BUSINESS CASES WITH DATA SCIENCE

MASTER DEGREE PROGRAM IN DATA SCIENCE AND ADVANCED ANALYTICS – MAJOR IN BUSINESS ANALYTICS

Business Case #4 – Recommender System

Data4Business Consulting

Débora Santos, number: 20200748

Pedro Henrique Medeiros, number: 20200742

Rebeca Pinheiro, number: 20201096

May, 2021

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

INDEX

1. INTRODUCTION

Thank you for choosing **Data4Business Consulting (D4B)** to help you with the challenge of better understanding your customer's preferences. Our main objective is to help **ManyGiftsUK** to build a recommender system that can facilitate user choices and suggest relevant items to customers.

The world is experiencing a great technological and digital revolution where understanding business data, customers, and their needs are essential for business success. Taking advantage of that, e-commerce is growing daily and now it is a fundamental strategy for any business to gain value, market share, and to stay relevant in the market. Consequently, the competition between sellers leads to a constant search for the improvement of their business models and decisions. One of the important challenges is helping customers sort through a large variety of offered products to easily find the ones they will enjoy the most. *[1]*

Through innovative technological programs, well-referenced data mining methods, and insights into digital marketing, the present report intends to provide an overview of the process behind the analysis, presents the results and insights you need to be successful in this new era.

In addition to the present report, the following deliverables will be submitted:

- Outcomes presentation to ManyGifts.
- Jupyter Notebook with the code of the entire process.

All files can be accessed in Github:

https://github.com/Debs86/Business_Cases_Projects/tree/main/BC4.

We are excited to take part in this challenge.

2. BUSINESS UNDERSTANDING

2.1. BACKGROUND

ManyGiftsUK (MGUK) is a non-store online retailer with 80 members of staff. The company was established in 1981, with a focus on selling unique all-occasion gifts. The company is based in the UK, but it has customers in many countries around the world. Many of its customers are wholesalers.

In the past, the merchant relied heavily on direct mailing catalogs, and orders were taken over phone calls. Two years ago, the company launched its website and shifted completely online. The company also uses Amazon.co.uk to market and sell its products.

Given the number of possible product choices available, having some extra guidance on these choices can improve the customer experience and lead to an increase in sales. Recommender systems are an essential feature in our digital world, as users are often overwhelmed by choice and need help finding what they are looking for. *[3]* These systems have become a very important part of the retail industries, by providing users personalized recommendations for products or services which hopefully suit their unique taste and needs.

In the past years, MGUK has accumulated a huge amount of data about many customers. Now, they expect to build a recommender system, therefore they have reached D4B.

2.2. BUSINESS OBJECTIVES

The customer's primary objective is to build models to answer the following problems:

- Implement a recommender system to make a recommendation to the customers.
- Cold start problem: offer relevant products to new customers.

2.3. BUSINESS SUCCESS CRITERIA

The main expected outcome will be implementing a recommendation system that appears on the MGUK website home page and offers a wide range of relevant products to both, new and old customers. The success of the proposed task will be evaluated by MGUK management and, if needed, we will go back to the model until we get an outcome that matches their expectation.

Also, another success criteria would be an increase in sales. For example, 35% of Amazon sales come from recommendations. We believe that by implementing a recommender system the sales would increase.

Finally, recommender systems would improve customers satisfaction providing them good recommendations and consequently easy choices.

2.4. SITUATION ASSESSMENT

2.4.1. Inventory of resources

This project was made following the CRISP-DM reference model (Cross Industry Standard Process for Data Mining). CRISP-DM is a standard process built at the end of the '90s and it was built by more than 200 members lead by a consortium of big companies. *CRISP-DM succeeds because it is soundly based on the practical, real-world experience of how people conduct data mining projects.*[2]

This project has the support of MGUK's Management and staff.

On the D4B Consulting side, this project will be conducted by a team of 3 Data Scientists and Business Analysts.

We have been provided by the MGUK team with a dataset with transactions of customers occurring between December 1, 2010, and December 09, 2011. Along with this dataset, its metadata file was also provided.

The main technology used to achieve the objectives of this report was Python. Python is one of the most important and commonly used program languages in data science projects. The main packages for recommender systems are surprise and implicit. As we are dealing with implicit data, the package used in this work was implicit. We also used sklearn to evaluate the models and scipy to create the sparse matrices.

2.4.2. Requirements, assumptions, and constraints

The completion date of the present phase of the project is May 03, 2021, but we expect to continue giving support and helping MGUK to achieve the next goals for the growth of the business.

