Big Data Analytics

Lab Project

Master in Artificial Intelligence

Universitat Rovira i Virgili

Mario Lozano Cortés - mario.lozano.cortes@estudiantat.upc.edu

Index

This document does not contain any additional information *compared to the associated jupyter notebook. The information is simply shown in document format for ease of reference and readability. The full version of the analyses can be found in the associated notebook.*

1. Introduction

This project's goal is to study and **analyse the existence of differences in the behaviour of men and women in the use of the dating app Tinder**. As described by *Wikipedia*, "*Tinder is an online dating and geosocial networking application. In Tinder, users "swipe right" to like or "swipe left" to dislike other users' profiles, which include their photos, a short bio, and a list of their interests. Tinder uses a "double opt-in" system where both users must like each other before they can exchange messages*".

The application makes use of a **freemium model**. Thus, the basic functionalities are free to use, while the advanced functionalities require a fee to be used. Some of the premium features are unlimited likes (100 in the basic version), super likes with messages, discovering who has liked your profile or highlighting your likes to other users. In this way, **the app must know how its users use each of the features in order to achieve the best possible business model**. Immediately, **we can distinguish two groups in the application, men and women** (as organised by the application itself). Thus, one of the first questions that arise is whether there are differences in the use of the application between the two groups. **This question is especially relevant for a correct segmentation of the market to offer an appropriate set of features to both men and women**.

In order to carry out the proposed analysis, **the following questions are defined** to be resolved throughout the analysis:

Questions:

- Who is more selective? Passes vs Likes by sex
- Who receives the more attention? Matches by sex
- Who uses the app the most? App opens by sex
- Who is most willing to pay for a subscription? Number of times the likes limit is reached per sex
- Who talks the most? Messaging behaviour by sex
- What is the minimum, mean and maximum percentage of one message-conversations for every sex? What about the number of ghostings after the initial message?
- Who uses more Instagram by sex?
- More used emojis by sex

2. Data collection and storage

This section details how the data was collected and how it was decided to store the data to enable analysis.

2.1 Data collection

swipestats.io provided the data at no cost for academic purposes. swipestats.io is an anonymous data visualization and comparison web service that seeks to help people understand their Tinder data. For using the service, a person must download its data from the Tinder app and upload it to swipestats to get interesting insights about their behaviour in the app.

The dataset consists of a single JSON file (560MB) and none a single description or explanation of the data is given. Thus, the most crucial task of the analysis is to understand the data at hand to be able to get valuable information from it.

2.2 Data storage

Relational DBs are based on the relational model, which organizes data into tables with rows and columns with minimal data repetition. Each row represents a record, and each column represents a field in the record. Nevertheless, the relationships between the tables as well as the column data types need to be defined prior to the use of the database. Some examples of relational databases include MySQL, Oracle, and Microsoft SQL Server.

NoSQL databases, on the other hand, are non-relational databases that are designed to handle large amounts of data that is structured, semi-structured, or unstructured. NoSQL databases are often used for storing large volumes of data that do not fit well into the tabular structure of a traditional relational database. Some examples of NoSQL databases include MongoDB and Apache CouchDB.

Thus, we can state several key differences between relational and NoSQL databases:

- **Data structure**: Relational databases use a tabular structure to store data, while NoSQL databases can use a variety of data structures, such as key-value pairs, documents, and graphs.
- **Query language**: Relational databases use SQL to manipulate and query data, while NoSQL databases may use a variety of query languages, such as MongoDB Query Language.
- **Flexibility:** NoSQL databases are generally more flexible than relational databases, as they can store data in a variety of formats and structures. This makes them well-suited for handling semi-structured and unstructured data.

In summary, **relational databases are good for structured data and support complex queries, while NoSQL databases are better for large volumes of unstructured data. Because of all these reasons, the choice for the task at hand is NoSQL and MongoDB since the data is semi-structured and given in a JSON, not following a tabular structure**.

3. Data description

The following section seeks to describe the data. It should be considered that **not a single description or explanation was given**. Hence, **an analysis should be driven in order to perfectly understand the data at hand, allowing the posterior analysis of the data and the extraction of valuable information**. Thus, the data is analysed field by field and each one gets described thanks to the experiments carried out and the actual use of the application to discover the concrete meaning of each one.

