Using covariances to understand tonal coarticulation in Mandarin

Hartsink, B.

B. Hartsink

Using covariances to understand tonal coarticulation in Mandarin

Bachelor thesis

July 2023

Thesis supervisor: Dr. V. Masarotto

Leiden University
Mathematical Institute

Abstract

In tonal languages such as Mandarin Chinese, the meaning of a word depends on the pitch variation of the tone. Since tones are often not pronounced in isolation, but rather concatenated, neighboring tones effect each other. This gives rise to tonal coarticulation. In this thesis, we will explore if, given two concatenated tones of the Mandarin word "ma", it is possible to predict the following tone on the basis of the coarticulation effect present in the first tone, and vice versa. The phonetic data used for this exploration hold a certain intrinsic smoothness that points naturally towards the functional data analysis domain as a tool to study them. Therefore, we will be using multiple functional data analysis techniques. We will start with k-means clustering on the raw data with the Euclidean distance and the Manhattan distance. Afterwards, we will study the effect on tone duration, for which we will be using duration analysis. Furthermore, previous research indicates that the coarticulation effect lies at the level of covariances. Hence, we will also be clustering functional covariances. In the last section, indications of the results will be discussed and suggestions will be made for further research. Lastly, plots obtained from the analyses are shown in the appendix.

Contents

1 Introduction **2**

2 Functional data analysis **3**
2.1 What are functional data? . . . 3
2.2 Examples of functional data analysis . . . 3
2.3 Representation of functional data . . . 4
2.4 Exploring functional data with covariances . . . 5

3 The data **6**

4 K-means clustering **7**

5 Results from k-means clustering on the effect of syllable 2 on syllable 1 **8**
5.1 K-means clustering for all speakers with the Euclidean distance . . . 8
5.2 K-means clustering for one speaker with the Euclidean distance . . . 10
5.3 K-means clustering for all speakers with the Manhattan distance . . . 12

6 Results from k-means clustering on the effect of syllable 1 on syllable 2 **14**
6.1 K-means clustering for all speakers with the Euclidean distance . . . 14

7 Duration analysis **15**
7.1 Censored data . . . 15
7.2 Key quantities in duration analysis . . . 15
7.3 Cox proportional hazards model . . . 16
7.3.1 Nice properties of the Cox PH model . . . 16
7.3.2 Hazard ratio and its interpretation . . . 16
7.3.3 Confidence intervals for the hazard ratio . . . 17

8 Results from duration analysis **18**
8.1 Effect of syllable 2 on duration syllable 1 . . . 18
8.2 Effect of syllable 1 on duration syllable 2 . . . 19

9 Results from analysis of covariances (ANOVA) **19**
9.1 2-sample permutation test . . . 19
9.2 Results from clustering of functional covariances . . . 20

10 Discussion and further research **21**

A Plots **24**

1 Introduction

In many languages, pitch variation plays a significant role. In European languages, its importance arises at sentence level. For example, the sentence 'Can you imagine life without ice cream?' can either be a question or a rhetorical question, depending on the way the sentence is pronounced as a whole. In tonal languages, pitch variation plays an even more important role. This is because the modulation of the sound conveys a different meaning for the same word. In Mandarin Chinese for example, the word "ma" can have four different meanings, depending on the way it is pronounced: "mother" (flat tone), "hemp" (increasing tone), "scold" (decreasing tone) and "horse" (downward-upward tone). Since words (and tones) are not pronounced in isolation in everyday speech, but are rather concatenated to one another, each tone is affected by the neighbouring tones. This gives rise to tonal coarticulation.

The department of experimental phonetics at the Leiden University Centre for Linguistics studies the coarticulation in Mandarin. A big part of phonetic analysis is based on speech recording and focusses on modelling fundamental frequency curves, also known as F0 curves. These curves are a graphical representation of the fundamental frequency of a sound wave over time and are usually measured in hertz. Their intrinsic smoothness points naturally at the functional data analysis domain as a tool to study them.

The aim of this thesis is to get a better understanding of the coarticulation effect in Mandarin Chinese. Recent literature [1] claims that the covariance structure between pitch intensities at different frequencies can be considered "a summary of what a language sounds like". Therefore, the technique that we will employ will involve clustering of functional covariances to explore the presence of tonal clusters in a phonetic dataset.

The research questions that we are trying to answer in this thesis, are:

- Can we infer the following tone from the coarticulation effect present in the first tone?
- Can we infer the previous tone from the coarticulation effect present in the second tone?

In section 2 we will start with explaining what functional data is and how it can be represented, as well as give some examples of functional data analysis. Furthermore, we will see how functional data can be explored through covariances. In section 3, we will present the data that we have used for the analyses throughout the project, followed by an introduction to k-means clustering in section 4. We will elaborate on the results obtained from k-means clustering in sections 5 as well as 6. In section 7 we will present a different way to analyze the data, namely with duration analysis. The results for this can be found in section 8. In section 9 we will dive into the analysis of covariances and the results obtained from this. In section 10 we will discuss the results and propose our ideas for further research.

2 Functional data analysis

In this section we will start with an introduction to what functional data are. Afterwards, we will give examples of the type of data on which a functional data analysis can be performed. Also, we will show how functional data can be presented. Lastly, we will see that it is possible to explore this type of data through their covariances. The content is mostly taken from [2], unless stated otherwise.

2.1 What are functional data?

Functional data are data that consist of smooth shapes, often curves, over a continuum of time or space. From each curve we observe discrete measured values. As mentioned in [3], the simplest form in which the data can be supplied in functional data analysis is:

$$x_n(t_{j,n}) \in \mathbb{R}, t_{j,n} \in [T_1, T_2], n = 1, 2, ..., N, j = 1, ..., J_n$$

where each x_n will correspond to a curve for which we will be able to compute the value for any given $t_{j,n}$. In order to obtain the intermediate values, we will need to perform interpolation. This can be done by using a basis function system, which we will elaborate on in 2.3. In addition, if the discrete measured values at $t_{j,n}$ contain a certain amount of observational error, one will also need to perform smoothing of the obtained curves.

In the following subsection, we will give a few examples of the type of data on which we can perform functional data analysis.

2.2 Examples of functional data analysis

A popular example of functional data is found in the Berkeley Growth Study [4]. This is a study on the height growth of boys and girls (original sample 31 boys and 30 girls). The following figure can be found in [2], which contains the heights of 10 girls measured at 31 ages during this study (Tuddenham and Snyder, 1954).

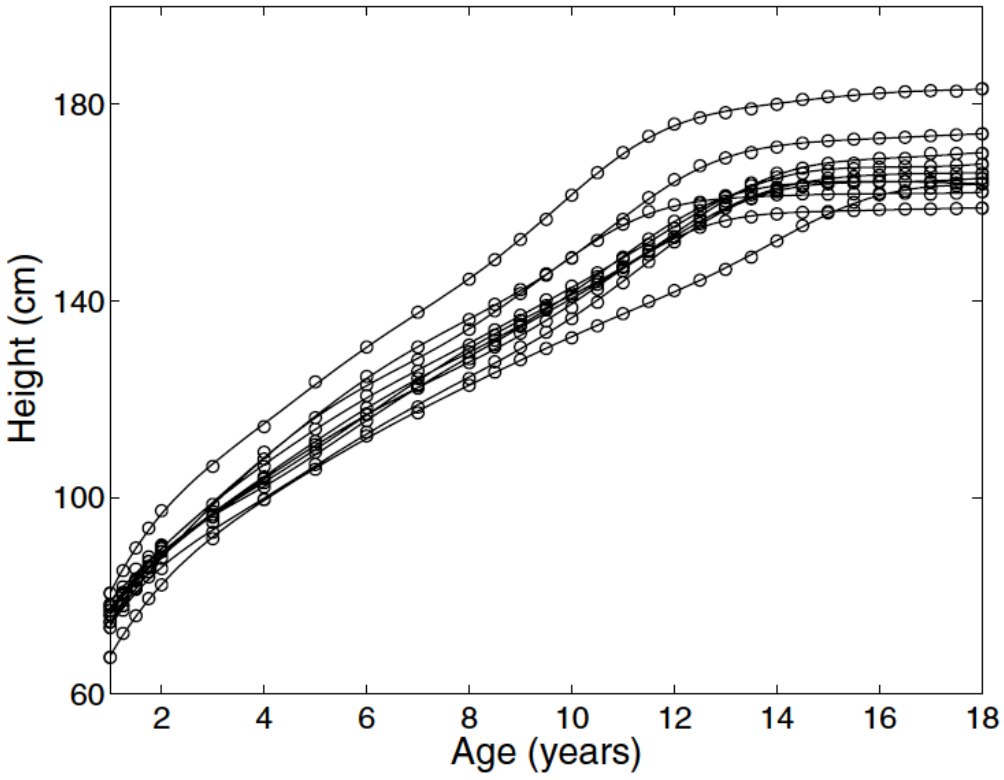

Figure 1: The heights of 10 girls measured at 31 ages. The circles indicate the unequally spaced ages of measurement.

It is even possible to use functional data analysis to analyze non-functional data. An example of this can be found in psychometrics. Many of the studies conducted in this field are based on tests that give a binary outcome. Think about tests that tell us if the participant managed to answer a particular question correctly or not. If the researchers would like to get a better understanding of the probability of success, they have to use a model that contains item response functions. The figure below, given in [2], shows three item response functions corresponding to a mathematics test taken.

