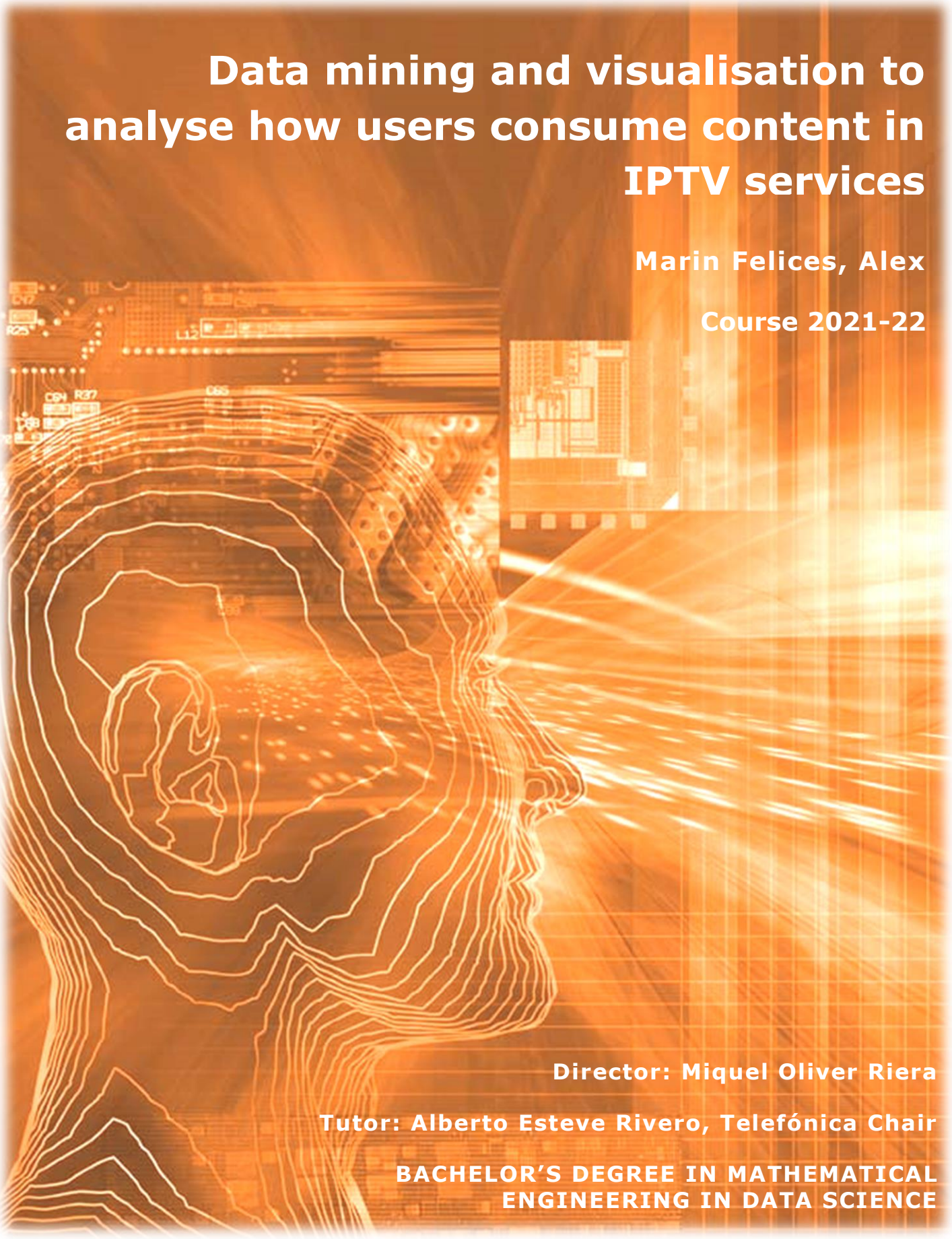
Data mining and visualisation to analyse how users consume content in IPTV services

Marin Felices, Alex

Course 2021-22



Director: Miquel Oliver Riera

Tutor: Alberto Esteve Rivero, Telefónica Chair

BACHELOR'S DEGREE IN MATHEMATICAL ENGINEERING IN DATA SCIENCE

upf. Universitat Pompeu Fabra Barcelona
Escola d'Enginyeria

Treball de Fi de Grau

Data mining and visualisation to analyse how users consume content in IPTV services

TREBALL FI DE GRAU BY

Alex Marin Felices

Director: Miquel Oliver Riera

Tutor: Alberto Esteve Rivero, Telefónica Chair

Bachelor's degree in Mathematical Engineering in Data Science

Course 2021-2022

Universitat Pompeu Fabra Barcelona
Escola d'Enginyeria

To my family and friends, who always support me and make me grow.

Acknowledgements

I would like to take this space to thank my family and friends for all the support they have given me throughout the whole project. Without them this would not have been possible. Also, to my tutor Miquel, for guiding me and making my project better with his advice.

Abstract

The growing number of streaming services like IPTV platforms, are now providing an increasingly diverse range of material. As these services expand in popularity and extra content is being added, users are suffering more difficulties and spending too much time during the selection process. Users have negative experiences as a result of this.

This study's main goals are to analyse user behaviour and understand what the user preferences are when navigating through an IPTV platform. The project is a collaboration with Telefónica and some of their employees who assisted and guided us throughout the research. Additionally, real user sessions data from their Movistar+ Chile platform was shared with us so we could perform the analysis. Both, data visualisation and data mining were used in order to unveil patterns in user behaviour.

Resum

El nombre creixent de serveis de streaming com les plataformes IPTV, ofereixen ara una gamma de material cada vegada més diversa. A mesura que aquests serveis augmenten en popularitat i s'hi afegeix contingut addicional, els usuaris pateixen més dificultats i passen massa temps durant el procés de selecció. Els usuaris tenen experiències negatives com a resultat d'això.

Els objectius principals d'aquest estudi són analitzar el comportament dels usuaris i entendre quines són les preferències de l'usuari quan navega per una plataforma IPTV. El projecte és una col·laboració amb Telefónica i alguns dels seus empleats que ens van assistir i guiar durant tota la recerca. A més, van compartir amb nosaltres les dades reals de sessions d'usuaris de la seva plataforma Movistar+ Chile perquè poguéssim realitzar l'anàlisi. Es van utilitzar tant la visualització de dades com la mineria de dades per tal de revelar patrons de comportament dels usuaris.

Resumen

El creciente número de servicios de streaming como las plataformas de IPTV, ahora brindan una gama cada vez más diversa de material. A medida que estos servicios aumentan en popularidad y se agrega contenido adicional, los usuarios sufren más dificultades y pasan demasiado tiempo durante el proceso de selección. Los usuarios tienen experiencias negativas a ver a raíz de esto.

Los objetivos principales de este estudio son analizar el comportamiento de los usuarios y comprender cuáles son las preferencias de los usuarios cuando navegan a través de una plataforma de IPTV. El proyecto es una colaboración con Telefónica y algunos de sus empleados que nos ayudaron y guiaron a lo largo de la investigación. Además, compartieron con nosotros datos de sesiones de usuarios reales de su plataforma Movistar+ Chile para que pudiéramos realizar el análisis. Tanto la visualización de datos como la extracción de datos se utilizaron para descubrir patrones en el comportamiento de los usuarios.

Table of Contents

List of figures

List of tables

1. INTRODUCTION

1.1 Challenge definition

Internet Protocol Television (IPTV) has grown in popularity among users ever since smart TVs and highly sophisticated Internet video streaming services have become more popular. However, as more online content becomes accessible for viewers to pick from, the long-standing problem of choosing what to watch not only persists, but increases [1,4,5]. IPTV service providers' recommendations lag behind Over-the-top (OTT) content providers such as Netflix, HBO, Amazon Prime Video, Hulu, etc; which are equipped with powerful recommender systems (RS). OTT providers' content is based on Video on demand (VOD), which allows the user to access specific content, at the time they request it, by viewing it online on their device. IPTV providers, on the other hand, continue to use the Electronic Program Guide (EPG) or Live content. Historically, EPG provides a vast list of channels, but not much RS adoption in IPTV has been observed so far. This can become frustrating and troublesome for users since the list usually contains hundreds of channels with varying types of content [1,4,5].

Nowadays we are constantly interacting with RSs. When we search for information, clothes or news on Google, for items on Amazon, movies on Netflix or music on Spotify, they all are using them to rank what we would like the most from their catalogue, or for the thing we are most interested about at that certain moment. These systems take as input data generated by us and other users to output the most accurate recommendations possible [2]. State of-the-art machine learning algorithms have been developed to efficiently create large-scale suggestions to millions of clients, ranging from content-based (CB) RS or collaborative filtering (CF), to more current and powerful like deep neural networks (DNN). These algorithms try to solve various problems such as group viewing, the cold-start problem, cross-system platforms, ethics, large-scale reproducibility, and a very big etc.

Nevertheless, in this study, the focus is not going to be on developing a recommender system but rather finding a deeper understanding of how users behave when watching content on an IPTV platform. Even if it is on wired devices like TVs and Smart TVs, or wireless ones such as Smartphones and Tablets, the goal is to find typical behaviours within the users. We hope to find trends within the data that indicate which types of content is watched regularly by a certain user, whether it is a program in particular or a wider theme genre. Moreover, we desire to find temporal tendencies that show certain hours or days where there tend to be more sessions and if we also find differences in the type of content watched.

For this study, we have been given access to user sessions from the IPTV platform from Telefónica, Movistar+ Latin America. Precisely, our focus is going to be put into content available in their platform provider in Chile and sessions from real users. We have also collaborated and received some help from different employees from Telefónica. They have guided this study and provided feedback and their opinion and expertise throughout the project.

1.2 Objectives of the TFG

The objectives of this project are:

1. To validate the initial hypothesis that a high percentage of the consumption made by users in the video service is based on routine consumption by the client and, therefore, it can be identified and anticipated for future recommendations, with the aim of facilitating the content to the user for its consumption. Being able to establish, on a given sample of users, in what percentage this first hypothesis is fulfilled.

2. To identify user consumption patterns in a theoretical model of recommendation that allows us to anticipate said recommendations. The consumption pattern can be defined upon the type of content seen along the week and for the different timeframes.

3. If the hypothesis of the routine is confirmed given a sample of users, the next step is to clean the data appropriately so that it allows to create an exploratory data analysis (EDA) to:

    - Identify bands of higher utilisation of users.
    - Observe whether user consumption is based on live or Video on Demand (VOD) content
    - Unmask hidden patterns within the data such as: most watched themes/subthemes, prime times, most frequent day of week, etc.
    - Find relationships between subthemes by using Apriori algorithm
    - Create user clusterings based on start time, content theme, subtheme and other segmentation variables.
    - Find differences in content consumption and behaviour based on the type of device.

1.3 Motivation

The growing number of streaming services, both live and VOD, are now providing an increasingly diverse range of virtual content. Users are having more difficulty and spending too much time during the selection process as these services and content expand in popularity. Users have negative experiences when deciding what to start watching as a result of this. If you have ever used one of these services, you will probably know how frustrating it can be to search and search for something to watch and eventually run out of time or end up watching something you do not enjoy.

In Movistar+ Chile, the users do not have the option to log in into separate users, so all members of a family or a group of friends have to use the same global profile. Therefore, this platform suffers from the group viewing problem. This is a big problem in the literacy of RS and many techniques and algorithms are being developed. As it was mentioned before, creating and perfecting a RS is not the goal of this project.

This solution's major purpose is to understand users' behaviour during their watching sessions in order to reduce the amount of time they spend making these decisions. As a result, techniques for uncovering, analysing and interpreting user habits are presented. By having a deeper understanding of these, the aim is to enhance the quality of recommendations both inside and outside the using session and increase the hit rate success of the RS. If RS developers are able to differentiate when a user watches a certain type of content and when another type, it can improve how many times the user clicks in one of the few options that can be recommended to him. The motivation is to improve the amount of times one of the recommendations is selected by the user.

