MATEMÁTICA
ESPOL - FCNM JOURNAL
2020, Vol. 18, No. 1 Enero - Junio

Application of the Cross-Entropy Method to the Dynamic Assortment Optimization Problem

Aplicación del Método de Entropía Cruzada al Problema de Optimización Dinámica de Surtido

José Manuel Vera Aray

Recepción: 21/03/2020 **Aceptación:** 08/06/2020 **Publicación:** 30/06/2020

Abstract This work considers an assortment optimization problem, under capacity constraint and unknown demand, where a retailer offers an assortment and observes the sale of one of the products according to a multinomial logit choice model. In this problem, named as the dynamic assortment optimization problem (DAOP), the retailer must offer different assortments in each period to learn the customer preferences. Therefore, the trade-off between exploration of new assortments and the exploitation of the best known assortment must be balanced. Similarities between sampling and exploration are established in order to apply the cross-entropy method as a policy for the solution of the DAOP. The cross-entropy method finds a probability distribution that samples an optimal solution by minimizing the cross-entropy between a target probability distribution and an arbitrarily selected probability distribution. This requires the DAOP to be formulated as a knapsack problem with a penalty for offering assortments that exceed capacity. The results are compared with adaptive exploration algorithms and, experimentally, the cross-entropy method shows competitive results. These results suggest that the cross-entropy method can be used to solve other sequential decision-making problems.
Keywords cross-entropy method, dynamic assortment optimization, multinomial logit choice model.

Resumen Este trabajo considera un problema de optimización de surtido, bajo restricción de capacidad y demanda desconocida, donde un vendedor ofrece un surtido y observa la venta de un producto según un modelo de elección logit multinomial. En este problema, llamado como el problema de optimización dinámica de surtido (PODS), el vendedor debe ofrecer diferentes surtidos en cada período para aprender las preferencias del consumidor. Por lo tanto, el *trade-off* entre la exploración de

José Manuel Vera Aray, M. Eng.
Lecturer at ESPOL Polytechnic University, Escuela Superior Politécnica del Litoral, ESPOL, Facultad de Ciencias Naturales y Matemáticas (FCNM), Campus Gustavo Galindo Km. 30.5 Vía Perimetral P.O. Box 09-01-5863, Guayaquil, Ecuador, e-mail: jomavera@espol.edu.ec, https://orcid.org/0000-0003-1840-1040

nuevos surtidos y la explotación del mejor surtido conocido debe ser equilibrado. Se estableció similitudes entre el muestreo y la exploración con el fin de aplicar el método de entropía cruzada como política para la solución del PODS. El método de entropía cruzada encuentra una distribución de probabilidad que muestrea una solución óptima al minimizar la entropía cruzada entre una distribución de probabilidad objetivo y una distribución de probabilidad seleccionada arbitrariamente. Esto requiere que el PODS se formule como un problema de la mochila con una penalización por ofrecer surtidos que superan la capacidad. Los resultados se comparan con algoritmos de exploración adaptativa y, experimentalmente, el método de entropía cruzada muestra resultados competitivos. Estos resultados sugieren que el método de entropía cruzada se puede utilizar para resolver otros problemas de toma de decisiones secuenciales.
Palabras Claves método de entropía cruzada, modelo de elección logit multinomial, optimización dinámica de surtido

1 Introduction

1.1 Overview of the problem

In revenue management, it is studied the problem of finding an assortment of products, that when offered to a customer it maximizes revenue. Usually, companies have constraints that do not allow to display all existing products in the assortment. It is also known that customer preferences and the relationship between products generally influence the assortment revenue, therefore it is important to determine the assortment that meets capacity constraints and maximizes revenue.

