Business Cases with Data Science

Spring Semester 2021

Course Coordinator: Professor Fernando Bação

Teaching assistants: David Silva and João Fonseca

Business Case 4

Recommender System for ManyGiftsUK

Project team

Henrique Costa - M20200652

Pedro Mendes - M20200648

Rui Soromenho - M20200578

Rafael Soromenho - M20200969

Contents

List of figures

1. INTRODUCTION

The goal of this project is to develop a recommendation system that can help an online retailer – ManyGiftsUK – in increasing its sales. The system should be able to identify which products are likely to be interesting for the company's customers, thus enabling strategies to push such items resulting in more purchases. To do this a dataset containing information on 25,900 transactions made between December 2010 and December 2011 will serve as basis for the analysis. Given the nature of information available, the project will focus on a collaborative filtering approach as it can make the best use of the implicit data that has been collected. Additionally, a strategy to produce recommendations for new customers, commonly referred to as 'Cold Start' problem, will be proposed. Below is a high-level overview of the steps that will be taken to produce the desired output:

- **Business Understanding and Data Exploration:** Developing intuition and understanding the data considering the business context.
- **Data Cleaning, Selection and Transformation**: Preparing data to enable modeling.
- **Modelling alternative techniques** to maximize the capability to predict user preferences.
- **Evaluate results and select best model** to apply.
- Provide input on the **deployment** strategy to implement the model and consider its **business implications.**

2. BUSINESS UNDERSTANDING

2.1 BACKGROUND

Recommendations are a key issue in the online retail industry, as illustrated in the example of Amazon for which around 35% of total sales are driven by recommendation systems. In the specific case addressed in this project, ManyGiftsUK intends to create a recommender system by leveraging transactional data collected over a 1-year period. There are two common strategies to build recommendations: content-based and collaborative filtering. While content-based approaches focus on profiling user and products according to their attributes and using these attributes to match the user with corresponding products, collaborative filtering builds on a history of user-item interactions to infer unknown user-item associations. The type of information collected by ManyGiftsUK seems to indicate that the second strategy, collaborative filtering, would better suit the task at hand. The general idea is that the company would be able to select a range of suitable products based on the recommender's output to display in its online shop main landing page in an attempt to boost sales performance.

2.2 BUSINESS OBJECTIVES, DATA MINING GOALS AND SUCCESS CRITERIA

The primary business goal of ManyGiftsUK is to improve sales performance. There are studies linking 'easier choices' such as the ones that can be offered to customers by using effective recommendation systems with increased purchases. Therefore, the reasoning guiding the work presented in this project is that a recommender system would be a suitable solution to maximize the sales performance.

The data mining output should consider two scenarios. A first is to provide recommendations for customers with a significant transaction history, a second is to generate recommendations for new customers – a challenge commonly referred to as 'Cold Start' problem. Success should be measured primarily using recall-based metrics - justification of the used metric is provided in more detail in the results evaluation section. The idea is that a set of recommended items would be generated using a training dataset. When looking at future purchases for a user the model would be evaluated by calculating if the purchased item was part of the list of recommended items. The same reasoning could be applied to new users. How often is the purchased item by new users part of the recommended item set? To evaluate results in the context of recommendation systems, defining a benchmark using a Popular Recommender is also advised.

A secondary goal of the project is to extract any additional insights that can help management better understand sales trends and customer profiles.

As per the above, the outputs of the project should focus on three key areas:

- A model that can return a set of recommended items for a user based on past transactions.
- A proposed strategy for generating recommendations for new customers.
- Discussion of the deployment of such models.

3. Predictive Analysis

3.1 Data understanding and exploration

The data is made of 25,900 transactions containing 541,909 user-item interactions. Each interaction is described across 8 variables:

- **InvoiceID:** Unique identifier of the transaction
- **CustomerID:** Unique identifier of the customer
- **StockCode:** Unique identifier of the product
- **Description:** Product description
- **UnitPrice:** Unit price of each product
- **Quantity:** The quantities of each product per transaction.
- **InvoiceDate:** Day and time when each transaction was completed.
- **Country:** Name of the country where the customer made the purchase.