In order to have this deep understanding of group users' behaviour we are going to analyse profoundly the data utilising measures such as:

- viewing platform parameters e.g., selection time, viewing sequence (total, partial, serial), device type used (web, PC, smart TV, Internet Protocol Television (IPTV) systems, tablets and smartphones)
- viewing behaviour e.g., start time and end time, frequent content themes and subthemes, number of sessions, average session duration.
- context parameters e.g., individual viewing, as a couple, as a group of friends or as a family; day of the week (workday, weekend), country from where the content is watched

In addition, a possible solution could include comparisons between various recommendation algorithms and diverse ways of displaying them inside the various devices available. In future work, the metrics could then be put to the test using the most prevalent Machine Learning (ML) techniques in order to verify the correctness of these.

2. STATE OF-THE-ART

2.1 IPTV platforms

Internet Protocol Television (IPTV) is a growing multimedia service that uses proprietary IP-based networks to transmit television, video, audio, and other interactive content [5]. It uses the Internet protocol (IP) to deliver information packets across a Broadband IP network. Internet service providers, such as Telefónica, oversee and control this platform. These are the same companies who send the content through their IP networks. These businesses can ensure service quality by utilising their private networks; the content requested by the user can be streamed live or downloaded from video servers situated throughout the provider's network [11].

Implementing IPTV technology with fibre optic has been one of the solution mechanisms to overcome existing limitations, and this allowed for a technical protocol that can be distributed through different platforms including cable, satellite, or Digital Terrestrial Television (DTT) [11].

Nowadays, the amount of content offered in these services is huge and makes it really difficult for the users to choose what they should watch. As a consequence, RSs play a major role when defining the success of each IPTV. Although in this study we will not design a RS, this is the issue that this study tries to tackle with a deeper understanding of how users behave, so RS can be improved as a result.

Despite the fact that there are an increasingly big number of IPTV providers available to users, and their huge popularity, it seems like their biggest competition does not come from other IPTV networks rather from the Over-the-Top (OTT) platforms such as Netflix, HBO, etc [1,4] which also tend to have really good and advanced RS. One of the main differences between these 2 types of providers, is the type of content that is offered. While the OTT TV platforms only provide the VOD service, there are two main types of content available inside an IPTV platform.

2.1.1 Electronic Program Guide (EPG)

The Electronic Program Guide is the source from where the users can get the most amount of content inside an IPTV. EPGs provide a long list of channels embedded in a hierarchical or multi-layer menu. It is different from online movies and on-demand content because they are mainly organised in program sequences whose broadcasting schedules are mostly fixed and periodic, e.g., daily news and TV series. Some programs are only broadcast once without fixed schedules, e.g., live sports broadcast and coverage of emerging events. This can become frustrating for end-users, since they have to navigate through these long channel lists that contain all channels' and programs' descriptions to choose from, and they change every day, making it difficult to search for what content to watch. Nevertheless, these platforms sometimes allow users to record the content or directly watch it later on, but the period of time before it becomes unavailable tends to be short.

As explained, one of the main problems with EPG is the navigation through the User Interface (UI) when trying to switch between channels. Traditionally, these types of services did not use to be designed or tailored to the specific user, so to solve this problem, some studies were conducted to improve the way users switch content. For example, studies in [6], [7] and [8] used channel popularity-based content pre-fetching to reduce channel switching delay, which is much longer in IPTV than in traditional TV [4]. Meanwhile, other studies used channel switching prediction to simply improve user experience of finding interesting channels to watch. If an IPTV can be on the same level as an OTT on this aspect, they should have a competitive advantage, since they provide different and a lot more content than OTT TVs.

2.1.2 Video on Demand (VOD)

Video on Demand (VOD) is one of the most popular IPTV services, and it is crucial for IPTV providers because it is their second-largest revenue source after monthly subscriptions [5]. Nevertheless, although VOD is a very important part, the main source of content still comes from the EPG. With the increasing popularity of VOD content in the OTT TVs, IPTV have found a need to incorporate this type of content inside their platform to be able to fight with big OTT corporations.

In the IPTV market in South Korea, for instance, which is led by major South Korean telecom operators, IPTV service providers have cooperated with global OTT services to strengthen their service competitiveness [9]. This is also the case for Telefónica, which have made it possible for users to enjoy direct access, through the Movistar+ platform, to these OTT TVs such as Netflix, HBO or Amazon Prime Video, among others.

Furthermore, IPTV users also can subscribe or use OTT service through IPTV without smart TV or an additional media player such as Chromecast and watch exclusive or original content from the OTT service that is not provided in their current IPTV service. This can boost the power the IPTVs have to hold current users or acquire new users, which before thought IPTV offer was insufficient or not attractive enough [9,10].

Both, global OTT platforms and local IPTV ones, can benefit from the strategic asset of an integrated service. While IPTVs get the advantage of offering this very demanded service and all of their exclusive content, it is also helpful for global OTT services to enter new markets and get new subscriptions [10].

2.1.3 Differences within IPTV platforms

Although all the IPTV platforms have some fundamental similarities and characteristics that all of them comply with, there are also big differences between them. These distinctions can be seen by the user or not accessible to them because they are in the backend.

Firstly, and maybe the most obvious difference, is the content they offer. Although it may be obvious to see that IPTVs from different countries or regions can have

different channels and contents, even ones that come from the same region can also have different agreements with different EPG channels and OTT TVs and offer very different content. For example, the content is not the same in Movistar+ Spain and the one in Chile, but it is also different from Orange TV Spain. Moreover, they can offer packages of channels, such as sport channels or movie channels which can be added to the standard subscription with an additional cost. These packages sometimes come with exclusive content or channels that no other platform can provide, which may give them a competitive advantage. They can also provide Pay per view (PPV) content.

A current example of a great relationship between OTTs and IPTVs is the deal between DAZN and Movistar+. DAZN is a rising sports OTT TV, that is acquiring more and more sports and leagues. One of the deals they signed was with Moto GP exclusive rights. Thus, in Spain they were the only ones able to provide this content. On the other hand, Movistar+, had the same deal with Formula 1 (F1). Since these are the 2 most viewed motorsport content in Spain, they decided to create a partnership. Now, both of them have exclusive rights for both sports. Movistar+ has incorporated 2 DAZN channels within its EPG platform, one for each sport, and DAZN is now able to provide F1 within its platform too [11].

2.3 OTT platforms

Over-the-Top (OTT) TV is an interactive platform that allows for the distribution and transmission of video streams and audio files to connected devices via the Internet [11]. Through the Internet network, it allows services to be accessed at any time without needing to go through the providers. The content is transported through periodic data and the protocol used is Hyper Text Markup Language (HTML) over networks managed through the Internet. The video and audio contents are available as long as the user has access to the Internet network.

OTT TV can deliver television to any destination where the user is connected to the Internet using any mobile device, which is considered a great advantage over other technologies. The user is able to access High Definition (HD) content from multiple platforms such as iPad, iPhone, Tablet, TVs, STB and PC, through Adaptive Streaming (AS) technology.

As explained before, the main characteristic is that all of these type services offer VOD content. Their main revenue stream comes from user subscriptions which normally pay monthly or yearly fees.

2.4 Movistar+ services

Once we have explained how the current situation is relating to the streaming platforms, we can define what exactly is the platform we are dealing with in this study. Movistar+ was born in 2015 and is a pay-tv platform owned by Telefónica that goes hand in hand with a satellite, ADSL and fibre optic television offer [12]. It reached the televisions of Spain after the association of Movistar TV and the old Canal+ platform, which maintained the monopoly of sports in Spain since 1990 [13].

Telefónica bought 56% of its competitor Yomvi, Canal+'s OTT, and became Movistar+, using satellite and broadband transmission systems. It became the leader in pay TV and VOD. Its competitive advantage was based on its content, exclusive rights to sports and, in particular, football matches [14]. Faced with this sports offer, OTT companies such as Netflix, HBO, Wuaki and Amazon were only able to keep up with Movistar+ by offering exclusive movies, series and sitcoms. It was not until 2019, when Netflix was finally included inside one of the packages offered by Movistar+ [14].

Although every country has different UIs, in Appendix 2.1 it can be observed an example of how Movistar+ Chile looks like.

Nowadays, the platform has expanded also to the Latin America market and offers its platform to a range of 14 countries. Despite being considered to be an IPTV system, it has evolved by integrating into its own platform accessibility to multiple OTT platforms and taking advantage of the best of both platforms. They also produce movies and series of their own which they are able to supply in exclusivity.

One of the problems of integrating these OTT services is that they lose visibility of the user's behaviour when it leaves their website or app and goes into the other service. So, they cannot collect any metadata if the user stops the content and when it does it, when it exits the streaming platform, if they disliked the content inside the other platform, etc. This will become an issue in this study since the session duration in VOD sessions may not be 100% reliable.

Even though OTT platforms tend to be distributed in different profiles, for different members of the family or friends, Movistar+ Chile still does not offer this option. For example, watching the web contents on a smart TV is significantly different from other connected devices like a smartphone or a PC. A smart TV is a multi-user, lean-back supported device, and normally enjoyed in groups. Moreover, the performance of a recommender system is questionable due to the dynamic interests of groups in front of a smart TV.

This also complicates the creation of an optimal RS, since they cannot know which member of the family is using the service, or if it is being watched by more than one member. Of course, you would not make the same recommendations to a teenager and an adult, or to a whole family. The usage of profiles, allow to make the distinction of recommendations within different members. This is not available right now, and that is why analysing the data and trying to observe patterns can try and help create time zones where a certain content tends to be watched.

This is still a current problem being studied with different approaches and solutions as it can be seen in [15,16,17,18,19,20,21,22,23,24,25,26,27]. This is one of the main motivations for this study to analyse how users behave in order to improve the recommendation users receive.

2.2 Data mining algorithms

With the advancement of information technology and the need to extract meaningful information from datasets [35], data mining has become an important procedure for uncovering hidden patterns in large amounts of data [39].

Data mining, also known as Knowledge Discovery in Databases (KDD) [31], is a technique that combines approaches from a variety of fields, including statistics, neural networks, database technology, machine learning, information retrieval, etc [34]. KDD's algorithms identify underlying patterns from data that can be many terabytes in size [30]. Data cleaning, data selection, data transformation, data pre-processing, data mining, and pattern evaluation are all processes in the KDD process [32].

2.2.1 Association rule mining with Apriori algorithm

There are numerous association data mining algorithms. However, one of the most essential data mining functionalities is association mining, which is also the most popular technique examined by researchers in [28]. The core of data mining is the extraction of association rules [36]. Association rules are “implication” or “if-then rules” with two measures which quantify the support and confidence of the rule for a given data set [29]. An important subject of research in dataset [34] is the mining of a database of sales transactions between objects for association rules. The advantages of these principles include the ability to detect unknown associations and produce results that can be used as a basis for decision-making and prediction [36].

The process of discovering association rules is divided into two phases [38, 33]:

1. In this first phase, we try to detect the frequent itemsets and generate the association rules. Every set of items is called itemset, if they occurred together greater than the minimum support threshold [37]. This is referred to as a frequent itemset. This phase is more relevant than the second one because finding frequent itemsets is simple, but it is expensive [28].

2. In the second phase, it can generate many rules from one itemset. Given an itemset {I1, I2, I3}, its rules are {I1 → I2, I3}, {I2 → I1, I3}, {I3 → I1, I2}, {I1, I2 → I3}, {I1, I3 → I1}, {I2, I3 → I1}. The total number of those rules is $n^2 - 1$ where n= number of items [28]. To validate the rule (e.g., X → Y), where X and Y are items, we have to base it on a confidence threshold. This threshold determines the ratio of the transactions which contain X and Y to the A% of transactions which contain X. This means that A% of the transactions which contain X also have to contain Y [28]. To put constraints on the rules, minimum support and confidence are defined by the user. So, the support and confidence thresholds should be applied for all the rules to prune the rules whose values are less than the thresholds. The problem that is addressed in association mining is efficiently finding from a large set of transactions the correlation among different items [36].

Apriori is a popular algorithm for extracting frequent itemsets from a huge database and obtaining the association rule for discovering knowledge [28].