If the customer preferences are known and these does not change over time, then the problem centers on finding the optimal assortment that should be offered throughout the selling season. This problem is called the *Static Assortment Optimization Problem*. When the customer preferences are unknown, then the seller must estimate them by sequentially offering assortments of products in the selling season. This problem is called the *Dynamic Assortment Optimization Problem* (DAOP) (Caro & Gallien, 2007; Sauré & Zeevi, 2013) and it is detailed in section 2.2. This situation is more common and has applications in retail, e-commerce, pricing and online advertising (Rusmevichientong, Shen, & Shmoys, 2010; Kök & Marshall, 2007; Slivkins, 2019). Since customer preferences are unknown, it is not expected to determine the optimal assortment at the beginning of the selling season but to offer assortments to explore their revenue and learn about customer preferences. Thus, in this setting is desirable to maximize the expected cumulative revenue in the selling season. In section 2.3 is shown that the maximization of the expected cumulative revenue is equivalent to minimizing the expected cumulative regret.

1.2 Contribution

The main contribution of this work is to apply the cross-entropy method, commonly used for optimization, to a sequential decision-making problem such as the DAOP. In general, optimization methods are not used for sequential decision-making problems due to a number of reasons. The main reason is that optimization methods expect that the outcome (usually a real number) of a decision to be deterministic and the decision-outcome relationship to be known. Unlike optimization problems, in sequential decision-making problems, the outcomes of the decisions are stochastic and the decision-maker is unaware of the decision-outcome probability distribution. For this reason, the decision maker must explore between decisions so as to learn the decision-outcome relationship. In optimization problems, the cross-entropy method "learns" a probability distribution, which generates an optimal solution, by sampling and minimizing the cross-entropy. This work establishes a parallelism between exploration and sampling and it is shown that the cross entropy method can be used in the DAOP.

2 Background

2.1 Stochastic Multiarmed Bandit

There are certain problems in which sequential decisions must be made to optimize over time, that is, decisions must take into account not only today but also the uncertainties of tomorrow. In addition, the decision maker has no knowledge of the relationship between decision and outcome. In other words, this type of problem is to optimize in the face of uncertainty. In particular, consider a situation where a decision maker faces a series of actions, each one with a random reward. Each action is independent of time and other actions. The distribution of rewards are unknown and the decision maker can adaptively learn a policy by taking different actions and observing the rewards. Thus, the decision maker must manage the trade-off of exploring different actions, in order to learn, versus exploiting an action that currently seem the best.

The Stochastic Multiarmed Bandit (MAB) is a model that captures the exploration-exploitation trade-off of the above problem. The name comes from the parallelism between the situation where a gambler must chose between arms in a slot machine and the situation where one must make an action in order to maximize reward. In the MAB problem the decision maker, in each round $t \in \{1,...,T\}$, must chose one of the arms (or actions). Let $I_t \in \{1,...,S\}$ denote the arm pulled at the tth timestep. Pulling the arm $I_t = i$ at timestep t gives a random reward $r_t \in \mathbb{R}$. The reward is observed immediately after choosing the arm and is randomly distributed with mean μ_i, where $i \in \{1,...,S\}$. Thus, $\mathbb{E}[r_t | I_t = i] = \mu_i$. Consequently, one must decide which arm to choose at each timestep in order to maximize the expected

cumulative rewards at timestep T based only in the observed outcomes by pulling arms (Agrawal, 2019).

2.2 Dynamic Assortment Optimization Problem

Given a set of products $\mathcal{N}$, let $\mathcal{S} = \{S \subseteq \mathcal{N} : |S| \leq K\}$ be the set of potential assortments. A seller at each timestep t offers an assortment $S_t \in \mathcal{S}$ and observes a customer purchase $c_t \in S_t \cup \{0\}$. The 0 element means "no-purchase decision". Initially, is unknown the customer preferences of the products. Each time an assortment is offered, a purchase decision is observed that gives information about the utility of the purchased product. The seller wants to find the assortment that gives the highest revenue. The dynamically selection of assortments can be modelled as an MAB problem. In this context, we have an MAB where each arm is an assortment of K products. Thus, every time the seller pulls an arm, he offers an assortment to the customers. The feedback of the pulled arm represents the revenue due to the offered assortment.

