Multi-Armed Bandits for Coupom Recommendation at the Starbucks Mobile Rewards App.

Joao F. G. Silva[1]

[1] Electrical Engineering Program – Federal University of Rio de Janeiro (UFRJ)
Udacity Machine Learning Engineer Nanodegree
Rio de Janeiro, Brazil.

guedes.joaofelipe@poli.ufrj.br

Abstract. *reward systems have been widely used to enhance customers' engagement in digital-based platforms. By offering these users a challenge and a correspondent reward, they not only can be attracted to have more interactions with a company's service, but most importantly it can lead them into becoming frequent users, thus enhancing a brand's impact on its customers. However, knowing which challenge to provide can be rather complex task since each customer profile responds differently to each offer. In order to overcome this problem, this project proposes a recommendation system that uses historical rewards usage to build a data-driven users' profiles so as to model the most suitable offer type to each profile.*

1. Introduction

In order to enhance customers' engagement, companies make use of coupons that are sent out through multiple marketing channels. This strategy makes customers not only attracted to a service, but also may establish a long-term client relationship, increasing the probability of turning random customers into brand advocates.

Nevertheless, due to the many aspects that rule marketing channels and customers' consumption profiles, having a proper offer strategy determines whether a certain customer is bound to take it or not. For example, younger people may be more leaned on consuming a low-difficult income sent to their mobile since they might have lower income and spend more time using their mobiles. Therefore, sending out the right offer through the right channel and to the proper customer poses a considerable challenge which greatly affects revenue.

One way to overcome these challenges is to apply analytical and data modeling techniques so that companies map how likely is a certain consumer to take an offer. By developing data-driven customers' profiles and recommender systems, one can suggest a personalized offer to a certain profile in a way that consumption probability is maximized.

In that sense, this project aims at creating such data-driven profiles and building a recommender system to suggest which offer to send to customer or possible lead. In particular, a dataset provided from Starbucks' reward mobile app is used to train a clustering model that creates the profiles and train a recommender algorithm to decide which offer to send. We analyze the proposed workflow in terms of two usage-prediction figures of merit: $precision@k$ and $recall@k$.

2. Theoretical Background

In this section, a brief overview of the proposed recommendation engine, a multi-armed bandit, is described. It was considered that the K-Means [MacQueen 1967] and the Singular Value Decomposition (SVD) [Golub and Reinsch 1970] algorithms have been thorougly covered in the literature, and thus shall not be explained in this document.

Figure 1. Representation of a multi-armed bandit: an agent has multiple arms to pull and each of them have a probability of giving a reward.

A MAB algorithm is inspired in the slot-machine problem, where an agent (the bandit) could repeatedly pull an arm out of many options, possibly gaining a reward from it. As the bandit interacts with the slot-machines, he notices that one of them seems to be giving more reward. Thus, he is tempted to exploit that one lever. However, other machines may give even more rewards. In this case, he needs to decide whether to exploit a single machine or to explore others [Sutton and Barto 2018].

In its simplest mathematical formulation, a MAB consists of k machines with their own probability distribution $(p_1, \dots, p_k)$, expected rewards $(\mu_1, \dots, \mu_k)$ and variances $(\sigma_1^2, \dots, \sigma_k^2)$ - all of which are initially unknown to the agent. At each epoch $t = 1, \dots T$, an arm a_i is pulled and a reward is received. The bandit should then choose which arm to pull next: the one who has so-far given the highest payoff or another arm which possibly leads to better payoff. The highest payoff in the T round is given by

$$R_T = T\mu^* - \sum_{t=1}^{T} \mu_i(t) \tag{1}$$

where $\mu^* = max_i \mu_i$ is the expected reward from the best arm. In other words, he needs to decide between the exploitation-exploration trade-off.

Several techniques have been employed to solve this problem, one of them being the ϵ-greedy approach [Głowacka 2019]. In this classical approach, each round the bandit chooses the arm with the highest empirical mean with probability $1 - \epsilon$, or selects a random arm with probability ϵ.

As it can be inferred, the ϵ parameter has a tight influence on the exploitation-exploration trade-off. For higher ϵ, the greedy action of choosing the highest empirical mean arm has lower probability of being chosen, thus leading the bandit to explore more options. Conversely, with lower ϵ, the algorithm tends to choose the greedy action.

The ϵ-greedy is the basis of a myriad of algorithms. In order to have more exploration in the initial rounds and more exploitation in the later, a round-varying ϵ can be applied with the Decay ϵ-greedy approach. In this case, the ϵ at round n is defined as in Equation 2

$$\epsilon = \frac{1}{1 + n\beta} \tag{2}$$

where β controls how fast ϵ reduces and n is the current round. Furthermore, a threshold ϵ can be defined so as to limit a minimum exploration at the later rounds.

MAB algorithms have been used in many applications where sequential decisions need to be taken, like recommender systems. In this case, the MAB is used to model a consumption profile. Considering that users (agents) are provided several items (slot-machines) and he may consume it according to a reward probability distribution function. For the current Starbucks reward project, they shall be used to model a customers' profile considering each arm as an offer and a customized reward function defined to grasp both obtained coupon-rewards and marketing achievements, as shall be seen later.

3. Analysis

In this section, the aforementioned Starbucks dataset shall be analyzed in more depth, so that some of its statistics are displayed.

