identify these four types of leads? What data and methods of collecting it are applicable and adequate? How do we identify those who are susceptible to persuasion? This is a relatively complex process, and perhaps because of this, there are relatively opaque, fragmented and incomplete data collection methodologies in the research literature that are suitable for modelling the net effect of personalized marketing campaigns. However, they all agree that the main empirical challenge begins with the collection of appropriate data.

Since the purpose of the modelling is to assess the difference between two random events that are essentially mutually exclusive at the respondent level (in the sense that a particular user cannot be and not be both the target of a targeted communication effect and have customer reactions observed), the only possible way to collect an adequate set of baseline data is to conduct an experiment. The design of the experiment involves randomly dividing a representative portion of the target audience (e.g., a client base or a list of registered households) into two subsamples, a treatment subsample and a control subsample (for reference). Representatives of the treatment group receive the communication influence, while representatives of the control group do not receive the communication. Their reactions are observed and recorded over a period of time. Figure 2 illustrates a possible conceptual scheme for designing an experiment in order to collect the data needed to construct and evaluate an incremental model. Based on these data recorded during the experiment, a classification model measuring the net effect is constructed and estimated. Using this model, other members of the target audience are estimated and targeted.

Figure 2. Experimental design to collect the required data for uplift modeling



Source: (Verbeke, Baesens, & Bravo, 2018, p. 162)

This scheme can also be followed iteratively, with the selection of respondents into the experimental group after each run of the experiment now a combination of both new randomly selected respondents and individuals predicted by the model. This violates the principles of randomization but provides more efficient training of the final model.

However, it should be kept in mind that if the actual communication campaign relies on a different communication message (or an instrument of influence) than the one used during the experiment, the model will have a lower predictive ability.

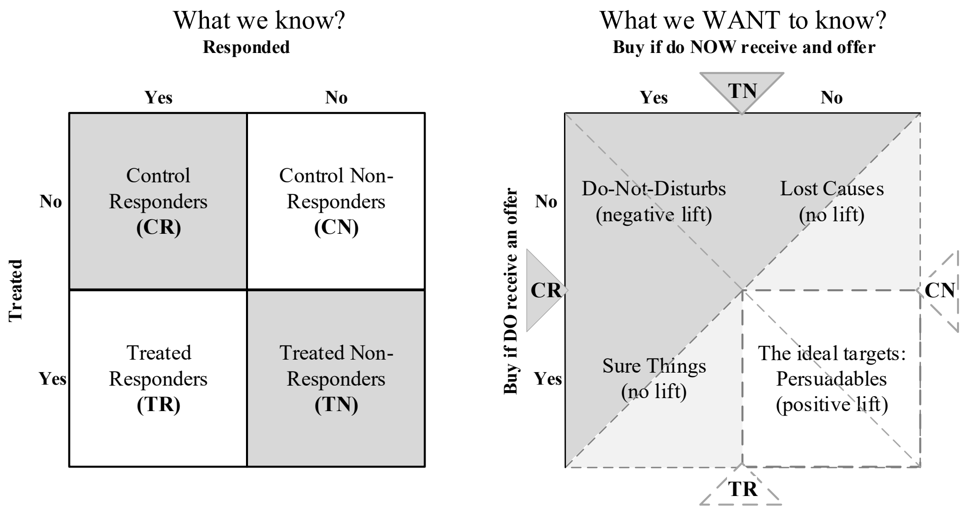
As a result of the data collected, it is possible to make the following hypothetical designations (Kane, Lo, & Zheng, 2014, pp. 222-223):

- Responders in the control group who responded (see (CR) left side illustration in Figure 3) were either in the "Loyal" or "Sleeping Dog" category, but in an unknown proportion - we only know of this fuzzy set as being composed of individuals who responded to being exposed to a communication impact. In these, both reaction and non-reaction can be observed if they are acted upon. It is logical to avoid them from personalized communication targeting in the future to save costs or to avoid provoking possible negative reactions.
- Respondents from the control group who did not respond (CN) were in unknown proportion from the neutral group of "Lost Causes" and from the group, "Persuadables". They did not respond during the experiment, but could theoretically both respond and not respond if a personalized influence were applied to them. It is very important to identify

the group of persuadables so that they can be included in the list for personalized targeting during the campaign.

- Respondents from the experimental group who responded (TR) were likely to be from either the loyal or the persuadable segment. The proportion between these two hypothetical classes is unknown to the researcher. The goal during the campaign should be related to targeting consumers susceptible to persuasion and avoiding the loyal group.
- Respondents who did not respond despite the treatment during the experiment (denoted by TN) represented a fuzzy set of neutrals and sleeping dogs. Hypothetically, those we define as sleeping dogs are also likely to respond if they were not targeted. The goal during the campaign should be related to avoiding sleeping dogs, as the associated costs are counterproductive and could even provoke negative reactions.