An initial verification shows that there is a considerable number of missing values in the variables CustomerID and Description. A missing CustomerID seems to indicate the customer is new or not registered. These customers were assigned a specific ID to be easily distinguishable from the registered customer base. Missing descriptions were set to unknown.

An exploratory analysis was performed with the intention of better understanding sales patterns and customer profiles.

- Sales over time

The dataset provided has information on ManyGift transactions from the 1st of December 2010 to the 9th of December 2011 on a total of 373 days, corresponding to 53 weeks. It is worth noticing that ManyGifts business thrived the most between the months of October and December with a special emphasis on the month of November where more than 80 000 transactions were recorded (on average, 45 000 were recorded by month). It is also worth noticing that most orders are done on daytime between 8AM and 6PM with very few transactions outside this timeframe.

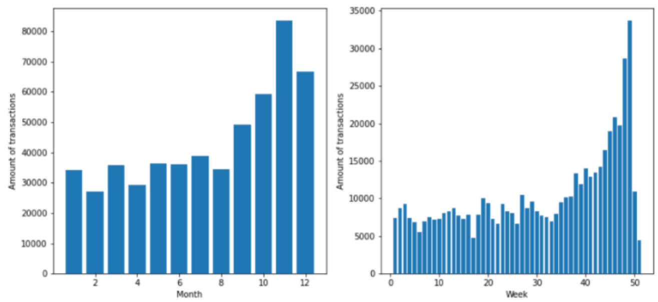

Figure 1 - Sales over time

- Sales per country

The dataset shows that ManyGifts has clients on 38 different countries however, when considering sales per country, the United Kingdom is the country where most of the products are sold to. In fact, the United Kingdom alone represents more than 80% of the total revenues in our dataset. Other markets like Ireland, the Netherlands, Germany, France and Australia also have an important contribution and together, represent about 10% of the total revenues.

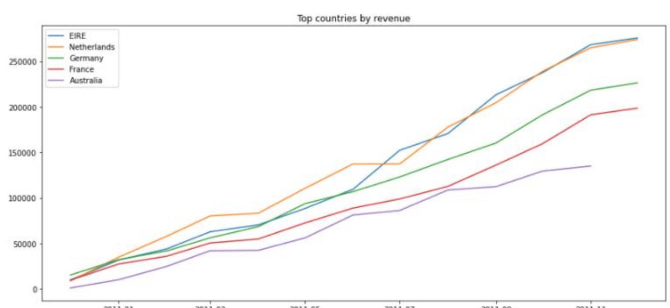

Figure 2 - Sales per Country

- Most sold items

The most sold items are shown below both from a monthly and a cumulative revenue perspectives. It is interesting to notice that no clear pattern is identifiable in either graph however (and in line with the findings of sales over time), it is possible to point out positive variations in revenue in the later stages of the year (although not as noticeable as when analyzing total sales).

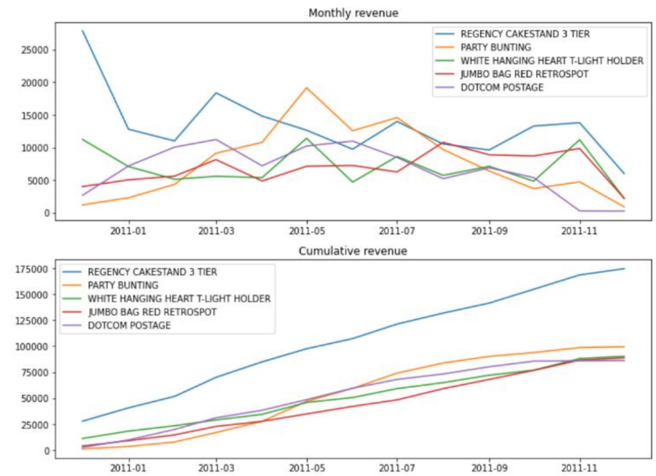

Figure 3 - Most sold items

- Customer profile

Regarding customer profile we should point out that a considerable number of ManyGifts customers have been clients for more than 300 days. This could mean one of two things: either they did their first purchase a long time ago and never came back as the service was below their expectations; or they are loyal clients that are happy with how ManyGifts works and keep on buying. If you consider the last two graphs below you can quickly conclude that the number of days between invoices is usually low and that most of our clients bought from ManyGifts in the last 90 days, meaning that the company is indeed fulfilling the customers expectations and needs.

