Assessing the real impact of marketing campaigns on consumer behaviour with uplift modelling

Mathias Cardarello Fierro (ID 963346)[*]
Wyllyan Canielli Scibelli (ID 963246)[*]

[*] Università degli Studi di Milano
25 March 2021

Abstract Marketing actions have been adopted as the main strategy that firms have to incentive the propensity of buying of their consumers. This paper introduces uplift modelling as a novel approach to estimate the causal effect of a marketing treatment, which facilitates targeting actions to responsive customers and efficient allocation of marketing resources. Uplift modelling has received increasing interest in the data analytics community as an improved framework for predictive analytics for data-driven decisions. A simulated dataset of a promotion campaign with two kinds of discounts was used to assess the impact on incremental sales, comparing the purchase uplift score between them, and to highlight the advantages of this approach over response models that only estimate the net buying propensity of customers.

Keywords causal inference, Class Variable Transformation, customer behaviour, marketing campaign, uplift modelling.

Introduction

According to Earley S. (2018) the more we know about our customers, the better we can stand out from our competitors. Understanding consumer behaviour, defined as “the behaviour that consumers display in searching for purchasing, using, evaluating and disposing of products, services and ideas” (Schiffman and Kanuk, 1997), is essential for developing the marketing strategy of firms. The increasing influence of advertising and promotions on the buying decisions of consumers undermines the rationality assumption of the theory of consumer behaviour, by tending to make them buy on impulse without having to go through the opportunity cost[1] of their decisions. To carry out this strategy, it is necessary to develop an entire data environment that allows decision-makers to create value through understanding customer necessities based on data quality, availability, modelling, and working on the data to obtain insights.

Boudet et al. (2018) showed that marketing strategies based on data-driven customer interactions in real-time can improve between 15% and 20% digital sales, even more, while significantly improving the Return of Investment (ROI)[2] on marketing spend across different channels. Following the same proposition, Fuller and Haefliger (2020) stated that "When firms mobilize unique proprietary behavioural customer data to give non-obvious value-adding anticipation of future customer need, the superior customer value can translate into competitive advantage, it can be obtained by design new products or services, based on customer behaviour modelling”.

[1] Opportunity costs represent the potential benefits an individual, investor, or business misses out on when choosing one alternative over another.
[2] Return on Investment (ROI) is a performance measure used to evaluate the efficiency or profitability of an investment or compare the efficiency of a number of different investments. ROI tries to directly measure the amount of return on a particular investment, relative to the investment’s cost.

The authors also indicated that this kind of marketing strategy based on data analytics[3] is used by companies in mainly different sectors to promote competitive advantages, to anticipate and articulate market necessities and gain competitive advantages launching or communicating new products or solutions. To achieve these goals, nowadays companies are implementing a data information system to be able to capture and analyse the whole client information, in an architecture known as Customer Relation system Management. A notorious example of the relevance of data when making decisions was the strategy of Netflix in making the movie "Roma", in which the company launched the movie based on insights about customer preferences. This approach, carried out by a data modelling approach, allowed to make a film that won important prizes as well as improved the revenue of the streaming platform (Fernandez-Manzano et al., 2016).

Once understood to be of strategic value, Hoffman and Fodor (2010) work on defining a new set of measurement techniques to capture the real value of marketing activities[4] in terms of ROI. The authors proposed the parameters to build a regular metric that can be compared with other kinds of investments, particularly because this is a common way to measure investments very usual for Chief Executive Officers (CEO) and Chief Financial Officers (CFO) across all sectors. The discussion was proposed to enrich the way marketing campaigns are measured regarding other investments, facilitating the analysis of the trade-off among investment decisions. These controversial topics are also discussed by Ambler et al. (2002), where there is an analysis of indicators applied in Spanish and UK markets to evaluate marketing campaigns. What is more, Ambler and Kokkinaki (2010) explore the concept of success in marketing, defining a set of indicators to represent the real impact of marketing campaigns. In conclusion, the common problem developed in these discussions is the right business measure to capture, compare and interpret the results of marketing actions.

Nevertheless, identifying the most accurate model to compare the conversion rate[5] between exposed and unexposed clients is a topic that has not been extensively treated in the literature. This requires the development of statistical models to capture the real impact that marketing actions have, and convert these results into business indicators that allows decision-makers to interpret them in the same sense as other investments.

In order to understand the right technique to measure campaign results, it is important to define the concept and the goal of this kind of actions. As described before, a marketing campaign can impact a customer throughout it whole lifetime through the realization of a campaign or a direct action, such as an email to promote a new product, and SMS to incentive a new purchase or a phone call to reactivate a client.

The aim of our work is to estimate the differential effect of a marketing campaign on consumers behaviour, using a novel technique for predictive modelling in a causal inference framework, i.e. uplift modelling. By using a simulated dataset in which the population is divided according to the exposure to two kinds of treatments, we compute the effectiveness of the campaigns by comparing the purchase uplift score between them.

---

[3] Data analytics is the science of analyzing raw data in order to make conclusions about that information.
[4] Marketing activities refer to the things an individual or organization undertakes to boost sales and also to improve its brand. Marketing activities are the set of processes for creating effective communication, exchanging, and delivering offerings that would add value to the customer.
[5] A conversion rate records the percentage of users who have completed a desired action (for example, a purchase). Conversion rates are calculated by taking the total number of users who 'convert', dividing it by the overall size of the audience and converting that figure into a percentage.

I. METHODOLOGY AND DATA DESCRIPTION

A. Introduction to predicting modelling

The popularization of data analytics among an increasing number of firms in the last decades boosted the development of predictive models that support decision making by foreseeing the user response behaviour, for example, from a particular marketing activity.

In the marketing sector, there are two categories of models commonly used: response and uplift models. In the first case, the model aims to predict the conversion probability of the recipients of a campaign with supervised classification algorithms[6], using a group of descriptive variables that characterize customers. However, uplift modelling is a set of techniques that estimates the differential impact of a marketing action on the buying propensity of an individual customer, i.e., the causal relationship between the action and customer reaction. As causal inference is crucial to measure the true impact of a marketing campaign, maximize campaign revenue, and allocate marketing resources efficiently (Gubela et al., 2020), we will focus our work on the application of uplift model strategies.

B. Uplift modelling

Uplift modelling, also known as incrementality modelling or persuasion modelling, is a predictive modelling technique that rely on randomized experiments to estimate the effect that a decision variable (treatment) has on an outcome variable (response), i.e., the conditional average treatment effect. It has been commonly applied in marketing, personalized medicine, and also in political campaigns.

In the marketing domain, uplift modelling has helped to improve the return on marketing investment by segmenting the customer base into four categories according to the recommendations of the model, as shown in Table 1: Four customer segments based on how a marketing action affect their behaviour. Customers who need a direct marketing nudge, such as a discount, to have a desired response (e.g., purchase) are categorized as *persuadable*. On the other hand, the *sleeping dogs* segment includes customers that have a negative reaction to the campaign, while they would have responded if they were not contacted. Finally, the customers in the third and fourth categories either never respond to any offer (lost causes) or always respond regardless of it (*sure things*). Therefore, the interest lies in targeting a marketing action to the *persuadable* customers, avoiding the other segments, which implies an efficient use of limited resources, thus an increase in business value (Gubela et al., 2019).

<table>
<tr><td colspan="2" rowspan="2"></td><td colspan="2">Desirable outcome if not treated</td></tr>
<tr><td>YES</td><td>NO</td></tr>
<tr><td rowspan="2">Desirable outcome if treated</td><td>YES</td><td>Sure things</td><td>Persuadable</td></tr>
<tr><td>NO</td><td>Sleeping dogs</td><td>Lost causes</td></tr>
</table>

Table 1: Four customer segments based on how a marketing action affect their behaviour

[6] Supervised learning refers to a class of algorithms that determine a predictive model using data points with known outcomes. The model is learned by training through an appropriate learning classification algorithm, such as random forests or neural networks, that typically works through some optimization routine to minimize a loss or error function.