Figure 3. Customer types identifiable by their reactions

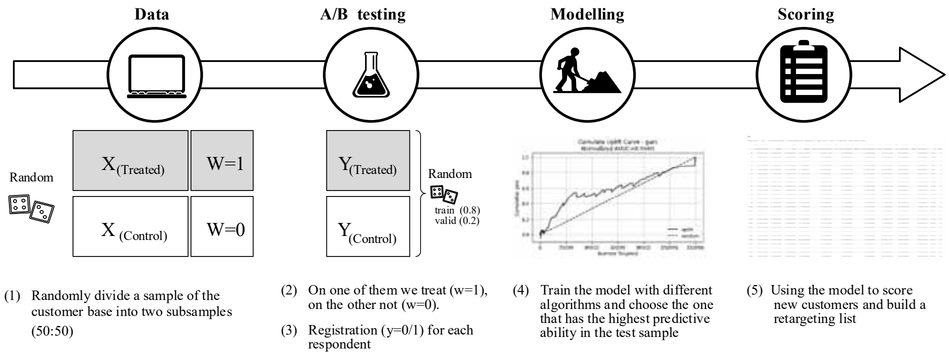


Source: Adapted from Kane, Lo, & Zheng (2014, p. 222).

Starting from the insight that the only important group to target is the persuadables (see Figure 3, right matrix), we come to the main technical problem related to their identification. The problem is clearly a classification one, since the dependent variable (consumer response) is logically viewed as dichotomous. Figure 4 illustrates schematically the conceptual logic of standing the classification model.

To train the model to predict the incremental score (net effect) of the campaign as accurately as possible at the individual respondent level, different approaches are possible. The first is indirect and is based on the use of two separate models, one predicting the probability of respondents responding without being influenced and the second predicting the probability of responding when influenced. This approach is intuitive, easy to follow and implement, and possible to implement without a full A/B test. The problem is its predictive ability. When using two parallel models, we effectively have two independent sources of systematic error, which hypothetically leads to a reduction in predictive accuracy. Furthermore, in this approach the net effect (the uplift) is not a target variable.

Figure 4. A conceptual scheme for building a predictive uplift model

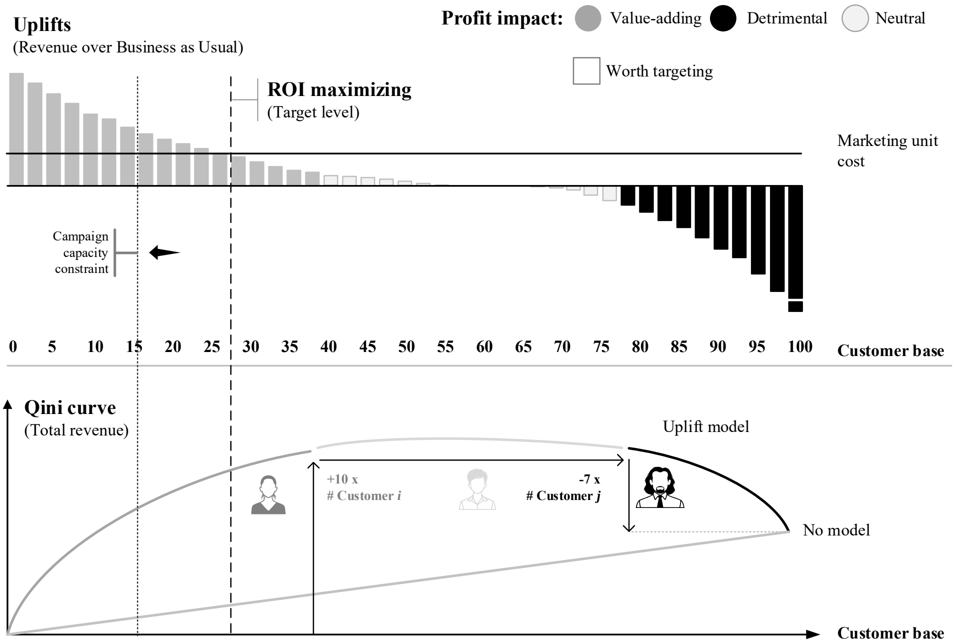


The second approach involves building and training a general model in which the target variable is precisely the net effect. In theory, this approach should be more accurate, but it is more difficult to train and in some algorithms is susceptible to overfitting. Recognizing these risks, we follow this approach when programming a prototype incremental model to predict the behaviour of consumers.

And we come to the core of the predictive model – its training algorithm. In practice, it is possible to apply all known supervised machine learning algorithms that predict the probability of occurrence of categories of a dichotomous variable as an algorithm for estimating and training such models. In this list, we can include purely statistical methods based on regression with nominal dependent variables (Lo, 2002; Lei & Wu, 2007; Kane, Lo, & Zheng, 2014), methods based on classification trees with different separation criteria, such as, e.g. CART (Breiman, Friedman, Olshen, Stone, & Olsen, 1984) or CHAID (Kass, 1980), as well as a number of modern ensemble methods, such as the random forest method (Guelman, Guillen, & Pérez-Marín, 2012; Guelman, Guillén, & Pérez-Marín, 2014; Soltys, Jaroszewicz, & Pzepakowski, 2015). Although it is not possible to claim a priori which of these methods provides the highest predictive ability of models, most practical applications are based on ensemble methods in particular. For the development of the proposed prototype, we step, namely, on the random forest method.

Each incremental model "produces" estimates of the net effect at the user level. These estimates can be interpreted as an absolute change in the probability that someone will respond if they are communicatively influenced instead of not. Based on these estimates, it is possible to sort potential customers in descending order and derive a list of those who have a positive net effect or a net effect above a certain threshold. This estimation concept and illustrated on the top of Figure 5. Starting from the assumption that communication is associated with a single variable cost for each contact, it is easy to derive a list of top-n users whose net effect would provide a positive campaign effect.

Figure 5. A conceptual framework for estimating the predictive ability of uplift models



Source: Adapted from Reinert & Zawisza (2020).