The keys found in the dataset are: *'_id', '___v', 'appOpens', 'conversations', 'conversationsMeta', 'matches', 'messages', 'messagesReceived', 'messagesSent', 'swipeLikes', 'swipePasses', 'swipes', 'user' and 'userId*'.

Description of data found in each key:

- **'_id'** is a unique and anonymous identifier for each instance of the dataset. Moreover, ___v is a versionKey that contains information about the internal revision of the document so it's not remarkable for the current analysis.

- **'appOpens'** refers to the number of times a user opens the app by date. The information is stored in a dictionary where the key is the date.

- **'conversations'** refers to messages sent by the user considered. The information is stored in a list of dictionaries where every dictionary stores a conversation with a match.

- **'conversationsMeta'** refers to the metadata of the messages sent by the user considered. The information is stored in a dictionary where the following data is found:

    - **nrOfConversations**: Total number of conversations held
    - **longestConversation**: Length of the longest conversation
    - **longestConversationInDays**: Length of the longest conversation considering the days passed since the first and last messages.
    - **averageConversationLength**: Average length of the conversations held
    - **averageConversationLengthInDays**: Average length of the conversations considering the days passed since the first and last messages.
    - **medianConversationLength**: Median length of the conversations held
    - **medianConversationLengthInDays**: Median length of the conversation considering the days passed since the first and last messages.
    - **nrOfOneMessageConversations**: Total number of conversations consisting of just one message

    - **percentOfOneMessageConversations**: Percentage of one message conversations as *nrOfOneMessageConversations/nrOfConversations*.
    - **nrOfGhostingsAfterInitialMessage**: Total number of times where a first message received (a match is starting the conversation) is not replied to by the user.

- **'matches'** refers to the number of total matches a user gets by date. The information is stored in a dictionary where the key is the date.

- **'messages'** refers to the number of total messages a user sends or receives by date. The information is stored in a dictionary where the keys are 'sent' and 'received'. In a similar way, each key refers to a dictionary where the key is the date and the value of the number of messages.

- **'messagesReceived', 'messagesSent'** and 'messages' contain the same information and thus, 'messagesReceived', 'messagesSent' can be deleted since it is redundant information.

- **'swipes'** refers to the number of total swipes a user performs. The information is stored in a dictionary where the keys are 'likes' and 'passes', each one referring to the swipes for people the user likes and for people the user doesn't like respectively. Similarly, each key refers to a dictionary where the key is the date and the value of the number of swipes.

- **'swipeLikes', 'swipePasses'** and 'swipes' contain the same information and thus, 'swipeLikes', 'swipePasses' can be deleted since it is redundant information.

- **'user'** refers to the personal data of the user considered. The information is stored in a dictionary where the following data is found:

    - **birthdate**
    - **ageFilterMin**: Minimum age parameter for profiles displayed to the user.
    - **ageFilterMax**: Maximum age parameter for profiles displayed to the user.
    - **createDate**: Profile creation date
    - **education**: Whether the profile has or has not high school or college education.
    - **gender**: M and F as possible values
    - **interestedIn**: Gender the profile is interested in. M, F or M and F.
    - **genderFilter**: Gender parameter for profiles displayed to the user.
    - **instagram**: Whether the profile links to an Instagram profile or not
    - **spotify**: Whether the profile links to a Spotify profile or not

- **jobs**: Dictionary with job information containing: *companyDisplayed* (whether the company is displayed or not), *titleDisplayed* (whether the job title is displayed or not) and title.
- **educationLevel**: Whether the profile has or has not high school or college education.
- **schools**: Dictionary with school information containing: displayed (whether the school name is displayed or not), and name.

- **'_id'** and 'userId' store the same information. Thus, 'userId' can be deleted since it is redundant information.

4. Data exploration and missing values

The exploratory data analysis phase is critical to get to know the data the project is working with. Hence, statistics and visualization of the most relevant features are created to get a general sense of the data. Some of the statistics and visualizations include **mean, median, standard deviation, minimum, maximum and boxplots**. Moreover, these techniques are also going to be used for outlier detection.

On the other hand, the question of missing values has as goal the identification of how missing values are indicated in the dataset and thus, how MongoDB is treating them. The vast majority of missing values are empty chains of text ('') and Mongo's null value is not used in the dataset.

