UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola Tècnica Superior d'Enginyeria
de Telecomunicació de Barcelona

Longitudinal patterns of SARS-CoV-2 antibody levels in a sample of Health Care Workers in Barcelona

Degree Thesis submitted to the Faculty of the
Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona
Universitat Politècnica de Catalunya
by

José Ramón Rodríguez Pardo

In partial fulfillment
of the requirements for the degree in
Telecommunications Technologies and Services Engineering

Advisors:
Margarita Cabrera Bean
Concepció Violán Fors
Lucía Amalia Carrasco Ribelles

Barcelona, June 2022

Abstract

Over the past two years, SARS-CoV-2 has spread around the world, affecting people differently. Both quantity and duration of antibody levels can vary widely between individuals. However, the factors that determine which immune response people will generate to COVID-19 are still unclear.

In this thesis, different clustering technologies are explored with the purpose of identifying different patterns of serological response caused by COVID-19 infection. Two unsupervised learning methods are applied, K-means and kmlShape. The latter can be conceived as an improved version of the K-means algorithm, specially designed to work with longitudinal data.

Finally, the clusters obtained with both methods are reported and the characteristics of the patients grouped in each cluster are statistically described. Different relationships have been found between the immune response of each group and individual factors such as age, gender, diseases and drugs. Different immune response groups can also determine COVID-19 reinfections.

Resum

En els dos últims anys, el SARS-CoV-2 s'ha estès per tot el món, afectant a les persones de forma diferent. Tant la quantitat com la durada dels nivells d'anticossos poden variar molt entre els individus. No obstant, els factors que determinen quina resposta immune generaran les persones davant el COVID-19 encara no estan clars.

En aquesta tesi s'exploren diferents tecnologies de clustering amb la finalitat d'identificar els diferents patrons de resposta serològica provocada per la infecció de COVID-19. A més, s'apliquen dos mètodes d'aprenentatge no supervisats, K-means i kmlShape. Aquest darrer es pot concebre com una versió millorada de l'algorisme K-means, dissenyat especialment per treballar amb dades longitudinals.

Finalment, es presenten els clústers obtinguts amb els dos mètodes i es descriuen estadísticament les característiques dels pacients agrupats en cada clúster. S'han trobat diferents relacions entre la resposta immune de cada grup i factors dels individus com l'edat, el gènere, les malalties i els fàrmacs. Els diferents grups de resposta immune també poden determinar les reinfeccions de COVID-19.

Resumen

En los dos últimos años, el SARS-CoV-2 se ha extendido por todo el mundo, afectando a las personas de forma diferente. Tanto la cantidad como la duración de los niveles de anticuerpos pueden variar mucho entre individuos. Sin embargo, los factores que determinan qué respuesta inmune generarán las personas ante el COVID-19 aún no están claros.

En esta tesis se exploran diferentes tecnologías de clustering con el fin de identificar los diferentes patrones de respuesta serológica provocados por la infección de COVID-19. Además, se aplican dos métodos de aprendizaje no supervisado, K-means y kmlShape. Este último puede concebirse como una versión mejorada del algoritmo K-means, especialmente diseñado para trabajar con datos longitudinales.

Finalmente, se presentan los clústeres obtenidos con ambos métodos y se describen estadísticamente las características de los pacientes agrupados en cada clúster. Se han encontrado diferentes relaciones entre la respuesta inmune de cada grupo y factores de los individuos como la edad, el género, las enfermedades y los fármacos. Los diferentes grupos de respuesta inmune también pueden determinar las reinfecciones de COVID-19.

Acknowledgements

I want to dedicate this section to the people that has helped me during the realization of my final degree thesis. Firstly, I would like to thank the Signal Processing and Communications group of the TSC department (UPC) and to the Research Care Group in Primary Care Services (GRENSSAP, 2017/SGR 445) of IDIAPJGol for their collaboration in making this project possible.

Secondly, I would like to thank my advisors Margarita Cabrera Bean from TSC and Lucía Amalia Carrasco Ribelles and Concepció Violán Fors from IDIAPJGol. They have provided me with exceptional guidance and advice during this research project.

Finally, I would like to thank and dedicate this work to my family, who have dedicated all their time, all their effort and all their resources to educate and train me as best as possible to face life.

Contents

List of Figures

List of Tables

Abbreviations

AIC Akaike information criterion

BIC Bayesian information criterion

COVID-19 Coronavirus disease 2019

CVI Cluster validity index

DTW Dynamic Time Warping

EHR Electronic Health Records

EX Exclusivity ratio

GMM Gaussian Mixture Model

HCW Health Care Worker

KNN K-nearest neighbours

O/E Observed/Expected ratio

SARS-CoV-2 Severe Acute Respiratory Syndrome coronavirus 2

SIDIAP Sistema d'Informació pel Desenvolupament de la Investigació en Atenció Primària

VRC Variance Ratio Criterion

WHO World Health Organization

Revision history and approval record

Revision	**Date**	**Purpose**
0	02/05/2022	Document creation
1	19/05/2022	Document revision
2	31/05/2022	Document revision
3	11/06/2022	Document revision
4	15/06/2022	Document revision
5	20/06/2022	Document approval

DOCUMENT DISTRIBUTION LIST

Name	**e-mail**
José Ramón Rodríguez Pardo	jose.ramon.rodriguez.pardo@estudiantat.upc.edu
Margarita Cabrera Bean	marga.cabrera@upc.edu
Concepció Violán Fors	cviolanf.mn.ics@gencat.cat
Lucía Amalia Carrasco Ribelles	lcarrasco@idiapjgol.info

Written by:		Reviewed and approved by:	
Date	02/05/2022	Date	20/06/2022
Name	José Ramón Rodríguez Pardo	Name	Margarita Cabreara Bean
			Concepció Violán Fors
			Lucía Amalia Carrasco Ribelles
Position	Project Author	Position	Project Supervisor

1 Introduction

1.1 Context and justification

Coronavirus disease 2019 (COVID-19) is an emerging disease that has globally affected 470 million people and already caused 6 million deaths. The first case of this predominantly respiratory viral disease was first reported in Wuhan, Hubei Province, China, in late December 2019, then Severe Acute Respiratory Syndrome coronavirus 2 infection (SARS-CoV-2) rapidly disseminated across the world in a short time. World Health Organization (WHO) declared it as a global pandemic on March 11, 2020 [1]. Since being declared a global pandemic, COVID-19 has overwhelmed most health systems and devastated many countries around the world in the context of new variants of SARS-CoV-2. [2, 3].

Nowadays, it is possible to make use of technologies to better manage the pandemic [4]. One of the most cutting-edge technologies in today's world is machine learning, based on algorithms used to analyse and draw inferences from patterns in data. So much so, that it has countless applications in predictive medicine with the fight against COVID-19 being one of the most crucial [5]. In this context, clustering is a machine learning technique widely used today, which makes it possible to group sets in which the objects in the same group are similar to each other, but different from those in the other groups.

SARS-CoV-2 is a virus that possesses 4 structural proteins whose structure can be seen in Appendix A. When a person has the COVID-19 their immune system acts against all the SARS-CoV-2 proteins by generating antibodies. This project arises from the need to identify different types of evolution of antibody levels, and describe the characteristics of the individuals having each type of evolution. Such knowledge would allow recommendations to be made, under the hypothesis that a reduction in the level of antibody levels may increase the risk of reinfection. Therefore, research into different clustering methods that group patients according to their immune responses is necessary.

This thesis is born from the ProHEpiC-19 project, which was intended to describe the kinetics of IgM (N) and IgG (N, S) antibodies against SARS-CoV-2 and to assess the relationship between the immune response and the COVID-19 severity [6]. This was done from an analysis with statistical methods on 17 months of serological responses from more than 800 healthcare workers (HCWs) from Barcelona. Instead of the time period analysed in [6], in our project, a 21-month analysis only of the evolution of IgG (N) antibody levels after infection will be performed using a database containing the results of serological tests performed on these HCWs. Furthermore, in our work unsupervised machine learning techniques that work especially with longitudinal data are introduced, providing a potentially better analysis of the immune response of infected individuals than conventional methods, which do not take into account the time variable.

1.2 Objectives

The project aims to study longitudinal clustering methods to obtain groups identifying distinct serological response patterns. This study involves both transversal and longitudinal clustering techniques. Moreover, a description of the individuals included in each cluster using descriptive statistics is necessary to understand which type of patients is gathered in each group. The project main goals are:

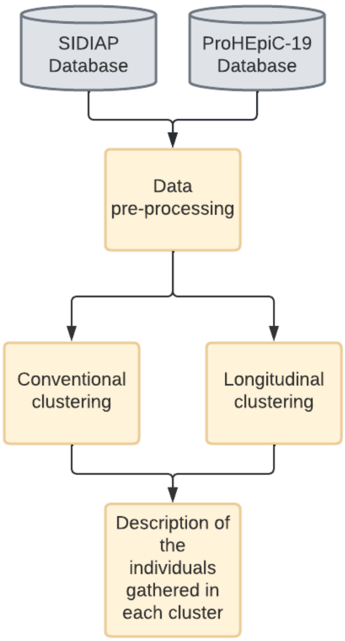
- Analyze the evolution of SARS-CoV-2 antibody levels in individuals after infection.
- Obtain the most suitable method to cluster patients according to their immune responses, from the comparison of conventional and longitudinal clustering methods including validity indices.
- Perform a description of the type of patients gathered in each cluster in order to understand the individual factors that trigger the immune response of each group.

All these objectives will be performed with data from real Electronic Health Records (EHR), including demographic information, diseases, drugs, and serological tests results from Barcelona HCWs.

1.3 Methods and procedures

The methods and procedures followed in this project, which will be detailed later on in the Methodology section, are succinctly depicted in Figure 1.1:

Figure 1.1: Project methodology



The complete work plan, including the internal tasks of each work package as well as the Gantt chart can be found in Appendix B.

1.4 Document structure

This thesis memory is made up of 6 different sections. The first chapter consists of an introduction to the thesis, in which the justification for carrying out the project and its objectives are explained. State-of-the-art in chapter 2 includes relevant and recent research that has been done on the subject matter as well as the technologies used in these kinds of projects. Chapter 3 consists of the methodology of the work, an explanation of everything that has been done and all the relevant methods that were used. Subsequently, chapter 4 presents the results obtained and the decisions that have been taken. The budget Chapter shows the estimation of the cost of the work dedicated to the thesis. Finally, the conclusions of the thesis are stated and future points of interest for further research in this field are described in chapter 6.

2 State of the art of the technology used or applied in this thesis

This section focuses on presenting the most relevant research that has been carried out on our subject matter.Subsection 2.1 mentions several techniques and methods used to cluster longitudinal data. Subsequently, section 2.2 presents different studies applied to COVID-19 which use this type of techniques to work with longitudinal data.

2.1 Technologies available for clustering data

Longitudinal studies can be used to measure the evolution of certain phenomena in fields such as health science, biology, economics, sociology or marketing. In contrast to cross sectional, in which subjects are observed at a single moment, longitudinal studies employ continuous or repeated measures to follow particular individuals over prolonged periods of time [7]. Temporal variations in an outcome of interest can be directly observed and studied by following subjects over time. One way to find different trajectory patterns in the data is through clustering, where individuals are separated into homogeneous groups. Although there are several ways to cluster longitudinal data, the methods can be mainly classified into two approaches: non-parametric and model-based methods.

Non-parametric clustering

Non-parametric methods operate by making no assumptions on how the data was generated. Therefore, non-parametric methods aim to define the similarity between subjects and clusters without making assumption on the data. To this end, the three fundamental points to these methods are the clustering algorithm, the similarity or dissimilarity measure and the number of clusters [8]. Within the non-parametric, most clustering algorithms can generally be grouped as either partitioning or hierarchical.

Partitional clustering aim to split n observations into $k >= 2$ distinct clusters, without an object being able to belong to two or more different clusters. The K-means algorithm is probably the partition algorithm most widely used, and it can use different similarity measures to calculate how similar two observations are. Currently, the R package KmL allows working with the K-means algorithm specially adapted for longitudinal data [9]. This, makes use of the most traditional distances such as Euclidean and Manhattan distances, the latter being more robust to outliers. However, these two distances do not work well when the time series are shifted or delayed as they compare point to point. For this, the KmL package also makes available Dynamic Time Warping (DTW) [10], a metric that allows to calculate the similarity considering a time lag of the trajectories described in section 3.2.1.
Another approach to longitudinal partitional algorithms is to provide clusters on the basis of trajectory shape, handling distortions in amplitude and phase. For this purpose, the two most commonly used algorithms are kmlShape and k-Shape. The kmlShape is an extension of KmL which uses the Fréchet distance as the similarity measure. This distance treats each trajectory as a curve in order to identify clusters based on the shape [11]. On the other hand, k-Shape is based in a procedure similar to the one used by K-means but

using cross-correlation as the measure of similarity [12]. Nevertheless, cross-correlation is not often adopted as a time series distance because inefficient implementation of this measure can result in a very slow process.

Hierarchical clustering is mainly based on creating cluster trees where the root node of the tree is a group containing all elements and the leaves are groups each of which contains one different object of the total. Unlike partitioning algorithms, these algorithms do not need to know the number of clusters beforehand, however, the problem they suffer comes when choosing where to cut the tree to find the best partition [13]. Hierarchical algorithms can be categorized as divisive (top-down) or agglomerative (bottom-up). The former starts with all objects belonging to a single cluster and then are split up until finally reaching a cluster per object. In contrast, agglomerative methods start by treating each observation as a cluster of its own and then iteratively agglomerates pairs of clusters until reaching one containing all elements [8]. To determine how close two groups are, popular distances can be used as Euclidean distance or DTW. Nevertheless, in this type of clustering distance measures are not mandatory, being possible to use other clustering methods normally density-based or graph-based, as a subroutine for constructing the hierarchy [14]. Compared to partitional methods, hierarchical ones are less commonly used for longitudinal data mainly due to their computational complexity [13].

Model-based clustering

Unlike non-parametric methods, model-based approaches assume that the data are generated from a finite mixture of distributions [15], each representing a different cluster. The parameters of each distribution are obtained by maximizing likelihood [13]. Among the model-based methods, the best known and most widely used is the Gaussian Mixture Model (GMM) explained mathematically in [16]. This method is mainly based on determining the different clusters where each of them is represented by a Gaussian distribution. In addition, the algorithm determines a mean and a variance for each cluster and a probability of belonging or not to that cluster for each observation.