The expected revenue depends on the relationship of the subset of products that make up the assortment and that the seller does not know. This relationship is the substitution that products experience due to customers preferences. This *substitution effect* influence the expected demand of each product in the assortment. The Multinomial Logit (MNL) choice model is used to model this effect. This choice model is commonly used and is well studied in the literature (Agrawal, Avadhanula, Goyal, & Zeevi, 2019, 2017; Rusmevichientong & Topaloglu, 2012; Wang, Chen, & Zhou, 2018). Under the MNL choice model, the probability of choosing a product i ($c = i$), assuming a utility of 0 to the "no-purchase decision", on the assortment s is:

$$P(c = i|S = s) = \begin{cases} \frac{e^{\mu_i}}{1+\sum_{j \in s} e^{\mu_j}} & \text{if } i \in s, \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where μ_i is the mean utility of product i.

Since the customer can only buy one unit of any product from the assortment, then the expected revenue of an assortment $S \in \mathcal{S}$ is:

$$R(S, \boldsymbol{\mu}) = \sum_{i \in S} \frac{r_i \cdot e^{\mu_i}}{1 + \sum_{j \in S} e^{\mu_j}}, \tag{2}$$

where $\boldsymbol{\mu} = \{\mu_1, ..., \mu_N\}$ is the mean utility vector and r_i is the revenue of product i.

If there are N products and each assortment is made up of K products, there is a exponential number of assortments that can be offered. We cannot explore indefinitely as this will possibly lead to many low-revenue assortments being offered. At each timestep the decision maker faces the dilemma of keep exploring new assortments or exploit the best assortment seen so far. The longer the exploration period, the

greater the probability of finding a near-optimal assortment but with a high cumulative regret. Conversely, if the exploration period is short, the cumulative revenue is expected to be lower at the beginning but at the end of the selling season it could end up being higher. This raises the following question, how long should we explore before we settle? This is the exploration versus exploitation trade-off. This trade-off and also the fact of making decision over time under uncertainty differentiates the DAOP from other optimization problem.

2.3 Regret

Because we are interested in maximizing the cumulative revenue as we explore new assortments, we want to find a "good" assortment in the fewest number of offerings. This is equivalent to minimizing losses due to "bad" assortments. In other words, maximizing the cumulative revenue is equivalent to minimizing the cumulative regret. The regret is the loss of revenue for offering a suboptimal assortment. Since the revenues are not deterministic, we want to minimize the expected cumulative regret. The expected cumulative regret of an algorithm π with a time horizon T is defined as,

$$Reg_{\pi}(T,\boldsymbol{\mu}) = \sum_{t=1}^{T} \left[R(S^*,\boldsymbol{\mu}) - \mathbb{E}_{\pi}[R(S_t,\boldsymbol{\mu})] \right], \tag{3}$$

where $S^* = \text{argmax}_{S \in \mathcal{S}} R(S,\boldsymbol{\mu})$.

The expected cumulative regret measures the performance of an algorithm on an MAB problem (Lattimore & Szepesvári, 2020) and as shown in section 2.2, the DAOP is an instance of an MAB.

3 Methodology

In this section, is detailed the Cross-Entropy Method for the DAOP. First, in section 3.1 the relationship of the DAOP with the knapsack problem is identified. Then, section 3.2 describes the associated stochastic problem of the knapsack problem. Finally, in section 3.4 the cross-entropy method for the associated stochastic problem is detailed.

3.1 Knapsack Problem

The Dynamic Assortment Optimization Problem (DAOP) is formulated as a Knapsack problem. The Knapsack problem is a combinatorial optimization problem

were, under capacity constraints, one must find the combination of elements that maximizes some objective function. In the DAOP context, the elements are the products and the constraint is the size of the assortment.