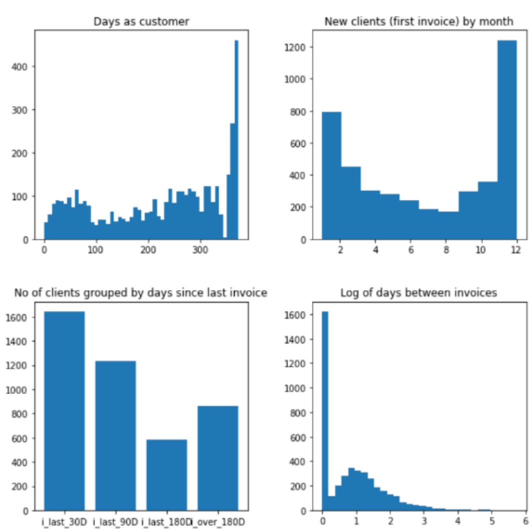

Figure 4 - Customer profile

3.2 Data preparation

To ensure only the relevant information is considered when training the collaborative-filtering approach further explained in the modeling section, the data was transformed to better encapsulate the user-item interaction patterns that are helpful for producing valuable recommendations for ManyGiftsUK. The most relevant steps in data preparation are summarized below. These are presented in more detail in the notebook that supports the report.

- Handling missing values - missing CustomerIDs were isolated to be tackled with a separate approach.
- Aggregate product when they appear several times in the same invoice - should be counted as one purchase.
- Exclude cancelations - by filtering the dataset excluding InvoiceIDs starting with C while also only including positive quantities.
- Exclude irrelevant products - by filtering the dataset excluding StockCode that correspond to products such as post or banking fees (full list available in the notebook).

3.3 Modeling and results evaluation

3.3.1 Collaborative-Filtering

The steps taken to produce predictions with such approach have several aspects that are particular to the specific type of information used – implicit data - and optimization performed. These are highlighted below.

Once the data has been prepared as per section 3.2., ensuring transactions that corresponded to cancelations were excluded as well as irrelevant products, a user-item matrix can be generated. This matrix will summarize the number of times a user has purchased each item. For the applied model, preference is indicated by the fact that a user has purchased a product in the past. Thus, the 'preference' matrix would be represented with a value of 1 in the case a past purchase is found in the dataset and a 0 otherwise. Moreover, the number of times the customer has purchased the item would indicate confidence in that preference. The weight of the confidence in the model results is regulated by a parameter defined as alpha. In the case of ManyGiftsUK the best results were attained with an alpha equal to 30. An issue arising from creating the above-described matrix is sparsity. Because in most cases there will be no history of interaction between a user and an item, the matrix will store mostly zeros. To minimize the negative impact of sparsity in the model ability to effectively estimate the user and item factors a threshold for minimum user-item interactions was introduced. Items bought less than 10 times or customers that purchased less than 10 products were not included in the optimization.

```
Raw dataset info
-----------------
Number of users: 4336
Number of items: 3661
Sparsity: 2.497%
Limited dataset info
-----------------
Number of iterations until convergence: 3
Number of users: 3746
Number of items: 2874
Sparsity: 3.624%
```

Figure 5 - Sparsity

The method used to split the dataset into train and test is also important, as it ensures recommendations are made for future purchases and information leakage is avoided. Train and test were divided based on a timestamp to ensure all transactions in the train set were done previously to the ones in test. The idea of implementing the presented model is to use the known interactions between users and items to estimate a set of user and item factors that best represent the patterns contained in the data. The factors can then be used to infer unknown user-item preferences. An alternating least squares method is used to find the optimal user and item factors that are in turn used to make recommendations. A range of factors was tested and the number of factors that produced the best results was 200. Generally, increasing the number of factors tends to improve the model results however it must be considered that using too many factors could result in overfitting.