Clustering validity

Once the partition is generated, it is important to evaluate the goodness-of-fit of the algorithm. For this purpose, clustering validity is used, which can be classified as internal or external depending on whether or not they use external information [17]. As clustering is an unsupervised method, there is usually no ground-truth that can be used as an external validity of the goodness of the algorithm. In this case, internal validity mainly consists of solving the problem of finding the ideal number of clusters for our data set. Although this problem currently has no optimal solution [18], several internal validity indices have been proposed, which can help to decide the optimum number after running the algorithm several times with different number of clusters and finally comparing the results. The most commonly used indices in non-parametric clustering are Silhouette [19], Davies-Bouldin [20] and Calinski-Harabasz [21]. On the other hand, for evaluating how well a model fits the data it was generated from and find the optimal number of clusters in model-based clustering, Bayesian information criterion (BIC) is often used which is closely related to Akaike information criterion (AIC), both studied in [22].

On the other hand, when there is a ground-truth, it is possible to validate which objects are well clustered and which are not. External validity indices are based on this knowledge to find an efficient method, where the Rand index [23] and the Jaccard coefficient [24] are the most well-known. In addition to the above-mentioned indices, a large list of internal and external indices applied to longitudinal data can be found in [13]. However, despite the numerous existing techniques in cluster validity, there is currently no unanimously accepted method.

2.2 Related work

Out of all the studies carried out with some relation to this thesis, there is a wide range of publications that group countries according to the evolution of various COVID-19 indicators. An example of country clustering was studied in [25], where 206 countries were grouped according to incidence and mortality rates. In this study, K-means with Euclidean distance was the method used to cluster the countries according to the evolution of these two rates. In addition, the elbow method was used to find the optimal number of clusters.

The evolution of different protein levels can be determined with the clinical course of COVID-19, which can be mild, moderate or severe. A supervised clustering was done in [26], where K-means with DTW was used to group COVID-19 individuals into three different clusters based on the longitudinal measurement of C-reactive protein levels, absolute neutrophil counts and absolute lymphocyte counts. The results showed distinct reactivity intensities and patterns for each cluster.

Hierarchical clustering also has applications in the field of COVID-19. This method was used in [27], where individuals were hierarchically clustered according to their antibody reactivity levels to different antigens. Before clustering, the optimal number of clusters was extracted using gap statistics validity [28].

Finally, in the following thesis [13] several longitudinal clustering techniques, both non-parametric and model-based, were studied and applied to real data in order to cluster countries according to 6 mobility trends (changes in the number of visits to different places such as parks or pharmacies) during the lockdown period. In addition, ten internal and ten external validity indices were implemented on artificial data with the purpose of finding the index that more often suggested the correct number of clusters. With this objective, it was observed that the index that provided the best results was the Calinski-Harabasz. On the other hand, the non-parametric clustering methods were the ones that obtained the best results compared to the model-based ones, also resulting in less complexity.

3 Methodology / project development:

3.1 Data pre-processing

Database

The data used in this project are extracted from two different databases. ProHEpiC-19 database contains serological test results from 5 May 2020 to 11 February 2022 on HCWs in Barcelona. Moreover, information on patients' demographics, symptoms as well as if they have been infected, reinfected and when they have been vaccinated is provided in this database. On the other hand, Sistema d'Informació pel Desenvolupament de la Investigació en Atenció Primària (SIDIAP) database contains information on the symptoms and drugs of these HCWs.

The serology table extracted from ProHEpiC-19 database collects the results of IgM(N) and IgG (N, S) antibody levels. However, in this study, only IgG (N) antibody levels are used. The main reason for this is that IgG (S) values can be altered by the effect of the vaccine, making it impossible to differentiate whether a positive value is due to an infection or a response to vaccination. In addition, as the study progressed, it became clear that the IgM (N) antibody did not continue evolving over time, so IgM (N) levels were no longer measured. As IgG (N) antibodies levels are measured repeatedly over a period of time, we can measure their evolution.

During the start of the project, the database was checked to ensure that the data were entirely correct. As a result, some duplicate items were removed and some infection dates were updated.

As the aim of the project is to group patients according to their immune responses generated by the SARS-CoV-2, the results reported in this work will be obtained only with infected patients, taking into account only the samples collected after infection but without considering reinfection samples, i.e., serological tests performed after reinfection.

Descriptive analysis

A previous descriptive analysis considering all patients was performed for different categorical and continuous variables, where descriptive statistics were applied in order to find out the principal characteristics of the population involved in the study. Numerical variables were represented with their mean and standard deviation while the categorical ones were represented with absolute frequencies and percentages. Missing values were found in the following variables: civil status (4.27%), origin (7.11%), job (15.9%), total diseases (0.24%), and total drugs (2.61%). This analysis was done with the R package compareGroups meant to facilitate the construction of bivariate tables [29]. The different programming languages and packages used for each section and sub-section are shown in Appendix C.

Data temporal dispersion

As part of the data exploration process, first step was to investigate whether the data had temporal dispersion in order to find out if there was much variation between measurement times. This step was necessary especially for the conventional methods since, as discussed later in 3.2.1, it was necessary to discretise the time scale into different intervals. In this way, even if these methods did not work taking into account the exact measurement day, each sample of antibody levels would represent a period of time identical to that of the other patients, avoiding direct comparison between two samples far apart in time.

Once COVID-19 infection was confirmed, the protocol was to perform serological tests 15, 30, 60, 90, 180, 270, 360, and 450 days after the baseline visit. However, in practice they were not done on these exact days. Therefore, for each patient, the difference in days between infection and the different tests was calculated and studied to see if the samples were measured on the days stated by the protocol or if there was a temporal dispersion in the data. Besides the possible dispersion, it was necessary to consider the number of patient samples as some of them dropped out of the study or simply did not show up on a given day, resulting in a lack of follow-up samples.

In the hypothetical case that all participants had the same number of samples and no temporal dispersion i.e., the same number of tests conducted around the same days, no prior transformation of the data would be necessary before going through clustering. However, since it was known that not all patients had been fully followed up, it was opted to perform different analyses before the clustering process.

Data selection and preparation

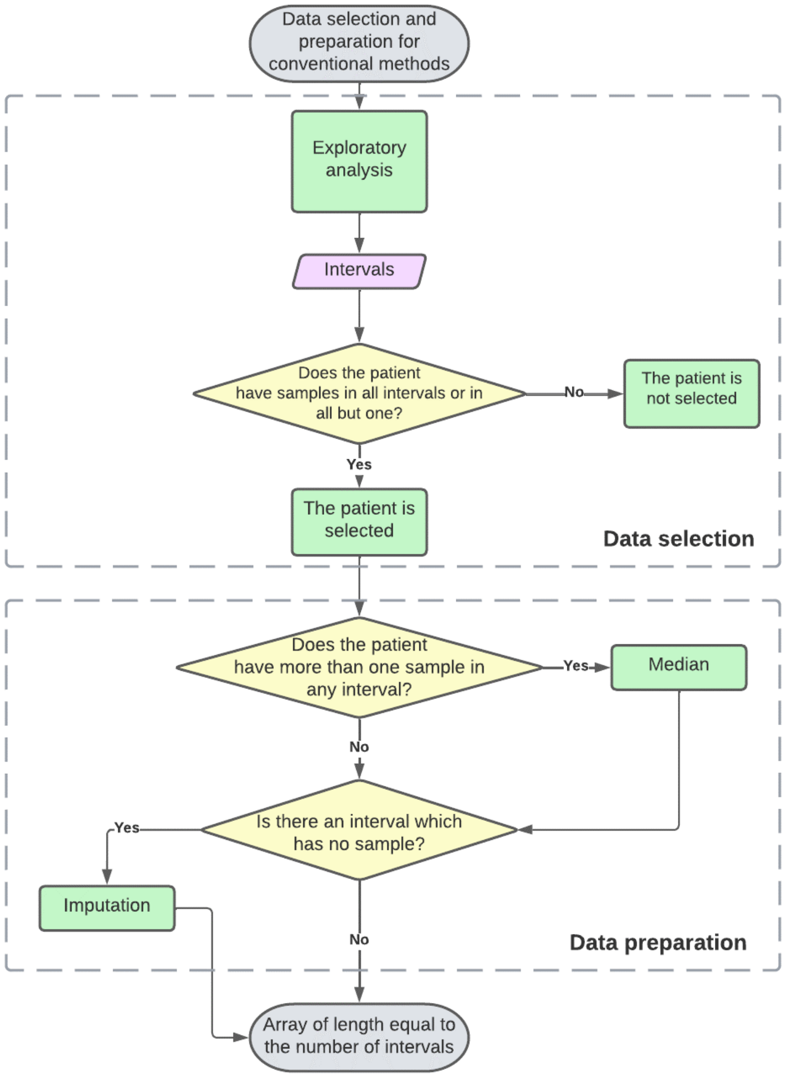
Once the existence of dispersion was verified, it was necessary to select infected patients who were similar in terms of the days on which their different samples were measured. To do this, different analyses were performed in order to select patients with sufficient and correctly distributed follow-up infection samples to then prepare these data according to the input requirements of the model. These analyses were approached differently depending on the type of clustering used. In this thesis, clustering methods were separated into two main groups: conventional methods, which are described in 3.2.1, and longitudinal methods described in 3.2.2.

Conventional methods

The requirement to be met in these methods is that the model input data, in this case, one array per individual, must have the same length. Each array is made up of the same number of samples and represents the evolution of the antibody levels through fixed time periods instead of using the exact measurement day as additional input to the model. In order to select those patients who were useful for the study by having correctly distributed data along the time axis, avoiding those with a high number of missing samples or with many samples on closely spaced days, the methodology presented in Figure 3.1 was followed.

Firstly, the time axis was divided into different periods e.g., {[0-15], [16-30], [31-65] . . . }. In order to find the optimal number of intervals to work with as well as their ranges, an exploratory analysis was conducted. These analyses consisted of testing different sets of periods to see which set grouped the most individuals which had at least one sample in all intervals or in all but one. The condition of having samples in all but one was imposed in order to gain sample size but without having a high percentage of imputations, as would result if patients with more than one interval with a missing value were taken. Once the optimal intervals were found, those patients who met the above requirement were selected. Secondly in the data preparation step, for each of the selected individuals, the data were modified in order to achieve the model input requirement of having data of equal length. If the patient had more than one sample in the same interval, the median of these samples was calculated. Subsequently, if the condition of having an interval without a sample was met, it was imputed. To fill in the missing value, a k-nearest neighbours (KNN) model was used, which predicted the missing value with reference to the mean of the 5-neighbour samples.

Figure 3.1: Data selection and preparation for conventional methods


Data selection and preparation for conventional methods
Exploratory analysis
Intervals
Does the patient have samples in all intervals or in all but one?
No
The patient is not selected
Yes
The patient is selected
Data selection
Does the patient have more than one sample in any interval?
Yes
Median
No
Is there an interval which has no sample?
Yes
Imputation
No
Data preparation
Array of length equal to the number of intervals

Once this was done, it was possible to have the data adapted to the input requirements of the model, having an array for each patient of length equal to the number of intervals. To better understand the process, a simple example is shown below:

Example data selection and preparation for conventional methods

Let us consider two different patients with serological test results on the following days:

p1: t1= [6,32,57,80] IgG1= [0.76, 1.81, 2.06, 1.04]
p2: t2= [2,34,93,120] IgG2= [2.71, 3.08, 3.14, 3.01]

In this context, this means that $p1$ had a test to measure antibody levels 6 days after infection resulting in a total value of 0.76, the next on day 32 with a value of 1.81 and so on. Let us also consider the following set of time periods drawn from the exploratory analysis:

Time intervals: [0-15],[16,30],[31-60],[61-90]

For this example, only patient $p1$ would be selected as it meets the requirement of having samples in all intervals or in all but one. In the preparation step for this patient, the median of antibody levels would be considered between the samples 57, 80 as they share the same interval. In addition, the missing sample from the interval [16 - 30] would be imputed.
The model input data will therefore be only the array $IgG1$ with length equal to the number of intervals, containing the evolution of IgG (N) antibody levels for $p1$.

Both the data review, the preparation process and the clustering process for conventional methods were done with the Python programming language.

Longitudinal methods

For this type of methods, the model input data is somewhat different than for conventional ones. In this case, as time between samples is considered by the model, for each patient, apart from the array containing the immune response, there is also an array indicating the time index of the sample, i.e., on which day the test was performed since the day of infection.

For the method used in this thesis explained in 3.2.2 and which is designed to work especially with longitudinal data, no prior preparation of the data was necessary since the model allowed to obtain results with inputs of different lengths. Nevertheless, a selection of patients was made, considering only those who had a minimum of 5 samples. A posteriori, in view of the first results, it was decided to add further conditions, which consisted of considering only one-year of follow-up and removing those patients who did not have their first visit before day 30 of infection and their last visit after day 270.

3.2 Clustering

Clustering is the task of grouping a set of observations in such a way that observations in the same group (called a cluster) are more similar in some sense to each other than to those in other groups. In this thesis, two types of clustering were carried out to group patients according to the evolution of their IgG (N) antibodies after infection. The first results were extracted with conventional methods and then clustering was performed with an algorithm applied directly to longitudinal data.

3.2.1 Conventional clustering

Conventional clustering methods, as expressed in the previous point, do not consider uniformly the time variable, or putting it further, these methods assume that the time samples have been taken at the same time days for all individuals. By having selected those individuals with similarly distributed data, each sample of the trajectory represents that it is taken in a time interval which is the same as for the other trajectories. This, avoids comparing antibody levels on days that are widely separated. To obtain the first results, it was decided to work with the K-means algorithm due to its simplicity and extensive use but also to compare them a posteriori with those clusters obtained with a variant designed to work especially with longitudinal data.

The K-means algorithm is the best-known and most widely used unsupervised clustering technique. It is a partitional method that aims to separate a set of n observations into K groups, where the value of K is fixed in advance and represents the number of centroids to be found. The algorithm starts by randomly selecting K data points to serve as the initial centers for the cluster, also known as centroids. To reach optimal partitioning, the algorithm iterates over two steps. In the assignment step, every point is put into the cluster of the nearest centroid. In this context, the 'nearest' is defined by a distance measure. In the update step, the centroid of every cluster is recalculated as the mean of all data points assigned to the cluster [30]. These two steps are repeated until a convergence criterion is met. The final result does not necessarily have to be the best clustering as it is highly dependent on the initialization. However, for each different initialization, it is ensured that the results converge and that the optimal partitioning is obtained.