The dataset contains a sample of customer behavior obtained during a few weeks. Once every few days, Starbucks sends out expirable offers to users which have their own difficulty and reward amount. On the whole, 3 files are provided in the dataset, *profile.json*, *portfolio.json* and *transcript.json*, and its content are summarized in Figure 2.

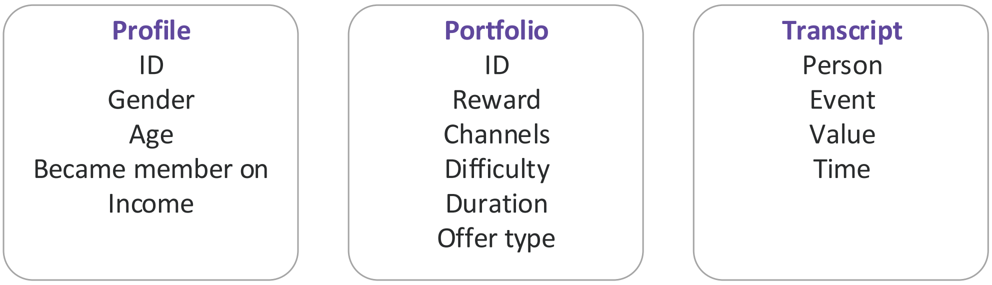

Figure 2. Content of each file on the Starbucks rewards dataset.

For the sampled data, there is a total of 17000 users' profiles and 10 offers from the portfolio. Since each offer can be sent multiple times to each user, 306534 registers are logged in the transcript as the events occur in the referred time frame. Next, some of these variables distributions shall be displayed together with their applied preprocessing.

Figure 3 shows the processed income distribution and the dashed line shows the mean average. In the given dataset, a total of 2175 customers do not have their income information filled. Seemingly, it could be a good strategy to fill in these values with the mean of the existing entries. However, in many real world scenarios, an income variable follows a long-tail distribution. This poses a elaborate task whenever null entries are found in this variable.

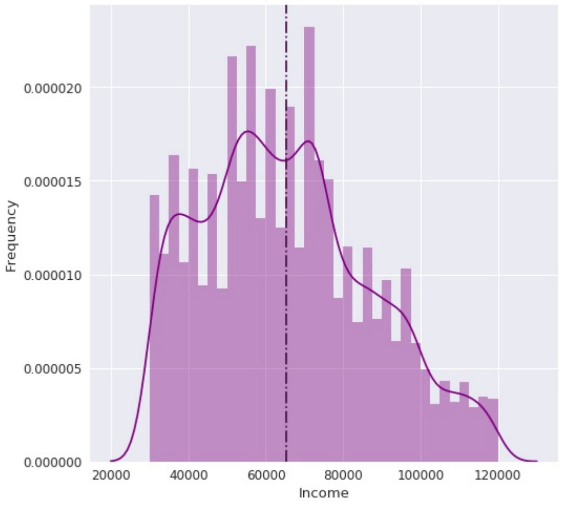

Figure 3. Income distribution.

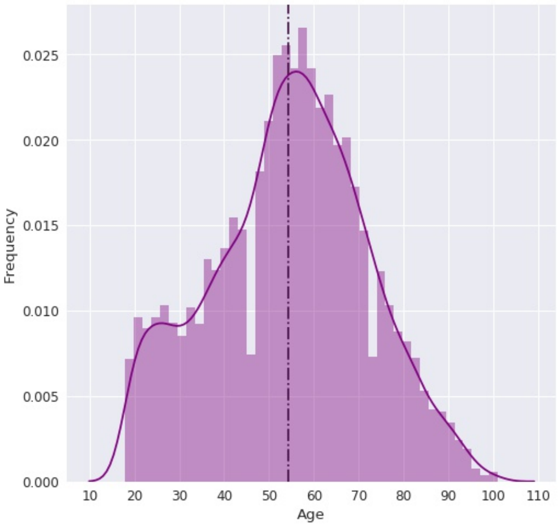

Figure 4. Age distribution.

Even though the empirical distribution does not seem to follow a proper long-tail distribution, it shall be considered that this is a result of the used sampling process to generate the dataset. Therefore, we shall not fill the 2175 missing value entries with any average. These rows are then dropped from the dataset, resulting in the distribution from Figure 3.

As opposed to income variables, an age feature can be roughly modeled as a normal distribution in many real-world scenarios. Accordingly, this is the case in this dataset. Consequently, all of its 118 missing values were replaced by the mean average of the existing values. After this treatment, the age distribution is displayed in Figure 4.

It is interesting to note that the mean age of customers is 54 years old, which is quite high. For most countries (the United States included, which is probably where the data was gathered), the average age is 38 years old. One of the causes for this high average might be the fact that Starbucks mostly sells coffee, which is hardly a product consumed by children.

The gender attribute has 3 original categories: “F” (feminine), “M” (masculine) and “O” (others). Out of all profiles, 2175 have null gender entries, which were later mapped as ”O” since they can be semantically viewed as not necessarily masculine or feminine. The resulting number of registers for each category is shown in Figure 5.

Finally, an evolution of the number of member subscriptions can be evaluated in Figure 6. It can be seen that there has been a significant rise in the volume of active customers from 2014 to 2019.

Some analytics can be also applied to the marketing-related features. By combining the portfolio information with the historical sent offers, the effectiveness of each marketing channel can be extracted considering the depth to which a customer enters a marketing funnel. Namely, in [Kotler et al. 2016] the author proposes the 5 A’s of digital marketing which relates to each step of the funnel: awareness, appeal, ask, act and advocate.