Following the formulation in (Botev, Kroese, Rubinstein, & L'Ecuyer, 2013), each product j have an associate revenue r_j and must be selected which product will be in the assortment of size K in order to maximize the revenue. The integer programming formulation is

$$\begin{aligned} \max_{\mathbf{x}} \quad & S(\mathbf{x}) = \sum_{j=1}^{n} r_j \cdot x_j \\ \text{subject to} & \sum_{j=1}^{n} x_j \leq K \\ & x_j \in \{0,1\}, \end{aligned} \tag{4}$$

where x_j are the binary decision variables; $x_j = 1$ if the product j is in the assortment and $x_j = 0$ otherwise. By adding a penalty to the objective function is defined (4) as a single function, as follows,

$$S(\mathbf{x}) \doteq \sum_{j=1}^{n} r_j \cdot x_j - \beta \cdot I_{\{\sum_j x_j > K\}}, \tag{5}$$

where β is a penalty value in case that the constraint in (4) is not satisfied. In the context of the DAOP, β is a penalty for having more than K products in the assortment.

3.1.1 Penalty on the objective function of the DAOP

It is assumed that the dynamic assortment optimization is done in a e-commerce environment. Given the flexibility of e-commerce, the online retailer may offer an assortment of size greater than K. In this situation, offering many products generates a high inventory maintenance cost but almost no display cost. Thus, the assortment size is considered as a soft constraint. Following this idea, the function (5) allows to explore assortments larger than K but at a higher cost.

3.2 Associated Stochastic Problem

The formulation (4) is only concerned with finding a vector $\mathbf{x}^*$ that maximizes $S(\mathbf{x})$. From this formulation can be derived an associated stochastic problem (ASP). In this associated problem it is estimated the maximum value γ^* and the parameters $\mathbf{u}$ of a probability density function that generates a random vector $\mathbf{X}$ with concentrated probability density. In particular, instead of finding the vector $\mathbf{x}^*$ we find the maxi-

mum value γ^* and the parameters $\mathbf{u}$ that generates, with high probability, a random vector $\mathbf{X}$ such that $S(\mathbf{X}) \geq \gamma^*$.

In general, lets define the maximum γ^* as:

$$\gamma^* = \max_{\mathbf{X} \in \mathcal{X}} S(\mathbf{X}), \tag{6}$$

where $\mathcal{X}$ is the set of all possible vectors $\mathbf{X}$. Then, the ASP is defined as,

$$\ell(\gamma) = P_{\mathbf{u}}(S(\mathbf{X}) \geq \gamma) = \mathbb{E}_{\mathbf{u}} I_{\{S(\mathbf{X}) \geq \gamma\}}, \tag{7}$$

where $\mathbf{X}$ is distributed according probability density function (pdf) $f(\cdot\,;\mathbf{u})$ with parameters $\mathbf{u}$ and I is the indicator function. Equation (7) defines the estimation of the parameters that maximizes the probability of the event $S(\mathbf{X}) \geq \gamma$. In particular, the CE method maximizes γ and estimates the parameters $\mathbf{u}$ of the pdf that generates a random vector $\mathbf{X}$ that maximizes $\ell(\gamma)$.

$\mathbf{x}$ is a binary vector in the knapsack problem formulation. Thus, it is assume that $\mathbf{X}$ is distributed as a multivariate Bernoulli distribution and its pdf is

$$f(\mathbf{X};\mathbf{u}) = \prod_{j=1}^{n} u_j^{x_j} \cdot (1-u_j)^{1-x_j}, \tag{8}$$

3.3 Cross-Entropy Method

The Cross-Entropy (CE) method was first introduced as an approach for the estimation of rare-event probabilities but since then it has been applied to many problems such as reinforcement learning and optimization (Botev et al., 2013). In the case of optimization problems, the method minimizes the *cross-entropy* or *Kullback-Leibler divergence* to estimate a vector sampling distribution with probability mass concentrated in a region of near-optimal vectors. The method starts with a specified random mechanism that, through sampling and observing the outcomes of the samples in some function, allows updating the parameters of this same mechanism (Rubinstein & Kroese, 2004). This allows to generate better samples and converge to the optimal parameters. In the context of the ASP, the CE method iteratively updates γ and $\mathbf{u}$, in order to find a γ' close to γ^* and the values of parameters $\mathbf{u}'$ that maximizes (7). In the following section is detailed the CE method for the ASP.