An example of such a rule is “if a market basket contains orange juice, then it also contains bread”. The classical Apriori algorithm as suggested by Agrawal et al. in [3] is one of the most important data mining algorithms. It uses a breadth first search approach, first finding all frequent 1-itemsets, and then discovering 2-itemsets and continues by finding increasingly larger frequent itemsets. For mathematical proofs and a more detailed explanation of this algorithm visit [28,29,38,40]

2.3 Data visualisation tools

An emerging and complementary tool of data analysis is data visualisation in order to envision relationships and then communicate those relationships convincingly to others [41]. As it is explained in [42], “Data visualisations tools are used in industry to support decision making and also in academia. In the business analytics visualisation is most useful to fully monitor all the activities and also to undertake decisions in time. In industry, analytics is very useful to understand the company’s market position”.

Data visualisation is often thought of as recent innovations. In truth, the use of graphs to represent quantitative data has a long history. These roots can be traced back to the earliest maps and visual representations, and in other professions like cartography, statistics or medicine [43]. Moreover, new breakthroughs in the fields of technology, mathematics, and empirical observation and recording all helped to enable the wider usage of graphics.

There are contributions from a variety of disciplines in the data visualisation field. Psychology, for example, researches data perception and the impact of certain components on how they are perceived, such as colours and shapes. Machine learning and data mining techniques are examples of new domains pioneered by computer science and statistics. Building infographic dashboards requires the use of graphic and multimedia designs. It could manifest itself in the form of infographics and interactive dashboards [42,43].

2.3.1 Tableau

Tableau is a free data visualisation software widely used in the data analytics industry [41]. It competes with Microsoft's visualisation tool, Power BI, every year to be the most used tool in the world. In this article [41], the authors introduce an exercise that teaches the fundamental Tableau concepts and commands needed to create charts, assemble them in a dashboard, and tell a story of patterns observed in the data. Since its functionalities are out of the scope of this project, to get more information on how this tool is used, please visit [41].

3. DATA PRE-PROCESSING

In this section, it is explained all the steps that were executed in order to prepare the raw data for the data analysis that would be performed in the following section.

3.1 Data gathering

This process of the project was done by the Telefónica tutoring team made up of 8 people with different backgrounds e.g., data engineering, business intelligence, project management, etc. For this step, it was necessary to extract information from user sessions. In our case, we were dealing with Latin America Telefónica users which included all the countries from the table in Appendix 3.1, with exception of Spain, which has a separate team and platform. The country that was selected for this project did not really matter, so they chose Chile as our subject.

Apart from this, it was crucial to define what a *watching session* is. In this case, the Telefónica team had already established some rules on this, which they use on a daily basis. For them, a session is created every time a user enters a new channel or a content ended in that channel and a new one started playing. So, in the case when someone is watching a certain content, it changes to some other channel and then goes back to the first content, we are dealing with 3 different sessions. For the case where a content ends and a new one starts, we would have 2 sessions.

This is different to some other IPTV platforms which consider a watching session as the whole period of time from which a user opens the platform in its device, watches a sequence of channels and then turns off the device or logs out of the platform. Thus, if we took the previous examples for this other case, we would only have one session.

Moreover, it was really important to decide which sessions were of interest. For this reason, an upper and lower thresholds were introduced. These were set up as follows: the lower threshold was determined to be 1 minute and the upper one was defined to be 3 hours.

We did not care about sessions that lasted less than 1 minute, since they probably include sessions where the user pressed a button by mistake, the user is navigating the channels to go from one to another one, or the user is just channel zapping. Channel zapping or switching is defined as the process in which a user is changing between channels. We consider there are two different types of channel switching:

- Jumping: this type of zapping consists of intentionally jumping from the current channel to another target channel, without going through other unwanted channels [1]. It can be done by typing the channel number on the TV remote, or any other method characteristic from tactile devices.

- Tuning: in this type, a user randomly navigates to the next or previous channel by pressing the channel up or down button on her remote (or any other method) and going to the adjacent channels.

So, with a lower threshold we are removing sessions created by tuning channel zapping between channels that are further apart.

The upper threshold of 3.5 hours is thought to delete sessions in which the user is not watching the content anymore but it is still playing in the background or the user is not even in front of the device and has forgotten about it.

After having defined what is a session and which ones we are interested in, the Telefónica team extracted all the channel-watching sessions from 270 users during 2 months. The resulting extraction consisted of 6 different datasets with a total of 203,065 sessions from Movistar+ customers placed in Chile between 31.10.2021 and 31.12.2022, both days included.

3.2 Data understanding, cleaning and preparation

Once the Challenge definition and Data gathering stages are covered, the next step is the Exploration Data Analysis (EDA) and Transformation. In this stage, Visual Analytics (VA) takes a key role to explore and understand the features and their relationships between them. Thanks to this process, a data scientist will have more context to determine the algorithm(s) to apply to the data according to the problem definition.

As in any data science project, the majority of the time has been dedicated to understanding and cleaning up the data provided, in this case, by Telefónica. wrangling refers to data transformation that involves data cleaning, structure, and enrichment. In this section I will explain all the steps taken to transform and adapt the data for the next process of the Data analytics workflow. For this I have used a Jupyter Notebook python file, which is one of the best tools for this type of task.

3.2.1 Merging all the data

As stated in the previous section, there were 6 different datasets of watching logs which relates to the 6 different types of sessions they define. These sessions are classified by 2 different properties:

1. User sessions coming from 3 different services (ITV, OPL and TVE DTH)

2. User sessions with 2 types of content (VOD and Live/EPG)

If you mix these 2 parameters you get 6 different types of sessions and therefore 6 datasets. This is not ideal, so the first step was to read all the data and put it all together into one single dataset. After merging all the data together, we ended up with the before mentioned value of 203,065 rows with a total of 41 columns.

```
RangeIndex: 203065 entries, 0 to 203064
Data columns (total 41 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   user_id                 203065 non-null  object
 1   customer_id             0 non-null       float64
 2   subscription_id         0 non-null       float64
 3   unique_user_id          203065 non-null  object
 4   device_id               203065 non-null  object
 5   channel_call_letter     200546 non-null  object
 6   channel_name            200546 non-null  object
 7   channel_type            0 non-null       float64
 8   channel_subtype         0 non-null       float64
 9   program_id              203065 non-null  int64
 10  program_name            203065 non-null  object
 11  normal_program_name     203065 non-null  object
 12  program_theme           203065 non-null  object
 13  program_subtheme        202711 non-null  object
 14  date_time_start         203065 non-null  object
 15  end_date_time           203065 non-null  object
 16  program_start           200546 non-null  object
 17  program_end             200546 non-null  object
 18  device_type_used        203065 non-null  object
 19  real_start_time         200546 non-null  object
 20  real_session_duration   200546 non-null  float64
 21  global_op_name          203065 non-null  object
 22  service_name            203065 non-null  object
 23  day                     203065 non-null  object
 24  audio_language          285 non-null     object
 25  subs_language           287 non-null     object
 26  producer                1024 non-null    object
 27  distributor             2519 non-null    object
 28  service_source          203065 non-null  int64
 29  service_subtype         203065 non-null  int64
 30  session_type            203065 non-null  int64
 31  profile_id              98164 non-null   float64
 32  global_op_id            203065 non-null  object
 33  service_type            203065 non-null  int64
 34  capture_day             203065 non-null  object
 35  duration                203065 non-null  int64
 36  type                    203065 non-null  object
 37  service                 203065 non-null  object
 38  offset                  506 non-null     float64
 39  buffering               506 non-null     float64
 40  commercialization_type  2519 non-null    float64
dtypes: float64(9), int64(6), object(26)
memory usage: 63.5+ MB
```

Figure 1: General information about the dataset and its variables

3.2.2 Dealing with NaNs, NaTs, NULLs and NAs

Once all the data is all in one place, the next step is to take a look for columns which may not provide any utility. The first thing that comes to mind are columns which are pretty empty and filled with Na, NaN, NaT … As it can be seen above, there are a few columns which do not have data at all or have very few samples being non-null. Therefore, the following 11 columns need to be dropped since they do not provide any useful information:

customer_id, *subscription_id*, *channel_type*, *channel_subtype*, *audio_language*, *subs_language*, *producer*, *distributor*, *offset*, *buffering* and *commercialization_type*.

If we take a look more closely, the VOD sessions have NA values for some columns such as *channel_name*, *program_start*, *program_end*, *program_subtheme*. It makes sense because the VOD content does not have a scheduled start or end time, a channel name, and it is more difficult to assign subthemes. For this reason, I applied the following changes.

- *channel_name*

The missing values for these rows were assigned a “new channel” named **VOD**.

- *program_start, program_end*

For his cases, I filled in the NaNs by inputting the values from the *date_time_start* and *end_date_time* respectively. I consider this to be the best solution, since it makes sense to put the same starting and end datetimes and this way, we can keep this field.

- *program_subtheme*

The missing subthemes were assigned its *program_theme* value. It allows us to keep this column without missing values despite it not providing extra information, just as in the previous cases.

3.2.3 Single valued columns

The following step requires making an analysis of how many different values each column has. For example, by doing this step, I can know how many different users, channels or programs are included in this data. Nevertheless, the part I am most interested, is finding columns which only have one value throughout the whole dataset, which practically makes those columns useless. Having a column where all samples have the same value is the same as not having the column at all.

```
MOVISTAR CHILE    203065
Name: global_op_name, dtype: int64
1    203065
Name: session_type, dtype: int64
0.0    98164
Name: profile_id, dtype: int64
CL    203065
Name: global_op_id, dtype: int64
```

Figure 2: Single valued variables, its values and other information about them

Above we can see the 4 variables which only contain one value. Firstly, the value appears, then the number of times it appears (in this case we see *profile_id*, has also approx. half values as NaN, so it is removed for two different reasons) and then we see the name of the variable and its datatype.

It is worth mentioning that *global_op_id* and *global_op_name* are only discarded because we only have data from one country, Chile in this case. If we were to analyse data from multiple countries, as it would in a bigger and posterior analysis, this would be the most important feature of them all. The content displayed in the different channels is mostly dependent on the country from which you are buying the services, and even the channels vary from country to country, and therefore this would be the most significant one when determining which content can be recommended.

After this step, we removed 15 columns, meaning that we are left with 26 columns.

3.2.4 Repeated or equivalent variables

From the names of the different variables we can observe that there are some columns with very similar names, which may indicate that they may be equivalent, or very similar, so it makes sense to only keep one of them. Here you can see the ones that satisfy this, and which ones are kept and which ones are **dropped**:

- *user_id*, ***unique_user_id***

We only need one of them and the Telefónica team prefers to use the *user_id* one. Both are actually unique within the same country, so for this study it was good enough. For next stages with more countries, there could be repetitions with other countries so the *unique_user_id* one would be better.

- *channel_name*, ***channel_call_letter***

For the channel names we have two very similar columns. However, the *channel_call_letter* one is an abbreviation of the channel name and therefore some of them are quite different, and it would require some extra work when trying to match the abbreviation with the full name of the channel. So, for this reason I chose the *channel_name* variable that contains the full name.

- *program_name*, ***normal_program_name***, ***program_id***

In this case, we were more interested in the program name than its id, since it could provide more information about the topic or similarities with other programs. However, for other use cases or for model creation, the id would be more useful because it is a numerical variable and not a categorical one. Then, between the 2 variables with names, the *normal_program_name* is reserved for normalizations of the names, but sometimes it is not done, so I prefered to keep the other one which is more consistent in its formatting.

- *date_time_start*, ***real_start_time**, **day**, **capture_day***

The day and capture day can be extracted directly from the date_time_start which also provides more usefulness and versatility. For the *real_start_time*, Telefónica would rather take a calculated parameter than an original one from the source, for its unreliability in some cases.

- *duration*, ***real_session_duration***

The Telefónica team prefers to use a calculated parameter with the difference between the end and the start of the session (*end_date_time* - *date_time_start*) rather than the original data from the source because it tends to be more reliable, as explained in the previous point.

3.2.5 Data types and value modification

As seen in the dataset information above, there are some string variables defined as objects, which are actually datetimes or integers. Additionally, the datetime ones are recorded in UTC time zone which does not reflect the true time when the session was done. Chile has actually different time zones that can be seen in Appendix 3.2 and that require some extra attention.