	pop_model	als_model
precision	0.117783	0.066594
map	0.054338	0.027960
ndcg	0.119659	0.065333
auc	0.523073	0.512717

Figure 6 - Models evaluation

To evaluate model performance a range of metrics was calculated. Different metrics provide alternative perspectives to evaluate performance. Given the set of assumptions previously explained related to how preference and confidence are defined in the proposed model, the primary metric used was recall at K with K equal to 10. This is because there could be several causes explaining why a user did not buy a certain product beyond preference (e.g. lack of knowledge or availability) making precision-based metrics less appropriate. Precision answers the question of how many of the recommended items did the user purchase which would not provide the best indication given the model assumptions. On the other hand, buying a product can be seen as an indication of preference, thus a better question to evaluate is: from the purchases the user made in the future, what products would have been included in the recommender system output? The result of recall at K can be interpreted as follows - in 6.7% of the cases an item purchased by a user in the future would have been recommended by the system in the top ten (as K was defined as equal to 10). Given that the model is currently underperforming a model in which the most sold items would be recommended to all users it requires further optimization until it is ready for deployment.

3.3.2 Strategies for the Cold-Start Problem

The cold-start problem occurs when new users arrive in e-commerce platforms. In this case, the taste and preferences of the new users are unknown and so it is impossible to "fill in the blank" using typical matrix factorization techniques. To solve this problem, it was decided to use a popularity-based strategy, with demographic filtering and seasonality.

Before going to our suggested approach to the cold start problem it is important that the reader also understands the difference between exploration and exploitation problems as this will also highly influence which products to show to customers. The exploitation approach, the one we used on our recommendations below, finds the items that sell very well and recommends them more often to other users. However, there is also another valid approach that should be taken into consideration: the exploration approach. The latter, wants to recommend to the users' items that have not been shown as much because these items can be even more popular that the ones we usually recommend.

When the user enters the system (in this case the website), we only have information about his/her location from his/her IP address. Using that location, we suggest a recommending system based on the most popular products sold in each country. For example, if I access ManyGiftsUK website for the first time from the UK the system will recommend the most sold products in the United Kingdom whereas if I access the website in Germany, the suggested items will be the German top sellers.

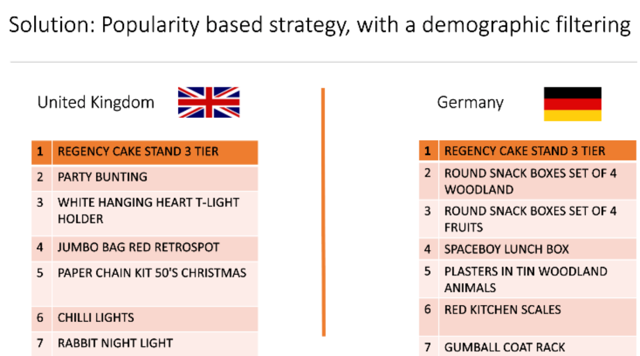

Figure 7 - Example of a solution for the cold start problem

Regarding seasonality, we also suggest recommending products to new users based on the month of the year and the most sold products sold in the last 7 days. That way you will not only be able to adapt your suggestions to seasonality but also quickly adapt your recommendations to the latest trends that are selling the most at that time.

4. Business implementation challenges & recommendations

In practice, the decision process to define how to determine recommendation for a certain user following the below logic.