3.4 CE method for the ASP

Given equations (5) and (8) the steps for the CE method are:

1. Initialize parameters $\hat{\mathbf{u}}_0$ of pdf f, the number of samples per iteration N, quantile q, smoothing parameter α and set $t = 1$.
2. Generate $\mathbf{x}_1, ..., \mathbf{x}_N \sim_{iid} f(\cdot\,; \mathbf{u}_{t-1})$. Calculate $S(\mathbf{x}_i)$ for all i and sorted them from smallest to largest: $S_1 \leq ... \leq S_N$.
3. **Update $\hat{\gamma}$:** Let $\hat{\gamma}_t = S_{N-N^q+1}$, where $N^q = \lceil qN \rceil$
4. **Update $\hat{\mathbf{u}}$:** Let each element of $\hat{\mathbf{u}}_t$ as

$$\tilde{u}_{t,j} = \frac{\sum_{k=1}^{N} I_{\{S(x_k) \geq \hat{\gamma}_t\}} \cdot x_{k,j}}{\sum_{k=1}^{N} I_{\{S(x_k) \geq \hat{\gamma}_t\}}}, \tag{9}$$

$$\hat{\mathbf{u}}_t = \alpha \cdot \tilde{\mathbf{u}}_t + (1-\alpha) \cdot \hat{\mathbf{u}}_{t-1}. \tag{10}$$

5. If stop criteria is met STOP; otherwise $t = t + 1$ and go to Step 2

4 Experiments

The experimental setups proposed in (Agrawal et al., 2019) were carried out using the CE method and the algorithms presented in (Agrawal et al., 2019) and (Sauré & Zeevi, 2013), then the results were compared. The performance metric was the cumulative regret.

In addition to comparing the method with the other algorithms, the following was analyzed:

1. The relation between separation of assortments and convergence rate to optimal assortment.
2. The relation between number of potential products N and expected cumulative regret.

4.1 Experiment 1

This experiment measures the robustness of an algorithm to different separability parameters ε. The greater ε, the more separation there is between the optimal assortment and the rest of assortments. Thus, the smaller the separability parameter, the more difficult it is to distinguish the optimal assortment. There are $N = 10$ products and must be found the optimal assortment of $K = 4$ products. The revenues are $r_j = 1$ for all $j \in \{1, ..., 10\}$. The utility parameters are $v_0 = 1$ and for $j = 1, ..., 10$,

$$v_j = \begin{cases} 0.25 + \varepsilon & \text{if } j \in \{1, 2, 9, 10\}, \\ 0.25 & \text{else}, \end{cases} \tag{11}$$

where $v_j = e^{\mu_j}$. The utility defined in (11) is used to calculate the expected revenue. The experiment was performed with the separability parameter $\varepsilon = \{0.05, 0.1, 0.15, 0.25\}$

and with a time horizon of $1x10^6$ timesteps. The penalty (β) of offering unfeasible assortments (assortments with more than K products) was established as 10% of the sum of the revenues of all products and the smoothing parameter $\alpha = 0.8$. The experiment was carried out 100 times and the mean cumulative regret for each of the algorithms is presented in Fig. 1.

4.2 Experiment 2

The second experiment used the "UCI Car Evaluation Database" (Dua & Graff, 2019). This dataset consists of consumer ratings of $N = 1,728$ cars. Following the setup specified in (Agrawal et al., 2019), the utility of the cars according to its attributes were estimated through Logistic Regression. Like experiment 1, it is assumed that $r_i = 1$ for $i = 1, ..., N$. The time horizon was $1x10^7$ timesteps. Given the unknown utility of each car we must find the optimal assortment of $K = 100$ cars.

