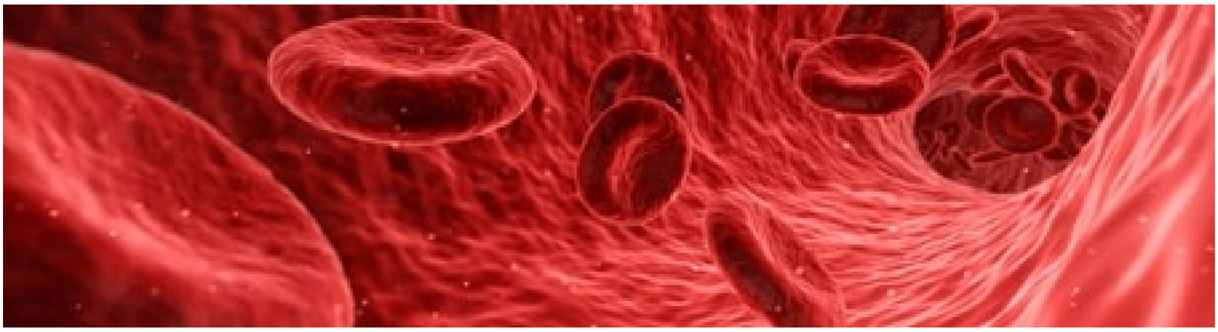


Unravelling Patterns from a Blood Donation Data Set using Machine Learning Approaches

Francisco Jorge Lopes

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Pedro Tiago Gonçalves Monteiro
Prof. Sara Alexandra Cordeiro Madeira

Examination Committee

Chairperson: Prof. Manuel Fernando Cabido Peres Lopes
Supervisor: Prof. Pedro Tiago Gonçalves Monteiro
Member of the Committee: Prof. Rui Miguel Carrasqueiro Henriques

November 2021

Acknowledgments

I would first like to begin by warmly thanking Professors Pedro Tiago Gonçalves Monteiro and Sara Alexandra Cordeiro Madeira for their invaluable help and availability throughout the whole development period of this thesis. Your suggestions and feedback pushed me to sharpen my thought process and brought this thesis to a higher level.

To my family, thank you for always being by my side, even if I was not physically next to you. Especially to my parents, thank you for supporting me throughout my whole academic journey and for always believing in me. I hope I have been and will keep making you proud.

To my friends Bernardo Esteves, Manuel Goulão, Francisco Esteves, João Reis, Henrique Almeida, João Mendes, and Gonçalo Lopes, and especially to Marco Cabral, João Pedro Sousa, and Bárbara Pedro, thank you for all your support, companionship, patience, and motivation. Thank you for helping me in my worst days and for calming me when my nerves were getting the best of me.

I would also like to thank IPST for providing the data set without which this thesis would have not been possible. The database complies with current legislation for the protection of data, in this case, from benevolent blood donors - their identity being unknown. Data were used for academic purposes only.

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference DSAIPA/AI/0033/2019 (project LAIfeBlood) and UIDB/50021/2020 (INESC-ID multi-annual funding).

Abstract

Blood management is a concerning problem for humans. Albeit the existence of technological progress in the field of substitutes for blood products, there will always be a need for whole blood from donors and the derived products that come from it, so an optimization of the available resources should be made. With that in mind, this thesis has the goal of discovering valuable information to optimize operations of IPST, making better use of the existing teams, resources, and daily inventory. For this reason, we start by presenting a background of the most relevant concepts, followed by a review of the state of the art in this domain. Afterwards, we perform an exploratory data analysis of the blood donation data set provided by IPST to have a better understanding of it. This analysis focuses on studying the time-geographic aspect of the donations, as well as the donors that performed them. We then discuss the machine learning approaches that we use to study and find patterns in the data set, and how they were applied. The results obtained highlight the characteristics of the most relevant donations made in 2017, 2018, and 2019, while providing interestingness metrics that are able to support those choices. We also present the results of a more geographically focused study that we performed, in which we discovered the characteristics of the most relevant patterns for each of the Portuguese districts in the same time frame. We end by discussing and presenting the main conclusions of our study.

Keywords

Blood Donation; Data Mining; Network Science; Geographic Patterns; Sequential Patterns

Resumo

A gestão do sangue é um problema preocupante para os humanos. Embora exista progresso tecnológico na área dos substitutos de hemoderivados, haverá sempre a necessidade de sangue total proveniente de doadores e dos produtos derivados que dele provêm, devendo ser feita uma otimização dos recursos disponíveis. Com isto em mente, esta tese tem como objetivo descobrir informações valiosas para otimizar as operações do IPST, fazendo melhor uso das equipas existentes, dos recursos e do inventário diário. Por esta razão, começamos por apresentar uma contextualização dos conceitos mais relevantes, seguido de uma revisão do estado da arte neste domínio. Posteriormente, realizamos uma análise exploratória de dados do conjunto de dados de doações de sangue fornecido pelo IPST para um melhor entendimento do mesmo. Esta análise foca-se em estudar o aspeto geográfico-temporal das doações, bem como os doadores que as realizaram. Em seguida, discutimos as abordagens de aprendizagem automática que usamos para estudar e encontrar padrões no conjunto de dados, e como estas foram aplicadas. Os resultados obtidos destacam as características das doações mais relevantes realizadas em 2017, 2018 e 2019, ao mesmo tempo que fornecem métricas de interesse capazes de suportar essas escolhas. Apresentamos também os resultados de um estudo que realizámos, focando mais o aspeto geográfico, e onde descobrimos as características dos padrões mais relevantes para cada um dos distritos portugueses, no mesmo período de tempo. Finalizamos com a discussão e a apresentação das principais conclusões do nosso estudo.

Palavras Chave

Doação de Sangue; Prospeção de Dados; Ciência das Redes Complexas; Padrões Geográficos; Padrões Sequenciais

Contents

List of Figures

List of Tables

Acronyms

ANN Artificial Neural Network

ASIS Aplicação de Sistema de Informação de Sangue

CART Classification And Regression Tree

CST Centro de Sangue e da Transplantação

ER Estabrook-Rogers Similarity Function

FPM Frequent Pattern Mining

GEM Ichino-Yaguchi Generalized Minkowski Metric

GOW Gower's Similarity Function

HEOM Heterogeneous Euclidean-Overlap Map

HVDM Heterogeneous Value Difference Metric

IPST Instituto Português do Sangue e da Transplantação

ISBT International Society of Blood Transfusion

KDD Knowledge Discovery from Data

MLP Multiple Layer Perceptron

PCA Principal Component Analysis

PNN Probabilistic Neural Network

PSN Patient Similarity Network

RBF Radial Basis Function

SVM Support Vector Machine

1

Introduction

Contents

The Instituto Português do Sangue e da Transplantação (IPST)[1] is a public Portuguese institute of the Ministry of Health and the recognized authority for the collection and regulation of blood donation and transplantation at the national level. Its headquarters are located in Lisboa and its operations are divided into three regional centers (south, center, and north). Each regional center has the same characteristics, having its own finite resources (e.g. personnel and equipment).

IPST is dedicated to supporting human life through areas of intervention transversal to the whole medical and surgical activity, by ensuring the sustainability of health care, assuring the supply of blood and its components, as well as the cells, tissues, and organs for transplantation. It is divided into three main Centros de Sangue e da Transplantação (CSTs) (or Blood and Transplant Centers, in English), one in Lisboa, another in Porto, and one in Coimbra. Its mission is to ensure and regulate transfusional medicine and transplantation activity on a national level, as well as to ensure the offer, collection, analysis, processing, preservation, storage, and distribution of human blood, blood components, organs, tissues, and cells of human nature. IPST's vision is translated to promoting the offering as a gesture that is transversal to the whole IPST's activity with the goal of contributing to human life on time and quality and to do so ensuring that good practices and innovation accompany the state of the art [5].

Blood donations can be made at hospitals, schools, companies, fire stations, etc. Upon collection, each blood unit is tagged with a unique number and associated bar code, according to the International Society of Blood Transfusion (ISBT) 128 norm, which is used to track information, from collection to transfusion (e.g. blood type, collection time, and location). IPST is able to keep track of this information at the national level with a dedicated software system called Aplicação de Sistema de Informação de Sangue (ASIS), which has operated since 2002 and contains historical data for blood activity in Portugal. Over the years, hospitals have been integrating their information systems with ASIS, to enable their communication of collections, storage, distribution, and transfusions. Currently, only 10-20% of national blood collections are not integrated with ASIS, since 5 hospitals have not yet implemented the proper automatic reporting procedures from their own information system into ASIS. Nevertheless, being the recognized authority, IPST still has access to periodic reports from these hospitals and is thus able to assess national collections and needs [6]. Accordingly, every year IPST creates:

- An **activity plan**, which defines the adopted strategies, ranks options, and plans actions and mobilization of resources for that specific year. In the medium-term, this plan is made based on annual targets set at the national level, driven by the contracts with donors' associations and planned events (e.g. festivals, World Youth Day, etc.), and in the short-term, the collection dates and locations are fixed, and the logistics are determined;

- An **activity report**, that talks about what decisions were taken, points out the deviations that were made from the initial plan, evaluates the results, and structures the information that might be

[1] http://ipst.pt/index.php/pt/

relevant for the future.

In 2014, an emergency plan was also presented, with the goal of minimizing the impact of an accident and/or catastrophe in the fulfillment of IPST's mission regarding blood, which influences the amount of blood components to be stored and readily available [7].

1.1 Problem

Blood and its respective components are needed for different kinds of situations, such as emergency or regular surgeries and other procedures, like the treatment of anemic patients, premature infants, cancer, liver diseases or burn injuries. However, even though there is an activity plan done by IPST, it is impossible to predict exactly the characteristics of blood donations that are collected or how many there will be. This may lead to a lack of blood units with specific requirements, due to which it is not possible to keep the health care system running as planned, or having too many blood units, which end up going to waste since they were not used.

According to the IPST's activity report from 2019 [8], the year was marked by the alignment to some measures and orientations that concern the efficient use of resources, reduction of waste, and inefficiencies. In this context, and with this objective of continuing to improve the efficiency of blood supply operations in Portugal, a project called LAIfeBlood was created with the goal of providing IPST with new tools that are capable of helping with this improvement.

1.2 Goal

As part of the LAIfeBlood project, the goal of this thesis is to first focus on analyzing the historical data made available by IPST (a blood donation data set) and then use them to find patterns. These patterns would then be studied to retrieve information and understand how they can be helpful to IPST. From them, we intend to provide new hints to IPST on how to fulfill its activity plans while increasing the resources and daily inventory for that year by optimizing the operations of IPST teams. Examples of useful hints could be:

- If two locations are close to each other, try to collect blood in both of them on the same day, or be somewhere in between them and ask people to go donate there instead;

- If there are brigades with high attendance in a specific location, try to increase the frequency of blood collections performed by them there over the year, and decrease the amount of time spent on each of them;

- If there are too many donations in a certain place, reduce the number of collections there, while increasing the number of collections in a place where there is a low number of donations, but with potential for a high donation rate;
- If the most relevant brigades for each district are identified, more resources can be provided to them, while reducing the resources given to the less relevant ones.

1.3 Organization of the document

In Chapter 2, we explain the main concepts referred throughout this document with the goal of helping the reader to better understand the different topics and their specific notations.

In Chapter 3, we present the work that has been done in the blood donation and blood management area, and which kind of approaches were applied to what type of data. We also present other relevant approaches in different areas, that served as base to the work done in this thesis.

In Chapter 4, we explore and analyze the data from the IPST data set. We present the main information from two different studies, one regarding the time-geographic aspect of the donations and the other regarding the donors that performed them.

In Chapter 5, we describe the preprocessing done for each of the approaches used, how they were applied, and the results obtained from performing them.

In Chapter 6, we discuss and analyze the main results obtained from approaches used as well as their limitations, and we propose some future work for this topic.

2

Background

Contents

This section has the goal of familiarizing the reader with the topics covered in this thesis, starting with an introduction to those topics and their respective main concepts. It then focuses on the main algorithms used in each topic by presenting a concise explanation for all of them. The chapter begins by talking about network science and how similarity networks can be studied, then it focuses on the topic of data mining and which different types of algorithms it includes, and finally presents a description of terms related to blood management and blood products.

2.1 Network Science

Network science is a field that aims to study and analyze complex networks [9]. A network (Fig. 2.1) is made up of various elements represented by nodes N (or vertices) and the connections made between those elements, known as edges E (or links). Some of the most important concepts regarding networks are presented below.

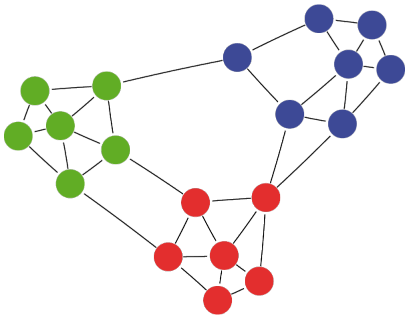


Figure 2.1: Example of a Network, via [1].

A network can be undirected, if all the edges are bidirectional, or directed if its edges have a direction from one node to another. For an undirected network, the degree k of a node i represents the number of edges connected to it, which can be calculated by

$$k_i = \sum_j a_{ij} \, , \tag{2.1}$$

where X_{ij} represents the use of an adjacency matrix. For a directed network, a node has two degrees: the out-degree, $k_i^{out} = \sum_j a_{ij}$, which represents the number of outgoing edges from the node, and the in-degree, $k_i^{in} = \sum_j a_{ij}$, which is the number of incoming edges to the node. Thus, in the case of a directed network, the total degree of a node is represented by

$$k_i^{total} = k_i^{out} + k_i^{in} \, . \tag{2.2}$$

Other characteristics of a network are:

- **Degree Distribution** P_k, which is the probability of having a node of degree k:

$$P_k = \frac{1}{N} \sum_i \delta(k_i - k)\,; \tag{2.3}$$

- **Average Degree** z, that represents the average number of edges for each node in the network:

$$z = \frac{1}{N} \sum_{ij} a_{ij}\,; \tag{2.4}$$

- **Average Path Length** $\langle L \rangle$, which is the average of all the shortest paths, L_{ij}, between all the pairs of nodes of the network:

$$\langle L \rangle = \frac{1}{N(N-1)} \sum_{i \neq j} L_{ij}\,; \tag{2.5}$$

- **Clustering Coefficient** $\langle C \rangle$, that measures how connected nodes are to each other. The clustering coefficient of the network is obtained by averaging the clustering coefficients of all nodes C_i:

$$\langle C \rangle = \frac{1}{N} \sum_i C_i = \frac{1}{N} \sum_i \frac{e_i}{k_i(k_i - 1)/2}\,. \tag{2.6}$$

There are different types of networks depending on the application area, such as social [10], biological [11], spatial [12], etc. This thesis focuses on similarity networks, since the goal is to have the nodes (donors) connected according to their similarities.

2.1.1 Similarity Networks

A similarity network is a type of network in which the edges represent the similarity between two nodes, i.e., if two nodes are similar to each other, then they are connected; if not, then they are not. Similarity can be measured using similarity metrics. However, it is important to keep in mind that the similarity between two nodes is subjective since what is considered similar depends from author to author depending on their own opinions and expertise.

To better address this issue of what can be considered similar, some authors have proposed similarity functions depending on the type of data that is being used. For heterogeneous data sets, the similarity functions need to be different from the ones used for homogeneous ones, like the Euclidean Distance or the Cosine Similarity. Some similarity functions that can be used for heterogeneous data sets are the Heterogeneous Euclidean-Overlap Map (HEOM), Gower's Similarity Function (GOW), Estabrook-Rogers Similarity Function (ER), Ichino-Yaguchi Generalized Minkowski Metric (GEM), and Heterogeneous Value Difference Metric (HVDM) [13].

For this thesis, we will only be focusing on the HEOM [14] since it is able to deal with the type of data considered in this thesis and is simple to implement and to make changes to (see Chapter 5). It consists of a Euclidean distance that treats a variable i differently depending on whether it is nominal or quantitative:

$$HEOM = \sqrt{\sum_{i-1}^{n} h_i(x_i - y_i)^2}\,, \tag{2.7}$$

$$h_i(a,b) = \begin{cases} 1 & \text{if } X \text{ or } b \text{ is a missing value} \\ I_D(a,b) & \text{if the } i\text{th variable is nominal} \\ (|a-b|/rng_i) & \text{if the } i\text{th variable is quantitative} \end{cases}\,, \tag{2.8}$$

where rng_i is the range of the ith variable and I_D is an overlap function.

2.1.2 Community Finding

Before explaining what community finding algorithms do, the concept of community should be explained. In network science, a *Community* is a group of nodes that have a high likelihood of being connected to each other rather than to the nodes of other communities (as represented in Fig. 2.1, where each color represents a different community).

Community finding algorithms focus on grouping communities according to different approaches, such as modularity optimization, which is the one that will be focused on in this thesis, due to its relevance for finding coherent communities.

One of the most relevant metrics in community finding is the *Modularity* M, which measures the strength of the division of a network into modules/communities. This way, modularity can be considered the difference between the number of edges within the communities that are found in a given network and the zero-modularity counterpart:

$$M = \frac{1}{E}\sum_{r=1}^{n} E_{r|net} - \frac{1}{E}\sum_{r=1}^{n} E_{r|random\ counterpart}\,. \tag{2.9}$$

A higher modularity value means that the network has a higher modular structure, which leads to the nodes in each community being more similar.

Some examples of *Modularity Maximization* algorithms are the `CNM` [15], `PL` [16], `WT` [17] and the `Louvain Method` [18]. The latter is a heuristic method for community finding proposed in 2008 by Blondel et al. [18], running in time $O(n\, log\, n)$ in the number of nodes of the network. In the paper in which the `Louvain Method` was proposed [18], it outperformed several other similar algorithms (like the ones mentioned above), not only by having shorter computational time, but also by achieving higher modularity and being able to deal with larger data sets. It works by applying passes. Each pass has 2 steps: Modularity Optimization and Community Aggregation. In the Modularity Optimization step, the modularity is optimized by local changes. Then for each node that joins the community of each immediate neighbor, the change in modularity is calculated. In the Community Aggregation step, a new network is built, in which the nodes are the communities that were found in step 1.

There are other types of Community Finding algorithms besides the ones that focus on Modularity Maximization, like focusing on overlapping communities or divisive procedures, but they will not be explained or looked into. This choice was made since they are not relevant for this thesis because, out of all the different types of algorithms, the `Louvain Method` (a Modularity Maximization algorithm) has lower computational complexity (which allows it to find communities in large networks, like the one that is being dealt with), and lower running time than any other of the studied algorithms, while providing good results in terms of accuracy [19].

2.2 Data Mining

According to Han et al. in [2], ”data mining, also popularly referred to as Knowledge Discovery from Data (KDD), is the automated or convenient extraction of patterns representing knowledge implicitly stored or captured in large databases, data warehouses, the Web, other massive information repositories, or data streams. It incorporates different techniques from various domains, such as machine learning, statistics, visualization, pattern recognition, information retrieval, database and data warehouse systems, etc., allowing for it to meet the need for effective, flexible, and scalable data analysis. This is paramount nowadays to extract knowledge from the large amounts of data that are collected”.

It can be divided into two broad areas: descriptive data mining, using mainly unsupervised learning methods; and predictive data mining, using supervised approaches. *Unsupervised learning* focuses on discovering information (in the form of groups, patterns, etc.) from input data that is not labeled and consists of *Clustering* and *Association* algorithms. *Supervised learning* uses labeled data in the form of a training set, composed of correct pairs of inputs and respective outputs, to predict or classify an outcome given a new input and is, for example, comprised of *Classification* algorithms.

2.2.1 Clustering

Clustering analysis is the most common example of unsupervised learning, which groups a set of data objects into smaller sets depending on how similar they are to each other. These smaller sets are called *Clusters* (Fig. 2.2 from Towards Data Science[1]) and are defined by having elements that are similar between each other, being at the same time dissimilar to the elements that belong to other clusters [2],

[1] https://towardsdatascience.com/semantic-similarity-classifier-and-clustering-sentences-based-on-semantic-similarity-a5a564e22304

which is similar to the concept of a community in community finding.

These algorithms can discover previously unknown groups within data. There are various clustering algorithms, such as `Expectation-Maximization Clustering` [20], `K-means` [21], `Hierarchical Clustering` [22], or `Two-Step` [23]. Each may create different clusters for the same input data. The ones referred to in this thesis are:

- **K-Means** starts by determining the centers of k clusters (centroids) in a random manner, where k has to be defined by a human. Then, using a similarity function (like the Euclidean or the Manhattan Distances [24]), each point of the data set is assigned to the closest cluster, and finally, the mean of all the points in the cluster is calculated to update the respective centroid. This is repeated until there are no changes made to the clusters or a maximum number of iterations has been reached [21];

- **Two-Step** is an algorithm that automatically determines the number of clusters. It starts by doing pre-clustering, dividing the data into a set of subclusters, and then applying hierarchical methods to these subclusters to create the final clusters. It typically employs a log-likelihood distance measure, in which the distance between two clusters is proportional to the decrease in log-likelihood as they are combined into one cluster [23].

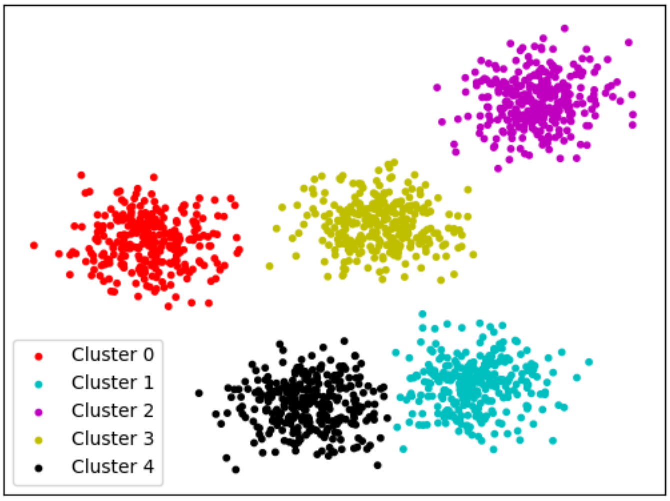


Figure 2.2: Example of Clusters after applying Clustering to a data set, each Cluster is represented by a different color.

There is also the case where, in several applications, we want to simultaneously cluster both the rows and columns of a data matrix. This means that the data analysis entails searching data matrices for sub-matrices that present unique patterns as clusters. This type of clustering approach, where we perform clustering in these two dimensions simultaneously, is part of a category named *Biclustering*.

While clustering generates a global model, biclustering generates a local one. Biclustering algorithms, unlike clustering algorithms, discover groups that show similar activity patterns under a specific subset of the experimental conditions. According to Henriques et al. [25] biclustering has demonstrated particular relevance in applications regarding: 1) expression data analysis (to discover putative transcription modules given by subsets of genes correlated in subsets of conditions [26]); and 2) network data analysis (to unravel functionally coherent nodes [27]).

2.2.2 Pattern Mining

Frequent Pattern Mining (FPM) is an important data mining paradigm that helps to discover *frequent patterns* that conceptually represent relations among discrete entities (or items). Depending on the complexity of these relations, different types of patterns are discovered. [28] There are several types of frequent patterns, for example, frequent itemsets, frequent subsequences (or sequential patterns), and frequent substructures. The most common kind of frequent patterns are *itemsets*, where the relation is the co-occurrence of items that appear together in a transactional data set, which contains data stored in an unstructured format at an atomic level, i.e., no conceptual definition and no data type is defined. An example is a market-basket, composed by a set of items that are bought together by a customer. There are also sequential patterns, which require a temporal or geographic ordering between items. Other examples are time-series data in financial markets, genome sequence data in bioinformatics, etc. All of these scenarios require efficient and flexible FPM algorithms and support data/index structures, which can be reused in a variety of domains.

Pattern mining algorithms are divided into four main categories, these being:

- **Transactional Mining**, or Itemset Mining, which focuses on finding frequently co-occurring itemsets in transactions. For this type of algorithms, the sequence of transactions and the structure of data are not taken into account (unstructured).

- **Sequential Mining**, which has the goal of discovering sequential patterns, given a data set of sequences, where each sequence is an ordered list of transactions (for example, ordered temporally or geographically) and each transaction is a set of items. A sequential pattern is also composed of a list of sets of items.

- **Tree Mining**, that finds different kinds of tree patterns, such as ordered or unordered embedded trees. Its goal is to find all common subtrees in a collection of trees (also called a forest), or even all common sub-forests (disconnected subtrees).

- **Graph Mining**, where the goal is to discover all the commonly occurring sub-graph patterns, given a database of graph objects. However, graphs allow multiple hierarchies, cycles, and arbitrary relations among entities or attributes, in comparison to trees.

However, the ones that will be focused on, for the purpose of this thesis, are solely the *Transactional* and *Sequential* types, as seen in Figure 2.3.

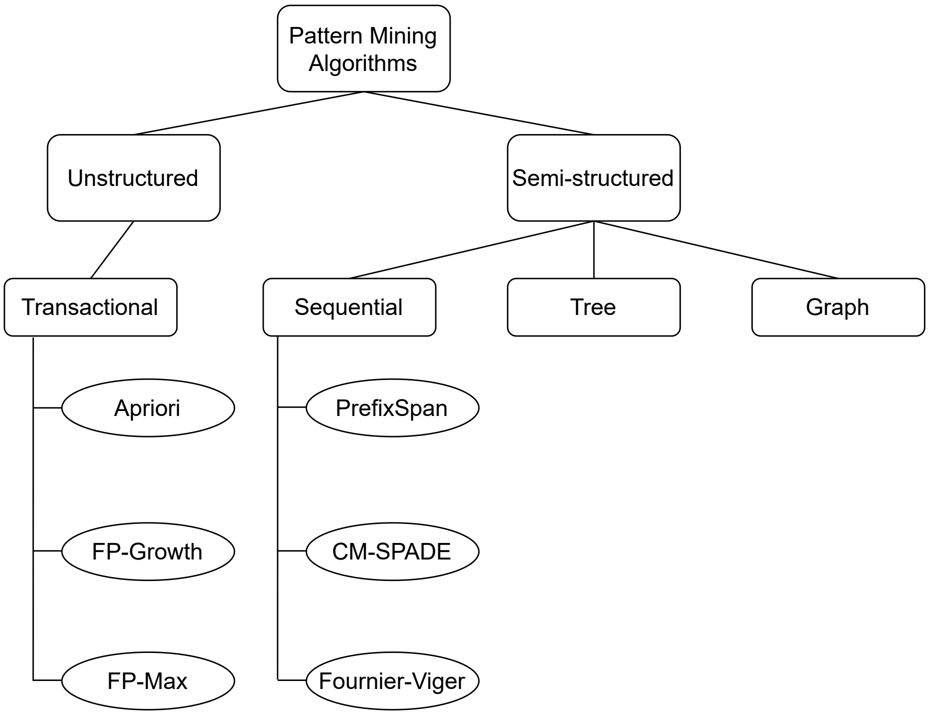


Figure 2.3: Pattern Mining Algorithms Used.

2.2.2.A Common Concepts

Firstly, to better understand the concepts needed, let i be an item and $I = \{i_1, i_2, ..., i_m\}$ be an *itemset*. Let D, the task-relevant data, be a set of database transactions where each *transaction* T is a nonempty itemset such that $T \subseteq I$. Each transaction is associated with an identifier, called a TID. Let X be a *set of items*. A transaction T is said to contain X if $X \subseteq T$.

2.2.2.B Transactional Mining & Association Rule Learning

Transactional Mining, as aforementioned, focuses on finding frequently co-occurring itemsets in transactions. Given a user-specified minimum absolute support threshold, an itemset is considered frequent if it satisfies that threshold. Other two important concepts are the ones of *maximal frequent itemsets* and *closed frequent itemsets*. These are important since a major challenge in mining frequent itemsets from a large data set is the fact that such mining often generates a huge number of itemsets satisfying the minimum absolute support threshold, especially when it is set at a low value. As explained in [2], an itemset X is maximal if it is not a sub-itemset of any other frequent itemset and closed if it does not have a super-itemset with the same support. X is closed in a data set D if there exists no proper super-itemset Y such that Y has the same support count as X in D. If X is both closed and frequent in D, it is a closed frequent itemset in D. X is a maximal frequent itemset in a data set D if X is frequent, and there exists no super-itemset Y such that $X \subset Y$ and Y is frequent in D. For a visualization of these concepts, refer to Fig. 2.4. An itemset can have a single or multiple items. The *absolute support*, or *occurrence frequency*, of an itemset is the number of transactions that contain the itemset in a transaction data set. It was presented by R. Agrawal [29] in 1993. The absolute support of an itemset X is represented by:

$$support(X) = P(X)\,. \tag{2.10}$$

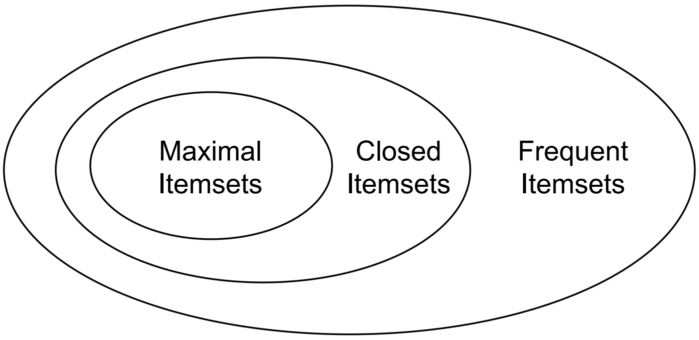


Figure 2.4: Types of Itemsets.

Association Rule Mining is divided into two different steps, which are:

1. Performing transactional mining: where each of the itemsets must occur enough times to meet the user-specified minimum absolute support threshold;

2. Creating association rules based on the frequent itemsets found: where such rules must satisfy the user-specified minimum relative support and minimum confidence thresholds.

An *association rule* is an implication of the form $X_1 \Rightarrow X_2$, where X_1 and X_2 are itemsets, $X_1 \neq \emptyset$, $X_2 \neq \emptyset$, and $X_1 \cap X_2 = \phi$. The rule $X_1 \Rightarrow X_2$ holds in the transaction data set D with *support* s, where s is the percentage of transactions in D that contain $X_1 \cup X_2$ (i.e. both X_1 and X_2). The rule $X_1 \Rightarrow X_2$ has *confidence* c in the transaction data set D, where c is the percentage of transactions in D containing X_1 that also contain X_2. This is taken to be the conditional probability, $P(X_2|X_1)$ [2]. More formally, this can be represented as:

$$support(X_1 \Rightarrow X_2) = P(X_1 \cap X_2)\,. \tag{2.11}$$

$$confidence(X_1 \Rightarrow X_2) = \frac{P(X_1 \cap X_2)}{P(X_1)}\,. \tag{2.12}$$

The support from 2.11 is often referred to as *relative support*, and should not be confused with the aforementioned absolute support (or occurrence frequency) from Eq. 2.10, since this is used to calculate the support of association rules, while the absolute support is for itemsets.