To carry out the causality analysis, uplift models require a stratified sample of customers divided into two groups, a treatment group that received the marketing action and a control group that was not exposed. This data usually is collected through a randomized control trial, for example, an A/B test[7].

In the basic notation of the econometric literature, based on Gutierrez and Gérardy (2016), we denote $Y_i(1)$ person i's outcome variable when he receives the active treatment and $Y_i(0)$ person i's outcome when he receives the control treatment. The causal effect τ_i of the active treatment is given by:

$$\tau_i = Y_i(1) - Y_i(0)$$

As τ_i is not observed, there were proposed a set of strategies to estimate the causal effect, which can be classified into three main categories, according to Gubela et.al. (2019): Basic, Advanced and Special.

The basic approach to predict t_i is called Two-Model method, which uses two separate probabilistic models to estimate $P(Y_i = 1|\, X_i)$ on treatment and control groups. Uplift then can be computed as the difference in the outcome probabilities caused by the treatment:

$$\tau_i(X_i) = P(Y_i = 1|\, T_i, X_i) - P(Y_i = 1|\, C_i, X_i)$$

where Y_i is the outcome variable (e.g., purchase), T_i and C_i indicate the membership of customer i to the treatment or control group, and X_i represent a set of explanatory variables that describes customer characteristics. In contrast, a response model only estimates $P(Y_i = 1|\, T_i, X_i)$, i.e., the net buying propensity of customers.

Among the advanced approaches, Jaskowski and Jaroszewicz (2012) proposed the Class Variable Transformation (CVT) for the binary observed outcome $Y_i^{obs} = \{0, 1\}$. The method consists in creating a new target variable Z_i as follows:

$$Z_i = Y_i^{obs} W_i + (1 - Y_i^{obs})(1 - W_i)$$

where W_i is a binary variable taking on value 1 if the person i receives the active treatment. Then, Z_i is equal to one if the observation belongs to the treatment group and $Y_i^{obs}= 1$, or the observation belongs to the control group and $Y_i^{obs} = 0$. In all other cases, the target variable takes on value zero. Under the assumption that control and treated groups are balanced across all profiles of individual, the authors proved that $\tau_i(X_i)$ can be estimated by modelling $P(Z_i = 1|\, X_i)$, i.e., $E(Z_i = 1|\, X_i)$, taking into account that:

$$\tau_i(X_i) = P(Y_i = 1|\, T_i, X_i) - P(Y_i = 1|\, C_i, X_i) = 2P(Z_i = 1|\, X_i) - 1$$

Although the required restrictive assumptions, binary outcome and balanced dataset, the CVT method tends to perform better than the Two-Model approach while remaining simple (Gutierrez and Gerardy, 2016). In this paper we use an extension of CVT, based on Lai's Generalized Weighed Uplift Method (LGWUM) presented in Kane et al. (2014). This approach assumes that the positive uplift lies in correctly identified *persuadables*, whilst the negative uplift can be found in the *sleeping dogs'* group. The uplift measure is defined as:

$$\tau_i(X_i) = P(Y_i = 1|\, T_i, X_i) + P(Y_i = 0|\, C_i, X_i) - P(Y_i = 0|\, T_i, X_i) - P(Y_i = 1|\, C_i, X_i)$$

[7] A/B testing is a user experience research methodology used to compare two versions of a single variable, typically by testing a subject's response to variant A against variant B, and determining which of the two variants is more effective.

Finally, we use the classification learning algorithm XGBoost[8] to estimate $P(Z_i = 1 | X_i)$, an implementation of gradient boosted decision trees that is increasingly used thanks to its great performance.

So as to evaluate the performance of the model, as we cannot observe the effect of being treated and not treated on an individual at the same time, an adapted version of the Gini coefficient called Qini coefficient is often used (Devriendt et al., 2021). For binary outcomes, the Qini coefficient is defined as the area between the actual uplift gains curve and a diagonal to that of the optimum Qini curve. While the first curve shows the cumulative number of incremental positive outcomes (uplift) as a function of the number of customers treated, the latter is a diagonal line representing random targeting.

C. Dataset

Due to the difficulty to obtain a real-world marketing campaign dataset, mainly for privacy reasons, we use fictional campaign data published on 2020-02-09 on the platform Kaggle[9]. It consists of a marketing promotion campaign with a total of 64,000 observations, which are split into three segments: two treatment groups, depending on whether the customer was exposed to a Buy One Get One (BOGO)[10] or a 50% discount campaign, as well as a control group that didn't receive the treatment. The dataset includes 9 variables related to customer purchase history, demographic information, if they buy or not (conversion rate) and some other relevant indicators for marketing actions. Table 2 provides detailed information about the variables of the dataset.

Variable	**Description**
Recency	Months since last purchase
History	Value of the historical purchases
Used_discount	Indicates if the customer used a discount before
Used_bogo	Indicates if the customer used a buy one get one before
zip_code	Class of the zip code as Suburban/Urban/Rural
Is_referral	Indicates if the customer was acquired from referral channel
Channel	Channels that the customer using, Phone/Web/Multichannel
Offer	The offers sent to the customers, Discount/But One Get One/No Offer
Conversion	Indicates if the customer buys or not

Table 2: Dataset details

II. Empirical analysis

In this section we present a brief exploratory analysis of our dataset, pointing out the main characteristics of the customers subjected to the fictional experiment. After performing a sanity check, we report the results of the uplift modelling based on the CVT approach and compare the purchase uplift score between BOGO and discount-based marketing campaigns. The code developed to perform the analysis in Python is in Appendix.

---

[8] XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework, designed for speed and performance.
[9] Kaggle is an online community of data scientists and machine learning practitioners that allows users to find and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers, and enter competitions to solve data science challenges.
[10] Buy One Get One is a common form of sales promotion in which customers can buy the same two products for the price of one.

A. *Exploratory data analysis*

Initially, we can observe in Figure 1 the distribution of the number of months since the last purchase (recency), where there is a greater concentration of customers who bought in the last 3 months (35%) or customers who bought between 9 and 10 months ago (22%). The average period since the last purchase is 5.8 months with a standard deviation of 3.5 months.

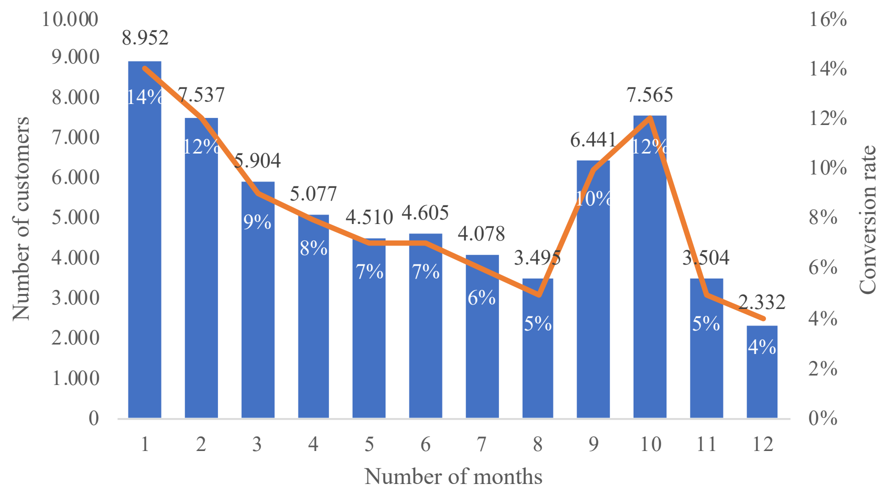

Figure 1: Distribution of the number of months since last purchase

Customers are distributed in three geographic profiles, being about 28,700 Suburban (44.9%), 25,600 Urban (40.0%) and 9,600 thousand Rural (14.9%). When the distribution of recency is considered between the three different profiles, the same sharing of customers is observed, as shown in Figure 2.

