CentraleSupélec

ORRA - Decision Making
V. Auriau, V. Mousseau

Clustering and modeling customer preferences

1 Understanding customer preferences

Identifying and understanding its customer base is a key objective for retailers. It lets them improve their offering and optimize costs and revenues. For example, a car dealership owner might identify a few segments among its customers. Some clients might be looking for large cars some others might only look at the speed of the car they buy. From this point, a few questions remain: Are there more typologies of customers ? How each of this segment take into account other criteria (such as price, autonomy, etc...) ? Are there segments that will be happy with few alternatives and others for which a great diversity of options are needed ?
The objective of this practical work is to come up with a model that will use past data to analyze the behavior diversity among different customers. The models needs to:

- Clusterize the customers through their choices so that customers with similar decisions are grouped together.
- Determine for each cluster the decision function that lets the customers rank all the alternatives.

2 Supermarkets, assortment and preferences

A supermarket usually offers approximately 20,000 products, while the purchasing center of a distributor lists more than 150,000 items. Distributors must, therefore, select the products to sell in each store. Before the use of big data, this selection was carried out manually, with standard assortments based on business experience at the store or distributor level. This historical manual approach does not exploit the large amount of data available, starting with the baskets of products that customers buy, rich in information on preferences and complementarity between products.
The retailer objective is to propose an optimal assortment, that is to say a set of products to be put on the shelves which maximizes a pre-defined objective: the expected final income, the average margin or the volume of weekly sales. This choice depends on the group's strategy as well as the product categories. The problem is usually solved with two successive stages. The first stage aims at estimating the utility attributed by customers to a product in a given assortment. This utility must take into account the phenomena of substitution and cannibalization between products, as well as customer preferences. The

second step is to propose an optimal set of products given the computed utility of each product. This optimization usually takes into account additional constraints defined by the category managers or the supply chain, for example.

3 Determining customer preferences

3.1 Objectives

For this practical work, we will focus on better understanding customer preferences. We see a customer as a decision function when he comes to the supermarket. Once facing the shelf of a type of product he wants to buy, the customer assesses what are the different alternatives available. Considering the size, price, brand, packaging, and any other information, the customer ranks all the products in his mind and finally chooses his preferred alternative. A family with two young children will have very different preferences than two roommates planning their next party. Unfortunately, a supermarket holds very few information on its customers. We only have at our disposal the list of products in the supermarket and the different receipts for the daily purchases. Our objective is twofold:

- We want to clusterize the customers through their purchases so that customers with similar decisions are grouped together.
- We want to determine for each cluster the decision function that lets the customers rank all the products.

4 Settings and problem formulation

4.1 Formulation as a Mixed Integer Programming problem

We consider that our dataset is an aggregation of P observed preferences for undetermined customers. A preference is a couple of alternative (x, y) with x the preferred alternative over y. x and y are described on a defined list of criteria such as the price, the speed, etc... For a couple $(x^{(j)}, y^{(j)}), j \leq P$, we note it $x^{(j)} \succcurlyeq y^{(j)}$.
We want to use UTA models to represent the customer decisions functions. The model being both complex enough to fit data and easily interpretable will let one easily take decisions upon estimation. The main goal of this practical work is to come up with an extended formulation of the classical UTA model that lets us regroup the observed preferences within clusters.

Problem definition The input of our problem is defined by:

- K, number of clusters,
- n number of criteria (product features),
- L number of linear segment UTA marginal functions, i.e., the scale on criterion i is subdivided into L intervals $[x_i^0, x_i^1[$, $[x_i^1, x_i^2[, \dots [x_i^{L-1}, x_i^L]$; the marginal values of the breakpoints are denoted $u_i^k(x_i^l),\ \forall k = 1..K,\ \forall i = 1..n,\ \forall l = 0..L$.
- the learning set, i.e., P pairs $(x^{(j)}, y^{(j)})$ where $x^{(j)} = (x_1^{(j)}, ..., x_n^{(j)})$ is preferred to $y^{(j)} = (y_1^{(j)}, ..., y_n^{(j)}),\ j = 1..P$. Each pair $(x^{(j)}, y^{(j)})$ should be correctly represented by at least one of the K additive value functions $u^k(.)$, i.e. $u^k(x^{(j)}) > u^k(y^{(j)})$.

For consistency, we will use the following notations:

- k cluster index, $k = 1..K$,

- i criterion index, $i = 1..n$ number of criteria (product features),
- j pairwise comparison index, $j = 1..P$
- l linear segment index, $l = 1..L$
- the decision function for cluster k is $u^k(x) = \sum_{i=1}^{n} u_i^k(x_i)$

Your objective is to learn K decisions functions u^k so that:

$$\forall k \in [|1, K|], \forall (x, y) \in \mathbb{R}^2, \epsilon \in \mathbb{R}^+, u^k(x) > u^k(y) + \epsilon \iff x \succcurlyeq_k y$$

and that:

$$\forall j \in [|1, P|], \exists k \in [|1, K|] \; s.t. \; x^{(j)} \succcurlyeq_k y^{(j)}$$

First Exercise As a first exercise, you are asked to write a Mixed-Integer Programming formulation. The MIP takes a list of preferences as inputs and estimates preferences clusters with their corresponding decision functions. More precisely for each cluster a decision function needs to be estimated in the form of a UTA model. You have to use the given notations and you can use the first practical work as help.