- *date_time_start*, *end_date_time*, *program_start*, *program_end*

*These a*re transformed from object into datetime64[ns] datatype, the nanoseconds part is removed and then the time is transformed from Coordinated Universal Time (UTC), UTC+0 to Chile Continental, which is UTC-3 for the period we are analysing.

- *duration*

This variable is transformed from object to integer datatype.

- *device_type_used*

For this case, we established a classification of *device_type_used* column values into more general classification since it had lots of different values that can be grouped together. In the Appendix 3.3 it can be seen how each device is classified into these groups: TV, STB IPTV, STB CATV, STB OTT, STREAMER, PC, TABLET and SMARTPHONE.

We have lots of elements that do not provide so much information separately, but if we group them, it provides a lot more information about which type of device is being used for that session.

- *program_theme*

For this variable, I have chosen to modify the names of the variables. There was a distinction between the VOD content themes and the Live themes. Apart from this, I also decided to put the repeated themes from both types together with the same value. In Appendix 3.4, a table with the transformations.

With these modifications, we put the focus on the actual subthemes without caring if it is VOD or Live, which can already be extracted with another column (*type*), and we simplify the names so it is easier and less painful to deal with.

- *program_name*

This modification is also following the purpose of simplification. Many programs are composed of many different chapters or episodes, and their full *program_name* tend to be composed of the general name of the program followed by the specific title of the episode, e.g., Robin Hood: El príncipe de los ladrones. What we want is that all chapters of the Robin Hood series are named just Robin Hood, so we can put them together as only one program. For that we can see that all of these cases, the main title is separated from the chapter title by the character ":", so we only keep the first part until that.

3.2.6 Final variables selection

Once we have dealt with all the problematic columns, we also have to decide whether the content of the variable is going to be useful or not, and drop the ones that do not provide meaningful data. After all the previous steps, we are left with only 17 columns from the starting 41. By looking at the variables left, we see there are 5 variables that all refer to the service. And from those 5, only 2 really matter.

As explained in the first step of the process where we merged all the different datasets, they were separated by *service*, which had 3 different values (ITV, OPL and TVE DTH). So, this one I feel like it is important to keep as it could bring some useful insights. Apart from this I decided to keep *service_name* which

separates between IpTv and Go services, which is another classification. In Appendix 3.5, it can be seen the classification made by Telefónica.

So, in terms of the variable selection, for this case, we see the variable *service_source* is practically the same but with numerical values (1 for IpTv and 2 for Go/OTT). Again, I have decided to use the categorical one instead of the numerical one, because for the data analysis is more comfortable to use, but when stepping into the model creation phase, we would prefer a numerical category. And then the other two that were discarded were, *service_subtype* and *service_type*, since they are both lower levels of a hierarchical variable that starts with *service_name* (or *service_source*). Therefore, I feel like it is not very useful to keep them, since it would be focusing too much on unimportant details and it is only taking up extra memory.

3.2.7 Outliers

After dealing with all the problematic columns, it is also necessary to look into the problematic samples, which tend to be outliers. An outlier is defined as a data point which is very different from the rest of the data based on some measure. Such a point often contains useful information on abnormal behaviour of the system described by the data [3]. In our case, the outliers are the samples whose session duration is not within the thresholds defined in the previous section (less than 1 minute or higher than 3.5 hours). When examining the data, I realised the previously filtered values by Telefónica might have been done on the live data only, since there were quite some sessions that did not satisfy the threshold requirements. As a result, I decided to remove those outliers. In total, there were 245 outliers in the dataset.

3.2.8 Creation of an auxiliary dataset

By looking into the values of the *subthemes* columns, it can be seen that they are composed values; meaning that a single column value can actually have more than one value inside it, e.g., “Interview, Politics, News”. This makes a lot of sense since it is very common to classify certain content with more than one *subtheme*. However, for our use case it is actually impeding us analyse with clarity which is the popularity of each of them separately, since the *subtheme* “Interview” is detected as a separate one than “Interview, Politics”, or “Interview, Politics”. What we actually need is that each sample that contains the *subtheme* “Interview”, whether is alone or mixed with others, is counted as a content with that subtheme.

A possible solution would be to just separate the column into different columns. An example would be something like the following…

Subtheme		Subtheme_1	Subtheme_2	Subtheme_3
Interview, Politics, News	→	Interview	Politics	News
Reality, Culinary		Reality	Culinary	

Table 1: Example of the transformation made to subthemes column

The problem with this solution is that if the sample with the most subthemes has 7 different ones, we would need 7 columns. Because normally the programs tend to have 2 or 3 subthemes only, this would create a very sparse dataset because it would have lots of empty rows for those extra columns.

As a consequence of this, I decided to create a separated auxiliary dataset, in which each row's subtheme is divided into as many different samples as subthemes are. With the previous solution we were separating the *subthemes* variable into different columns inside the same dataset. With this one, however, we are creating separate samples in a different dataset where we keep some of the important columns. Accordingly, if a sample has 3 different subthemes, it will be separated into 3 different samples, having all the same data except for the subtheme value.

When doing this, it is of vital importance to keep the Session ID from the original dataset. As I said, we are keeping the same data and this can cause problems when analysing the duration of the contents and the number of sessions. The total duration is now a lot bigger since there are lots of "repeated rows" and the number of sessions has also increased because we have a lot more rows. More precisely, we went from having 202,820 rows to 423,777, which is more than double the number. Therefore, it is crucial to keep the session ID, so we can know which group of samples from the auxiliary dataset (subthemes dataset) actually come from the same viewing session. Below, it can be observed how the dataset looks like in the Pandas dataframe, with 3 different sessions from the same user.

	user_id	channel_name	program_theme	program_subtheme	date_time_start	duration
0	f1b9ff065d526bb021216d90a92455a2f036f26c74ac95...	CHV	Special	Reality	2021-12-18 22:31:22	964
0	f1b9ff065d526bb021216d90a92455a2f036f26c74ac95...	CHV	Special	Culinary	2021-12-18 22:31:22	964
1	f1b9ff065d526bb021216d90a92455a2f036f26c74ac95...	MEGA	Special	Interview	2021-12-19 19:57:34	1437
2	f1b9ff065d526bb021216d90a92455a2f036f26c74ac95...	MEGA	News	Interview	2021-12-19 13:01:22	3338
2	f1b9ff065d526bb021216d90a92455a2f036f26c74ac95...	MEGA	News	Politics	2021-12-19 13:01:22	3338

Figure 3: First 5 rows of the subthemes pandas dataframe

As displayed, the first column contains the Session ID for each sample. So in the previous case, the first two rows actually come from the same session.

3.3 Dataset files

After all of the data preparation and manipulation, we finally have a dataset we can work on in the next stage where we are going to analyse the data. In the end, we are left with 14 of the 41 initial columns and 202,820 of the 203,065 sessions in the Sessions dataset, and 6 columns and 423,777 entries in the Subthemes dataset. A general overview of both datasets can be seen in Appendix 3.6.

There are multiple file formats accepted by Tableau, but the most common ones are JavaScript Object Notation (JSON) and Comma-Separated Values (CSV). I decided to have both of my datasets as .csv files.

4. DATA RESULTS

4.1 Data Visualisation: Dashboards

Once we have done an exhaustive study of the data and the problem we are dealing with, and the data has been cleaned and prepared for the analysis, I proceeded to elaborate some interactive dashboards using Tableau. These dashboards are used as a tool to filter, analyse and find the main trends hidden within the data.

For almost all the dashboards, it is possible to select different views depending on the variable we want to look into by making use of the Categorical and Numerical selectors. The categorical selectors can choose between the following variables: *User*, *Channel*, *Device*, *Theme*, *Subtheme*, *Program*, *Service, Content Type*, *Service Name*, and *None* (if user does not want any). Moreover, it is possible to select which numerical variable we are interested in. In this project we do not have many of them, so it is difficult to play around with them, but the 3 main indicators I have used are: *Number of Sessions*, *Total Duration (h)*, and *Avg. Duration (min)*.

Appendix 4.1 contains other dashboards that we constructed but are not included in this section. They are also really useful and can also help find valuable insights.

4.1.1 By duration

For the first dashboard I chose to look into the duration of the sessions. On the top-left we can have a general look into the total viewing time during the 2 studied months, the average total viewing time by user and by session. Then on the top-right we have the legends and 3 selectors. With them we can choose the different variables to show in the treemap. Size and variable are Numerical selectors, so we can do different combinations of the 3 main numerical variables mentioned before. For the Objective Variable we use a Categorical selector to choose which is the detail we are looking into. Finally, on the bottom-left we have a user clustering based on the 3 numerical variables, which is done automatically by Tableau.

Figure 4 depicts a snapshot from a possible configuration of the dashboard.

In the scatterplot we can see there are 3 general types of users. In pink we have users which do not use the service very frequently and with very short sessions. The cyan group is made up of users that have a lot of sessions and a lot of viewing time in very short sessions. So their sessions are frequent but short. Finally, we have the purple cluster. They are the opposite of the cyan one, they do not use the service so much, but when they do their sessions are long and intense.

By looking into the treemap, we can clearly see that most of the content is consumed through a TV or an STB (Set Top Box) IPTV. However, and here is an important discovery, the average session type is much bigger in PC, Streamer and Tablet. This breaks with Telefónica's and my initial hypothesis that bigger devices meant longer sessions. I consider this to be produced by the fact that in

big screens and lay-back devices users tend to do more zapping and smartphones the device may be too small. However, with medium sized ones users can get the best of both cases, and go directly to the desired content and take advantage of the portability; and have a big enough screen.

Figure 4: Dashboard by duration created with Chile's sessions dataset.

In Figure 5, it is shown the difference between TV and STB IPTV devices, which are the two most common ones. They are practically separated from one another. TV is being used by a lot more users with not a big total duration while STB IPTV has less users who consume a lot more content. Also TV has a very widespread average duration and the other tends to fall on the shorter side of the spectrum.

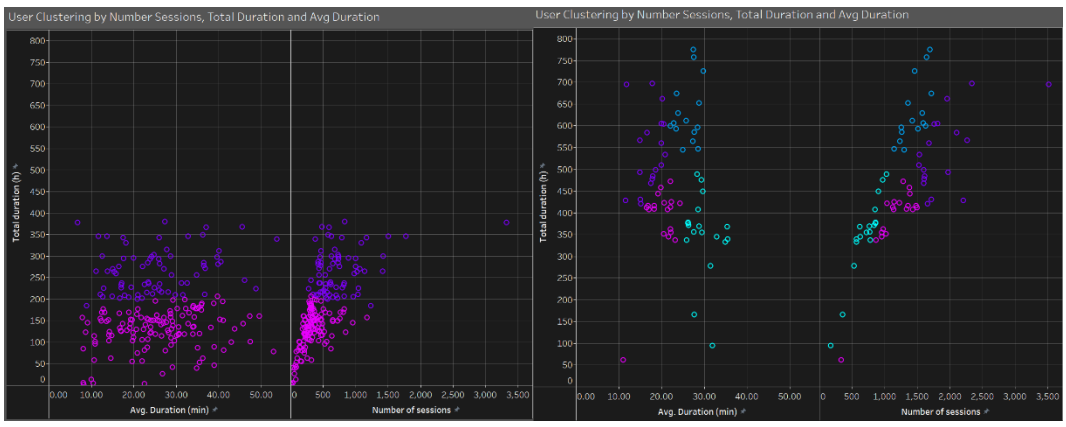


Figure 5: User clusterings for TV (left) and STB IPTV (right) sessions.