Here it is defined the objective function $S(\mathbf{x})$ as,

$$S(\mathbf{x}) = \begin{cases} \frac{K}{\sum_{i=1}^{n} x_i} \cdot \frac{\sum_{i=1}^{n} r_i \cdot x_i \cdot e^{\mu_i}}{1+\sum_{i=1}^{n} x_i \cdot e^{\mu_i}} & \text{if } \sum_{i=1}^{n} x_i > K, \\ \frac{\sum_{i=1}^{n} r_i \cdot x_i \cdot e^{\mu_i}}{1+\sum_{i=1}^{n} x_i \cdot e^{\mu_i}} & \text{else.} \end{cases} \tag{12}$$

The objective function (12) indicates that the cost of an assortment larger than K will have a proportional reduction according to its size. Larger the size, greater the reduction of the revenue of the assortment.

4.3 Experiment 3

To analyze the convergence rate of the CE method for problems of different sizes, experiment 1 was carried out with different numbers of products N and different sizes of assortment K. Table 1 shows the different setups of the experiment.

Table 1 Setups for Experiment 3

N	K
30	6
50	10
80	16
100	20

Source: Own Creation

5 Numerical Results

Fig. 1 shows the results of experiment 1. The CE method performed better than the two others algorithms and this was reflected in a lower cumulative regret at timestep T. On early timesteps, the CE method had worse cumulative regret, but it learned the optimal assortment fast enough to produce zero regret in the remaining timesteps. Furthermore, this result shows that the CE method is influenced by the separation. The higher the separability parameter, the less cumulative regret the algorithm produces. On the other hand, the number of timesteps needed to find the optimal assortment is not so sensitive of the separation of assortments.

In Fig. 2 is shown that the CE method found the optimal assortment in a small fraction of the time horizon but generated a higher cumulative regret in comparison with Agrawal's algorithm. Given that Sauré's algorithm presents a linear regret rate, it will increase as the time horizon increases. The same can be deduced with Agrawal's algorithm, since it does not have a zero regret rate either. In contrast, the CE method has zero regret rate after less than 10% of the time horizon so a greater time horizon will not increase its cumulative regret.

Overall, the results of experiments 1 and 2 (Fig. 1 and Fig. 2) display faster convergence of the CE method to the optimal assortment than the other methods but it had higher regret rate in early timesteps.

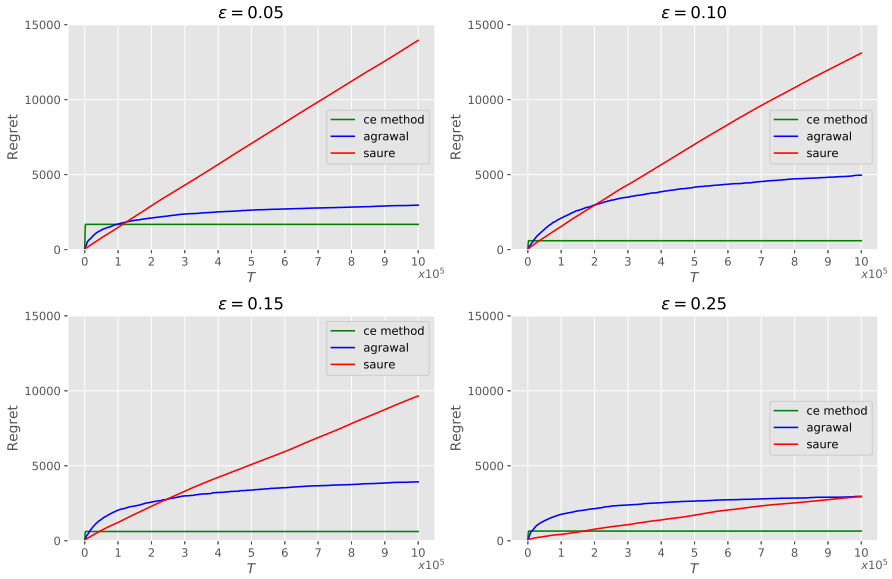

Fig. 1 Cumulative regret of the CE method and the compared algorithms for different values of separability parameter ε in Experiment 1