Recommendation systems are divided into two main categories:

1. Collaborative filtering utilizes the past data of user's interactions as well similar choices made by other users. "Similarity" is measured against the similarity of users. *[3]*
2. Content-based filtering uses the knowledge about each product to recommend items with similar properties. "Similarity" is measured against product attributes. *[3]*

In a collaborative filter, the system can find similarities based on purchase history but also based on demographic data. One of the constraints of this project is the lack of demographic data. It is not a mandatory requirement, but it would improve the quality of the recommendations. Examples of demographic data are genre, age, and location.

In addition, it would also not be possible to find similarities based on product attributes as the data does not provide details of products, only code, and name.

Regarding the type of data, recommender systems make use of two types of data: Explicit or implicit. These concepts will be explained later in the section terminology. On this project, we only have implicit data available.

2.4.3. Risks and contingencies

Table 2.1 identifies a list of risks and contingencies proposed.

Risk	Contingency
Insufficient features of customers' behaviors/ characteristics	Work with remaining features
Insufficient product attributes	Work with remaining features or ask for different variables
Only one year of transaction data and a peak sale in November.	Ask for more observations (transactions)

Table 2.1 - Risks and contingency.

2.4.3.1. Terminology

Business glossary

Recommender systems usually make use of two types of data:

- Explicit data: It is direct feedback from the customer regarding a product or service. From this rating is possible to understand the like or dislike user level in an intuitive way. From explicit data is possible to calculate similarity and provide recommendations according to the users' ratings. Examples: Rating, score, or likes.
- Implicit data: Attached to users' behaviors, implicit data is collected based on an indirect way of user's shows preferences. In this case, the focus is on knowing what the customer has consumed and the confidence we have in whether or not he likes a certain product or service. Examples of implicit data are how long the user spent on a website, how many clicks were made in a webpage by a user, how many times a song was played, and so on.

Data mining glossary

- Recommendation system: Machine learning system which helps users discover new products and services. The system seeks to predict the "rating" or "preference" a user would give to products or services and filter the one with the highest predict rating to the user. There are three main ways:
    - Collaborative filtering (see section 2.4.2)
    - Content-based filtering (see section 2.4.2)
    - Hybrid: combine multiple recommendation techniques.
- AUC - ROC Curve (Area Under Receiver Operating Characteristics) [2]: This is a performance measurement in which ROC is a probability curve and AUC represents the measure of separability. The higher the AUC is, the better the model is at predicting the true positives and negatives. It will also work well for our purposes of ranking recommendations. A greater AUC means we are recommending items that end up being purchased near the top of the list of recommended items.

2.5. Determine Data Mining goals

The data mining goals states project objectives in technical terms:

1. Create a model that will be able to predict good recommendations for the users.

Success criteria: High percentages of AUC.

2. Implement a recommendation system.
   Success criteria: Implement the recommender system on the website homepage.
3. Lead with cold start problem.
   Success criteria: Implement the recommender system on the website homepage and make suggestions to improve the quality of recommendations.

2.6. PROJECT PLAN

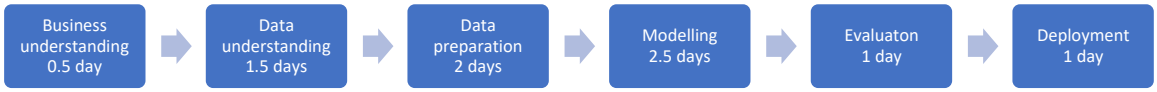


Figure 2.1 - Project's timeline.

Resources wise, for the business understanding we plan to use all the information provided in the kickoff meeting's presentation. For the core stages of the project, we plan to use Python to work the data provided. To present the results, we expect to use Word for the report and Powerpoint for the presentation.

The performance of the model will be directly connected with the quality of the input data. For this reason, we identify the Modelling stage as dependent on the Data preparation stage. During the project, we must go and back between Data preparation and Modelling many times, repeat this iteratively until we get the desired outcome.

For the Modelling stage, we aim to build a matrix factorization model using Alternating Least Squares algorithm. We opted for this model because it presented the best results compared with other algorithms. Further details will be presented in section 3.3 – Modelling and evaluation. The model evaluation will be made using Area Under the curve (AUC).

3. PREDICTIVE ANALYSIS

In this section, we go through the process of understanding and preparing the data for modeling, the modeling itself, the different algorithms used, and, finally, the evaluation of the results.