These two measures allow us to identify the *interestingness* of a rule and were introduced by R. Agrawal et al. [29] in 1993. If a certain rule satisfies a minimum support threshold and a minimum confidence threshold specified by the user, then it can be considered as strong. Additional interestingness measures can be applied for the evaluation of association rules, such as the *lift*, *leverage* and *conviction*.

The lift metric is used to calculate how much more frequently the antecedent and consequent of a rule $X_1 \Rightarrow X_2$ appear together than we would expect if they were statistically independent. It was introduced by S. Brin et al. [30] in 1997. The lift between the occurrence of X_1 and X_2 can be measured by computing

$$lift(X_1 \Rightarrow X_2) = \frac{P(X_1 \cap X_2)}{P(X_1)P(X_2)}\,. \tag{2.13}$$

If the resulting value of Eq. 2.13 is less than 1, then the occurrence of X_1 is negatively correlated with the occurrence of X_2, meaning that the occurrence of one likely leads to the absence of the other one. If the resulting value is greater than 1, then X_1 and X_2 are positively correlated, meaning that the occurrence of one implies the occurrence of the other. If the resulting value is equal to 1, then X_1 and X_2 are independent and there is no correlation between them [2].

The conviction was also introduced by S. Brin [30] in 1997. The conviction between the occurrence of X_1 and X_2 can be measured by computing

$$conviction(X_1 \Rightarrow X_2) = \frac{P(X_1)P(\overline{X_2})}{P(X_1 \cap \overline{X_2})}\,. \tag{2.14}$$

If the conviction results in a high value, it means that the consequent depends highly on the antecedent. In the case of a confidence score equal to 1, the denominator becomes 0 (since it would be 1 - 1), and the conviction score will result in ”inf”. Similarly to the lift, if the items are independent, the result of Eq. 2.14 will be 1.

G. Piatetsky-Shapiro [31] introduced the leverage in 1993. It calculates the difference between the observed frequency of X_1 and X_2 appearing together and the frequency that would be expected if they were independent. The leverage between the occurrences of X_1 and X_2 can be measured by computing

$$leverage(X_1 \Rightarrow X_2) = P(X_1 \cap X_2) - P(X_1)P(X_2)\,. \tag{2.15}$$

If the result of Eq. 2.15 is 0, it indicates that X_1 and X_2 are independent.

The most well known and the basis of many other algorithms are `Apriori` and `FP-Growth`. These two, together with `FP-Max`, were used/tested in this thesis and are further explained below:

Apriori was proposed in 1994 by R. Agrawal and R. Srikant [32]. It uses prior knowledge of frequent itemset properties, hence its name. Defining the number of items in an itemset as k, an itemset of size k is called k-itemset. Let the set of frequent itemsets of size k be F_k and their candidates be C_k, and both maintain a field, support count.

The algorithm itself starts by counting the support of individual items (1-itemsets) and which of them are considered frequent according to the minimum support defined by the user. A subsequent pass consists of two phases. First, the frequent itemsets F_{k-1} found in the $(k-1)$-th pass are used to generate the candidate itemsets C_k. Next, the data set is scanned and the support of each candidate itemset in C_k is counted. At the end of the pass, the candidate itemsets that are actually frequent are determined, and are used for the next step. This process continues until no new frequent itemsets are found [33].

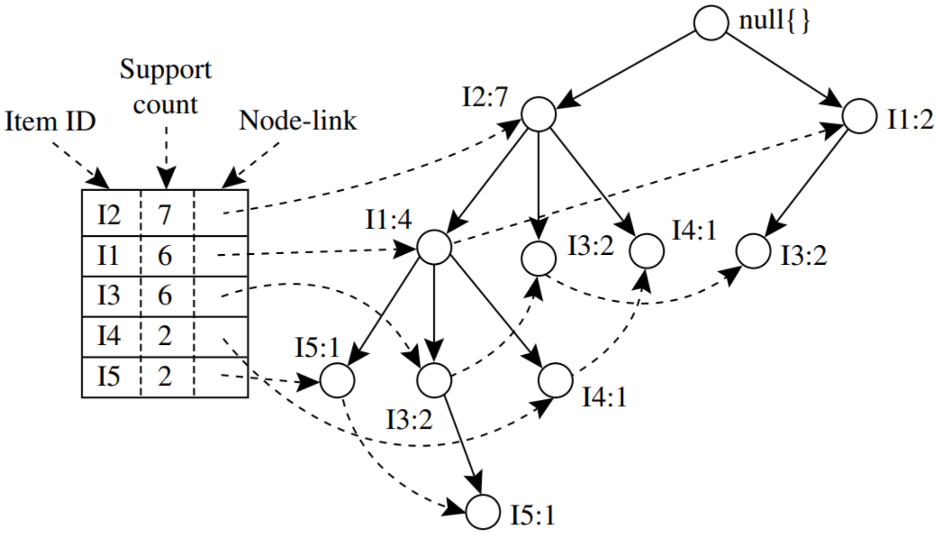
By convention, `Apriori` assumes that items are sorted in lexicographic order, because despite the usually high number of items, the most expensive operation is to count the support of each candidate. This happens since each iteration k involves a passage through all transactions, with each transaction being checked to see if each candidate of size k is contained in it. So, as a way to optimize the process, the algorithm goes through the item in lexicographical order and follows the *anti-monotone* property, where an itemset is not considered to be frequent if any of its subsets is not frequent.

In the join step, candidate generation is done by crossing two frequent itemsets, so F_{k-1} is joined with F_{k-1}. Taking advantage of the lexicographical order of the items, a new candidate of size k only derives from the crossing of two frequent $(k-1)$-itemsets if the two itemsets have the same sequence when the last element is removed, and if the last item from the first set is lexicographically smaller than the last item from the second set. The created candidate has the maximum prefix shared, followed by the last item from the first and second sets. In the prune step, using the anti-monotone property, all itemsets $c \in C_k$ for which some $(k-1)$-subset is not in F_{k-1} (meaning that it is not frequent) are deleted [33].

FP-Growth or Frequent Pattern Growth was presented by J. Han, J. Pei, and Y. Yin in 2000 [34]. Additional progress was presented by the same authors, together with Runying Mao, in 2004 [35]. The algorithm makes use of a divide-and-conquer strategy to decompose both the mining tasks and the data sets. Its goal is to avoid the generation of candidates and inherent verification, as well as costly scans of the data set.

To better understand this algorithm, the concept of *FP-Tree* (or frequent-pattern-tree) needs to be understood. An FP-Tree is an extended prefix-tree structure that retains crucial and quantitative information regarding the frequent patterns. The nodes in the tree are formed by frequent items with length-1, and they are arranged in a manner that the ones that occur more frequently have higher chances of sharing nodes than the ones that appear less frequently. It also has a root node labeled as ”null”, a set of item-prefix subtrees where each node has the support represented by the portion of the path reaching this node, as well as a label and a reference to the next node, and finally a frequent-item-header table (or FP-tree header). The FP-tree header table contains two fields, one for frequent items and the other for a pointer pointing to the first node in the FP-tree carrying the item. Additionally, every node of the FP-Tree referring to the same item can be connected by the order of their creation. Thus, it is easy to find every pattern with the same item, just start with the corresponding node in the header table.

As for the algorithm itself, it begins by condensing the database representing frequent items into an FP-tree (Fig. 2.5), which facilitates counting the support of each itemset. Then, it divides this condensed database into a set of conditional databases, where each has been associated with one frequent item (also called “pattern fragment”), and then mines each database separately. So, for each “pattern fragment”, only the data sets associated with it need to be examined. In this way, such an approach may reduce the size of the data sets to be searched substantially, as well as reduce the “growth” of the patterns being examined. Another important aspect is the fact that when using an FP-Tree, in order to find the patterns, it is not necessary to scan the database, it is only necessary to scan the FP-Tree.



Item ID	Support count	Node-link
I2	7	
I1	6	
I3	6	
I4	2	
I5	2	

Figure 2.5: FP-Tree registering condensed, frequent pattern information, via [2].

With this strategy, `FP-Growth` is able to mine interesting patterns efficiently, even if they have difficult non-anti-monotonic constraints. This is an improvement over `Apriori`, where the candidate set generation is costly, since it creates numerous candidates (mainly when there are prolific and/or long patterns). As referred in [2], a study made about the `FP-Growth`’s performance showed that it was efficient and scalable for mining both long and short frequent patterns, and was around an order of magnitude faster than the `Apriori` algorithm.

FP-Max was described by G. Grahne and J. Zhu in 2003 [36]. This algorithm is an extension of the previously mentioned `FP-Growth`, but it focuses on being highly efficient for maximal itemset mining. So, like the `FP-Growth` algorithm, `FP-Max` is also recursive. It also starts with an initial call, where an FP-tree is constructed from the first scan of the database. The items that compose the current call's conditional base are stored in a linked list called Head. Before recursively invoking `FP-Max`, it is known that the set containing all items in the Head and the items in the FP-tree is not a subset of any existing Maximal Frequent Itemsets (or MFI). If the FP-tree contains only one single path, that one path, together with the Head, is an MFI of the database. Then, the MFI-tree data structure is used to keep track of all the MFI's. If the FP-tree cannot be considered a single-path tree, then for each item in the header-table, that item is attached to the Head, and the subset checking function is invoked to see if the new Head, along with all frequent items in the Head-conditional pattern base is a subset of any existing MFI. If this is not the case, `FP-Max` will be called recursively.

2.2.2.C Sequential Pattern Mining

Whereas, a frequent itemset typically refers to a set of items that often appear together in a transactional data set, like items that are frequently bought together in markets by many customers. A frequently occurring subsequence, such as the pattern that customers, tend to purchase first a certain item i_1, followed by a different one i_2, and then another one i_3, is a (frequent) sequential pattern. This is where Sequential Pattern Mining is useful.

Agrawal and Srikant proposed the problem of sequential pattern mining in 1996 [37], as the challenge of mining interesting subsequences in a set of sequences. Although it was developed to be applied to sequences, it can also be applied to time series once discretization techniques are used to convert time-series to sequences. In data mining, there are two commonly used types of sequential data: *time-series* and *sequences*. A sequence is an ordered list of nominal values (symbols), whereas a time-series is an ordered list of numbers. For example, time-series are used frequently to represent data like temperature readings, stock prices, and electricity consumption readings, whereas sequences are frequently used to represent data like sentences in sequences of items purchased by customers in retail stores, texts (sequences of words), and sequences of webpages visited by users. As so, sequences are the type being focused on in this thesis.

A *sequence* is an ordered list of itemsets $s = \langle I_1, I_2, ..., I_n \rangle$ such that $I_k \subseteq I$ $(1 \leq k \leq n)$. A sequence $s_a = \langle A_1, A_2, ..., A_n \rangle$ is said to be of length k or a k-sequence if it contains k items, or in other words, if $k = |A_1| + |A_2| + ... + |A_n|$. A sequence database SDB is a list of sequences $SDB = \langle s_1, s_2, ..., s_p \rangle$ having sequence identifiers (SIDs) $1, 2...p$. A sequence $s_a = \langle A_1, A_2, ..., A_n \rangle$ is said to be contained in another sequence $s_b = \langle B_1, B_2, ..., B_m \rangle$ if and only if there exist integers $1 \leq i_1 < i_2 < ... < i_n \leq m$ such that $X_1 \subseteq B_{i1}, A_2 \subseteq B_{i2}, ..., A_n \subseteq B_{in}$ (denoted as $s_a \sqsubseteq s_b$) [38].

These algorithms usually assume that there exists a total order of the items denoted as $\succ$, which represents the order in which items in a database should be processed to detect sequential patterns (for example, the order of processing items could be lexicographical, $c \succ b \succ a$). However, any other total order on items from I, like the order of increasing or decreasing support, may be used. It is important to note that the choice of the order $\succ$ has no bearing on the outcome of a sequential pattern mining algorithm. The order $\succ$ is used to ensure that algorithms explore potential sequential patterns following an order in particular, this way avoiding the consideration of the same pattern more than once. The search space of sequential patterns is explored by every sequential pattern mining algorithm using two basic operations called s-extensions and i-extensions, which are used to create a $(k+1)$-sequence (a sequence that has $k+1$ items) from a k-sequence.

The objective of sequential pattern mining is to find interesting subsequences in a sequence database, which are sequential relationships between items of interest to the user. This can be done using a variety of measures for determining how interesting a subsequence is. For the original problem of sequential pattern mining, the measure used was the support. This type of pattern mining can be performed using a variety of algorithms, where the main varying characteristics are the degrees of efficiency and complexity. There are algorithms focused on extracting closed or maximal sequential patterns, similarly to transactional mining. Time constraints are also taken into account by some algorithms, allowing the discovery of sequences with specific lengths or gaps between time points. In general, sequential pattern mining algorithms differ in four main aspects:

- The type of search that they use (depth-first or breadth-first);
- If they use an internal or external type of database representation;
- How are the next patterns that are going to be explored in the search space generated or determined;
- How the support of patterns to see if they satisfy the minimum support is counted.

The sequential pattern mining algorithms considered in this thesis were `PrefixSpan`, `CM-SPADE`, and `Fournier08`. However, it is important to present a few key algorithms that paved the way for the creation of the last two.

PrefixSpan is the most popular pattern-growth algorithm for sequential pattern mining and was presented by J. Pei et al. in 2004 [39]. Before explaining the algorithm itself, we will further explain the concept of a pattern-growth algorithm. Pattern-growth algorithms, in addition to the breadth-first search algorithms and vertical algorithms, are another essential type of algorithm for sequential pattern mining. Pattern-growth focuses on a depth-first search approach and was created to solve a shortcoming of the other two types of algorithms, namely the generation of candidate patterns that might not be present in the database. This is due to the fact that they build candidate patterns by merging smaller patterns, but this process does not involve accessing the database, which is why they might generate patterns that are not present in the database. This issue is overcome by pattern-growth algorithms, since they recursively scan the database to find larger patterns. As a result, they only take into account the patterns that are found in the database. Database scans, on the other hand, can however be costly. Pattern-growth algorithms have developed the notion of projected database to lower the cost of said database scans, which has the goal of reducing the size of databases since larger patterns are considered by the depth-first search.

The `PrefixSpan` algorithm itself works as follows. It uses a depth-first search to explore the search space of sequential patterns. It begins with sequential patterns that one single item and explores larger patterns by appending items to patterns to form larger patterns in a recursive manner. The items are appended to patterns according to a total order $\prec$ (precedent) of items, which might be the lexicographical order or any other total order, to make sure that no patterns are generated twice. `PrefixSpan`'s primary operation is to scan the original sequence database in order to calculate the support of single items and find the frequent items (according to the minimum support threshold). The algorithm then outputs each of those items as a frequent sequential pattern, which are considered and used as seeds to perform the depth-first search. `PrefixSpan`, during the depth-first search, starts by generating the projected database of the pattern s_a, for a given sequential pattern s_a of length k. Then it analyzes the projected database of s_a to count the support of items, with the goal of finding items that can be appended to s_a by i-extension or s-extension to form $(k+1)$-sequential patterns. This approach is then repeated recursively as a depth-first search to discover all frequent sequential patterns.

AprioriAll was proposed by R. Agrawal and R. Srikant in 1995 [40]. It was the first sequential pattern mining algorithm, which was later improved and the basis of a new version, called `GSP` [37]. Both these algorithms took inspiration from the `Apriori` algorithm for frequent transactional mining. Since it was one of the first sequential pattern mining algorithms, the `GSP` algorithm is well-known. However, many algorithms have been proven to be more efficient than `GSP` in recent years, especially due to the algorithm's limitations (such as, multiple database scans, non-existent candidates, and maintaining candidates in memory).

Spade (standing for Sequential Pattern Discovery using Equivalence classes) was presented by M. J. Zaki in 2001 [41] and is a depth-first search alternative to the `GSP` algorithm, that avoids some of the latter's disadvantages. The `Spade` algorithm took inspiration from a frequent itemset mining algorithm named `Eclat` (which was also proposed by Zaki in 2000 [42]). Instead of using a horizontal database representation, it uses a vertical one. This vertical representation shows the itemsets where each item i occurs in the sequence database. This information is referred to as the ID-list of a certain item. After scanning the horizontal database once, its vertical representation (the ID-lists of all single items) can be constructed. It is worth noting that the process of converting a vertical database to a horizontal database can be done the other way around, as the only difference between the two is how the data is stored.

CM-Spade is an improved version of the `Spade` algorithm that has been proposed in 2014 by P. Fournier-Viger et al. [43]. It is based on the observations that `Spade` generates numerous candidate patterns and that the join operation to generate the ID-list is computationally expensive. The algorithm introduced the notion of co-occurrence pruning, focused on reducing the number of join operations. It starts with a database scan to generate a structure called the Co-occurrence Map (CMAP) that contains all frequent 2-sequences. This way, for each pattern s_a (that has been considered by the search procedure), if its last two items are not a frequent 2-sequence, s_a can be removed without having to generate its ID-list, and not having to execute the join operation. In the paper where the algorithm was presented it was reported to have outperformed by more than an order of magnitude the `GSP`, `Spam`, `Spade`, and `PrefixSpan` algorithms, meaning that `CM-Spade` is currently the fastest algorithm.

Fournier08 (or Fournier-Viger) is an algorithm created by P. Fournier-Viger et al. in 2008 [44] that extends the Hirate-Yamana algorithm [45], and integrates it with the `BIDE+` algorithm closure checking [46], while also implementing the BackScan pruning from the latter. This way, it finds only closed frequent time-extended sequences.

To better explain the `Fournier08` algorithm, we present two algorithms that served as initial inspiration for it:

- **Hirate-Yamana** is an algorithm that was presented as an extension of the `PrefixSpan` algorithm by Y. Hirate and H. Yamana in 2006 [45]. Its goal is to find all the frequent time-extended sequences present in a database that respect the minimum support defined. This is done by using a depth-first search. The Hirate-Yamana algorithm also uses the anti-monotone property. It works by projecting a database into a set of smaller projected databases in a recursive manner. This method enables growing patterns one action at a time by identifying locally frequent actions.
- **BIDE+** (or BI-Directional Extension) is also an algorithm that was proposed as an extension of the `PrefixSpan` algorithm by J. Wang et al. in 2007 [46]. It allows checking if a sequence can be considered closed without having to keep track of a set of candidates for closed sequences, unlike many closed pattern algorithms. The BIDE scheme examines if a pattern can be considered closed by looking for one action with the same support that could extend said pattern in the original

sequences that include it. The BackScan pruning of this algorithm allows for it to stop growing certain sequences that are guaranteed not to yield any closed sequences. This BackScan pruning also has the advantage of often outperforming regular sequential pattern mining techniques, like `PrefixSpan`, in terms of time performance.

2.2.3 Classification

Even though no classification algorithms were used in this thesis (as it focused on an unsupervised learning exploratory approach), most of the literature regarding blood donation uses this type of algorithms. For this reason, we decided to shortly present some of the main concepts related to them, especially for us to later explain and make a comparison on why we could not use them due to the type of data in the data set used.

The objective of classification algorithms is to discover functions or models that identify certain classes by using and analyzing a set of training data, where the class label is known, and then using the discovered functions or models to predict how new objects (whose class label is not known) should be labeled or defined [47]. There are several types of Classification Algorithms such as Support Vector Machines (SVMs), Decision Trees, Artificial Neural Networks (ANNs), etc.

Decision Tree algorithms (Fig. 2.6), such as Classification And Regression Tree (CART), `ID3`, and `C4.5` [48–50] are chosen for their interpretability potential. A decision tree is a tree-like chart of decisions and respective consequences, where the nodes represent tests on the value of certain attributes, the branches represent the respective aftermath of such tests and the leaves are the classes or class distributions.

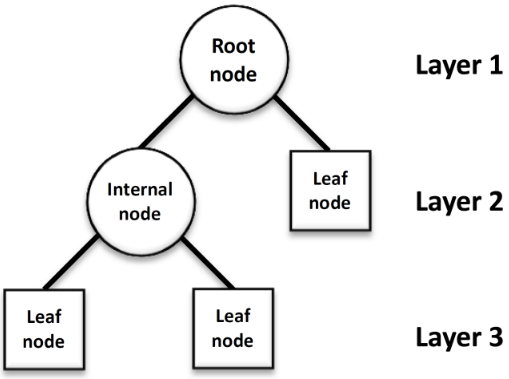


Figure 2.6: Example of a Decision Tree, via [3].

C4.5 is an extension of the ID3 algorithm [49]. It uses a greedy approach and at each iteration it finds the feature with the highest information gain, creates a node and splits its values into different tree branches, thus splitting the training instances/data set. A pruning step can be applied to remove branches that do not contribute significantly to the process, this way improving the tree's ability to generalize to new data [50]. It can produce non-binary trees.

CART builds binary trees, i.e., each node has two outgoing edges. To build the tree, the Gini-index (calculates the probability of a randomly selected feature being classified incorrectly according to the class distribution in the subset) is usually used to do the splitting of the tree at each node until the stopping rules are fulfilled. The resulting tree can also be pruned [48].

Both C4.5 and CART are able to deal with numeric and categorical features and can handle outliers.

2.3 Blood Management

This section focuses on presenting some of the terms related to blood products and blood management. According to Beliën and Forcé [51], who present a ”review of the literature on inventory and supply chain management of blood products”, there are eight classification fields that a person should take into account when referring to this topic:

- **Type of blood products**, such as red blood cells, blood platelets, plasma, whole blood, and frozen blood;
- **Solution Method**, mostly because nowadays, there is a trend for papers involving 'soft computational approaches', like simulation, statistical analysis, or evaluation/best practices. However, 'hard computational approaches', such as linear programming, integer programming, or stochastic dynamic programming, can also be used;
- **Hierarchical level**, which makes a division between individual hospital level, regional blood center level and supply chain level, and where it is shown that from 2000 to 2010 there has been an increase in the publication on papers around this topic;
- **Type of problem**, if it is inbound, which focuses on inventory or collection planning issues, or outbound, where problems with the supply or distribution scheduling are considered;
- **Type of approach**, such as stochastic vs deterministic, and how a stochastic approach is often preferred for this type of environment, since it represents the practical aspect of it more closely by having some inherent randomness;
- **Type of algorithm**, if exact or heuristic algorithms are used, because it depends on the possibility of the problem being optimally solvable or not;
- **Performance measures**, where the two most common ones are those considering the number of outdated units/wastage and the number of units short of demand, but others like transportation costs, availability, quality of the blood, donor donation frequency, might also be addressed;

- **Type of study**, for example if there have been practical implementations or case studies of the problem in question.

Regarding the collection and distribution of blood products, Blake and Hardy [4] explain how it works in Canada for red cell units and evaluate the country's inventory policies for regional blood distribution sites by using a simulation-based methodology that represents it (Fig. 2.7). As a conclusion, they found that a generic modeling framework was useful for regional blood supply chains.

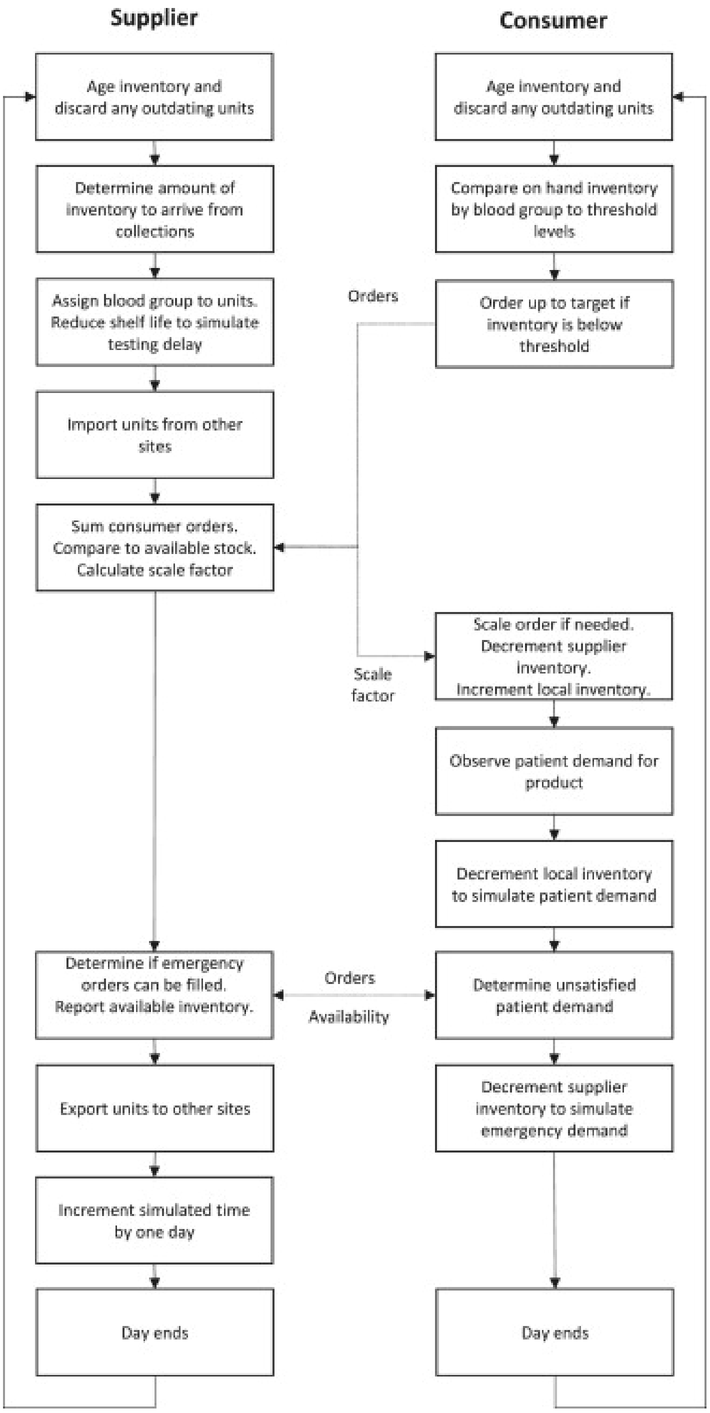

Supplier
Consumer
Age inventory and discard any outdating units
Determine amount of inventory to arrive from collections
Assign blood group to units. Reduce shelf life to simulate testing delay
Import units from other sites
Sum consumer orders. Compare to available stock. Calculate scale factor
Determine if emergency orders can be filled. Report available inventory.
Export units to other sites
Increment simulated time by one day
Day ends
Age inventory and discard any outdating units
Compare on hand inventory by blood group to threshold levels
Order up to target if inventory is below threshold
Orders
Scale factor
Scale order if needed. Decrement supplier inventory. Increment local inventory.
Observe patient demand for product
Decrement local inventory to simulate patient demand
Orders
Availability
Determine unsatisfied patient demand
Decrement supplier inventory to simulate emergency demand
Day ends

Figure 2.7: Conceptual flow diagram illustrating the simulated product flow of Canada's collection and distribution of red cell units, via [4].

3

Literature Review

Contents

This section presents the literature review that was done in order to understand the different types of algorithms that have been used by different authors regarding the topic of blood donation, especially which was the type of data used, how the data was applied to said algorithms, and the results that were obtained. It starts with the network science topic, beginning with a review of papers regarding similarity functions, then similarity networks and finally community finding. Then it focuses on data mining approaches, such as clustering, decision trees, and pattern mining.

3.1 Network Science

The concept of similarity networks is not new, however, with the emergence of the patient similarity network paradigm, it has been brought to the spotlight again [52]. With the goal of adapting this concept and applying it in this dissertation, we did a review of papers regarding similarity functions and networks, as well as community finding. Table 3.1 presents a summary of the papers that were read regarding the state of the art on this topic, relevant to our thesis.

Table 3.1: List of the papers reviewed and their topics related to Network Science.

Paper	Similarity Function	Similarity Network	Community Finding
V. Blondel et al. (2008) [18]			X
A. Carreiro et al. (2013) [53]	X	X	X
S. Harikumar & S. Pv (2015) [54]	X		
S. Klenk et al. (2010) [55]	X	X	
J. Lumijärvi et al. (2004) [13]	X		
S. Pai & G. D. Bader (2018) [52]		X	
D. R. Wilson & T. R. Martinez (1997) [14]	X		

3.1.1 Similarity Function

Most of the papers regarding this topic talk about the distance function. It is important to have in mind that these are different and that a similarity function S corresponds to $S = 1 - D$, where D is a distance function. In this context, Lumijarvi et al. [13] refer five different Heterogeneous Proximity Functions:

- **HEOM**: Proposed in [14] and uses the Euclidean distance and treats nominal and quantitative variables differently, where if the i-th variable is quantitative, an overlap function is used;
- **GOW**: Unlike the HEOM, it uses the Manhattan distance instead of the Euclidean distance, and the values are normalized by definition into the range [0,1] by dividing the sum of similarities by the number of observed value pairs;
- **ER**: Missing and nominal values are treated as in the GOW function, but the Manhattan distance has an upper bound for the quantitative values;

- **GEM**: Based on the Cartesian model, it can compute distances for sets and intervals of attribute values by utilizing the Cartesian join and meet operators. Unlike the other functions shown so far, GEM normalizes the difference in a nominal variable with the size of the value domain;
- **HVDM**: Makes use of the class information to compute conditional probabilities. The function is evaluated for the nominal values with the normalized and simplified Value Difference Metric.

The comparison of the performance of the functions was made using a Nearest Neighbor classifier and a collection of heterogeneous medical data sets. In conclusion, the HVDM outperformed the other metrics for the data sets used. But, we have to keep in mind that this type of algorithm can only be applied when the classes are known [13].

Harikumar [54] proposed another distance metric in the form of a triplet. Before applying the similarity function, the numeric attributes are normalized. The distance between:

- two numerical attributes is calculated with the L1-norm distance;
- two categorical attributes is calculated with a probabilistic approach based on Modified Huang's cost function;
- two binary attributes is done using Hamming distance.

Finally, the distance between two objects is calculated as the sum of all the previous distances calculated.