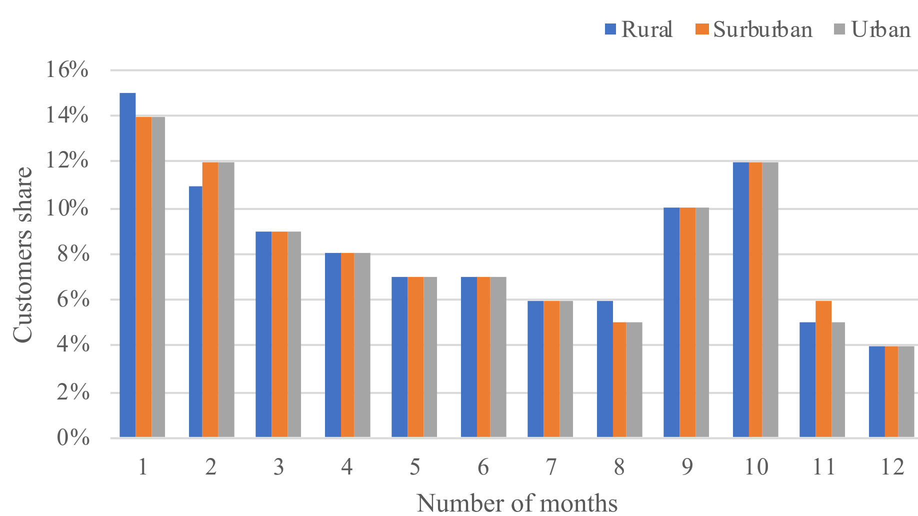

Figure 2: Distribution of customers by the number of months since last purchase, per geographic profile

Regarding the expense per customer, the average across all customers is USD 242 per purchase. This value has a marginal change by geographic profile, as shown in Table 3.

Geographic profile	Average expense per customer (USD)
Rural	243
Suburban	240
Urban	244
Total	**242**

Table 3: Average expense per customer in USD, per geographic profile

However, Figure 3 indicates that there is a greater variation in the average ticket value related to the last month of purchase, because a customer who buys on more recent dates has a higher average expenditure than those whose last purchase was earlier. This variation remains the same when we look at the different geographic profiles over time.

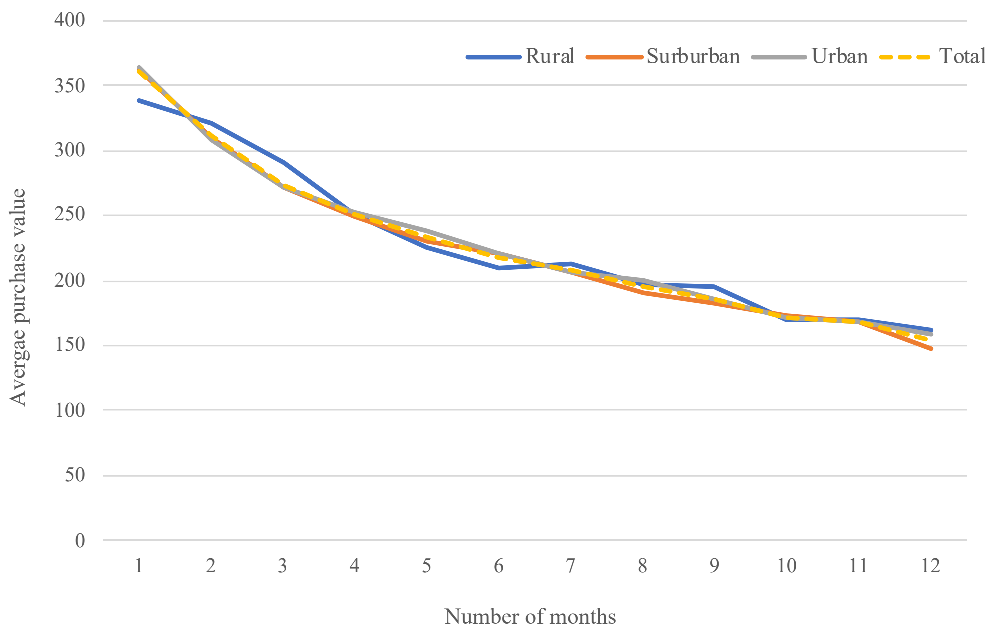

Figure 3: Average purchase value per customer in USD by the number of months since last purchase, per geographic profile

When observing specific data of the marketing campaign, we see that the number of customers are equally distributed across the three segments (BOGO, Discount and No Offer), and this distribution remains homogeneous given the geographic profile, as observed in Figure 4.

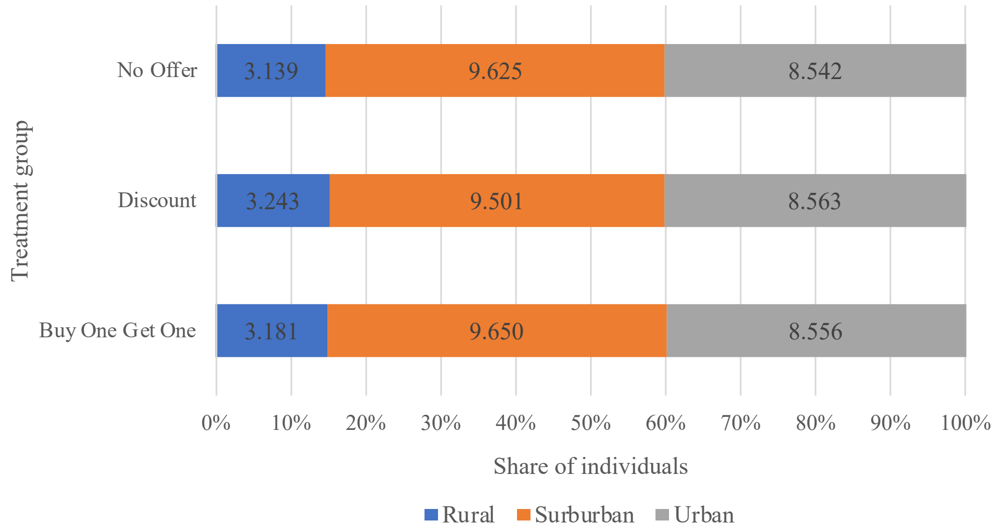

Figure 4: Population per treatment group, per geographic profile

When looking at the channel through which the marketing action is performed, we see an equivalent use of the Phone and Web, 44% each, and a lower use of the Multichannel action (12%), which indicates the use of more than one channel for the same customer. The distribution is also homogeneous by geographic profile, as shown below in Figure 5.

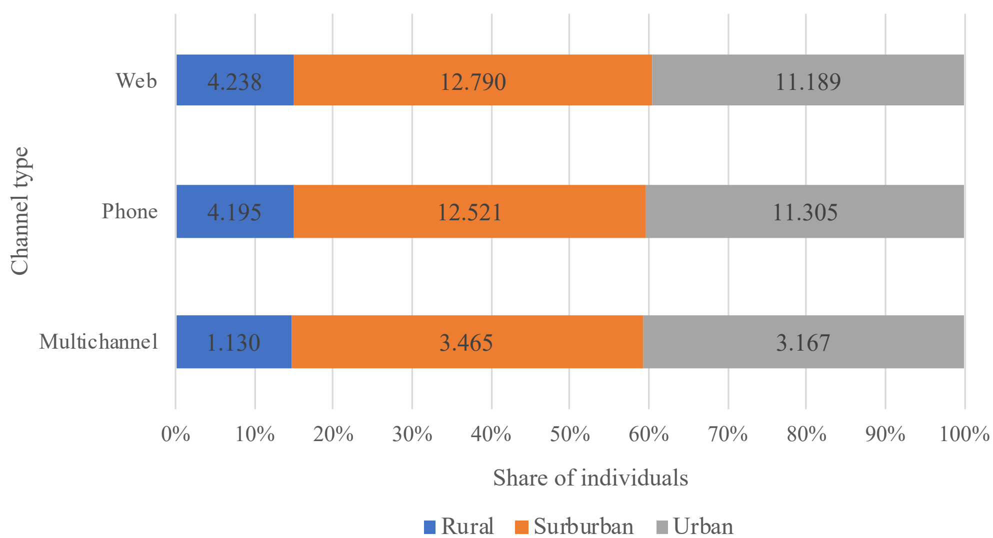

Figure 5: Population per channel, per geographic profile

Then, the conversion rate was 15% on average, but we see that the Multichannel and Web campaigns had results above the average, with 17% and 16% respectively, while the telephone channel had only a 13% conversion. The marketing campaign also resulted in a higher conversion rate for the Rural profile (19%), with the others having a conversion of 14%. This higher performance is observed across all channels, as shown in Table 4.