3.1. DATA UNDERSTANDING

At this stage, we analyzed the dataset to understand its potential and limitations. First, we looked at the data in the excel file to check inconsistencies. The data understanding step is good to understand what variables are in the dataset, what they mean, the number of variables (8 features, from which 6 categorical and 2 numerical as shown in Table 3.1) and observations (541909 purchase transactions), if there are inconsistencies, if there are missing values (135.080 *CustomerID* and 1.454 *Description*) and/or duplicated values (10.147).

We have also looked at the metadata file provided to understand the meaning of each feature to understand their relevancy in the project.

Numeric	Categorical
Quantity, UnitPrice	*InvoiceNo, StockCode, Description, InvoiceDate, CustomerID, Country*

Table 3.1 - Numerical and categorical features.

The dataset is composed of 541,909 transactions, 25,900 invoices (cancels and no cancels), 3,959 StockCodes, 4,373 customers from 38 countries.

Going into more details, September to November are the months where there is a higher volume of sales (Figure 3.1). Most customers have only one invoice and few of them bought more than 15 times (Figure 3.2).

The cancelations transactions represent 1,7% of the total transactions. Also, transactions without a *CustomerID* represent almost 25% of the total transactions.

More than 80% of the sales volume coming from United Kingdom customers (Figure 3.3).

Figure 3.1 – Volume sales by month

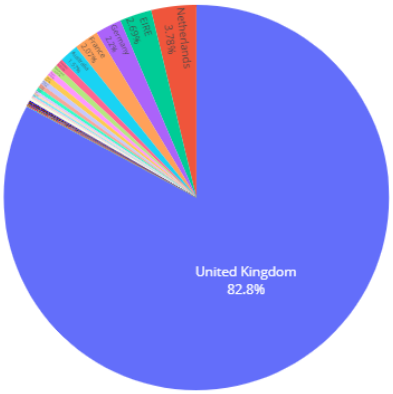


Figure 3.3 – Sales volumes by country

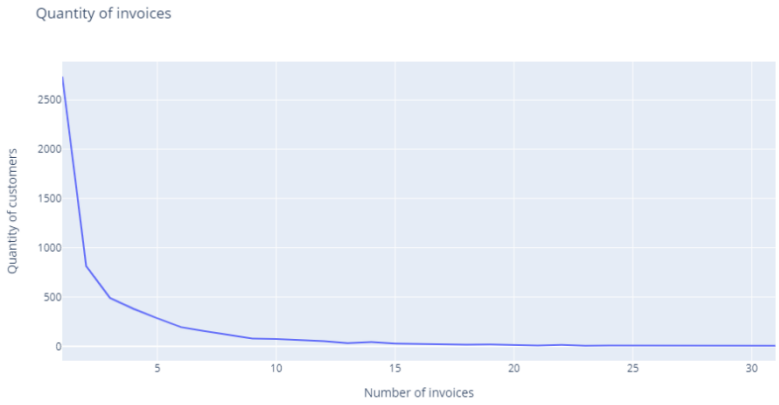


Figure 3.2 – Number of invoices vs. Number of customers

3.2. DATA PREPARATION

At this stage, the dataset should be prepared to ensure a good quality of the data to the algorithms achieve good performance.

On the first step of data preparation, we removed duplicate observations. Next, we eliminated all observations whose *StockCode* does not fit with the described format on the metadata ("Nominal, a 5-digit integral number uniquely assigned to each distinct product"). Therefore, we only keep the observations whose *StockCode* is 5 numbers, and the only exception that was made is that we also keep *StockCode* with 5 numbers followed by one letter. We also remove the remaining observations with a price equal to 0. Regarding these last 2 inconsistencies (stock code and unit price), we could

verify that most of the transactions are about adjustments, post and amazon fees, damages, and others that could not be used by a recommender system. We removed 2% of the data until this step.

The next step was feature engineering which we built 1 new feature, for data analysis purpose:

New variables	Description
Month_Year	*Month and Year related to that transaction.*

Table 3.2 - New variables.

After these changes, the missing descriptions were already dropped. For the missing *CustomerID*, we applied two approaches: For a better understanding of the data, we create a fake *CustomerID* for each observation, according to the invoice number. Observations with the same *InvoiceNo* have the same *CustomerID*. The algorithms for the recommender system work with datasets in a matrix format. For the recommender system purpose, we drop all the observations without CustomerID. Regarding the recommendations to them, we are going to make recommendations in the same way we are going to make to new customers as we do not have a track of their transactions.