Since the choice of training algorithm as well as the tuning of the chosen algorithm itself affects the predictive ability of the model, it is possible to use some accompanying metrics and tools to evaluate the predictive accuracy of different variants of the classification model. Among the most popular are, for example, the cumulative incremental effect function (Uplift-curve) and the area under the uplift curve (AUUC), as well as its variant known as the Qini-curve and the area under this curve (AECU) (Radcliffe, 2007; Radcliffe & Surry, 2011). The higher an incremental model's predictive ability, the higher the maximum of these curves and, consequently, the larger the area under them. For comparison, a so-called "random" model is always used – i.e. if users are randomly selected up to a certain threshold – see the straight line in the bottom graph of Figure 5.

4. Prototyping Uplift Models

The practical deployment of the methodology described above faces a number of challenges. Firstly, the need for large data sets. For the machine learning of an incremental model, it is advisable to work with tens or even hundreds of thousands of records of recorded behaviour, and within a controlled experimental order of magnitude. Any A/B testing platform on the Internet or popular social networks could be used. The challenge is the computational power as well as the relatively complex programming code required to process the data. Various open-source programming libraries are known, such as \`Tools4uplift\` (Belbahri, Gandouet, Murua, & Nia, 2021), `grf` (Athey, Tibshirani, & Wager, 2019; Tibshirani, et al., 2022), `randomForest` (Breiman, Cutler, Liaw, & Wiener, 2022), and no longer maintained `uplift` package (Guelman, Guillén, & Pérez-Marín, 2014) using the R programming language. From our research and experimentation, we found that a much better environment and functionality is currently offered by incremental modelling libraries written for the Python programming language. Definitely, one of the most complete is the CasualML package (Uber Technologies, Inc, 2019), as well as `scikit-uplift` (Shevchenko , 2021). Trying to use the best of "both" worlds, for the practical implementation of the developed prototype, we used the Distributed Random Forest (DRF) package developed in Java for the open-source machine learning platform H2O.ai (Ambati, 2014). The advantage of this platform is the convenient interface and libraries to work directly with R (R Core Team, 2022) or Python (Python Software Foundation, 2022).

Of course, the choice of programming environment depends on many factors, but from our experience running intensive tests with an anonymized set of 150000 customer response records we could not find any significant advantages or disadvantages of the two programming environments.

The full Python code and all relevant instructions and outputs can be found on **Appendix A**. The complete R programming language code to perform the analogous procedure with the same data is provided in **Appendix B**.

5. ROI Impact of Customer Selections Based on Net Scoring

Compared to traditional digital marketing campaigns pursuing demand generation and/or retention, incremental modelling can provide a higher return on marketing investment because the focus is only on incremental responses. By targeting only the customers that can be persuaded through treatment within a campaign, the cost per contact, and therefore the return per unit spent, can be significantly improved.

In theory, customers can be divided into different types according to their behaviour with and without campaigns. Classical gross scoring using propensity modelling, on the one hand, and uplift modelling, on the other, have different capabilities to identify these customer types. In incremental modelling, the purchase or non-purchase of a product is no longer assumed for all customer types. Rather, there are different levels of random noise (i.e., purchase rate) that can be raised or lowered by campaigns to a certain degree (Michel, Schnakenburg, & von Martens, 2019, p. 256).

Assuming the hypothesis that the current customer base includes all four possible customer types (Sure Things, Persuadables, Sleeping Dogs, and Lost Causes), a reasonable assumption can be made that the positive financial impact of incremental modelling can be obtained both as a consequence of cost-cutting by not targeting Sure Things and by prompting additional purchases by targeting Persuadables (Radcliffe, 2007).

Michel, Schnakenburg, and von Martens (2019, pp. 256-284) consider several hypothetical scenarios, depending on the level of the purchase rate among the four types of customers, and prove that when selecting customers based on an uplift model, the total profit outperforms the total profit gained by the best classical scenario due to the fact that only those customers are targeted where a campaign provides an additional benefit in comparison to not targeting them. They also concluded that campaign net ROI is only influenced by the uplift and downlift but not by varying random noise (i.e., purchase rate) differences between different customer types (Michel, Schnakenburg, & von Martens, 2019, p. 284). Baier and Stöecker (Baier & Stöcker, 2021) also report similar conclusions after reproducible empirical experiments with publicly available datasets.

6. Limitations and Conclusion

When building uplift models, there are often hundreds of available features (predictors) for building such models. Maintaining all the features in a single model can be expensive and inefficient. Feature selection is an essential step in the modelling process for multiple reasons: improving the accuracy of estimates by removing irrelevant features, speeding up model training and prediction speed, reducing the workload of monitoring and maintaining the feature data pipeline, and providing a better opportunity for model interpretation and diagnosis. However, feature selection methods for elevation modelling are rarely discussed in the literature. Although various feature selection methods exist for standard machine learning models, these methods are not optimal for solving the problem of feature selection for modelling ascent processes. Research and development to solve this problem is still ongoing and some promising solutions are already available (Zhao, Zhang, Harinen, & Yung, 2022).

As a conclusion, we should emphasize that, unlike traditional models that solve the accuracy-maximization problem, incremental modelling is a much more reliable analytical approach to solve the performance-maximization problem of digital marketing campaigns. Of course, this approach should not be seen as a silver bullet. A number of pitfalls, such as choosing the right model estimation algorithm or the quality and balance of the data, can significantly impact the predictive ability of the model. Perhaps most relevant here would be the statement by John Tukey, one of the greatest statisticians of modern times, stating that „Far better an approximate answer to the right question, which is often vague, then an exact answer to the wrong question, which can always be made precise“ (Tukey, 1962, p. 13).