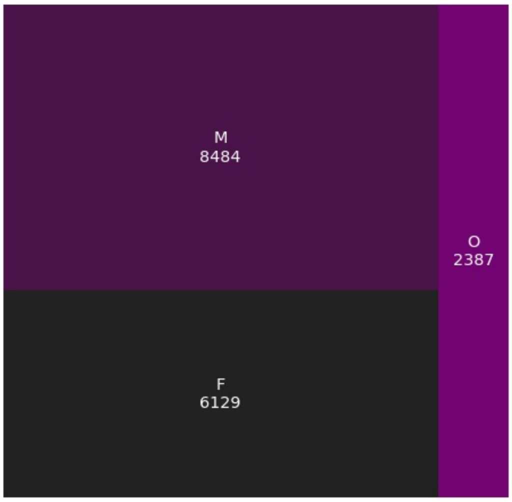

Figure 5. Number of profiles for each gender.

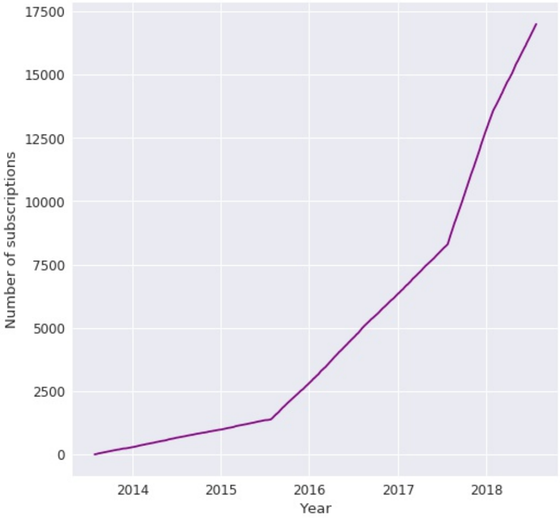

Figure 6. Evolution of subscription.

Although not all of these steps are mapped in the transcript dataset, some landmarks may be used to grasp how faw a customer has gone in the funnel. In particular, the following proxies can be made:

- Aware - Offer View
- Act - Purchase Completed
- Advocate - Future Purchases

So, these landmarks were obtained and its visual flow is depicted in Figure 7.

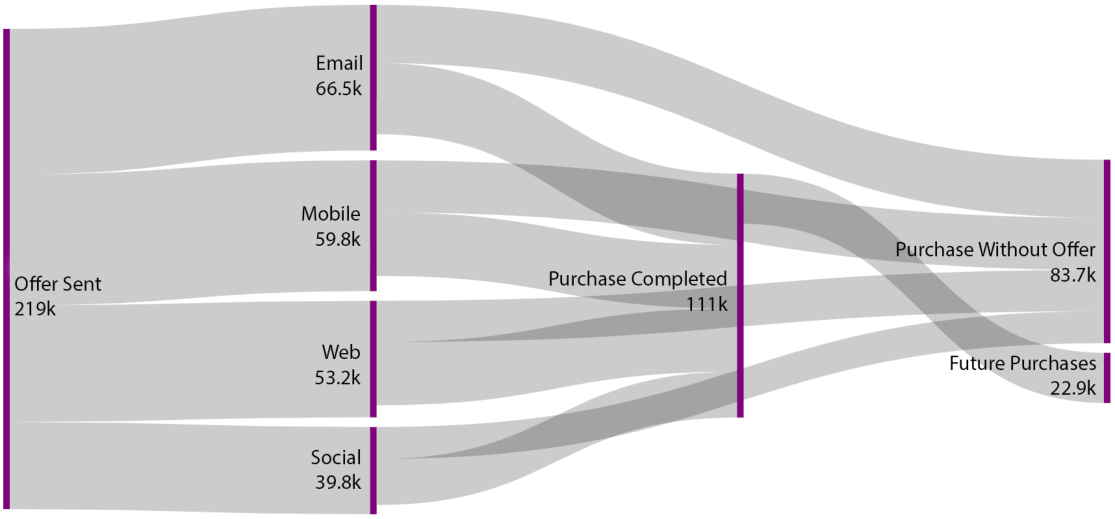

Figure 7. Marketing funnel of the rewards mobile app.

After a offer is sent, we are interested in analyzing according to each marketing channel which offers were viewed, which turned into purchase and which led to future purchases. It is worth saying that an offer may be sent through multiple channels, but the effectiveness will be treated separately.

By analyzing the obtained flow, we see that all $219k$ sent offers are viewed and most of them are sent by email ($66.5k$) or mobile channels ($59.8k$). Having a high ratio of viewed offers brings a positive characteristic of the rewards app which is making customers aware of the brand and their proposed offers. When these proposals turn into the act of purchase, it can be inferred that customers are not only aware of the brand's service but are also attracted to it.

However, a purchase may be made with or without an offer. In other words, a customer may buy a product from the portfolio yet not using the offered reward. In this particular dataset, a total of $83.7k$ purchases were made in this scenario. This indicates that even though the proper marketing channel was used, the reward offer was not attractive to the user. This, however, is a concern for the product's characteristics and not the marketing strategy. Perhaps by changing the sent offer using the same channel, the customer would have taken the offer and receiving a reward for it.

At the end of the funnel, one the most important tasks a offer proposal can achieve is to lead the customer for a future purchase, which is a proxy of customer advocacy. When customers tend to buy repeatedly from a company, he/she is likely to advocate for their peers, thus generating even more awareness and possibly attraction of the brand in the market. This is a noticeable drawback of the mobile rewards app, given that only $22.9k$ from the $111k$ offers led to future purchases.