The boxplots generated are shown below:

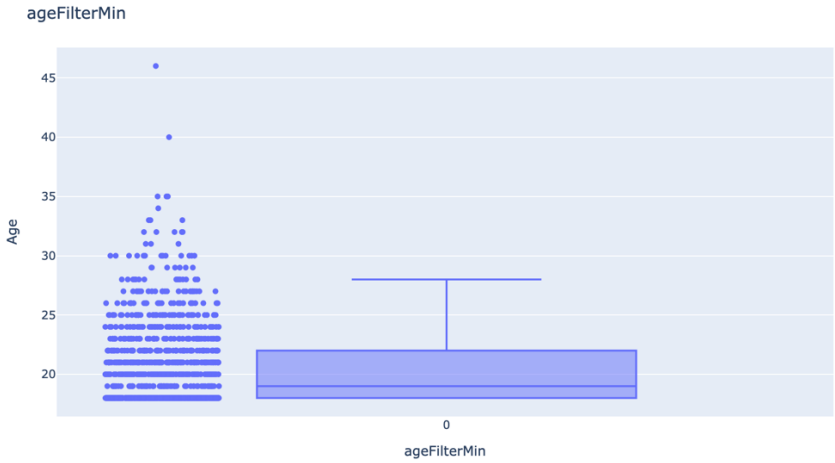

***Figure 1**: Age filter min boxplot*

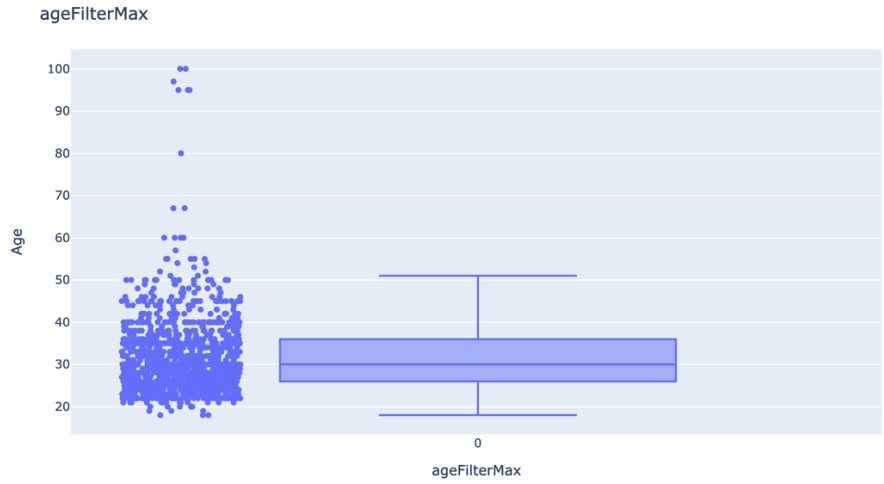

***Figure 2**: Age filter max boxplot*

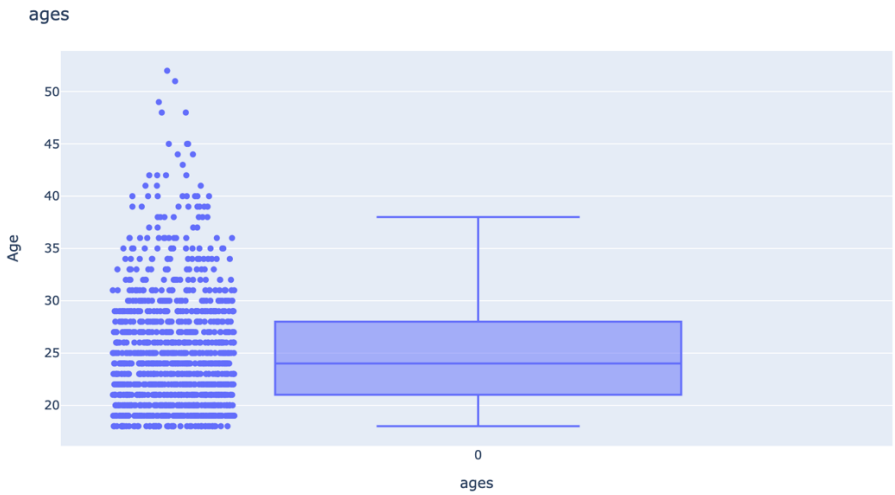

***Figure 3**: Age boxplot*

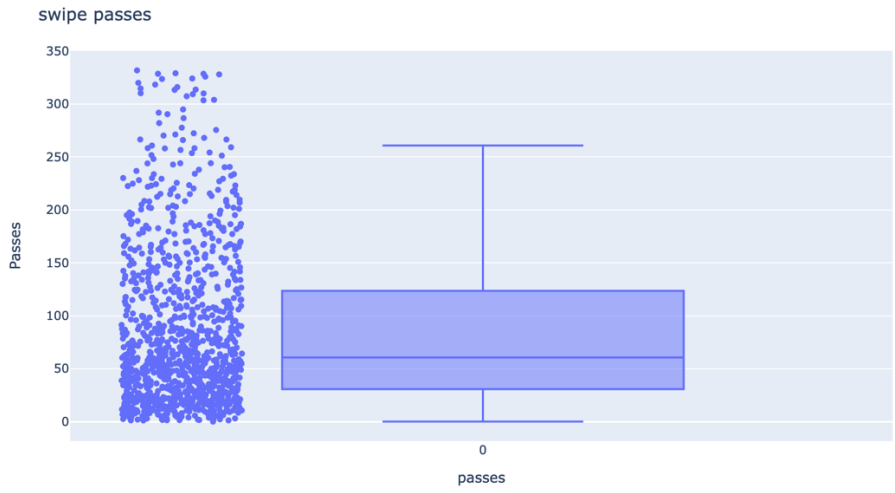

***Figure 4**: Swipe passes boxplot*

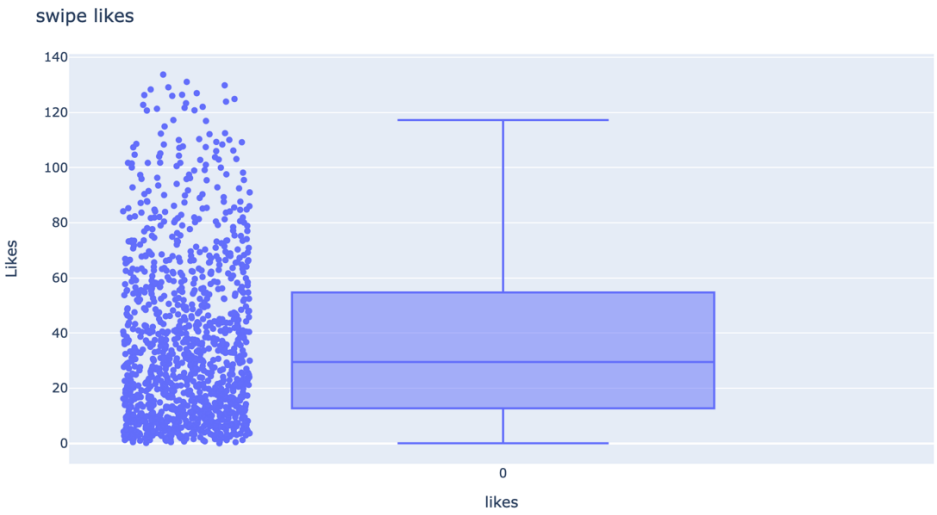

***Figure 5**: Likes boxplot*

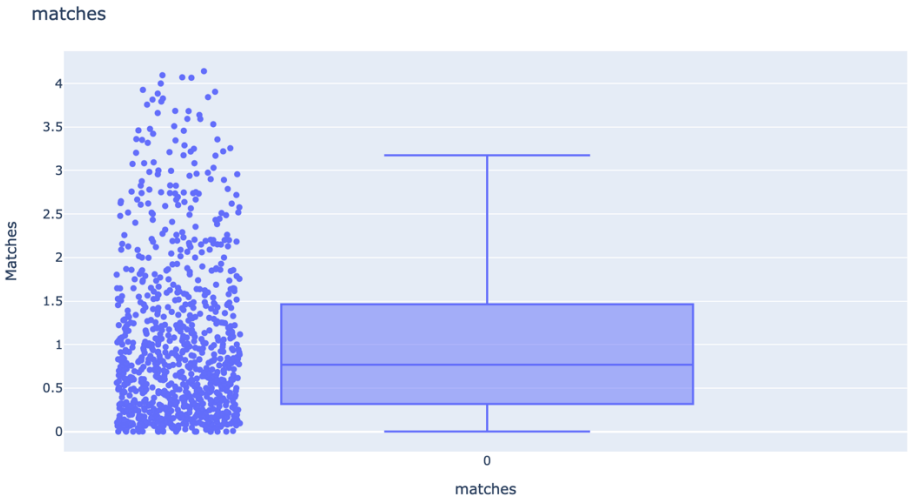

***Figure 6**: Matches boxplot*

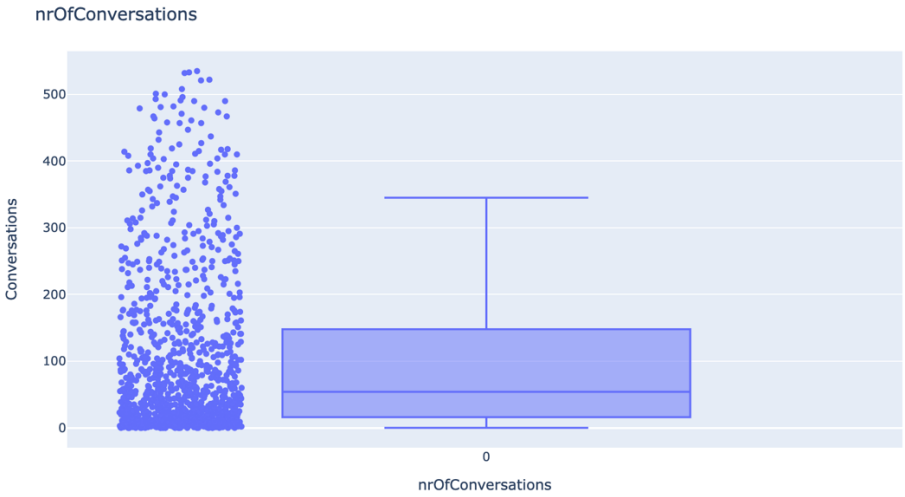

***Figure 7**: Number of conversations boxplot*

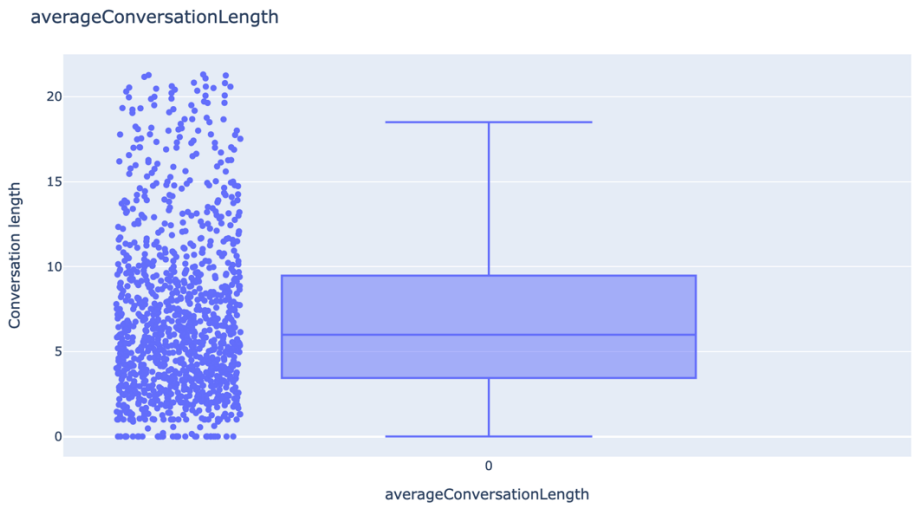

***Figure 8**: Average conversation length boxplot*

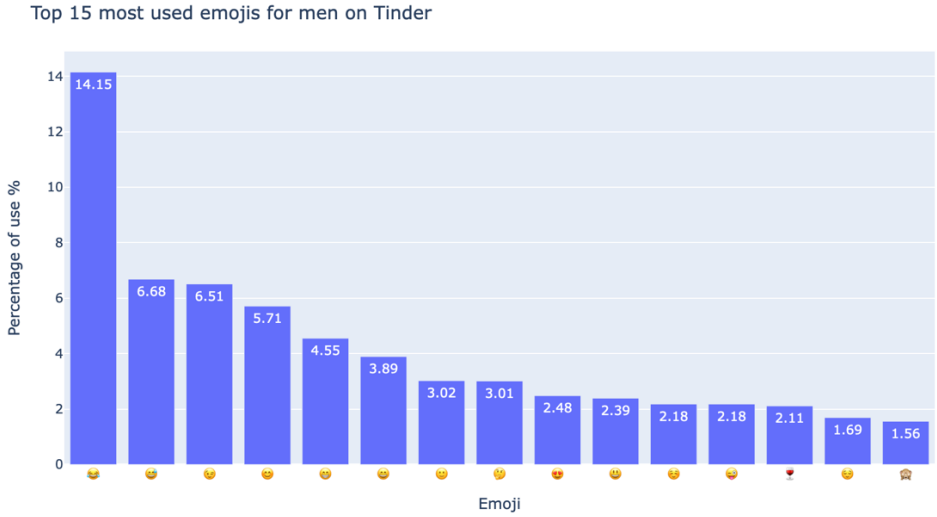

***Figure 9**: Top 15 most used emojis for men*

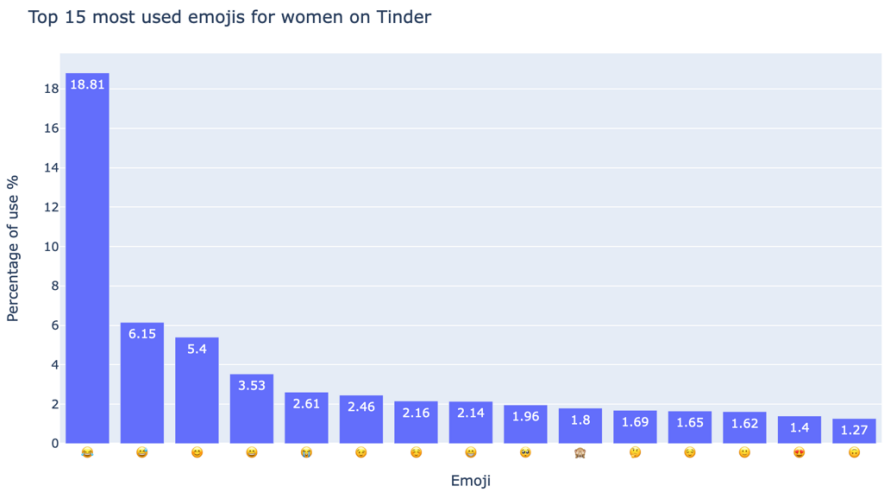

***Figure 10**: Top 15 most used emojis for women*

5. Analysis and conclusions