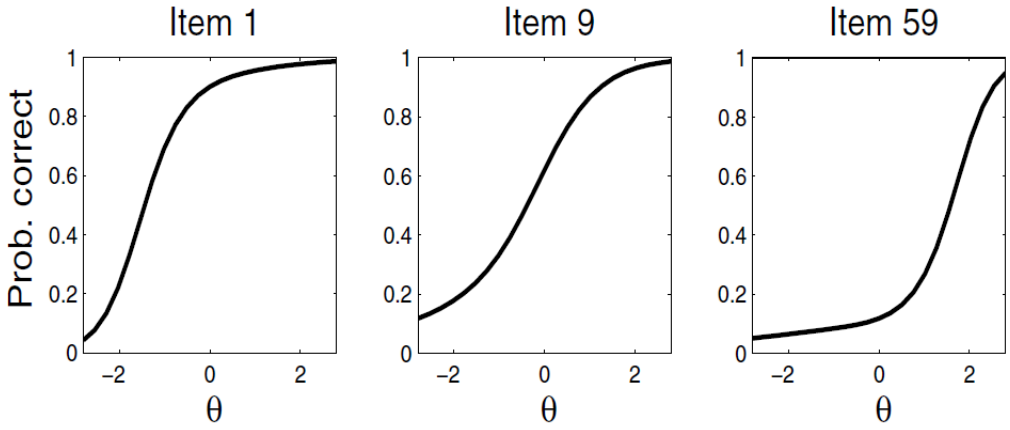

Figure 2: Each panel shows an item response function relating an examinee's position θ on a latent ability continuum to the probability of getting a test item in a mathematics test correct.

2.3 Representation of functional data

As mentioned in 2.1, we are able to perform interpolation by using a basis function system. The idea is that we take linear combinations of known mutually independent functions ϕ_k, which are called basis functions. This results in a linear expansion:

$$x(t_j) = \sum_{k=1}^{K} c_k \phi_k(t_j)$$

with c_k a coefficient corresponding to the k-th basis function and K the number of functions. If we take $K = n$ and define each ϕ_k in the following way [5]:

$$\phi_k(t_j) = \begin{cases} 1, & \text{if } j = k \\ 0, & \text{else} \end{cases}$$

it is possible to choose each c_k in such a way that this leads to $x(t_j) = y_j$ for every j, where y_j is the discrete measured value at t_j.

We will see that the data covered in this thesis are non-periodic. Splines are the most frequently used approximation system for non-periodic functional data. The B-spline basis system from de Boor (2001) is commonly used. For more information on this system, we refer to [2] and [6].

The interval over which we would like to approximate the function in question gets divided into L subintervals by means of breakpoints τ_l with $l \in \{1, 2, ..., L-1\}$. Over each of these subintervals we would like to construct a polynomial of a specified order m. These polynomials together form a spline function. Such a function meets the following requirements:

- the order is one more than the degree
- adjacent segments of the spline hold the same function values at their junction
- derivatives up to order $m-2$ of adjacent segments have the same function values at their junction

Furthermore, the number of parameters needed to estimate the model, also known as the degrees of freedom, can be calculated via the following formula:

$$df = m + L - 1$$

2.4 Exploring functional data with covariances

As mentioned in the introduction 1, earlier research has indicated that the covariance structure of phonetic data holds significant features. In this subsection we will see how to get more insight into functional data via their covariances.

In multivariate analysis, the covariance between two random variables X and Y for which $\mathbb{E}(X)$ and $\mathbb{E}(Y)$ exist, is a measure of dependence of X on Y and vice versa. It is given by [7]

$$\text{cov}(X, Y) = \mathbb{E}([X - \mathbb{E}(X)][Y - \mathbb{E}(Y)])$$

and can be rewritten in the following way

$$\text{cov}(X, Y) = \mathbb{E}(XY) - \mathbb{E}(X)\mathbb{E}(Y)$$

In functional data analysis, a similar type of object exists, namely the covariance operator. Since this object is defined on a real and separable Hilbert space, we will first explain what such a space is.

In [2] the following is stated: "A Hilbert space is a collection of objects x for which there exists:

- linear combinations $ax_1 + bx_2$
- an inner product $\langle x_1, x_2 \rangle$ for any pair x_1 and x_2
- a property called *completeness*, namely that convergent sequences of elements converge to elements within the space."

A Hilbert space that has a countable Hilbert space basis is called a separable Hilbert space [8].

We are now able to give the definition of the theoretical covariance operator as defined in [9]. Let H be a real and separable Hilbert space. Denote by $\langle \cdot, \cdot \rangle : H \times H \to \mathbb{R}$ the inner product and by $|| \cdot || : H \to [0, \infty)$ the induced norm. Let $\{X_{1,j}\}_{j=1}^{n_1}, ..., \{X_{N,j}\}_{j=1}^{n_N}$ be N independent samples of i.i.d. random elements in H, such that each of the mean functions $\mu_i = \mathbb{E}\{X_{i,j}\}$ is well-defined. Then, the covariance operators are given by

$$\Sigma_i = \mathbb{E}\{(X_{i,j} - \mu_i) \otimes (X_{i,j} - \mu_i)\}$$

with $\otimes$ the outer product on H.

In practice, we use the empirical covariance operator (also known as the covariance function) to estimate the theoretical covariance operator. Given functional observations $x_i(t)$ with $i \in \{1, 2, ..., N\}$ and their mean function $\bar{x}(t)$, the empirical covariance operator of a pair of (time) points $(t1, t2)$ is defined as follows [2]:

$$\text{cov}_X(t_1, t_2) = (N-1)^{-1} \sum_{i=1}^{N} \{x_i(t_1) - \bar{x}(t_1)\}\{x_i(t_2) - \bar{x}(t_2)\}$$

One might also come across the definition with the term $(N-1)^{-1}$ replaced by N^{-1}. These definitions are equivalent, since the difference in outcome will be very small and therefore neglegible. As we will see in 9, we can construct a covariance matrix for each tonal combination by evaluating the covariance function at each pair of time points (t_1, t_2). Such a matrix is the finite-dimensional analog of the covariance operator [10] and will be useful in determining the existence of differences between the curves.

3 The data

The data that we consider in this project come from Prof. dr. Y. Chen, from the Leiden University Centre for Linguistics. Her webpage can be found at `https://www.universiteitleiden.nl/en/staffmembers/yiya-chen#tab-1`. The dataset contains bi syllabic speech recordings containing two sequentially concatenated 'ma' sounds from 12 native Mandarin Chinese speakers. Since there exist four lexical tones in Mandarin [11], there are 16 different tonal combinations in total. We denote the four different tones as follows: $T1$ corresponds to the flat tone ('mother'), $T2$ to the increasing tone ('hemp'), $T3$ to the downward-upward tone ('horse') and $T4$ to the decreasing tone ('scold'). The figure below shows the shape of the four lexical tones. The research question that we try to answer in this project, is whether we can predict the following tone on the basis of hearing the first tone.

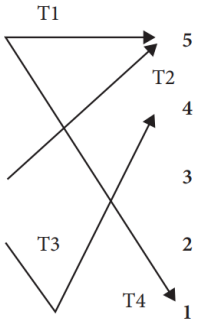

Figure 3: The four tones of Mandarin Chinese. Source: [11].

The curves are sampled and normalized over 20 points in time. The time points $1, ..., 10$ correspond to syllable 1 (the first syllable) and the time points $11, ..., 20$ correspond to syllable 2 (the second syllable). Furthermore, the experiment has been repeated 4 times. However, for some of the speakers, not all of the repetitions are present in the data.

In the study for which the data was originally gathered, the researchers investigated the influence of a cognitive load on the curves by assigning a mnemonic task to each speaker. Therefore, a distinction is made between the curves with cognitive load (denoted by CL6) and without cognitive load (denoted by CL0). In this project we are not interested in the cognitive load, so we mostly work with the curves without cognitive load, unless stated otherwise. In the figure below we have plotted the F0 curves against time for all speakers during the first repetition.

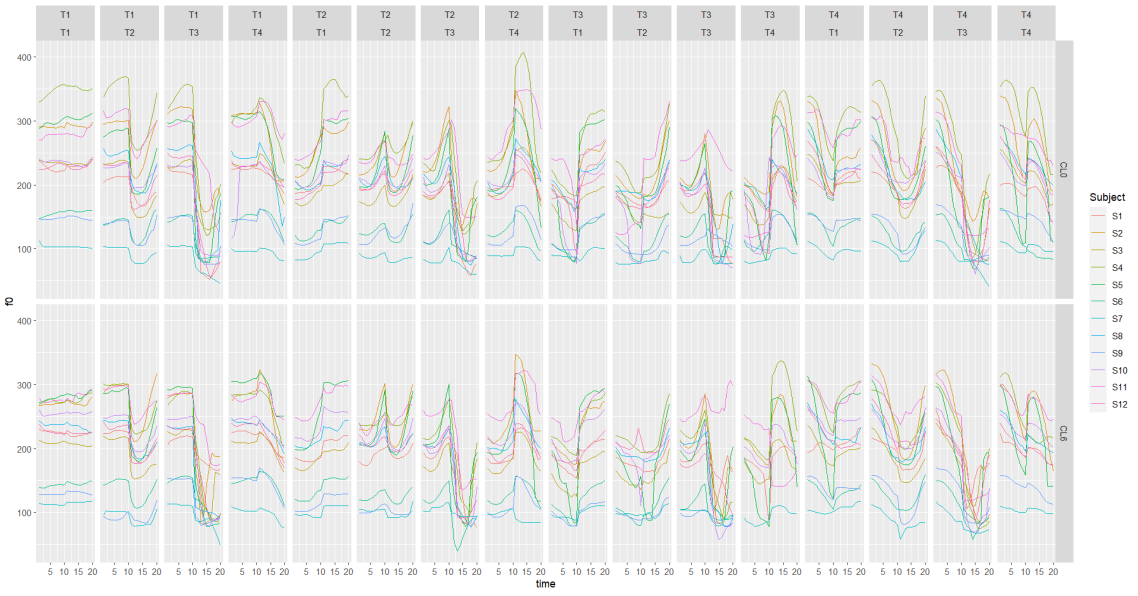

Figure 4: The frequency profiles for all speakers during the first repetition against time.

Focussing on the curves in the CL0 row, we see quite some variation between the speakers. Although this is to be expected, we have to keep this in mind while analyzing the data. Other things that could cause noise is the recording equipment and the (small) amount of data. For the frequency profiles during the other repetitions, and per speaker, we refer to A. Due to the smooth nature of the curves, the data naturally fit into the FDA context.