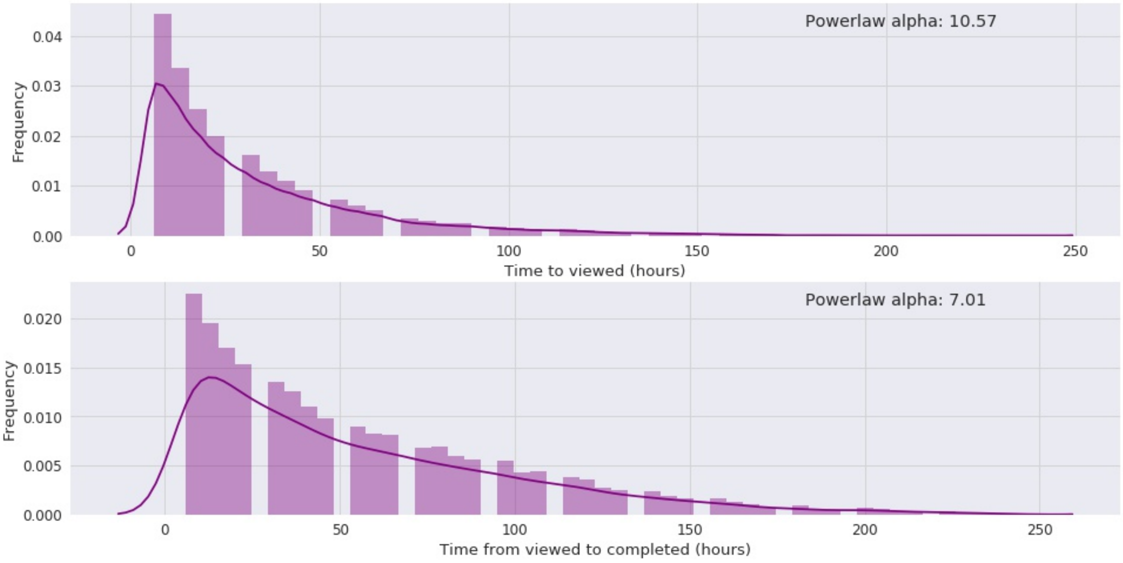

Figure 8. Elapsed time in the marketing funnel.

Finally, after an offer is sent, the time it takes for users to engage in the various steps of the funnel is highly correlated to a lead status. A lead is a potential buyer whose status can be either cold, warm or qualified. As soon as an user views an offer, he may be viewed as a potential lead considering that he is interested in it. For example, for email offers, when a customer opens the email he might be attracted to its head content. In this case, the may be considered a warm lead.

However, if he/she does not continue to engage the offer, he starts becoming colder and colder as time goes by, meaning his interest in taking the offer diminishes over time.

So ideally, users should spend a short period of time from viewing and offer and completing it. A distribution of this time frame for the provided dataset is displayed in Figure 8.

It can be seen that the time distribution (in hours) from viewing and offer to complete it follows approximately a long-tail distribution. This means that most purchases come from offers which are promptly completed. To analyze how long is the tail of the distribution, we can fit the data to a powerlaw distribution whose probability distribution function is defined according to $p(x) \propto x^{-\alpha}$ - the lower the α, the longer the tail.

Naturally, these elapsed times are highly affected by the offers' properties, such as difficulty and duration. Figure 9 shows the number of purchases for each offer. In particular, the less completed offer (ID = *0b1e***d7*) is the onewith the highest difficulty (20) and the highest duration (10 days). The most completed offers have medium difficulty and duration, and this might be due to the most appropriate trade-off between difficulty and reward amount for users.