Geografical profile	Multichannel	Phone	Web	Total
Rural	21%	17%	21%	**19%**
Suburban	17%	12%	15%	**14%**
Urban	16%	12%	15%	**14%**
Total	**17%**	**13%**	**16%**	**15%**

Table 4: Average conversion rate by channel, per geographic profile

To conclude the descriptive analysis, Figure 6 shows a correlation matrix heatmap between the features of our dataset, as a way to explore possible relations between desired behaviour and customer characteristics. One of the strongest correlations that we can observe is between the variables *conversion* and *is referral* (-0.07), showing that if the customer was acquired by a referred the conversion rate tend to be lower (negative correlation).

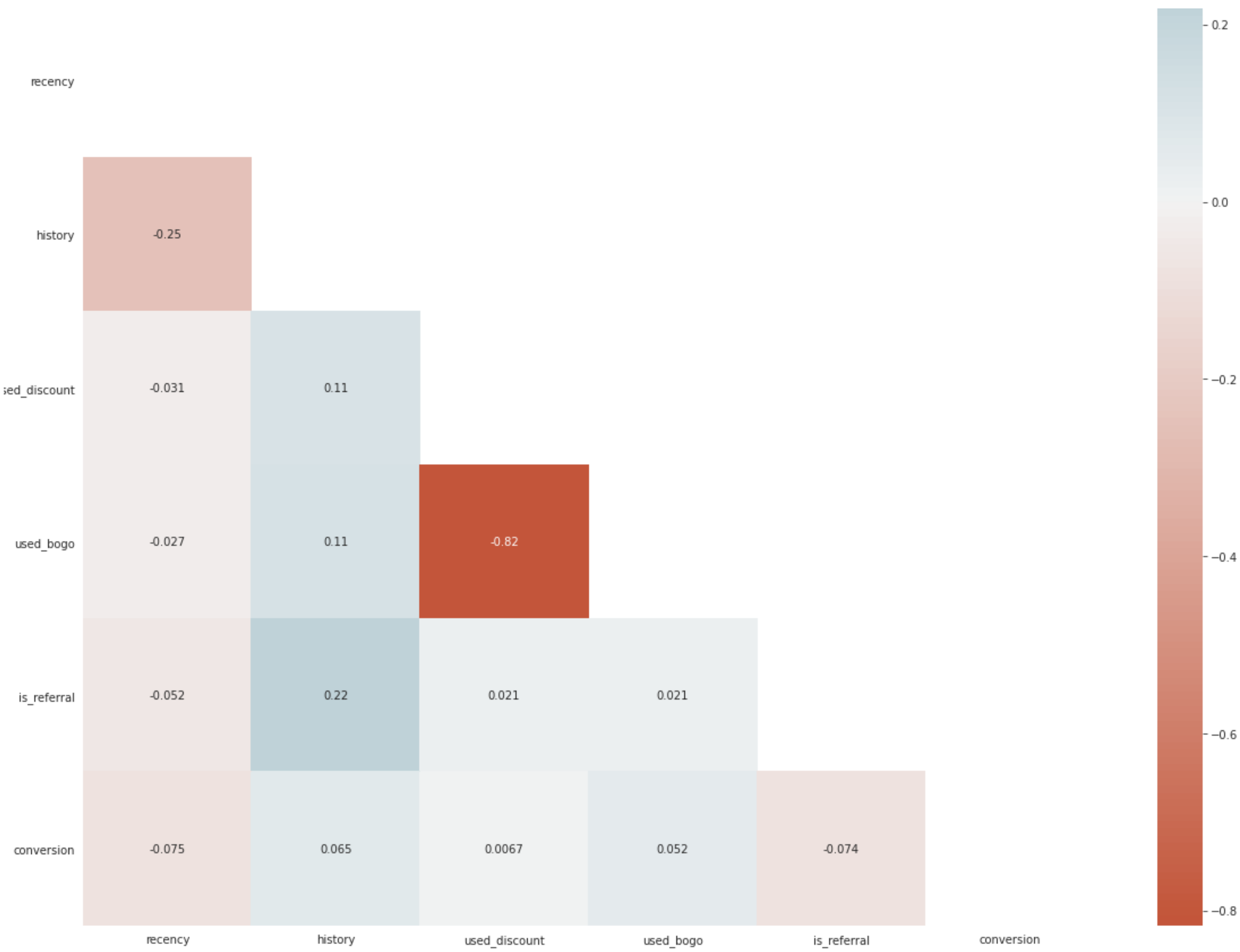

Figure 6: Correlation matrix heatmap of all features

B. *Uplift modelling*

a. *Sanity check*

Before starting the uplift modelling, although it originated from a suitable randomized trial, we performed a sanity check to verify whether in our dataset the required assumptions are satisfied. At first, we see that there is no missing data and, as mentioned in the previous section, there is an equal distribution among the three types of treatment (33%). Then, we checked that no customers in the control population were exposed to the campaign during the test.

As the desired outcome of the campaign is for customers to buy, we also check in advance whether the two treatments had a positive effect on consumer behaviour by comparing the conversion rate across the populations. The difference in conversion between control and BOGO treatment is 4.52% while between control and Discount treatment the difference reaches 7.66%. After running a Z test on the difference of proportions[11], we can conclude that both differences are statistically significant at 99% of confidence. Therefore, the effect of the campaign on the propensity of buying of the customers is positive in both types of treatments.

b. *Main results*

At first, we estimate the probability that a given customer belongs to one of the four classes previously described, using the LGWUM procedure to calculate the uplift score. As shown in Figure 7 the distribution of uplift is mostly positive across both campaigns, which makes sense since we know from our descriptive analysis that the treatment has a higher conversion rate on average. However, within the population treated with BOGO, there are some observations with negative uplift meaning the campaign discourages individuals from purchasing, therefore they are *sleeping dogs*.

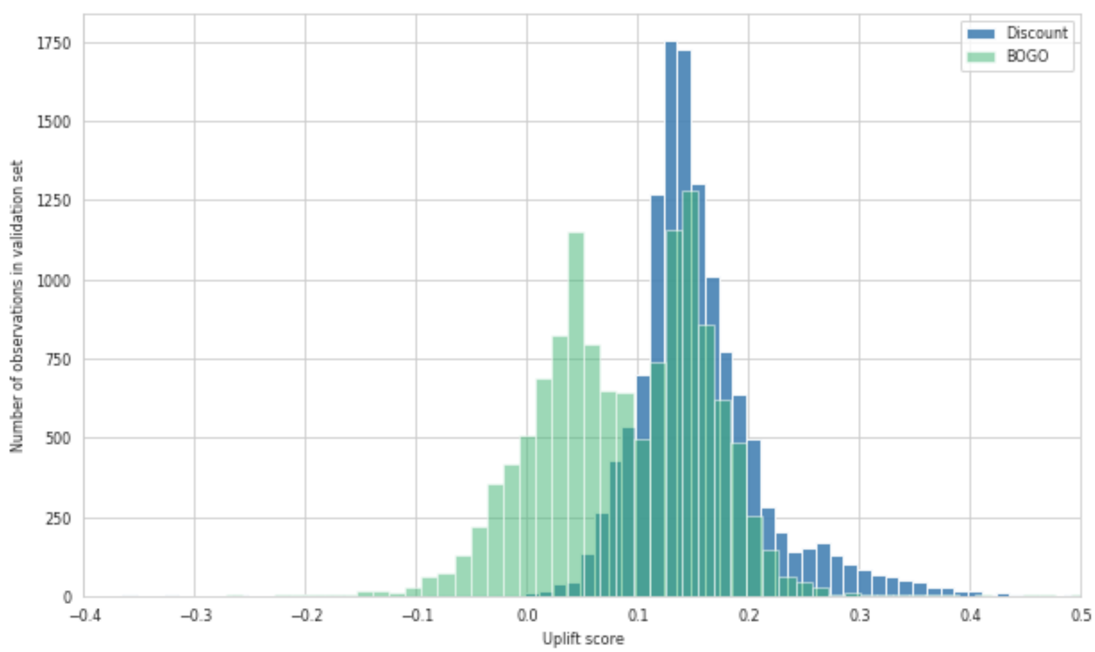

Figure 7: Comparison of uplift score distribution across BOGO and Discount models

[11] A Z-test for two independent proportions examines if some event occurs equally often in two subpopulations. The null hypothesis for a Z-test for independent proportions is that the difference between two population proportions is zero.