4 K-means clustering

Our first attempt to analyze the data is seeing if it is possible to classify similar data together, in the hope that all first syllables followed by the same second syllable will be grouped together. The technique that we are going to employ for this is clustering, and more specifically, k-means clustering, which we will introduce in this section. The content is mostly taken from [2] and [12], unless stated otherwise.

Clustering is a widely used unsupervised learning method in which a dataset is partitioned into groups, such that similar data points are contained within the same group and dissimilar ones are separated. A successful clustering analysis involves choosing a suitable clustering approach. In this thesis, we will be focussing on k-means clustering, which is one of the most popular approaches out there.

K-means clustering is a clustering method for which the user is able to specify the number of clusters, denoted by k. Each of these clusters corresponds to a specific centroid and is made out of objects such as points, vectors, matrices (which is the case in this project, as we will see later) or curves that were assigned to this centroid as the closest one. There are several ways to implement this method. As stated in [13], Lloyd's algorithm is one of the most popular heuristics for this.

At the start, k centroids are selected from the dataset. Afterwards, the other vectors get assigned to the centroid for which the distance is the smallest. Lloyd's algorithm uses the squared Euclidean distance as the dissimilarity measure. Let x_i and $x_{i'}$ be two vectors from a Euclidean $p-$space. The squared Euclidean distance between these two vectors is defined as follows [12]:

$$d(x_i, x_{i'}) = \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = ||x_i - x_{i'}||^2$$

The clusters should be constructed in such a way that the distance between a vector and the centroid in its cluster is minimal. This gives rise to minimizing the following criterion, which is known as the within-point scatter:

$$\begin{aligned} W(C) &= \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} d(x_i, x_{i'}) \\ &= \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} ||x_i - x_{i'}||^2 \end{aligned}$$

where $C(i)$ is the cluster of the $i-$th observation.

Minimizing the within-point scatter ensures that the clusters are obtained in such a way that the between-cluster point scatter $B(C)$ is being maximized, since these two functions are related through the following constant:

$$T = W(C) + B(C)$$

which is known as the total point scatter. After assignment of the points, a new centroid gets computed for each cluster as the mean of the data points in that particular cluster. Since it is the mean, it is not a point that is actually contained in the dataset. This is one of the differences between k-means clustering and k-medoids clustering. For the reader that is interested to read more about the latter, we refer to [12]. The assignment of points to its closest centroid and the computation of new centroids continues until all the centroids remain the same or if none of the points get assigned to a different cluster in the following iteration.

Another way to implement the k-means method is with the Hartigan and Wong (1979) algorithm. The kmeans() function from the stats package uses this algorithm by default. Therefore, we will be using the function with this method. For who is interested to read how this algorithm works in detail, we refer to [14] for an extensive description of the algorithm.

5 Results from k-means clustering on the effect of syllable 2 on syllable 1

In order to see if the effect of syllable 2 on syllable 1, if it even exists, is captured in the raw data, we have performed k-means clustering for:

- all speakers with the Euclidean distance
- one speaker with the Euclidean distance
- all speakers with the Manhattan distance

All of these analyses have been done for all tonal combinations and all repetitions, unless stated otherwise. In this section we will elaborate on the above analyses. Besides that, we will examine the results on the basis of the plot for $T1Tx$, where $x \in \{1, 2, 3, 4\}$. One may assume that the plots for the other tonal combinations contain similar results as the one for $T1Tx$, unless stated otherwise. In the latter case, we will discuss these plots in detail too. For the other tonal combinations, we refer to A.

5.1 K-means clustering for all speakers with the Euclidean distance

The first analysis entails k-means clustering for all speakers based on their F0 curves. Since the first syllable is fixed and the second syllable varies, it is only necessary to consider the curves evaluated at the first 10 time points. For the clustering we have used the kmeans() function from the stats package from [15]. This function partitions the points from a given data matrix into k clusters, where the number k is given by the user. Since the partitioning is done by minimizing the sum of squares between the points and the mean of the cluster, this means that the distance measure used is the Euclidean distance. As mentioned in 4, one of the ways in which k-means clustering can be implemented is with the Hartigan-Wong algorithm. Since the kmeans() function uses this algorithm by default, we have used this particular algorithm during our analysis. To plot the clusters, we have used the function fviz_cluster() from [16]. As stated in the rdocumentation, this function uses principal component analysis to create two principal components if the number of variables is greater than two. As stated in [17], "Principal component analysis (PCA) is the problem of fitting a low-dimensional affine subspace to a set of data points in a high-dimensional space." The first component needs to have the largest possible variance [18]. The data are then plotted conforming to these components.

As there are four different tones, one would initially expect $k = 4$ to be a suitable number of clusters to choose in order to see some effect. Therefore, we began the analysis with $k = 4$. The plot containing the clusters for the tonal combinations $T1Tx$, where $x \in \{1, 2, 3, 4\}$, can be found in the figure below. Each data point is labeled by its tonal combination, followed by its speaker and repetition.

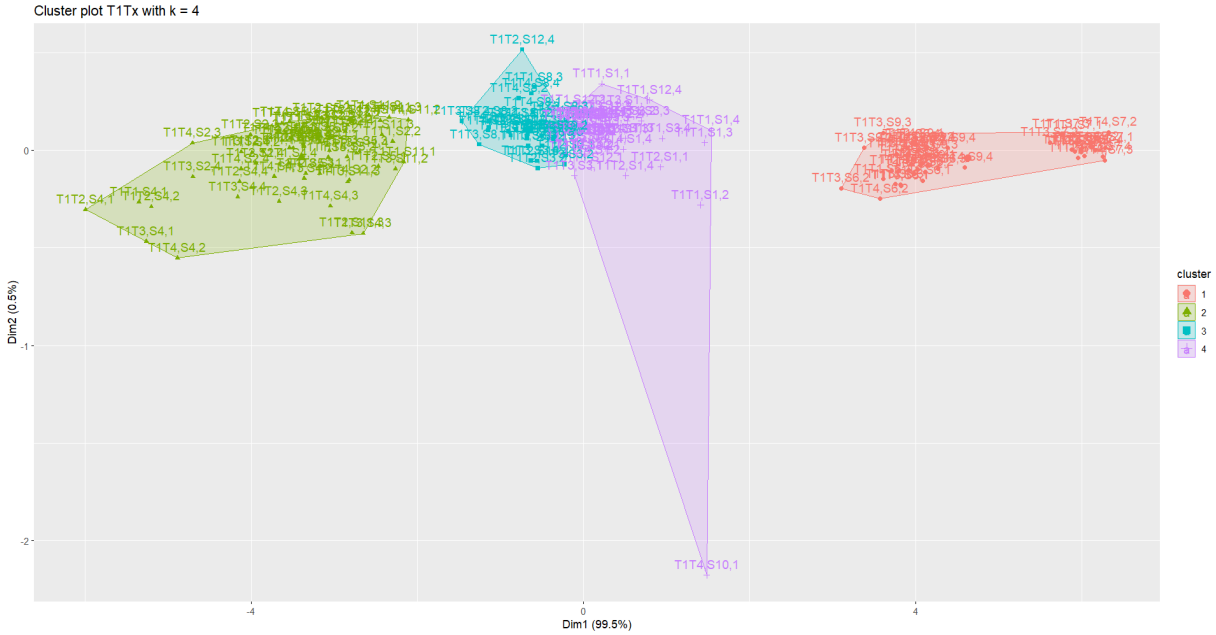

Figure 5: Clusters for $T1Tx$ with the Euclidean distance using the kmeans() function with $k = 4$ and where all speakers are considered.

From the figure we see that the tonal combinations are distributed in such a way that each cluster contains all of the combinations in a large amount. Therefore, there is not one cluster that corresponds to a specific tonal combination. Since the same holds for the plots of the other tonal combinations, the effect of coarticulation seems to play no role in this classification.

To see if there might be an effect if we decrease the number of clusters, we have performed the same analysis for $k = 3$. If we compare the plot of $T1Tx$ for this analysis to the one for $k = 4$, we see that two of the clusters have been merged in order to get the imposed number of clusters. The other two clusters remain the same. Similar results hold for the other tonal combinations. Therefore, the results are non-significant for $k = 3$ as well.

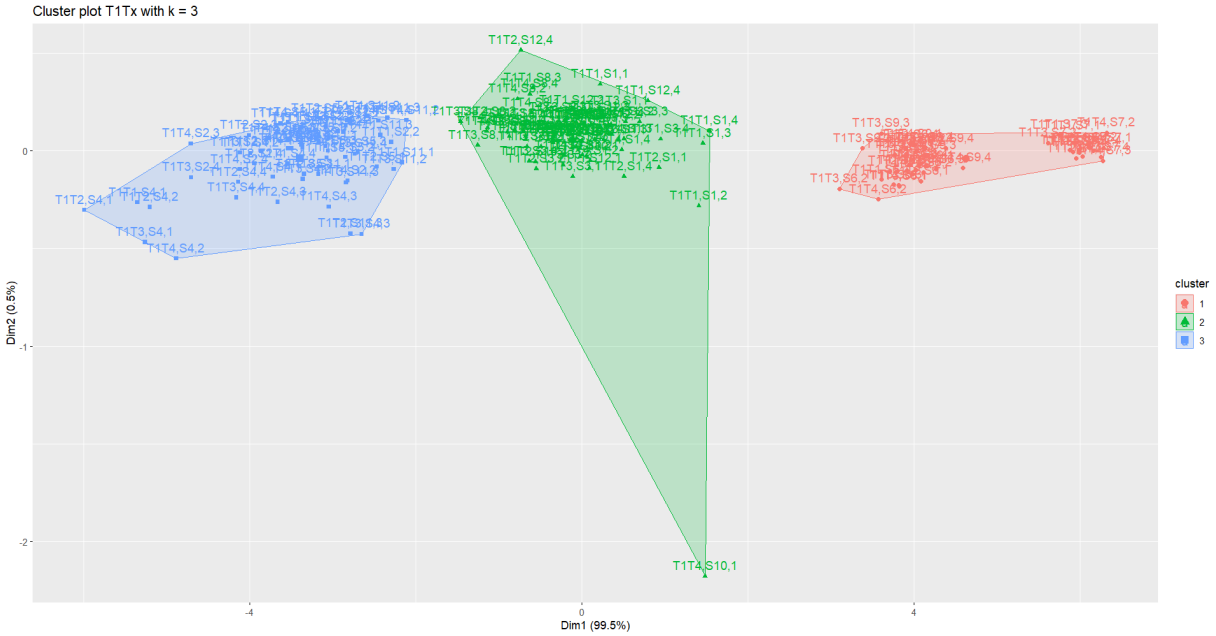

Figure 6: Clusters for $T1Tx$ with the Euclidean distance using the kmeans() function with $k = 3$ and where all speakers are considered.