The basic idea of the algorithm is to define clusters so that the within cluster variation is minimized. For the formulation of the K-means algorithm, let us consider a data set with n observations $X = \{x_1, \cdots, x_n\}$ to be clustered into a set of K groups $C = \{C_k, k = 1, \cdots, K\}$ with μ_k being the centroid of the cluster C_k. In our work each observation x_i represents the antibody levels trajectory of a certain individual produced after infection. The squared error between the centroid μ_k and the observations in cluster C_k is defined as [31]:

$$J(C_k) = \sum_{x_i \in C_k} ||x_i - \mu_k||^2 \tag{3.1}$$

The objective of the K-means algorithm is to minimize the sum of the squared error over all K clusters, which is calculated as:

$$J = \sum_{k=1}^{K} J(C_k) = \sum_{k=1}^{K} \sum_{x_i \in C_k} ||x_i - \mu_k||^2 \tag{3.2}$$

In this project, the K-means algorithm was applied with two different distances. Firstly, with the Euclidean distance and subsequently with DTW. The latter was done as an intermediate step before moving on to methods designed to especially cluster time series.

3.2.1.1 K-means with Euclidean distance

The first results were obtained by using the most traditional metric when applying the K-means algorithm, called Euclidean distance. The Euclidean distance between two trajectories is given by:

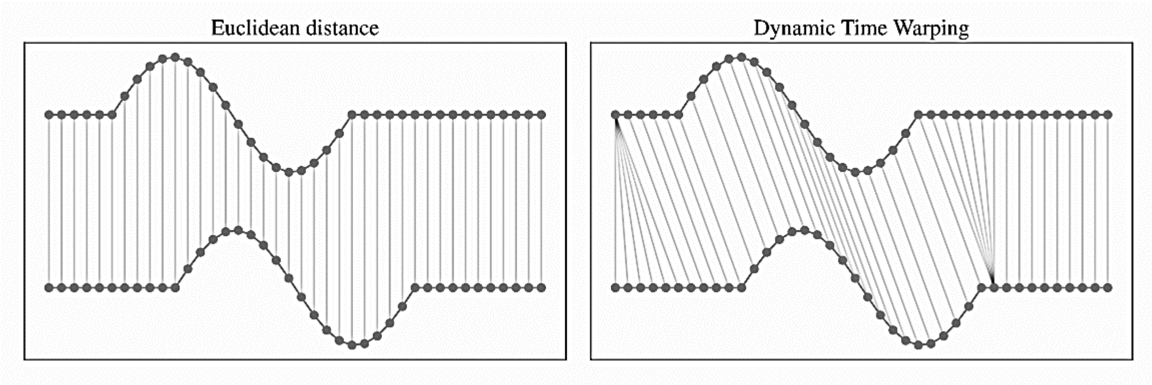
$$d(x_i, x_j) = ||x_i - x_j|| = \sqrt{\sum_{t=1}^{T}(x_i(t) - x_j(t)} \tag{3.3}$$

Where in our thesis, $x_i(t)$ represents the antibody levels of the individual x_i at time t. For conventional methods as explained before, t represents an instant within a time period. K-means with Euclidean distance was performed with the Scikit-learn KMeans package of Python [32].

3.2.1.2 K-means with DTW

Dynamic Time Warping is a popular distance measure for time series, able to manage time distortions by realigning time series when comparing them. As it is shown in Figure 3.2 while two series may have a similar shape, they might not be aligned on the time axis. So, DTW is equivalent to minimizing the Euclidean distance between aligned time series [10].

Figure 3.2: Comparison between DTW and Euclidean distance [10]



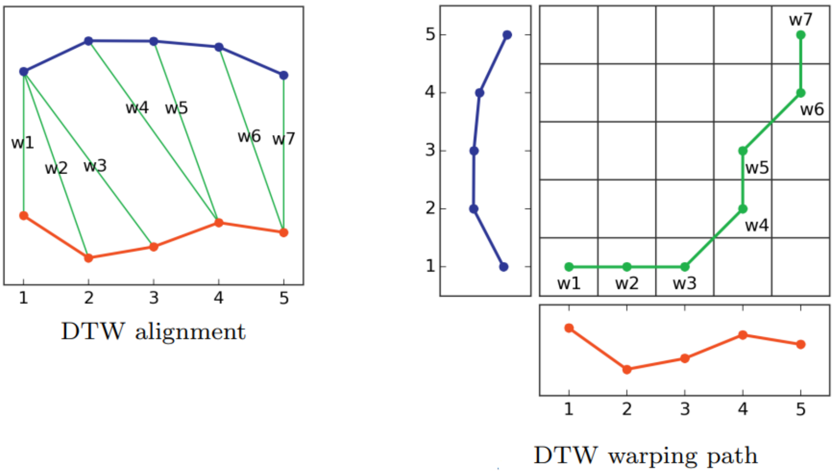
The goal of DTW is to find the optimal alignment between two-time series that achieves minimum global cost. The global cost is defined as the summation of the cost between each pair of points in the alignment. So, the cost between each pair of points can be expressed as [33]:

$$\delta(a_r, b_c) = (a_r - b_c)^2 \tag{3.4}$$

DTW distance, i.e., the optimal global cost of DTW can be calculated as follows:

$$D(A_r, B_c) = \delta(a_r, b_c) + min\{ \begin{matrix} D(A_r, B_{c-1}) \\ D(A_{r-1}, B_c) \\ D(A_{r-1}, B_{c-1}) \end{matrix} \tag{3.5}$$

Figure 3.3: DTW alignment and DTW warping path [33]



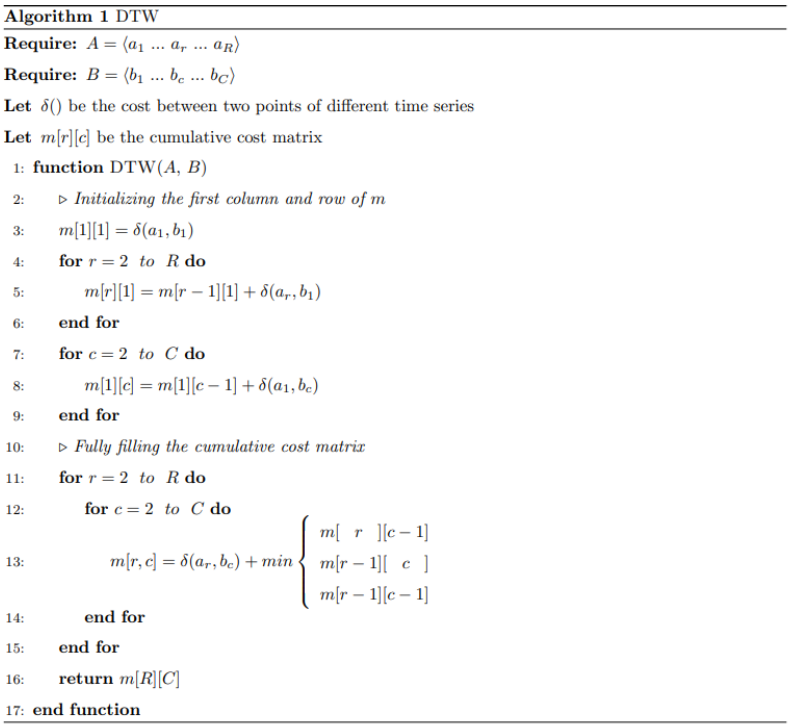
DTW alignment

DTW warping path

As it can be seen this leads to a dynamic programming problem. Therefore, the process of DTW consists of fully filling a cumulative cost matrix as shown in Figure 3.3, of dimensions RxC (dimensions of the two-time series), which in our case as required in tslearn package used in this thesis $R = C$ [34]. The green line indicates the alignment of the time series chosen from the minimum values of the cumulative cost matrix.

To fill the matrix, the steps presented in the pseudocode of Figure 3.4 are followed. The last matrix cell m[R][C] holds the final DTW distance between the time series.

Figure 3.4: Pseudocode of DTW algorithm [33]



```
Algorithm 1 DTW
Require: $A = \langle a_1 \dots a_r \dots a_R \rangle$
Require: $B = \langle b_1 \dots b_c \dots b_C \rangle$
Let $\delta()$ be the cost between two points of different time series
Let $m[r][c]$ be the cumulative cost matrix
function DTW($A$, $B$)
    ▷ Initializing the first column and row of $m$
    $m[1][1] = \delta(a_1, b_1)$
    for $r = 2$ to $R$ do
        $m[r][1] = m[r-1][1] + \delta(a_r, b_1)$
    end for
    for $c = 2$ to $C$ do
        $m[1][c] = m[1][c-1] + \delta(a_1, b_c)$
    end for
    ▷ Fully filling the cumulative cost matrix
    for $r = 2$ to $R$ do
        for $c = 2$ to $C$ do
            $m[r, c] = \delta(a_r, b_c) + min \begin{cases} m[\ r\ ][c-1] \\ m[r-1][\ c\ ] \\ m[r-1][c-1] \end{cases}$
        end for
    end for
    return $m[R][C]$
end function
```

3.2.2 Longitudinal clustering

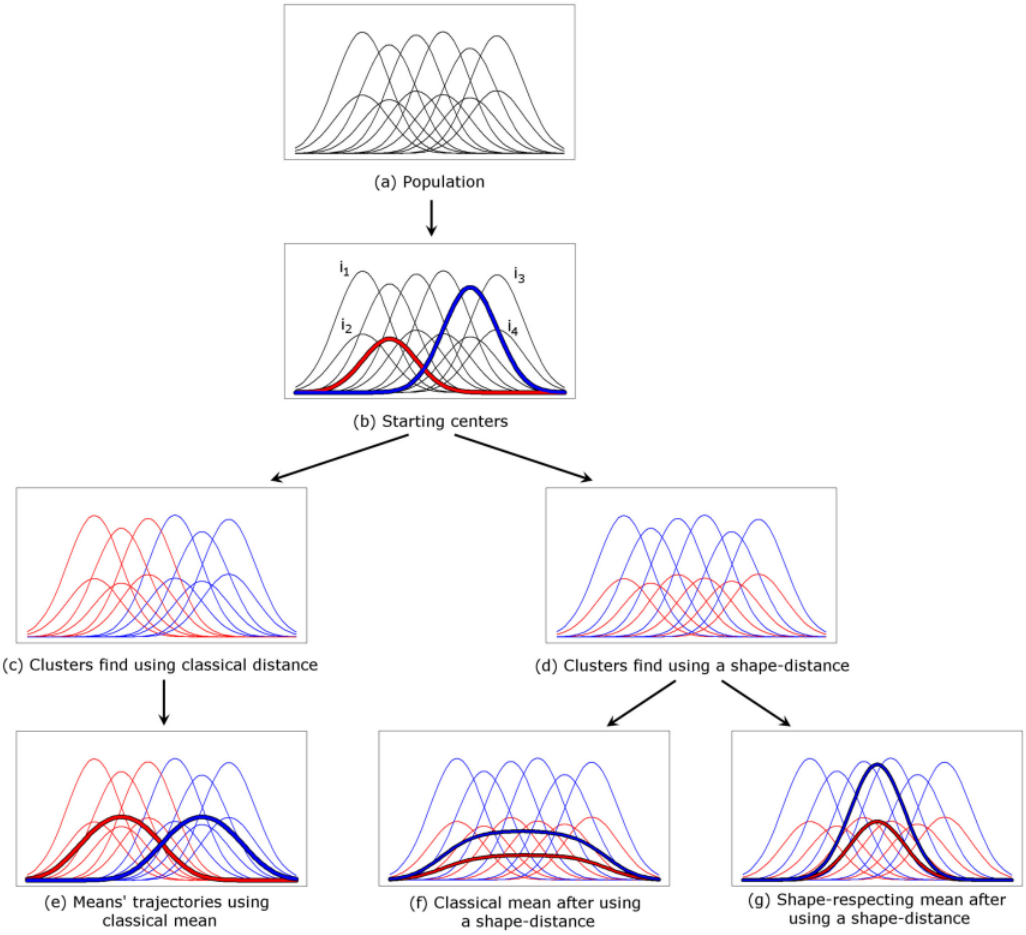
Since we were working with longitudinal data, once the results were extracted with the conventional methods explained above, it was decided to use a method that could work specifically with this type of data, taking into account the exact time point of each sample. For this, a variant of K-means algorithm called kmlShape was used.

3.2.2.1 kmlShape

kmlShape is a clustering algorithm that clusters trajectories according to their shape. It applies K-means within the context of a shape-respecting partitioning [11]. As briefly reminded in the introduction of the clustering section, K-means uses two tools: a distance and a mean. KmlShape is a variant which uses: the Fréchet distance to calculate the distances between individuals and cluster centers; and Fréchet mean to construct the centers of the clusters.

As shown in Figure 3.5c using the Euclidean distance does not allow solving the similar-shape clustering problem. However, using the Fréchet distance which is a shape-respecting distance leads to the partition presented in Figure 3.5d, giving a correct grouping. From here, using a conventional way to compute the mean leads to non-representative centroids as shown in Figure 3.5f. That is the reason kmlShape also uses a shape-respecting mean leading to the clustering shown in Figure 3.5g. The peak of the centroid represents the group well in terms of amplitude but less so in terms of time axis. Nevertheless, in our case, as individuals have their antibody levels peak around the same day, this effect will not be seen to a large extent.

Figure 3.5: The impact of using the Euclidean distance, the Euclidean mean, the Fréchet distance and the Fréchet mean [11]

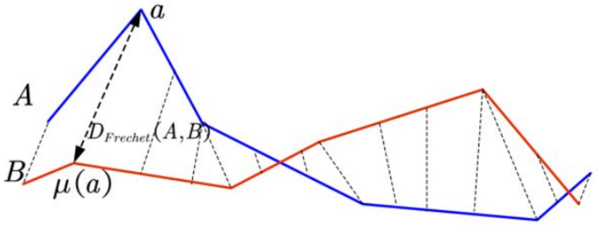

(a) Population
(b) Starting centers
i_1
i_2
i_3
i_4
(c) Clusters find using classical distance
(d) Clusters find using a shape-distance
(e) Means' trajectories using classical mean
(f) Classical mean after using a shape-distance
(g) Shape-respecting mean after using a shape-distance

Fréchet distance and Fréchet mean

Fréchet distance is often represented intuitively by an example of a person traversing a finite curved path while walking their dog on a leash, with the dog traversing a separate finite curved path. They both can vary their speed, but neither of them can move backwards. The Fréchet distance between the two curves is the length of the shortest leash sufficient for both to traverse their separate paths from start to finish. Moreover, unlike classical distances, the calculation of Fréchet distance does not require trajectories of the same length.

Fréchet distance is visually represented in Figure 3.6.