Klenk et al. [55] proposed the use of a similarity function with a weighting term to calculate the similarity for patient data. This term was assigned to each dimension and corresponds to its influence on the similarity. Besides this weighting factor, there is also a function d_k which could be the absolute, squared, or binary distance, depending on the type of data:

$$d_k(x,y) = 1 \text{ if } x = y \text{ else } d_k(x,y) = 0\,. \tag{3.1}$$

3.1.2 Similarity Network

Pai et al. [52] explain what Patient Similarity Networks (PSNs) are and that they cluster and classify patients based on their similarities in various features, together with the advantages and disadvantages of using them (Table 3.2). In a PSN, the nodes represent the patients and the edge weights reflect the data type similarity. PSNs handle heterogeneous data since using an adequate similarity measure any data type can be converted into a similarity network. The data in PSNs is transformed from the raw values, so the sensitive raw data does not need to be directly used, preserving the privacy of the patients.

Table 3.2: Advantages and disadvantages of similarity networks.

Advantages	Disadvantages
- Interpretable - Handling of missing data - History of success in gene protein function prediction	- New paradigm - Needs improvements on scalability - Only supports categorical outcomes currently

Carreiro et al. [53] use PSNs to find communities of patients. They start by calculating the similarity of patients by treating each type of data differently and the global distance measure is an average of the distance computed for each partition, although it is recognized that a weighted version could be used instead. Then the similarity matrix is used to create a similarity network, used by a community finding algorithm to find the communities.

3.1.3 Community Finding

In the paper presented by A. Carreiro et al. [53], the similarity network was used as an input to Gephi and a community finding algorithm from that software was used to find and analyze the communities. Finally, information regarding the communities was retrieved, such as the features that characterize each one and their importance.

In 2008, Blondel et al. [18] proposed a new method, which was later called the `Louvain Method`. This paper presented and compared it with other community finding algorithms, which were outperformed by it, by having a smaller computational time and achieving higher modularity, as well as being able to deal with larger data sets. This is the method used in this thesis and its results will be presented below.

3.2 Data Mining

The literature regarding data mining on issues related to blood donation is considerably small, but it has been growing in the past few years. We follow an overview of the current state of the art that is relevant to this project. Table 3.3 presents the literature reviewed, as well as the methods used in them.

3.2.1 Clustering

K-Means was used by Ashoori and Taheri in [57] to find clusters of blood donors and study them to identify and describe the donors' behavior. This was done using a data set with 1998 samples and six attributes (age, blood donation status, blood group, gender, the highest education background, and marital status). After applying the `K-Means` algorithm (with value 2 to 6 for the number of clusters), the Dunn's Validity Index was used to calculate the optimal number of clusters. The results were compared with the ones obtained by using the `Two-Step` algorithm.

Table 3.3: List of the papers reviewed regarding blood donation and blood management and their topics related to Data Mining.

	Clustering		Decision Trees			Artificial Neural Networks	Support Vector Machines	Fuzzy Sequential Pattern Mining
Paper	K-Means	Two-Step	CART	C4.5	Others	MLP/PNN	RBF	
M. Ashoori et al. (2016) [56]			X		X			
M. Ashoori & Z. Taheri (2013) [57]	X	X						
W. Boonyanusith & P. Jittamai (2012) [58]				X		X		
M. Darwiche et al. (2010) [59]						X	X	
M. M. Mostafa (2009) [60]						X		
P. Ramachandran et al. (2011) [61]				X				
T. Santhanam & S. Sundaram (2010) [62]			X					
M. Testik et al. (2010) [63]		X	X					
B. Venkateswarlu & G.V.S. Raju (2013) [64]	X							
T. Li et al. (2010) [65]								X

Venkateswarlu and Raju [64] recognize the shortcomings of the `K-Means` algorithm for large data sets. With that in mind, the paper presents an improvement to the `K-Means` algorithm by improving the initial centroids with the distribution of data, obtaining better accuracy and lower computational time than the original one for the same data sets. The size of the data sets used to test the algorithm varies between $1\,000$ and $10\,000$ records. However, no description of what type of information was in the data set was made.

More papers have also presented other sorts of centroid initialization [66–68] that have some perks if used in certain situations, depending on the problem, while also having shortcomings if used in others.

Two-Step was applied in [57] to find clusters of blood donors, allowing the authors to confirm the optimal number of clusters obtained by using the Dunn's Validity Index, since it automatically selects the number of clusters and then proceeded to study the clusters obtained for that number.

Testik et al. [63] used this algorithm with the goal of identifying donor arrival patterns between days and within a day. There was a preference for the `Two-Step` procedure to `K-Means` clustering since, according to the authors of this paper, it is preferred when the data set is large (for example $1\,000$ records) or when there is a mixture of continuous and categorical variables [63]. $1\,095$ records were used in this experiment. The records contained time-wise information regarding the donation (year, month, day of the month, day of the week) as well as the arrival rates of donors. The day of the week and hour of the day were concluded to be two important variables in estimating the arrival rates to the blood center studied, so these two were the ones chosen as input variables for the `CART` algorithm which was later used.

3.2.2 Decision Trees

CART was used in [63] after applying the Two-Step algorithm (as aforementioned). This algorithm was used as a way of gaining insight into the clusters' models and discovering characteristics that distinguished them. This way, ten hourly patterns within three daily patterns were found in total.

Ashoori et al. [56] presented the only paper that compared the use of different techniques of decision trees besides CART or C4.5, such as C5.0, CHAID, and QUEST. Although this paper was not in English, a small paragraph regarding the results was available in English, presenting a final comparison of the results showing that only the C5.0 algorithm had better accuracy than CART.

Santhanam and Sundaram [62] applied the CART algorithm to a data set with the goal of identifying blood donation behaviors according to different types of donors that were previously identified. The data set was composed of the information regarding 748 donors, represented by the number of months since the last donation, the total number of donations, the total amount of blood donated in cubic centimeters, the number of months since the first donation, and a binary variable representing whether they donated blood in March 2007.

C4.5 was used by Ramachandran et al. [61] to classify blood donors. With that in mind, the analysis of a blood donor data set was made using the C4.5 (also known as J48) decision tree algorithm implemented in Weka[1] to develop a system for time analysis of this type of data set. The data set was composed of information regarding 2 387 donors, such as Bag number, name, age, sex, blood group, weight, HIV, etc.

To classify a group of people into donors and non-donors, C4.5 and ANN were used and compared by Boonyanusith and Jittamai [58]. The study was conducted considering the answers made by 400 people to a questionnaire, and the aspects evaluated were altruistic values, knowledge of blood donation, perceived risks, attitudes towards blood donation, and the intention to donate blood.

3.2.3 Artificial Neural Networks

Multiple Layer Perceptron (MLP) was used and compared with C4.5 in [58]. It outperformed the C4.5 decision tree algorithm and was able to classify donors better than non-donors, precision and recall wise.

Probabilistic Neural Network (PNN) was compared with the use of the MLP for the profiling of blood donors by Mostafa [60]. The data set had 430 records, each with information regarding the donor such as sex, age, educational level, altruistic values, perceived risks of blood donation, blood donation knowledge, attitudes toward blood donation, and intention to donate blood. In this paper, it was possible to identify different dimensions of the blood donors' behavior. The results obtained by the ANN meth-

[1] https://www.cs.waikato.ac.nz/ml/weka/

ods were then compared with a standard statistical method (Linear Discriminant Analysis), which was outperformed by both the PNN and the MLP in terms of accuracy rate.

3.2.4 Support Vector Machines

ANN and SVM methods were used in a group of 600 patients by Darwiche et al. [59] who compared the use of an ANN (specifically, a MLP) and SVM (using a Radial Basis Function (RBF)) to study the usability of the latter in the prediction of blood transfusion donation. The information regarding the donor was the number of months since the last donation, the total number of donations, the total of blood donated in c.c., the number of months since the first donation, and a binary variable representing whether they donated blood in March 2007. Firstly, a Principal Component Analysis (PCA) was used to reduce the dimension of the data set, and then it was fed to the aforementioned methods. Conclusively, the SVM approach, especially the Gaussian RBF, performed better than the MLP one and indicates that the time taken was considerably smaller (MLP = 9 days vs SVM = 33 minutes).

3.2.5 Pattern Mining

Even though, to the best of our knowledge, only a single paper applies a Pattern Mining algorithm ([65]), this type of algorithms seemed to be an interesting approach to use on the data set that was used in this thesis. For this reason, the rest of the literature regarding pattern mining focuses on topics other than blood donation, and that served as a basis for what type of approaches were used, such as papers where the algorithms were proposed (Table 3.4), can be represented or were used in another area of expertise.

Table 3.4: List of the papers reviewed regarding Pattern Mining algorithms.

	Algorithm	Proposed by	Paper
Transactional Pattern Mining & Association Rule Mining	Apriori	R. Agrawal & R. Srikant (1994)	[32]
	FP-Growth	J. Han et al. (2000, 2004)	[34,35]
	FP-Max	G. Grahne & J. Zhu (2003)	[36]
Sequential Pattern Mining	AprioriAll	R. Agrawal & R. Srikant (1995)	[40]
	GSP	R. Agrawal & R. Srikant (1996)	[37]
	Spade	AM. J. Zaki (2001)	[41]
	CM-Spade	P. Fournier-Viger et al. (2014)	[43]
	PrefixSpan	J. Pei et al. (2004)	[39]
	Hirate-Yamana	Y. Hirate and H. Yamana (2006)	[45]
	BIDE+	J. Wang et al. (2007)	[46]
	Fournier08	Fournier-Viger et al. (2008)	[44]

3.2.5.A Transactional Pattern Mining & Association Rule Mining

In order to better understand the basics of pattern mining, as well as the most used algorithms in this paradigm, we started by considering the book by Han et al. [2] and the paper by Chaoji et al. [28]. After that, we reached the literature where Transactional Pattern Mining algorithms were proposed, such as Apriori [32], FP-Growth [34, 35] and FP-Max [36], in order to better understand them and decide which ones should be used in this thesis and why.

The representation of the results obtained with Association Rule Mining algorithms is often difficult, since it is mostly based on the values of the support, confidence and lift of each rule and there is a large number of found rules, making it difficult to interpret those results. With this in mind, Michael Hahsler and Radoslaw Karpienko [69, 70] present arulesViz[2], an R-extension package that provides the most popular visualization techniques for association rules, such as grouped matrix-based visualization and graph-based visualization and explains how they can be implemented.

3.2.5.B Sequential Pattern mining

T. Li et al. [65] present a Fuzzy Sequential Pattern Mining approach to discover sub-sequences that were frequent, since the classical sequential pattern mining algorithms are not able to deal directly (without discretization) with numerical data. It was used on a data set with 748 rows (donors) and each record includes the number of months since the last donation, the total number of donations, the total of blood donated in c.c., the number of months since the first donation, and a binary variable representing whether they donated blood in March 2007.

P. Fournier-Viger et al. conducted a survey regarding recent sequential pattern mining algorithms and their utility [38]. It starts by presenting the main concepts and terminology of sequential pattern mining and its main task is defined. Then it focuses on referring the main algorithms and their strategies to solve the problems this paradigm presents. These algorithms include AprioriAll [40], GSP [37], Spade [41], CM-Spade [43], Hirate-Yamana [45], BIDE+ [46] and Fournier08 [44], and it presents the CM-Spade as the fastest currently. Finally, it discusses the limitations that they might have (especially the more traditional ones) and how they have been or might be overcome. Another important aspect is the discussion of open-source implementations, more specifically the ones available at the SPMF data mining library [3] [71–73], which was used to test some of these algorithms for this thesis.

A. S. Martins et al. [74] used transactional mining and sequential pattern mining in a data set of amyotrophic lateral sclerosis patients. This data set contained static data that had been gathered at diagnosis and longitudinal data from the patient's follow-up. The goal of the study was to use the obtained pattern as features in prognostic models, allowing them to take the disease's progression into consideration in predictions and improving the interpretability of the model. The algorithms used were AprioriTID for transactional mining (this way computing the frequent and closed itemsets) and Fournier08 for sequential pattern mining.

[2] https://github.com/mhahsler/arulesViz

[3] https://www.philippe-fournier-viger.com/spmf/

4

Exploratory Data Analysis

Contents

This chapter aims to present a description of the data set used for this thesis. It begins by presenting an overall contextualization of what the data set is composed of, such as the number of rows and columns (or features), and then goes on into a more thorough description of the features and the distribution of the most relevant ones.

4.1 IPST Data Set

The data set made available by IPST is composed of heterogeneous data gathered from 1995 until August 2020, and it consists of $5\,787\,731$ rows and 55 columns (or features). Each row of the data set represents an instance of a blood donation by a donor (from a total of $1\,055\,831$ different donors) that was collected by a specific blood collection brigade. For this reason, blood donation and blood collection have the same meaning in this thesis. Each column has information about the donor, the donation, the association, and the brigade responsible for the blood collection. This information can vary from the blood type of the donor, the date of the donation, to the identifier of the brigade. The data can be divided into five core concepts of the data set that each feature can be attached to. A feature can belong to one or more of these core concepts. This allows us to group the features by them and facilitate how the features are defined and their utility. Below is the description of each of these features and how they can be grouped into these core concepts:

- A *donor* is characterized by the features:
  - 'dador_data_nascimento', the birthdate of the donor (e.g. 2000-12-31 00:00:00);
  - 'dador_sexo', the gender of the donor;
  - 'dador_raca', the race of the donor;
  - 'dador_estado_civil', the marital status of the donor;
  - 'dador_profissao', the donor's job;
  - 'dador_nacionalidade', the nationality of the donor;
  - 'dador_naturalidade_distrito', 'dador_naturalidade_concelho', 'dador_naturalidade_frequesia', the district, council, and parish of the donor's place of birth, respectively;
  - 'dador_codigo_postal_primeiras_posicoes', 'dador_localidade_postal', a number and a location respectively, that together define the donor's zip code (e.g. 1000 Lisboa);
  - 'dador_tipo_sangue_abo', 'dador_tipo_sangue_rh', the blood type group and the RhD respectively, that combined define the donor's blood type (e.g. A +, O -, etc.);
  - 'dador_total_dadivas', the donor's total number of donations in the scope of IPST or not;
  - 'dador_total_dadivas_ipst', the donor's total number of donations only within the scope of IPST.

- A *donors' association* is identified by the pair of features 'associacao_centro_sangue_transplante', the respective CST with SL for Lisboa, SP for Porto, and SC for Coimbra, and 'associacao_id_por_centro_sangue', a unique identifier for each association composed of numbers and/or letters. These features combined create the association's unique total identifier (e.g. SL A32).
- A blood collection *brigade* is identified by the composition of the feature 'brigada_centro_sangue_transplante', the respective CST with SL for Lisboa, SP for Porto, SC for Coimbra and LY for Algarve, together with the feature 'brigada_id_por_centro_sangue', an identifier composed of numbers and/or letters, which has to be different from every other association's identifier, since there can be the same value for the association's and brigade's CST. These two features together form the brigade's unique total identifier (e.g. SL 2PF). It is characterized by the features:
  - 'brigada_tipo', the brigade's type (if it was in a school, hospital, company, etc.);
  - 'brigada_tipo_colheita', defines if the brigade was done in a private space (e.g. provided by a company or a school), or if it was done in space owned by IPST (either fixed or mobile);
  - 'brigada_interna_externa', if the brigade was inside or outside the scope of IPST / organized by IPST;
  - 'brigada_estado_actual', the current state of the brigade, if it is active or not;
  - 'brigada_codigo_postal_primeiras_posicoes', the number of the zip code of a brigade (e.g. 1000);
  - 'brigada_distrito', 'brigada_concelho', 'brigada_freguesia', the district, council, and parish where the brigade is originally from, respectively.
- A *blood collection session* is identified by the three features 'brigada_centro_sangue_transplante', 'brigada_id_por_centro_sangue' which have both already been explained above in the identification of a brigade, and 'colheita_data' that represents the data in which the blood collection was made (e.g. 2000-12-31 00:00:00). It is characterized by the features:
  - 'colheita_nr_dadores_previstos', the number of donors that were expected to show up to the session to donate blood;
  - 'colheita_acesso', represents if the access to the session was public or restricted.
- A *blood collection*, or blood donation, is characterized by the features 'brigada_centro_sangue_transplante', 'brigada_id_por_centro_sangue', which have both already been explained above in the identification of a brigade, 'colheita_data', which was explained above in the identification of a blood collection session. It is characterized by the features:

- 'colheita_tipo', refers to whether the collection made was standard or done in a special way;
- 'colheita_componente_colhido_tipo', representing the blood product that was collected, such as whole blood or blood platelets;
- from 'colheita_reacao_adversa_tipo' to 'colheita_aferese_interocorrencia_outros' presented in Table 4.1, which describe other more specific characteristics of the occurrences or symptoms of the donor to the collection, such as hemorrhages, hematomas, and other complications;
- the pair composed of 'colheita_fase' and 'colheita_estado' shows the phase that a collection is in and its respective state (e.g. T C means approved and waiting collection, E R means disapproved, etc.);
- 'colheita_conclusao', shows the conclusion of the collection and if the blood unit was accepted or not in case of an infection or medication;
- 'colheita_dias_dador_suspenso', the number of days that a donor has been suspended from donating blood, for example, after it has been identified that they have an infection, or they are taking a certain medication.

Finally, the data set has numerous missing values which might be due to the fact that some information can no longer be legally collected since the beginning of the data set's time frame, the data was lost while migrating and upgrading to new storing systems/software or the way the data has been stored (e.g. the features that characterize a blood collection like 'colheita_aferese_interocorrencia_outros' only have any sort of value (by using an "X") when the occurrences are reported, instead of registering them in binary with 0 for when the occurrence did not happen and 1 for when it happened). The distribution of the missing values over the different features can be seen in Figure 4.1. There is also a large number of wrong values, which might be mostly due to human error, for example, by wrongly filling in the information in a questionnaire or by wrongly inputting it in the software.

Table 4.1: Features that characterize a blood collection according to the occurrences or symptoms of the donor.

Feature	Core Concept	Example
colheita_reacao_adversa_tipo	Blood Collection	These variables have an "X" when the situation they represent has been reported, otherwise they are empty
colheita_aferese_reacao_adversa_tipo	Blood Collection	
colheita_reacao_vaso_vagal	Blood Collection	
colheita_reacao_sintoma_local_hematoma	Blood Collection	
colheita_reacao_sintoma_local_hemorragia	Blood Collection	
colheita_reacao_sintoma_local_puncao_arterial	Blood Collection	
colheita_reacao_sintoma_local_outra	Blood Collection	
colheita_aferese_reacao_sintoma_local_infiltracao	Blood Collection	
colheita_aferese_reacao_sintoma_geral_citrato	Blood Collection	
colheita_interocorrencia_2puncao	Blood Collection	
colheita_interocorrencia_coagulo	Blood Collection	
colheita_interocorrencia_baixo_debito	Blood Collection	
colheita_interocorrencia_impossivel_puncionar	Blood Collection	
colheita_interocorrencia_outros	Blood Collection	
colheita_aferese_interocorrencia_2puncao	Blood Collection	
colheita_aferese_interocorrencia_coagulo	Blood Collection	
colheita_aferese_interocorrencia_baixo_debito	Blood Collection	
colheita_aferese_interocorrencia_outros	Blood Collection	

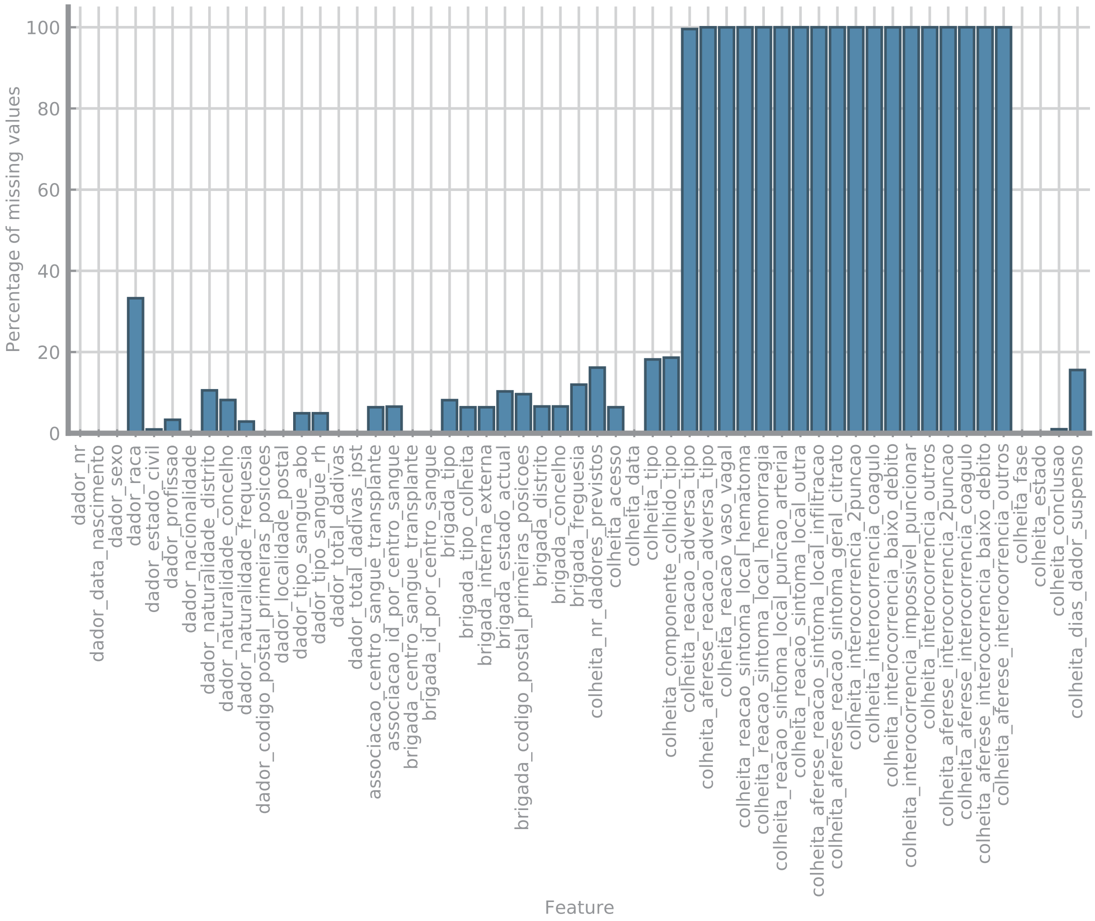


Figure 4.1: Percentage of the missing values over the different features in the data set.

4.2 Data Distribution

To do a general statistical analysis of the data set, we used the Pandas library[1] to study and analyze our data set, and the Matplotlib library[2] to plot the graphs that will be presented below. Firstly, we start by making a *time-geographic* focused study. In the stacked bar chart from Figure 4.2, we can see the total amount of donations for each year together with the distribution of the contributions to them by the three different CSTs (SP, SL, SC as defined in the identification of a donors' association) present in the feature 'associacao_centro_sangue_transplante'. The null bar represents the donations with a missing value in this feature, so they did not have a CST assigned to them in the data set. This division was made since the data set can be organized or divided by CST, which is a logical division of the data to better explore it, since they are independent of each other. Since there was no feature in the data set specifically with the year of the donation (only the respective date), a new feature 'colheita_ano' with the donation's year was created to plot this stacked bar chart, and we had to remove the outliers, since the feature "colheita_ano" contained some wrong values. In this plot, we can see that there was an increase in the number of donations registered by IPST up until 2009, when it began to decrease again until 2020 (please bear in mind that the data for 2020 only goes up until August of that year). This figure also highlights the increase in donations for the Porto CST in comparison to the other two, even though the number of donations country-wide has been decreasing over the last 10 years. Another important aspect to note in this chart is the fact that from the years 2017 until 2019, there has not been much variance in the number of donations.

The bar charts in Figure 4.3 show the distribution of the donations collected by brigades originating in the four districts with the four highest and lowest number of donations from 1995 to 2019 (we did not count 2020 since the data does not represent the complete year). They show that, overall, each of these districts has had a decrease in the number of donations. Another aspect that can be seen is that the districts composing the Porto CST (Porto and Braga) have had a lower decreasing rate (as it was expected) in comparison to Aveiro, which is part of the Coimbra CST or the bottom four, which are part of the Lisboa CST. Which might explain why the Porto CST is the one that since 2010 has been the one maintaining the highest number of donations. We can also observe that the districts with a low number of donations are the ones that compose the southern part of Portugal (Beja, Évora, Faro and Portalegre), and that there have not been any donations from Beja and Portalegre since 2014 and 2015, respectively.

Figure 4.4 presents bar charts that show the distribution of the number of donations collected by brigades originating in every district in each of the last 12 years (not counting 2020 since the data does not represent the complete year). It allows us to better understand how the Porto CST has been

[1] https://pandas.pydata.org/

[2] https://matplotlib.org/

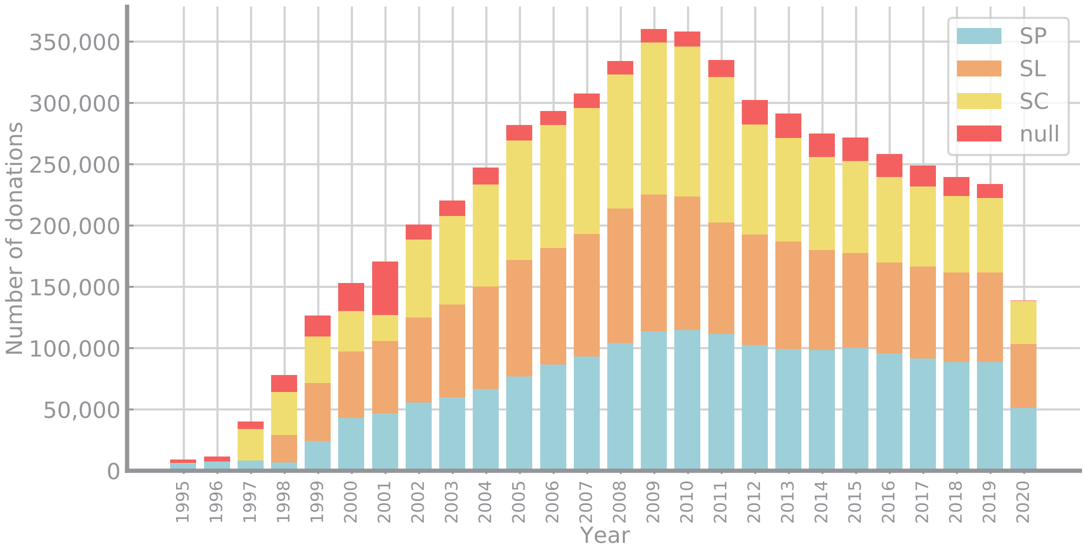


Figure 4.2: Distribution of the number of donations from 1995 to 2020, with the contribution of each CST to the total number of donations for each of those years. Each color represents a different CST. SP is the Porto CST, SL the Lisboa one, and SC the Coimbra one.

maintaining a higher number of donations over the years in comparison to the Lisboa CST, even though the district of Lisboa itself is the one with the highest number of donations every year. As expected after seeing the previous figure, in this one we can again see a low number of donations in the districts that compose the southern part of Portugal and how constant over the years it has been.

The multiple line plot in Figure 4.5 shows the distribution of the donations over the months for each year from 1995 to 2020. Each year is represented by a line with a different color, while having the outliers represented by the lines plotted in gray. The outliers correspond to the years of 1995, 1996, 1997 and 2020. It shows that the months with the highest likelihood of receiving more donations are March, May, October, and November. We can also see that the months that correspond to the summer, like June, July, and August, together with the months that correspond to the winter, mainly January and February, are the ones with the lowest donation rate.

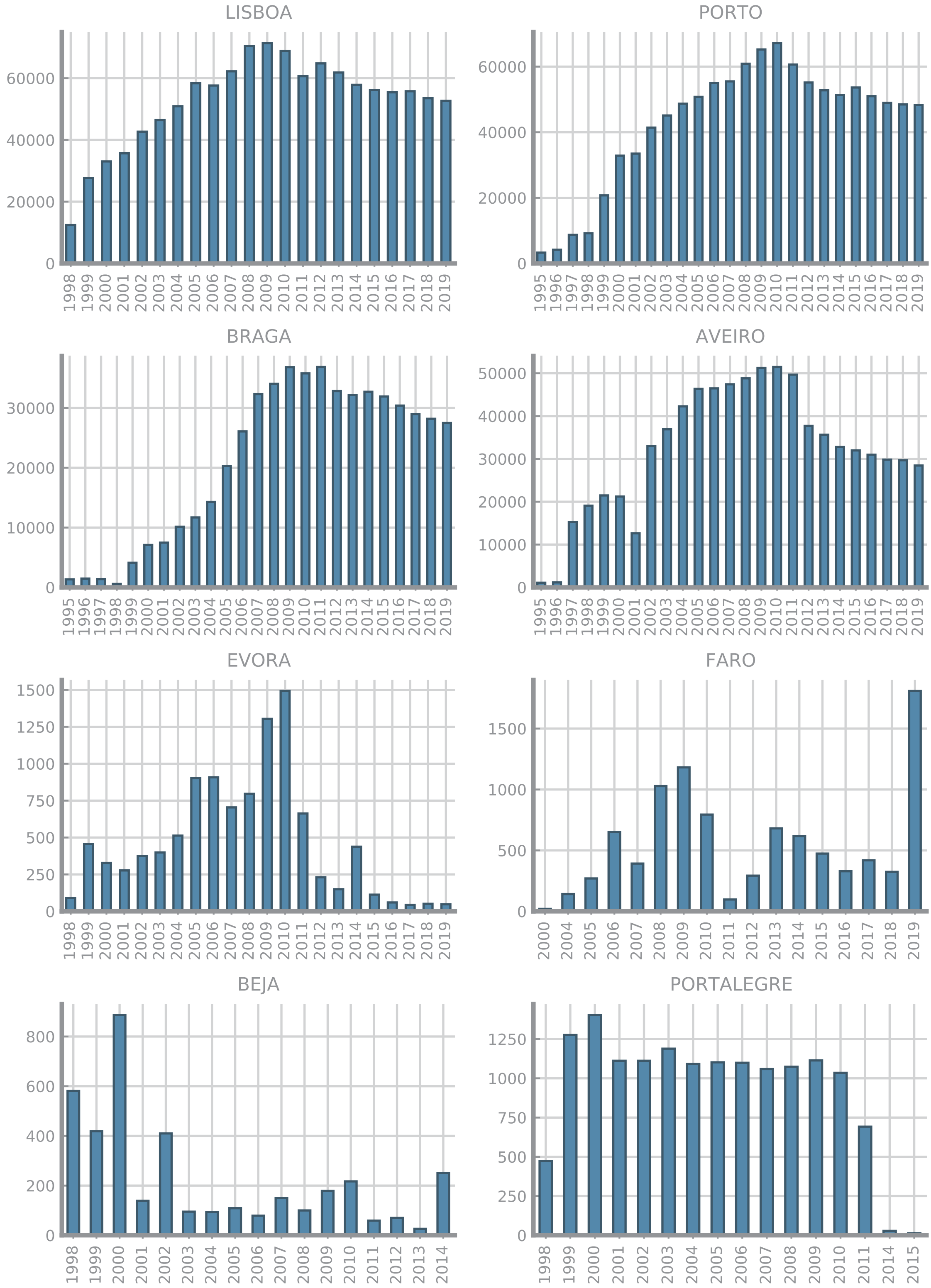


Figure 4.3: Distribution of the number of donations collected by brigades originating in the four districts with the four highest and lowest number of donations from 1995 to 2019.