Of course, the frontiers of incremental modelling research have not been reached. Constantly evolving and emerging machine learning algorithms are constant challenges for new and novel experiments to train predictive uplift models and deploy them in marketing practice. One of the promising but not yet fully developed fields for experimentation and research is, for example, the non-parametric bayesian approach to estimate the incremental impact of a treatment (Rafla, Voisine, Crémilleux, & Boullé, 2022) and its application for optimizing direct marketing campaigns.

References

Abrevaya, J., Hsu, Y.-C., Lieli, R. P. (2014). Estimating Conditional Average Threatment Effects. - Journal of Business & Economic Statistics, 33(4), pp. 485-505.

Athey, S., Tibshirani, J., Wager, S. (2019). Generalized random forests. - The Annals of Statistics, 47(2), pp. 1148-1178. doi:10.1214/18-aos1709.

Baier, D., Stöcker, B. (2021). Profit uplift modeling for direct marketing campaigns: approaches and applications for online shops. - Journal of Business Economics, 92, pp. 645-673. doi:10.1007/s11573-021-01068-3.

Belbahri, M., Gandouet, O., Murua, A., Nia, V. P. (2021). 'tools4uplift': Tools for Uplift Modeling. CRAN.

Breiman, L., Cutler, A., Liaw, A., Wiener, M. (2022). 'randomForest': Breiman and Cutler's Random Forests for Classification and Regression. CRAN.

Breiman, L., Friedman, J. H., Olshen, R., Stone, C. J., Olsen, R. A. (1984). Classification and Regression Trees. Monterey, CA: Wadsworth and Brooks.

Debray, T. P., Damen, J. A., Snell, K. I., Ensor, J., Hooft, L., Reitsma, J. B., . . . Moons, K. G. (2017). A guide to systematic review and meta-analysis of prediction model preformance. The BMJ, 356:i6460. doi:10.1136/bmj.i6460.

Guelman, L., Guillen, M., Pérez-Marín, A. M. (2012). Random Forests for Uplift Modeling: An Insurance Customer Retention Case. - Lecture Notes in Business Information Processing, 115, pp. 123-133. doi:10.1007/978-3-642-30433-0_13.

Guelman, L., Guillén, M., Pérez-Marín, A. M. (2014). Optimal personalized treatment rules for marketing interventions: A review of methods, a new proposal, and an insurance case study (Vol. 6). Barcelona: UB Riskcenter Working Paper Series.

Gutierrez, P., Gerardy, J.-Y. (2016). Causal Inference and Uplift Modeling: A review of the literature. - Journal of Machine Learning Research: Workshop and Conference Proceedings, 67, pp. 1-13.

Hahn, J. (1998). On the Role of the Propensity Score in Efficient Semiparametric Estimation of Average Treatment Effects. - Econometrics, 66, pp. 315-331.

Heckman, J., Ichimura, H., Todd, P. (1997). Matching as an Econometric Evaluation Estimator: Evidence from Evaluating a Job Training Program. - Review of Economic Studies, 64, pp. 605-654.

Heckman, J., Ichimura, H., Todd, P. (1998). Matching as an Econometric Evaluation Estimator. - Review of Econometric Studies, 65, pp. 261-294.

Javaheri, S. H., Sepehri, M. M., Teimourpour, B. (2014). Response Modeling in Direct Marketing: A Data Mining-Based Approach for Target Selection. - In: Zhao, Y., Cen, Y. Data Mining Applications with R. Amsterdam: Academic Press, pp. 153-180, doi:10.1016/B978-0-12-411511-8.00006-2.

Kane, K., Lo, V. Y., Zheng, J. (2014). Mining for the truly responsive customers and prospects using true-lift modeling: Comparison of new and existing methods. - Journal of Marketing Analytics, 2(4), pp. 218-238.

Kass, G. V. (1980). An Exploratory Technique for Investigating Large Quantities of Categorical Data. - Applied Statistics, 29(2), pp. 119-127.

Lei, P., Wu, Q. (2007). Introduction to Structural Equation Modeling: Issues and Practical Considerations. - Educational Measurement: Issues And Practice, 26(3), pp. 33-43.

Lo, V. S. (2002). The True Lift Model - A Novel Data Mining Approach to Response Modeling in Database Marketing. - SIGKDD Explorations, 4(2), pp. 78-86.

Michel, R., Schnakenburg, I., von Martens, T. (2019). Targeting Uplift. Cham: Springer. doi:10.1007/978-3-030-22625-1.

Python Software Foundation. (2022). Python Language Referenceq , version 3.11. www.python.org.

R Core Team. (2022). R: A language and environment. R Foundation for Statistical Computing. Vienna, Austria. Retrieved from https://www.R-project.org/.

Radcliffe, N. J. (2007). Generating incremental sales: Maximizing the incremental impact of cross-selling, up-selling and deep-selling through uplift modelling. Stochastic Solutions Limited. Retrieved 4 12, 2022, from https://bit.ly/3xnfg1s.

Radcliffe, N. J., Surry, P. D. (2011). Real-World Uplift Modeling with Significance-Based Uplift Trees. Stochastic Solutions. https://bit.ly/3O5Hy5B.