Then, we compute the Qini curve as an evaluation measure that shows the cumulative number of incremental positive outcomes as a function of the number of individuals treated. As it can be seen in the separation between the normalized Qini curves and the diagonal that represents a random targeting strategy, the models distinguish customers with desired outcomes from individuals with a negative outcome. The Qini score within the discount treatment group (0.04) is 33% greater than the one in the BOGO model (0.03), suggesting that the incremental positive outcome from applying a 50% discount campaign is higher.

We can interpret the modelling results as if we offer a discount to every customer, we will increase the number of purchases by 4%. However, we can achieve the same outcome by applying the treatment to the top five deciles of customers, according to Figure 9, as they are the only customers who show an uplift above the random treatment. This is because the targeted customers within this segment are all persuadables, but then the next group of customers to be treated are sure things and lost causes because they do not contribute to a greater uplift effect. Consequently, an efficient allocation of limited resources for a marketing action should be to target a discount campaign on the top 50% for treatment, as these are the customers that we expect to be the most persuadable.

Providing that the conversion rate is higher among the non-referral users, thanks to the insights of the exploratory analysis, we carried out the same analysis for that subset of users intending to find better results in terms of uplift. However, we obtained the same Qini scores of the original analysis, so focusing marketing actions on the segment of non-referral users does not imply a higher uplift in purchases.

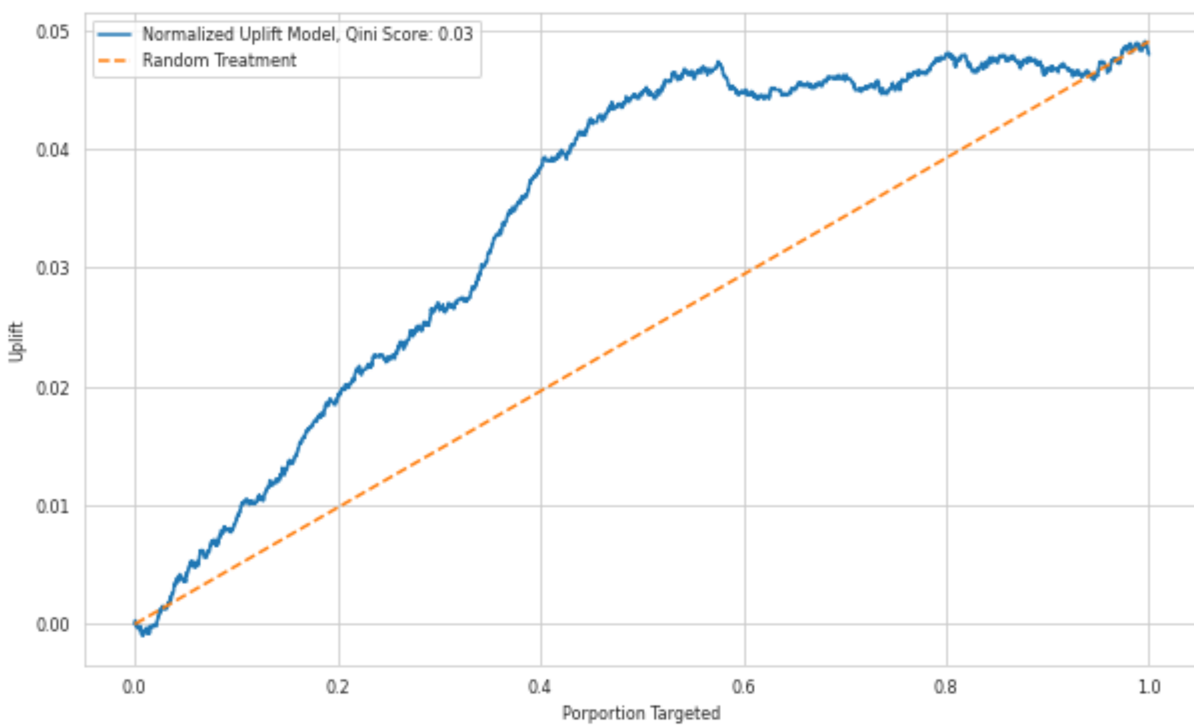

Figure 8: Normalized Qini curve for BOGO treatment

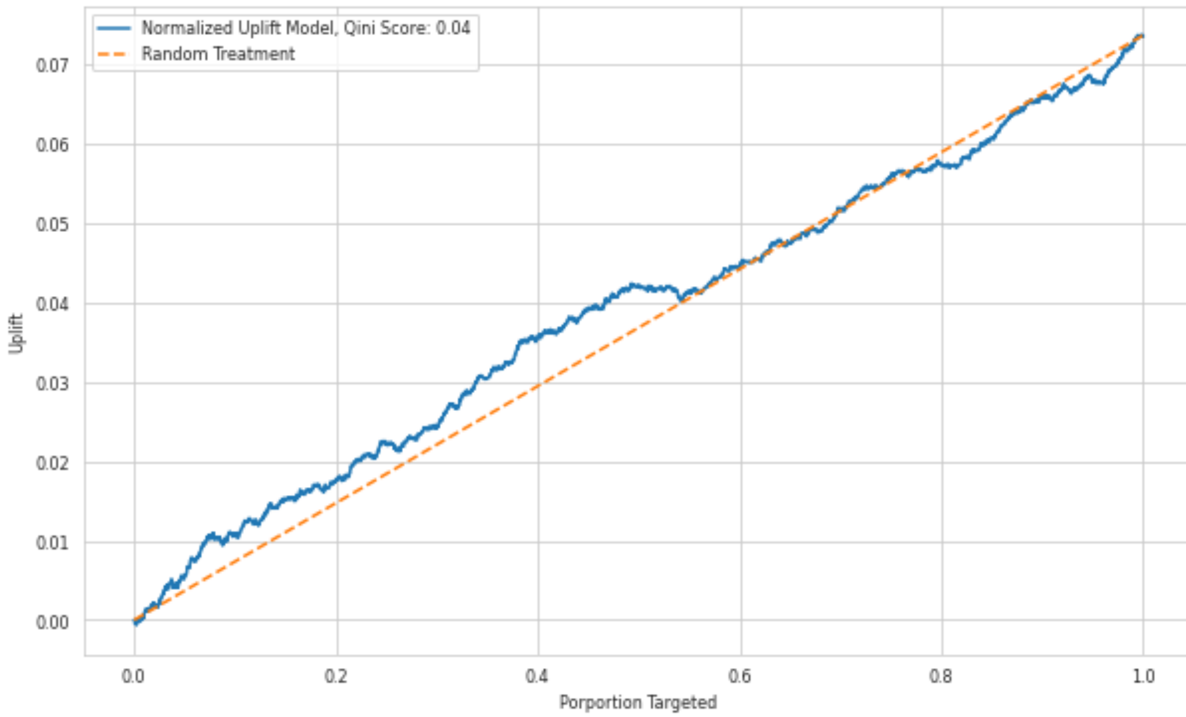

Figure 9: Normalized Qini curve for Discount treatment

III. Discussion and conclusions

Nowadays, an increasing number of firms are adopting smarter marketing actions to upsurge their competitive advantages over their competitors, relying on Machine Learning techniques to processes the great amount of data obtained from customers. However, the number of resources required to carry out such strategies are not unlimited so decision-makers should focus on increasing the efficiency of their choices, with the support of data analysis techniques that properly assess its real impact on business value. In this context, one of the main objectives for which uplift model techniques were developed is to guide marketing decisions to generate a greater impact on consumers purchase propensity, through an optimal-targeted campaign.