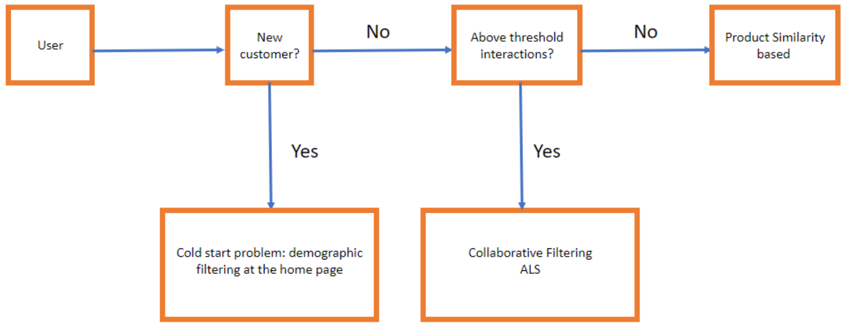

Figure 8 - User process

On the deployment phase challenges in implementing the recommender system might happen with time. The three main challenges we identified relate to lack of data, changing data and changing user preferences. Let us analyze each one of them and also possible solutions for these problems:

- **Lack of data:** This is probably the most important challenge: recommender systems need a lot of data to effectively make recommendations. It is no coincidence that the companies most identified with having excellent recommendations are those with a lot of consumer user data like Google, Amazon, Netflix. The more items and user data a recommender system has to work with, the stronger the chances of getting good recommendations are and so, if needed, ManyGiftsUK can retrieve more data from open-source datasets or establish partnerships with similar enterprises and share their database.
- **Changing data:** Past behavior of users might not always be a good tool to do recommendations because consumption trends are always (and rapidly) changing. For that reason, it would be important to keep a close eye on social media in order to predict what the consumers are interested in an ever-changing world.
- **Changing user preferences:** One day a user might be browsing ManyGiftsUK for new books for himself, but the next day he will be on your website searching for a birthday present for his sister. The point here is that one's preferences and needs will be changing constantly and so it is also important to collect and understand your user features and their psychographics — how customers think, feel, and react as well as their values, attitudes, and biases.

One recommendation that we believe will help the company facing all these challenges is the collection of explicit data. Explicit data is given when a customer clearly shows how they feel about a product through a rating or a like, for example. This type of data is clear, unambiguous and gives us a definite picture of the user preferences.

5. Conclusions

Building an effective recommendation system is a challenging task. The best recommender systems are produced when a significant quantity as well as quality of data is available including both explicit and implicit information.

While the project provides an overview of how implicit data based almost exclusively on transactions can be leveraged to create recommender systems, the results show the limited capability for the collaborative-filtering approach presented as final output. For that reason, we have recommended the use of explicit data to build a more trustworthy recommendation system.

6. References

- Macmanus, R. (2009, January 28). *5 Problems of Recommender Systems*. ReadWrite. https://readwrite.com/2009/01/28/5_problems_of_recommender_systems
- Milankovich, M. (2017, July 5). *The Cold Start Problem for Recommender Systems - Yusp*. Medium. https://medium.com/yusp/the-cold-start-problem-for-recommender-systems-89a76505a7
- Chubak's Machine Learning Imporium. (2020, December 15). *Solving the Cold Start Problem in Recommender Systems*. https://chubek.github.io/recommender/2020/12/15/cold-start-problem-copy.html
- Y. Hu, Y. Koren and C. Volinsky, "Collaborative Filtering for Implicit Feedback Datasets," *2008 Eighth IEEE International Conference on Data Mining*, 2008, pp. 263-272, doi: 10.1109/ICDM.2008.22.
- Rosenthal, E. (2016, October 19). *Intro to Implicit Matrix Factorization: Classic ALS with Sketchfab Models | Ethan Rosenthal*. Ethanrosenthal.Com. https://www.ethanrosenthal.com/2016/10/19/implicit-mf-part-1/
- G. (2019, December 8). *Recommender Systems in Python 101*. Kaggle. https://www.kaggle.com/gspmoreira/recommender-systems-in-python-101#Recommender-Systems-in-Python-101
- Macmanus, R. (2009, January 28). *5 Problems of Recommender Systems*. ReadWrite. https://readwrite.com/2009/01/28/5_problems_of_recommender_systems/
- Clayton, T. (2019, September 5). *The Difference Between Implicit and Explicit Data for Business*. Medium. https://blog.mirumee.com/the-difference-between-implicit-and-explicit-data-for-business-351f70ff3fbf