Rafla, M., Voisine, N., Crémilleux, B., Boullé, M. (2022). A Non-Parametric Bayesian Approach for Uplift Discretization and Feature Selection. European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases. Grenoble, France: HAL. Retrieved 10 26, 2022, from https://hal.archives-ouvertes.fr/hal-03788912.

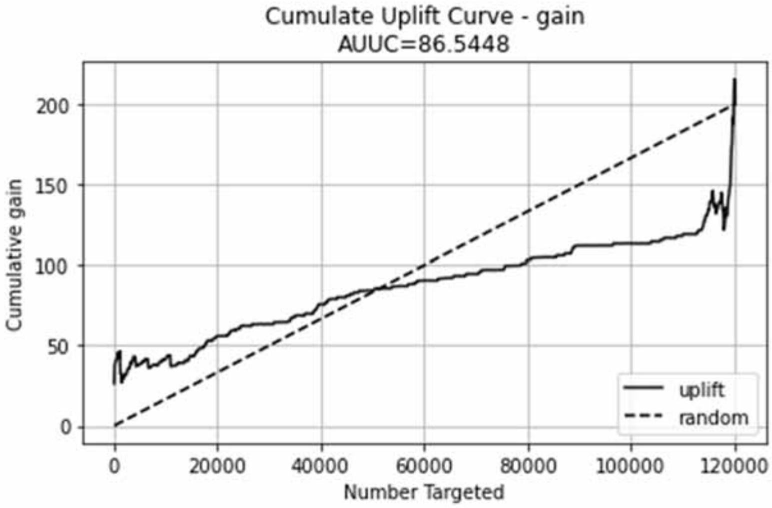
Reinert, A., Zawisza, M. (2020, May 26). Uplift modeling for marketing campaigns. Retrieved July 3, 2022, from https://bit.ly/3yCanBv.

Shevchenko, M. (2021). scikit-uplift [python]. https://www.uplift-modeling.com/en/latest/index.html.
Siegel, E. (2011). Uplift Modeling: Predictive Analytics Can't Optimize Marketing Decisions Without It. Prediction Impact and sponsored by Pitney Bowes Business Insight. Retrieved 4 12, 2022, from https://bit.ly/3vd6ct5.
Siegel, E. (2016). Preductive analytics: The power to predict who will click, buy, lie, or die. Hoboken: Wiley.
Soltys, M. S., Jaroszewicz, S., Pzepakowski, P. (2015). Ensemble Methods for Uplift Modeling. - Data Mining and Knowledge Discovery, 29(6), pp. 1531-1559.
Tibshirani, J., Athey, S., Friedberg, R., Hadad, V., Hirshberg, D., Miner, L., . . . Wright, M. (2022). 'grf': Generalized Random Forests. CRAN.
Tukey, J. W. (1962). The Future of Data Analysis. - The Annals of Mathematical Statistics, 33(1), pp. 1-67.
Uber Technologies, Inc. (2019). CausalML. https://causalml.readthedocs.io/en/latest/about.html.
Verbeke, W., Baesens, B., Bravo, C. (2018). Profit-Driven Business Analytics: A Practitioner's Guide to Transforming Big Data into Added Value. Hoboken, NJ: John Wiley & Sons, Inc.
Vittal, S. (2006). Optimal Targeting through Uplift Modeling: Generating higher demand and increasing customer retention while reducing marketing costs. Portrait Software.
Zhao, Z., Zhang, Y., Harinen, T., Yung, M. (2022). Feature Selection Methods for Uplift Modeling. Cornell University. doi:10.48550/arXiv.2005.03447.