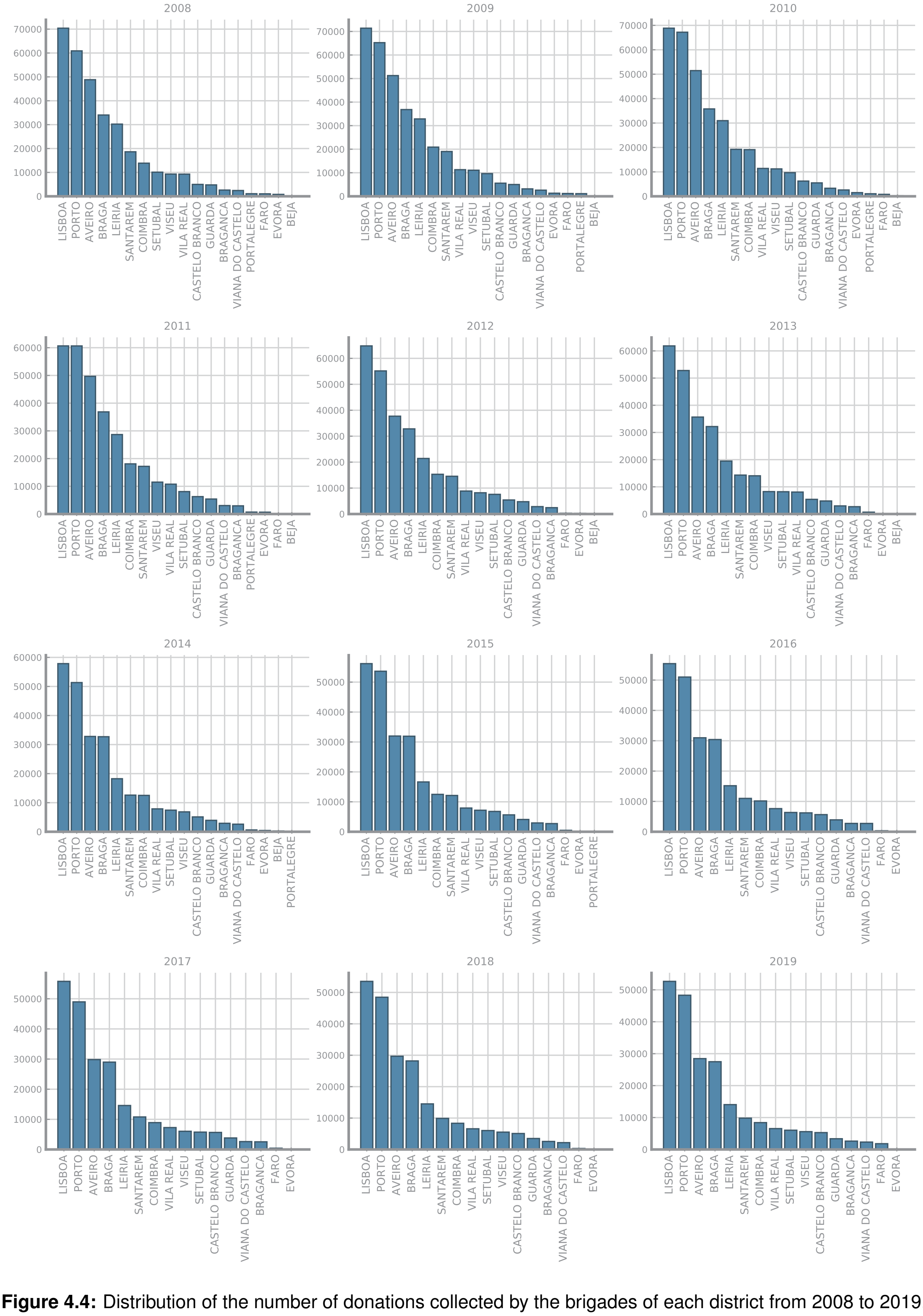


Figure 4.4: Distribution of the number of donations collected by the brigades of each district from 2008 to 2019.

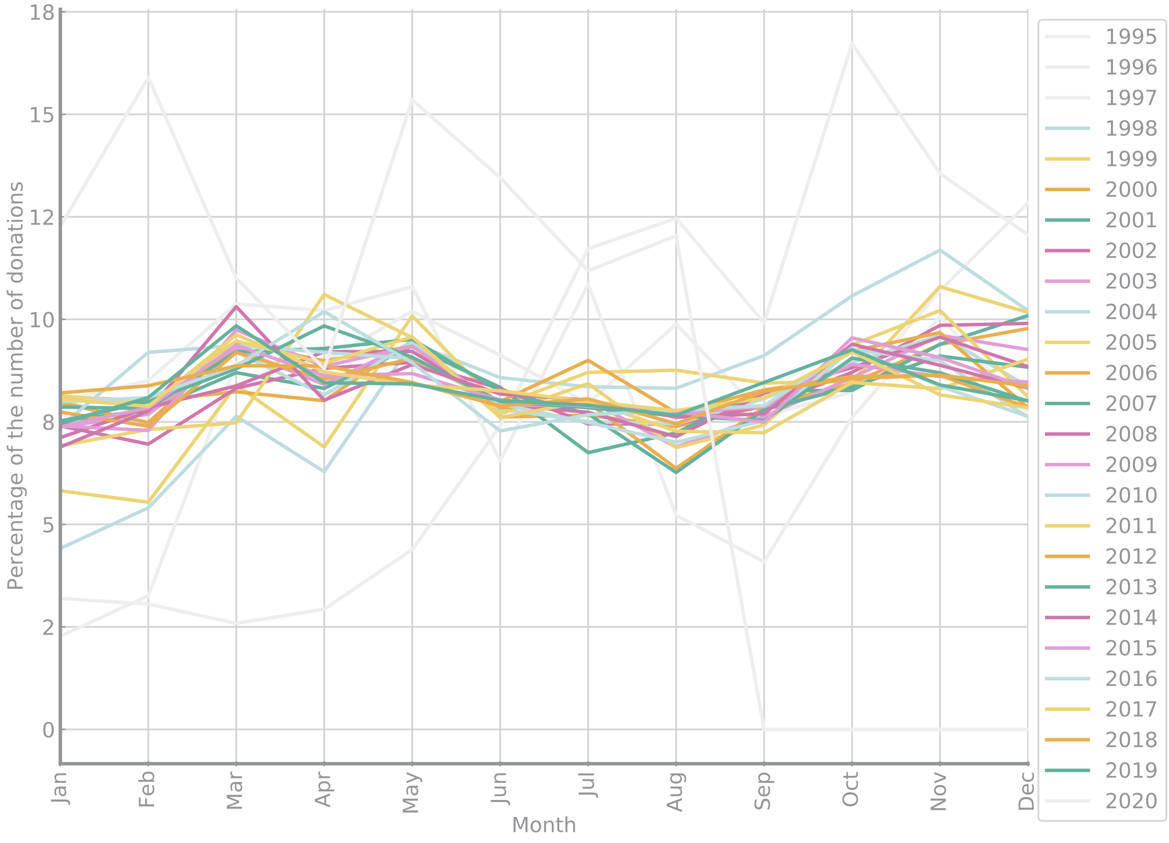


Figure 4.5: Percentage of the number of donations over the months in a year from 1995 to 2020. The lines plotted in gray represent outliers.

Moving on to a *donor* focused study, where we look at the distributions of all the donations according to the different characteristics of the donors that made them (e.g. their birthplace, blood type, or age). The stacked bar graphs below also highlight in orange the contribution of the 2017, 2018 and 2019 time frame, which will be used in the subsequent chapter.

The stacked bar chart in Figure 4.6 studies the district of origin of the brigades and of birth of the donors. From the donor chart (bottom chart), we can see that Porto is the district where most of the donors come from. However, this result is not exactly expected since, as we can see from the brigade chart (top chart), the brigades originating in Lisboa are the ones that receive the highest number of donations. This presents an interesting result: the citizens born in the Porto district are the ones with the highest donation ratio, but they do not all donate blood to brigades originally from the Porto district, showing a large gap between those values. Apart from this fact, it is interesting to see how similar both charts are.

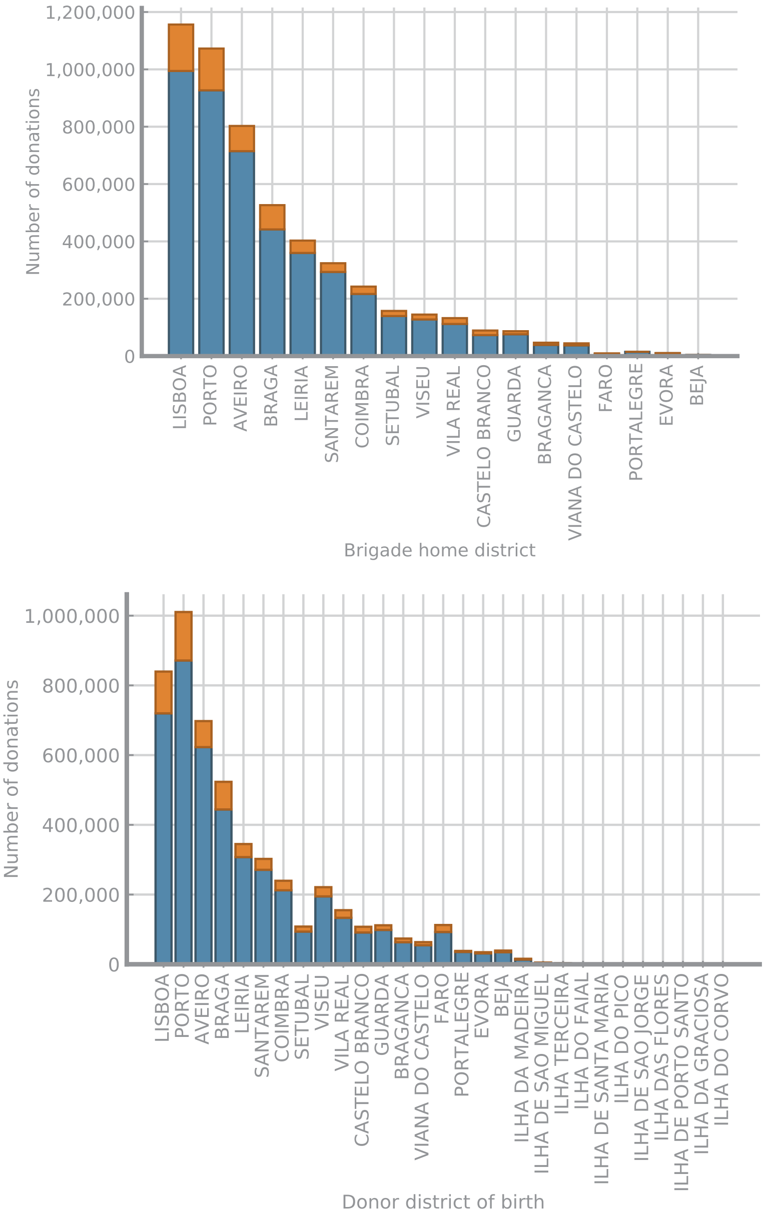


Figure 4.6: Distribution of the total number of donations according to the brigades' home districts (top) and of the districts where the donors are born (bottom). We highlight the contribution of the time frame considered for the subsequent chapter in orange.

Regarding more characteristics of the donors, we can see the bar chart in Figure 4.7. The chart shows us that the most frequent blood types over the entirety of the donations are the expected ones, A+ and O+. There is also a small amount of AB- and B-, which are the rarest types. We can also see that there are a number of wrong values with incorrect terminology (that were compacted into the Wrong bar in the chart), which is composed of the cases where we can see:

- A1, A2, A1B, and A2B for the blood type group feature ('dador_tipo_sangue_abo');

- ? for the RhD feature ('dador_tipo_sangue_rh');
- Missing values in either the blood type group or RhD features.

After contacting IPST, we were informed that A1 and A2 can be determined simply as A, while A1B and A2B can be determined simply as AB. "?" is the same as a missing value in the RhD feature. And any missing values in any of the two features can not be accepted, since we require the pair to represent a blood type correctly.

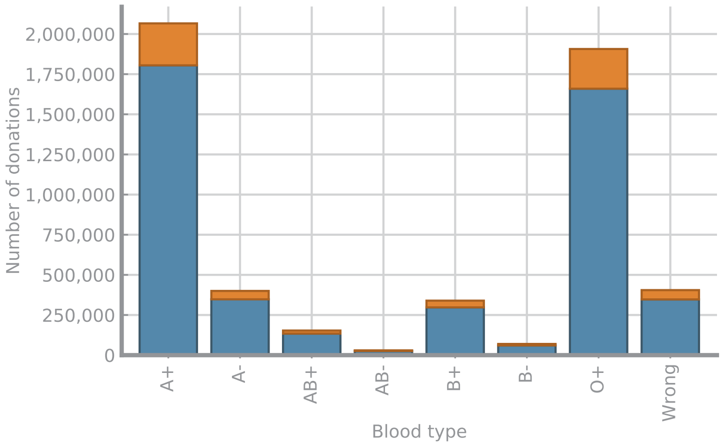


Figure 4.7: Distribution of the total number of donations according to the blood type of the donors. We highlight the contribution of the time frame considered for the subsequent chapter in orange.

In Figure 4.8, we can see the distribution of the donations according to the gender of the donors. It shows that, overall, there are more males donating blood than females. However, in the highlighted portion, there have been slightly more females than males donating.

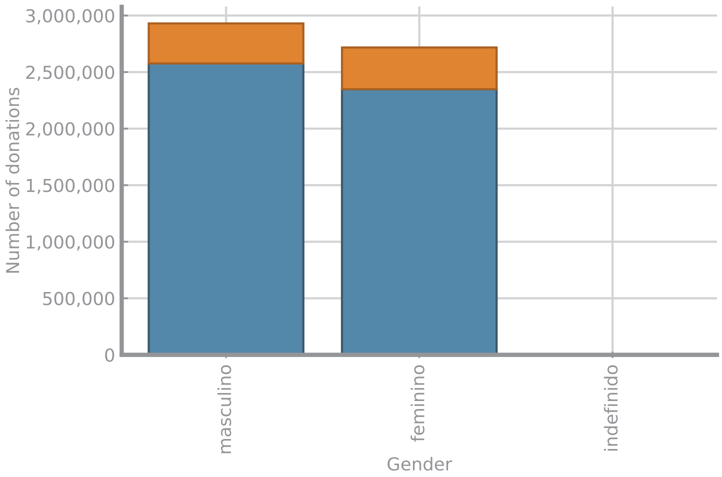


Figure 4.8: Distribution of the total number of donations according to the gender of the donors. We highlight the contribution of the time frame considered for the subsequent chapter in orange.

The stacked bar chart from Figure 4.9 represents the distribution of the total number of donations, taking into account the current age of the donors. The age of the donors was calculated by using the 'dador_data_nascimento' feature and subtracting it from the current date. However, there were numerous incorrect/impossible ages that could not be considered properly and were removed. The minimum age requirement to donate blood is 18 years old, and the maximum is 65 years old. This way, we only present the ages from 18 to 90 years old, since donors that were 65 in 1995 would now be 91 years old. From it, we observe that the age bracket with the highest number of donations is the one where the donors are between the ages of 40 and 50 years old. We can also see from the highlighted portion, there has been an increase in the number of donations by younger donors (from 20 to 30 years old), especially in the 20 to 25 year bracket.

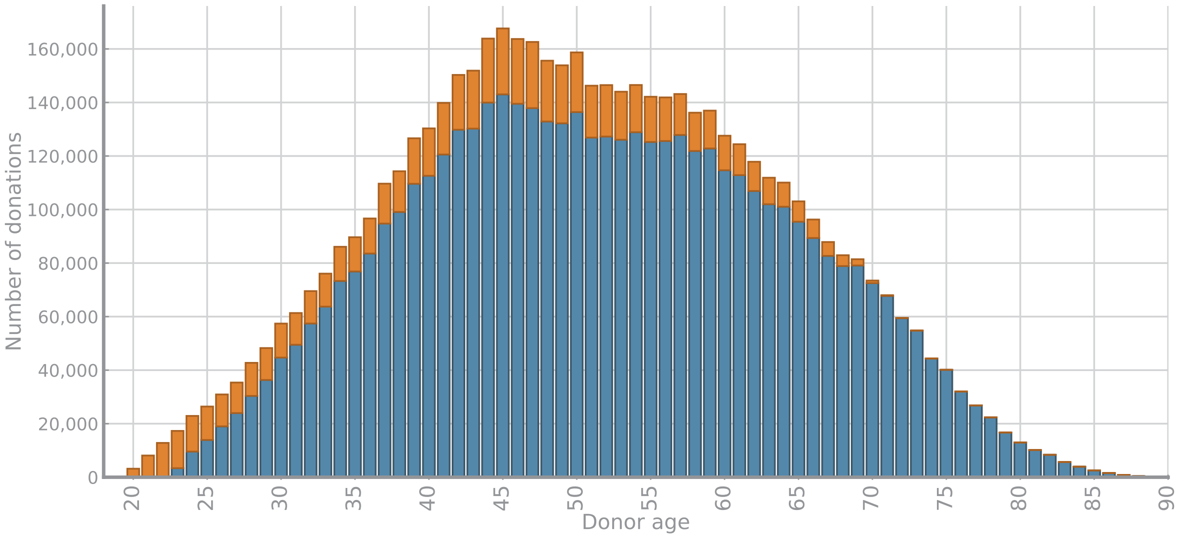


Figure 4.9: Distribution of the total number of donations according to the current age of the donors. We highlight the contribution of the time frame considered for the subsequent chapter in orange.

Ultimately, we have the stacked bar chart from Figure 4.10 which presents the top 10 jobs of the donors. We decided to pick only the top 10 as there are $3\,001$ different values registered in the 'dador_profissao' feature. We can see that overall the most common value for this feature is "DESCONHECIDA" (i.e. unknown), however for the highlighted portion the most common value changes to "EMPREGADO FABRIL" (i.e. factory employee). Please keep in mind that the values in this feature might not be totally up-to-date considering that a donor might have first been registered with a certain job and then never updated it at a later date. Besides this fact, there are two main issues that we can see in this chart and for which we still need an explanation from IPST:

- The high amount of donations with the value "ESTUDANTE" (i.e. student) seems rather odd, considering that there is also the value "ESTUDANTE DO ENSINO SUPERIOR" (i.e. higher education student), and that the minimum age requirement to donate blood is 18 years old and by that age most students would already be studying in a higher education establishment;

- We are unsure as to what ”DESCONHECIDA” is meant to represent, since we do not know in which situations it was used. Some ideas of what the value could represent are people that were still studying, so they did not yet have a concrete job, but there would be the values ”ESTUDANTE” or ”ESTUDANTE DO ENSINO SUPERIOR” to represent that situation, and the case where the donor did/does not have a job, but that condition could be represented with the value ”DESEMPREGADO” (i.e. unemployed).

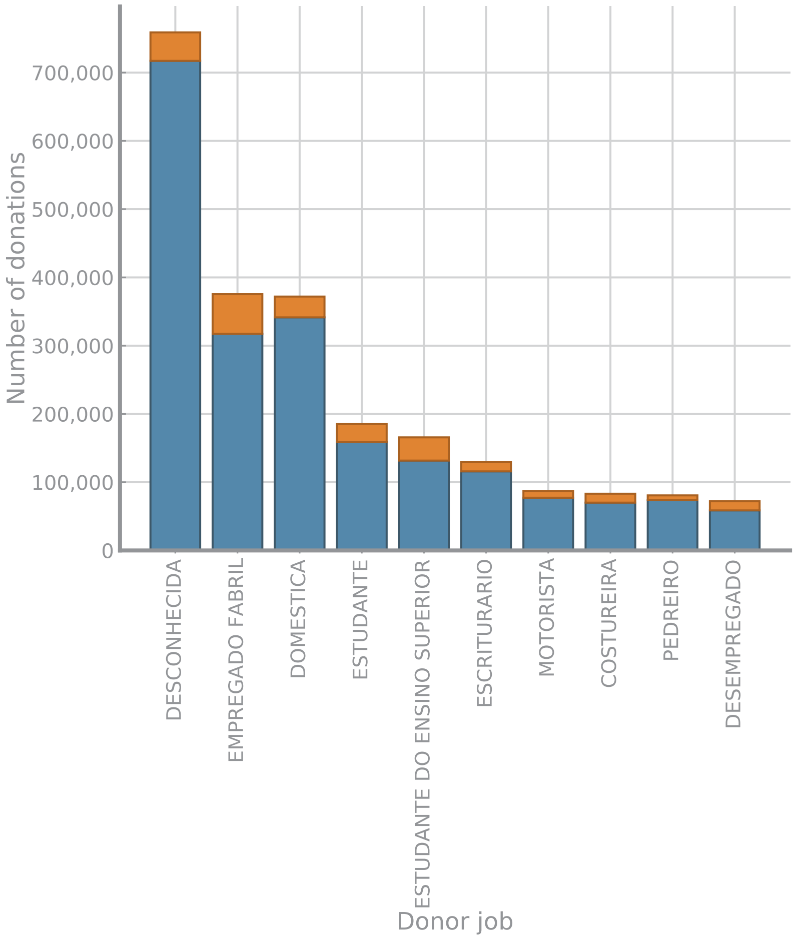


Figure 4.10: Distribution of the total number of donations according to the top 10 jobs of the donors. We highlight the contribution of the time frame considered for the subsequent chapter in orange.

5

Unravelling Patterns

Contents

This chapter starts by presenting an initial data pre-processing that was performed before applying any of the approaches tested in this thesis and used through all of them. Then it is divided into sections, presenting each of the approaches, starting with an explanation of the extra pre-processing that was done in order to run each of the algorithms used, and the results obtained after running them together with the respective discussion.

5.1 Initial Data Pre-Processing

We will now present the initial data pre-processing that was maintained throughout the different approaches used. The remaining data pre-processing details will be explained in each of the subsequent sections, as each approach requires different characteristics in order for the data set to be used.

We started by transforming some of the features present in the data set and removing the wrong values present in them, as follows:

- The feature 'dador_data_nascimento' was transformed into 'dador_idade' to represent the age of the donor as an integer, this way being easier to compare. Then we only kept the rows where the age was above or equal to 18 years old and below or equal to 91. This was done since the minimum age requirement to donate blood is 18 years old and the maximum is 65 years old, and the maximum case would be the one where the donors that donated in 1995 were 65 years old, then now (in 2021) would be 91 years old;
- The feature 'colheita_data' was divided into four different features 'colheita_year', 'colheita_month', 'colheita_week', and 'colheita_day', which represent respectively the year, month, week, and day of the donation. Again, because comparing integers or strings (only for the month, since we chose to represent them as, for example, Jan, Feb, etc.) is easier;
- Correcting the features 'dador_tipo_sangue_abo' and 'dador_tipo_sangue_rh' by transforming the values A1, A2, A1B, and A2B into A, A, AB and AB respectively, and keeping only the rows where the RhD was either "+" or "-", since those are the only two possible values.

None of the approaches was able to process the whole data set (with $5\,369\,765$ rows and 57 features) with acceptable computational time and resources. To address this issue, the data set was divided into different data sets of donations for 2017, 2018, and 2019 (with $239\,414$, $229\,922$, and $223\,545$ rows, respectively). In this way, the criteria for the sampling done was the year, considering that as shown in the figures from Chapter 4, in these three years there is a similar distribution of the data, and we can see that there was a stabilization of the data, so most of the characteristics and distributions of the features from the previous years are maintained in this time frame. From this point on, whenever we are talking about further changes to the data sets, these three are the ones we are referring to.

5.2 Community Finding

In this community finding approach, we used a similarity network and the modularity maximization algorithm named `Louvain Method` [18], which were explained in Section 2.1 from Chapter 2. The respective workflow can be seen in Figure 5.1.

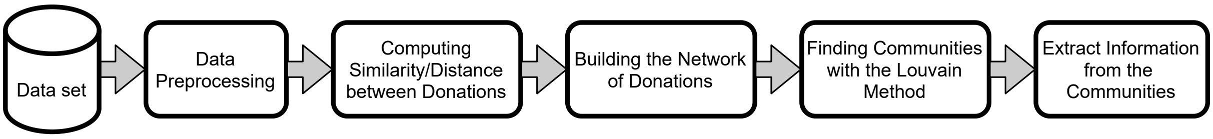

Data set
Data Preprocessing
Computing Similarity/Distance between Donations
Building the Network of Donations
Finding Communities with the Louvain Method
Extract Information from the Communities

Figure 5.1: Workflow for the community finding approach.

5.2.1 Data Pre-Processing

We began by removing the features where the data was unbalanced (a large number of observations belonged to only one class) since to make a similarity network these would often have the same value between two different nodes. This is problematic because when thinking in terms of a similarity function, such similarity will have the same weight as any other, this way biasing the connection between both nodes.

We decided to choose only one of/collapse the features that gave the same information as others. This was mostly because different columns were referring to the location in which the donation was made but with different granularities. With this, the similarity network would have a bias towards two nodes that share the same location, since it is represented by more than a single feature in the data set. For example in the case of the three features that characterize a blood collection's brigade 'brigada_codigo_postal_primeiras_posicoes', 'brigada_freguesia', 'brigada_concelho', 'brigada_distrito', we ended up only choosing 'brigada_distrito', since the district where the brigade was originally from had already a large number of different values to determine the similarity between two nodes.

Then we removed the features that would not add any new information to the network that was going to be created, since they gave only information about the donation that would not be relevant to the similarity between two nodes (for example, 'colheita_dias_dador_suspenso', which represents the number of days until a donor is allowed to donate blood again) or because they had a high percentage of missing values (which we considered as having more than a 20% threshold of missing values).

Finally, the number of features selected had to be small since the algorithm scaled with the number of features, meaning that the more features used, the more computational time and resources the algorithm would require to run. So, after performing the data pre-processing, we were able to select the seven most relevant features:

- ’doacao_nr’, a unique identifier for each donation, since there was no feature that was able to identify a donation by itself, and was used as the identifier for each node in the network;
- ’dador_idade’, allowing us to see if the donors belonged to the same age group;
- ’dador_sexo’, determining if the gender of the donors was the same;
- ’dador_tipo_sangue_abo’, to check if the blood type group of the donors was the same;
- ’dador_tipo_sangue_rh’, to check if the RhD of the donors was the same;
- ’brigada_distrito”, to see if the donors had their blood collected by brigades that belonged to the same district;
- ’colheita_mes’, allowing us to check if the donors donated blood during the same time of the year (same month in this case).

5.2.2 Similarity Network Creation

The similarity network consisted of the relationship between two donations. To create this network, we needed to have a measurement that reflected this relationship. As previously stated, PSNs for a specific disease was studied in [52]. The plan was to apply a similar approach to the blood donation data set to then study the results obtained.

The data set was composed of heterogeneous data, so we needed to use different distance metrics to represent how close the relationship between two nodes was, while having to deal with any missing values. Since each node was represented in the form of tabular data, which has the values of its features:

1. If the data was categorical (for example blood type, gender, and location of the donation), the approach chosen to compare it was to check if the value present in a column X was the same for both nodes. If it was, then they would be considered similar and given the shortest distance possible. If it was not, then the two nodes would have the longest distance between them;
2. If the data was quantitative (the donor’s age being the only case), these ages were compared by intervals of age ranges, like $(-x < Age < x)$ to check if the other donor’s age could be included in that age range.

Then, to combine the data, a global distance was calculated by using an approach to the HEOM [14] for each type of data. This distance function was able to deal with nominal and quantitative data types in a data set. However, how quantitative values were treated was different from the one referred to in Equation 2.8. Since we were dealing with age ranges, it was adapted to:

$$h_i(a,b) = \begin{cases} 1 & \text{if } a \text{ or } b \text{ is a missing value} \\ I_D(a,b) & \text{if the } i\text{th variable is nominal} \\ J_D(a,b) & \text{if the } i\text{th variable is quantitative} \end{cases}, \tag{5.1}$$

where I_D and J_D are overlap functions:

$$I_D(a,b) = \begin{cases} 0 & \text{if } a = b \\ 1 & \text{otherwise} \end{cases} \tag{5.2}$$

$$J_D(a,b) = \begin{cases} 0 & \text{if } abs(a,b) < x \\ 1 & \text{otherwise} \end{cases} \tag{5.3}$$

Weights could also be added to each feature, but more information and knowledge about the problem would be needed.

Towards building the similarity network, due to the size of the data sets, the computational time and RAM needed to run the tools and techniques chosen was too high, even after transforming the three data sets into arrays. This was seen in both the creation of the similarity network (when comparing each row to every other row, to find out if they were similar) and the community finding algorithm (when creating the network from the edge list file given). Meaning that for this algorithm, the data sets of 2017, 2018, and 2019 were still too large to run with acceptable computational time and resources, so after decreasing the number of rows several times, we discovered that in order to be able to run this approach, we needed a random sample of $n = 10\,000$ rows (donations) from each of them.

After computing the distances with the tweaked HEOM (Equation 2.7), the network can be built. However, for network analysis, the most common approach is to use similarities instead of distances. The similarity is calculated by $S(i,j) = 1 - D(i,j)$, where S and D are the similarity and distance between two nodes, respectively. Then, a threshold was defined to determine which links are to be created and added to the edge list file. The edge list file was composed of two columns, one with the starting and the other with the ending node of the link. The threshold was defined at 75%, which would consider two donations to be similar if and only if they had four or more features in common out of the six features ('dador_idade', 'dador_sexo', 'dador_tipo_sangue_abo', 'dador_tipo_sangue_rh', 'brigada_distrito', and 'colheita_mes'). This threshold value was chosen since it reflected our idea of what it meant to be similar. The edge list file was then used as input for NetworkX[1], a Python package for the creation and study of complex networks [75]. From the point of view of the resulting network, the nodes represent the donations and the links represent that the two donations connected are similar (following the threshold that was set).