Based on simulated data of a marketing campaign obtained from an online community of Data Scientists, the technique was successfully applied to measure the real impact of the treatments and assess the added value of uplift modelling. As the case study shows, this technique delivers business value by decreasing the number of contacted customers, lowering marketing costs, and avoiding the adverse response that could result from customer contact not optimally targeted. The optimal proportion of customers to be targeted was determined by maximizing the purchases generated by the campaign, identifying the customers known as *persuadables*. By stating that the discount campaign is more effective, the analysis was an improvement over standard predictive analytics that introduces risk by putting each business decision to action without properly weighing the alternatives.

This approach can therefore be developed as a standard for measuring the outcome of real-life marketing actions, since it can correctly compute the difference in the expected result between control and treatment population in an experiment, such as an A/B test. For this reason, uplift modelling can be seen as a technique that measures the causal effect of a marketing incentive on customer behaviour. Besides, the procedure involves Machine Learning algorithms with a low processing consumption that can be applied using already known data analysis tools. Such advantages would facilitate the adoption of the technique by companies across industries, regardless of its size. Moreover, the relatively easy compression of the results that this approach allows implies that decision makers could base their actions on more accurate and reliable data-analysis.

The main limitation of our study is that we applied uplift modelling using a simulated dataset, therefore it is important to extend the analysis to a real-world application. As this technique may behave differently when processing different feature sets, future research should continue studying the behaviour of uplift modelling in other applications out of the marketing context. However, using data from real cases requires taking a series of precautions when performing the analysis, i.e., check if the employed data set does not originate from a proper randomized trial. Moreover, spillover effects from simultaneous actions may affect the customer behaviour that we observe in the data.

Finally, it might also be interesting to include in the study other performance assessment methods for uplift, such as dividing the dataset into more accurate test and control samples or implementing an evaluation of the campaign by introducing the Maximum Profit Uplift measure presented in Devriendt et al. (2021).

References

Ambler, Tim, and Flora Kokkinaki. “Measures of Marketing Success.” *Journal of Marketing Management*, vol. 13, no. 7, Oct. 1997, pp. 665–78, doi:10.1080/0267257X.1997.9964503.

Anas Kobbi. *Uplift Modelling and Performance Criteria Assessment*. Université de Montréal, 2018.

Asamoah, Emmanuel Selase, and Miloslava Chovancová. “An Overview of the Theory of Microeconomics (Consumer Behaviour and Market Structures) in Fast Food Marketing.” *Ekonomika a Management*, vol. 2011, no. 1, 2011, https://ideas.repec.org/a/prg/jnleam/v2011y2011i1id127.html.

Baden-Fuller, Charles, and David J. Teece. “Generating and Leveraging Customer Data Assets: Solutions Business Models.” *Academy of Management Proceedings*, vol. 2020, no. 1, July 2020, p. 12248, doi:10.5465/AMBPP.2020.12248abstract.

Barajas, Joel, et al. “Advertising Incrementality Measurement Using Controlled Geo-Experiments: The Universal App Campaign Case Study.” *ResearchGate*, 2020, https://www.researchgate.net/publication/344372807_Advertising_Incrementality_Measurement_using_Controlled_Geo-Experiments_The_Universal_App_Campaign_Case_Study.

Chapter 5 Story Uplift Modelling: EXplainable Predictions for Optimized Marketing Campaigns | XAI Stories. https://pbiecek.github.io/xai_stories/story-uplift-marketing1.html. Accessed 24 Mar. 2021.

Devriendt, Floris, et al. “Why You Should Stop Predicting Customer Churn and Start Using Uplift Models.” *Information Sciences*, vol. 548, 2021, pp. 497–515, doi:10.1016/j.ins.2019.12.075.

Diemert, Eustache, et al. “A Large Scale Benchmark for Uplift Modelling.” *KDD*, 2018, doi:10.1145/nnnnnnn.nnnnnnn.

Early, Seth. “The Role of a Customer Data Platform | Earley Information Science.” *IT Professional*, 2018, https://www.earley.com/knowledge/articles/role-customer-data-platform.

Gubela, Robin, et al. “Conversion Uplift in E-Commerce: A Systematic Benchmark of Modelling Strategies.” *International Journal of Information Technology & Decision Making*, vol. 18, no. 03, 2019, pp. 747–91, doi:10.1142/S0219622019500172.

Gubela, Robin M., et al. “Response Transformation and Profit Decomposition for Revenue Uplift Modelling.” *European Journal of Operational Research*, vol. 283, no. 2, 2020, pp. 647–61, doi:10.1016/j.ejor.2019.11.030.

Gutierrez, Pierre, and Jean-Yves Gérardy. “Causal Inference and Uplift Modelling: A Review of the Literature.” *International Conference on Predictive Applications and APIs*, PMLR, 2017, pp. 1–13, http://proceedings.mlr.press/v67/gutierrez17a.html.

“Investopedia.” *Investopedia*, https://www.investopedia.com/. Accessed 20 Mar. 2021.

Marketing Promotion Campaign Uplift Modelling. Kaggle. https://www.kaggle.com/davinwijaya/customer-retention. Accessed 6 Mar. 2021.

Montserrat Guillen, et al. *Customer Retention and Price Elasticity Motor Insurance Policies Homogeneous with Respect to Loyalty?* http://www.actuaries.org/ASTIN/Colloquia/Hague/Presentations/Guillen.pdf. University of Barcelona.

Nicholas J. Radcliffe. *Hillstrom's MineThatData Email Analytics Challenge: An Approach Using Uplift Modelling*. Stochastic Solutions Limited, 2008, http://minethatdata.com/Stochastic_Solutions_E-Mail_Challenge_2008.04.30.pdf.

Olaya, Diego, et al. “A Survey and Benchmarking Study of Multitreatment Uplift Modelling.” *Data Mining and Knowledge Discovery*, vol. 34, no. 2, Mar. 2020, pp. 273–308, doi:10.1007/s10618-019-00670-y.

Siegel, Eric. “Uplift Modelling: Predictive Analytics Can’t Optimize Marketing Decisions Without It.” *Prediction Impact*, 2011, http://www.predictiveanalyticsworld.com/machinelearningtimes/uplift-modelling-predictive-analytics-cant-optimize-marketing-decisions-without-it/5372/.

The Heartbeat of Modern Marketing: Data Activation and Personalization | McKinsey. https://www.mckinsey.com/business-functions/marketing-and-sales/our-insights/the-heartbeat-of-modern-marketing#. Accessed 24 Mar. 2021.

“Uplift Modelling in Display Remarketing.” *Wayfair*, 2 May 2018, https://tech.wayfair.com/data-science/2018/05/uplift-modelling-in-display-remarketing/.

Wikipedia, the Free Encyclopedia. https://en.wikipedia.org/. Accessed 21 Mar. 2021.