3.3. MODELING AND EVALUATION

In this step, we are going to fit the algorithms for the recommender system.

The first stage of this process was to reduce the sparsity of the data. As we have a big dataset, for a better performance of the model, we must have only transactions that matter. Transactions related to customers or items without significant history were removed from the data.

In the next stage, we created a new dataset only with the columns that the algorithm will use: *CustomerID*, *StockCode,* and *Quantity*, grouped by *Quantity*. All observations where the sum of quantity was 0 were removed from the dataset.

As we stated before, the data used in this work is implicit. Also, we are going to use a collaborative filtering approach as we are dealing with past data of purchases of the customers. To have a better performance of the algorithm we still need to solve the problem of our data having many different dimensions, but we need to compile them in few dimensions. In other words, the many clicks of one user in a website just express a couple of tastes, or any purchases of an item express only some tastes. To solve that, there is a technique called matrix factorization that works to reduce the data dimensionality transforming the original data "all users by all items" matrix into two small matrices much smaller that represents "all items by some taste dimensions" and "all users by some taste dimensions". These dimensions are called latent or hidden features and it is learned from the data. This dimensionality reduction makes the work much more computationally efficient, brings better results because it is working in a more compact space and it also allows us to find connections between users who have no specific items in common but share common tastes*. [4]*

Therefore, the third stage of this step was applying some transformations on the data to have 2 matrices: one for fitting the model and the other will be used to make recommendations. The column quantity was used as a measure of the level of confidence. If the customers buy a large quantity of a product, it means that he really liked that item.

The last stage before fit the model was to split the data into train and test datasets. In this case, the split is done by making a "mask" of the data. The test size contains all original data, and on the training set, 20% of the data is replaced by zeros.

Now that we have our data in the required input format, we applied 3 different algorithms: Alternating Least Squares (ALS), Bayesian Personalized Ranking (BPR), and Logistic Matrix Factorization (LMF) with the following parameters:

Parameters	Description	Values
Alpha	*Used to reflect the importance of confidence [5]*	*15*
Factors	*Number of latent factors.*	*20*
Regulation	*Regularization to avoid overfitting*	*0.1*
Iterations	*Number of iterations to fit the model*	*50*

To measure the performance of each algorithm, we used a function that calculates AUC (explained in section 2.4.3.1) for each user that had at least one item masked on the train set and later calculated the average AUC of all users. This function also allows a comparison with the AUC of the most popular items for all users.

The results are presented in Figure 3.4.

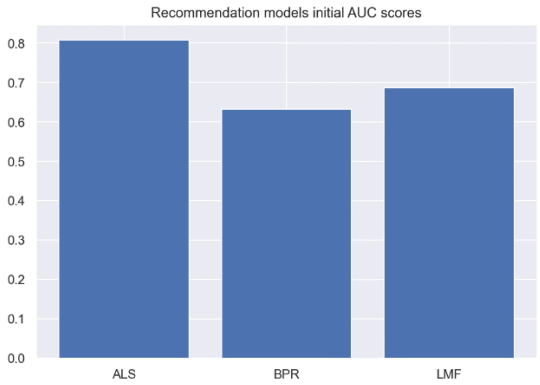
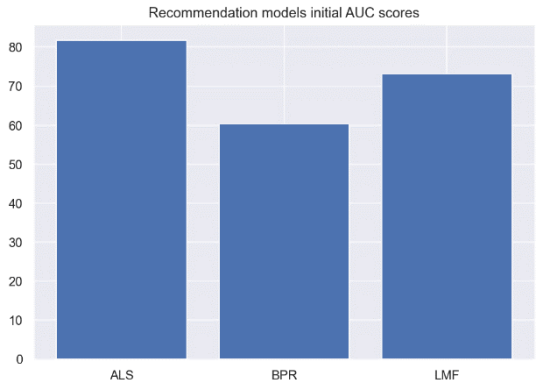


Figure 3.4 - Models score.

The next stage was trying to improve the scores by tuning the parameters. We create a function to tunning the parameters and show the parameters that should be used for the best results. The details with the parameters tuned can be found in the notebook. The table with the best parameters of each model and the graphic with the results are presented below:



Parameter	ALS	BPR	LMF
Alpha	*15*	*20*	40
Factors	*50*	*40*	60
Regulation	*0.1*	*0.01*	0.05
Iterations	*40*	*60*	50

Table 3.3 - Tuning RF parameters and results.

The model with the best performance was Alternative Least Square. Therefore, the recommendations will be made using this algorithm.