[1] https://networkx.org/

5.2.3 Results

After using the Louvain method (available in NetworkX [76]) to find communities in each of the three similarity networks, their modularity and giant connected components were calculated respectively. The giant connected component, which is the largest connected group of nodes, is useful since it allows one to filter the nodes of a network by keeping that group while removing the disconnected groups of insignificant size.

Table 5.1: Networks' and Giant Connected Components' (GCC) characteristics given $10\,000$ donations of each year.

	Network		GCC		
Year	#Nodes	#Links	Modularity	#Nodes	#Links
2017	$9\,598$	$1\,543\,913$	0.588	$9\,598$	$1\,543\,913$
2018	$9\,563$	$1\,527\,430$	0.632	$9\,562$	$1\,527\,429$
2019	$9\,594$	$1\,541\,782$	0.646	$9\,593$	$1\,541\,781$

Table 5.1 shows the modularity value and the giant connected component size for the donation similarity networks of 2017, 2018, and 2019, given $10\,000$ nodes. The values obtained for the modularity are considerably good and the GCCs cover almost all the nodes of each network. After studying the group of nodes that were left for each network, none was significant enough and ended being filtered out. With NetworkX and its spring layout, the following plots (Figure 5.2) were obtained by coloring each of the communities with a different color to visualize the results better. From them, it is possible to see that there were 6 different communities created for 2017, 7 created for 2018, and 6 for 2019.

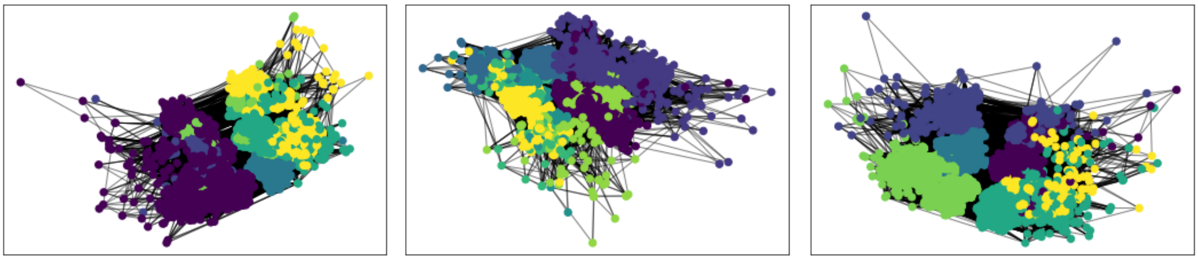


Figure 5.2: Representation of the communities found with NetworkX for the donation similarity networks of 2017, 2018 and 2019, respectively. Different Colors represent different communities.

After finding the communities, it was still difficult to understand which characteristics each partition had in particular and what distinguished each group from each other. To analyze this, a study of which features better explain each community was done. To do so, the features were plotted for each community of each year's network. From their distributions in Figure 5.3, and in Figures A.1 and A.2 from Appendix A, it was possible to see that the three features that better represent each community would be Blood Type, Gender, and District. For the feature:

- **Blood Type**, there was an almost clear separation between two sets of communities, such that over the three years, most of the communities can be divided into either having blood type A+ or O+, since in 2017's partitions 1 and 3, 2018's partitions 1 and 7, and 2019's partitions 4 and 5, there was a higher number of blood type A+, while in 2017's partition 4, 2018's partitions 2 and 3, and 2019's partitions 1 and 3, there was a higher number of O+;

- **Gender**, the partitions were considerably well divided into Feminine or Masculine, since in 2017's partitions 3 and 4, 2018's partitions 3 and 7, and 2019's partitions 3 and 5, there was a higher number of females, while in 2017's partitions 1 and 2, 2018's partitions 1 and 2, and 2019's partitions 1 and 4, there was a higher number of males;
- **District**, the partitions were decently divided into districts from the northern part of the country (Porto, Braga, and Aveiro) or districts from the central part of the country (Lisboa and Leiria), since in 2017's partitions 1 and 5, 2018's partitions 5 and 7, and 2019's partition 5, there was a higher number of donations in the north while in 2017's partition 2, 2018's partitions 2 and 4, and 2019's partitions 1 and 4, there was a higher number of donations in the center;

Table 5.2 displays this information in a more compact way.

Table 5.2: Partitions of each community that are represented by which feature.

Year	Men	Women	North	Center	A+	O+
2017	1, 2	3, 4	1, 5	2	1, 3	4
2018	1, 2	3, 7	5, 7	2, 4	1, 7	2, 3
2019	1, 4	3, 5	5	1, 4	4, 5	1, 3

From this information, an important thing to note is the fact that the communities that were found follow a pattern throughout the years, since most communities appear connected by two features. The patterns found were: Gender=Male, Blood Type=A+ (2017-2019); Gender=Male, Blood Type=O+ (2017-2019); Gender=Females, Blood Type=A+ (2017-2019); Gender=Females, Blood Type=O+ (2017-2019); Region=North, Blood Type=A+ (2017-2019); Gender=Males, Region=Center (2017-2019); Gender=Females, Region=North (2018-2019); Blood Type=O+, Region=Center (2018-2019). We also found a pattern connected by three features: Gender=Male, Blood Type=O+, Region=Center (2018-2019). So, using these features, it was possible to extract some characteristics that can distinguish each of the communities that were found.

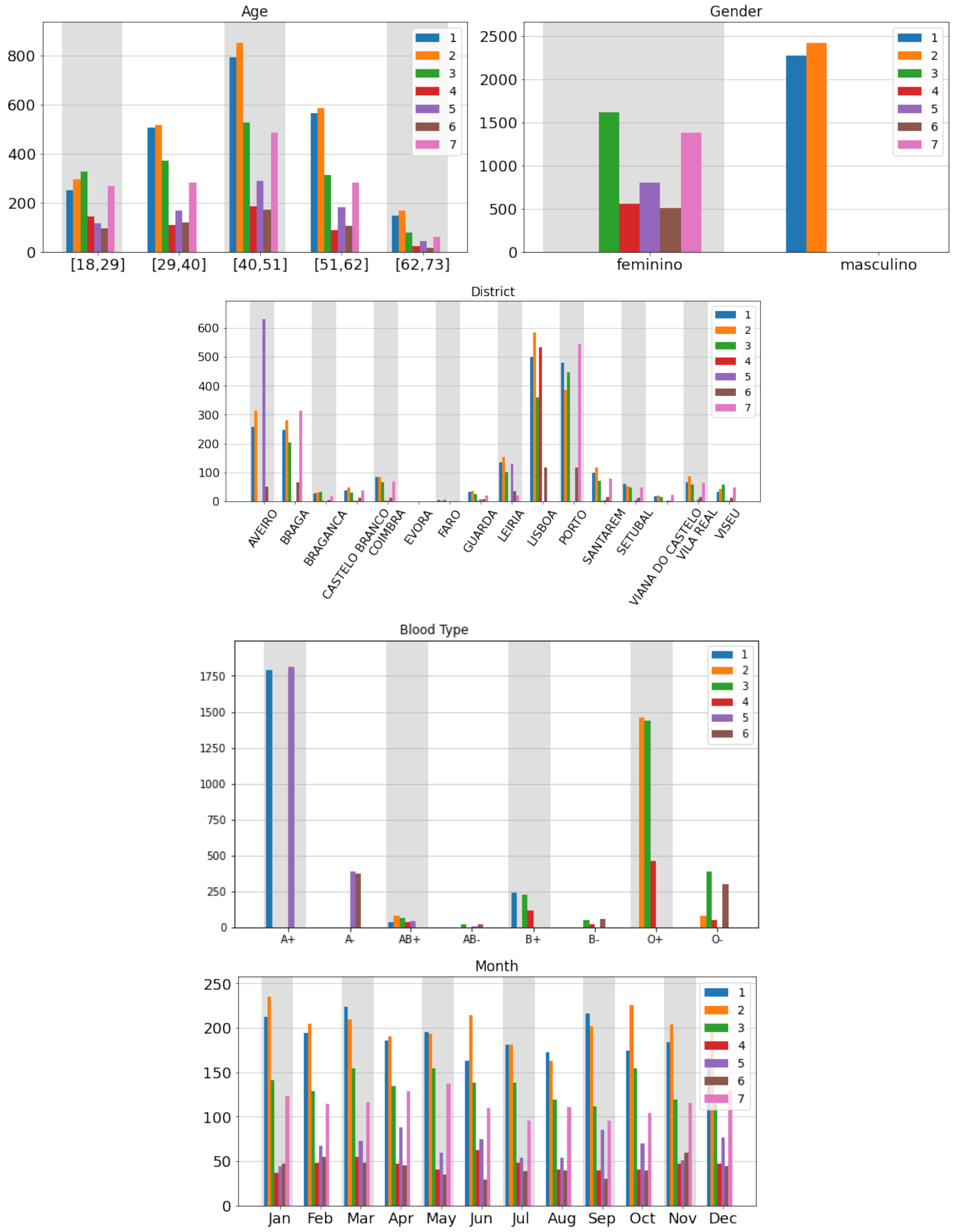


Figure 5.3: Plotting of the node distribution in each feature for the different communities discovered in the blood donation network of 2018.

5.3 Transactional Mining & Association Rule Learning

In this pattern mining approach, we tested the `Apriori` algorithm [32], `FP-Growth` algorithm [34,35] and `FP-Max` algorithm [36] for transactional mining, which were explained in Section 2.2 from Chapter 2. The respective workflow can be seen in Figure 5.4.

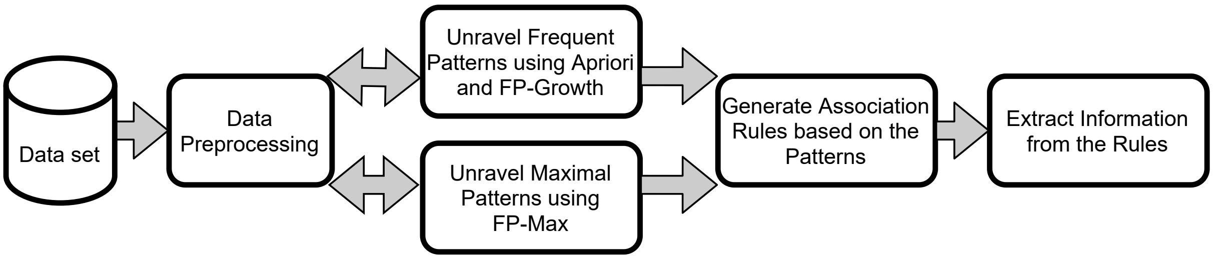

Data set
Data Preprocessing
Unravel Frequent Patterns using Apriori and FP-Growth
Unravel Maximal Patterns using FP-Max
Generate Association Rules based on the Patterns
Extract Information from the Rules

Figure 5.4: Workflow for the transactional mining and association rule learning approach.

5.3.1 Data Pre-processing

Using the three data sets (2017, 2018, and 2019) created in the previous section (Section 5.1), we started by reducing the number of columns, since the algorithms used do not scale well horizontally, as they are focused on performing well over transaction data sets with a high number of rows (transactions) and a small number of features (items).

To perform this reduction, we started by removing the features that would not be relevant to the experiment. This way, the first features to be removed were the ones with a high percentage of missing values, which we considered as having more than a 20% threshold of missing values. This choice was made because even though the algorithms we planned on using for this approach were able to deal with missing values, the information that we could obtain from features with so few values was negligible. As well as the fact that the information itself that these features provided would not be relevant for the type of exploration we intended on doing, being only a hindrance to the algorithms, since it would increase the resources needed, because the data set would be bigger horizontally, which decreases the performance of the algorithms, which would not provide any real benefit for the experiment.

Additionally, with the same goal in mind, we tested the algorithms with different sets of features in smaller data sets. After some discussion, we arrived at the conclusion that we should remove some features that were not relevant to our experiment, since they did not present any sort of new information that could be used for our goal. This was because it was already included in other features, such as the several ones that refer to where the donation took place, or because the values themselves were not important to our goal, either by not having actually relevant information (e.g. the feature 'colheita_dias_dador_suspenso', which gave us the number of days left until a donor was allowed to donate blood again) or by having unbalanced distributions of the data in them, favoring only one of the values (e.g. the feature ’dador_nacionalidade’, where the value ’PORTUGAL’ appears 98% of the time).

To reduce the number of features even more, we decided to collapse some of the remaining ones into a single one. These features, which could be new or not, were:

- ’brigada_info’ and was composed of the several features that together characterized and identified a blood collection brigade. Seeing that a brigade keeps its characteristics over the time, by grouping them into one, we would have a feature that has the information about the brigade that does not change over time. As so, the features ’associacao_id_por_centro_sangue’ and ’brigada_id_por_centro_sangue’ were joined to form a brigade’s identifier, considering each association only has one brigade with the same identifier, and the features ’brigada_tipo_colheita’, ’brigada_tipo’, and ’brigada_freguesia’ were attached to that identifier in order to better characterize the brigade (e.g. 0000AAA_BRIGADA_EMPRESAS_LISBOA);

- ’dador_tipo_sangue’, which is composed of the features ’dador_tipo_sangue_abo’ and ’dador_tipo_sangue_rh’ and is a string of the complete blood type of a donor (e.g. A- or AB+).

- ’colheita_semana’, since each data set already represents a single year, the day of the donation was repeated over each month, so it could not properly locate a donation in a single year, whereas the month presented too coarse of a granularity. With this in mind, we decided to use this feature, which represents the week of the year when the donation was made.

Lastly, we were left with four features with integer values. These features, when represented in the form of patterns and association rules, cannot be distinguished between them, so we do not know what the value we are seeing represents. For this reason, we decided to tag each of these features as follows:

- ’dador_total_dadivas’, was tagged with ”DTD_” before the value;

- ’dador_idade’, was tagged with ”DI_” before the value;

- ’colheita_semana’, was tagged with ”CS_” before the value;

- ’colheita_nr_dadores_previstos’, was kept as an integer.

To use the algorithms, we utilized the Mlxtend library[2], which is a Python library with multiple data science tools. The implementations of the algorithms in the Mlxtend use binary data sets. So, in order to use our data set, we had to first transform it into a binary one. In it, each line i is a donation, each column j is a feature’s value, and in position ij the value 1 represents that the feature value j appears at donation i.

[2] http://rasbt.github.io/mlxtend/

5.3.2 Applying the Algorithms

With the binary data set, we were then able to input it into the Mlxtend library and perform transactional mining using the `Apriori` and `FP-Growth` algorithms, and obtain the frequent patterns. After considering different minimum support thresholds, we were able to determine that for our data, the `FP-Growth` algorithm had better performance than the `Apriori` algorithm. We needed a considerably small support threshold to obtain a meaningful number of frequent patterns, this way containing more than just the obvious ones. The amounts obtained for different values of minimum support thresholds can be seen in Table 5.3.

Table 5.3: Number of Frequent Patterns obtained for the years 2017, 2018, and 2019 with minimum support 0.05, 0.02 and 0.01 using the `Apriori` and `FP-Growth` algorithms.

Year	Minimum Support	#Frequent Patterns
2017	0.05	88
	0.02	413
	0.01	$1\,250$
2018	0.05	89
	0.02	423
	0.01	$1\,250$
2019	0.05	87
	0.02	425
	0.01	$1\,254$

After obtaining the frequent patterns using a minimum support of 0.01, we could perform the next step in association rule mining, which was to generate the association rules based on them. We used the `generate_rules` method from the Mlxtend library to generate the association rules. This function allowed us to choose a metric of interest (between confidence and lift) and then determine a threshold according to it.

Afterwards, we used the `FP-Max` algorithm to obtain the maximal frequent patterns while keeping the minimum support threshold. This way, the total number of patterns obtained was smaller, due to the removal of the sub-patterns of larger frequent patterns. One downside of this algorithm, which had to be kept in mind, was the fact that it focuses only on the maximal support and limits the capability of generating rules since the antecedent and the consequent supports were not computed beforehand. As such, to be able to create rules based on the maximal frequent patterns, we needed to set the field ”support_only” of the `generate_rules` method as ”True”. This way, we were able to obtain a data set with only the support for each rule. So, to complete the rest of the data set, we merged the maximal frequent patterns data set with the frequent patterns data set, according to the patterns that were present in the former, to obtain the values of the other metrics that had already been calculated using `FP-Growth`.

To better visualize and interpret the association rules we obtained, we used the ArulesViz library in R. The two types of plot we decided to use from ArulesViz were the scatterplot and the grouped matrix plot. To utilize it, we had to transform the data set into the proper input format for the library to be able to interpret it. This had to be done seeing that the Arules library[3], in which ArulesViz is based on, only has the implementation of the Apriori algorithm. However, the output presented using the Apriori algorithm was different from the one we had obtained with the Mlxtend library for the same algorithm and the FP-Growth one. The difference was that the Arules implementation of Apriori presented the rules restricted to only a single item in the consequent, whereas the Mlxtend implementation allowed the consequent to have one or more items.

5.3.3 Results

We begin by presenting the patterns obtained with the FP-Growth and FP-Max algorithms for the data sets of the years 2017, 2018, and 2019, using a minimum support threshold of 0.01. In Table 5.4, we present the top 10 patterns obtained after sorting them by support. However, presenting them only by support mostly shows itemsets with less information, especially using the FP-Growth algorithm, where the top pattern only has 1 item. So, to obtain more information from the patterns, we decided to sort them by length first and then by support, which can be seen in Table 5.5. This table shows that over the course of the three years the top 10 patterns did not change substantially, mainly when it comes to the districts, brigades, and blood types that appear, which was to be expected considering they correspond to the most common values in the data sets. An interesting fact is that although the top 2 patterns were the same in all three years, the third one changed in 2019. Such information is relevant, especially when put together with the fact that in 2017 there were only two patterns where the donors were single ('SOLTEIRO'), while in 2018 and 2019 there were three, with increasing support over time as well. This presents an increase in the number of donors that are single and donate blood to brigades from the district of Lisboa.

Having obtained the frequent and maximum patterns, we generated the association rules based on them. After comparing the results with different minimum support and confidence thresholds, we chose to keep the same minimum support as before (i.e. 0.01) and a level of confidence above 50% (minimum confidence threshold of 0.5) to perform the rest of the experiment. Table 5.6 displays the top 5 rules using the FP-Max algorithm with a threshold as so. It is sorted by lift to highlight the more significant associations, since the higher the lift value, the more dependent on each other the antecedent and consequent are. From it, we can see that the district of Coimbra is the one being represented the most. It highlights the fact that the brigade "00144PF_POSTO FIXO_OUTRA_COIMBRA" is highly likely to receive blood donations from males that reside in Coimbra, when the number of expected donations is 12, and the other possible combinations with different levels of certainty.

The tables like the one in Table 5.6 present two different support metrics besides the relative support

[3] https://github.com/mhahsler/arules

Table 5.4: Top 10 patterns obtained for the data sets of 2017, 2018, and 2019 using the `FP-Growth` and `FP-Max` algorithms. Sorted by support.

FP-Growth		FP-Max	
Support	Itemsets	Support	Itemsets
2017			
0.5018	’feminino’	0.0267	’feminino’, ’CASADO’, ’40.0’
0.3775	’A+’	0.0253	’masculino’, ’50.0’, ’CASADO’, ’A+’
0.3361	’SOLTEIRO’	0.0249	’masculino’, ’80.0’, ’CASADO’
0.2224	’LISBOA’	0.0238	’80.0’, ’feminino’, ’CASADO’
0.0434	’DTD_5.0’	0.0237	’masculino’, ’O+’, ’50.0’, ’CASADO’
0.0223	’10.0’	0.0237	’PORTO’, ’feminino’, ’CASADO’, ’A+’
0.0176	’CS_29’	0.0236	’SOLTEIRO’, ’60.0’, ’feminino’
0.0170	’DI_32.0’	0.0232	’O-’, ’CASADO’, ’masculino’
0.0112	’HOSPHSM_POSTO AVANCADO_OUTRA_’	0.0231	’DTD_1.0’, ’SOLTEIRO’, ’feminino’
0.5447	’CASADO’	0.0225	’DTD_7.0’, ’feminino’
2018			
0.5421	’CASADO’	0.0252	’feminino’, ’CASADO’, ’40.0’
0.4945	’masculino’	0.0248	’masculino’, ’50.0’, ’CASADO’, ’O+’
0.3764	’A+’	0.0241	’PORTO’, ’feminino’, ’CASADO’, ’A+’
0.2201	’LISBOA’	0.0238	’DTD_1.0’, ’SOLTEIRO’, ’feminino’
0.0258	’DI_52.0’	0.0229	’masculino’, ’CASADO’, ’O-’
0.0206	’10.0’	0.0225	’masculino’, ’DIVORCIADO’
0.0165	’CS_31’	0.0223	’A-’, ’CASADO’, ’masculino’
0.0114	’HOSPHSM_POSTO AVANCADO_OUTRA_’	0.0221	’O-’, ’feminino’, ’CASADO’
0.0599	’B+’	0.0220	’BRAGA’, ’SOLTEIRO’, ’feminino’
0.0203	’DTD_20.0’	0.0216	’80.0’, ’CASADO’, ’feminino’
2019			
0.5348	’CASADO’	0.0256	’masculino’, ’50.0’, ’CASADO’, ’A+’
0.5080	’feminino’	0.0247	’DTD_2.0’, ’SOLTEIRO’, ’feminino’
0.4920	’masculino’	0.0243	’masculino’, ’CASADO’, ’40.0’
0.3778	’A+’	0.0241	’PORTO’, ’feminino’, ’CASADO’, ’A+’
0.3550	’O+’	0.0241	’masculino’, ’50.0’, ’CASADO’, ’O+’
0.3471	’SOLTEIRO’	0.0240	’feminino’, ’CASADO’, ’40.0’
0.2845	’masculino’, ’CASADO’	0.0227	’masculino’, ’CASADO’, ’PORTO’, ’A+’
0.2503	’feminino’, ’CASADO’	0.0226	’masculino’, ’CASADO’, ’O-’
0.2307	’50.0’	0.0224	’BRAGA’, ’SOLTEIRO’, ’feminino’
0.2217	’LISBOA’	0.0223	’O-’, ’feminino’, ’CASADO’

that was explained in Chapter 2, these being the ’antecedent support’ and ’consequent support’. The former corresponds to the support of the antecedent’s itemset A, while the latter corresponds to the consequent’s itemset C . The ’support’ (or relative support) is then the computation of the support of the combined itemset $A \cup C$ (calculated by the minimum between the respective supports in Mlxtend).

Table 5.5: Top 10 patterns obtained for the data sets of 2017, 2018, and 2019 using the `FP-Growth` and `FP-Max` algorithms with a minimum support threshold of 0.01. Sorted by length and then support.

Support	Itemsets
2017	
0.0198	'CASADO', 'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO'
0.0193	'CASADO', 'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0148	'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA', 'A+'
0.0145	'masculino', 'SOLTEIRO', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0143	'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA', 'O+'
0.0132	'feminino', 'SOLTEIRO', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0125	'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'A+'
0.0112	'CASADO', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'A+'
0.0111	'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'O+'
0.0103	'CASADO', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA', 'A+'
2018	
0.0205	'CASADO', 'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO'
0.0178	'CASADO', 'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0139	'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA', 'A+'
0.0137	'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA', 'O+'
0.0135	'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'A+'
0.0134	'masculino', 'SOLTEIRO', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0120	'feminino', 'SOLTEIRO', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0118	'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'O+'
0.0116	'CASADO', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'A+'
0.0111	'masculino', 'SOLTEIRO', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO'
2019	
0.0204	'CASADO', 'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO'
0.0183	'CASADO', 'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0148	'masculino', 'SOLTEIRO', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0145	'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA', 'A+'
0.0142	'masculino', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA', 'O+'
0.0133	'feminino', 'SOLTEIRO', 'A322PF_POSTO FIXO_OUTRA_LISBOA', '50.0', 'LISBOA'
0.0130	'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'A+'
0.0117	'masculino', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'O+'
0.0113	'CASADO', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO', 'A+'
0.0110	'masculino', 'SOLTEIRO', '0009PP_POSTO FIXO_OUTRA_PORTO', '50.0', 'PORTO'

Table 5.6: Top 5 association rules obtained for the data sets of 2017, 2018, and 2019 using the `FP-Max` algorithm with minimum support and confidence thresholds of 0.01 and 0.05, respectively. Sorted by lift.

Antecedents	Consequents	Support	Antecedent support	Consequent support	Confidence	Lift	Leverage	Conviction
2017								
COIMBRA, 12.0, masculino	00144PF_POSTO FIXO_OUTRA_COIMBRA	0.0104	0.0104	0.0173	1	57.6678	0.0103	inf
12.0, masculino	COIMBRA, 00144PF_POSTO FIXO_OUTRA_COIMBRA	0.0104	0.0104	0.0173	1	57.6678	0.0103	inf
00144PF_POSTO FIXO_OUTRA_COIMBRA	COIMBRA, 12.0, masculino	0.0104	0.0173	0.0104	0.6023	57.6678	0.0103	2.4880
COIMBRA, 00144PF_POSTO FIXO_OUTRA_COIMBRA	12.0, masculino	0.0104	0.0173	0.0104	0.6023	57.6678	0.0103	2.4880
COIMBRA, 00144PF_POSTO FIXO_OUTRA_COIMBRA, masculino	12	0.0104	0.0105	0.0173	0.9956	57.6373	0.0103	224.3296
2018								
00144PF_POSTO FIXO_OUTRA_COIMBRA	COIMBRA, 12.0	0.0162	0.0162	0.0162	1	61.8338	0.0159	inf
COIMBRA, 00144PF_POSTO FIXO_OUTRA_COIMBRA	12	0.0162	0.0162	0.0162	1	61.8338	0.0159	inf
COIMBRA, 12.0	00144PF_POSTO FIXO_OUTRA_COIMBRA	0.0162	0.0162	0.0162	1	61.8338	0.0159	inf
12	COIMBRA, 00144PF_POSTO FIXO_OUTRA_COIMBRA	0.0162	0.0162	0.0162	1	61.8338	0.0159	inf
0096AXO_BRIGADA_OUTRA_VILA REAL	80.0, VILA REAL	0.0113	0.0132	0.0159	0.8576	54.0293	0.0111	6.9094
2019								
12.0, masculino	COIMBRA, 00144PF_POSTO FIXO_OUTRA_COIMBRA	0.0101	0.0101	0.0165	1	60.7075	0.0099	inf
00144PF_POSTO FIXO_OUTRA_COIMBRA, masculino	COIMBRA, 12.0	0.0101	0.0101	0.0165	1	60.7075	0.0099	inf
COIMBRA, 00144PF_POSTO FIXO_OUTRA_COIMBRA, masculino	12	0.0101	0.0101	0.0165	1	60.7075	0.0099	inf
COIMBRA, 12.0, masculino	00144PF_POSTO FIXO_OUTRA_COIMBRA	0.0101	0.0101	0.0165	1	60.7075	0.0099	inf
00144PF_POSTO FIXO_OUTRA_COIMBRA	COIMBRA, 12.0, masculino	0.0101	0.0165	0.0101	0.6119	60.7075	0.0099	2.5507

We then decided that besides studying the results only for the whole country, it would also be interesting to study the differences between the districts to analyze the data with more detail. So, we divided each of the data sets according to the home districts of the brigades that performed the blood collections. For this study, we decided to keep the same minimum support and confidence thresholds for all the new data sets.

We will now present some of the most interesting results we obtained for the top 5 districts according to the brigade's district of origin, which could be seen in the previous chapter in Figure 4.6. Due to the size of the tables and in order for them to be readable, we only present the columns for the antecedents, consequents, support, confidence and lift, since these were the three metrics that we considered to show more relevant information. Seeing that the top results are mostly filled with similar information we opted by presenting the results for the `FP-Max` algorithm, because if we were looking at the top rules generated with the `FP-Growth` algorithm these would contain even more rules with similar information, as it would include the association rules that were generated with sub-sequences. As you will see, we chose to present a table with the top 10 associations rules for each district sorted by their lift, followed by a grouped matrix plot. However, the year between these two may be different, considering we decided to present the table and matrix that respectively presented the highest amount of relevant information regarding each brigade's district of origin for the 2017-2019 time frame. The choice of presenting a grouped matrix plot was done since it allows us to study some of the rules and possible cores (items that are often repeated). The antecedents (LHS), which define the columns, are grouped using clustering. The groups chosen are represented by the most interesting item in them, this is the item with the highest ratio of support in the group to support in all rules. And the circles are used to represent the consequent-antecedent connection.

5.3.3.A Lisboa

As an example for this district, we can see Table 5.7 showing the top 10 association rules for Lisboa in 2017, and the grouped matrix in Figure 5.5 for the district in 2019, both obtained using the `FP-Max` algorithm.

The Lisboa district's association rules with the highest lift over the three years highlighted donations that were made by donors that were university students. We defined a donor as so if they presented the values ”ESTUDANTE” or ”ESTUDANTE DO ENSINO SUPERIOR”, since with most donations starting at the age of 20 years old, it would be unlikely that the ”ESTUDANTE” would not already be in university as well. Following this, we can also see that Lisboa has a tendency for donations done in universities (”FACULDADES”), which was to be expected looking at the donors' jobs highlighted for the district. We can also see a high tendency for donations in other establishments (”OUTRA”). We believe these refer to mostly hospitals, since they are often related to brigade identifiers like ”HOSPHSM”, which in this case

Table 5.7: Top 10 association rules obtained for the data set of Lisboa in 2017 using the `FP-Max` algorithm with minimum support and confidence thresholds of 0.01 and 0.05, respectively. Sorted by lift.