4.1.2 By number of sessions

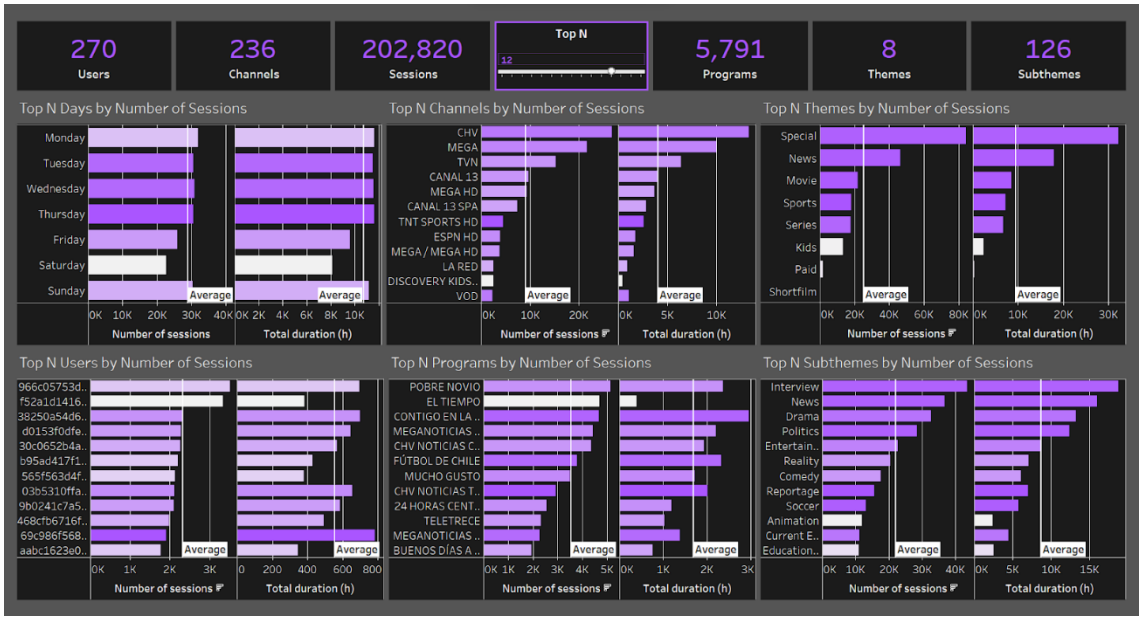


Figure 6: Dashboard by number of sessions created with Chile's sessions dataset.

With this dashboard in Figure 6 we can have a general view how the dataset is structured, in terms of how many unique sessions, users, channels, etc; we have. Then it is possible to select the level of detail of each of them with the top N parameter. In this case it is showing the first 12 elements for each case ordered by the number of sessions. For all of them we can see the number of sessions and total duration, and the colour is determined by the average session duration. All of them act like a filter, so it is possible to observe which are the top N “Sports” programs that a certain user watches on Saturday in the “ESPN” channel.

From this, we can for example see that Saturday is the day with less consumption and shorter sessions, which may tell us that Saturdays people tend to go out instead of watching content. We can also see that the weather program, “El tiempo” is consumed a lot but in very short sessions (~5 min). From the channel “Discovery Kids”, the theme “Kids” and the subtheme “Animation”, we see that also those sessions tend to be short (~10/12 min). For all cases it makes sense since weather news and kids programs are more inclined to be rather short. It is also insightful to see that when “VOD” is considered a channel, it is the 12th with the most sessions.

4.1.3 By start day and hour

Figure 7 shows a simple visualisation of how much content is consumed on a given day and hour. The *Objective variable* is a *Numerical selector* which determines the line value, and the *Detail variable* is a *Categorical selector*, to show the same graph detailed by the top N elements from that variable. There is a filter also to choose any specific element if desired.

Figure 7: Dashboard by start day and hour created with Chile's sessions dataset.

Contrary to what people may think, “Friday” and “Saturday” are days where the least amount of content is watched. During the working days the quantity of time is pretty consistent. By looking at the starting hour one, we can see there are different peaks during the day. The first one comes around 8 am, which is when people normally have breakfast. Then at 1 pm and 9 pm when people tend to have lunch and dinner. This may prove there are some time windows where people tend to watch content.

Figure 8: Dashboard by start day and hour filtered by top 4 themes.

Our next question would be what type of content is being watched at those times. The initial guess and from seeing from the previous dashboard that “News” is the second most common *theme* and *subtheme*, is that people tend to eat at the table with all the family and watch the news. In Figure 8, it can indeed be seen that at

6 am, 1 pm and 9 pm “News” is the most watched *theme*. There we can also see that when “News” is not the main one, “Special” is the one being consumed, and they pretty much alternate during the day. By looking into the *starting day*, we can see the tendency to decrease on “Saturdays” is not true for “Sports” and “Movies”. Moreover, on “Wednesday” we see an increase in sports which may be caused by international competitions in football such as “The Champions League” or “La Copa Libertadores”.

4.1.4 By Subthemes

One of the most important variables to look for when making recommendations, if not the most, might be the *subthemes* the user is most interested in at a given hour. So, if a user connects at 8 am and we know it normally watches “News” at that hour, the service should recommend that type of content. Taking the “News” example from before, I wanted to look into what type of “News” content is being watched.

In Figure 9, it can be seen that we have a very similar dashboard with the *Numerical Selector* and a top N parameter and a theme filter. For this case, I wanted to look into the top 3 *subthemes* from “News” *theme*. The same 3 peaks can be seen with different sizes, but again, we can see a very clear tendency with this type of content and there are definitely time windows where this content is watched. Because the trends are really similar, this may also mean that in the “News” *theme*, the content may be classified with “News”, “Interview” and “Politics” *subthemes* together. This is going to be studied in the following section with the use of a data mining algorithm.

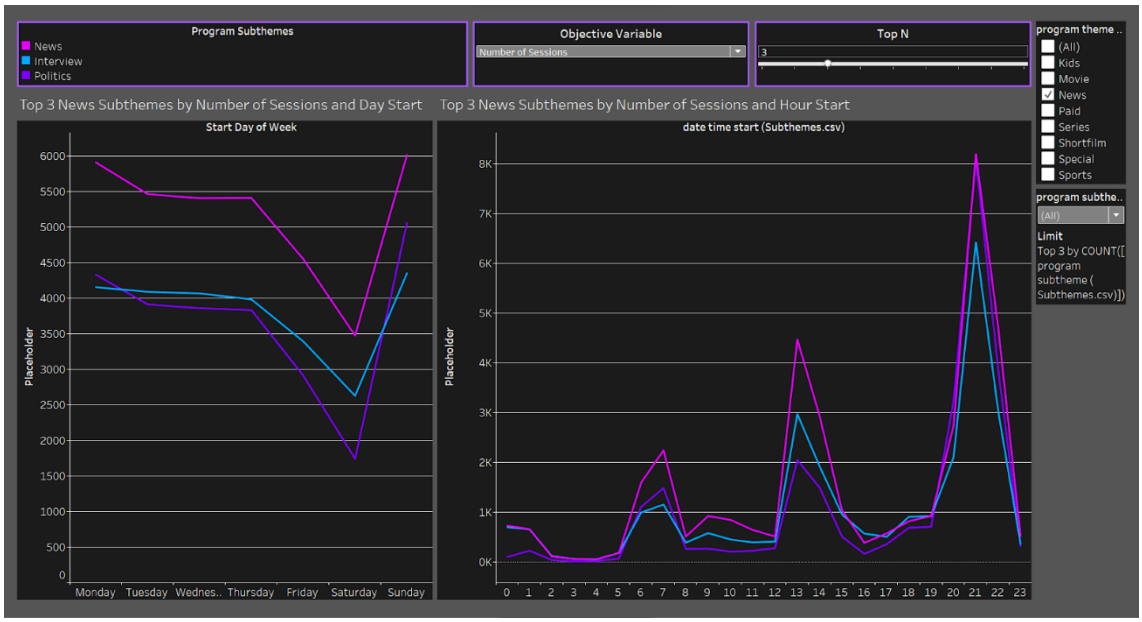


Figure 9: Dashboard by subthemes filtered by top 3 “News” subthemes.

4.2 Data Mining: Association rule mining with apriori algorithm

This part of the analysis was done with python inside a Jupyter Notebook file. The code is using a python apriori library [45] which allows for a quick implementation of the algorithm..

The goal of this analysis is to determine and find subthemes that tend to appear together inside the content. So for example we are looking for a rule such as "if a session contains News subtheme then it also contains Interview subtheme". Later on, we will add other variables to the same analysis, such as the starting hour, starting day or the device type.

Here we can see an example of the calculations that are done to see if a certain association rule that was mined meets the requirements and its support, confidence and lift threshold are met. The example is for the rule News → Interview:

$$\text{Support(News)} = \frac{\text{Transactions containing (News)}}{\text{Total Transactions}} = 0.78$$

$$\text{Support(Interview)} = \frac{\text{Transactions containing (Interview)}}{\text{Total Transactions}} = 0.75$$

$$Confidence(News \rightarrow Interview) = \frac{Transactions\ containing\ both\ (News\ and\ Interview)}{Transactions\ containing(News)} = \frac{8}{10} = 0.80$$

$$Lift(News \rightarrow Interview) = \frac{Confidence\ (News \rightarrow Interview)}{Support\ (Interview)} = \frac{0.8}{0.75} = 1.07$$

In [44] it is explained that by filtering the resulting rules with different thresholds we can achieve distinct results. The support shows the number of occurrences of rules, confidence shows the probability of occurrence of rules. If the lift > 1, it means positive correlation, and if lift < 1, it means negative correlation. When lift = 1, there's no correlation between prefactor set and postfactor set.

The strongest rules come from filterings with high values for the 3 thresholds. Basically, the strongest ones are the ones with high support, high confidence and high lift.

For this study I have tried many threshold combinations in order to find the best rules. This has also been done for various variable combinations (subthemes, subthemes + start hour, subthemes + device type, …) in the Jupyter Notebook file where I developed the code. Some of the ones that do not appear in the following subsection can be found in Appendix 4.2. For each of them, I have played with the thresholds. First looking for rules with only high support. Then only the ones with high confidence or lift. In the next stage I searched for the rules which had 2 of the variables with high values. In the end, and logically, the strongest rules are the ones with a high support, confidence and lift.

Having high values for all of them means that the elements from the itemset appear frequently, the probability of occurrence of the rule is also increased and the correlation between the prefactor and postfactor is very positive. However, in this report I will directly show which are the most significant rules and what conclusions we can extract from them.

With this study we can find frequent itemsets within the dataset and their significant “implications”. I display the rules in a table format. Every time there is a colour change within the tables it means the itemset that forms that rule is different. So, if there are two contiguous rows with the same colour it means they come from the same itemset.

4.2.1 Apriori on subthemes transactions

For this analysis only a set of transactions of subthemes is being considered. Each session has a list of subthemes that go from 1 to a maximum of 7 subthemes. The thresholds and strongest rules are:

- min_support=0.026, min_confidence=0.5, min_lift=5

If => Then	Support	Confidence	Lift
['Adventure'] => ['Animation']	0.03	0.58	9.91
['Animation'] => ['Adventure']	0.03	0.50	9.91
['Soap'] => ['Drama']	0.04	0.91	5.66
['Current Events', 'Politics'] => ['News']	0.03	0.98	5.39
['News', 'Interview'] => ['Politics']	0.09	0.77	5.53
['Politics'] => ['News', 'Interview']	0.09	0.67	5.53

Table 2: Strongest rules for subthemes transactions.

By analysing the rules in Table 2, it can be seen that the rules involving the itemset ['Adventure', 'Animation'] have the highest lift values. This means that they are highly correlated. Nevertheless, the support is smaller than the other ones and the confidence level is around random values. For the second and third itemset, we can see a very high confidence level, which is really good. However, the support is not super big, which makes it not very present within the transactions.

The reason why I think the last set of rules are the strongest ones is because the confidence level is quite high, even if not as big as the previous two cases, but the support is around 3 times as big. This shows that this itemset is very frequent and together with having very correlated items and a more than decent confidence level it makes it a very strong rule. This also proves right our hypothesis from the previous section when we observed these 3 subthemes were experiencing peaks of consumption at the same time.