In order to see if the lack of effect might be caused by a suboptimal number of clusters chosen, we have used the function fviz_nbclust() from [16] to determine and visualize the optimum number of clusters to select. We haven chosen the within cluster sums of squares method for this, which is the default method for this function. The below figure shows the result for $T1Tx$.

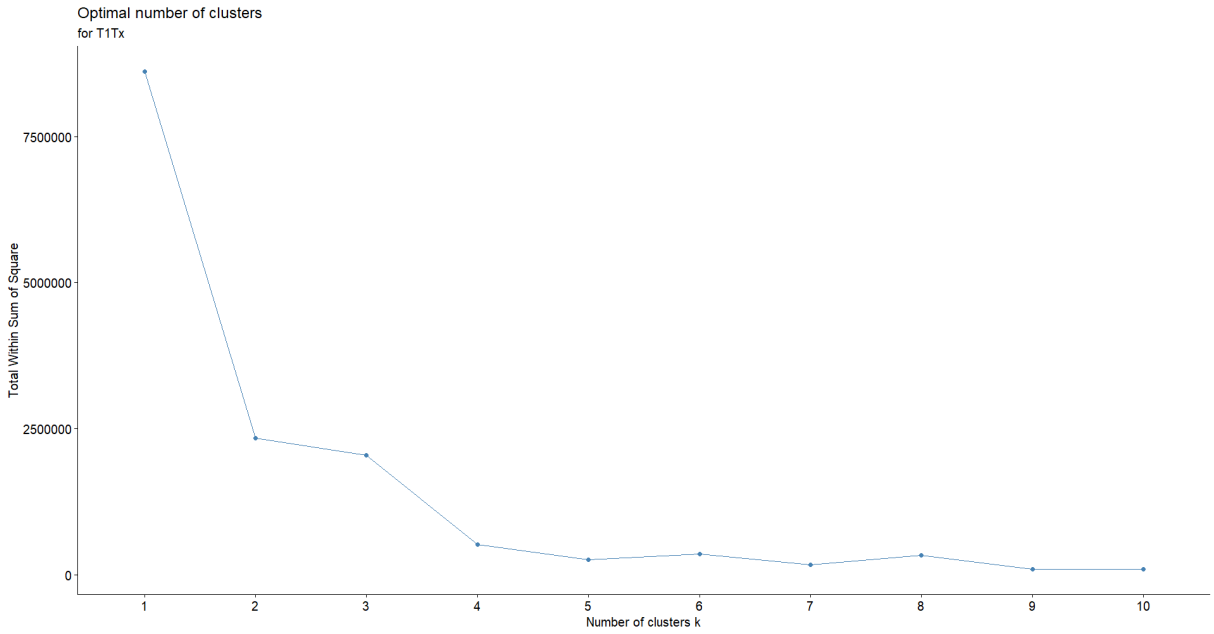

Figure 7: Optimal number of clusters for the tonal combination $T1Tx$.

To determine the optimum from this figure, we have used the so-called elbow method. As used in [19], the optimum is there where the graph makes an "elbow bend". The figure above indicates that this is at $k = 2$ or $k = 3$. The same holds for the other tonal combinations, as one can find in A. So, the optimum does not differ drastically from the initial number of clusters that we have chosen. Therefore, it is less likely that the lack of effect is caused by the number of clusters chosen and could rather be found in the noise / pattern of the data.

5.2 K-means clustering for one speaker with the Euclidean distance

One of the reasons why we might not detect an effect is due to noise that is overwhelming the differences in coarticulation. This noise can be caused by the recording equipment, differences between repetitions or differences between speakers. We can reduce the amount of noise by focussing on only one speaker for the k-means cluster analysis with the Euclidean distance.

The analysis that we will be discussing in this subsection is similar to the one that we have seen in the previous one 5.1, except now we only consider speaker 1. Apart from the fact that there is no missing data for this speaker (all curves for all four repetitions are available), there is no other particular reason why we have chosen this speaker. Although results are likely to vary between speakers, we do not expect this to happen in such a great amount that we detect a difference in (in)significance of the results. Below is the plot of the analysis.

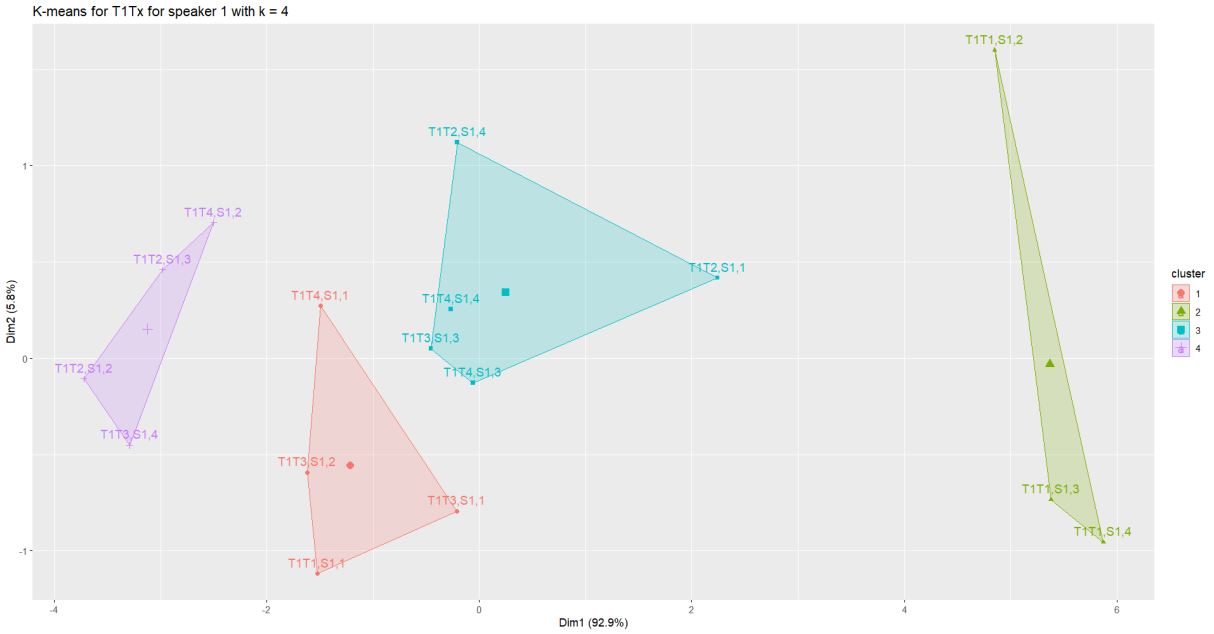

Figure 8: Clusters for $T1Tx$ for speaker 1 with the Eulidean distance using the kmeans() function with $k = 4$.

Here, the tonal combination $T1T1$ seems to represent the rightmost cluster. This indicates that there might be an effect of the second syllable on the first syllable if both syllables are tone $T1$. We expect this effect to be stronger if the data would have been less noisy.

We have also plotted the clusters for $T3Tx$ below, since it appears to differ from the plots for the other tonal combinations.

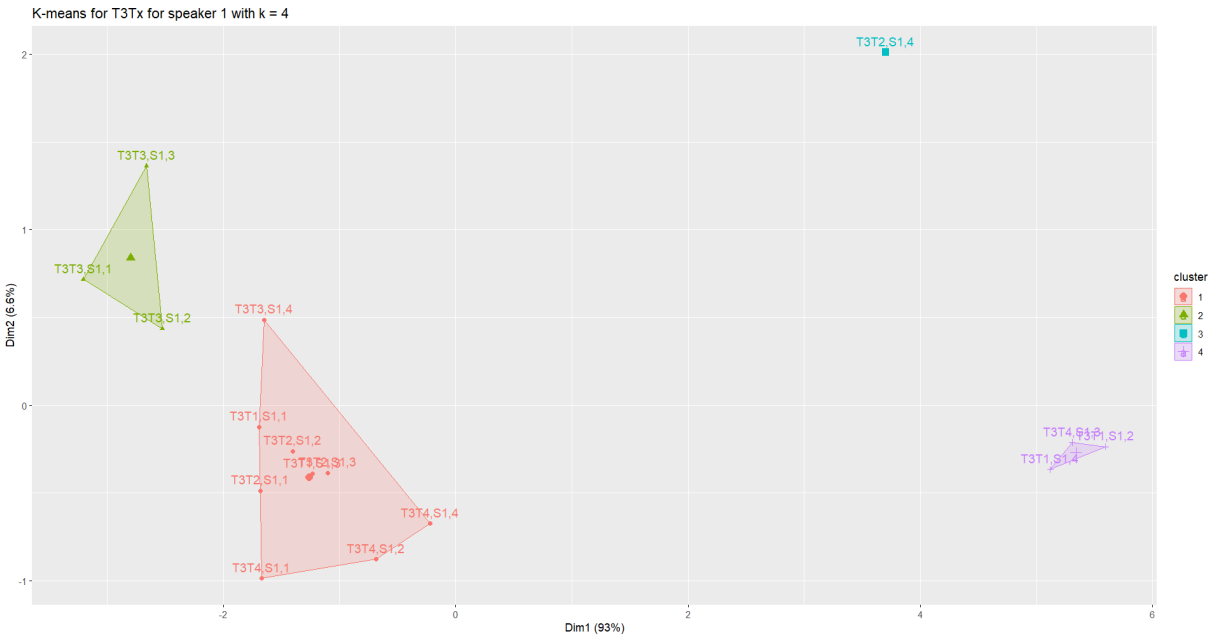

Figure 9: Clusters for $T3Tx$ for speaker 1 with the Eulidean distance using the kmeans() function with $k = 4$.