APPENDIX

```
# **Setup**
"""

# Import libraries
import numpy as np, matplotlib as mpl, matplotlib.pyplot as plt, pandas as pd
import seaborn as sns, math, os, warnings
warnings.filterwarnings('ignore')
from statsmodels.stats.proportion import proportions_ztest
from sklearn.metrics import auc
import xgboost as xgb
import sklearn as sklearn

# Get file's directory
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# Import data
df_data = pd.read_csv('/content/data.csv')
df_model = df_data.copy()

"""# **Data description**"""

# Let's take a look at our dataset
df_model.head(5)

# Total number of samples
print('Total number of samples: {}'.format(len(df_data)))

# Checking for null data
df_model.info()

# Checking for object data
df_model.describe(include=np.object)

# Checking unique object data
object_cols = [col for col in df_model.columns if df_model[col].dtype == "object"]
for obj in object_cols:
    print('\n', obj)
    for unique in df_model[obj].unique():
        print("{} {}".format(unique,sum(df_model[obj] == unique)))

# Distribution graphs (histogram/bar graph) of column data
def plotPerColumnDistribution(df, nGraphShown, nGraphPerRow):
    nunique = df.nunique()
    df = df[[col for col in df if nunique[col] > 1 and nunique[col] < 50]] # For displaying purposes, pick
columns that have between 1 and 50 unique values
    nRow, nCol = df.shape
    columnNames = list(df)
    nGraphRow = (nCol + nGraphPerRow - 1) / nGraphPerRow
    plt.figure(num = None, figsize = (6 * nGraphPerRow, 8 * nGraphRow), dpi = 80, facecolor = 'w', edgecolor =
'k')
    for i in range(min(nCol, nGraphShown)):
        plt.subplot(nGraphRow, nGraphPerRow, i + 1)
        columnDf = df.iloc[:, i]
        if (not np.issubdtype(type(columnDf.iloc[0]), np.number)):
            valueCounts = columnDf.value_counts()
            valueCounts.plot.bar()
        else:
            columnDf.hist()
        plt.ylabel('counts')
        plt.xticks(rotation = 90)
        plt.title(f'{columnNames[i]} (column {i})')
    plt.tight_layout(pad = 1.0, w_pad = 1.0, h_pad = 1.0)
    plt.show()

plotPerColumnDistribution(df_model, 10, 5)

import matplotlib.pyplot as plt
import matplotlib.style as style
import seaborn as sns

style.use('ggplot')
```

```
sns.set_style('whitegrid')
plt.subplots(figsize = (20,15))
## Plotting heatmap.

# Generate a mask for the upper triangle (taken from seaborn example gallery)
mask = np.zeros_like(df_data.corr(), dtype=np.bool)
mask[np.triu_indices_from(mask)] = True


sns.heatmap(df_data.corr(), cmap=sns.diverging_palette(20, 220, n=200), annot=True, mask=mask, center = 0, )
#plt.title("Heatmap of all the Features", fontsize = 30)
plt.yticks(rotation=0)

""" **Data transformation**"""

# Rename target column
df_model = df_model.rename(columns={'conversion': 'target'})

# Rename & Label encode treatment column
df_model = df_model.rename(columns={'offer': 'treatment'})
df_model.treatment = df_model.treatment.map({'No Offer': 0, 'Buy One Get One': -1, 'Discount': 1})

# One-Hot Encoding
df_model = pd.get_dummies(df_model)

# Share by group type
df_model['treatment'].value_counts(normalize = True)

# Split data with bogo and discount
df_model_bogo = df_model.copy().loc[df_model.treatment <=0].reset_index(drop=True)
df_model_discount = df_model.copy().loc[df_model.treatment >=0].reset_index(drop=True)

df_model_bogo.head()

df_model_discount.head()

# Share by group type (BOGO)
df_model_bogo['treatment'].value_counts(normalize = True)

# Share by group type (Discount)
df_model_discount['treatment'].value_counts(normalize = True)

# Conversion by treatment type: here we see that the marketing campaign increased the conversion
con_results_df = df_model.groupby('treatment').agg({'target':['mean', 'sum', 'count']})
con_results_df

print(f'Difference in conversion between control and BOGO treatment:
{np.round(df_model_bogo.groupby("treatment")["target"].mean()[-1]-
df_model_bogo.groupby("treatment")["target"].mean()[0], 4)}')
print(f'Difference in conversion between control and Discount treatment:
{np.round(df_model_discount.groupby("treatment")["target"].mean()[1]-
df_model_discount.groupby("treatment")["target"].mean()[0], 4)}')

#Differences are small, we run a Z test on proportions (Discount)
proportions_ztest(count=con_results_df[('target', 'sum')][1:3],
                  nobs=con_results_df[('target', 'count')][1:3])[1]

#Differences are small, we run a Z test on proportions (BOGO)
proportions_ztest(count=con_results_df[('target', 'sum')][0:2],
                  nobs=con_results_df[('target', 'count')][0:2])[1]

#Lets see if there is any effect not being referred has on the target (conversion)
df_model[df_model['is_referral']==0].groupby('treatment').agg({'target':['mean', 'sum', 'count']})

df_model[df_model['is_referral']==1].groupby('treatment').agg({'target':['mean', 'sum', 'count']})

"""The share of users that convert is higher if they were not referred"""

#Difference in conversion between control and BOGO treatment for referrals:
np.round(df_model_bogo[df_model_bogo['is_referral']==1].groupby("treatment")["target"].mean()[-1]-
df_model_bogo[df_model_bogo['is_referral']==1].groupby("treatment")["target"].mean()[0], 4)

#Difference in conversion between control and BOGO treatment for non referrals:
np.round(df_model_bogo[df_model_bogo['is_referral']==0].groupby("treatment")["target"].mean()[-1]-
df_model_bogo[df_model_bogo['is_referral']==0].groupby("treatment")["target"].mean()[0], 4)
```

```
#Difference in conversion between control and Discount treatment for referrals:
np.round(df_model_discount[df_model_discount['is_referral']==1].groupby("treatment")["target"].mean()[1]-
df_model_discount[df_model_discount['is_referral']==1].groupby("treatment")["target"].mean()[0], 4)

#Difference in conversion between control and Discount treatment for non referrals:
np.round(df_model_discount[df_model_discount['is_referral']==0].groupby("treatment")["target"].mean()[1]-
df_model_discount[df_model_discount['is_referral']==0].groupby("treatment")["target"].mean()[0], 4)

"""# **Uplift Modeling**

# **1. Class Variable Transformation**

**Target Class Declaration**

Control Non-Responders(CN)

Customers that don't make a purchase without an offer (value = 0)
Control Responders(CR)

Customers that make a purchase without an offer (value = 1)
Treatment Non-Responders(TN)

Customer that don't make a purchase and receive an offer (value = 2)
Treatment Responders(TR)

Customers that make a purchase and receive an offer (value = 3)
"""

# Function to declare Target Class
def declare_tc(df:pd.DataFrame):
    """Declare target class
    """
    #CN:
    df['target_class'] = 0
    #CR:
    df.loc[(df.treatment == 0) & (df.target != 0),'target_class'] = 1
    #TN:
    df.loc[(df.treatment != 0) & (df.target == 0),'target_class'] = 2
    #TR:
    df.loc[(df.treatment != 0) & (df.target != 0),'target_class'] = 3
    return df

# run the functions for each treatment (BOGO and Discount)
df_model_bogo = declare_tc(df_model_bogo)
df_model_discount = declare_tc(df_model_discount)

# Functions for Uplift
from sklearn.model_selection import train_test_split
import xgboost as xgb
def uplift_split(df_model:pd.DataFrame):
    """Train-Test Split
    """
    X = df_model.drop(['target','target_class'],axis=1)
    y = df_model.target_class
    X_train, X_test, \
    y_train, y_test  = train_test_split(X,