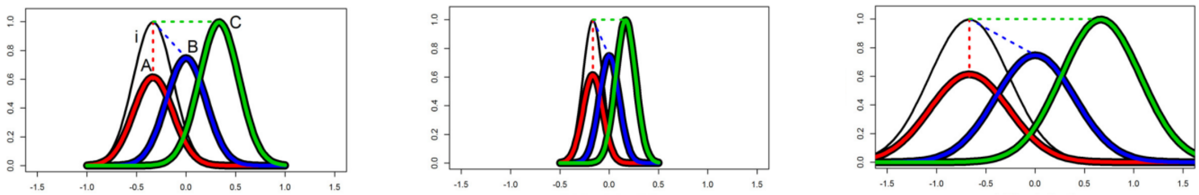
Figure 3.6: Fréchet distance measurement [35]



The Fréchet mean between the two trajectories is the middle of the leash that links the dog to the person. As formal definition is dense, mathematical formulation of both tools can be found in [11].

For clustering our data, the kmlShape package available in R was used. Fréchet's distance parameter called timeScale allows to modify the time scale, increasing or decreasing the cost of the horizontal shift. If timeScale is very big as represented in Figure 3.7c, then the Fréchet's distance is more similar to the Euclidean distance, taking into account when the peak occurs rather than its amplitude. On the other hand, if timeScale is very small as represented in Figure 3.7b, then it is more similar to the Dynamic Time Warping distance, making the model in this case more invariant to time shifts, and clustering the green curve with the black one in the same group as the distance between them is smaller. For this reason, an exploratory analysis was carried out, from which the value of the parameter that produced the best results was extracted.

Figure 3.7: Effect of timeScale parameter (Figure adapted from [11])



(a) Initial population

(b) Small timeScale

(c) Big timeScale

3.2.3 Validity indices

When working with this type of clustering algorithms, one problem is to specify the number of clusters beforehand, but the correct choice of K is often ambiguous. Cluster validity indices (CVIs) can be used to identify the optimal number of clusters by evaluating the degree of similarity or dissimilarity between the data (internal validity indices). In addition, CVIs can also be applied to compare the true partition with the one obtained from the clustering as well as for comparing two different partition results (external validity indices). When applying clustering methods, both types were used with different finalities.

3.2.3.1 Internal validity indices

As K-means is an unsupervised method and therefore the prior partition is not known, two internal validity indices were used in order to find the number of clusters that gave the best grouping for our data. Since the validity indices are more designed to work with classical distances, for the conventional methods explained previously in 3.2.1 two different indices were applied. On the other hand, for the longitudinal clustering method described in 3.2.2 only one of them was applied with a small modification.

Validity indices for K-means

Silhouette

Silhouette is an internal validity index used when the ground truth labels are not known, as a measure of how similar an observation is to its own cluster (cohesion) compared to other clusters (separation). Silhouette value ranges from -1 to +1, indicating that the observation is 'well-clustered' when a value close to 1 is reached and on the contrary, indicating that an object is 'misclassified' when closer to -1 [19].

Silhouette is calculated for each observation i as follows:

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}} \tag{3.6}$$

Where:

- Cohesion $a(i)$ is the mean distance between object i to all other objects in the same cluster, and represents how well object i is assigned to its cluster
- Separation $b(i)$ is the mean distance of object i to all objects in the nearest cluster, of which object i is not a member

So, when $s(i)$ is close to 1, implies that $a(i) << b(i)$ and as $a(i)$ measures the dissimilarity between i to its own cluster, it presents a small value meaning that the observation is well clustered.

Total silhouette score can be defined as the mean s(i) overall data observations of the dataset:

$$S = \frac{1}{N}\sum_{i=1}^{N} s(i) \tag{3.7}$$

Calinski-Harabasz

Calinski-Harabasz index is another internal validity index, which was also calculated to evaluate the model and find the K that best fits our dataset. It is sometimes called the variance ratio criterion (VRC), and is defined as [21]:

$$VRC = \frac{BGSS}{WGSS} \cdot \frac{N-K}{K-1} \tag{3.8}$$

Where BGSS is the overall between-group sum of squares, WGSS is the within-cluster sum of squares, K is the number of clusters and N is the number of observations.

BGSS used to evaluate intercluster distance is defined as:

$$BGSS = \sum_{k=1}^{K} n_k ||m_k - m||^2 \tag{3.9}$$

Where n_k is the number of elements in the cluster k , m_k is the centroid of cluster k and m is the mean of the dataset.

WGSS is defined as:

$$WGSS = \sum_{k=1}^{K} \sum_{x \in C_k} ||x - m_k||^2 \tag{3.10}$$

Where K is the total number of clusters, x is an observation and m_k is the centroid of the cluster k.

To determine the optimal number of clusters, VRC has to be maximized with respect to K. The larger the BGSS and the smaller the WGSS, the better data partitioning.

Validity indices for kmlShape

As was done for the conventional methods, validity indices were also applied. Calinski-Harabasz criterion is best suited for k-means clustering solutions with squared Euclidean distances so it was decided not to apply it. Moreover, in the case of partitioning using the Fréchet distance, the problem is more complicated because the classical criteria are designed to be used with classical distances [11]. Nevertheless, Silhouette index with Fréchet distance was fully programmed as the Silhouette base function in the R package was implemented only with the Euclidean distance. This was done with the purpose of choosing the "right" number of clusters also for the kmlShape algorithm.

3.2.3.2 External validity indices

External validity indices are mainly used to compare the true partition with the one obtained from the clustering. In order to do so, a knowledge of the true partition is needed but, since the three methods applied in this project are unsupervised, this external information is not known. Even so, as mentioned before, the external CVIs can also be applied to compare the similarity between the different sets of clusters obtained from different clustering algorithms. For this end, the Jaccard index was programmed as it was not implemented in R and subsequently the matching matrix tool was used.

Jaccard index

The Jaccard index measures the similarity between two datasets. The values of the index vary between 0 and 1, indicating that the two datasets are more similar when closer to 1. In our case, the two datasets were the different partitions generated by applying the different clustering techniques.

For the formulation of how to calculate Jaccard index the following example is proposed [13]. Let us consider a dataset with four observations $X = \{x_1, x_2, x_3, x_4\}$ and the following two partitions $P = \{\{x_1, x_2, x_4\}, x_3\}$, $Q = \{\{x_1, x_3, x_4\}, x_4\}$.

From here, all the combinations without repetition of two elements of the dataset are made and each one is classified in 4 possibilities:

a: The pair of observations belong to the same cluster according to P and Q

b: The pair of observations belong to the same cluster according to P but not Q

c: The pair of observations belong to the same cluster according to Q but not P

d: The pair of observations do not belong to the same cluster according to either P or Q

This can be simplified in the following concordance table:

Table 3.1: Concordance table for each pair of observations

		Q	
		Pairs in Q	Pairs not in Q
P	Pairs in P	a	c
	Pairs not in P	b	d

From here, Jaccard index can be calculated as:

$$Jaccard = \frac{a}{a + b + c} \tag{3.11}$$

Matching matrix

Matching matrix is not considered an external validity index, but it is a tool that is also used for comparing to what extent each two partitions are similar. The matrix dimensions depend on the number of clusters of each method.

Normally, the confusion matrix is used in classification algorithms to calculate the precision by comparing the predicted class with respect of the real one, but when used in unsupervised learning such as for clustering, it is called matching matrix. In our case, the matching matrix was calculated in order to find similarities between clusters obtained by different algorithms. This allowed to see the equivalences between clusters of different methods. While green cells in Table 3.2 show the number of patients clustered in the same group in both methods, red cells show the number of individuals who have done so differently.

Table 3.2: Matching matrix

	0	a	b	c
Partition 2	**1**	d	e	f
	2	g	h	i
		0	**1**	**2**
			Partition 1	

As an example, let us consider that the horizontal axis represents the different labels obtained by applying the K-means algorithm with Euclidean distance and vertical axis the labels obtained with K-means and DTW. In this context $'g'$ will store the number of individuals that have been clustered in group 0 with Euclidean distance and in group 1 with DTW. If all red cells had a value of 0, it would mean that both algorithms have generated the same partition.

3.3 Description of the individuals gathered in each cluster

3.3.1 Descriptive statistics

After using clustering methods where individuals were clustered according to the different antibody levels trajectories, descriptive statistics were used. With it, the different variables characterising the individuals were statistically described for each group. Numerical variables such as the age, the number of diseases or the number of different drugs consumed were described by their mean and their standard deviation. On the other hand, categorical variables such as sex, COVID-19 severity, reinfections and the different diseases were described by counts and percentages. Once described, differences between groups were studied by statistical tests and measured through the obtained p-values. This allowed finding the most statistically significant variables (p<0.05). In this way it was possible to see in which variables the different groups differed most in order to later be able to draw conclusions.

Observed/expected ratio and exclusivity ratio

The observed/expected (O/E) ratio and the exclusivity ratio (EX) where also computed in this work with the purpose of describing the clusters and finding whether a variable was overrepresented or not in any given cluster. Only the categorical variables such as COVID-19 severity, reinfection, chronic condition and drugs are the variables which can be deeply analysed using O/E and EX. For the formulation let us consider reinfections variable.

For the calculation of the O/E ratio let us define first prevalence P. Prevalence is defined as the proportion of people who have been reinfected. It is calculated as follows:

$$P = \frac{Number\, of\, people\, reinfected}{Total\, number\, of\, people} \tag{3.12}$$

From here O/E ratio is calculated as follows:

$$O/E = \frac{Prevalence\, of\, reinfections\, in\, the\, cluster}{Prevalence\, of\, reinfections\, in\, the\, study} \tag{3.13}$$

Exclusivity ratio is defined as the proportion of individuals with the chronic condition for a certain cluster over the total of individuals with the chronic condition.

$$EX = \frac{Number\, of\, people\, reinfected\, in\, the\, cluster}{Number\, of\, people\, reinfected\, in\, the\, study} \tag{3.14}$$

A significant variable is observed when $O/E >= 2 or EX >= 0.3$ [36].

3.3.2 Clinical interpretation

Once the statistical description was available, the different groups obtained through the clustering methods were studied and compared using a clinical perspective. This allowed to identify which clinical variables were more determinant or related to the immune response to SARS-CoV-2 elicited by COVID-19.

For this purpose, special attention was paid to whether there were differences in the mean age or sex of the individuals in each cluster, their previous diseases or the number of drugs they were taking. In addition, consideration was also given to whether their COVID-19 infection had been more or less severe, and thus antibodies had reached higher or lower levels, and to whether this was related to whether individuals in one cluster had been reinfected more than those in another. All this analysis can be used to define and adjust prevention policies such as, for example, vaccination.

4 Results

4.1 Data pre-processing

Participants characteristics

A total of 844 individuals were recruited, of whom 671 (79.5%) were infected by COVID-19. Table 4.1 presents the demographic description as well as the number of diseases and drugs for the study participants according to their clinical condition.

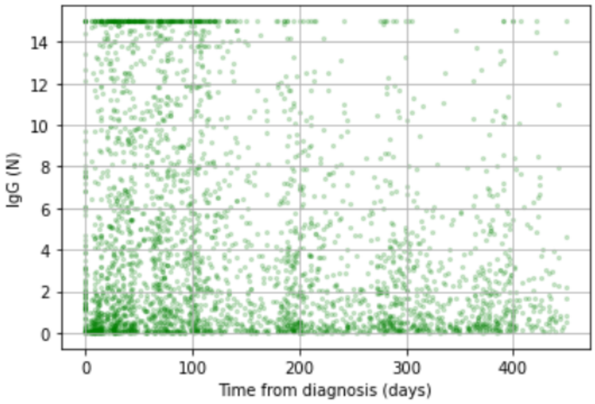
Table 4.1: Descriptive analysis for the study participants. Categorical variables are described as N (%), and numerical variables as median (IQR)

	Healthy N=173 (20.5%)	**Asymptomatic N=205 (24.3%)**	**Mild-moderate illness N=431 (51.1%)**	**Severe-critical illness N=35 (4.1%)**	**Total N=844**
Sex:					
Female	138 (79.8%)	148 (72.2%)	324 (75.2%)	17 (48.6%)	627 (74.3%)
Male	35 (20.2%)	57 (27.8%)	107 (24.8%)	18 (51.4%)	217 (25.7%)
Age:	47.6 (11.5)	43.4 (12.6)	43.8 (11.7)	54.5 (9.05)	44.9 (12.0)
Civil status:					
Divorced	26 (15.5%)	17 (8.72%)	36 (8.74%)	3 (9.09%)	82 (10.1%)
Married	114 (67.9%)	124 (63.6%)	303 (73.5%)	27 (81.8%)	568 (70.3%)
Single	23 (13.7%)	48 (24.6%)	68 (16.5%)	2 (6.06%)	141 (17.5%)
Widow/er	5 (2.98%)	6 (3.08%)	5 (1.21%)	1 (3.03%)	17 (2.10%)
Origin:					
EU	1 (0.60%)	0 (0.00%)	1 (0.25%)	0 (0.00%)	2 (0.26%)
Other	4 (2.41%)	6 (3.17%)	18 (4.53%)	0 (0.00%)	28 (3.57%)
South America	5 (3.01%)	4 (2.12%)	25 (6.30%)	1 (3.12%)	35 (4.46%)
Spain	156 (94.0%)	179 (94.7%)	353 (88.9%)	31 (96.9%)	719 (91.7%)
Education:					
Higher level	18 (10.4%)	19 (9.27%)	42 (9.74%)	2 (5.71%)	81 (9.60%)
Other	22 (12.7%)	59 (28.8%)	125 (29.0%)	9 (25.7%)	215 (25.5%)
University	133 (76.9%)	127 (62.0%)	264 (61.3%)	24 (68.6%)	548 (64.9%)
Job:					
Doctor	65 (40.4%)	45 (26.0%)	100 (28.8%)	10 (34.5%)	220 (31.0%)
Management	25 (15.5%)	19 (11.0%)	51 (14.7%)	4 (13.8%)	99 (13.9%)
Nurse	52 (32.3%)	68 (39.3%)	108 (31.1%)	11 (37.9%)	239 (33.7%)
Nurse assistant	9 (5.59%)	22 (12.7%)	34 (9.80%)	2 (6.90%)	67 (9.44%)
Other	6 (3.73%)	19 (11.0%)	48 (13.8%)	2 (6.90%)	75 (10.6%)
Social worker	4 (2.48%)	0 (0.00%)	6 (1.73%)	0 (0.00%)	10 (1.41%)
Number of chronic conditions:					
0 diseases	19 (11.0%)	30 (14.7%)	55 (12.8%)	1 (2.86%)	105 (12.5%)
1 disease	34 (19.8%)	46 (22.5%)	59 (13.7%)	5 (14.3%)	144 (17.1%)
2-4 diseases	59 (34.3%)	66 (32.4%)	168 (39.0%)	10 (28.6%)	303 (36.0%)
5 or more diseases	60 (34.9%)	62 (30.4%)	149 (34.6%)	19 (54.3%)	290 (34.4%)
Number of distinct drugs:					
0-3 drugs	7 (4.22%)	23 (11.6%)	32 (7.58%)	2 (5.71%)	64 (7.79%)
4-6 drugs	17 (10.2%)	24 (12.1%)	36 (8.53%)	3 (8.57%)	80 (9.73%)
7-10 drugs	25 (15.1%)	38 (19.1%)	68 (16.1%)	2 (5.71%)	133 (16.2%)
>10 drugs	117 (70.5%)	114 (57.3%)	286 (67.8%)	28 (80.0%)	545 (66.3%)

Data temporal dispersion

Figure 4.1 shows all samples taken in the first 15 months of infection. There was a notable temporal dispersion in the data, because as presented in the figure, the serological tests performed after infection were performed on different days and not on the specific days given by the protocol. However, a higher concentration was observed around days 15, 30, 60, 90, 180, 270, 360.