As we can see, most of the $T3T3$ combinations are contained within the leftmost cluster. This indicates that there might be an effect of the second syllable on the first syllable if both syllables are tone $T3$. Again, we expect this effect to be stronger if the data would have been less noisy. This potential effect is interesting, since this particular tone appears to capture more variation in the data than the other tones.

Therefore, one would initially not expect this tone to be (more or less) isolated from the other tones.

5.3 K-means clustering for all speakers with the Manhattan distance

Up till now we have only seen k-means clustering for the Euclidean distance. We are curious to see if the results differ if we take a different distance for the clustering. Since the kmeans() function from the stats package does not allow other distances than the Euclidean distance, we have used the KMeans() function from [20] for this. To exclude whether a drastic difference in results is caused by a difference in implementation between these two functions for the k-means algorithm rather than a different choice for the distance, we have first compared the results of the two functions with the Euclidean distance.

Figure 10: Clusters for $T1Tx$ for speaker 1 with the Euclidean distance using the KMeans() function with $k = 4$.

As we can see, the KMeans() function from [20] produces similar clusters for all tonal combinations as we have seen for the kmeans() function from the stats package 5.1. Therefore, we continue with this new function with a different distance than the Euclidean distance, namely the Manhattan distance. The reason for this choice is that it is one of the predefined distances for this function. The Manhattan distance between two points (x_1, y_1) and (x_2, y_2) is defined as in [21]:

$$d_t = |x_2 - x_1| + |y_2 - y_1|$$

The plot for $T1Tx$ is given below.

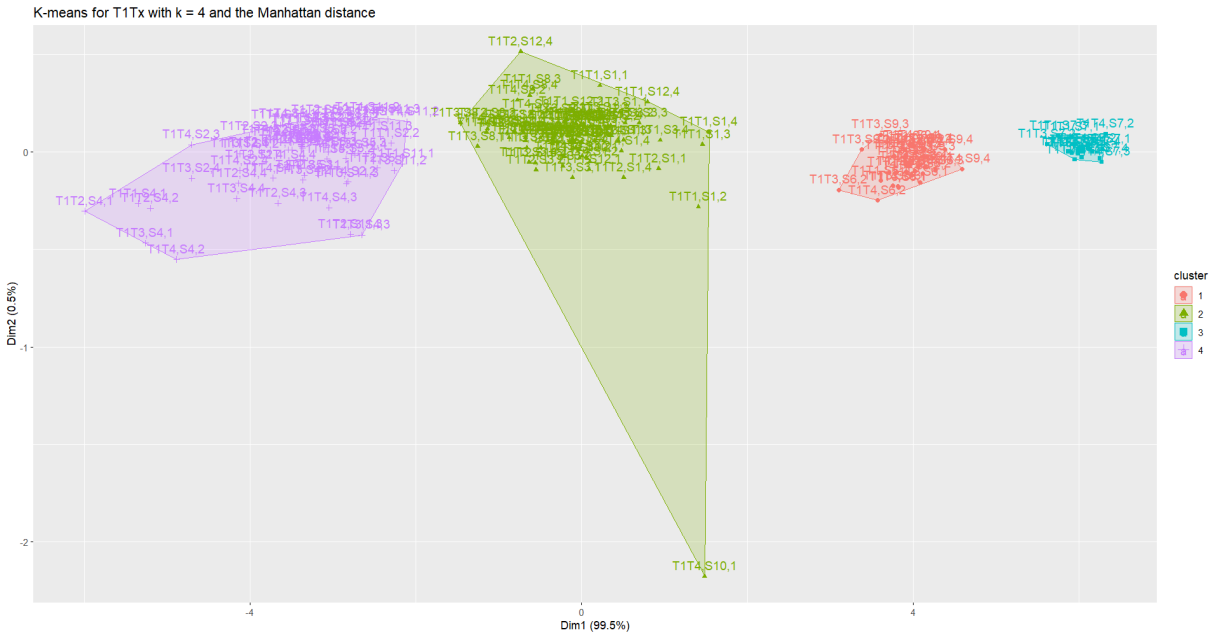

Figure 11: Clusters for $T1Tx$ with the Manhattan distance using the KMeans() function and where all speakers are considered.

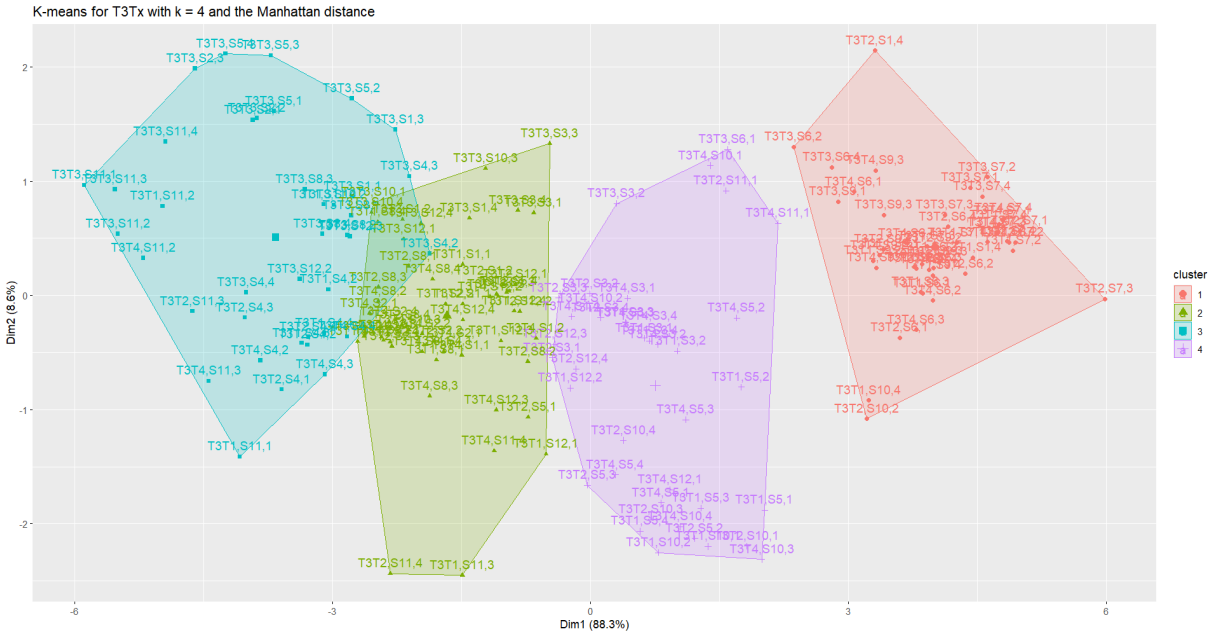

Figure 12: Clusters for $T3Tx$ with the Manhattan distance using the KMeans() function and where all speakers are considered.

The difference between this plot and the one obtained for the Euclidean distance (with the KMeans() function) lies in the formation of the three rightmost clusters. The two clusters in the middle of the first plot have emerged into one cluster in the second plots. Also, the rightmost cluster of the first plot has been disintegrated into two clusters in the second plot. However, this different formation of the clusters still does not give us enough insight to make a statement about the effect we are investigating.

As for the plot of $T3Tx$, we see an overlap between the two leftmost clusters. Looking at the individual curves might shed light on what commonality causes the overlapping. However, since this is not relevant to our research question, we will not continue to analyze this.

Furthermore, time constraint for the thesis and the lack of promising results from clustering have brought us to move towards different analysis methods. However, it is worth keeping in mind that clustering is metric dependent. Thus, for future work, different metrics that are more refined and less linear could be more insightful than the metrics that we have used here.

6 Results from k-means clustering on the effect of syllable 1 on syllable 2

6.1 K-means clustering for all speakers with the Euclidean distance

Since we are also interested in the (existence) of the opposite effect, we have also performed k-means clustering where the second syllable is fixed and the first syllable varies. This time, we only need to consider the last 10 time points. The rest of the analysis is analogue to the one where the first syllable is fixed and the second syllable varies. Below is the plot that we have obtained for $T1Tx$ with $k = 4$.

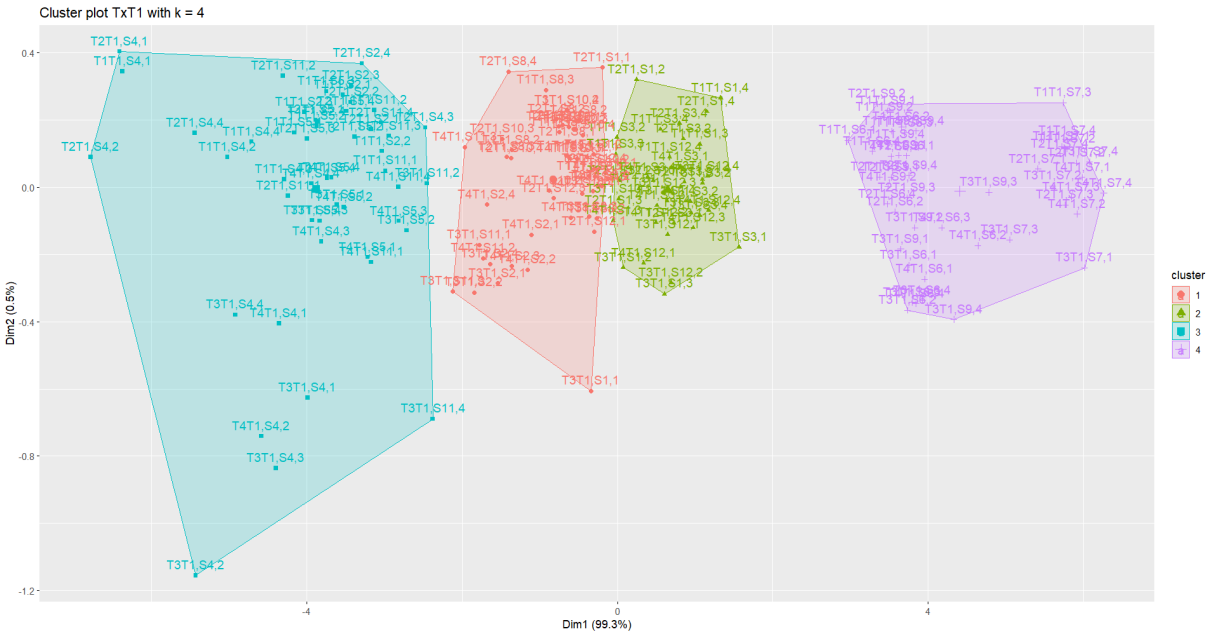

Figure 13: Clusters for $TxT1$ with the Euclidean distance using the kmeans() function with $k = 4$ and where all speakers are considered.