Antecedents	Consequents	Support	Confidence	Lift
HOSPHES_POSTO AVANCADO_OUTRA_LISBOA	5	0.0123	0.9833	77.6769
5	HOSPHES_POSTO AVANCADO_OUTRA_LISBOA	0.0123	0.9686	77.6769
SOLTEIRO, 120.0	A32PST_BRIGADA_ FACULDADES_LISBOA	0.0114	0.9479	74.9894
A32PST_BRIGADA_ FACULDADES_LISBOA	SOLTEIRO, 120.0	0.0114	0.8982	74.9894
A32PST_BRIGADA_FACULDADES _LISBOA, SOLTEIRO	120	0.0114	1.0000	74.7496
120	A32PST_BRIGADA_FACULDADES _LISBOA, SOLTEIRO	0.0114	0.8487	74.7496
10.0, ESTUDANTE	HOSPHSM_POSTO AVANCADO _OUTRA_, SOLTEIRO	0.0110	0.7975	25.5754
HOSPHSM_POSTO AVANCADO _OUTRA_, ESTUDANTE	SOLTEIRO, 10.0	0.0110	0.9749	21.0812
ESTUDANTE, SOLTEIRO, 10.0	HOSPHSM_POSTO AVANCADO_OUTRA_	0.0110	0.8728	17.1590
feminino, HOSPHSM_POSTO AVANCADO_OUTRA_	SOLTEIRO, 10.0	0.0192	0.6749	14.5934

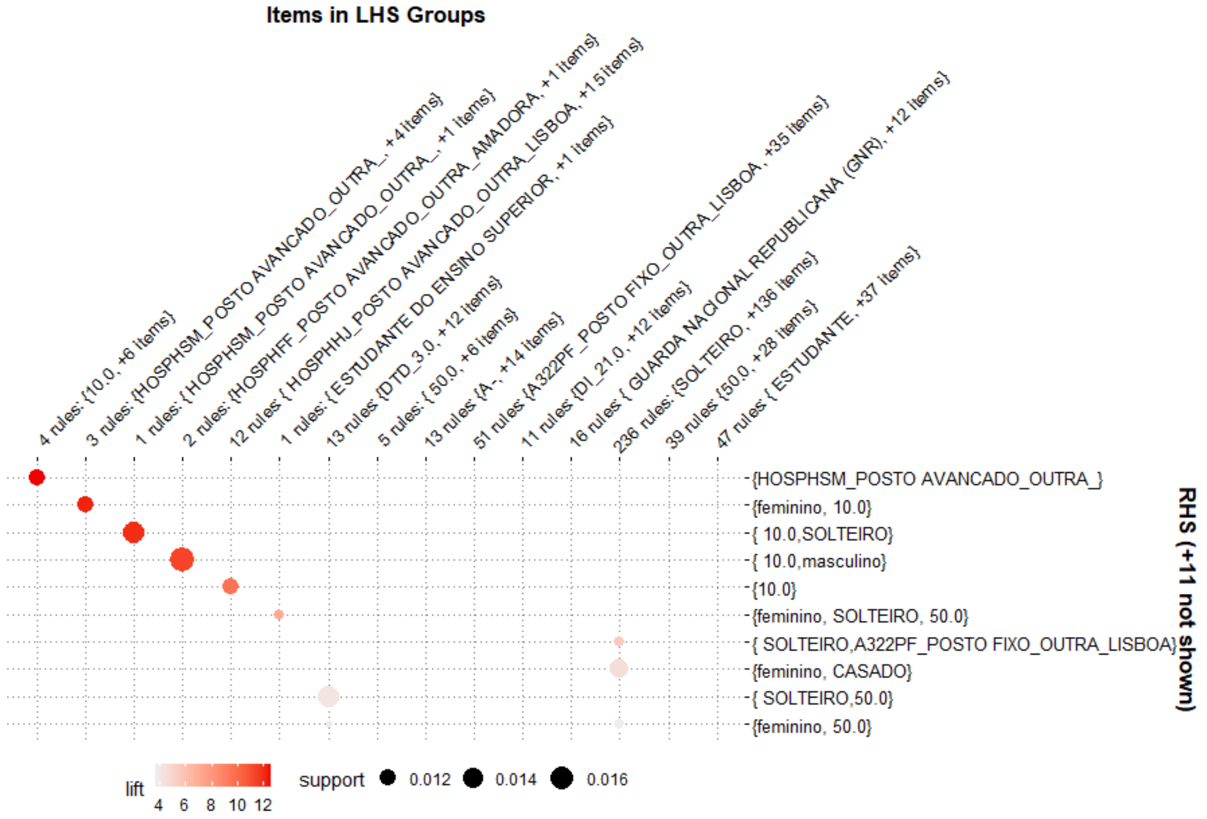


Figure 5.5: Grouped matrix of the association rules obtained with FP-Max for Lisboa in 2019.

refers to Hospital Santa Maria, and the fact that there is a large number of hospitals in Lisboa validates this idea. The rules mostly highlight donors that are single, which once again corroborates the idea that the main donor population in Lisboa is composed of students or people that finished university recently and are still establishing their lives. Regarding the gender of the donor, Lisboa is overall balanced in terms of the rules that are highlighted, but we can see an increase in the lift of rules containing females over the three-year time frame. Lisboa's rules highlight the blood type of the donor, focusing on blood types A+ and O+. This information was to be expected, since these two are the most common blood types both in the data sets and in the world. The weeks in which the donations are made are not highlighted in the top 20 rules. This may be explained by the fact that Lisboa has a high number of donations done in hospitals and most of them are open over the entirety of each year to accept blood donations, meaning that there is an equal distribution of donations over the year. Similarly, the number of total donations made by a donor is not highlighted, meaning that the values do not present a tendency. Another aspect that might justify the fact that Lisboa has a high number of donations done in hospitals, is that the number of donations expected by brigade is mostly 10 and sometimes 120. The latter was expected, since Lisboa is the capital of Portugal, so it would be normal to expect a high number of donors to show up to a blood collection done by a brigade, but 10 does not follow this thought process. However, if we look at the fact that hospitals are open throughout the year, this value can be explained as the number of expected donors for a single day in one of them.

5.3.3.B Porto

Table 5.8, where we show the top 10 association rules according to their lift, and the grouped matrix from Figure 5.6, serves as an example of the results for the Porto district obtained using the `FP-Max` algorithm, by presenting the results for 2017.

The association rules from Porto regarding the job of the donor only highlighted that the donations were made by donors that were factory employees (”EMPREGADO FABRIL”) over the three years. Regarding the place where the donations took place, the district's top rules are divided between three main categories: schools, localities, and fire stations (”ESCOLAS”, ”LOCALIDADES”, and ”BOMBEIROS”). This result was not expected, since if most donors were factory employees, we would expect to see a higher number of donations in places such as their companies. The rules highlight donors that are married. This could be expected, especially taking into account the fact that the donors that were being highlighted were factory employees. As a result, they can already be associated as part of the working class and have mostly already established their lives. Porto, similarly to Lisboa, is also balanced overall regarding the gender of the donor, taking into account the rules that are highlighted. However, we can again see an increase in the lift of rules containing females over the three-year time frame. Porto's rules also highlight the blood type of the donor, focusing on blood types A+ and O+. This information was to be

Table 5.8: Top 10 association rules obtained for the data set of Porto in 2017 using the `FP-Max` algorithm with minimum support and confidence thresholds of 0.01 and 0.05, respectively. Sorted by lift.

Antecedents	Consequents	Support	Confidence	Lift
GONDOMAR, 120.0	0005ZQ_BRIGADA_ESCOLAS_GONDOMAR	0.0105	0.9980	46.5729
60.0, POVOA DO VARZIM	0081AQA_BRIGADA_LOCALIDADES_POVOA DO VARZIM	0.0116	0.9786	31.3346
feminino, 18AOA_BRIGADA_ESCOLAS_GUIMARAES	EMPREGADO FABRIL, 140.0	0.0104	0.5285	28.5918
EMPREGADO FABRIL, 140.0	feminino, 18AOA_BRIGADA_ESCOLAS_GUIMARAES	0.0104	0.5624	28.5918
0005ZQ_BRIGADA_ESCOLAS_GONDOMAR, 120.0	GONDOMAR	0.0105	0.5872	26.8193
MATOSINHOS, 80.0	0095AUL_POSTO MOVEL_ESCOLAS_MATOSINHOS	0.0122	0.9897	26.5126
EMPREGADO FABRIL, 140.0	CASADO, 18AOA_BRIGADA_ESCOLAS_GUIMARAES	0.0123	0.6644	26.4334
FELGUEIRAS, 140.0	18AOA_BRIGADA_ESCOLAS_GUIMARAES	0.0117	0.9651	25.9264
EMPREGADO FABRIL,18AOA_BRIGADA_ESCOLAS_GUIMARAES	feminino, 140.0	0.0104	0.5605	25.5757
POVOA DO VARZIM	60.0, 0081AQA_BRIGADA_LOCALIDADES_POVOA DO VARZIM	0.0116	0.6229	24.7823

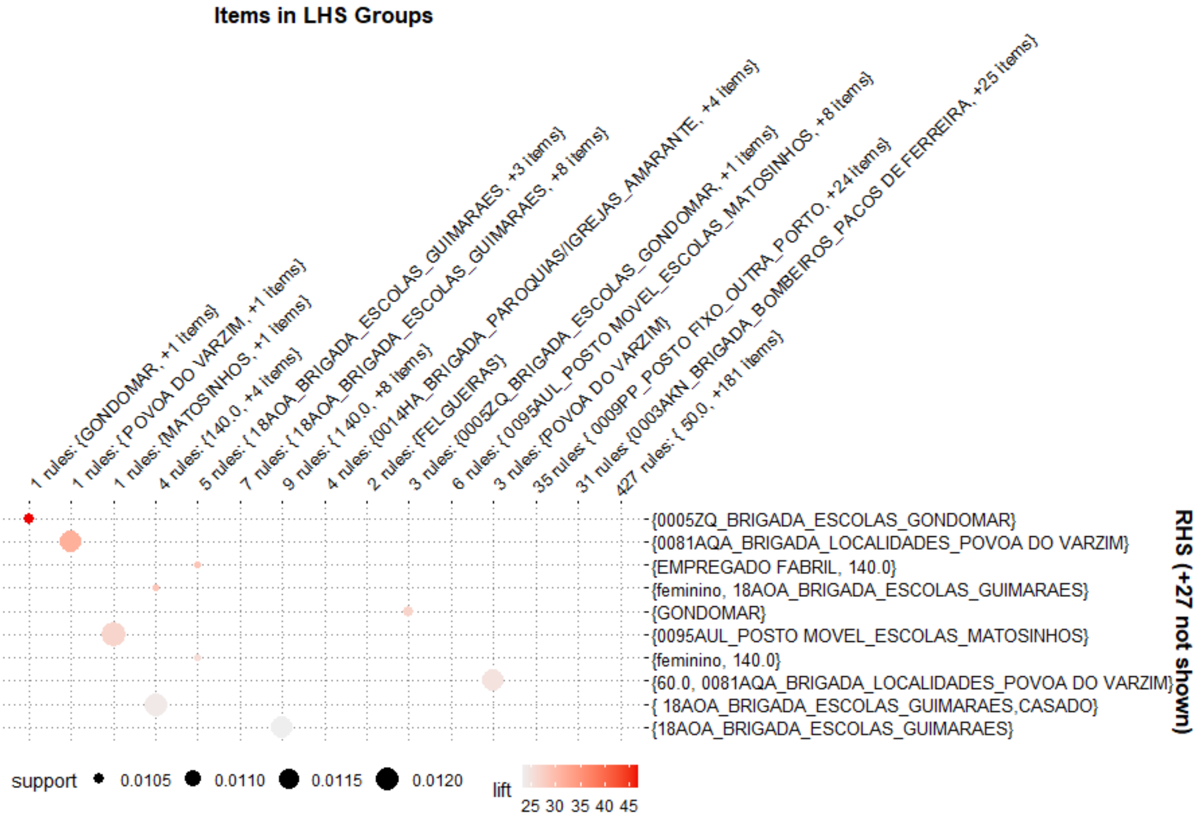


Figure 5.6: Grouped matrix of the association rules obtained with FP-Max for Porto in 2017.

expected, since these two are the most common blood types both in the data sets and in the world. The top 20 rules do not highlight the weeks in which the donations are made, similarly to Lisboa. This shows that the brigades from the Porto district work frequently throughout the year, and not only in specific time frames. Similarly, the number of total donations made by a donor is not highlighted, meaning that the values do not present a tendency. Since the brigades original from the Porto district cover a large area of the country and the district itself has an elevated number of donors, the values that are highlighted for the total number of expected donations in different brigades are not surprising, varying between 60 and 150 donations.

5.3.3.C Aveiro

For this district, we can see the examples shown in Table 5.9 where we present the top 10 association rules according to the lift for the district of Aveiro in 2019, and the grouped matrix in Figure 5.7 for the district in 2018. Both were generated with the information obtained by using the `FP-Max` algorithm.

Table 5.9: Top 10 association rules obtained for the data set of Aveiro in 2017 using the `FP-Max` algorithm with minimum support and confidence thresholds of 0.01 and 0.05, respectively. Sorted by lift.

Antecedents	Consequents	Support	Confidence	Lift
0008CU_BRIGADA_ESCOLAS_ALPENDURADA E MATOS	ALPENDURADA E MATOS	0.0127	0.6828	53.3034
ALPENDURADA E MATOS	0008CU_BRIGADA_ESCOLAS_ALPENDURADA E MATOS	0.0127	0.9944	53.3034
EMPREGADO FABRIL, 0014VYS_BRIGADA_EMPRESAS_OVAR	250.0	0.0108	0.8059	48.8743
250.0	EMPREGADO FABRIL, 0014VYS_BRIGADA_EMPRESAS_OVAR	0.0108	0.6543	48.8743
250.0	CASADO, 0014VYS_BRIGADA_EMPRESAS_OVAR	0.0113	0.6849	43.3364
CASADO, 0014VYS_BRIGADA_EMPRESAS_OVAR	250.0	0.0113	0.7146	43.3364
250.0	feminino, 0014VYS_BRIGADA_EMPRESAS_OVAR	0.0124	0.7505	41.3532
feminino, 0014VYS_BRIGADA_EMPRESAS_OVAR	250.0	0.0124	0.6819	41.3532
CASADO, 250.0	0014VYS_BRIGADA_EMPRESAS_OVAR	0.0113	1.0000	40.0491
EMPREGADO FABRIL, 250.0	0014VYS_BRIGADA_EMPRESAS_OVAR	0.0108	1.0000	40.0491

We could see that in the top 10 rules for each year, only 2019 presented any highlighted information regarding the jobs of the donors, which was that the donors were factory employees (”EMPREGADO FABRIL”), similarly to Porto. However, in the grouped matrix for 2018 we can see a reference to the same type of donors, showing that even though it is not as highlighted in terms of lift, it is still a relevant part of Aveiro’s donors. Contrary to what was seen in the Porto district, but following what we were expecting at first, in Aveiro we can see that one of the most common places where donations were done were companies/factories (”EMPRESAS”), this way being highlighted in the top association rules. Such a fact was expected, since, as we just saw, factory employees were highlighted as donors for this district, so it

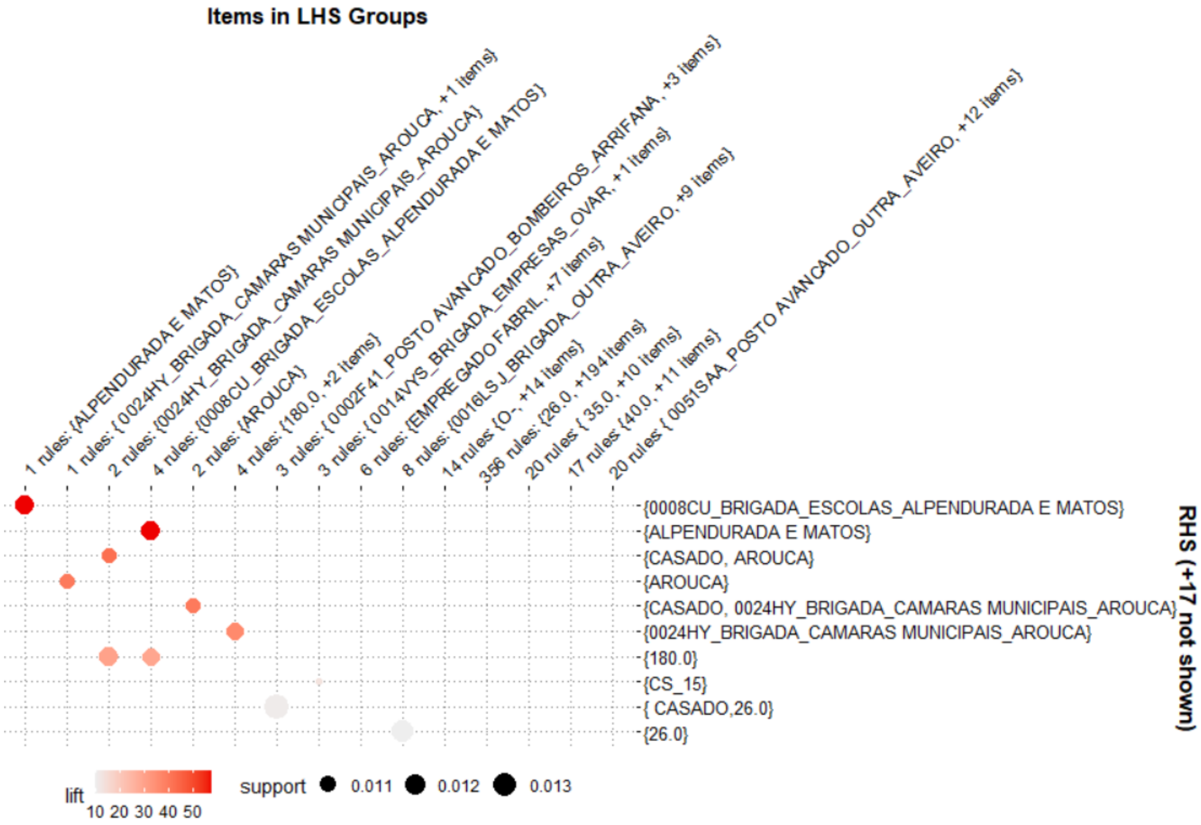


Figure 5.7: Grouped matrix of the association rules obtained with FP-Max for Aveiro in 2018.

makes sense that there are brigades that go and collect blood at the specific companies/factories where they work for a higher donation rate. Other places that were highlighted for this district were schools, city halls, and fire stations (”ESCOLAS”, ”CAMARAS MUNICIPAIS”, and ”BOMBEIROS”). Following the same thought process that was explained for the Porto district, we were expecting that the donors in Aveiro were mostly married. Such a hypothesis was verified by the fact that the top rules only presented information regarding married donors. Aveiro presented a similar distribution of male and female donors in 2018, but in 2017 and 2019, the top 20 rules according to the lift only highlighted female donors. For blood type, we do not see any rules with regard to it except in 2018, where we see some rules containing blood types A+ and O+. This shows that there is a higher number of other blood types in the district in 2017 and 2019, so the rules are not able to select one single type as a tendency. Regarding the weeks of the year that were highlighted for this district, we can see a reference to the weeks 13 in 2019 and 15 in 2018, which are located at the end of March and beginning of April, respectively. This means that there was a brigade that might have been repeated around the same time of the year in the two years that had a high number of blood donations. The number of total donations done by a donor is not highlighted, meaning that the values do not present a tendency. The number of donations expected per brigade varies between 26 and 250. The brigades expecting 26 donations refer to ones mostly done in fire stations, and the ones with higher donation expectancies were done in companies/factories. This was expected, since, as we have seen, the district is characterized by donors that are factory employees, so it is fair to expect more donations to brigades that collect blood where they work.

5.3.3.D Braga

As an example for this district, we can see Table 5.10 showing the results for Braga in 2019, more specifically the top 10 association rules according to the lift, and the grouped matrix in Figure 5.8 for the district in 2018. Both were created using the information obtained after applying the FP-Max algorithm to the data sets of the Braga district.

Table 5.10: Top 10 association rules obtained for the data set of Braga in 2019 using the FP-Max algorithm with minimum support and confidence thresholds of 0.01 and 0.05, respectively. Sorted by lift.

Antecedents	Consequents	Support	Confidence	Lift
SOLTEIRO, 200.0	0097SC_BRIGADA_BOMBEIROS_BRAGA	0.0119	0.6856	54.5312
0097SC_BRIGADA_BOMBEIROS_BRAGA	SOLTEIRO, 200.0	0.0119	0.9458	54.5312
CABECEIRAS DE BASTO, feminino	0097KU_BRIGADA_ESCOLAS_CABECEIRAS DE BASTO	0.0106	0.8515	50.5302
0097KU_BRIGADA_ESCOLAS_CABECEIRAS DE BASTO	CABECEIRAS DE BASTO, feminino	0.0106	0.6315	50.5302
CABECEIRAS DE BASTO	feminino, 0097KU_BRIGADA_ESCOLAS_CABECEIRAS DE BASTO	0.0106	0.6315	49.7761
feminino, 0097KU_BRIGADA_ESCOLAS_CABECEIRAS DE BASTO	CABECEIRAS DE BASTO	0.0106	0.8388	49.7761
18DL_BRIGADA_ESCOLAS_GUIMARAES	MOREIRA DE CONEGOS	0.0110	0.5026	36.5624
MOREIRA DE CONEGOS	18DL_BRIGADA_ESCOLAS_GUIMARAES	0.0110	0.7989	36.5624
SOLTEIRO, 0097SC_BRIGADA_BOMBEIROS_BRAGA	200	0.0119	1.0000	28.8601
0098MO_BRIGADA_ESCUTEIROS_LISBOA	CS_27	0.0124	0.6350	28.2751

Starting by studying the jobs of the donors, the top rules for Braga do not present any highlights regarding this topic. This means that the donations done in this district are made by different types of donors according to their job. Most of the donations, according to the highlighting done by the association rules sorted by their lift, are done in schools and fire stations (”ESCOLA” and ”BOMBEIROS”). It also shows a strong correlation between schools and the donor's location of residence, named ”CABECEIRAS DE BASTO”. This means that donors that reside in that area, are likely to donate blood to a brigade that collects blood at schools. The Braga district is mostly defined by female donors over the three different years, as we can see from the different top rules obtained. Regarding blood type, Braga's association rules do not present any highlight towards a specific type, not even the most common ones. This means that the donations done throughout the district do not have a tendency towards any sort of blood type when creating association rules, since there is no majority. Similarly, the number of total donations made by a donor is not highlighted, meaning that the values do not present a tendency. Week 27 was highlighted in the top association rules of 2019, but overall, there are not any specific times of the year when blood donations occur more often in Braga.

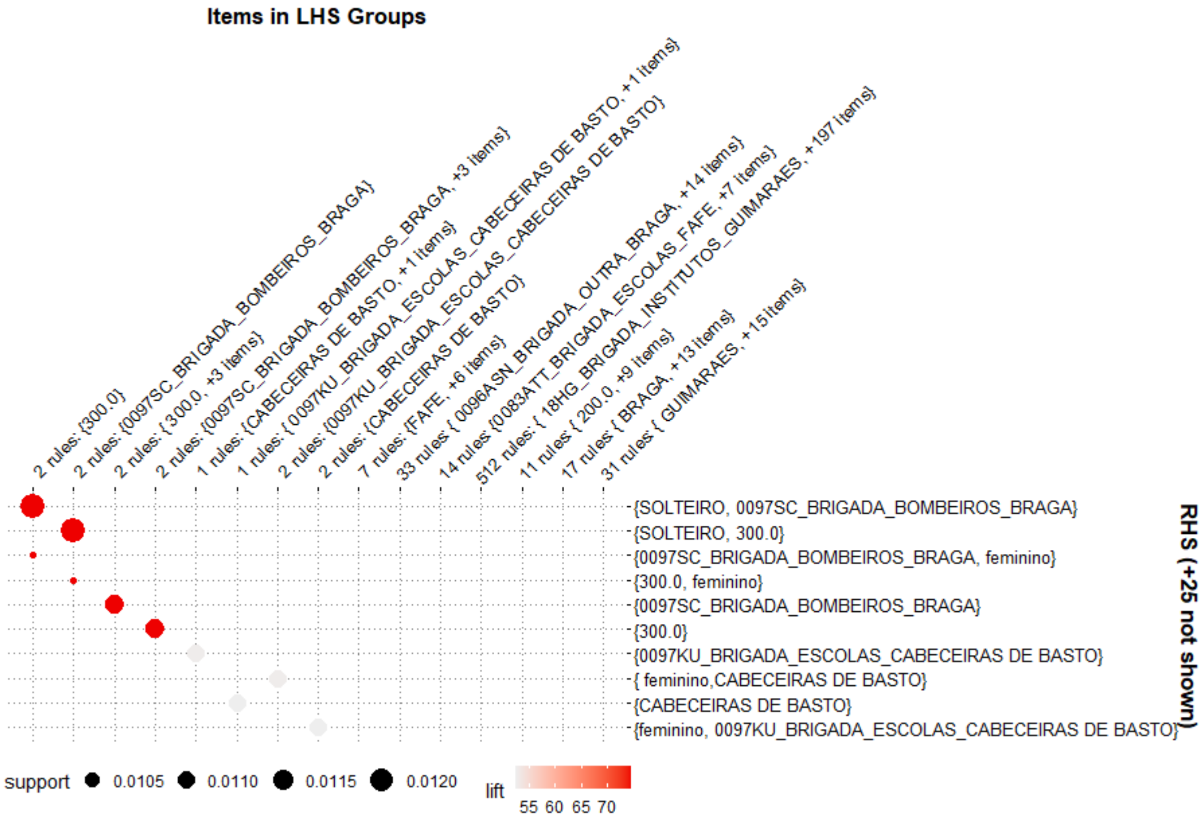


Figure 5.8: Grouped matrix of the association rules obtained with FP-Max for Braga in 2018.

5.3.3.E Leiria

Table 5.11 shows the results for Leiria in 2017 sorted by lift, and the grouped matrix in Figure 5.9 for the district in 2019, both obtained using the FP-Max algorithm.

The top rules generated for Leiria do not present any tendency towards the job of the donors that performed the donations. This means that there is not a majority of donors according to their job in this district. Regarding the place where the donations took place, we see a highlight of the ”OUTRA” value. If we keep the same idea that we had for Lisboa, this represents donations that were done in hospitals. The fact that the rules obtained do not present any association with the marital status of the donors from Leiria, shows that they have a similar distribution between being single and married. The same cannot be said about the gender, since the rules that present any information regarding that feature of the donor, always give us the information that the donor was female. Similarly to the Brage district, there is no specific information regarding the blood type of the donors that can be obtained from the top rules. Meaning that, since there is no majority, the donations done throughout the district do not have a tendency towards any sort of blood type when creating association rules. The rules obtained show, however, a tendency towards certain weeks of the year, such as 8, 14, 16, 17, 25,

Table 5.11: Top 10 association rules obtained for the data set of Leiria in 2017 using the `FP-Max` algorithm with minimum support and confidence thresholds of 0.01 and 0.05, respectively. Sorted by lift.

Antecedents	Consequents	Support	Confidence	Lift
A14P20_BRIGADA_OUTRA_CALDAS DA RAINHA, CS_45	230	0.0129	1.0000	77.7459
230	A14P20_BRIGADA_OUTRA_CALDAS DA RAINHA, CS_45	0.0129	1.0000	77.7459
90.0, BAJOUCA	0014P99_BRIGADA_OUTRA_BAJOUCA	0.0103	0.9864	75.8500
0014P99_BRIGADA_OUTRA_BAJOUCA	90.0, BAJOUCA	0.0103	0.7923	75.8500
110	001409P_BRIGADA_OUTRA_ANSIAO	0.0146	1.0000	68.3107
001409P_BRIGADA_OUTRA_ANSIAO	110	0.0146	1.0000	68.3107
400	CS_28, A14P20_BRIGADA_OUTRA_CALDAS DA RAINHA	0.0148	1.0000	67.6538
CS_28, A14P20_BRIGADA_OUTRA_CALDAS DA RAINHA	400	0.0148	1.0000	67.6538
100.0, CARANGUEJEIRA	001413P_BRIGADA_OUTRA_CARANGUEJEIRA	0.0109	1.0000	65.1481
001413P_BRIGADA_OUTRA_CARANGUEJEIRA	100.0, CARANGUEJEIRA	0.0109	0.7130	65.1481

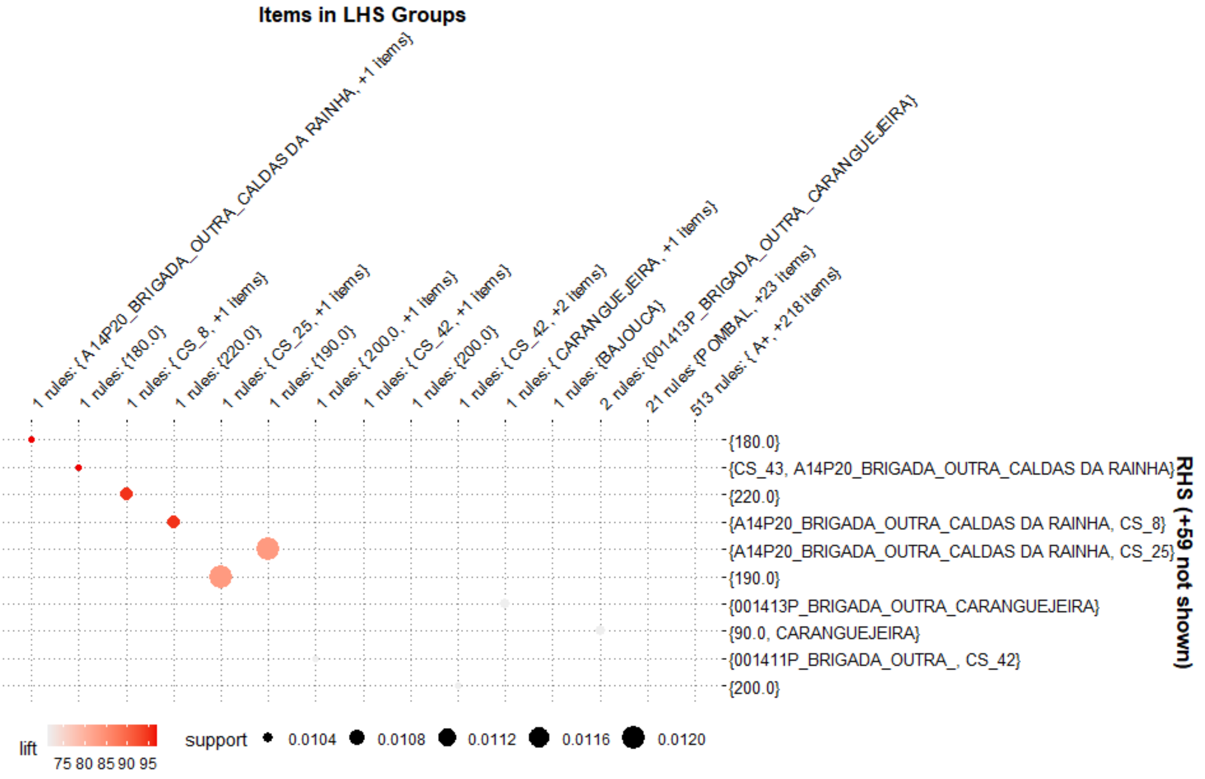


Figure 5.9: Grouped matrix of the association rules obtained with FP-Max for Leiria in 2019.