Second Exercise Your second exercise will be to implement the model in Python and find a solution of this MIP for the first dataset provided. This first dataset, 'dataset_4' can be found in the linked GitHub repository. It is constituted of $P = 400$ pairs from $K = 2$ clusters of decision functions with $L = 5$ linear pieces, on $n = 4$ criteria. Your solution should be implemented in the TwoClustersMIP class and should follow the code signature. You can refer yourself to the notebook example.ipynb that will help you handle the provided code. Note that the cluster labels are given, letting you test your solution.
You can compare the results of your cluster-UTA model with different values of ϵ. How would you interpret such results? What would be your recommendation for the choice of its final value?

Third Exercise As you can test yourself, the MIP formulation does not scale well with larger datasets. It is important to develop another approach that will better scale with potentially thousands of data. Your third exercise is to come up with solutions that will let you estimate the model in reasonable time. For this purpose we provide the 'car preferences dataset'. The dataset is a collection of choices among six different car alternatives. Each car and the customer are described with several features. The objective is estimate a model on this dataset and to analyze the results. You are free to use any method you think fit to answer the problem. However keep in mind that an interpretable solution is always appreciated, particularly by business stakeholders.

5 Organisation of the practical work

5.1 Groups & schedules

The practical work is organized in two work sessions, December 19^{th} (8.15-11.30 am) and January 23^{rd} (1.30-4.45 pm), and an oral presentation of your results on Tuesday 13^{th} (from 1.30 pm) of February. For the second session, create groups of three students and register your group here.

5.2 Summary and Deliverables

You will present your results during an oral presentation organized the on Tuesday 13^{th} (from 1.30 pm) of February. Exact time will be communicated later. Along the presentation, we are waiting for:
You are asked to:

- Write a Mixed-Integer Progamming model that would solve both the clustering and learning of a UTA model on each cluster
- Code this MIP inside the TwoClusterMIP class in python/model.py. It should work on the dataset_4 dataset.
- Explain and code a heuristic model that works on the car dataset. It should be done inside the HeuristicModel class.

We are waiting for:

- A report summarizing you results as well as your thought process or even non-working models if you consider it to be interesting.
- Your solution of the first assignement should be clearly written in this report. For clarity, you should clearly state variables, constraints and objective of the MIP.
- A well organized Python codebase of the presented results. It is possible to provide a GitHub repository or a file archive. Add some documentation directly in the code or in the report for better understanding. The code must be easily run for testing purposes.
- In particular the code should contain your solutions in the class TwoClustersMIP and HeuristicModel in the models.py file. If you use additional libraries, add them inside the config/env.yml file. The command 'python evaluation.py' will be used to check that the models run. Make sure that it works and that your code complies with the formalism. For the MIP, the dataset used will be a new one, with the same standards as 'dataset_4'.

5.3 Datasets & Starting Kit

All datasets and helpful functions can be found on the GitHub repository. You can fork fork or clone it and use the different provided files as a starting point. Make sure that you use git lfs to download the datasets files. Particularly, we provide:

Datasets Two datasets, one for each assignement are given. 'dataset_4' is to be used for the first assignment and the car dataset for the second one. The first dataset is composed of three arrays: X, Y and Z. Each element j is so that: the cluster $Z[j]$ has expressed the $j-th$ preference: $X[j] \succcurlyeq_{Z[j]} Y[j]$. The second dataset is a pandas DataFrame that should be self explanatory. A notebook is provided to help you download it.

Helpful code example The notebook example.ipynb shows how to use the main Python object: data loader, model and metrics. You can also refer to the documentation within the code of these classes to understand how it works. An example with a random model is also provided.

Code Structure The file evaluation.py will be run for evaluation with others, not provided, datasets. For the first assignment you need to fill the Python class models.TwoClustersMIP with your solution. Make sure that this script works and precise additional Python libraries you might have used. For the second assignment it is not mandatory to follow such classes structure, but highly recommended.

5.4 Evaluation Metrics

Two metrics will be used to evaluate the performance of your solutions. The Python code is provided to help you in the file metrics.py. You are encouraged to introduce and use additional metrics you find relevant.

Explained Pairs This metric represents the ratio of pairs from a dataset that is explained by the model. We consider that a pair is explained if at least one of the clusters expresses the same preference as the pair.

$$pe = \frac{1}{N} \sum_{j=1}^{N} \left[\mathbb{1}_{\exists k \leq K | u^k(x^{(j)}) \geq u^k(y^{(j)})} \right]$$

Clustering Intersection This metric represents how well the reconstituted clusters match the ground truth ones.

$$ci = \frac{1}{N} \sum_{j=1}^{N} \left[\mathbb{1}_{z^{(j)}_{pred} = z^{(j)}_{true}} \right]$$

with $z^{(j)}$ representing the cluster associated to the pair $(x^{(j)}, y^{(j)})$.

Final Tips & Tricks

- In order to write your MIP, you should write the variables, the constraints and finally the optimization objective of the problem. Use the same notations as this subject.
- Be sure to understand the provided code. The notebook but also the documentation and the code is here to help you.
- You should use the Linear Programming Python libraries Gurobi (recommended) with free academic licenses or OR-tools which is open-source.
- Data and results visualisation is an important part of understanding a problem. Don't hesitate to share any insightful graph or plot you have done.
- If you have questions or need help you can contact me at vincent.auriau@artefact.com