Again, we see that the tonal combinations are more or less evenly distributed over all the clusters. Therefore, none of the clusters represents a specific tonal combination. Since these results are similar to the ones obtained for the reversed order, we do not expect to see much of a difference if we perform the other analyses on this order as well. Therefore, we have omitted these analyses here.

7 Duration analysis

One of the ways that we can measure the effect of one syllable on the other is by studying their durations. This gives rise to the following two questions:

- What is the effect of syllable 2 on the duration of syllable 1?
- What is the effect of syllable 1 on the duration of syllable 2?

We can try to answer these questions with duration analysis. Although duration analysis is better known as survival analysis, the former term is more suitable in the context of this thesis. As stated in [22], "Survival analysis is the study of survival times and of the factors that influence them." Therefore, the outcome variable of such an analysis is the time until a specified event occurs. An example of such an event is the death of a patient in a clinical study.

In the following subsections we will be covering the type of data encountered in duration analysis, the key characteristics of duration analysis and the Cox proportional hazards model. The content is mostly taken from [23], unless stated otherwise.

7.1 Censored data

The data that we consider in a duration analysis are often censored. This type of data can be described as in [23]: "In essence, censoring occurs when we have some information about individual survival time, but we don't know the survival time exactly." In the case of a clinical trial, where the subjects of interest are the participants of the trial, the most common reasons for censoring are:

- The study ends before the event of interest occurs for a specific subject.
- Lost to follow-up of a subject.
- Withdrawal from the study by a subject.

Censoring can appear in the following ways:

- Left censoring: occurs when the true survival time is less than or equal to the observed survival time.
- Interval censoring: occurs when the true survival time lies within a known interval, which has been specified.
- Right censoring: occurs when the true survival time is equal to or greater than the observed survival time.

Most of the data that is considered in duration analysis is right-censored data. This type of censoring generally occurs if one of the three reasons that were earlier described happens.

7.2 Key quantities in duration analysis

One of the quantities that is considered in duration analysis is the survival function (also known as the survivor function). This is the probability that the subject of interest lasts (survives) longer than a particular time t. Therefore, the definition is as follows:

$$S(t) = P(T > t)$$

where the random variable $T \geq 0$ denotes the survival time of the subject. The survival function holds the following theoretical properties: it is non-increasing, $S(0) = 1$ and $S(\infty) = 0$.

Another quantity that we consider is the hazard function:

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t \mid T \geq t)}{\Delta t}$$

This function is mathematically difficult to interpret. The following conceptual interpretation is given in [23]: "The hazard function $h(t)$ gives the instantaneous potential per unit time for the event to occur, given that the individual has survived up to time t." Loosely put, the hazard function gives an indication of the failure rate for a particular subject, given that it has survived up to time t.

The relationship between the survival function and the hazard function can be captured in the following two formulae:

$$S(t) = e^{-\int_0^t h(u)du},$$

$$h(t) = -\left[\frac{dS(t)/d(t)}{S(t)}\right]$$

7.3 Cox proportional hazards model

The questions stated at the beginning of this section are of the form 'How does a particular independent variable influence a particular dependent variable?' These type of questions can be studied through regression analysis. One popular survival regression model is the Cox regression model, also known as the Cox proportional hazards model. For this model, the hazard function at time t is expressed as the following product:

$$h(t, X) = h_0(t)e^{\sum_{i=1}^{p} \beta_i X_i}$$

where t is the time, $h_0(t)$ is the baseline hazard function, $X = (X_1, X_2, ..., X_p)$ are the p covariates (also known as the explanatory variables) and β_i the corresponding regression coefficients. The term $h_0(t)$ is called the baseline hazard function for the following reason. If all the X_i's are equal to zero, we get:

$$h(t, X) = h_0(t)e^{\sum_{i=1}^{p} \beta_i X_i} = h_0(t)e^0 = h_0(t)$$

In the following subsections we will see how the hazards can be used.

7.3.1 Nice properties of the Cox PH model

The Cox PH model has some nice properties that often makes it preferred over other duration analysis models. Some of them include:

- It is a semiparametric model. This means that the baseline hazard is unspecified, while the model is still capable of producing results that approximate the results of the correct parametric model well.
- The outcome of the estimated hazards are always non-negative due to the exponential expression. This non-negativity is important, since the following should hold for every hazard function: $0 \leq h(t, X) < \infty$.
- It uses survival times and censoring, which is not the case for every model. Therefore, it uses more information than for example the logistic model, which ignores survival times and censoring.

Another nice property of this model is that it holds the proportional hazards assumption. To understand what this assumption entails, we will first need to cover the concept of a hazard ratio.

7.3.2 Hazard ratio and its interpretation

The hazards of two subjects can be compared through their hazard ratio, which in the case of the Cox PH model can be estimated as follows:

$$\hat{HR} = \frac{\hat{h}(t, X^*)}{\hat{h}(t, X)} = \frac{\hat{h_0}(t)e^{\sum_{i=1}^{p} \hat{\beta}_i X_i^*}}{\hat{h_0}(t)e^{\sum_{i=1}^{p} \hat{\beta}_i X_i}}$$

with $X = (X_1, X_2, ..., X_p)$ the covariates of one subject and $X^* = (X_1^*, X_2^*, ..., X_p^*)$ the covariates of the other subject. This formula can be rewritten in such a way that the effect of the covariates can be measured without knowing the baseline hazard:

$$\hat{HR} = \frac{\hat{h_0}(t)e^{\sum_{i=1}^{p} \hat{\beta}_i X_i^*}}{\hat{h_0}(t)e^{\sum_{i=1}^{p} \hat{\beta}_i X_i}} = \frac{e^{\sum_{i=1}^{p} \hat{\beta}_i X_i^*}}{e^{\sum_{i=1}^{p} \hat{\beta}_i X_i}} = e^{\sum_{i=1}^{p} \hat{\beta}_i (X_i^* - X_i)}$$

This equation is directly related to the proportional hazards assumption, which, as stated in [23], holds the following: "The PH assumption requires that the HR is constant over time, or equivalently, that the hazard for one individual is proportional to the hazard for any other individual, where the proportionality constant is independent of time." This assumption can be achieved by only accepting covariates that are time-independent. In the case that the covariates are time-dependent, one needs to use the extended Cox model. We recommend reading chapter 6 of [23] if one is interested in this extension as well. The covariates that we consider in this thesis do not change over time, so the non-extended version suffices.

The outcome of this hazard ratio can be loosely interpreted as follows:

- $\hat{HR} > 1$ indicates that the failure rate for the subject with X^* as the covariates is greater than for the subject with X as the covariates, given that both subjects have survived up to time t.
- $\hat{HR} < 1$ indicates that the failure rate for the subject with X^* as the covariates is greater than for the subject with X as the covariates, given that both subjects have survived up to time t.

We are also able to compare the hazard of a subject with X as the covariates to the baseline hazard. For this, we need to rewrite the first equation as:

$$\hat{HR} = \frac{\hat{h(t, X)}}{\hat{h_0(t)}} = e^{\sum_{i=1}^{p} \hat{\beta}_i X_i}$$

The loose interpretation of the outcome of this ratio is similar to the one that we have seen before:

- $\hat{HR} > 1$ indicates that the failure rate for subject 1 is greater than for the control subject, given that they have both survived up to time t.
- $\hat{HR} < 1$ indicates that the failure rate for subject 1 is smaller than for the control subject, given that they have both survived up to time t.

In this project, the hazard ratio can be used to compare the duration of two syllables and the interpretation in this context is as follows:

- $\hat{HR} > 1$ indicates that the duration of the second syllable is longer than the duration of the first syllable.
- $\hat{HR} < 1$ indicates that the duration of the second syllable is shorter than the duration of the first syllable.

7.3.3 Confidence intervals for the hazard ratio

In the previous subsection we have seen how to estimate the hazard ratio. A 95% confidence interval (so $\alpha = 0.05$) for such a hazard ratio and for which the model does not contain any interaction effects, is given as in [23]:

$$\exp\left[\hat{\beta} \pm 1.96\sqrt{\hat{\text{Var}}\hat{\beta}}\right]$$

The value 1.96 is obtained by reading of the $z-$table for the Normal distribution at $z = 1 - \frac{0.05}{2} = 0.9750$. Consequently, if we choose the significance level to be $\alpha = 0.10$ instead, we are able to construct a 90% confidence interval for this type of model by reading of the $z-$table at $z = 0.9495$ (since this value is nearest to and smaller than $1 - \frac{0.10}{2} = 0.95$). From the table we obtain $P(Z < 0.9495) = 1.64$, so the 90% confidence interval is:

$$\exp\left[\hat{\beta} \pm 1.64\sqrt{\hat{\text{Var}}\hat{\beta}}\right]$$

8 Results from duration analysis

To get a better understanding of the effect of syllable 1 on the duration of syllable 2 and vice versa, we have performed duration analyses. The data that we consider contain the duration for syllable 1 as well as for syllable 2 in milliseconds for each possible tonal combination, each speaker and each repetition. We have used the coxph() function from [24] for this analysis, which is based on the Cox Proportional-Hazards model described in 7.3. The reference tone for both analyses is $T1$. In the following subsections we will present the results on both duration analyses.