Appendix A: Uplift model estimation using random forest with Python

```
# [1] Install required Python libraries
import h2o
from h2o.estimators import H2OUpliftRandomForestEstimator
h2o.init()

# [2] Data preparation¶
# Importing the dataset into H2O:
data = h2o.import_file("https://krst@data.eacademybg.com/varna2022/campaign150k_data.csv")
# [2.1] Viewing the data frame
data.head(6)
V1       V2       V3       V4       V5       V6       V7        V8       V9       V10      V11      V12       treatment conversion
24.618   10.0597  8.21438  4.67988  10.2805  4.11545  -1.28821  4.83381  3.97186  13.1901  5.30037  -0.168679 1         0
12.6164  10.0597  8.91077  4.67988  10.2805  4.11545  0.294443  4.83381  3.9554   13.1901  5.30037  -0.168679 1         0
22.0088  10.0597  8.21438  4.67988  10.2805  4.11545  -5.98767  4.83381  3.97186  13.1901  5.30037  -0.168679 1         0
22.174   10.0597  8.21438  4.67988  10.2805  4.11545  -1.28821  4.83381  3.97186  13.1901  5.30037  -0.168679 0         0
23.0819  10.0597  8.21438  4.67988  10.2805  4.11545  -1.28821  4.83381  3.97186  13.1901  5.30037  -0.168679 1         0
24.5316  10.0597  8.21438  4.67988  10.2805  4.11545  -4.59546  4.83381  3.97186  13.1901  5.30037  -0.168679 1         0

data.shape
(150000, 14)

# [2.2]  Set predictors (Xi), reponse (yi) and treatment (wi)¶
# Choosing predictors
predictors = ["V1", "V2", "V3", "V4", "V5", "V6","V7", "V8"]
# Set the response variable (yi) as a factor
response = "conversion"
data[response] = data[response].asfactor()
# Set the treatment (wi) as factor
treatment_column = "treatment"
data[treatment_column] = data[treatment_column].asfactor()

# [2.3] Split the dataset into a training and validation subsets: (80:20)
train, valid = data.split_frame(ratios=[.8], seed=5250)

# [3] Build and train the model
# [3.1] Model construction:
uplift_model = H2OUpliftRandomForestEstimator(ntrees=100, max_depth=10,
                                 treatment_column=treatment_column,
                                 uplift_metric="KL", min_rows=10,seed=5250,
                                 auuc_type="qini")
```

```
# [3.2] Training the model
uplift_model.train(x=predictors, y=response, training_frame=train,
                   validation_frame=valid)
...
...
...
# [4] Evaluating predictive performance of the model
# Performance evaluation:
perf = uplift_model.model_performance()
perf

ModelMetricsBinomialUplift: upliftdrf
** Reported on train data. **

AUUC: 73.61579189051791
AUUC normalized: 0.43332750179173507

AUUC table (number of bins: 1000): All types of AUUC value
         uplift_type          qini       lift       gain
0        AUUC value           73.615792  0.003023   86.544773
1        AUUC normalized      0.433328   0.003023   0.433183
2        AUUC random value    85.026676  0.000833   99.993177

Qini value: -11.410884352633914


# [5] Make predictions on validation subset
pred = uplift_model.predict(valid)
pred.head(6)
uplift_predict        p_y1_ct1             p_y1_ct0
0.000833997           0.00159892 0.000764924
0.000855111           0.00163856 0.000783453
0.000859351           0.00164198 0.000782633
0.000694148           0.00141544 0.000721296
0.000694148           0.00141544 0.000721296
0.000706745           0.00141739 0.000710649

# Plotting Uplift-curve from performance
perf.plot_uplift(metric="gain", plot=True)
```




```
# Get Qini AUUC
print(perf.auuc())
73.61579189051791
```

```
# Get all AUUC values as a table
print(perf.auuc_table())

AUUC table (number of bins: 1000): All types of AUUC value
          uplift_type          qini       lift       gain
0         AUUC value 73.615792  0.003023   86.544773
1         AUUC normalized       0.433328   0.003023   0.433183
2         AUUC random value     85.026676  0.000833   99.993177

# Get thresholds and metric scores
print(perf.thresholds_and_metric_scores())
Metrics for Thresholds: Cumulative Uplift metrics for a given percentile
```

	thresholds	qini	lift	gain	qini_normalized	lift_normalized	gain_normalized	qini_random	lift_random	gain_random	n	idx
0	0.021879	23.833333	0.218654	26.457187	0.140291	0.218654	0.132426	0.171186	0.000002	0.201319	121	0
1	0.007815	32.457143	0.157559	37.971706	0.191054	0.157559	0.190060	0.340958	0.000003	0.400974	241	1
2	0.004210	34.530612	0.110675	39.953689	0.203259	0.110675	0.199980	0.510730	0.000005	0.600629	361	2
3	0.003049	35.460317	0.084833	40.804815	0.208731	0.084833	0.204240	0.680502	0.000007	0.800284	481	3
4	0.002468	36.513514	0.069286	41.640648	0.214931	0.069286	0.208424	0.850273	0.000008	0.999939	601	4
5	0.002092	36.850575	0.058124	41.907357	0.216915	0.058124	0.209759	1.020045	0.000010	1.199595	721	5
6	0.001855	39.264151	0.053421	44.926736	0.231122	0.053421	0.224872	1.189817	0.000012	1.399250	841	6
7	0.001696	40.000000	0.047790	45.925926	0.235454	0.047790	0.229873	1.359588	0.000013	1.598905	961	7
8	0.001583	36.724138	0.039235	42.413240	0.216171	0.039235	0.212291	1.529360	0.000015	1.798560	1081	8
9	0.001480	40.041916	0.038725	46.509034	0.235700	0.038725	0.232792	1.699132	0.000017	1.998215	1201	9
10	0.001399	28.016216	0.024662	32.578716	0.164913	0.024662	0.163066	1.868903	0.000018	2.197870	1321	10
11	0.001341	28.553922	0.023083	33.262895	0.168078	0.023083	0.166491	2.038675	0.000020	2.397525	1441	11
12	0.001286	23.200000	0.017288	27.003279	0.136563	0.017288	0.135160	2.209862	0.000022	2.598844	1562	12
13	0.001249	24.983051	0.017277	29.060506	0.147059	0.017277	0.145457	2.379633	0.000023	2.798499	1682	13
14	0.001227	26.124031	0.016920	30.489316	0.153775	0.016920	0.152608	2.549405	0.000025	2.998154	1802	14
15	0.001206	25.626866	0.015494	29.779224	0.150848	0.015494	0.149054	2.719177	0.000027	3.197810	1922	15
16	0.001189	25.680702	0.014616	29.846325	0.151165	0.014616	0.149390	2.888948	0.000028	3.397465	2042	16
17	0.001177	27.661238	0.014912	32.239135	0.162823	0.014912	0.161367	3.058720	0.000030	3.597120	2162	17
18	0.001166	27.654321	0.014124	32.230419	0.162783	0.014124	0.161323	3.228492	0.000032	3.796775	2282	18
19	0.001155	27.482353	0.013328	32.013876	0.161770	0.013328	0.160239	3.398264	0.000033	3.996430	2402	19