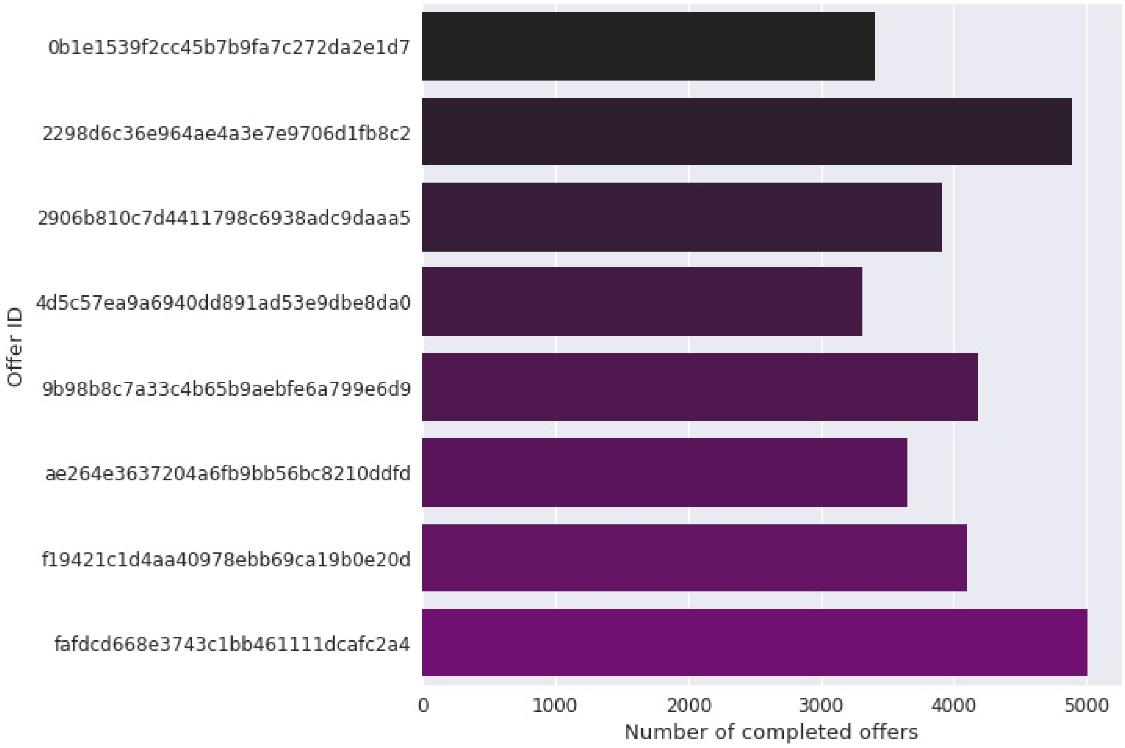

Figure 9. Most completed offers.

After raising the characteristics of the given dataset, we shall move on to establishing the modeling methodology to recommend products for customers.

4. Methodology

Since offer information is scattered in the transcript file, which is also separated from products and profiles' informations, first a preprocess step is performed to get in a single row-oriented file all information of a sent offer. In other words, for every sent offer to a person, a row is constructed with columns like:

- person id, offer id (strings)
- offer viewed, offer completed, future purchase (binary)
- offer reward (integer)

This new set facilitates the construction of a new column related to the constructed reward function to be used in the MAB algorithm. In particular, the following reward equation was used:

$$mabReward = offerViewed + offerCompleted \cdot offerReward + 2 \cdot futurePurchase$$

The heuristics used to formulate this equation follows the strategy of marketing funnel. If the customer viewed an offer, the marketing strategy gains +1 of reward. In case the offer is completed, the offer reward amount is also added. Finally, in order to account for strategies which led to future purchases (and, as aforementioned, possible advocates), +2 points are added. Naturally, this equation needs a more in-depth analysis in future works.

In addition, an utility matrix can be constructed having users as rows and products as columns. The value in each element is the mean reward each user had for a specific product. The sparse matrix is then used to train and validate the SVD algorithm, which outputs a densely filled matrix with estimates of unknown MAB rewards.

Furthermore, customers' informations can be attached to the matrix in order to train the clustering algorithm. In that case, data-driven customer profiles can be drawn considering their features and the rewards they have taken from each product. By taking the trained centroids as rewards, each cluster is finally modeled as a MAB, where each arm is a product. Recommendations, at last, are simply pulls from the models' arms.

5. Results

After preprocessing all variables and creating the column-wise dataset, the correlation between these inputs can be seen in Figure 10.

In general, completed offers have a strong and positive correlation to future purchases, which tells us that a fidelity component is present. However, they have a negative correlation with the informational offer type, which might indicate that such type is not suitable for the general customers. In fact, in terms of offer type, the one that positively correlates to other strategic variables is BOGO, which seems to be the most suitable to completed offers and, consequently, to MAB reward.

Another perspective which can be drawn from the correlation analysis is the fact that younger customers tend to complete the challenges, even though lower age correlates to lower incomes. Finally, transactions that are made without offer completion and the transaction amount have negatiev correlation. This might mean that, when consumers don't use the reward program, they tend to spend a smaller amount.

After analyzing all variables, we proceeded to analyze the number of factors to use in the SVD matrix factorization, which was supported by Figure 11.