8.1 Effect of syllable 2 on duration syllable 1

The table below contains the results on the effect of syllable 2 on the duration of syllable 1.

	coef	exp(coef)	se(coef)	z	Pr(>|z|)
duration$syllable2S2T2	-0.001915	0.998087	0.074189	-0.026	0.979
duration$syllable2S2T3	-0.122258	0.884920	0.074342	-1.645	0.100
duration$syllable2S2T4	0.034389	1.034987	0.074352	0.463	0.644

	exp(coef)	exp(-coef)	lower .95	upper .95
duration$syllable2S2T2	0.9981	1.0019	0.8630	1.154
duration$syllable2S2T3	0.8849	1.1300	0.7649	1.024
duration$syllable2S2T4	1.0350	0.9662	0.8946	1.197

Figure 14: Results from the duration analysis with the Cox Proportional-Hazards model on the effect of syllable 2 on the duration of syllable 1.

From the table we obtain the following estimates for $\exp(\beta)$:

- If syllable 2 is $T2$, it holds that $\exp(\hat{\beta}) < 1$.
- If syllable 2 is $T3$, it holds that $\exp(\hat{\beta}) < 1$.
- If syllable 2 is $T4$, it holds that $\exp(\hat{\beta}) > 1$.

This means that $\hat{HR} < 1$ if syllable 2 is $T2$ or $T3$. This indicates that in these two cases, the duration of syllable 2 is shorter than the duration of syllablle 1 (which is $T1$), given that both syllables have lasted at least until time t. Furthermore, if syllable 2 is $T4$, we have that $\hat{HR} > 1$. In this case, it is indicated that the duration of syllable 2 is longer than the duration of syllable 1 (which again is $T1$).

The table also shows the 95% confidence intervals for the $\exp(\beta)$ values. If syllable 2 is $T3$, this interval is equal to $[0.7649, 1.024]$. Note that the larger part of this interval is below 1, since $\frac{0.7649+1.024}{2} = 0.89445$. For this tone we also have that $\exp(\hat{\beta}) = 0.8849 < 1$. From the table it follows that the p-value for this tone is $p = 0.100$. This means that we do not reject the null hypothesis if $\alpha = 0.05$. However, we have to keep in mind that the noise and small amount of data have an effect on the significance of the results. Therefore, we think that $\alpha = 0.10$ would be a more suitable significance level. For the same tone, the 90% confidence level then is equal to $[0.7833, 0.9997]$ and this way it lies completely below 1. This means that we are 90% sure that the real hazard ratio is below 1 in this case. This might indicate the existence of an effect. Since the p-value is now equal to the significance level, we interpret this result as slightly significant. If the data were larger and less noisy, we would expect a stronger effect for this tone.

8.2 Effect of syllable 1 on duration syllable 2

The table below contains the results on the effect of syllable 1 on the duration of syllable 2.

	coef	exp(coef)	se(coef)	z	Pr(>|z|)
duration$syllable1S1T2	0.037820	1.038544	0.074262	0.509	0.611
duration$syllable1S1T3	0.027741	1.028130	0.074373	0.373	0.709
duration$syllable1S1T4	0.003397	1.003403	0.073694	0.046	0.963

	exp(coef)	exp(-coef)	lower .95	upper .95
duration$syllable1S1T2	1.039	0.9629	0.8979	1.201
duration$syllable1S1T3	1.028	0.9726	0.8887	1.189
duration$syllable1S1T4	1.003	0.9966	0.8685	1.159

Figure 15: Results from the duration analysis with the Cox Proportional-Hazards model on the effect of syllable 1 on the duration of syllable 2.

If syllable 1 is tone $T2, T3$ or $T4$, it follows from the table that $\exp(\hat{\beta}) > 1$. This indicates that in each of these cases, the duration of syllable 1 is longer than the duration of syllable 2 (which is tone $T1$). Since a greater part of the values of each of the 95% confidence intervals lies above 1 and $\exp(\hat{\beta}) > 1$ and $p >> 0.05$ for all the tones, there is most likely no effect for this analysis.

9 Results from analysis of covariances (ANOVA)

If there exist differences between the curves, they can be studied by computing and clustering their covariances. As we have seen in 2.4, the covariance matrix is a finite dimensional analog of the covariance operator. Each entry in such a matrix is the covariance function evaluated at a particular pair of time points. For the analysis on the effect on syllable 2 on syllable 1, we only need to consider the first 10 time points. Hence, the covariance matrices computed for this analysis are 10×10 matrices and can be found in A.

The covariances need to differ enough in order to gain insight from clustering. In the following subsection we will give a N-sample permutation test that can be used to test whether there exist differences between the covariances at all and if so, how big these differences are. Also, we are now trying to employ some specific FDA techniques in an attempt to analyze the curves in their entirety.

9.1 2-sample permutation test

In order to see if it is useful to analyze the computed covariance operators any further, testing the equality of these covariances is essential. In this subsection, we will focus on the test that we have used for this. Since we want to perform the test without any parametric assumption on the sample, we compute the p-value via a $N-$sample permutation test, where N is the number of independent groups for which we want to test their covariances against one another. Here, the test statistic is inspired by optimal transport and is given by some suitable sum of optimal transport maps. For more details on this we refer to [25].

We have chosen $N = 2$ for our analysis, because this is faster and easier compared to $N > 2$. This gives rise to the following three null hypotheses and their alternative hypotheses:

$$H_0 : \{\Sigma_1 = \Sigma_2\},\ H_1 : \{\Sigma_1 \neq \Sigma_2\}$$

$$H_0 : \{\Sigma_1 = \Sigma_3\},\ H_1 : \{\Sigma_1 \neq \Sigma_3\}$$

$$H_0 : \{\Sigma_1 = \Sigma_4\},\ H_1 : \{\Sigma_1 \neq \Sigma_4\}$$

where Σ_j denotes the covariance operator of the tonal combination $T1Tj$, for $j \in \{1, 2, 3, 4\}$.

For general N, the procedure for the permutation test is as follows.

- Reassign the $\left(\sum_{j=1}^{N} n_j\right)$ curves $\{X_{i,j},\ i = 1, \ldots, n_j,\ j = 1, \ldots, N\}$ into N groups, respecting the sizes of the initial groups. Call these new "data" $X^*_{i,j}$. Note that n_j, which denotes the number of curves for the $j-$th independent group, is at most $12 \times 4 = 48$ (12 speakers, 4 repetitions and some missing data).

- Construct the empirical covariance $\hat{\Sigma}^*_j$ for the jth group $\{X^*_{i,j}\}_{i=1}^{n_j}$, $j = 1, \ldots, N$.

- Compute the empirical (weighted) Fréchet mean $\hat{\Sigma}^*$ of $\{\hat{\Sigma}^*_1, \ldots, \hat{\Sigma}^*_N\}$.

- Construct

$$\hat{\mathbf{t}}^*_j = (\hat{\Sigma}^*)^{-1/2}\big((\hat{\Sigma}^*)^{1/2}\hat{\Sigma}^*_j(\hat{\Sigma}^*)^{1/2}\big)^{1/2}(\hat{\Sigma}^*)^{-1/2}$$

  and compute

$$T^*_r = \sum_{j=1}^{N} n_j \|\hat{\mathbf{t}}^*_j - \mathscr{I}_{q\times q}\|^2_r$$

Iterating this procedure for all possible re-assignments of the indexes gives the distribution of the permuted statistics T^*_r, which in turn can be used to generate a p-value for T_r under the null hypothesis. Under H_0, all possible permutations of the operators labels have equal probability $p = 1/N!$. Obtaining an exact test would thus require $N!$ permutations of the labels, making it computationally prohibitive for large N. Therefore, for general N, rather than computing an exact p-value, one could resort to a Monte Carlo sample of permutations. Since in our case N is small ($N = 2$), the Monte Carlo sampling method is not needed. However, because we want to perform three different tests simultaneously on the same data, we will need to use a multiple-comparison correction in order to maintain the original significance level of the entire set, which is $\alpha = 0.05$. We will be using a Bonferroni correction for this, which entails dividing the original α by the number of comparisons that are made [26]. In our case, the Bonferroni correction gives us the following significance level for each comparison: $\frac{0.05}{3} = 0.0167$.

When we run the 2-sample permutation test in R, we obtain the same $p-$value for all three comparisons, namely $p = 0.0099$, so $p < 0.0167$ for each comparison. Therefore, we reject the null hypothesis and accept the alternative hypothesis of each test. We conclude that the covariance operators probably differ enough from each other to attempt to analyze them via clustering. In the following subsection we will present the results obtained from this.

9.2 Results from clustering of functional covariances

The figure below shows the PCA plot that we have obtained from the cluster analysis and by applying multidimensional scaling techniques. Σ_{CiRj} corresponds to the computed covariance operator of the tonal combination $T1Ti$ and repetition j and $\overline{\Sigma}_{Ci}$ corresponds to the barycenter of the i-th obtained cluster, with $i, j \in \{1, 2, 3, 4\}$. Initially we expected that by splitting the curves per tonal combination per repetition, we would see that the difference between different tonal combinations would be greater than the difference between different repetitions of the same tonal combination. However, the clusters are hard to see in the plot. Some of the Σ_{CiRj} are quite close to a certain barycenter, but not enough to have a clear image of the cluster that they belong to. We expect that this as well is caused by noise that is overpowering the results.