Figure 4.1: Temporal dispersion on the dataset



On the other hand, as shown in Figure 4.2 later, this dispersion was minimised after the individual selection process for the conventional methods by taking only those individuals that had samples around the same days.

Data selection

As previously mentioned, a total of 671 patients were infected by COVID-19. However, since the start of the study, not all of them underwent the same number of serological tests (samples) to measure their antibody levels. Table 4.2 shows the number of infected patients, grouped according to the total number of follow-up samples they contain. From here it can be seen, for example, that there are 124 participants who have 6 follow-up samples and only 12 who have 10.

Table 4.2: Number of infected patients, grouped according to the total number of follow-up samples

n_samples	1	2	3	4	5	6	7	8	9	10	11	12	13	15
n_patients	20	36	51	28	43	124	134	163	47	12	4	4	4	1

The above table takes into account all tests performed on patients with a COVID-19 infection in their history. However, as mentioned in database section of 3.1, it was necessary to filter only those samples that have been obtained post-infection and prior to reinfection.

Once this filter was applied in order to work only with the useful ones, a total of 625 participants were obtained who had at least one immune response measurement. Table 4.3 shows the new grouping of patients according to their number of infection samples. This decrease and the fact that there are a large number of patients with few follow-up samples was due to two possible reasons: the participants were infected before the start of the study, i.e., between March and May 2020, when serological tests were not done systematically, or they simply dropped out the study at some point either before or after the infection.

Table 4.3: Number of infected patients, grouped according to the total number of samples after removing those obtained post-infection and pre-reinfection

n_samples	1	2	3	4	5	6	7	8	9	10
n_patients	59	68	47	33	40	95	101	137	39	6

In addition to this general selection, used for example when knowing the data dispersion of Figure 4.1, a different selection of individuals was made from this, depending on the type of clustering model to be applied.

Conventional clustering

Once the data were explored and from the knowledge that serological tests were repeated at day 15, 30, 60, 60, 90, 180, 270 and 360 after infection, the first sets of intervals were proposed to find out how many patients were grouped in each case. Table 4.4 shows for each set of periods, the number of participants who had samples in all intervals (A0) and those who had samples in all but one (A1).

Table 4.4: Number of patients grouped for different sets of intervals

Intervals	**A0**	**A1**
[0,10],[11,25],[26,45],[46,75],[76,140],[141,230]	57	108
[0,10],[11,20],[25,45],[50,75],[80,120],[160,200]	19	85
[0,10],[11,25],[26,45],[46,75],[76,140],[141,230], [231,315]	50	87
[0,10],[11,20],[25,45],[50,75],[80,120],[160,200], [250,290]	14	63
[0,10],[11,25],[26,45],[46,75],[76,140],[141,230], [231,315], [316,440]	45	73
[0,10],[11,20],[25,45],[50,75],[80,120],[160,200], [250,290], [330,420]	11	56

Based on these initial analyses, it was decided to use intervals with continuity, as despite having slightly more dispersion, they considered a larger number of patients than those without continuity. In addition, after one year of follow-up, some patients did not have more samples, so it was decided to work with a total of 7 intervals to analyse the immune response in the first 365 days. To finish the exploratory analysis, the ranges for these 7 intervals with continuity were optimised by choosing those that considered the maximum number of patients. The final set of intervals is as follows.

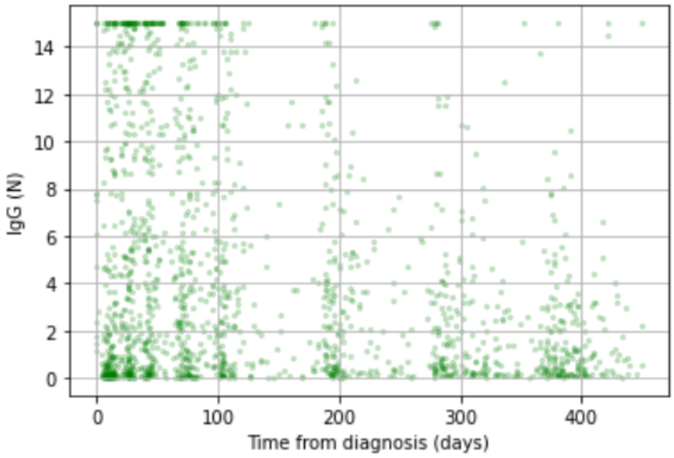
Table 4.5: Patients grouped with the chosen set of intervals

Intervals	**A0**	**A1**
[0,18],[19,34],[35,63],[64,90],[91,175],[176,270],[271,366]	122	80

These 202 selected patients went through the data preparation process explained in 3.1 obtaining 7 samples for each patient, making a total of 1414 samples.

Figure 4.2 shows that once this process for the conventional methods of selecting patients with equally distributed samples is done, the data did not present as much temporal dispersion as it did with the whole dataset. This concentration of data benefited when clustering with K-means since, although only antibody levels were taken into account and not the day on which they were measured, it ensured that the different samples of individuals were not compared on days temporarily separated.

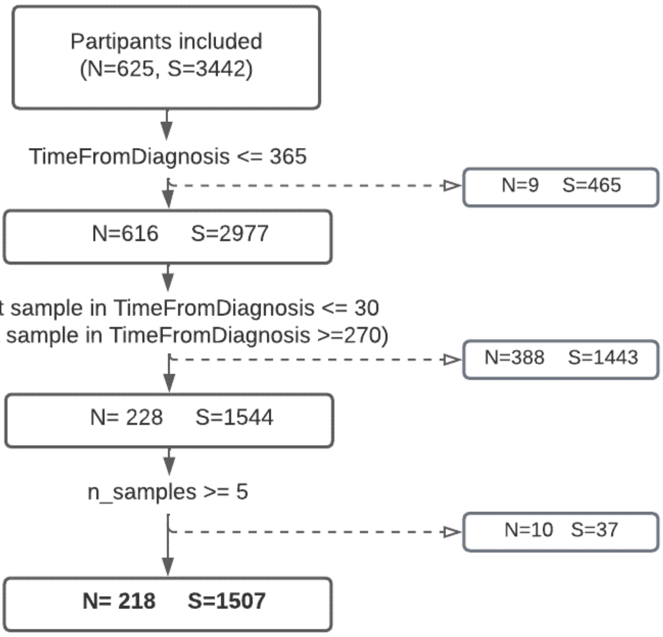
Figure 4.2: Temporal dispersion after data selection for conventional methods



Longitudinal clustering

Figure 4.3 presents the number of patients (N) and total samples (S) resulting from the application of the different conditions in order to select those individuals that will later be clustered with the kmlShape algorithm.

Figure 4.3: Participants selection results for kmlShape algorithm. N=number of patients, S= total number of samples



To carry out the first tests with the longitudinal method, only the condition that the participants had a minimum number of samples was considered. However, the results with only this condition were not valuable as the model was heavily influenced by when individuals had their first or last sample. For example, it was observed that a centroid started on day 40, clustering individuals which their follow-up samples started around that day independently of the evolution of their antibody levels. In view of this, three more conditions were considered to avoid this problem.

The first was to perform the analysis of antibody levels for only one year of follow-up, and once it was filtered to eliminate all those samples taken later than 365 days after infection, it was observed that the number of samples was considerably reduced. 9 patients out of 625 were also reduced, which means that they only underwent serological tests once the year of infection had passed. Of these 616 patients, two new conditions were applied to ensure that participants had data at the beginning and end of follow-up. After eliminating those which did not have the first sample before day 30 since diagnosis and the last after day 270, a total of 228 was obtained. This considerable reduction is due to the fact that a large part of the participants did not have serological test results in these first days after infection, which turned out to be quite important days to differentiate the different immune responses. Finally, only individuals who had 5 or more samples were chosen once applied these conditions, resulting in a total of 218 patients.

As kmlShape allowed entries of different lengths, the 218 individuals that were clustered with this algorithm are distributed as follows.

Table 4.6: Distribution of patients selected for kmlShape, according to number of samples

n_samples	5	6	7	8	9
n_patients	19	34	113	51	1

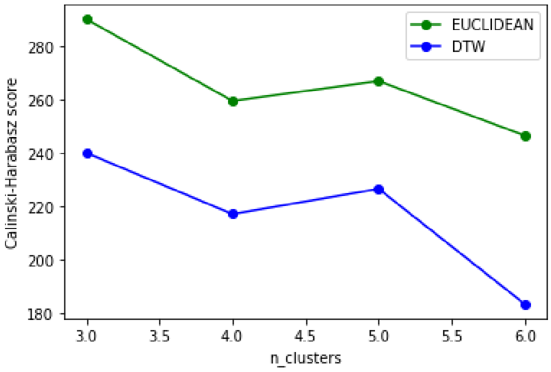
4.2 Clustering

4.2.1 Conventional clustering

This section presents the clustering results using the K-means algorithm with the Euclidean distance and secondly the K-means with the DTW. Both techniques made use of the 202 patients selected in the data pre-processing step.

To find the best number of clusters to group the patients, the validity indices were used. Figure 4.4 shows the results of the Calinski-Harabasz index for the two techniques mentioned, where it was found that both methods gave similar results.

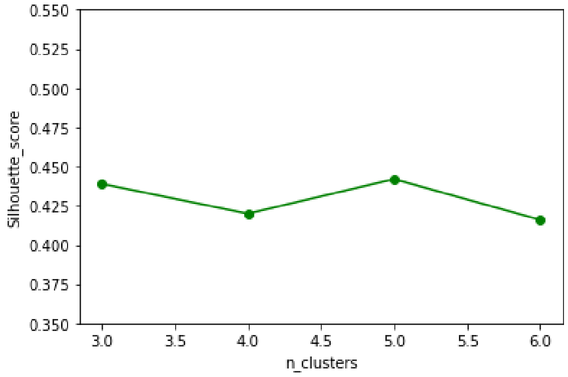
Figure 4.4: Calinski-Harabasz index for K-means with Euclidean distance and K-means with DTW



Nevertheless, when using DTW patients were not equally distributed and the temporal alignment was sometimes not as desired, since in some cases samples were compared on days that were quite far apart in time. Because of this, partitions were created such as the one that can be seen later in Figure 4.11b, where the peak of the cluster with lower antibody levels was around day 100, when it should be around day 30. In addition, despite obtaining validity index higher values when grouping with 3 and 5 clusters, no clear conclusion was reached as to the optimal number of intervals. Because of this, for the K-means with the Euclidean distance, which gave the best results analytically and visually, it was proposed to work with another index to see if clearer conclusions could be drawn.

Figure 4.5 shows the different scores of the Silhouette validity index applied to the K-means with Euclidean distance.

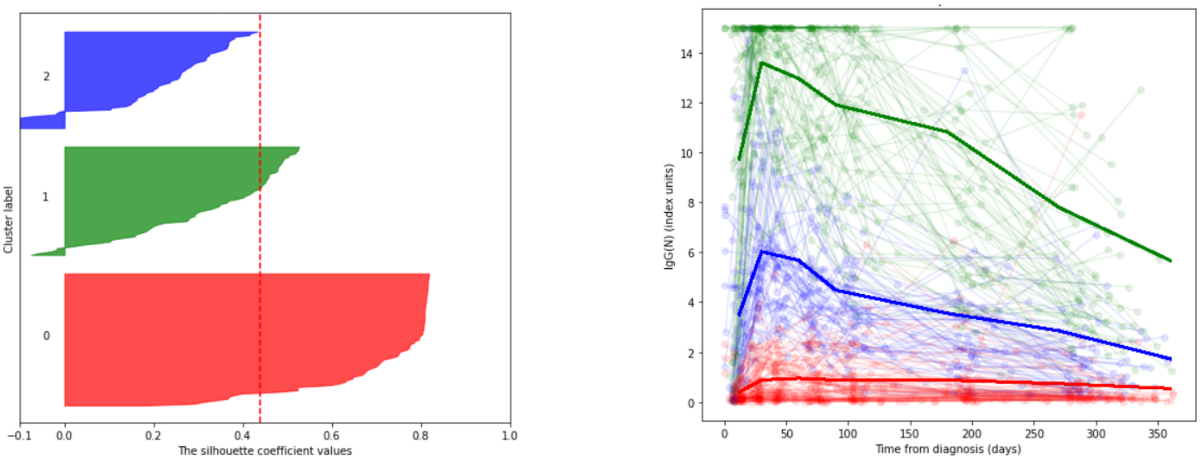
Figure 4.5: Silhouette coefficient for K-means with Euclidean distance



Again, although no significant differences were observed in finding the optimal number of clusters, it was decided to choose the groupings of 3 and 5 clusters, since in both indices a slightly better result was obtained.

Figure 4.6a shows the silhouette scores for each cluster. Figure 4.6b shows the final partition made with 3 clusters, separating clusters by colours where the thick lines are the centroids of the groups and the thin lines are the individuals assigned to each cluster. A 'better clustering' was observed for individuals with lower antibody levels grouped in cluster 0, as their Silhouette scores were closer to 1 with no negative values for any patient. Also, from the thickness of the Silhouette plot it was possible to visualise the size of the cluster. It was observed that for clustering with 3 clusters, the thickness was similar for all three, indicating that each group had a similar number of patients. However, when it was clustered with 5 clusters as shown in Figure 4.7a, one group was seen to have a significantly higher number of patients than the others, including a total of 79 out of 202 who had higher Silhouette scores.