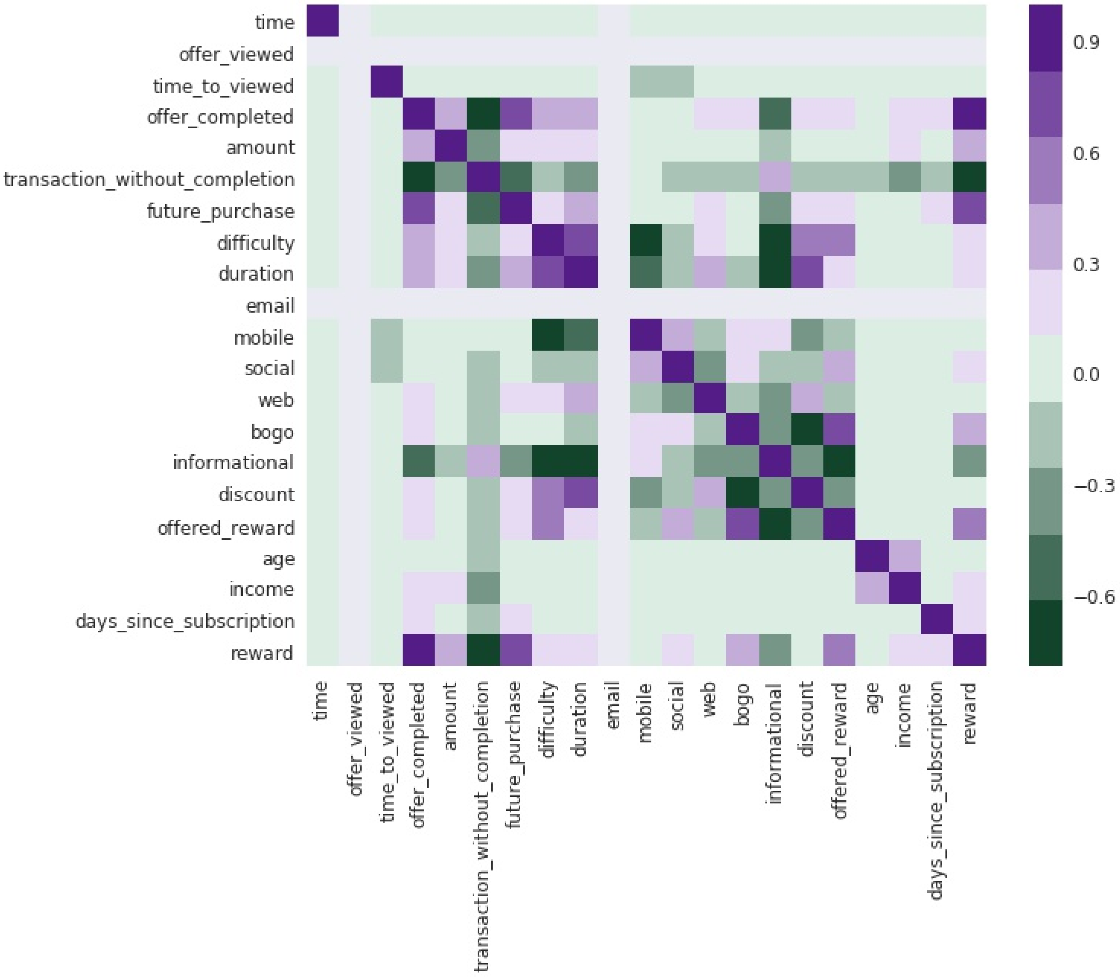

Figure 10. Offers correlation heatmap.

In this plot, the root mean squared error (RMSE) was obtained between the original MAB reward in the sparse utility matrix and its predicted value, considering known-entries. The RMSE was obtained for a 5-fold cross validation and its mean and standard deviation are shown. After analysis, it was chosen to use 100 factors for the SVD.

Given that the MAB reward is a auxiliary feedback score, a following analysis of recommendation usage was performed. For that, the precision@k and recall@k were calculated for the trained SVD model and the mean and standard deviation for the 5-fold cross validation is shown in Figures 12 and 13.

Naturally, given that only 1 coupon is sent out to customers, the best precision and recall happen when $k = 1$. This indicates that, even though the MAB reward is an auxiliary score, it can be well suited to predict which single offer is best to recommend to a certain user. In case more offers are sent, the performance of these figures of merit tend to diminish.

Since each MAB is supposed to model a single user cluster, the number of clusters to be used needs to be specified. For that purpose, the k-means algorithm was trained for multiple k and the mean test silhouette score was obtained for a 5-fold cross validation,

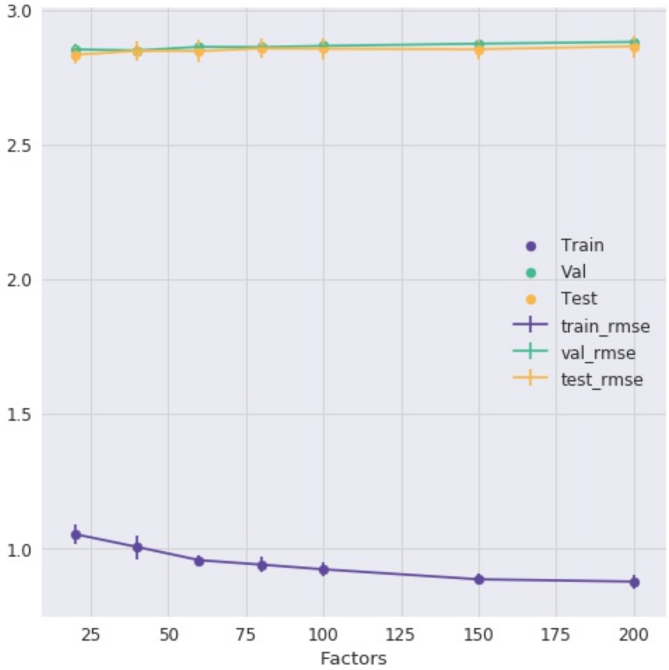

Figure 11. Mean RMSE by SVD number of factors in a 5-fold cross validation.

and the results are shown in Figure 14. After its analysis, the number of clusters to be used was chosen to be $k = 5$.

Finally, after obtaining the clusters' centroids, a MAB can be trained for each cluster. For this project, the decay ϵ-greedy approach was chosen with $\beta = 0.001$. Since its output is highly stochastic, the mean average reward was obtained for a total of 20 realizations, and its evolution throughout the epochs is portrayed in Figure 15.

From the obtained results, it can be inferred that some clusters tend to have better performance when modeled as MABs, such as cluster 6, 4 and 0. An analysis of these users' profiles showed that these clusters tend to have lower incomes and low- to medium-aged customers, which corroborates to the aforementioned findings from the offers correlation heatmap.

Given that the chosen MAB strategy only outputs a single recommendation per arm-pull, a further grasp on the portfolio usage was obtained so as to check whether a single offer was being recommended for all clusters - this would lead to specializing in a single product and, thus, reducing portfolio diversity. The number of picks for each offer in all clusters is shown in Figure 16.