Finally, we created 2 functions to make sure the good performance of the algorithm by checking its recommendations. The first function is to find similar items given an item. The second function is to make recommendations to a customer and compare the recommendations with the most items bought by that customer. The results can be found in the notebook.

4. RESULTS EVALUATION

The model developed by B4C reached good values and will certainly help Many Gifts the UK to implement recommendations to your customers and help them to make better choices.

The recommender model predicts recommendations correctly 81,7% of the time. It will probably lead to an increase in sales, which was stated as one of the business goals.

The next step will be implementing the recommender system on the homepage of your website.

Regarding the cold start problem, in this first stage, we created a function that recommends items for new users according to their country. The recommendations will be done according to the last two months' purchases of other users based on the same country of the new customer. It will also be implemented on the MGUK website.

5. DEPLOYMENT AND MAINTENANCE PLANS

We understand that the process of implementing the recommendation system is very critical to your business. Retails shops market is very dynamic and every day, there are new purchases on the system. So, we understand that it would be very important to implement the analysis provided integrated with the purchase system. Integrating both, the recommendations will be updated on a daily basis and it will improve the customers' choices.

Another important point is that we would like to make some suggestions. We suggest to MGUK to improve the items data. It would be important if we have more attributes regarding items to facilitate the recommendation based on the similarity of items. The second suggestion is about data collection from the customers. As we stated before, from explicit data we knew if the customer liked or disliked an item. It would be very interesting if you could get ratings from your customers about the items purchased. Another interesting information about a customer to collect is regarding their preferences: kind of items they are interested in for example. It would also help in the cold start problem.

We are going to monitor the performance of the model after it is implemented for some months and make some adjustments if necessary. Also, if the suggestions above were implemented, we could improve the recommendation models. But it is not mandatory.

We also suggest updating the model from time to time as the behavior of the customers, and even the market trends, can change, reducing the model performance.

Lastly, considering this model was built with only a few thousands of observations, we consider that re-training the model, when new data is available, would potentially improve its quality.

6. CONCLUSIONS

As state in section 2.2 – Business objectives, the main objective of this project was to implement a predictive model to make recommendations for old and new customers.

The model developed by B4C reached good values on many important measures such as 81,7% of AUC, which means the model will predict the right recommendations 81,7% of the time. It would certainly help customers make better choices and probably increase sales volume.

In addition, we set some risks on this project. One of these risks is the model performance, as we have been working with only one year of the data. We implemented some actions to reduce this risk, but the model will have an improvement margin if we get additional datasets to test its performance.

We hope ManyGiftsUK is satisfied with D4B work and we can continue working together.

7. REFERENCES

[1] Hu, Y., Koren, Y., Volinsky, C.. Collaborative Filtering for Implicit Feedback Datasets.

[2] Chapman, P, Clinton, J, Kerber, R., Khabaza, T., Reinartz, T, Shearer, C. & Wirth, R. (2000). *CRISP-DM 1.0*, CRISP-DM consortium

[3] Shetty, B., 2019, AN IN-DEPTH GUIDE TO HOW RECOMMENDER SYSTEMS WORK, viewed 21 April 2021, < https://builtin.com/data-science/recommender-systems >

[4] Victor,2017, ALS Implicit Collaborative Filtering, viewed on 24 April 2021, <https://medium.com/radon-dev/als-implicit-collaborative-filtering-5ed653ba39fe>

[5] Ting, Yao, 2019, Implicit Feedback Recommendation System (II) — Collaborative Filtering, viewed on 24 April 2021, <https://medium.com/@teddywang0202/implicit-feedback-recommendation-system-ii-collaborative-filtering-27be600197f1>

[6] Frederickson, B.,2021, Implicit Documentation. Release 0.4.4, viewed 02 May 2021, <https://readthedocs.org/projects/implicit/downloads/pdf/latest/>

[7] Mohamed M., Khafagy M., Ibrahim M.(2020) Two Recommendation System Algorithms Used SVD And Association Rule On Implicit And Explicit Data Sets, viewed 22 April 2021, <http://www.ijstr.org/final-print/jan2020/Two-Recommendation-System-Algorithms-Used-Svd-And-Association-Rule-On-Implicit-And-Explicit-Data-Sets.pdf>

[8] Ting, Yao, 2019, Implicit Feedback Recommendation System (I) – Intro and Datasets EDA, viewed 22 April 2021, <https://medium.com/@teddywang0202/implicit-feedback-recommendation-system-i-intro-and-datasets-eda-eda16764602a>