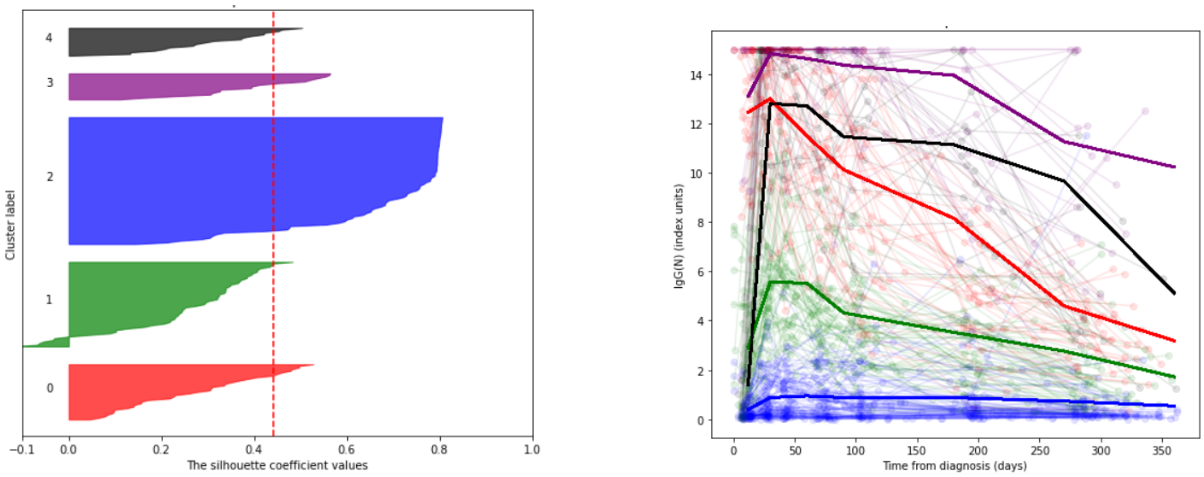
Figure 4.6: Silhouette analysis and final partition with 3 clusters



(a) Silhouette plot

(b) Final partition

Figure 4.7: Silhouette analysis and final partition with 5 clusters



(a) Silhouette plot

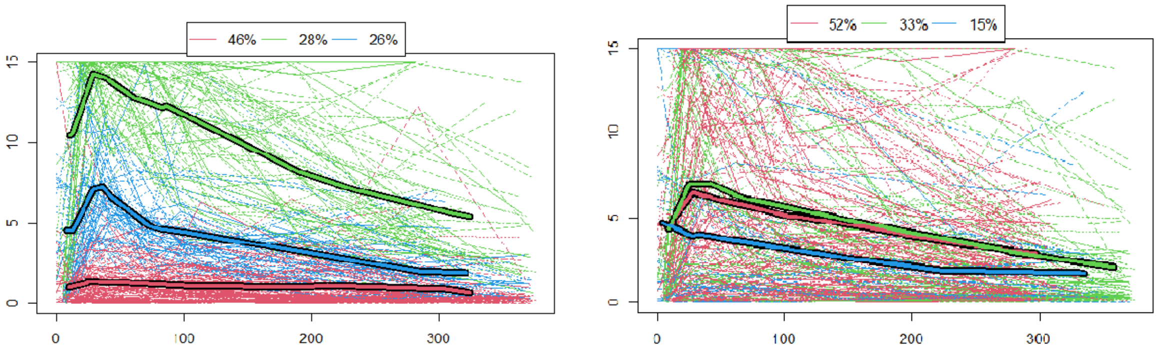
(b) Final partition

In all clusters, a generalized increase in antibody levels was observed up to day 30 from diagnosis, followed by a decrease. However, the values of the antibody levels were quite different for each group, so it was important to describe statistically each one in order to find the characteristics of the patients that caused these levels.

4.2.2 Longitudinal clustering

As explained in 3.2.2, modifying the timeScale parameter of the Fréchet distancefor the kmlShape algorithm, it was possible to give different cost to the horizontal shift. In order to find the value that best suited our dataset, several clustering were done by trying different numbers. Figure 4.8 shows a comparison of the partitions resulting from the application of two different timeScales.

Figure 4.8: Impact of timeScale parameter in kmlShape algorithm. X -axis represents time from diagnosis in days and Y-axis represents the IgG (N) antibody levels.



(a) timeScale = 0.05

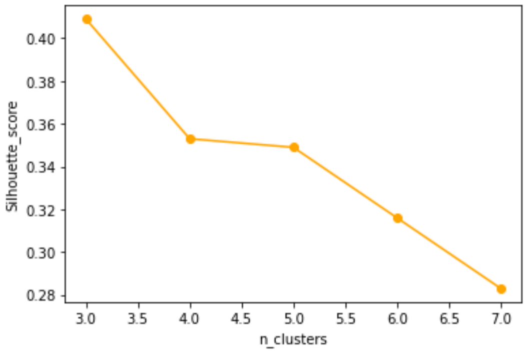
(b) timeScale = 0.7

It was observed that choosing a smaller timeScale, making the Fréchet distance more similar to the DTW, gave better results, as it allowed the different groups to be distinguished according to their antibody levels as it can be seen in the final partition of Figure 4.8a. In addition, when a higher timeScale was chosen, it was observed that the creation of the different clusters was influenced by the day on which the patients had their last sample. This can be seen by comparing the green cluster and the red cluster in Figure 4.8b.

After these analyses were done, it was decided to work with a timeScale = 0.05.

The next step was to obtain the optimal number of clusters for the kmlShape. For this, the Silhouette index was adapted to work with the Fréchet distance. For the 218 selected patients and clustering with timeScale = 0.05, the results in Figure 4.9 were obtained for this index.

Figure 4.9: Silhouette coefficient for kmlShape

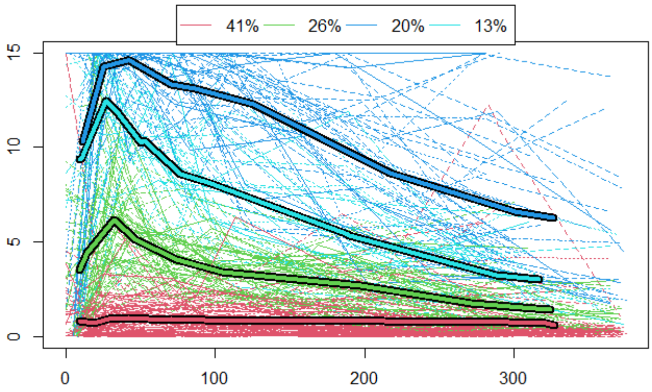


From Figure 4.9 it can be seen that the best clustering for kmlShape algorithm was obtained when the population was grouped into 3 clusters. However, since clinical analysis is the most useful validity to understand the results and to confirm this choice, the descriptive statistics produced a posteriori were studied for the clustering with 3, 4, 5 clusters, which were the options that gave the best results in terms of Silhouette. From this study, different conclusions were drawn regarding the optimal number of clusters.

Firstly, it was seen that with 3 clusters there were significant differences between the groups, but these results were not sufficiently innovative at the clinical level, i.e., they did not contribute anything new to what was expected. Secondly, when 5 clusters were used, it was observed that three large groups were generated and the other two were too small, which were splits of the former.

Given these circumstances, it was decided that the optimal partition was the one created with 4 clusters depicted in Figure 4.10, as it provided the most value at the clinical level. The percentages shown at the top indicate how many individuals are in each group out of the 218 total.

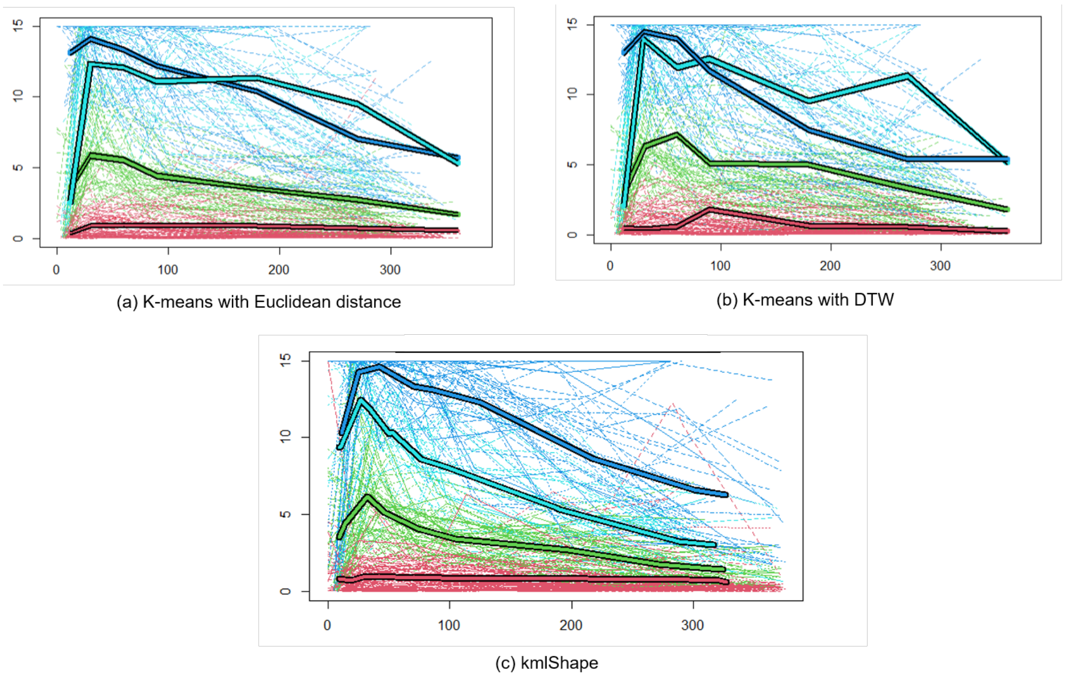
Figure 4.10: kmlShape with 4 clusters



4.2.3 Clusterings comparison

After choosing 4 clusters as the best option, Jaccard's external index and the coincidence matrix were used to compare the degree of similarity between two partitions. To do this, the partitions created with the three different algorithms were compared two by two. In the case of the two K-means clusterings, they were compared using the 202 selected patients as they made use of the same dataset. However, for the comparison of these two conventional methods with kmlShape only those patients selected in both methods were used, resulting in a total of 175 of the 218 selected for longitudinal methods.

Figure 4.11: Final partitions with 4 clusters. Cluster 0 is represented with red color. Cluster 1 is represented with dark blue color. Cluster 2 is represented with green color. Cluster 3 is represented with cyan color.



(a) K-means with Euclidean distance

(b) K-means with DTW

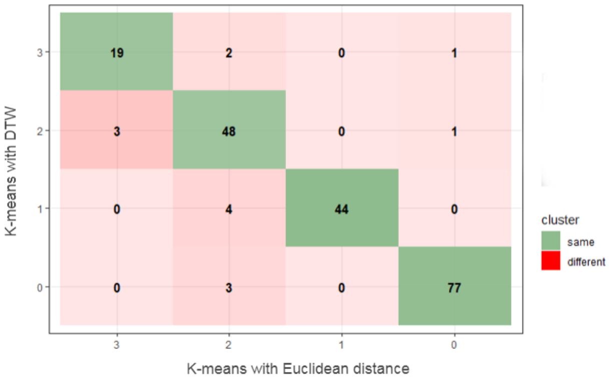
(c) kmlShape

Figure 4.11a and Figure 4.11b are the same as shown in Figure 4.6 and Figure 4.7, produced with the two conventional methods, but now adapted to R in order to have the same cluster names and to be able to compare them with the following techniques.

Matching matrix

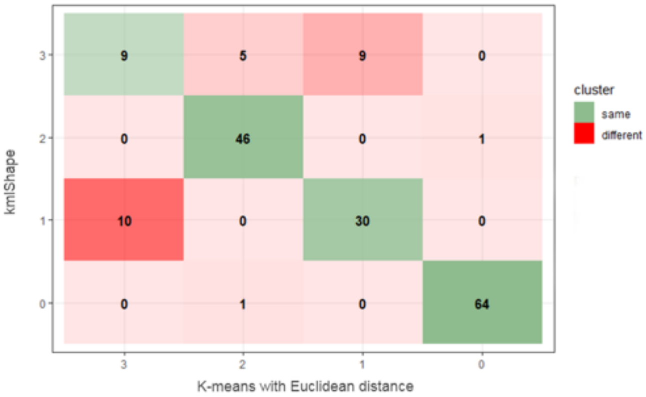
Thanks to the matching matrix, similarities and differences between different clusterings were found. Figure 4.12 shows the matching matrix between K-means with Euclidean distance and K-means with DTW where 93% of the patients were grouped in the same cluster. The most significant changes occurred in cluster 2 (green), where it was seen that for this group 9 out of 57 participants were clustered in the other groups.

Figure 4.12: Matching matrix between K-means with Euclidean distance and K-means with DTW



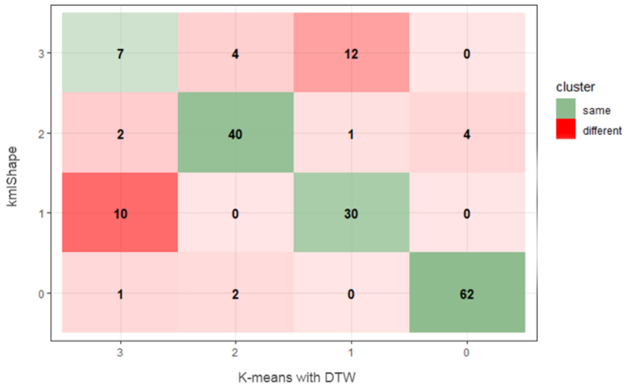
As can be seen in the following figures, more differences were observed when comparing the conventional methods with the longitudinal one. When comparing K-means with Euclidean distance and kmlShape (Figure 4.13), 26 of the 175 patients used in both methods were clustered differently. On the other hand, fewer similarities were seen when comparing K-means with DTW and kmlShape (Figure 4.14), as 37 of the 175 patients were clustered differently.

Figure 4.13: Matching matrix between K-means with Euclidean distance and kmlShape



It was observed that when comparing the conventional methods with the longitudinal one, the partitions differed in the same way, as more than 70% of the differences were observed between clusters that grouped patients with higher antibody levels (clusters 1 and 3).

Figure 4.14: Matching matrix between K-means with DTW and kmlShape



Jaccard index

Table 4.7 shows the different index values obtained by comparing two by two the partitions shown in Figure 4.11. As explained in the methodology section, the Jaccard index measures the similarity between two partitions of a dataset, indicating that they are more similar when the index is closer to one. Once this index was calculated, the same conclusions were reached as with the matching matrix, with the partitions of the conventional methods being the most similar to each other and those created by applying the K-means with DTW and kmlShape being the most different.

Table 4.7: Jaccard index results

Clusterings	Jaccard
K-means with Euclidean distance and K-means with DTW	0.7957
K-means with Euclidean distance and kmlShape	0.7217
K-means with DTW and kmlShape	0.6269

Once all the tests had been carried out, the kmlShape algorithm was chosen as the optimal one. This was because this method was specially designed to work with longitudinal data without quantifying the time in which each sample was taken, being the one that best modelled the different temporal evolutions of the antibody levels. Moreover, it was also taken into account for the decision, the fact that kmlShape do not require all individuals to have the same number of samples, so for health studies where dropouts occur it may be the most convenient option. Subsequently, descriptive statistics corroborated that it was the optimal method, as it was the one that allowed finding the most significant differences between groups.