As it can be seen, 3 out of 10 offers are highly picked for most clusters. This means that, in general, 30% of the portfolio is recommended. This number is highly affected by the exploration-rate after the MAB converges. When the exploration is low, the model tends to recommend only the offers which have shown to provide the highest reward. In order to have a lower exploitation and thus a more diverse usage of the portfolio, the ϵ parameter can be set to be higher. However, this may impact on the model's performance in terms of usage prediction.

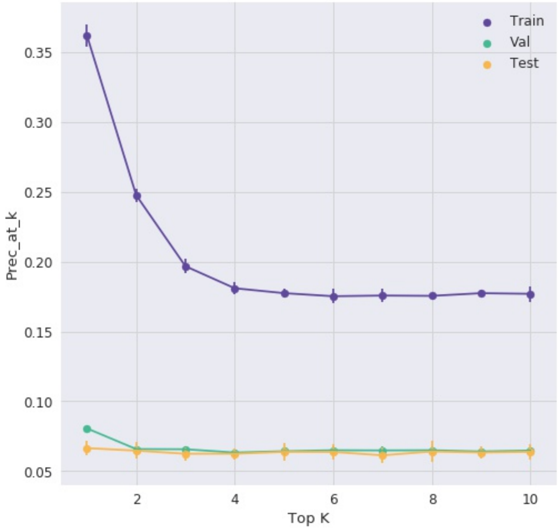

Figure 12. Precision@k for the SVD.

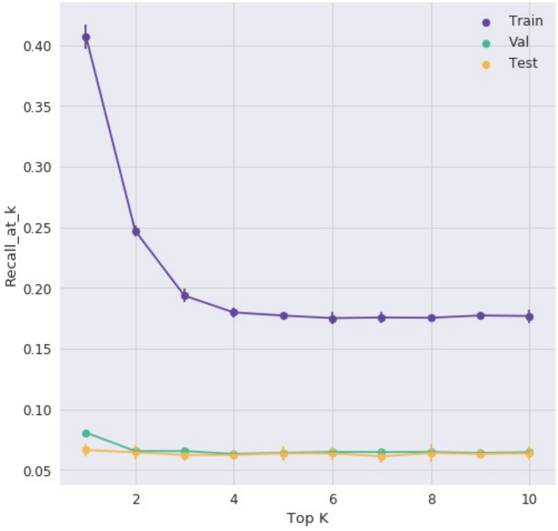

Figure 13. Recall@k for the SVD.

6. Conclusion and Future Works

The current project article described how machine learning techniques can be employed with marketing frameworks to create a proper coupon recommendation engine. In particular, a clustering algorithm was proposed to create data-driven customers profile which then could be modeled as a multi-armed bandit.

The approach of using customers' meta-data is handful when users have a single or few interactions with a platform. This approach is used to tackle the cold-start problem in recommender system, where insufficient information about users consumption profile leads to low-quality recommendation.

However, we have seen that, despite forming representative clusters, MAB algorithms have to be well designed so that exploitation-exploration trade-off is properly handled. In particular, if an algorithm exploits too much certain item, then catalog diversity is compromised, thus bringing revenue to a small fraction of available items.

All in all, different MAB approaches may be implemented in future works to understand how diverse is their recommendation. Furthermore, the design of a well-suited reward function greatly affects the results from this approach. Although in this project a somewhat linear approach was suggested as a MAB reward, considering marketing-oriented features, a deeper analysis of this reward function shall be studied in future projects.

References

Golub, G. and Reinsch, C. (1970). Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5):403–420.

Głowacka, D. (2019). *Bandit Algorithms in Information Retrieval*.

Kotler, P., Kartajaya, H., and Setiawan, I. (2016). *Marketing 4.0: Moving from Traditional to Digital*. Wiley.

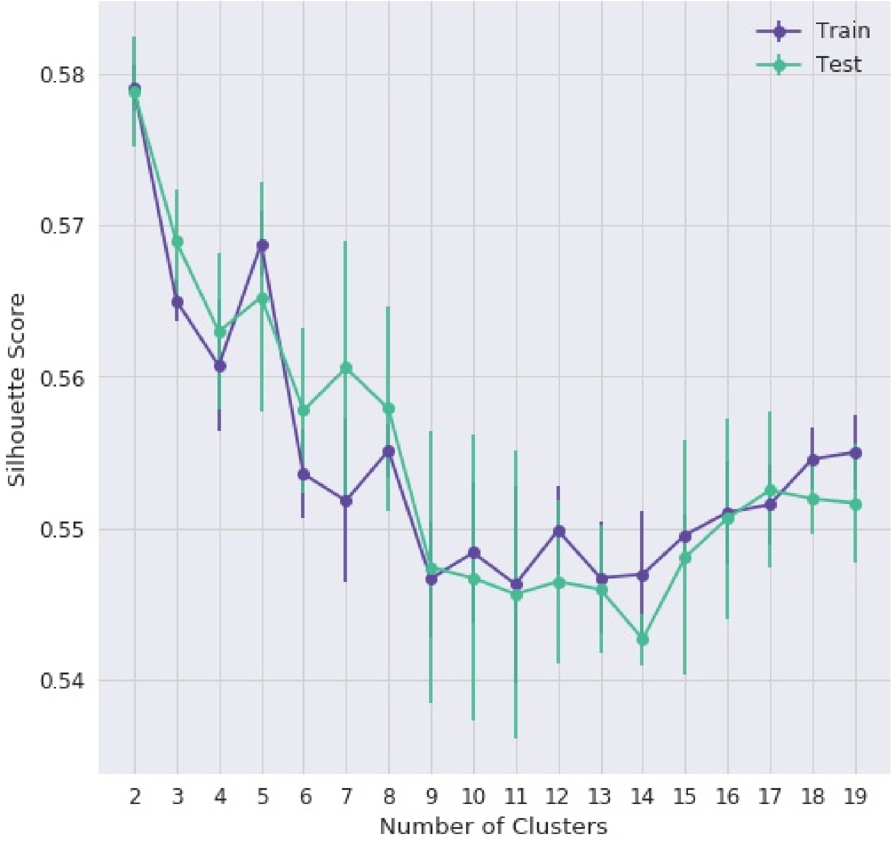

Figure 14. Clustering silhouette score.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In Cam, L. M. L. and Neyman, J., editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. The MIT Press, second edition.