Source: Own Creation

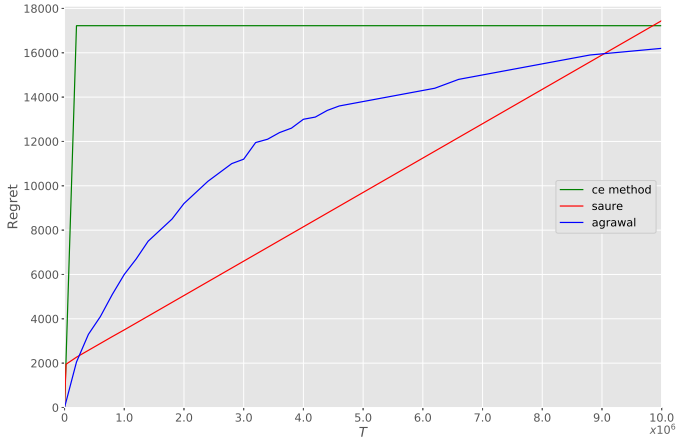

Fig. 2 Cumulative regret of the CE method and the compared algorithms in Experiment 2
Source: Own Creation

The results of experiment 3 showed that in all instances, even in very challenging settings like $N = 100$, the CE method arrived to the optimal assortment in less than the 10% of time horizon. Moreover, Fig. 3 shows that separability does not affect the convergence rate. The scale of the problem have more significance rather the separability of the assortments.

6 Discussion

Depending on how the penalty for offering unfeasible assortments is defined, the CE method can produce higher or lower cumulative regrets. Since the CE method finds the optimal assortment fast enough, one might wonder if this is more beneficial than lower cumulative regret, e.g., taking too long to find a satisfactory solution in situations where the customers preferences change over time would lead to a worse performance overall. Experimentally, the CE method showed a $O\left(\sqrt{|\mathcal{S}|T\log(|\mathcal{S}|T)}\right)$ cumulative regret upper bound on the DAOP instances. Although experimentally the CE method has a comparable performance to that of known adaptive exploration algorithms (Agrawal et al., 2019, 2017; Sauré & Zeevi, 2013; Slivkins, 2019), the upper and lower regret bounds of the CE method that allow theoretical comparison with other algorithms were not derived. As an extension to this work, it would be important to derive the regret bounds of the CE method on the DAOP.

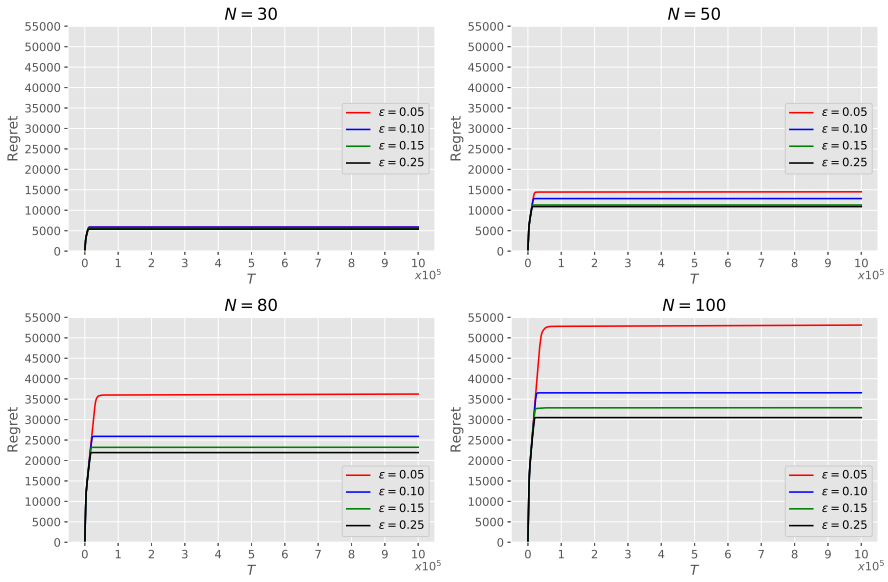

Fig. 3 Cumulative regret of the CE method for different assortment sizes N in Experiment 3
Source: Own Creation