4.2.2 Apriori on subthemes + hour transactions

For this analysis a set of transactions of subthemes and the start hour is being considered. As previously mentioned, from now on I will only show the strongest rules, with higher thresholds in all 3 of them. The thresholds and strongest rules are:

- min_support=0.035, min_confidence=0.5, min_lift=5

If => Then	Support	Confidence	Lift
['Soap'] => ['Drama']	0.04	0.91	5.66
['News', '21'] => ['Politics']	0.04	0.98	7.02
['21', 'Politics'] => ['News']	0.04	0.98	5.41
['News', 'Interview'] => ['Politics']	0.09	0.77	5.53
['Politics'] => ['News', 'Interview']	0.09	0.67	5.53

Table 3: Strongest rules for subthemes + hour transactions.

For this transactions dataset I have increased the min_support threshold. It can be seen the first and third set of rules were present also in the first case. However, we now see a new itemset containing 2 subthemes and one starting hour. As said before, I do not believe it is the strongest set of rules but they have very high confidence, a decent support and a really good lift value. Moreover, it again proves that “if at 9 pm people watch a news program then it will also be a politics one”, and vice versa. This was also discovered in the visualisation section.

4.2.3 Apriori on subthemes + hour + day + device type transactions

For this analysis a set of transactions of subthemes, start hour and day and the device type is being considered. The thresholds and strongest rules are:

- min_support=0.022, min_confidence=0.95, min_lift=5

If => Then	Support	Confidence	Lift
['News', '21'] => ['Politics']	0.04	0.98	7.02
['21', 'Politics'] => ['News']	0.04	0.98	5.41
['News', '21', 'Interview'] => ['Politics']	0.03	1.00	7.11
['21', 'Interview', 'Politics'] => ['News']	0.03	1.00	5.48
['News', '21', 'STB IPTV'] => ['Politics']	0.02	0.98	7.00
['21', 'STB IPTV', 'Politics'] => ['News']	0.02	0.97	5.35

Table 4: Strongest rules for subthemes + hour + day + device type transactions.

For this case, I was able to find some rules with high threshold values. A support of 0.022 may not seem like a lot but with more than 200,000 sessions, this accounts for around 4.5 thousand sessions. I am only sharing the rules which their itemset contains elements from different variables e.g., subtheme, device, hour. Only 6 of the 48 rules are shown.

The rules in Table 4 have one thing in common. They are the most frequent items in their domain. From the four most frequent subthemes, three (['Interview', 'News', 'Politics']) appear here, together with the most frequent hour (9 pm) and second most device (STB IPTV). The rules they generated do not have a very large support, the lift is not super high but the confidence is really good. Overall it is not a great set of rules, but in certain cases it would provide utility.

My goal was to find a decent rule which for a given hour and device it would extract which subthemes are the most frequent. However, this has not been obtained because the confidence in those rules is very low.

4.2.4 Apriori analysis conclusions

Through all of the transaction sets, there has been a common appearance. Rules involving itemset ['Business', 'News', 'Politics', 'Interview', 'Reportage'] and all of their combinations have appeared in all of the cases. The support for these rules is always around 0.22, the confidence level is never lower than 0.95 and the left value ranges from 5 to a very impressive 40.

This association rules analysis has proven that these items are very frequent, and they tend to appear with each other a lot.

The starting day has not really provided any usefulness while the starting day and device can provide some value in certain conditions. Those conditions are, being the peak hours or the most common device types.

I believe these association rules could be of service when designing a recommender system, and further analysis could be done by mixing even more variables or different ones. For example, we could try and find frequent itemsets containing subthemes and users, to find the favourite subthemes for each user. However, a very big amount of session data would be required. With only 2 months of history data, it is difficult to find the user's preferences.

5. CONCLUSIONS AND FUTURE WORK

Summing up the study's goals and motivation, it was its main purpose to generate insight into how users behave inside an IPTV platform. What content do they prefer to watch? How long are its sessions? In which days or hours are the users consuming more content? Are the sessions different depending on the device or the service? Are there time windows with more consumption or different types of content?

These are some of the questions that were made with a focus to improve the knowledge of user behaviour inside these types of platforms in order to improve how recommender systems work and their performance.

Firstly, it has explained which is the state-of-the-art regarding IPTV platforms. To be more precise, it has also been described how these services work, their advantages and main problems. Services which provide both EPG and VOD content in a single-user experience like Movistar+ Latin America does currently. We have also compared them to their biggest competition as of today, OTT platforms.

This study has also provided a small insight into how the most famous data mining algorithm works. The association rules mining is a very used algorithm, which has also experienced some improvements over the time which makes it a very strong analytical tool. Paired with data visualisation tools such as Tableau, it is possible to extract very valuable information for the streaming business.

Before digging into the data, and as it should be done in every data science project, it is crucial to understand what the problem is you are trying to solve, what type of data is needed, and how you need the data to be structured. Once you know this, it is also very important to play with the data; see how the data is shaped, are there any outliers, do we really need all of the variables, are we missing any data, and a very big etc.

Once you have cleaned and prepared the data, it is finally possible to use algorithms or visualisation tools to analyse the data. In this study I have used Tableau and the Apriori algorithm.

From tableau we have extracted very useful insight. We have seen that although TV and STB IPTV are the most common devices, the users are having very different experiences regarding the duration of the sessions and how many sessions they create. We have also proved the hypothesis wrong that on bigger devices the sessions tend to be larger, but rather it is the medium sized and wireless devices the ones where the users have longer sessions. This may be produced by the zapping behaviour in those bigger devices. Also, the initial thought that on weekends users would consume more content is not true for the majority of subthemes. On Fridays and Saturdays, we have seen a big decrease in the amount of sessions overall, although movies and sports do not follow that trend. Also, it was observed a tendency for kids programs and channels to have very short sessions, which is somewhat predictable because they have small attention spans.

Additionally, during the day, special and news content always alternate being the most consumed content and we have visualised how the news “peak hours” were always at times where people tend to eat.

As explained, there is also a big difference between the amount of VOD content being consumed versus the amount of live content. As it was clarified in chapter 2, although VOD is the cause of a big amount of the income generated by these types of service, it is not the main preference of IPTV users.

After this analysis, I believed it was also important to analyse how different subthemes mix with each other. The idea was to find items which frequently tend to go together. This study displayed that rules containing various News subthemes tend to be very correlated. The top 3 of those were Interview, News and Politics, which are 3 of the top 4 most consumed ones, being Drama the third one. It was also unveiled that under certain conditions, the device used and the starting hour can also provide an additional degree of information into what the user may be looking for.

In conclusion, I think this project has met most of its objectives and has set up a firm ground for future work. Although some clustering was done by means of the integrated Tableau services, I believe it would be a great idea to try and do user clustering with more advanced techniques. It would also be a great opportunity to visualise this data for other countries and see if the same patterns can be unveiled or if there are key differences. Finally, I think this study can help build the base of a strong recommender system. This knowledge can be introduced into the logic and design creation of a system in order to improve its performance and improve the user experience in IPTV services.

6. APPENDICES

Appendix 2.1

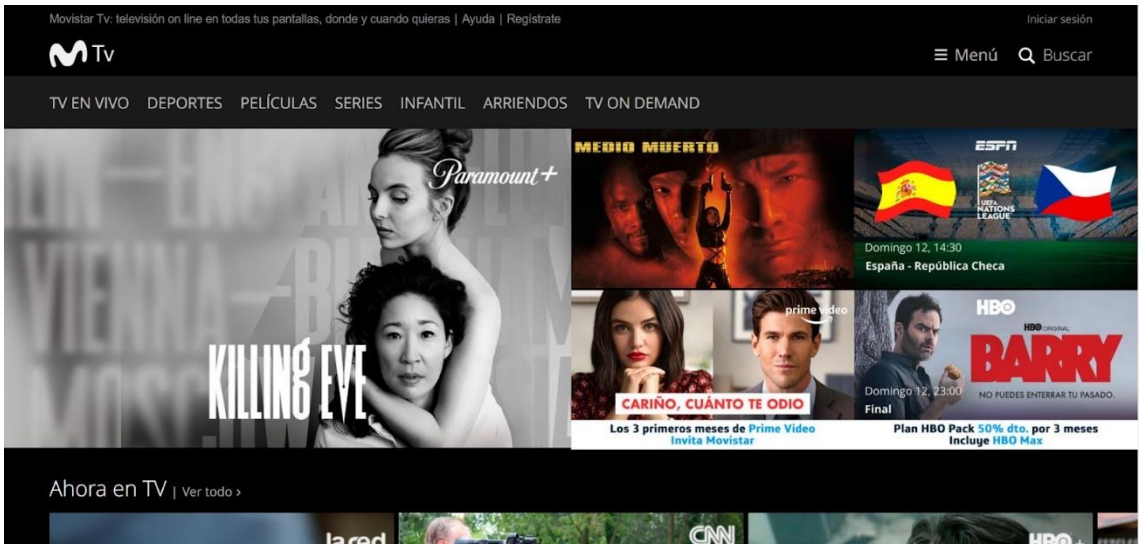


Figure 10: Snapshot from Movistar+ Chile webpage.

Source: https://www.movistarplay.cl/

Appendix 3.1

ID Number	Country	ID Number	Country
1	Spain	8	Colombia
2	Argentina	10	Guatemala
3	Chile	11	Panama
4	Uruguay	12	El Salvador
5	Peru	13	Costa Rica
6	Ecuador	14	Nicaragua
7	Venezuela	201	Brazil

Table 5: List of the 14 countries where Movistar+ is available.

Appendix 3.2

Figure 11 shows the 3 different time zones Chile is divided by. “Chile continental, Juan Fernández y Desventuradas” (red), which contains practically most of the country's population, and the one we considered for this project. “Región de Magallanes y de la Antártica Chilena” (blue) and “Isla de Pascua e isla Salas y Gómez” (green), are not that populated and represent a minority of the country.

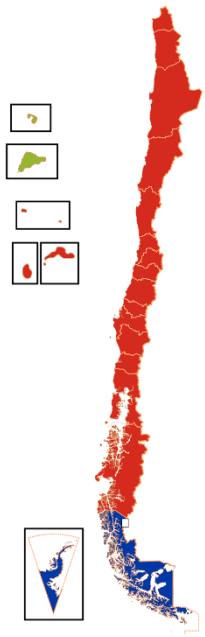


Figure 11: Division of Chile by time zones.

Source: https://commons.wikimedia.org/wiki/File:Zonas_horarias_de_Chile.svg

Appendix 3.3

Group name	Group members
TV	'tvLg_no-Accedo', 'tvSamsung_2017+', 'tvAndroidTv', 'tvSamsung', 'tvLg', 'tvPanasonic', 'tvPhilips', 'tvSony'
STB IPTV	'OpenIPTV_STB', 'Mediaroom'
STB CATV	'stbHybridZapperCable', 'stbProteusCableUHD', 'stbProteusCableHD', 'stbHybridZapperCable1Gb', 'stbHybridPVRCable', 'stbHybridZapperSat'
STB OTT	'stbAndroidTv', 'FireTV'
STREAMER	'Chromecast'
PC	'PC'
TABLET	'tabApple', 'tabAndroid', 'tabWin'
SMARTPHONE	'sphAndroid', 'sphApple'

Table 6: New classification for *device_type_used*.

Appendix 3.4

New values	Old values
Special	MSEPG_SPECIAL
News	MSEPG_NEWS
Movie	MSEPG_MOVIE, Movie
Sports	MSEPG_SPORTS
Series	MSEPG_SERIES, Episode
Kids	MSEPG_KIDS
Paid	MSEPG_PaidProgramming
Shortfilm	MSEPG_SHORTFILM

Table 7: New classification for *program_theme*.

Appendix 3.5

ID Lvl 1 (*service _source*)	**Name** (*service _name*)	**ID Lvl 2** (*service_ subtype*)	**Name**	**ID Lvl 3** (*service _type*)	**Name**
1	IpTv	101	IPTVSub	1001	Iptv
				1002	IptvOpenPlatform
2	Go (OTT)	102	InternetTV	1003	ItvGo
		103	DTH	1006	DthNonConnected
				1007	DthHybrid
		104	CATV	1008	CaTvNonConnected
				1009	CaTvHybrid
		105	MultiscreenDthCaTv	1010	TVEverywhereDthCaTV
		106	MultiscreenIptv	1011	MultiscreenIptvMr
				1012	MultiscreenIptvOpl

Table 8: Hierarchical classification of service variables.