28, 42, 43, and 45. This means that the brigades performing blood collections at the end of March, beginning of April, end of June, and end of October are the ones with the highest lift. The number of total donations done by a donor is not highlighted, meaning that the values do not present a tendency.

The total number of donations expected per brigade varies between 90 and 400. The latter was only registered in 2017, meaning that it might have been reduced in the other years. In comparison to Lisboa, however, considering Leiria also had donations in hospitals, we do not see a small expectancy value that represented a daily collection. This might mean that even though the hospitals are the main place where blood is donated in this district, they do not collect it daily. Instead, the brigades expect a higher number of donations each time.

5.4 Sequential Pattern Mining

In this sequential pattern mining approach we tested the `PrefixSpan` algorithm [39], the `CM-Spade` algorithm [43], and the `Fournier08` algorithm [44], which were explained in Section 2.2 from Chapter 2. The respective workflow can be seen in Figure 5.10.

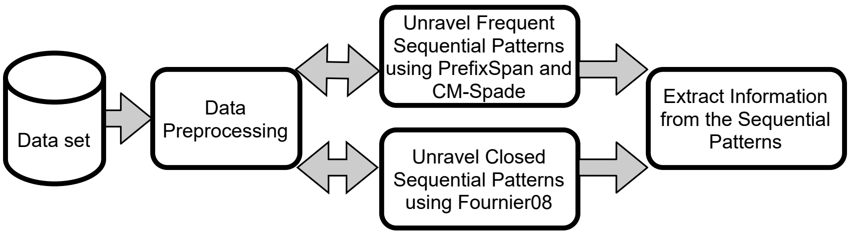

Data set
Data Preprocessing
Unravel Frequent Sequential Patterns using PrefixSpan and CM-Spade
Unravel Closed Sequential Patterns using Fournier08
Extract Information from the Sequential Patterns

Figure 5.10: Workflow for the sequential pattern mining approach.

5.4.1 Data Pre-processing

The initial part of the pre-processing done for this approach was similar to the one done for the transactional mining and association rule learning one, so we will not cover it again. We began by using the three data sets that were created in Section 5.1 and performed the same pre-processing thought process up until the point before creating tags for each feature with integer values.

To use the algorithms for this approach, we utilized the implementations available in the SPMF library[4]. SPMF is a Java open-source software and data mining library, created by Philippe Fournier-Viger, and contains implementations of different pattern mining algorithms. The input for the algorithms we planned on using was a sequence data set or a time-extended sequence data set. The difference between a sequence data set and a time-extended one is that, besides having sequences where each sequence is a list of itemsets, each itemset is annotated with a timestamp.

The input file format for the SPMF library's implementation of the algorithms we used had to respect the following characteristics:

[4] `https://www.philippe-fournier-viger.com/spmf/`

- It is a text file where each line represents a sequence (or a time-extended sequence) from a sequence data set;
- Each of the items in the itemsets is represented by a positive integer, and items from the same itemset within a sequence are separated by single spaces;
- The end of an itemset is indicated by a ”-1”, and after all the itemsets, the end of a sequence is indicated by a ”-2”;
- If it is a time-extended sequence, each itemset is first represented by its timestamp, which is a positive integer between the ”$<$” and ”$>$” symbols.

The implementations of these algorithms assume that the items are sorted according to a total order in each itemset, and that no item appears more than once in the same itemset.

In order to use our data, we created the input file with the respective format, while indexing each of the different values present in our data sets to unique integer values. Each sequence represented a blood collection brigade, and each itemset in the sequence represented a donation that was collected by that brigade.

5.4.2 Applying the Algorithms

We started by using the `PrefixSpan` algorithm, since it is one of the most popular sequential pattern mining algorithms. However, due to the large number of different values present in some of the features, the number of different indexed values was too large for the algorithm to be able to run with acceptable computational time. So we started reducing the cardinality of the features by:

- Keeping only the 10 most common values from the feature ’dador_profissao’ and referring to the other ones as a new value ”OUTRA”;
- Replacing the ’dador_localidade_postal’ feature with ’dador_naturalidade_distrito’, reducing the value cardinality from $4\,555$ to 29 and joined the different values corresponding to all the Portuguese islands to a single new value ”ILHAS” reducing the value cardinality from 29 to 19;
- Binning the values in the feature ’dador_total_dadivas’ to $[0,5[$, $[5,10[$, $[10,20[$, $[20,30[$, $[30,40[$, $[40,60[$, $[60,80[$, $80+$. The bin dimension increases since the amount of donors with a higher donation count are not as common as donors with a smaller one;
- Binning the values in the feature ’dador_idade’ to $[18,25[$, $[25,35[$, $[35,45[$, $[45,55[$, $[55,65[$, $65+$. The first bin starts at 18, since that is the minimum legal age to be able to donate blood;
- Binning the values in the feature ’colheita_nr_dadores_previstos’ to $[0,25[$, $[25,50[$, $[50,100[$, $[100,150[$, $150+$;

- Binning the values in the feature 'colheita_mes' so that a year would be divided into trimesters.

However, this change was not enough to decrease the computational time. So, we looked at and tested the other sequential pattern mining algorithms available in the SPMF library, and we concluded that the fastest algorithm was CM-Spade, as it was mentioned in the paper where the algorithm was proposed [43] and in the survey conducted by P. Fournier-Viger et al. [38]. The comparison between the PrefixSpan and CM-Spade can be seen in Figure 5.11, where we compare the scalability of the algorithms in data sets with 1 or 2 itemsets (i.e. donations) per sequence (i.e. brigade), while increasing the total number of sequences.

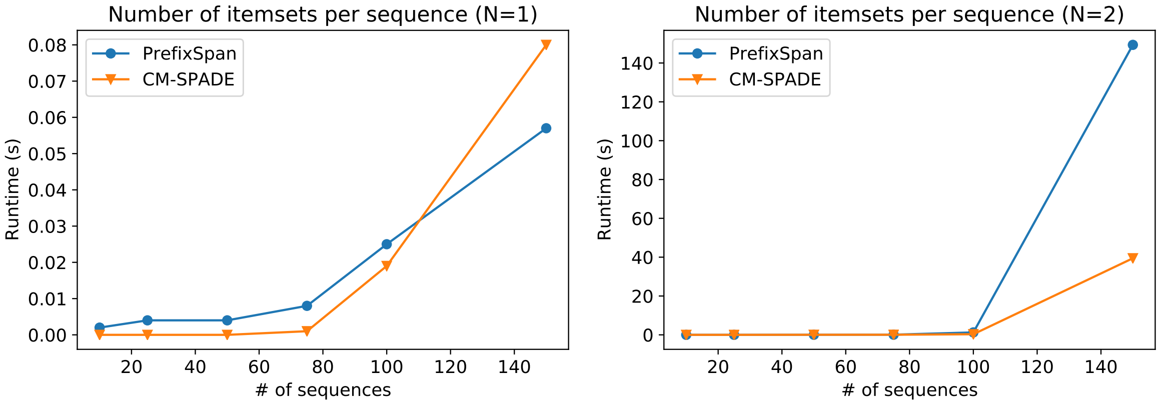


Figure 5.11: Runtime scalability of the PrefixSpan and CM-Spade algorithms, according to the number of itemsets (donations) per sequence (brigade) over data sets with an increasing total number of sequences. Left: represents data sets with 1 itemset per sequence. Right: focuses on data sets with 2 itemsets per sequence.

Even so, our data sets had more than 2 itemsets per sequence. For example, the data set for Lisboa in 2019, had $52\,649$ donations collected by 384 different brigades, meaning that on average each sequence had 137 itemsets. Figure 5.12 shows the discrepancy in the runtime taken when increasing the number of itemsets per sequence (from 1 to 3 itemsets) for data sets with the same number of sequences. This meant that the CM-Spade algorithm was not able to run our data sets with acceptable computational time either.

To solve this issue, we compressed the data sets as much as possible, which was done by reducing the total amount of donations per brigade to a maximum of 4 (one per trimester). To do this, we calculated the mode of each of the features composing all the donations done in a brigade in each trimester and collapsed them into the format of a single donation. Following the example given above regarding the 2019 Lisboa data set, it had now been reduced to an average of 1 itemset per sequence.

To increase the amount of information obtained from the data sets, we started to gradually remove some of the changes that had been made to the data sets until, after several attempts, we reached a point where we decided to stop. After such attempts, we ended up completely removing the changes

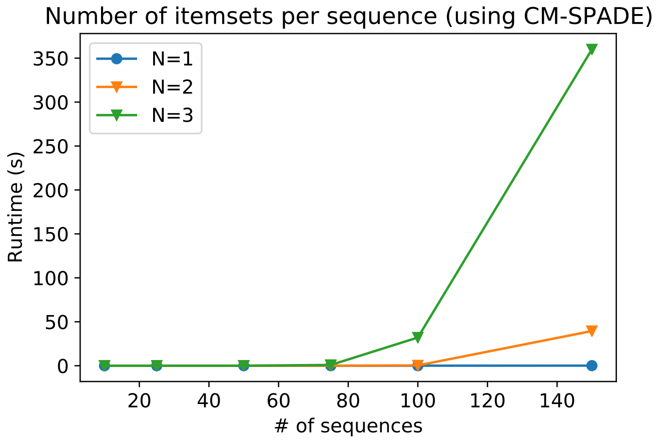


Figure 5.12: Runtime scalability of the CM-Spade algorithm, according to the number of itemsets (donations) per sequence (brigade) over data sets with an increasing total number of sequences. Comparison between data sets with 1 to 3 itemsets per sequence.

done to the features 'dador_total_dadivas' and 'colheita_nr_dadores_previstos', removing the binning that had been performed, and to the features 'dador_profissao' and 'colheita_mes', where now all the different jobs and months were being taken into account, respectively. The removal of the changes made to the 'colheita_mes' feature, also meant that the total amount of donations per brigade was increased to 12 instead of 4, so now in the 2019 Lisboa data set we had an average of 2 itemsets per sequence (793 donations collected by 384 different brigades). The changes done to the features 'dador_naturalidade_distrito' and 'dador_idade' were kept. Seeing that the readability of the patterns was difficult, we also decided to tag each of the features with the respective feature name followed by an "=" symbol (e.g. dador_tipo_sangue = O+).

Finally, with our data sets transformed into the proper input file format, we used them as input to the SPMF library's implementation of the CM-Spade and Fournier08 algorithms. Since the Fournier08 algorithm needed a time-extended file, we used the month m to represent the timestamps $< m >$.

5.4.3 Results

We will now present the results we obtained using the CM-Spade and Fournier08 algorithms for the top 5 districts, according to the brigade's district of origin, in 2017, 2018, and 2019. Firstly, we start by presenting the top 5 patterns according to the support for the three years using the Fournier08 algorithm. In this first part, we decided to only show the patterns obtained with this algorithm, since it outputs closed patterns. Meanwhile, CM-Spade outputs frequent patterns, which fills the top results with patterns that are sub-sequences. Secondly, we show the top 5 patterns with the greatest length obtained with both the CM-Spade and Fournier08 algorithms. This choice was made considering that we wanted to present the patterns with the highest amount of information, which would therefore be more relevant for us than the ones obtained using only the support metric.

We were not able to maintain the same minimum support threshold throughout the experience for each district when using the CM-Spade algorithm, seeing that for some years (mainly 2019), the SPMF GUI would crash when trying to show the patterns using the Pattern Viewer tool, which was the only way of exporting our results as a CSV file to further study the results according to their support and length. For this reason, we will present the threshold defined for each district in their own sections. When using the Fournier08 algorithm, the minimum support threshold was 0.05.

For the results obtained with both algorithms, please note that the end of an itemset is represented by a $|$. For the ones obtained with the Fournier08 algorithm, please keep in mind that since the timestamps start at 0, January is represented by that value, February is represented by 1, and so forth until 11, which represents December. As the dimension of the tables containing the patterns with the greatest lengths for the three years was too large, we decided to only present the table with the results for 2019 in this chapter. These algorithms allow us to check the support of a sequence of donations that were done by donors with particular characteristics in a certain brigade over a year. Taking into account that we applied the mode for each month in order to reduce the data set, please keep in mind that each itemset characterizes the most common type of information for each feature regarding a donor in each month.

5.4.3.A Lisboa

We begin by presenting Table 5.12 which displays the top 5 patterns in 2017, 2018, and 2019 for this district using the Fournier08 algorithm with minimum support threshold of 0.05 according to their support. From it, we can see that throughout the years, the donors in Lisboa had a higher probability of being born in this district, and being married, female, as well as having blood type A+.

Table 5.13 shows the top 5 patterns with the greatest length obtained using the CM-Spade algorithm for the 2019 data set with a minimum support threshold of 0.05. As an example, in a certain month, if we have donations that were done by a majority of donors that were married and born in Lisboa, there is 0.0521 support of in a following month the donations being mostly done by donors that were female, between 45 and 55 years old, married and born in Lisboa, which was then followed by a month with mostly donations where the donor was female, married and born in Lisboa. We can look at Table 5.14 with the same train of thought, with the nuance that now we are taking into account the exact months as time stamps to determine the sequences' order, since we are using the Fournier08 algorithm. So for Lisboa, we can see that the top 5 patterns highlight the donations made in the sequence of months of January, then May, and then September.

The top patterns for this district are generally characterized by months when the donors were mostly married, female, had blood type A+, and were born in Lisboa. There were also some patterns referring to the age of the donors as being between 45 and 55 years old, and of them working at an unknown

(”DESCONHECIDA”) job.

Table 5.12: Top 5 patterns obtained for the 2017, 2018, and 2019 data sets of Lisboa using the `Fournier08` algorithm with minimum support threshold of 0.05. Sorted by support.

Pattern	Support
2017	
<0>dador_naturalidade_distrito = LISBOA |	1.0000
<0>dador_sexo = feminino dador_naturalidade_distrito = LISBOA |	0.7732
<0>dador_tipo_sangue = A+ dador_naturalidade_distrito = LISBOA |	0.7423
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.6469
<0>dador_profissao = DESCONHECIDA dador_naturalidade_distrito = LISBOA |	0.6211
2018	
<0>dador_naturalidade_distrito = LISBOA |	0.9975
<0>dador_sexo = feminino dador_naturalidade_distrito = LISBOA |	0.7563
<0>dador_tipo_sangue = A+ |	0.6859
<0>dador_tipo_sangue = A+ dador_naturalidade_distrito = LISBOA |	0.6834
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.6156
2019	
<0>dador_naturalidade_distrito = LISBOA |	0.9974
<0>dador_sexo = feminino dador_naturalidade_distrito = LISBOA |	0.7448
<0>dador_tipo_sangue = A+ |	0.7109
<0>dador_tipo_sangue = A+ dador_naturalidade_distrito = LISBOA |	0.7083
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.6172

Table 5.13: Top 5 patterns obtained for the 2019 data set of Lisboa using the `CM-Spade` algorithm with minimum support threshold of 0.05. Sorted by length.

Pattern	Support

dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |
dador_sexo = feminino dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |
dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |
	0.0521

dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.0521

dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |
dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |
dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |
	0.0521

dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.0599

dador_estado_civil = CASADO dador_profissao = DESCONHECIDA dador_naturalidade_distrito = LISBOA |
dador_estado_civil = CASADO dador_profissao = DESCONHECIDA dador_naturalidade_distrito = LISBOA |
dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |
	0.0521

5.4.3.B Porto

Looking at Table 5.15 we can see the top 5 patterns in 2017, 2018, and 2019 for this district using the `Fournier08` algorithm with minimum support threshold of 0.05 according to their support. From it, we can see that throughout the years, the donors that donated blood in Porto had a higher probability of being born in this district, as well as being married, female, and, in 2018 and 2019, having blood type A+.

Table 5.14: Top 5 patterns obtained for the 2019 data set of Lisboa using the `Fournier08` algorithm with minimum support threshold of 0.05. Sorted by length.

Pattern	Support
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | <8>dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.0599
<0>dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | <8>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.0599
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | <8>dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.0573
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | <4>dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA | <8>dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.0547
<0>dador_sexo = feminino dador_naturalidade_distrito = LISBOA | <4>dador_sexo = feminino dador_naturalidade_distrito = LISBOA | <8>dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LISBOA |	0.0547

Table 5.15: Top 5 patterns obtained for the 2017, 2018, and 2019 data sets of Porto using the `Fournier08` algorithm with minimum support threshold of 0.05. Sorted by support.

Pattern	Support
2017	
<0>dador_naturalidade_distrito = PORTO |	0.9647
<0>dador_sexo = feminino |	0.8941
<0>dador_sexo = feminino dador_naturalidade_distrito = PORTO |	0.8627
<0>dador_estado_civil = CASADO |	0.8314
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |	0.8078
2018	
<0>dador_naturalidade_distrito = PORTO |	0.9654
<0>dador_sexo = feminino |	0.8788
<0>dador_sexo = feminino dador_naturalidade_distrito = PORTO |	0.8485
<0>dador_estado_civil = CASADO |	0.7922
<0>dador_tipo_sangue = A+ |	0.7836
2019	
<0>dador_naturalidade_distrito = PORTO |	0.9700
<0>dador_sexo = feminino |	0.8927
<0>dador_sexo = feminino dador_naturalidade_distrito = PORTO |	0.8712
<0>dador_tipo_sangue = A+ |	0.8069
<0>dador_estado_civil = CASADO |	0.7811

In table 5.16, we can see the top 5 patterns with the greatest length obtained using the `CM-Spade` algorithm for the 2019 data set with a minimum support threshold of 0.05. For example, in this district for a certain month, if we have donations that were done by a majority of donors that were female, between 45 and 55 years old, had blood type A+, married and born in Porto, we have a 0.0601 support of two of the following months in that brigade, with most of their donations being done by donors with the same characteristics in that year. A similar thought process can be followed in Table 5.17, with the nuance that now we are taking into account the exact months as time stamps to determine the sequences' order. So for Porto, we can see that the top 5 patterns highlight the donations made in the sequence of months of

January, then May, and then September.

Overall, the top patterns for this district are characterized by months when the donors were mostly married, female, between 45 and 55 years old, had blood type A+, and were born in Porto.

Table 5.16: Top 5 patterns obtained for the 2019 data set of Porto using the `CM-Spade` algorithm with minimum support threshold of 0.05. Sorted by length.

Pattern	Support

dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
	0.0601

dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
	0.0601

dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
	0.0601

dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
	0.0601

dador_sexo = feminino dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |
	0.0644

5.4.3.C Aveiro

Table 5.18 presents us the top 5 patterns in 2017, 2018, and 2019 for this district using the `Fournier08` algorithm with minimum support threshold of 0.05 according to their support. From it, we can see that throughout the years, the donors that donated blood in Aveiro had a higher probability of being born in this district, as well as being married and female.

Table 5.19 presents the top 5 patterns according to the greatest length, using the `CM-Spade` algorithm for the 2019 data set with a minimum support threshold of 0.07. For example, in a certain month, if we have donations that were done by a majority of donors that were female, between 45 and 55 years old, had blood type A+, married, worked as factory employees ("EMPREGADO FABRIL"), and were born in Aveiro, there is a 0.0723 support of a following month being mainly composed of donations being done by donors that were female, had blood type A+, married, worked as factory employees, and were born in this district, which was then followed by a month when the donors had blood type A+, were married, worked as factory employees, and were born in Aveiro. Similarly, we can look at Table 5.20, but we need

Table 5.17: Top 5 patterns obtained for the 2019 data set of Porto using the `Fournier08` algorithm with minimum support threshold of 0.05. Sorted by length.

Pattern	Support
<0>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <4>dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <8>dador_sexo = feminino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |	0.0601
<0>dador_sexo = feminino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <4>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <8>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |	0.0515
<0>dador_sexo = feminino dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <4>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <8>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |	0.0515
<0>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <4>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <8>dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_naturalidade_distrito = PORTO |	0.0515
<0>dador_sexo = feminino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <4>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO | <8>dador_sexo = feminino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = PORTO |	0.064378

Table 5.18: Top 5 patterns obtained for the 2017, 2018, and 2019 data sets of Aveiro using the `Fournier08` algorithm with minimum support threshold of 0.05. Sorted by support.

Pattern	Support
2017	
<0>dador_naturalidade_distrito = AVEIRO |	0.9888
<0>dador_estado_civil = CASADO |	0.9382
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO |	0.9270
<0>dador_sexo = feminino |	0.7416
<0>dador_sexo = feminino dador_naturalidade_distrito = AVEIRO |	0.7303
2018	
<0>dador_naturalidade_distrito = AVEIRO |	0.9886
<0>dador_estado_civil = CASADO |	0.9314
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO |	0.9200
<0>dador_sexo = feminino |	0.7543
<0>dador_sexo = feminino dador_naturalidade_distrito = AVEIRO |	0.7429
2019	
<0>dador_naturalidade_distrito = AVEIRO |	0.9940
<0>dador_estado_civil = CASADO |	0.9036
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO |	0.8976
<0>dador_sexo = feminino |	0.7590
<0>dador_sexo = feminino dador_naturalidade_distrito = AVEIRO |	0.7530

to keep in mind that now we are taking into account the exact months as time stamps to determine the sequences' order. So for Aveiro, we can see that the top 5 patterns highlight the donations made in the sequence of months starting in January and ending in October.

In this district, the top patterns discovered are characterized by months when the donors are mainly married, female, and were born in Aveiro. There were also some patterns indicating that a majority of them worked as factory employees, were between 45 and 55 years old, and had blood type A+.

Table 5.19: Top 5 patterns obtained for the 2019 data set of Aveiro using the `CM-Spade` algorithm with minimum support threshold of 0.07. Sorted by length.

<table>
<tr><th>Pattern</th><th>Support</th></tr>
<tr><td>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_sexo = feminino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO |</td><td>0.0723</td></tr>
<tr><td>dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO |</td><td>0.0723</td></tr>
<tr><td>dador_sexo = feminino dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO |</td><td>0.0723</td></tr>
<tr><td>dador_sexo = feminino dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO |</td><td>0.0723</td></tr>
<tr><td>dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_sexo = feminino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO |</td><td>0.0783</td></tr>
</table>

5.4.3.D Braga

Table 5.21 displays the top 5 patterns in 2017, 2018, and 2019 for this district using the `Fournier08` algorithm with minimum support threshold of 0.05 according to their support. From this table, it is visible that throughout the years, the donors that donated blood in Braga had a higher probability of being born in this district, as well as being married and female.

In table 5.22, we can see the top 5 patterns with the greatest length obtained using the `CM-Spade` algorithm for the 2019 data set with a minimum support threshold of 0.05. If a certain month in this district was made off donations that were done by a majority of donors who were between 45 and 55 years old, had blood type A+, were married, worked as factory employees (”EMPREGADO FABRIL”), and were born in Braga, we have a 0.0504 support of a subsequent month in the same brigade being mostly done by donors that were female, between 45 and 55 years old, had blood type A+, married,

Table 5.20: Top 5 patterns obtained for the 2019 data set of Aveiro using the `Fournier08` algorithm with minimum support threshold of 0.05. Sorted by length.

Pattern	Support
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <1>dador_idade = [45,55[dador_naturalidade_distrito = AVEIRO | <2>dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <3>dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <5>dador_naturalidade_distrito = AVEIRO | <6>dador_naturalidade_distrito = AVEIRO | <7>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <8>dador_tipo_sangue = A+ dador_naturalidade_distrito = AVEIRO | <9>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO |	0.0542
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <1>dador_idade = [45,55[dador_naturalidade_distrito = AVEIRO | <2>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <3>dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <5>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <6>dador_naturalidade_distrito = AVEIRO | <7>dador_naturalidade_distrito = AVEIRO | <8>dador_naturalidade_distrito = AVEIRO | <9>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <10>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO |	0.0542
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <1>dador_naturalidade_distrito = AVEIRO | <2>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <3>dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <5>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <6>dador_naturalidade_distrito = AVEIRO | <7>dador_naturalidade_distrito = AVEIRO | <8>dador_naturalidade_distrito = AVEIRO | <9>dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <10>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO |	0.0542
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <1>dador_naturalidade_distrito = AVEIRO | <2>dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | <3>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <5>dador_naturalidade_distrito = AVEIRO | <6>dador_naturalidade_distrito = AVEIRO | <7>dador_naturalidade_distrito = AVEIRO | <8>dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <9>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <10>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO |	0.0542
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <1>dador_naturalidade_distrito = AVEIRO | <2>dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = AVEIRO | <3>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <5>dador_naturalidade_distrito = AVEIRO | <6>dador_naturalidade_distrito = AVEIRO | <7>dador_naturalidade_distrito = AVEIRO | <8>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <9>dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO | <10>dador_estado_civil = CASADO dador_naturalidade_distrito = AVEIRO |	0.0542

worked as factory employees, and were born in this district, which was then followed by a month when the donations were made by donors who were between 45 and 55 years old, had blood type A+, were married, worked as factory employees, and were born in Braga, and finally followed by another month when the donors were between 45 and 55 years old, married, worked as factory employees, and were born in this district. Table 5.23 presents the sequential patterns' results with timestamps, so now we take into account the exact months as time stamps to determine the sequences' order. So, for Braga, we can see that the top 5 patterns highlight the donations made in the sequence of months starting in January, followed by May, and then September.

Table 5.21: Top 5 patterns obtained for the 2017, 2018, and 2019 data sets of Braga using the Fournier08 algorithm with minimum support threshold of 0.05. Sorted by support.

Pattern	Support
2017	
<0>dador_naturalidade_distrito = BRAGA |	0.9679
<0>dador_estado_civil = CASADO |	0.8526
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = BRAGA |	0.8269
<0>dador_sexo = feminino |	0.8013
<0>dador_sexo = feminino dador_naturalidade_distrito = BRAGA |	0.7756
2018	
<0>dador_naturalidade_distrito = BRAGA |	0.9671
<0>dador_estado_civil = CASADO |	0.8289
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = BRAGA |	0.7961
<0>dador_sexo = feminino |	0.7697
<0>dador_sexo = feminino dador_naturalidade_distrito = BRAGA |	0.7368
2019	
<0>dador_naturalidade_distrito = BRAGA |	0.9712
<0>dador_estado_civil = CASADO |	0.8561
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = BRAGA |	0.8345
<0>dador_sexo = feminino |	0.8273
<0>dador_sexo = feminino dador_naturalidade_distrito = BRAGA |	0.8058

To sum up, the top patterns for Braga are characterized by months when the donors were mostly married, between 45 and 55 years old, working as factory employees, and were born in Braga. There were also some patterns, referring to the majority of the donors as being female and having blood type A+.

Table 5.22: Top 5 patterns obtained for the 2019 data set of Braga using the `CM-Spade` algorithm with minimum support threshold of 0.05. Sorted by length.

<table>
<tr><th>Pattern</th><th>Support</th></tr>
<tr><td>dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
<tr><td>dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
<tr><td>dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_sexo = feminino dador_idade = DI_[45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
<tr><td>dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_sexo = feminino dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
<tr><td>dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA | dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
</table>

Table 5.23: Top 5 patterns obtained for the 2019 data set of Braga using the `Fournier08` algorithm with minimum support threshold of 0.05. Sorted by length.

<table>
<tr><th>Pattern</th><th>Support</th></tr>
<tr><td><0>dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<4>dador_sexo = feminino dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<8>dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
<tr><td><0>dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<4>dador_sexo = feminino dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<8>dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
<tr><td><0>dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<4>dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<8>dador_sexo = feminino dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
<tr><td><0>dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<4>dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<8>dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
<tr><td><0>dador_idade = [45,55[dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<4>dador_idade = [45,55[dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |
<8>dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = BRAGA |</td><td>0.0504</td></tr>
</table>

5.4.3.E Leiria

In Table 5.24, we present the top 5 patterns in 2017, 2018, and 2019 for this district using the Fournier08 algorithm with a minimum support threshold of 0.05 according to their support. From it, we can see that throughout the years, the donors that donated blood in Leiria had a higher probability of being born in this district, as well as being married and having blood type A+. In 2017, we can also see that there was a high probability of them being female, and in 2019 of them being between 45 and 55 years old.

Table 5.24: Top 5 patterns obtained for the 2017, 2018, and 2019 data sets of Leiria using the Fournier08 algorithm with minimum support threshold of 0.05. Sorted by support.

Pattern	Support
2017	
<0>dador_naturalidade_distrito = LEIRIA |	0.9808
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA |	0.8269
<0>dador_tipo_sangue = A+ |	0.6731
<0>dador_sexo = feminino |	0.6538
<0>dador_sexo = feminino dador_naturalidade_distrito = LEIRIA |	0.6442
2018	
<0>dador_naturalidade_distrito = LEIRIA |	0.9709
<0>dador_estado_civil = CASADO |	0.8544
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA |	0.8447
<0>dador_tipo_sangue = A+ |	0.7864
<0>dador_tipo_sangue = A+ dador_naturalidade_distrito = LEIRIA |	0.7573
2019	
<0>dador_naturalidade_distrito = LEIRIA |	1.0000
<0>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA |	0.8152
<0>dador_tipo_sangue = A+ dador_naturalidade_distrito = LEIRIA |	0.7717
<0>dador_idade = [45,55[dador_naturalidade_distrito = LEIRIA |	0.7065
<0>dador_sexo = feminino dador_naturalidade_distrito = LEIRIA |	0.6739

Following up with Table 5.25, which displays the top 5 patterns with the greatest length obtained using the CM-Spade algorithm for the 2019 data set, with a minimum support threshold of 0.06. Looking at the first pattern presented in this table, if in a certain month we have donations that were made by a majority of donors that were male, married, and born in Leiria, we have a 0.0652 support of a subsequent month of donations in the same brigade having mostly donors that had blood type A+, were married, and born in this district. Then, it is followed by a month when the donations were mostly made by donors who were between 45 and 55 years old, married, and born in Leiria. This is then followed by another month when the donors were male, married, and born in the district. And finally, followed by another month with mostly donors that were married, worked as factory employees, and were born in this district. A similar thought process is presented in Table 5.26, with the nuance that now we are taking into account the exact months as time stamps to determine the sequences' order. So for Leiria, we can see that the top 5 patterns highlight the donations made in the sequence of months starting in January and ending September.

Overall, the top patterns for this district are generally characterized by months when the donors were mostly married, female, and were born in Leiria. There were also some patterns indicating that a majority of them worked as factory employees, were between 45 and 55 years old, and had blood type A+.