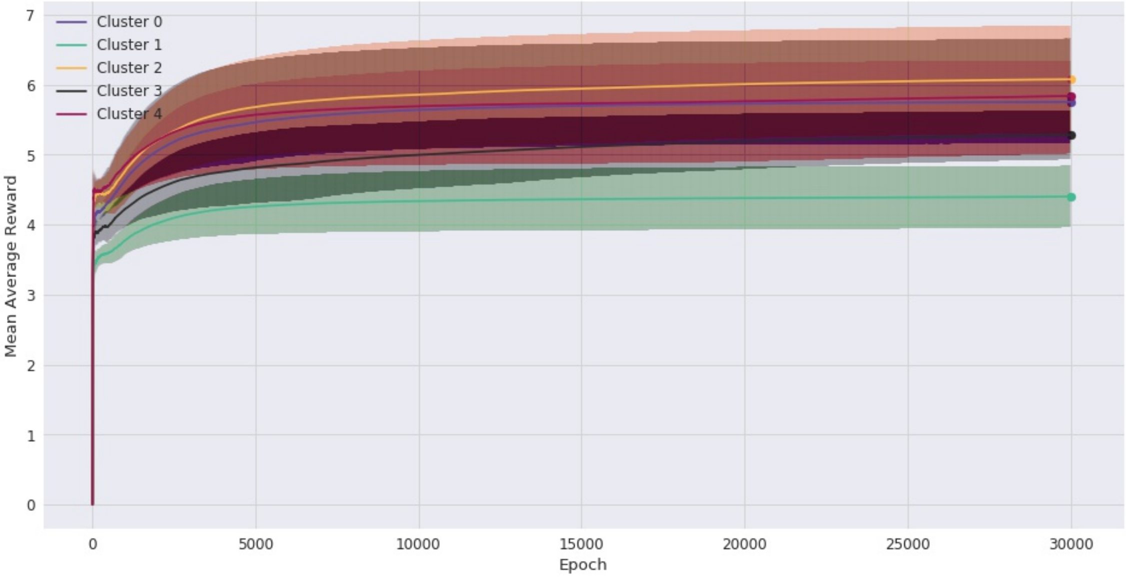

Figure 15. MAB mean average reward for 20 realizations.

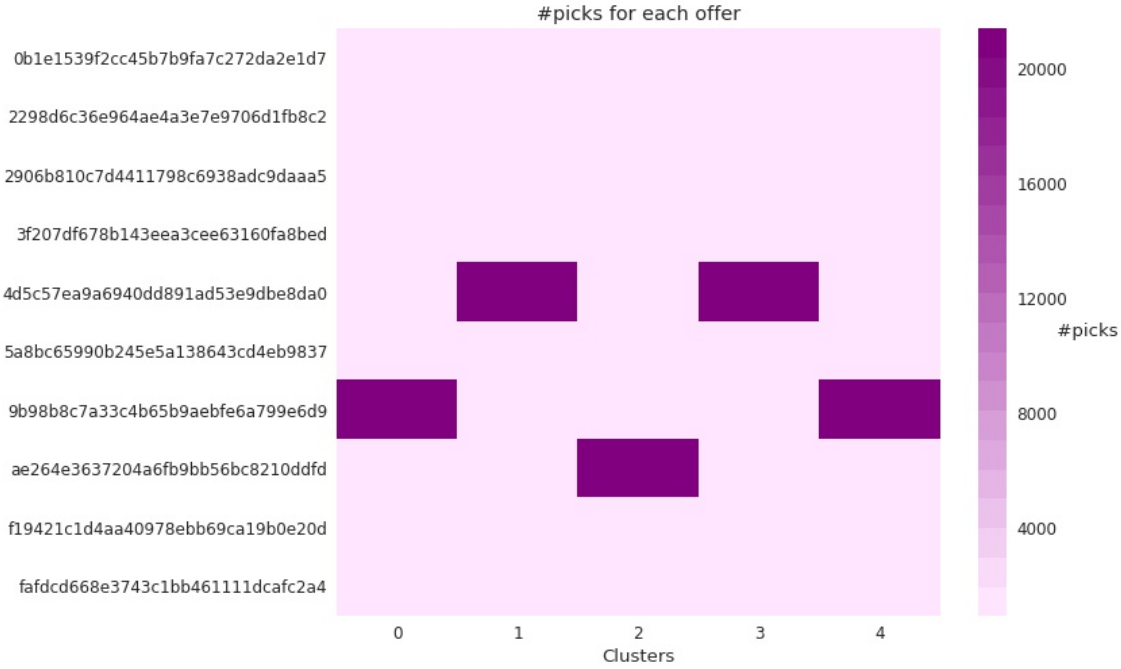

Figure 16. Number of offer picks by cluster.