In green we can see the PAYTV services, and in purple, the OTT ones. At the user-device level, the sets are disjoint. However, at the user level only they are not. Hence, a user can be in both 1010 (Smartphone) and 1007 (STB DTH Hybrid).

Appendix 3.6

Appendix 3.6.1

```
Data columns (total 14 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   user_id           202820 non-null  object
 1   channel_name      202820 non-null  object
 2   program_name      202820 non-null  object
 3   program_theme     202820 non-null  object
 4   program_subtheme  202820 non-null  object
 5   date_time_start   202820 non-null  datetime64[ns]
 6   end_date_time     202820 non-null  datetime64[ns]
 7   program_start     202820 non-null  datetime64[ns]
 8   program_end       202820 non-null  datetime64[ns]
 9   device_type_used  202820 non-null  object
 10  service_name      202820 non-null  object
 11  duration          202820 non-null  int32
 12  type              202820 non-null  object
 13  service           202820 non-null  object
dtypes: datetime64[ns](4), int32(1), object(9)
memory usage: 22.4+ MB
```

Figure 12: General information about the Sessions dataset

Appendix 3.6.2

```
Data columns (total 6 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   user_id           423777 non-null  object
 1   channel_name      423777 non-null  object
 2   program_theme     423777 non-null  object
 3   program_subtheme  423777 non-null  object
 4   date_time_start   423777 non-null  datetime64[ns]
 5   duration          423777 non-null  int32
dtypes: datetime64[ns](1), int32(1), object(4)
memory usage: 21.0+ MB
```

Figure 13: General information about the Subthemes dataset

Appendix 4.1

Appendix 4.1.1

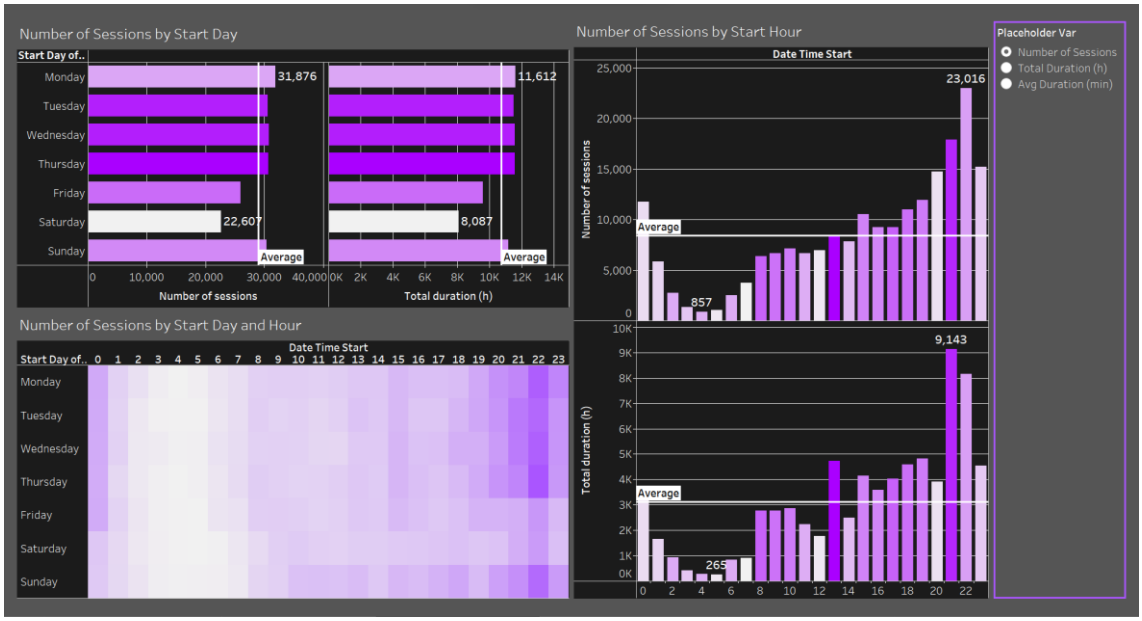


Figure 14: Dashboard by day and hour.

Figure 14 shows a dashboard that is very similar to the one by Number of Sessions, but with a deeper focus on the temporal characteristics of the sessions. It is possible to see the Number of sessions and total duration for each day and hour while having the average duration as colour. In the bottom-left it can be put even more focus on each hour and day based on the *Numerical Selector.* This dashboard can help to get a very general idea of the temporal patterns.

Appendix 4.1.2

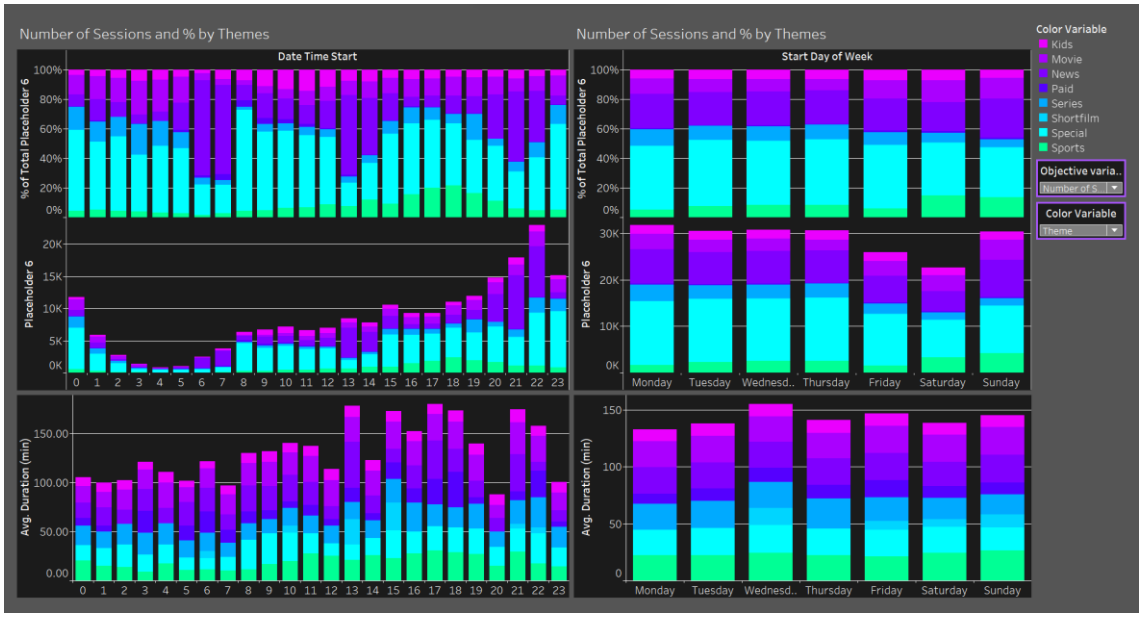


Figure 15: Dashboard by day and time start filtered by theme.

Figure 15 displays for a certain categorical variable, which can be chosen as wanted, what is the distribution among hours and days. In the example, the configuration of filters allows to see for each hour and day how many sessions there is at each hour and the % that represents within that hour. Then on the bottom it can also be seen the average session duration.

The graphs with the percentages exhibit in a very clear way which is the most common theme for each hour and day. As mentioned in previously during the work, “News” and “Special” alternate during the day as the most consumed content.

Appendix 4.1.3

Figure 16: Dashboard by objective variable.

Figure 16 displays four different graphs. The two on the right show the average duration for a selected categorical variable. The top-right one allows to see the Distribution of that categorical variable depending on a second categorical variable. Finally, the bottom-left one allows to see the general distribution within the entire data set.

With the combination of the two charts on the left, it is very easy to see how those variables are distributed. For example, in this case, we can see that IpTv service name is always going to come from OPL sessions. However, OPL sessions do not necessarily need to come from an IpTv service. Moreover, the two graphs on the right allow to put focus on the type of sessions users have in terms of duration.

Appendix 4.1.4

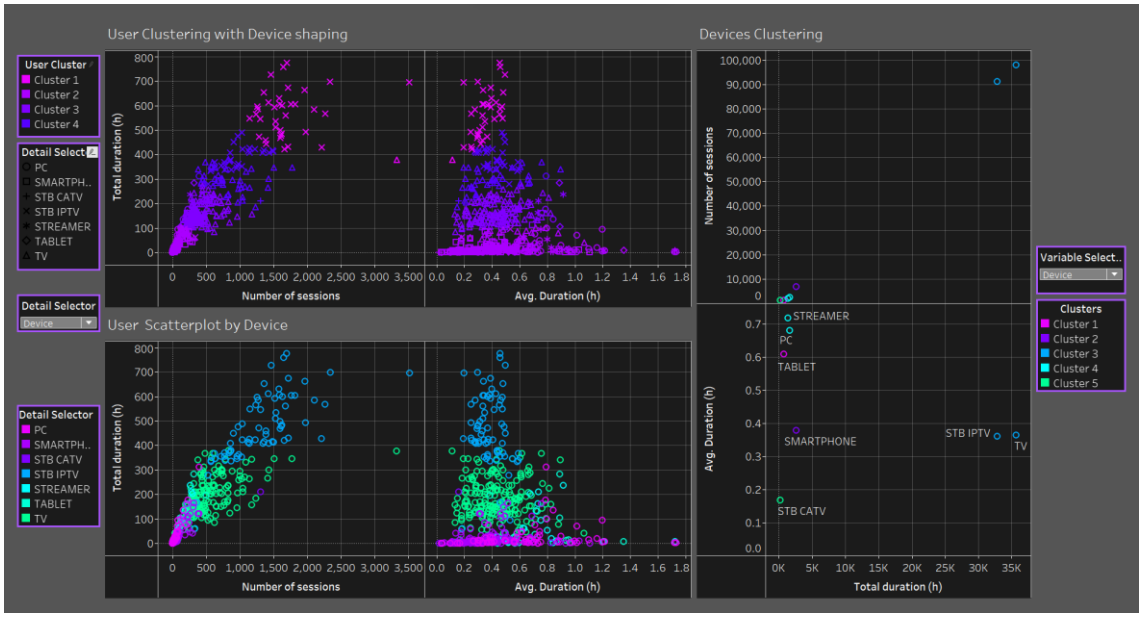


Figure 17: Dashboard by scatterplots with a focus on device.

Since we did not have time to create our own clustering model, we decided to use Tableau's integrated clustering tool to generate different clusters. This can be seen in Figure 17. On the top-left, a user clustering is shown. The main functionality of the scatterplot below and the shapes of the clustering one is to show the difference between the clustering and the actual different groups. It can be seen for example that although STB IPTV and TV are really separated from the other devices and from each other, the clustering separates them roughly in 3 different groups or clusters.

The scatterplot on the right is another clustering done by a categorical variable. In this case there are 5 different clusters for 7 different device types. Thus, there are some devices that can be put into the same group. The obvious one is TV and STB IPTV, which we have seen throughout the project that they are very similar. The other cluster formed by 2 different devices is made up of Streamer and PC. Both of the devices can be wired or wireless and they are medium-sized devices. We also see Tablet is very close to those, and specially to PC. We believe it makes sense as it is a very similar device as a PC.

Appendix 4.2

Appendix 4.2.1

Apriori algorithm on subthemes + weekday transactions. For this analysis, a set of transactions of subthemes and start weekdays is being considered. The thresholds and strongest rules are:

- min_support=0.03, min_confidence=0.2, min_lift=1

If => Then	Support	Confidence	Lift
['Tuesday'] => ['Drama']	0.03	0.20	1.27
['Friday'] => ['Interview']	0.03	0.24	1.11
['Monday'] => ['Interview']	0.03	0.22	1.01
['Thursday'] => ['Interview']	0.04	0.23	1.08
['Tuesday'] => ['Interview']	0.03	0.22	1.04
['Wednesday'] => ['Interview']	0.03	0.22	1.04
['Sunday'] => ['News']	0.03	0.20	1.10
['Politics'] => ['Sunday']	0.03	0.21	1.49
['Sunday'] => ['Politics']	0.03	0.21	1.49

Table 9: Strongest rules for subthemes + weekday transactions

For this case, I was not able to find any rules with the same threshold values as the previous cases. Therefore, I had to lower the standards in order to find at least one rule including a weekday inside their itemset. I have only shown these rules, but with these threshold values there were a total of 35 different rules. The other rules are practically the same as in the previous case, so I do not include them.