4.3 Description of the individuals gathered in each cluster

4.3.1 Descriptive statistics

Table 4.8 lists the main variables that characterise the individuals in the different groups formed with the chosen optimal algorithm, the kmlShape with 4 clusters represented in Figure 4.10.

Statistically significant differences in antibody levels between clusters were found in age, COVID-19 severity and number of symptoms. These significant variables and other characteristics that distinguish the four groups are presented in the clinical interpretation section

Table 4.8: Descriptive statistics of the individuals in each cluster using kmlShape

	N=89 (41%)	**N=57 (26%)**	**N=43 (20%)**	**N=29 (13%)**	**overall-value**
Sex:					0.020
Female	68 (76.4%)	43 (75.4%)	29 (67.4%)	14 (48.3%)	
Male	21 (23.6%)	14 (24.6%)	14 (32.6%)	15 (51.7%)	
Age	42.5 (11.6)	40.9 (10.9)	51.8 (12.2)	45.9 (13.1)	<0.001
Reinfection:					0.245
0	84 (94.4%)	50 (87.7%)	42 (97.7%)	28 (96.6%)	
1	5 (5.62%)	7 (12.3%)	1 (2.33%)	1 (3.45%)	
Number of vaccines:					0.980
0	83 (93.3%)	54 (94.7%)	41 (95.3%)	28 (96.6%)	
1	4 (4.49%)	3 (5.26%)	2 (4.65%)	1 (3.45%)	
2	2 (2.25%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	
COVID-19 severity:					<0.001
Asymptomatic infection	33 (37.1%)	4 (7.02%)	8 (18.6%)	6 (20.7%)	
Mild moderate Illness	56 (62.9%)	52 (91.2%)	30 (69.8%)	22 (75.9%)	
Severe or critical Illness	0 (0.00%)	1 (1.75%)	5 (11.6%)	1 (3.45%)	
Number of symptoms	2.99 (3.31)	5.44 (3.71)	5.12 (3.49)	5.62 (4.75)	<0.001
Number of chronic conditions	3.33 (2.54)	3.91 (3.00)	4.88 (4.29)	5.34 (3.88)	0.009
Number of distinct drugs	12.7 (9.32)	14.9 (10.2)	15.4 (10.8)	15.3 (10.6)	0.348
Number of invoiced drugs	66.7 (153)	92.6 (166)	196 (259)	123 (178)	0.002

Observed/expected ratio and exclusivity ratio

To better describe the groups, O/E ratio and exclusivity ratio EX were calculated as shown in Table 4.9 for the COVID-19 severity and reinfections variables. High percentage of reinfections were found in clusters with lower antibody levels. Moreover, significant differences were also found in COVID-19 severity variable. The study of these results is shown in Table 4.9 and expressed in the next section.

Table 4.9: O/E and Exclusivity of reinfections and COVID-19 severity in each cluster. Significant values to better describe a cluster are show in yellow cells (O/E>=2 or EX>=0.3)

	N=89		N=57		N=43		N=29	
	O/E	**EX**	**O/E**	**EX**	**O/E**	**EX**	**O/E**	**EX**
Reinfection	0.874	0.357	1.912	0.500	0.362	0.071	0.537	0.071
COVID-19 severity:								
Asymptomatic infection	1.584	0.647	0.299	0.078	0.795	0.157	0.884	0.117
Mild moderate illness	0.857	0.350	1.243	0.325	0.950	0.188	1.033	0.137
Severe or critical illness	0	0	0.546	0.143	3.621	0.714	1.074	0.143

4.3.2 Clinical interpretation

The different groups of the final partition shown before in Figure 4.10 were studied and compared from a clinical perspective as shown below.

Red cluster

Group with the lowest antibody levels, representing 41% of the included participants. The average age is 42.5 years and 76.4% are women. 65% of total asymptomatic cases are found in this group, which also has the lowest number of chronic diseases, the fewest symptoms during infection and the fewest different medications. They also account for 35.7% of total reinfections. Therefore, this is a group of, widely speaking, healthy individuals who had an asymptomatic COVID-19 infection and did not develop lasting antibodies. Thus, a considerable number of reinfections occur in this group.

Green cluster

Represents 26% of the included participants. It is the youngest group with an average age of 41 years and with a higher proportion of women. 91.2% of the individuals had a mild COVID-19 infection. It is the cluster with the second lowest number of chronic conditions, but also the second group in number of symptoms during infection. 50% of the total reinfections are found in this cluster. Therefore, this is a group of healthy individuals who had a mild COVID-19 infection. They developed medium antibody levels that quickly decreased. Thus, most reinfections occur in this group.

Dark blue cluster

Representing 20% of the included participants, it is the group with the highest and more lasting in time antibody levels. It is the oldest group with an average age of 51.8 years and with 71% of the total number of severe COVID cases. It also has the highest number of different medications and the highest number of billed drugs. Despite being the second group with the highest number of chronic conditions. Only 1 reinfection is found in this cluster. Therefore, this is a older group with a poor health status. This status could have influenced the severity of their COVID-19 infection, which developed higher levels of antibody levels. Thus, the number of reinfections is low.

Cyan cluster

13% of the included individuals with an average age of 46 years. It is the group with the higher proportion of men (51.3%) and with the higher number of symptoms during the COVID-19 infection. It contains a high percentage of COVID-19 mild infection cases and is the cluster with the highest number of chronic conditions. The patients in this group are the second with the fewest number of different medicines, but also the second with the greatest number of invoiced drugs. Only 1 reinfection is found in this cluster. Similarly, this group of adults have a poor health status. However, their chronic conditions did not interfere that much with COVID-19 as those from the dark blue cluster. Therefore, their response to COVID-19, even though high, was lower than theirs. The number of reinfections is also low.

5 Budget

In this chapter the total cost of the project is estimated.

Only the cost of salaries is estimated since, as stated in the methodology chapter, the implementation of the code to carry out the analyses was done in Python and R, which does not involve any cost since they are Open-source programming languages. For the student we assume a salary of 9€/h, which is the standard undergraduate internship salary and for the 3 project supervisors we estimate an average salary of 30€/h with a dedication of 3h/week. The project budget is estimated in Table 5.1.

Table 5.1: Project budget

	Amount	**Cost/hour**	**Dedication**	**Total**
Student	1	9 €/h	450 h	4.050 €
Project supervisor	3	30 €/h	54 h	4.860 €
				8.910 €

6 Conclusions and future development

The main objective of the project was to study different clustering methods in order to group individuals according to the evolution of their immune response to SARS-CoV-2. Different techniques are presented in this study, including both conventional techniques such as K-means and others specifically designed to deal with longitudinal data such as kmlShape. In addition, different validity indices were studied to find the optimal number of clusters.

Both the Calinski-Harabasz index and the Silhouette index showed that the optimal number of clusters was 3 or 5. However, once the clinical analysis had been carried out, it was seen that the results with these numbers were not very innovative at a clinical level beyond what was expected, so 4 was chosen as the optimal number of clusters. Therefore, clinical interpretation is the most useful method for explaining the results, and it also helped to choose the model that best grouped the individuals. Between K-means with Euclidean distance, K-means with DTW and kmlShape, the latter was finally chosen as the method with the most significant differences between the groups. This was in line with what was expected, as kmlShape was the method especially designed to work with longitudinal data.

The last objective was to describe the patients included in each cluster in order to identify which characteristics of the individuals triggered the immune response of each group. To do this, a clinical interpretation was made from the descriptive statistics of each cluster. Four groups with different characteristics were found: , two 'healthier' groups one of asymptomatic people and one of people with mild COVID-19 infection with lower level of antibodies and high number of reinfections. The other two groups consisted of older people with more previous chronic conditions, which caused them to pass COVID-19 more severely. This severity led them to develop higher and longer-lasting antibody levels, resulting in a lower number of reinfections.

Future development

These longitudinal techniques could be applied to other types of immune responses, not only antibody responses, but also cellular responses. In addition, they could be applied to find types of evolutions of immune response to other viruses such as HIV or the flu. By identifying these groups, different predictive models of mortality or reinfection could be developed. These models could be more accurate thanks to division, instead of having a general predictive model applied to the complete population. These predictive models should also be developed considering the longitudinal evolution, so either survival Cox models or recurrent neural networks based Deep Learning models could be developed from this work.

References

[1] World Health Organization. Coronavirus disease (COVID-19) pandemic. `https://covid19.who.int/`.

[2] Marco Cascella, Michael Rajnik, Abdul Aleem, Scott C Dulebohn, and Raffaela Di Napoli. Features, evaluation, and treatment of coronavirus (COVID-19). *Statpearls*, 2022.

[3] Finlay Campbell, Brett Archer, Henry Laurenson-Schafer, Yuka Jinnai, Franck Konings, Neale Batra, Boris Pavlin, Katelijn Vandemaele, Maria D Van Kerkhove, Thibaut Jombart, et al. Increased transmissibility and global spread of SARS-CoV-2 variants of concern as at June 2021. *Eurosurveillance*, 26(24):2100509, 2021.

[4] Ali Khaleghi, Mohammad Reza Mohammadi, Gila Pirzad Jahromi, and Hadi Zarafshan. New ways to manage pandemics: Using technologies in the era of covid-19: A narrative review. *Iranian Journal of Psychiatry*, 15(3):236, 2020.

[5] Muzammil Khan, Muhammad Taqi Mehran, Zeeshan Ul Haq, Zahid Ullah, Salman Raza Naqvi, Mehreen Ihsan, and Haider Abbass. Applications of artificial intelligence in COVID-19 pandemic: A comprehensive review. *Expert Systems with Applications*, 185:115695, 2021.

[6] Concepción Violán, Pere Torán, Lucía A. Carrasco-Ribelles, et al. Antibody kinetics to SARS-CoV-2 at 13.5 months, by disease severity. *medRxiv*, 2021.

[7] Edward Joseph Caruana, Marius Roman, Jules Hernández-Sánchez, and Piergiorgio Solli. Longitudinal studies. *Journal of Thoracic Disease*, 7(11):E537, 2015.

[8] Brianna Christine Heggeseth. *Longitudinal cluster analysis with applications to growth trajectories*. University of California, Berkeley, 2013.

[9] Christophe Genolini and Bruno Falissard. Kml: A package to cluster longitudinal data. *Computer Methods and Programs in Biomedicine*, 104(3):e112–e121, 2011.

[10] Romain Tavenard. "An Introduction to Dynamic Time Warping.". `https://rtavenar.github.io/blog/dtw.html`.

[11] Christophe Genolini, René Ecochard, Mamoun Benghezal, Tarak Driss, Sandrine Andrieu, and Fabien Subtil. kmlShape: an efficient method to cluster longitudinal data (time-series) according to their shapes. *PLOS One*, 11(6):e0150738, 2016.

[12] John Paparrizos and Luis Gravano. k-Shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 ACM SIGMOD international conference on management of data*, pages 1855–1870, 2015.

[13] André Alexandre da Silva Galvão Garcia. Clustering of longitudinal data: Application to COVID-19 data. Master's thesis, Universidade do Porto, 2020.

[14] Shivam Agarwal. Data mining: Data mining concepts and techniques. In *2013 International Conference on Machine Intelligence and Research Advancement*, pages 203–207. IEEE, 2013.

[15] Eke Nnanna Arua. Clustering of longitudinal viral loads in the Western Cape. Master's thesis, Faculty of Health Sciences, 2020.

[16] Paul D McNicholas and T Brendan Murphy. Model-based clustering of longitudinal data. *Canadian Journal of Statistics*, 38(1):153–168, 2010.

[17] Yanchi Liu, Zhongmou Li, Hui Xiong, Xuedong Gao, and Junjie Wu. Understanding of internal clustering validation measures. In *2010 IEEE International Conference on Data Mining*, pages 911–916. IEEE, 2010.

[18] Brian S Everitt, Sabine Landau, Morven Leese, and Daniel Stahl. Cluster analysis 5th ed, 2011.

[19] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[20] David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 224–227, 1979.

[21] Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

[22] Abelardo Montesinos-López. *Estudio del AIC y BIC en la selección de modelos de vida con datos censurados.* PhD thesis, University of Guanajuato Guanajuato, Mexico, 2011.

[23] William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

[24] Paul Jaccard. Nouvelles recherches sur la distribution florale. *Bull. Soc. Vaud. Sci. Nat.*, 44:223–270, 1908.

[25] Kimiya Gohari, Anoshirvan Kazemnejad, Ali Sheidaei, and Sarah Hajari. Clustering of countries according to the COVID-19 incidence and mortality rates. *BMC Public Health*, 22(1):1–12, 2022.

[26] Eunyoung Emily Lee, Kyoung-Ho Song, Woochang Hwang, Sin Young Ham, Hyeonju Jeong, Jeong-Han Kim, Hong Sang Oh, Yu Min Kang, Eun Bong Lee, Nam Joong Kim, et al. Pattern of inflammatory immune response determines the clinical course and outcome of COVID-19: unbiased clustering analysis. *Scientific Reports*, 11(1):1–8, 2021.

[27] Rafael Assis, Aarti Jain, Rie Nakajima, Algis Jasinskas, Saahir Khan, Huw Davies, Laurence Corash, Larry J Dumont, Kathleen Kelly, Graham Simmons, et al. Distinct SARS-CoV-2 antibody reactivity patterns in coronavirus convalescent plasma revealed by a coronavirus antigen microarray. *Scientific Reports*, 11(1):1–12, 2021.

[28] Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B*, 63(2):411–423, 2001.

[29] Isaac Subirana, Héctor Sanz, and Joan Vila. Building bivariate tables: the compareGroups package for R. *Journal of Statistical Software*, 57:1–16, 2014.

[30] Pasi Fränti and Sami Sieranoja. How much can k-means be improved by using better initialization and repeats? *Pattern Recognition*, 93:95–112, 2019.

[31] Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666, 2010.

[32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[33] Zheng Zhang, Romain Tavenard, Adeline Bailly, Xiaotong Tang, Ping Tang, and Thomas Corpetti. Dynamic time warping under limited warping path length. *Information Sciences*, 393:91–107, 2017.

[34] Romain Tavenard, Johann Faouzi, Gilles Vandewiele, Felix Divo, Guillaume Androz, Chester Holtz, Marie Payne, Roman Yurchak, Marc Rußwurm, Kushal Kolar, et al. Tslearn, a machine learning toolkit for time series data. *J. Mach. Learn. Res.*, 21(118):1–6, 2020.