                                       y,
                                       test_size=0.3,
                                       random_state=42,
                                       stratify=df_model['treatment'])
    return X_train,X_test, y_train, y_test


def uplift_model(X_train:pd.DataFrame,
                 X_test:pd.DataFrame,
                 y_train:pd.DataFrame,
                 y_test:pd.DataFrame):
    """Using XGB to get the uplift score
    """
    # Create new dataframe
    result = pd.DataFrame(X_test).copy()

    # Fit the model
    uplift_model \
    = xgb.XGBClassifier().fit(X_train.drop('treatment', axis=1), y_train)
```

```
    # Predict using test-data
    uplift_proba \
    = uplift_model.predict_proba(X_test.drop('treatment', axis=1))
    result['proba_CN'] = uplift_proba[:,0]
    result['proba_CR'] = uplift_proba[:,1]
    result['proba_TN'] = uplift_proba[:,2]
    result['proba_TR'] = uplift_proba[:,3]
    result['uplift_score'] = result.eval('\
    proba_CN/(proba_CN+proba_CR) \
    + proba_TR/(proba_TN+proba_TR) \
    - proba_TN/(proba_TN+proba_TR) \
    - proba_CR/(proba_CN+proba_CR)')
    # Put the result
    result['target_class'] = y_test
    return result


def uplift(df_model:pd.DataFrame):
    """Combine the split and Modeling function
    """
    X_train, X_test, y_train, y_test = uplift_split(df_model)
    result = uplift_model(X_train, X_test, y_train, y_test)
    return result

# Run the uplift function
bogo_uplift = uplift(df_model_bogo)
discount_uplift = uplift(df_model_discount)

"""*BOGO*"""

bogo_uplift.head()

plt.figure(figsize = (10,6))
plt.xlim(-.4, .5)
plt.hist(bogo_uplift.uplift_score, bins=100, color=['#2077B4'])
plt.xlabel('Uplift score')
plt.ylabel('Number of observations in BOGO validation set')

"""*Discount*"""

discount_uplift.head()

discount_uplift[discount_uplift.uplift_score == discount_uplift.uplift_score.max()]

plt.figure(figsize = (10,6))
plt.xlim(-.4, .5)
plt.hist(discount_uplift.uplift_score, bins=100, color=['#2077B4'])
plt.xlabel('Uplift score')
plt.ylabel('Number of observations in Discount validation set')

plt.figure(figsize = (10,6))
plt.xlim(-.4, .5)
plt.hist(discount_uplift.uplift_score, bins=100, alpha=0.9, color='steelblue', label='Discount')
plt.hist(bogo_uplift.uplift_score, bins=100, alpha=0.5, color='mediumseagreen', label='BOGO')
plt.legend(loc='upper right')
plt.xlabel('Uplift score')
plt.ylabel('Number of observations in validation set')
plt.show()

"""**Model evaluation**"""

def qini_rank(uplift):
    # Function to Rank the data by the uplift score
    ranked = pd.DataFrame({'ranked uplift':[], 'target_class':[]})
    ranked['target_class'] = uplift['target_class']
    ranked['uplift_score'] = uplift['uplift_score']
    ranked['ranked uplift'] = ranked.uplift_score.rank(pct=True, ascending=False)
    # Data Ranking
    ranked = ranked.sort_values(by='ranked uplift').reset_index(drop=True)
    return ranked

def qini_eval(ranked):
    uplift_model, random_model = ranked.copy(), ranked.copy()
    # Using Treatment and Control Group to calculate the uplift (Incremental gain)
    C, T = sum(ranked['target_class'] <= 1), sum(ranked['target_class'] >= 2)
```

```
    ranked['cr'] = 0
    ranked['tr'] = 0
    ranked.loc[ranked.target_class == 1,'cr'] = 1
    ranked.loc[ranked.target_class == 3,'tr'] = 1
    ranked['cr/c'] = ranked.cr.cumsum() / C
    ranked['tr/t'] = ranked.tr.cumsum() / T
    # Calculate and put the uplift and random value into dataframe
    uplift_model['uplift'] = round(ranked['tr/t'] - ranked['cr/c'],5)
    random_model['uplift'] = round(ranked['ranked uplift'] * uplift_model['uplift'].iloc[-1],5)

    uplift_model['Number_of_exposed_customers'] = np.arange(len(uplift_model))+1
    uplift_model['conversions_gained'] = uplift_model.uplift*len(uplift_model)

    # Add q0
    q0 = pd.DataFrame({'ranked uplift':0, 'uplift':0, 'target_class': None}, index =[0])
    uplift_model = pd.concat([q0, uplift_model]).reset_index(drop = True)
    random_model = pd.concat([q0, random_model]).reset_index(drop = True)
    # Add model name & concat
    uplift_model['model'] = 'Uplift model'
    random_model['model'] = 'Random model'
    merged = pd.concat([uplift_model, random_model]).sort_values(by='ranked uplift').reset_index(drop = True)
    return merged, uplift_model

def uplift_curve(uplift_model):
    plt.figure(figsize = (10,6))
    # plot the data
    ax = uplift_model['conversions_gained'].plot(color=['#2077B4'])
    # Plot settings
    sns.set_style('whitegrid')
    handles, labels = ax.get_legend_handles_labels()
    plt.xlabel('Number of customers treated')
    plt.ylabel('Incremental purchases')
    plt.grid(b=True, which='major')
    return ax

def qini_plot(merged:pd.DataFrame, uplift_model:pd.DataFrame):
    gain_x = uplift_model['ranked uplift']
    gain_y = uplift_model.uplift
    qini = auc(gain_x, gain_y)
    # plot the data
    plt.figure(figsize = (10,6))
    mpl.rcParams['font.size'] = 8
    qini = auc(gain_x, gain_y)

    ax = plt.plot(gain_x, gain_y, color= '#2077B4',
        label='Normalized Uplift Model, Qini Score: {}'.format(round(qini,2)))

    plt.plot([0, gain_x.max()], [0, gain_y.max()],
        '--', color='tab:orange',
        label='Random Treatment')
    plt.legend()
    plt.xlabel('Porportion Targeted')
    plt.ylabel('Uplift')
    plt.grid(b=True, which='major')

    return ax

def plot_uplift(result:pd.DataFrame):
    # Function to plot the uplift curve
    ranked = qini_rank(result)
    merged, uplift_model = qini_eval(ranked)
    ax1 = uplift_curve(uplift_model)

    return ax1

def plot_qini(result:pd.DataFrame):
    # Function to plot the qini curve
    ranked = qini_rank(result)
    merged, uplift_model = qini_eval(ranked)
    ax2 = qini_plot(merged, uplift_model)

    return ax2

plot_uplift(bogo_uplift)

plot_qini(bogo_uplift)
```

```
plot_uplift(discount_uplift)

plot_qini(discount_uplift)

"""As the Qini score is low for both type of campaigns, we will see if positive outcomes is much larger
providing users are not refferal.
Therefore, we have to take a subset with the not referred users.

**Subset of Not referred users**
"""

df_model_bogo_2 = df_model_bogo.loc[(df_model_bogo.is_referral == 0)]

df_model_discount_2 = df_model_discount.loc[(df_model_discount.is_referral == 0)]

# check
df_model_bogo_2['is_referral'].value_counts(normalize = True)

df_model_discount_2['is_referral'].value_counts(normalize = True)

# run the target functions for each treatment (BOGO and Discount)

df_model_bogo_2 = declare_tc(df_model_bogo_2)
df_model_discount_2 = declare_tc(df_model_discount_2)

# Run the uplift function
bogo_uplift_2 = uplift(df_model_bogo_2)
discount_uplift_2 = uplift(df_model_discount_2)

plt.figure(figsize = (10,6))
plt.xlim(-.4, .5)
plt.hist(bogo_uplift_2.uplift_score, bins=100, color=['#2077B4'])
plt.xlabel('Uplift score')
plt.ylabel('Number of observations in BOGO validation set')

plt.figure(figsize = (10,6))
plt.xlim(-.4, .5)
plt.hist(discount_uplift_2.uplift_score, bins=100, alpha=0.9, color='steelblue', label='Discount')
plt.hist(bogo_uplift_2.uplift_score, bins=100, alpha=0.5, color='mediumseagreen', label='BOGO')
plt.legend(loc='upper right')
plt.xlabel('Uplift score')
plt.ylabel('Number of observations in validation set')
plt.show()

plot_uplift(bogo_uplift_2)

plot_qini(bogo_uplift_2)

plot_uplift(discount_uplift_2)

plot_qini(discount_uplift_2)
```