7 Conclusions

The CE method was successfully applied to the DAOP and obtained competitive results. The results imply that when the time horizon is large, the CE method may even perform better than the compared algorithms. This suggests the potential use of the CE method on other sequential decision-making problems. The CE Method arrived to the optimal assortment during the selling horizon unlike the compared algorithms. The exploration phase of the CE method depends on the number of products N, but regardless of the number it is relatively short compared to the time horizon. In contrast, the compared algorithms did not find the optimal assortment and the cumulative regret could increase with the time horizon T.

A disadvantage of the CE method is that it starts exploring unfeasible assortments and therefore incurs in penalties. The experiments showed that the regret rate is extremely high in early timesteps but only during a relative short period compared to the time horizon. After this short period, the CE method found the optimal assortment and had a zero regret for the rest of the timesteps.

8 Bibliography

References

Agrawal, S. (2019). Recent advances in multiarmed bandits for sequential decision making. *INFORMS TutORials in Operations Research*, 167-188. doi: 10 .1287/educ.2019.0204

Agrawal, S., Avadhanula, V., Goyal, V., & Zeevi, A. (2017). Thompson Sampling for the MNL-Bandit. Retrieved from `http://arxiv.org/abs/1706.00977`

Agrawal, S., Avadhanula, V., Goyal, V., & Zeevi, A. (2019). Mnl-bandit: A dynamic learning approach to assortment selection. *Operations Research*, *67*(5), 1453–1485. doi: 10.1287/opre.2018.1832

Botev, Z. I., Kroese, D. P., Rubinstein, R. Y., & L'Ecuyer, P. (2013). The cross-entropy method for optimization from estimation to optimization. *Handbook of statistics*, 35–59. doi: 10.1016/j.memsci.2013.11.020

Caro, F., & Gallien, J. (2007). Dynamic assortment with demand learning for seasonal consumer goods. *Management Science*, *53*(2), 276-292. doi: 10 .1287/mnsc.1060.0613

Dua, D., & Graff, C. (2019). *Uci machine learning repository.* Retrieved from `http://archive.ics.uci.edu/ml/datasets/car+evaluation`.

Kök, A. G., & Marshall, L. F. (2007). Demand estimation and assortment optimization under substitution: Methodology and application. *Operations Research*, *55*(6), 1001-1021. doi: 10.1287/opre.1070.0409

Lattimore, T., & Szepesvári, C. (2020). *Bandit algorithms*. Cambridge University Press.

Rubinstein, R. Y., & Kroese, D. P. (2004). *The cross-entropy method: A unified approach to combinatorial optimization, monte-carlo simulation, and machine learning*. Springer Science & Business Media. doi: 10.1007/978-1-4757-4321-0

Rusmevichientong, P., Shen, Z. M., & Shmoys, D. B. (2010). Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations Research*, *58*(6), 1666-1680. doi: 10.1287/opre.1100.0866

Rusmevichientong, P., & Topaloglu, H. (2012). Robust assortment optimization in revenue management under the multinomial logit choice model. *Operations Research*, *60*(4), 865-882. doi: 10.1287/opre.1120.1063

Sauré, D., & Zeevi, A. (2013). Optimal dynamic assortment planning with demand learning. *Manufacturing & Service Operations Management*, *15*(3), 387-404. doi: 10.1287/msom.2013.0429

Slivkins, A. (2019). Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning*, *12*(1-2), 1-286. doi: 10.1561/2200000068

Wang, Y., Chen, X., & Zhou, Y. (2018). Near-optimal policies for dynamic multinomial logit assortment selection models. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems 31* (p. 3101-3110). Curran Associates, Inc. Retrieved from `http://papers.nips.cc/paper/7573-near-optimal-policies-for-dynamic-multinomial-logit-assortment-selection-models.pdf`