Additionally, the correlation between items in these rules is positive but very low and it is very close to =1 which shows no correlation at all. To add up, the support values are on the smaller side and the confidence is really small.

Appendix 4.2.2

Apriori on subthemes + hour + weekday transactions. For this analysis, a set of transactions of subthemes, start hour and the start weekday is being considered. The thresholds and strongest rules are:

- min_support=0.035, min_confidence=0.2, min_lift=1

If => Then	Support	Confidence	Lift
['21'] => ['News']	0.04	0.46	2.53
['News'] => ['21']	0.04	0.22	2.53
['21'] => ['Politics']	0.04	0.46	3.28
['Politics'] => ['21']	0.04	0.29	3.28
['Thursday'] => ['Interview']	0.04	0.23	1.08
['21'] => ['News', 'Politics']	0.04	0.45	3.81
['News'] => ['21', 'Politics']	0.04	0.22	5.41
['Politics'] => ['21', 'News']	0.04	0.29	7.02
['News', '21'] => ['Politics']	0.04	0.98	7.02
['21', 'Politics'] => ['News']	0.04	0.98	5.41
['News', 'Politics'] => ['21']	0.04	0.34	3.81

Table 10: Strongest rules for subthemes + hour + weekday transactions

For this case, I was not able to find any rules with the same threshold values as the previous cases which also contained relevant values from multiple variables. Therefore, I proceeded as in the previous case and only 11 of the 33 rules are shown. As seen before, the starting day variable does not really provide strong rules, as the only one that appears have super small lift and confidence. The first rules of the other 3 sets of rules could potentially be good rules. They have a confidence close to 0.5 and a lift which is not terrible.

The most interesting thing is that they show that given an hour it is indeed possible to predict what type of content the user wants to watch. Nevertheless, they are not the strongest rules we have seen.

Appendix 4.2.3

Apriori on subthemes + device type transactions. For this analysis, a set of transactions of subthemes and the device type is being considered. The thresholds and strongest rules are:

- min_support=0.03, min_confidence=0.8, min_lift=1

If => Then	Support	Confidence	Lift
['Current Events'] => ['News']	0.05	0.85	4.68
['Soap'] => ['Drama']	0.04	0.91	5.66
['Politics'] => ['News']	0.12	0.85	4.67
['Interview', 'Politics'] => ['News']	0.09	0.88	4.87
['STB IPTV', 'Politics'] => ['News']	0.06	0.86	4.71
['TV', 'Politics'] => ['News']	0.05	0.84	4.63
['STB IPTV', 'Interview', 'Politics'] => ['News']	0.04	0.88	4.87
['TV', 'Interview', 'Politics'] => ['News']	0.04	0.88	4.86

Table 11: Strongest rules for subthemes + device type transactions

Table 11 display rules which contain device items paired with subthemes items. This was an expected output because TV and STB IPTV are really present inside the dataset and when paired with the most common subthemes, it makes sense for them to be linked and be frequent itemsets. Again, not all the rules that passed the threshold are being shown.

7. BIBLIOGRAPHY

[1] Li, G., Qiu, L., Yu, C., Cao, H., Liu, Y., & Yang, C. (2020). IPTV channel zapping recommendation with attention mechanism. IEEE Transactions on Multimedia, 23, 538-549.

[2] Milano, S., Taddeo, M., & Floridi, L. (2020). Recommender systems and their ethical challenges. Ai & Society, 35(4), 957-967.

[3] Aggarwal, C. C., & Yu, P. S. (2001). Outlier detection for high dimensional data. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, *30*(2). https://doi.org/10.1145/376284.375668

[4] Yu, C., Ding, H., Cao, H., Liu, Y., & Yang, C. (2017). Follow me: Personalized IPTV channel switching guide. *Proceedings of the 8th ACM Multimedia Systems Conference, MMSys 2017*. https://doi.org/10.1145/3083187.3083194

[5] Pripuzic, K., Zarko, I. P., Podobnik, V., Lovrek, I., Cavka, M., Petkovic, I., Stulic, P., & Gojceta, M. (2013). Building an IPTV VoD recommender system: An experience report. *Proceedings of the 12th International Conference on Telecommunications, ConTEL 2013*.

[6] Lee, E., Ku, J. Y., & Bahn, H. (2014). An efficient hot channel identification scheme for IPTV channel navigation. IEEE Transactions on Consumer Electronics, 60(1), 124-129.

[7] Yang, C., & Liu, Y. (2015). On achieving short channel switching delay and playback lag in IP-based TV systems. IEEE transactions on multimedia, 17(7), 1096-1106.

[8] Zare, S., & Rahbar, A. G. (2016). Program-driven approach to reduce latency during surfing periods in IPTV networks. Multimedia Tools and Applications, 75(23), 16059-16071.

[9] Lim, C. (2021). Examining factors affecting local IPTV users' intention to subscribe to global OTT service through their local IPTV service.

[10] Daidj, N., & Egert, C. (2018). Towards new coopetition-based business models? The case of Netflix on the French market. Journal of Research in Marketing and Entrepreneurship.

[11] Cumbicus Naranjo, S. C. (2016). Estudio comparativo entre las plataformas tecnológicas de transmisión IPTV y OTT TV (Over The Top-Tv) para brindar servicios de televisión (Bachelor's thesis, Quito, 2017).

[12] Garcés Bravo, V. (2021). La irrupción de DAZN como plataforma de streaming (Bachelor's thesis).

[13] Cascajosa Virino, C. (2018). De la televisión de pago al video bajo demanda. Análisis de la primera temporada de la estrategia de producción original de ficción de Movistar+. *Fonseca, Journal of Communication*, *0*(17). https://doi.org/10.14201/fjc2018175774

[14] Medina, M; Herrero, M., & Portilla, I. (2019). La evolución del mercado de la televisión de pago y del perfil de los suscriptores. *Revista Latina de Comunicación*, *74*.

[15] Yu, Z., Zhou, X., Hao, Y., & Gu, J. (2006). TV program recommendation for multiple viewers based on user profile merging. *User Modeling and User-Adapted Interaction*, *16*(1). https://doi.org/10.1007/s11257-006-9005-6

[16] Yu, Z., Zhou, X., Hao, Y., & Gu, J. (2006). TV program recommendation for multiple viewers based on user profile merging. *User Modeling and User-Adapted Interaction*, *16*(1). https://doi.org/10.1007/s11257-006-9005-6

[17] Masthoff, J. (2011). Group Recommender Systems: Combining Individual Models. In *Recommender Systems Handbook*. https://doi.org/10.1007/978-0-387-85820-3_21

[18] Cantador, I., & Castells, P. (2012). Group recommender systems: New perspectives in the Social web. *Intelligent Systems Reference Library*, *32*. https://doi.org/10.1007/978-3-642-25694-3_7

[19] Gorla, J., Lathia, N., Robertson, S., & Wang, J. (2013). Probabilistic group recommendation via information matching. *WWW 2013 - Proceedings of the 22nd International Conference on World Wide Web*. https://doi.org/10.1145/2488388.2488432

[20] Kim, N. R., & Lee, J. H. (2014). Group recommendation system: Focusing on home group user in TV domain. *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems, SCIS 2014 and 15th International Symposium on Advanced Intelligent Systems, ISIS 2014*. https://doi.org/10.1109/SCIS-ISIS.2014.7044866

[21] Elmisery, A. M., Rho, S., Sertovic, M., Boudaoud, K., & Seo, S. (2017). Privacy aware group based recommender system in multimedia services. *Multimedia Tools and Applications*, *76*(24). https://doi.org/10.1007/s11042-017-4950-0

[22] Lian, T., Li, Z., Chen, Z., & Ma, J. (2017). The impact of profile coherence on recommendation performance for shared accounts on smart TVs. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10390 LNCS*. https://doi.org/10.1007/978-3-319-68699-8_3

[23] Villavicencio, C., Schiaffino, S., Andres Diaz-Pace, J., & Monteserin, A. (2019). Group recommender systems: A multi-agent solution. *Knowledge-Based Systems*, *164*. https://doi.org/10.1016/j.knosys.2018.11.013

[24] Ma, M., Chen, Z., Ren, P., Ma, J., Lin, Y., & de Rijke, M. (2019). Π-Net: A parallel information-sharing network for shared-account cross-domain sequential recommendations. *SIGIR 2019 - Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. https://doi.org/10.1145/3331184.3331200

[25] Erglis, A., Berzins, G., Arhipova, I., Alksnis, A., & Ansonska, E. (2020). Prototype proposal for profiling and identification of tv viewers using watching patterns. *ICEIS 2020 - Proceedings of the 22nd International Conference on Enterprise Information Systems*, *1*. https://doi.org/10.5220/0009458805710578

[26] Alam, I., Khusro, S., & Khan, M. (2021). Personalized content recommendations on smart TV: Challenges, opportunities, and future research directions. *Entertainment Computing*, *38*. https://doi.org/10.1016/j.entcom.2021.100418

[27] Alam, I., Khusro, S., & Khan, M. (2019). Factors Affecting the Performance of Recommender Systems in a Smart TV Environment. *Technologies*, *7*(2). https://doi.org/10.3390/technologies7020041

[28] Al-Maolegi, M., & Arkok, B. (2014). An improved Apriori algorithm for association rules. arXiv preprint arXiv:1403.3948.

[29] Hegland, M. (2007). The apriori algorithm–a tutorial. Mathematics and computation in imaging science and information processing, 209-262.

[30] Rao, S., & Gupta, P. (2012). Implementing improved algorithm over apriori data mining association rule algorithm 1.

[31] Nasereddin, H. H. (2011). Stream Data Mining. Int. J. Web Appl., 3(2), 90-97.

[32] Crespo, F., & Weber, R. (2005). A methodology for dynamic data mining based on fuzzy clustering. Fuzzy Sets and Systems, 150(2), 267-284.

[33] Srikant, R. (1996). Fast algorithms for mining association rules and sequential patterns. The University of Wisconsin-Madison.

[34] Han, J., Pei, J., & Kamber, M. (2011). Data mining: concepts and techniques. Elsevier.

[35] Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. AI magazine, 17(3), 37-37.

[36] AL-Zawaidah, F. H., Jbara, Y. H., & Marwan, A. L. (2011). An improved algorithm for mining association rules in large databases. World of Computer science and information technology journal, 1(7), 311-316.

[37] Edelstein, H. A. (1999). Introduction to Data Mining and Knowledge Discovery 3rd edition. In *Two Crows Corporation* (Vol. 2).

[38] Agrawal, R., Imieliński, T., & Swami, A. (1993, June). Mining association rules between sets of items in large databases. In Proceedings of the 1993 ACM SIGMOD international conference on Management of data (pp. 207-216).

[39] Halkidi, M. (2000, March). Quality Assessment and Uncertainty Handling in Data Mining Process. In EDBT PhD Workshop (pp. 1-4).

[40] Agrawal, R., & Srikant, R. (1994, September). Fast algorithms for mining association rules. In Proc. 20th int. conf. very large data bases, VLDB (Vol. 1215, pp. 487-499).

[41] Batt, S., Grealis, T., Harmon, O., & Tomolonis, P. (2020). Learning Tableau: A data visualization tool. *Journal of Economic Education*, *51*(3–4). https://doi.org/10.1080/00220485.2020.1804503

[42] Aparicio, M., & Costa, C. J. (2015). Data visualization. Communication design quarterly review, 3(1), 7-11.

[43] Friendly, M. (2008). A brief history of data visualization. In Handbook of data visualization (pp. 15-56). Springer, Berlin, Heidelberg.

[44] Yang, Y., Yuan, Z. Z., Sun, D. Y., & Wen, X. L. (2019). Analysis of the factors influencing highway crash risk in different regional types based on improved Apriori algorithm. Advances in transportation studies, 49, 165-178.

[45] Python Package Index - PyPI. (n.d.). Python Software Foundation. Retrieved from https://pypi.org/project/apyori/