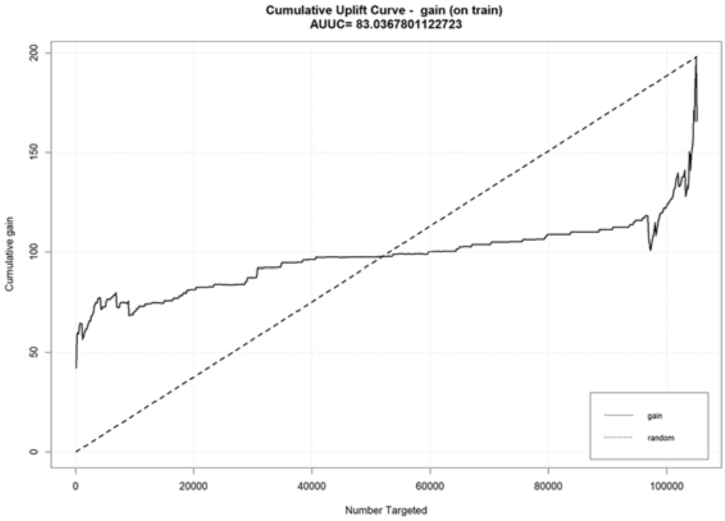
```
See the whole table with table.as_data_frame()

# Get Qini value
print(perf.qini())
-11.410884352633914

# Get AECU values as a table
print(perf.aecu_table())
AECU values table: All types of AECU value
          uplift_type          qini       lift       gain
0         AECU value -         11.410884  0.002191   -13.448404

# [6] Generate a targeting list ...
import pandas as pd
import numpy as np
retarget = pred.as_data_frame()
print(retarget)
       uplift_predict  p_y1_ct1  p_y1_ct0
0            0.000834  0.001599  0.000765
1            0.000855  0.001639  0.000783
2            0.000859  0.001642  0.000783
3            0.000694  0.001415  0.000721
4            0.000694  0.001415  0.000721
...               ...       ...       ...
29916        0.000711  0.001409  0.000698
29917        0.000855  0.001639  0.000783
29918        0.000710  0.001407  0.000697
29919        0.000834  0.001599  0.000765

[29920 rows x 3 columns]

retarget.sort_values(by="uplift_predict", ascending=False)
          uplift_predict       p_y1_ct1   p_y1_ct0
27156     0.348100             0.737838   0.389738
3951      0.258166             0.646571   0.388405
10159     0.196595             0.513095   0.316500
283       0.177039             0.516372   0.339333
23119     0.155474             0.457569   0.302095
...       ...         ...                 ...
16988     -0.265292            0.325839   0.591131
3108      -0.279841            0.356671   0.636512
21128     -0.287042            0.362041   0.649083
19369     -0.287148            0.355435   0.642583
26857     -0.289549            0.347701   0.637250
```

```
29920 rows × 3 columns

# Filter users with an uplift score above 0.3
print(retarget.query("`uplift_predict` > 0.1"))
       uplift_predict  p_y1_ct1  p_y1_ct0
283          0.177039  0.516372  0.339333
1457         0.154484  0.404246  0.249762
3951         0.258166  0.646571  0.388405
8245         0.116092  0.411742  0.295650
8316         0.137342  0.444509  0.307167
10159        0.196595  0.513095  0.316500
14691        0.137385  0.404885  0.267500
17646        0.123868  0.422574  0.298705
18728        0.141731  0.504136  0.362405
19072        0.121187  0.448187  0.327000
19319        0.122234  0.481234  0.359000
23119        0.155474  0.457569  0.302095
24615        0.143553  0.387815  0.244262
27156        0.348100  0.737838  0.389738
27726        0.107917  0.443774  0.335857

# List of the Top-100 customers with the highest uplift
retarget['uplift_predict'].nlargest(n=100)
27156    0.348100
3951     0.258166
10159    0.196595
283      0.177039
23119    0.155474
           ...
18445    0.002946
5711     0.002795
28534    0.002773
4133     0.002738
Name: uplift_predict, Length: 100, dtype: float64
```