Throughout this study, the differences between the two predominant groups on Tinder, men, and women, have been analysed. To do so, data provided by swipestats.io from real users has been used, and an important descriptive and cleaning task has been carried out on the data, as it did not have any type of explanation and it contained redundancies in certain fields. In this way, the data exploration process has been carried out on each of the fields that made up the dataset and it has been managed and stored in a NoSQL database, specifically MongoDB.

Among the insights discovered, the following stand out:

- **Men give many more likes than women (x3.1 times).**
- **Women ignore many more profiles than men (x1.4 times).**
- **Women get many more matches than men (x2 times).**
- **There is no significant difference between the number of times men and women log on to the application per day.**
- **Men reach the daily likes limit many more times than women (x5.1 times).**
- **There is no significant difference in the length of conversations between men and women.**
- **Men tend to have a higher number of worse conversations as their percentage of single-message conversations is much higher than that of women (x1.4 times).**
- **Women do much more ghosting than men (x2.7 times).**
- **Women tend to link their Instagram profiles slightly more than men (26.67% women vs 21.45% men).**

There are no major differences between the emojis used by men and women, with the two preferred emojis being the same for both sexes. The emojis used by men but not by women (in the Top 15 most common) are 😃, 😜, 🍷. The emojis used by women but not by men (in the Top 15 most common) are 😭, 🥺, 🙃.

Therefore, it can be deduced that there are significant differences between men's and women's use of Tinder. **The data obtained support the hypothesis that women have a more selective pattern of dating behaviour than men**. This information is drawn from the fact that there are no significant differences in terms of the number of times the application is used per day, but there are significant differences in the ratio of profiles that are valid for a possible date (like + match + long enough conversation) between men and women.

Thus, **the assessment of the project is highly positive**, given that the proposed objectives have been met satisfactorily. A clear question has been defined, the necessary data have been obtained, appropriate management, exploration and cleaning have been proposed and a

complete analysis has been carried out, which has provided valuable answers to the questions defined. Thus, **the extracted information is highly relevant for making business decisions concerning the freemium model used in the app**.