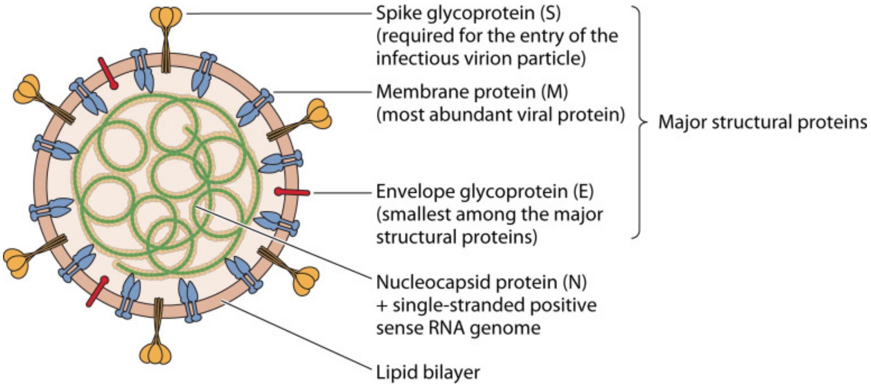
[35] Ning Guo, Mengyu Ma, Wei Xiong, Luo Chen, and Ning Jing. An efficient query algorithm for trajectory similarity based on fréchet distance threshold. *ISPRS International Journal of Geo-Information*, 6(11):326, 2017.

[36] Concepción Violán, Quintí Foguet-Boreu, Sergio Fernández-Bertolín, Marina Guisado-Clavero, Margarita Cabrera-Bean, Francesc Formiga, Jose Maria Valderas, and Albert Roso-Llorach. Soft clustering using real-world data for the identification of multimorbidity patterns in an elderly population: cross-sectional study in a mediterranean population. *BMJ Open*, 9(8), 2019.

A SARS-CoV-2 structure

The structure of Coronaviruses, of the family Coronaviridae, encodes four major structural proteins, namely, spike (S), membrane (M), envelope (E), and nucleocapsid (N) (Figure A.1). Coronavirus S protein is a large, multifunctional class I viral transmembrane protein. Homotrimers of the virus-encoded S protein make up the distinctive crown-like appearance on the surface of the virus. Functionally is required to enter the infectious virion particles into the cell through interaction with various host cellular receptors. Subtype S1 makes up the sizeable receptor-binding domain of the S protein, while S2 forms the stalk of the spike molecule. The M protein is the most abundant viral protein present in the virion particle, giving a definite shape to the viral envelope. It binds to the nucleocapsid and acts as a central organizer of coronavirus assembly. The N protein plays a role in complex formation with the viral genome and is also involved in other aspects of the CoV replication cycle. The host cellular response to viral infection facilitates M protein interaction needed during virion assembly. About E protein, the inactivation or absence of this protein is related to the alter virulence of coronaviruses due to changes in morphology, release, and tropism. [1] [2]

Figure A.1: The virus structure of SARS-CoV-2

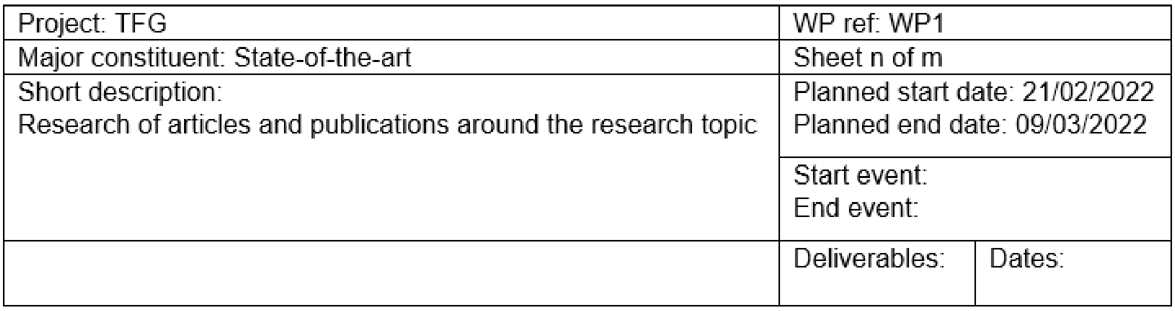

Spike glycoprotein (S)
(required for the entry of the infectious virion particle)
Membrane protein (M)
(most abundant viral protein)
Major structural proteins
Envelope glycoprotein (E)
(smallest among the major structural proteins)
Nucleocapsid protein (N)
+ single-stranded positive sense RNA genome
Lipid bilayer

[1] Dhama Kuldeep et al. Coronavirus Disease 2019–COVID-19. Clinical Microbiology Reviews. 2020;33(4):1–48.

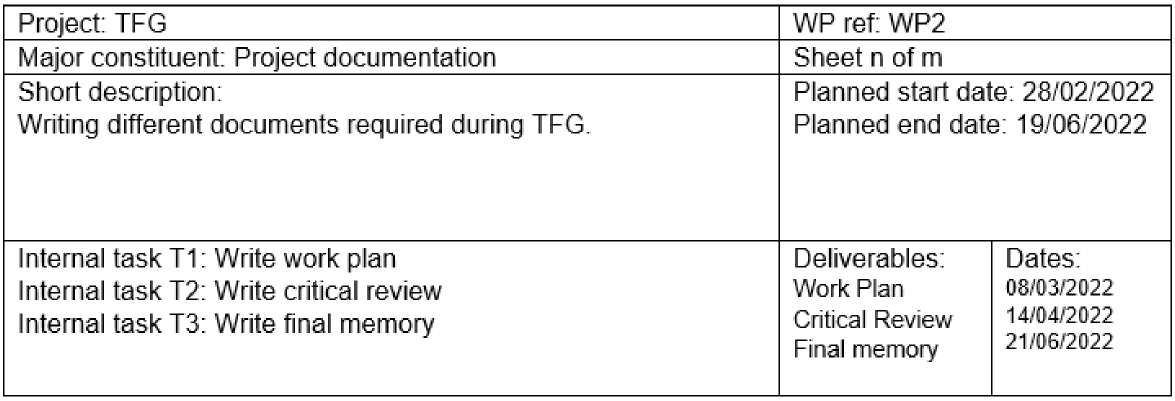
[2] Maier HJ et al. Coronaviruses: An Overview of Their Replication and Pathogenesis. Coronaviruses: Methods and Protocols. 2015;1282(1):1–282.

B Work Plan

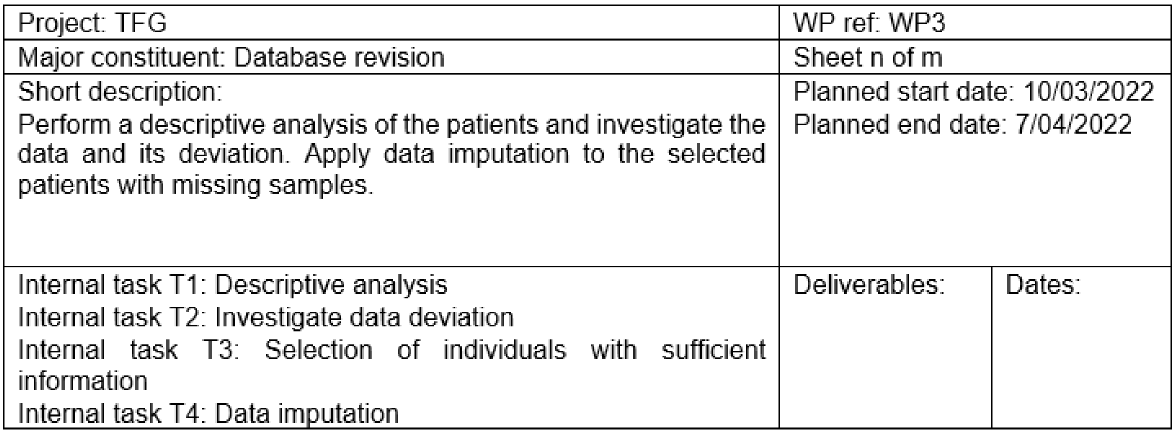
B.1 Work Packages



Project: TFG	WP ref: WP1	
Major constituent: State-of-the-art	Sheet n of m	
Short description: Research of articles and publications around the research topic	Planned start date: 21/02/2022 Planned end date: 09/03/2022	
	Start event: End event:	
	Deliverables:	Dates:



Project: TFG	WP ref: WP2	
Major constituent: Project documentation	Sheet n of m	
Short description: Writing different documents required during TFG.	Planned start date: 28/02/2022 Planned end date: 19/06/2022	
Internal task T1: Write work plan Internal task T2: Write critical review Internal task T3: Write final memory	Deliverables: Work Plan Critical Review Final memory	Dates: 08/03/2022 14/04/2022 21/06/2022



Project: TFG	WP ref: WP3	
Major constituent: Database revision	Sheet n of m	
Short description: Perform a descriptive analysis of the patients and investigate the data and its deviation. Apply data imputation to the selected patients with missing samples.	Planned start date: 10/03/2022 Planned end date: 7/04/2022	
Internal task T1: Descriptive analysis Internal task T2: Investigate data deviation Internal task T3: Selection of individuals with sufficient information Internal task T4: Data imputation	Deliverables:	Dates:

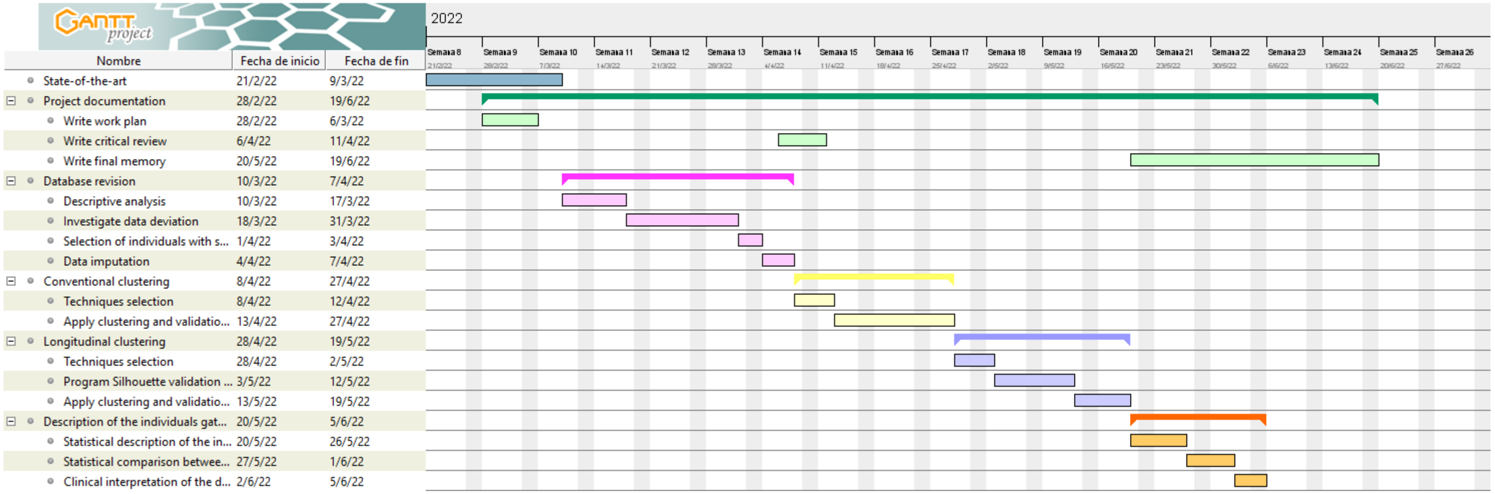
Project: TFG	WP ref: WP4	
Major constituent: Conventional clustering techniques	Sheet n of m	
Short description: Obtaining results from different conventional clustering techniques.	Planned start date: 8/04/2022 Planned end date: 27/04/2022	
Internal task T1: Techniques selection Internal task T2: Apply clustering and validity techniques	Deliverables:	Dates:

Project: TFG	WP ref: WP5	
Major constituent: Longitudinal clustering techniques	Sheet n of m	
Short description: Obtaining results from different longitudinal clustering techniques.	Planned start date: 28/04/2022 Planned end date: 19/05/2022	
Internal task T1: Techniques selection Internal task T2: Program Silhouette validity index based on Fréchet distance. Internal task T3: Apply clustering and validity techniques	Deliverables:	Dates:

Project: TFG	WP ref: WP6	
Major constituent: Description of the individuals gathered in each cluster	Sheet n of m	
Short description: Statistical description of the individuals gathered in each cluster and comparison between these groups.	Planned start date: 20/05/2022 Planned end date: 05/06/2022	
Internal task T1: Statistical description of the individuals included in each cluster considering the available information: demography, disease severity, chronic diseases, drugs Internal task T2: Statistical comparison between the groups considering the available information Internal task T3: Clinical interpretation of the differences between groups	Deliverables:	Dates:

B.2 Gantt Chart

Figure B.1: Project Gantt chart



Nombre	Fecha de inicio	Fecha de fin
State-of-the-art	21/2/22	9/3/22
Project documentation	28/2/22	19/6/22
Write work plan	28/2/22	6/3/22
Write critical review	6/4/22	11/4/22
Write final memory	20/5/22	19/6/22
Database revision	10/3/22	7/4/22
Descriptive analysis	10/3/22	17/3/22
Investigate data deviation	18/3/22	31/3/22
Selection of individuals with s...	1/4/22	3/4/22
Data imputation	4/4/22	7/4/22
Conventional clustering	8/4/22	27/4/22
Techniques selection	8/4/22	12/4/22
Apply clustering and validatio...	13/4/22	27/4/22
Longitudinal clustering	28/4/22	19/5/22
Techniques selection	28/4/22	2/5/22
Program Silhouette validation ...	3/5/22	12/5/22
Apply clustering and validatio...	13/5/22	19/5/22
Description of the individuals gat...	20/5/22	5/6/22
Statistical description of the in...	20/5/22	26/5/22
Statistical comparison betwee...	27/5/22	1/6/22
Clinical interpretation of the d...	2/6/22	5/6/22

C Programming languages and packages used during the project

Table C.1: Programming languages and packages used during the project according to different tasks

Task	Programming language	Packages used
Data pre-processing		
Descriptive analysis	R	compareGroups 4.0
Data temporal dispersion	Python	-
Data selection and preparation	Python	sklearn.impute.KNNImputer
Clustering		
Conventional clustering		
K-means with Euclidean distance	Python	sklearn.cluster.Kmeans
K-means with DTW	Python	tslearn.clustering.TimeSeriesKMeans
Longitudinal clustering		
kmlShape	R	kmlShape
Validity indices		
Conventional clustering	Python	sklearn.metrics.calinski_harabasz_score sklearn.metrics.silhouette_score
Longitudinal clustering	R	-
Description of the individuals gathered in each cluster		
Descriptive statistics	R	compareGroups 4.0