Appendix B: Uplift model estimation using random forest with R

```
# (1) Loading R packages ----
library(h2o)
h2o.init()
library(tidyverse)

# (2) Data preparation----
Uplift  <- h2o.importFile("https://krst@data.eacademybg.com/varna2022/campaign150k_data.csv")
h2o.head(uplift)
h2o.str(uplift)
dim(uplift)
colnames(uplift)
h2o.mean(uplift$treatment, na.rm = TRUE)
h2o.mean(uplift, return_frame = TRUE)

# (3) Set predictors (Xi), reponse (yi) and treatment (wi)----
predictors <- c("V1", "V2", "V3", "V4", "V5", "V5", "V7", "V8")

# (4.1) Set the response variable as factor----
uplift$conversion <- as.factor(uplift$conversion)

# (4.2) Set treatment variable as factor....
uplift$treatment <- as.factor(uplift$treatment)
h2o.str(uplift)
```

```
# (5) Split the dataset into a training and validation subsets: (80:20)----
data_split <- h2o.splitFrame(data = uplift, ratios = 0.7, seed = 5250)
train <- data_split[[1]]
test <- data_split[[2]]
h2o.head(test)
dim(train)
dim(test)

# (6) Build and train the model----
uplift.model <- h2o.upliftRandomForest(training_frame = train,
                                       validation_frame = test,
                                       x=predictors,
                                       y="conversion",
                                       ntrees = 100,
                                       max_depth = 10,
                                       treatment_column = "treatment",
                                       uplift_metric = "KL",
                                       min_rows = 10,
                                       seed = 5250,
                                       auuc_type = "qini")

# (7) Evaluating predictive performance of the model ----
perf <- h2o.performance(uplift.model)
perf
H2OBinomialUpliftMetrics: upliftdrf
** Reported on training data. **
** Metrics reported on Out-Of-Bag training samples **
Default AUUC:  83.03678
All types of AUUC:
AUUC table (number of bins: 1000): All types of AUUC value
        uplift_type      qini     lift      gain
1        AUUC value 83.036780 0.005005 97.472329
2   AUUC normalized  0.589343 0.005005  0.588269
3 AUUC random value 70.518418 0.000789 82.928738
Default Qini value: 12.51836
All types of AECU values:
AECU values table: All types of AECU value
  uplift_type      qini     lift      gain
1  AECU value 12.518362 0.004216 14.543590
# (8) ] Make predictions on test subset ----
predict <- h2o.predict(uplift.model, newdata = test)
predict
> predict
  uplift_predict    p_y1_ct1     p_y1_ct0
1   0.0007126356 0.001567692 0.0008550563
2   0.0007409614 0.001568139 0.0008271779
3   0.0007246861 0.001534196 0.0008095104
4   0.0007126356 0.001567692 0.0008550563
5   0.0006557158 0.001450066 0.0007943504
6   0.0006612131 0.001458709 0.0007974959

[44875 rows x 3 columns]

# (9) Ploting Uplift Curve ----
options(scipen=999)
plot(perf, metric="gain")
```




```
# Get Qini AUUC
print(h2o.auuc(perf))
[1] 83.03678

# Get all AUUC values as a table ----
> print(h2o.auuc_table(perf))
AUUC table (number of bins: 1000): All types of AUUC value
        uplift_type      qini     lift      gain
1        AUUC value 83.036780 0.005005 97.472329
2   AUUC normalized  0.589343 0.005005  0.588269
3 AUUC random value 70.518418 0.000789 82.928738

# Get thresholds and metric scores
> print(h2o.thresholds_and_metric_scores(perf))
Metrics for Thresholds: Cumulative Uplift metrics for a given percentile
  thresholds      qini     lift      gain qini_normalized lift_normalized
1   0.027349 37.000000 0.397849 42.172043        0.262603        0.397849
2   0.002585 50.000000 0.274725 57.967033        0.354868        0.274725
3   0.001873 53.000000 0.189964 60.028674        0.376161        0.189964
4   0.001558 53.000000 0.141711 59.660428        0.376161        0.141711
5   0.001325 53.000000 0.113006 59.441365        0.376161        0.113006
  gain_normalized qini_random lift_random gain_random   n idx
1        0.254519    0.142070    0.000002    0.167072 106   0
2        0.349845    0.282800    0.000003    0.332569 211   1
3        0.362288    0.423530    0.000005    0.498065 316   2
4        0.360065    0.564259    0.000006    0.663562 421   3
5        0.358743    0.704989    0.000008    0.829058 526   4
# Get Qini value
> print(h2o.qini(perf))
[1] 12.51836

# Get AECU values as a table
> print(h2o.aecu_table(perf))
AECU values table: All types of AECU value
  uplift_type      qini     lift      gain
1  AECU value 12.518362 0.004216 14.543590
```

```
# (10) Get list of the Top-n customers by different criteria
> head(predict)
  uplift_predict    p_y1_ct1     p_y1_ct0
1   0.0007126356 0.001567692 0.0008550563
2   0.0007409614 0.001568139 0.0008271779
3   0.0007246861 0.001534196 0.0008095104
4   0.0007126356 0.001567692 0.0008550563
5   0.0006557158 0.001450066 0.0007943504
6   0.0006612131 0.001458709 0.0007974959

df <- as.data.frame(predict)
library(plyr)
> head(arrange(df, desc(uplift_predict)), n = 10)
   uplift_predict  p_y1_ct1  p_y1_ct0
1       0.3874419 0.7321086 0.3446667
2       0.3874419 0.7321086 0.3446667
3       0.3561692 0.6545026 0.2983333
4       0.3541946 0.5996946 0.2455000
5       0.3293510 0.6260177 0.2966667
6       0.3130664 0.5585664 0.2455000
7       0.3085958 0.5540959 0.2455000
8       0.2569252 0.6004252 0.3435000
9       0.1723426 0.5340092 0.3616667
10      0.1642396 0.5255729 0.3613333

# Filtering of customers with uplift score above 0.15
library(dplyr)
> df %>% filter(uplift_predict >0.15)
   uplift_predict  p_y1_ct1  p_y1_ct0
1       0.3085958 0.5540959 0.2455000
2       0.1642396 0.5255729 0.3613333
3       0.1723426 0.5340092 0.3616667
4       0.3293510 0.6260177 0.2966667
5       0.3541946 0.5996946 0.2455000
6       0.3874419 0.7321086 0.3446667
7       0.1639728 0.5368062 0.3728333
8       0.3130664 0.5585664 0.2455000
9       0.3561692 0.6545026 0.2983333
10      0.2569252 0.6004252 0.3435000
11      0.1568397 0.5094349 0.3525952
12      0.3874419 0.7321086 0.3446667
```