Table 5.25: Top 5 patterns obtained for the 2019 data set of Leiria using the `CM-Spade` algorithm with minimum support threshold of 0.06. Sorted by length.

<table>
<tr><th>Pattern</th><th>Support</th></tr>
<tr><td>dador_sexo = masculino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_sexo = masculino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = LEIRIA |</td><td>0.0652</td></tr>
<tr><td>dador_sexo = masculino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_sexo = masculino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = LEIRIA |</td><td>0.0652</td></tr>
<tr><td>dador_sexo = masculino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_sexo = feminino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_sexo = masculino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_estado_civil = CASADO dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = LEIRIA |</td><td>0.0652</td></tr>
<tr><td>dador_sexo = masculino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_sexo = masculino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = LEIRIA |</td><td>0.0652</td></tr>
<tr><td>dador_sexo = masculino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_sexo = feminino dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_sexo = masculino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | dador_profissao = EMPREGADO_FABRIL dador_naturalidade_distrito = LEIRIA |</td><td>0.0652</td></tr>
</table>

Table 5.26: Top 5 patterns obtained for the 2019 data set of Leiria using the `Fournier08` algorithm with minimum support threshold of 0.05. Sorted by length.

<table>
<tr><th>Pattern</th><th>Support</th></tr>
<tr><td><0>dador_sexo = masculino dador_naturalidade_distrito = LEIRIA | <1>dador_naturalidade_distrito = LEIRIA | <2>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <3>dador_sexo = feminino dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <5>dador_sexo = masculino dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <6>dador_naturalidade_distrito = LEIRIA | <7>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <8>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA |</td><td>0.0543</td></tr>
<tr><td><0>dador_naturalidade_distrito = LEIRIA | <1>dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <2>dador_naturalidade_distrito = LEIRIA | <3>dador_naturalidade_distrito = LEIRIA | <4>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <5>dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <6>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <7>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <8>dador_naturalidade_distrito = LEIRIA | <9>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <10>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA |</td><td>0.0543</td></tr>
<tr><td><0>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <1>dador_naturalidade_distrito = LEIRIA | <2>dador_sexo = masculino dador_tipo_sangue = A+ dador_naturalidade_distrito = LEIRIA | <3>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <4>dador_sexo = feminino dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <5>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <6>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <7>dador_naturalidade_distrito = LEIRIA | <8>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA |</td><td>0.0543</td></tr>
<tr><td><0>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <1>dador_naturalidade_distrito = LEIRIA | <2>dador_tipo_sangue = A+ dador_naturalidade_distrito = LEIRIA | <3>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <4>dador_idade = DI_[45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <5>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <6>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <7>dador_naturalidade_distrito = LEIRIA | <8>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <9>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA |</td><td>0.0543</td></tr>
<tr><td><0>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <1>dador_naturalidade_distrito = LEIRIA | <2>dador_tipo_sangue = A+ dador_naturalidade_distrito = LEIRIA | <3>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <4>dador_sexo = feminino dador_idade = [45,55[dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <5>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <6>dador_tipo_sangue = A+ dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA | <7>dador_naturalidade_distrito = LEIRIA | <8>dador_estado_civil = CASADO dador_naturalidade_distrito = LEIRIA |</td><td>0.0543</td></tr>
</table>

6

Conclusion

Contents

This chapter presents and discusses the dissertation's final conclusions, as well as the contributions of our study to the topic of research. It also recalls some of the limitations of the algorithms used in our analysis and discusses future approaches to the object of study.

6.1 Conclusions

In this thesis, we have applied three different novel approaches to a blood donation data set from IPST. We started by transforming the data set into a similarity network to be used as input for the `Louvain Method`, which is a community finding algorithm. Then we applied the `Apriori`, `FP-Growth`, and `FP-Max` algorithms to the data set to perform a transactional mining and association rule learning approach. Finally, we also applied sequential pattern mining methods, namely `PrefixSpan`, `CM-Spade`, and `Fournier08` to the data set. We discuss below the results and performance of the different approaches from the point of view of computational time and resources needed to run them, as well as the results they generated.

Regarding performance, out of the three approaches, the one best suited for the IPST data was the transactional mining and association rule learning approach, seeing that it was not only able to deal with the highest amount of data, but was also the fastest one to generate results. However, and even if it was the best out of these three approaches, it is important to keep in mind that it was still similarly affected by the fact that our data was composed of a large data set, both vertically and horizontally. These approaches are mostly used to deal with transactional data, so they were optimized to deal with data sets with a high number of rows, but a low number of columns, which was not the case in our data set and was reflected in the performance of the used approaches. To deal with this, we had to reduce the data set's size and divide it into smaller subsets according to specific years (2017, 2018, and 2019) and/or districts and perform other changes as a form of pre-processing, which are described in Chapter 5 in more detail.

As for the quality of the results themselves, we can consider them to be relevant or interesting if they satisfy at least one of the following criteria: 1) can be easily interpreted by humans; 2) are valid with a certain degree of certainty; 3) can be useful in some way; and 4) present new information. Taking these criteria into account, out of the three, the one that presented the worst results was the community finding approach. This choice was made since it was the one that lacked a means of explanation that could be easily interpreted by humans, considering that the modularity focuses on measuring the strength of the division of a network into modules. So, it does not present any real readability for a human, and it does not present any degree of certainty, especially since we were the ones defining the similarity threshold in order to create the network, generating a bias. Focusing on the other two approaches, we can see that there are some differences in the results obtained between them in the top 5 districts in terms of length. This might be due to the fact that, in order to perform sequential pattern mining, we had to join the information of a brigade for each month in a single row by performing the mode of the different features. In the sequential pattern mining approach, it is also important to keep in mind that the results obtained by `CM-Spade` and `Fournier08` when sorted by length are different, presenting two types of information. In the former, as long as the pattern follows a sequence, the time frame does not need to be taken into account, while in the latter, we also take into account the exact time frame, so the results present more accuracy when defining the time sequence.

To conclude, we were able to find patterns that contained information to be provided to IPST in the form of hints to satisfy its activity plans while optimizing the operations of the teams they organize, which was the goal of this thesis. Some examples of hints that could be given according to the pattern we uncovered include:

- The most relevant brigades of each Portuguese district were identified, so more resources could be dispatched to them, while reducing the amount of resources spent on the less relevant ones. This could be done, for example, by improving the frequency of blood collections in a place where the most relevant brigades have been collecting blood donations, while decreasing the frequency of less relevant ones;

- The jobs with the highest number of donations were also identified for each district, so IPST could create new brigades or dispatch older ones to places near their places of work, such as universities in Lisboa to collect blood from higher education students, or factories in Aveiro to collect blood from factory employees;

- The months with the highest donation rate were also identified, which allows us to tell IPST hints on when they could scatter their brigades throughout the year for each district;

- We identified patterns that showed that there is an increase in the number of donors that are in the 20 to 25 year bracket, or female donors over the past three years. So, IPST could advertise more to donors with these characteristics in order to bring a higher number of them to donate or keep donating, since it has been proven to be successful. Another option would be to advertise for the other types of donors that are still part of a minority to increase the number of donations made by them.

We would also like to bring up the fact that even though some of the information could be gathered without using this sort of algorithms, when using pattern mining algorithms we are able to retain the probability of some aspect happening in the real world, for example, in the form of the support metric. This can be considered relevant information when we take into account that these algorithms might justify some of the ideas that people that work in this field may have, or even contradict them.

6.2 Limitations and Future Work

Besides the amount of data present in the data set proving to decrease the performance of the approaches, it together with the quality of the data (for example, there were cases where the cardinality for some of the features was too high, which hindered the obtainment of more concrete results, or features where there was a high number of wrong or missing values) were two issues that were referred throughout this document as having an impact on the approaches chosen. However, we believe that if we had chosen any other approach, the same issues would have come up. With this in mind, we believe that in the future, the data needs to be treated better as a whole, removing the incongruities as well as taking care of the unknown or other values present in the data set.

When this reiteration of the data is done, we could then apply our approaches again to see the difference in the results, as well as new ones. One example of a new approach would be a biclustering one. For it, we could use algorithms like Qubic2 [77] or the ones available in the BicPams software [25] to study a biclustering approach and the modules discovered by it when applied to the IPST data.

To have a professional point of view on the topic and to know the utility and novelty of our results, we have also communicated the most relevant results to IPST.

Bibliography

[1] T. Poisot, “An a posteriori measure of network modularity,” *F1000Research*, vol. 2, p. 130, 12 2013.

[2] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. USA: Elsevier, 2011.

[3] M.-H. Chiu, Y.-R. Yu, H. Liaw, and L. Hao, “The use of facial micro-expression state and tree-forest model for predicting conceptual-conflict based conceptual change,” in *ESERA*, 01 2016, p. 187.

[4] J. T. Blake and M. Hardy, “A generic modelling framework to evaluate network blood management policies: The canadian blood services experience,” *Operations Research for Health Care*, vol. 3, no. 3, pp. 116–128, 2014.

[5] IPST, “Plano de atividades do IPST,IP 2020-2022 | Homologado pela Ministra da saúde,” 8 2020.

[6] FCT, “LAIfeBlood Project,” 2019. [Online]. Available: https://www.fct.pt/

[7] IPST, “Plano nacional de emergência para eventos com potencial impacto na missão do IPST,IP,” 5 2014.

[8] IPST , “Relatório de atividades do IPST,IP - 2019 | Homologado pela Ministra da saúde,” 8 2020.

[9] National Research Council *et al.*, *Network science*. Washington, DC: The National Academies Press, 2006.

[10] S. Wasserman, K. Faust *et al.*, *Social network analysis: Methods and applications*. Cambridge university press, 1994, vol. 8.

[11] S. R. Proulx, D. E. Promislow, and P. C. Phillips, “Network thinking in ecology and evolution,” *Trends in ecology & evolution*, vol. 20, no. 6, pp. 345–353, 2005.

[12] M. Barthélemy, “Spatial networks,” *Physics Reports*, vol. 499, no. 1-3, pp. 1–101, 2011.

[13] J. Lumijärvi, J. Laurikkala, and M. Juhola, “A comparison of different heterogeneous proximity functions and euclidean distance,” *Studies in health technology and informatics*, vol. 107, no. Pt 2, pp. 1362–1366, 2004.

[14] D. R. Wilson and T. R. Martinez, “Improved heterogeneous distance functions,” *Journal of artificial intelligence research*, vol. 6, no. 1, pp. 1–34, 01 1997.

[15] A. Clauset, M. E. Newman, and C. Moore, “Finding community structure in very large networks,” *Physical review E*, vol. 70, no. 6, p. 066111, 2004.

[16] P. Pons and M. Latapy, “Computing communities in large networks using random walks,” in *Computer and Information Sciences - ISCIS 2005*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 284–293.

[17] K. Wakita and T. Tsurumi, “Finding community structure in mega-scale social networks,” in *Proceedings of the 16th International Conference on World Wide Web*, ser. WWW ’07. New York, NY, USA: Association for Computing Machinery, 2007, pp. 1275–1276.

[18] V. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” *Journal of Statistical Mechanics Theory and Experiment*, vol. 2008, no. 10, p. P10008, 04 2008.

[19] A.-L. Barabási *et al.*, *Network science*, 1st ed. Cambridge university press, 07 2016. [Online]. Available: http://networksciencebook.com/chapter/9#testing

[20] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the em algorithm,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[21] J. MacQueen *et al.*, “Some methods for classification and analysis of multivariate observations,” in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1. Berkeley, California: University of California Press, 1967, pp. 281–297.

[22] S. C. Johnson, “Hierarchical clustering schemes,” *Psychometrika*, vol. 32, no. 3, pp. 241–254, 1967.

[23] SPSS, “The SPSS twostep cluster component,” 2001.

[24] A. Singh, A. Yadav, and A. Rana, “K-means with three different distance metrics,” *International Journal of Computer Applications*, vol. 67, no. 10, 2013.

[25] R. Henriques, F. L. Ferreira, and S. C. Madeira, “BicPAMS: software for biological data analysis with pattern-based biclustering,” *BMC Bioinformatics*, vol. 18, 02 2017. [Online]. Available: https://doi.org/10.1186/s12859-017-1493-3

[26] S. C . Madeira and A. Oliveira, “Biclustering algorithms for biological data analysis: a survey. ieee/acm trans comput biol bioinform 1:24-45,” *IEEE/ACM transactions on computational biology and bioinformatics / IEEE, ACM*, vol. 1, pp. 24–45, 02 2004.

[27] R. Henriques and S. C . Madeira, “Bic2pam: constraint-guided biclustering for biological data analysis with domain knowledge,” *Algorithms for Molecular Biology*, vol. 11, pp. 1–23, 09 2016.

[28] V. Chaoji, M. Hasan, S. Salem, and M. Zaki, “An integrated, generic approach to pattern mining: Data mining template library,” *Data Min. Knowl. Discov.*, vol. 17, pp. 457–495, 12 2008.

[29] R. Agrawal, T. Imieliński, and A. Swami, “Mining association rules between sets of items in large databases,” in *ACM SIGMOD Conference on Management of Data*. New York, NY, USA: Association for Computing Machinery, 1993, p. 207–216. [Online]. Available: https://doi.org/10.1145/170035.170072

[30] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur, “Dynamic itemset counting and implication rules for market basket data,” in *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD ’97. New York, NY, USA: Association for Computing Machinery, 1997, p. 255–264. [Online]. Available: https://doi.org/10.1145/253260.253325

[31] G. Piatetsky-Shapiro, “Discovery, analysis, and presentation of strong rules,” in *Knowledge Discovery in Databases*, 1991, pp. 229–248.

[32] R. Agrawal, R. Srikant *et al.*, “Fast algorithms for mining association rules,” in *Proc. 20th int. conf. very large data bases, VLDB*, vol. 1215. Citeseer, 1994, pp. 487–499.

[33] X. Wu and V. Kumar, *The Top Ten Algorithms in Data Mining*. CRC Press, 2009.

[34] J. Han, J. Pei, and Y. Yin, “Mining frequent patterns without candidate generation,” *ACM sigmod record*, vol. 29, no. 2, pp. 1–12, 2000.

[35] J. Han, J. Pei, Y. Yin, and R. Mao, “Mining frequent patterns without candidate generation: A frequent-pattern tree approach,” *Data mining and knowledge discovery*, vol. 8, no. 1, pp. 53–87, 2004.

[36] G. Grahne and J. Zhu, “High performance mining of maximal frequent itemsets,” in *6th International workshop on high performance data mining*, vol. 16, 2003, p. 34.

[37] R. Srikant and R. Agrawal, “Mining sequential patterns: Generalizations and performance improvements,” in *International conference on extending database technology*. Springer, 1996, pp. 1–17.

[38] P. Fournier-Viger, J. C.-W. Lin, R. U. Kiran, Y. S. Koh, and R. Thomas, “A survey of sequential pattern mining,” *Data Science and Pattern Recognition*, vol. 1, no. 1, pp. 54–77, 2017.

[39] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu, “Mining sequential patterns by pattern-growth: The prefixspan approach,” *IEEE Transactions on knowledge and data engineering*, vol. 16, no. 11, pp. 1424–1440, 2004.

[40] R. Agrawal and R. Srikant, “Mining sequential patterns,” in *Proceedings of the eleventh international conference on data engineering*. IEEE, 1995, pp. 3–14.

[41] M. J. Zaki, “Spade: An efficient algorithm for mining frequent sequences,” *Machine learning*, vol. 42, no. 1, pp. 31–60, 2001.

[42] M. J. Zaki , “Scalable algorithms for association mining,” *IEEE transactions on knowledge and data engineering*, vol. 12, no. 3, pp. 372–390, 2000.

[43] P. Fournier-Viger, A. Gomariz, M. Campos, and R. Thomas, “Fast vertical mining of sequential patterns using co-occurrence information,” in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2014, pp. 40–52.

[44] P. Fournier-Viger, R. Nkambou, and E. M. Nguifo, “A knowledge discovery framework for learning task models from user interactions in intelligent tutoring systems,” in *Mexican international conference on artificial intelligence*. Springer, 2008, pp. 765–778.

[45] Y. Hirate and H. Yamana, “Generalized sequential pattern mining with item intervals.” *J. Comput.*, vol. 1, no. 3, pp. 51–60, 2006.

[46] J. Wang, J. Han, and C. Li, “Frequent closed sequence mining without candidate maintenance,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 8, pp. 1042–1056, 2007.

[47] N. S. Che Khalid, M. Burhanuddin, A. Ahmad, and M. K. Abd Ghani, “Classification techniques in blood donors sector – a survey,” in *e-Proceeding of Software Engineering Postgraduates Workshop*. Malaysia: UTEM, 11 2013, pp. 114–118.

[48] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

[49] J. R. Quilan, “Induction of decision trees,” *Machine Learning*, vol. 1, pp. 81–106, 1986.

[50] J. R. Quinlan, *C4. 5: programs for machine learning*. Elsevier, 1993.

[51] J. Beliën and H. Forcé, “Supply chain management of blood products: A literature review,” *European Journal of Operational Research*, vol. 217, no. 1, pp. 1–16, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221711004516

[52] S. Pai and G. D. Bader, “Patient similarity networks for precision medicine,” *Journal of Molecular Biology*, vol. 430, no. 18, Part A, pp. 2924–2938, 2018.

[53] A. Carreiro, S. C . Madeira, and A. Francisco, “Unravelling communities of als patients using network mining,” in *ACM SIGKDD Workshop on Data Mining in Healtcare*, 08 2013.

[54] S. Harikumar and S. Pv, “K-medoid clustering for heterogeneous datasets,” *Procedia Computer Science*, vol. 70, pp. 226–237, 12 2015.

[55] S. Klenk, J. Dippon, P. Fritz, and G. Heidemann, “Determining patient similarity in medical social networks,” in *MEDEX 2010 Proceedings*, vol. 572, 01 2010, pp. 6–14.

[56] M. Ashoori, S. Mohammadi, and H. Eivary, “Exploring blood donors’ status through clustering: A method to improve the quality of services in blood transfusion centers,” *Journal of Knowledge & Health*, vol. 11, pp. 73–82, 12 2016.

[57] M. Ashoori and Z. Taheri, “Using clustering methods for identifying blood donors behavior,” in *5th Iranian Conference on Electrical and Electronics Engineering (ICEEE)*, 08 2013, pp. 4055–4057.

[58] W. Boonyanusith and P. Jittamai, “Blood donor classification using neural network and decision tree techniques,” in *World Congress on Engineering and Computer Science*, vol. 1, 08 2012, pp. 499–503.

[59] M. Darwiche, M. Feuilloy, G. Bousaleh, and D. Schang, “Prediction of blood transfusion donation,” in *Proceedings of the Fourth IEEE International Conference on Research Challenges in Information Science*, 05 2010, pp. 51–56.

[60] M. M. Mostafa, “Profiling blood donors in egypt: A neural network analysis,” *Expert Systems with Applications*, vol. 36, pp. 5031–5038, 04 2009.

[61] P. Ramachandran *et al.*, “Classifying blood donors using data mining techniques,” *International Journal of Computer Science & Engineering Technology*, vol. 1, pp. 10–13, 02 2011.

[62] T. Santhanam and S. Sundaram, “Application of cart algorithm in blood donors classification,” *Journal of Computer Science*, vol. 6, p. 548, 06 2010.

[63] M. Testik, B. Yuksel-Ozkaya, S. Aksu, and O. Ilhan, “Discovering blood donor arrival patterns using data mining: A method to investigate service quality at blood centers,” *Journal of medical systems*, vol. 36, pp. 579–94, 05 2010.

[64] B. Venkateswarlu and G. V. S. Raju, “Mine blood donors information through improved k-means clustering,” *ArXiv*, vol. abs/1309.2597, 2013.

[65] T. Li, Y. Chen, Xiangwei Mu, and Ming Yang, “An improved fuzzy k-means clustering with k-center initialization,” in *Third International Workshop on Advanced Computational Intelligence*. IEEE, 2010, pp. 157–161.

[66] B. Yi, H. Qiao, F. Yang, and C. Xu, “An improved initialization center algorithm for k-means clustering,” in *2010 International Conference on Computational Intelligence and Software Engineering*. IEEE, 2010, pp. 1–4.

[67] F. Zabihi, M. Ramezan, M. M. Pedram, and A. Memariani, “Mine blood donors information through improved k-means clustering,” *The Journal of Mathematics and Computer Science*, vol. 2, pp. 37–43, 01 2011.

[68] P. G.T., W. E.D., and V. F.M., “K-means initialization methods for improving clustering by simulated annealing,” in *Advances in Artificial Intelligence – IBERAMIA*. Berlin, Heidelberg: Springer, 2008, pp. 133–142.

[69] M. Hahsler and S. Chelluboina, “Visualizing association rules in hierarchical groups,” *42nd Symposium on the Interface: Statistical, Machine Learning, and Visualization Algorithms*, 01 2011.

[70] M. Hahsler, “arulesviz: Interactive visualization of association rules with r,” *R Journal*, vol. 9, pp. 163–175, 12 2017.

[71] SPMF, 2008-2021, last accessed 24 Aug 2021. [Online]. Available: https://www.philippe-fournier-viger.com/spmf/

[72] P. Fournier Viger, A. Gomariz, T. Gueniche, A. Soltani, C.-W. Wu, and V. Tseng, “Spmf: A java open-source pattern mining library,” *Journal of Machine Learning Research*, vol. 15, pp. 3389–3393, 01 2015.

[73] P. Fournier-Viger, J. C.-W. Lin, A. Gomariz, T. Gueniche, A. Soltani, Z. Deng, and H. T. Lam, “The spmf open-source data mining library version 2,” in *Machine Learning and Knowledge Discovery in Databases*, B. Berendt, B. Bringmann, É. Fromont, G. Garriga, P. Miettinen, N. Tatti, and V. Tresp, Eds. Cham: Springer International Publishing, 2016, pp. 36–40.

[74] A. S. Martins, M. Gromicho, S. Pinto, M. Carvalho, and S. C. Madeira, “Learning prognostic models using disease progression patterns: Predicting the need for non-invasive ventilation in amyotrophic lateral sclerosis,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, no. 01, pp. 1–1, may 2021.

[75] NetworkX, 2014-2020, last accessed 28 Oct 2020. [Online]. Available: https://networkx.org/

[76] T. Aynaud, 2009, last accessed 28 Oct 2020. [Online]. Available: https://python-louvain.readthedocs.io/en/latest/

[77] J. Xie, A. Ma, Y. Zhang, B. Liu, S. Cao, C. Wang, J. Xu, C. Zhang, and Q. Ma, “QUBIC2: a novel and robust biclustering algorithm for analyses and interpretation of large-scale

RNA-Seq data," *Bioinformatics*, vol. 36, no. 4, pp. 1143–1149, 09 2019. [Online]. Available: https://doi.org/10.1093/bioinformatics/btz692

A

Appendix A

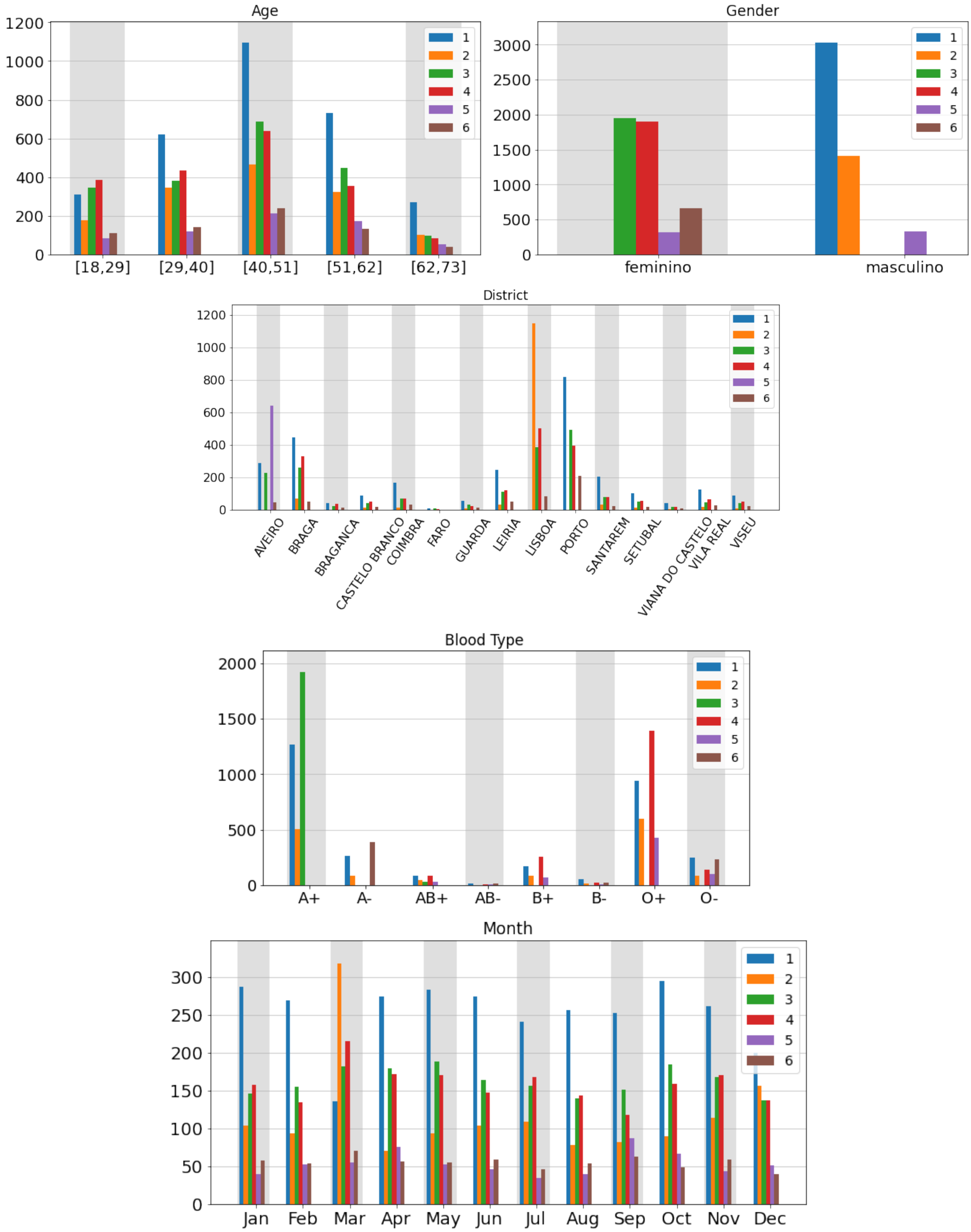


Figure A.1: Plotting of the node distribution in each feature for the different communities discovered in the blood donation network of 2017.

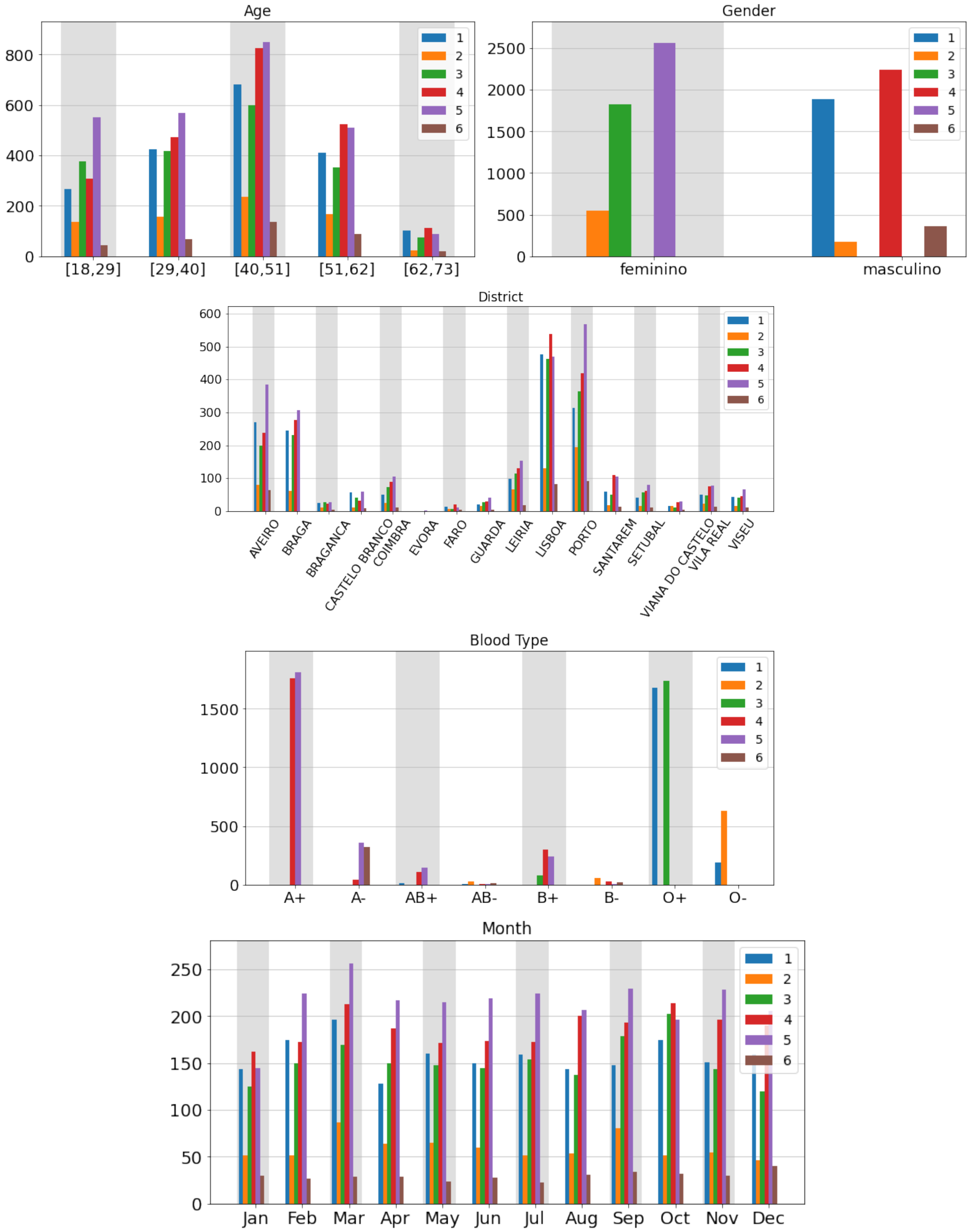


Figure A.2: Plotting of the node distribution in each feature for the different communities discovered in the blood donation network of 2019.