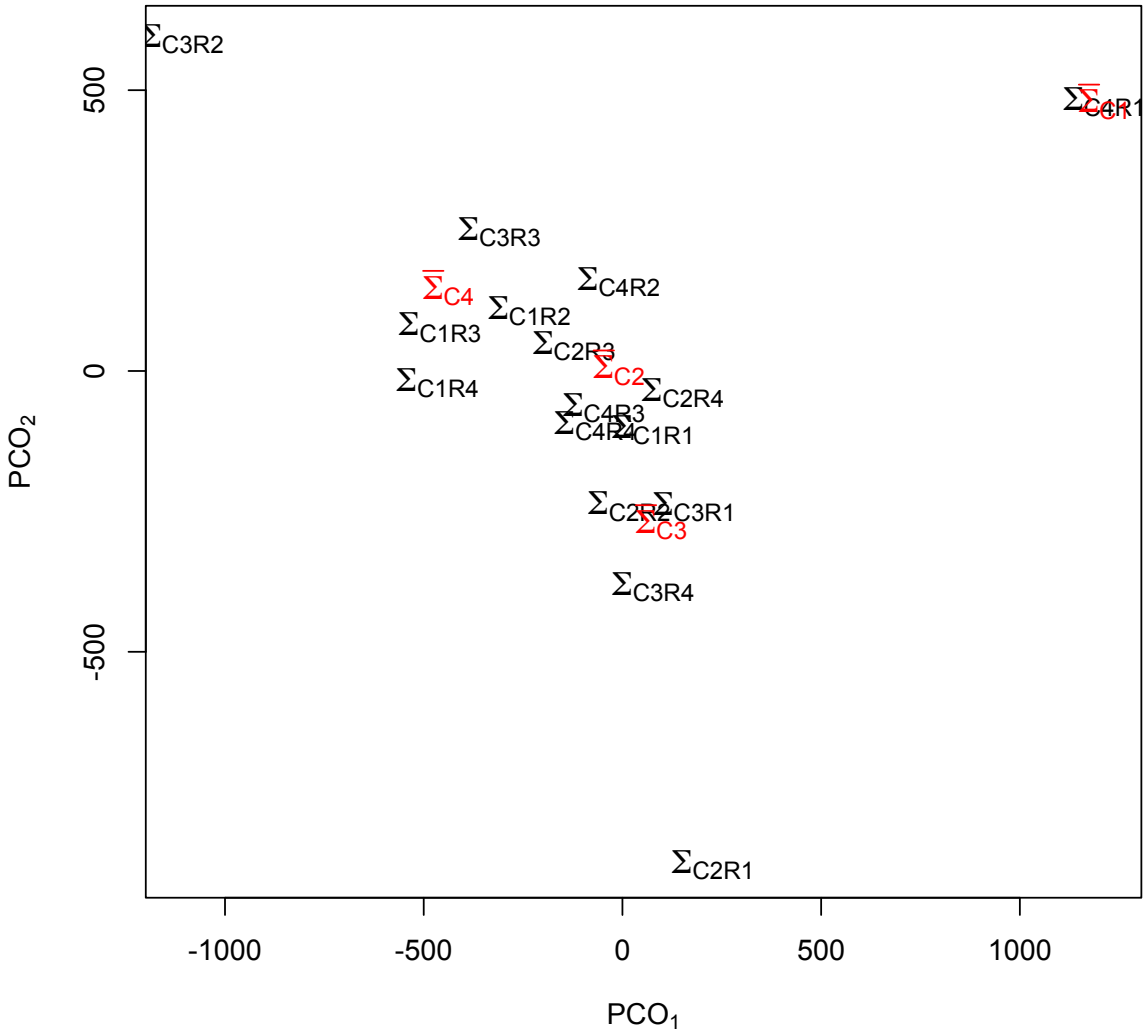

Figure 16: PCA plot containing the covariance operators and the barycenters of the clusters.

10 Discussion and further research

In this section we will discuss the results that we have obtained throughout the project and offer suggestions for further research. This thesis is born from real research questions on a real dataset provided by Prof. dr. Y. Chen. The specific dataset used eventually did not seem suitable for a precise analysis, as it is very noisy. However, we could get some indications from the results. It seems to be that the duration of a tone is different if this tone is followed by tone $T3$, but this difference is likely to be nested at the covariance level. All of this makes the methods employed promising to apply in a further research, in which a larger and cleaner dataset should be used. It could also be insightful to do the k-means clustering analyses that we have done in sections 5 and 6 for metrics that are more refined and less linear. Furthermore, we suggest to use different clustering methods, specifically for the covariance operators. Lastly, if the differences were more clear, further analysis of the differences between speakers and between repetitions could also be interesting.

References

[1] John A. D. Aston, Jeng-Min Chiou, and Jonathan P. Evans. Linguistic Pitch Analysis Using Functional Principal Component Mixed Effect Models. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 59(2):297–317, 10 2009.

[2] J. Ramsay and B.W. Silverman. *Functional Data Analysis*. Springer Series in Statistics. Springer New York, 2006.

[3] Piotr Kokoszka and Matthew Reimherr. *Introduction to functional data analysis*. CRC press, 2017.

[4] Harold E. Jones and Nancy Bayley. The berkeley growth study. *Child Development*, 12(2):167–173, 1941.

[5] John Maidens. An illustrated guide to interpolation methods. `https://maidens.github.io/jekyll/update/2016/08/10/An-illustrated-guide-to-interpolation-methods.html#:~:text=The%20fundamental%20building%20blocks%20of,k%CF%86k(x).`, 2016.

[6] Carl De Boor. On calculating with b-splines. *Journal of Approximation theory*, 6(1):50–62, 1972.

[7] Geoffrey Grimmett and Dominic JA Welsh. *Probability: an introduction*. Oxford University Press, 2014.

[8] Philippe Blanchard and Erwin Brüning. *Mathematical methods in Physics: Distributions, Hilbert space operators, variational methods, and applications in quantum physics*, volume 69. Birkhäuser, 2015.

[9] V. Masarotto and G. Masarotto. Covariance-based soft clustering of functional data based on the wasserstein-procrustes metric, 2022.

[10] Arthur Gretton. Notes on mean embeddings and covariance operators, 2015.

[11] Jeroen Van de Weijer and Marjoleine Sloos. The four tones of mandarin chinese: Representation and acquisition. *Linguistics in the Netherlands*, 31(1):180–191, 2014.

[12] T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2009.

[13] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine Piatko, Ruth Silverman, and Angela Y Wu. The analysis of a simple k-means clustering algorithm. In *Proceedings of the sixteenth annual symposium on Computational geometry*, pages 100–109, 2000.

[14] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

[15] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2022.

[16] Alboukadel Kassambara and Fabian Mundt. *factoextra: Extract and Visualize the Results of Multivariate Data Analyses*, 2020. R package version 1.0.7.

[17] René Vidal, Yi Ma, and S Sastry. Generalized principal component analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1–15, 2005.

[18] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[19] Mengyao Cui et al. Introduction to the k-means clustering algorithm based on the elbow method. *Accounting, Auditing and Finance*, 1(1):5–8, 2020.

[20] Antoine Lucas. *amap: Another Multidimensional Analysis Package*, 2022. R package version 0.8-19.

[21] Kevin P. Thompson. The nature of length, area, and volume in taxicab geometry, 2011.

[22] Dirk F Moore. *Applied survival analysis using R*, volume 473. Springer, 2016.

[23] David G Kleinbaum and Mitchel Klein. *Survival analysis a self-learning text*. Springer, 1996.

[24] Terry M Therneau. *A Package for Survival Analysis in R*, 2022. R package version 3.4-0.

[25] Valentina Masarotto, Victor M. Panaretos, and Yoav Zemel. Transportation-based functional anova and pca for covariance operators, 2022.

[26] Eric W Weisstein. Bonferroni correction. *https://mathworld. wolfram. com/*, 2004.

A Plots

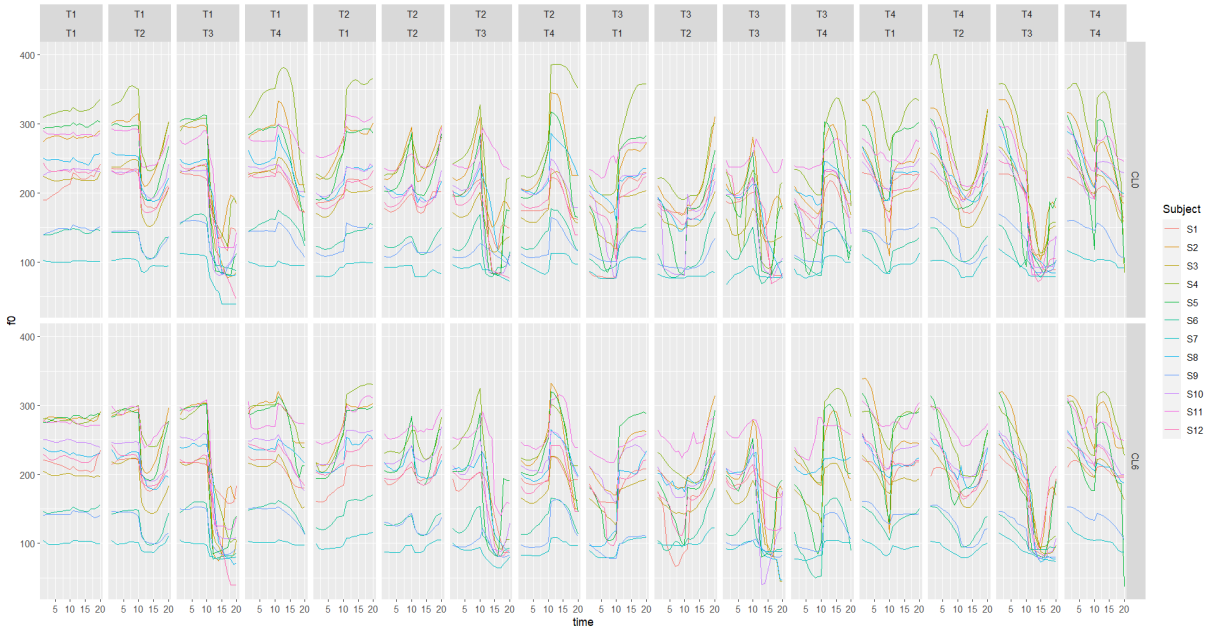

Figure 17: Frequency profile repetition 2.

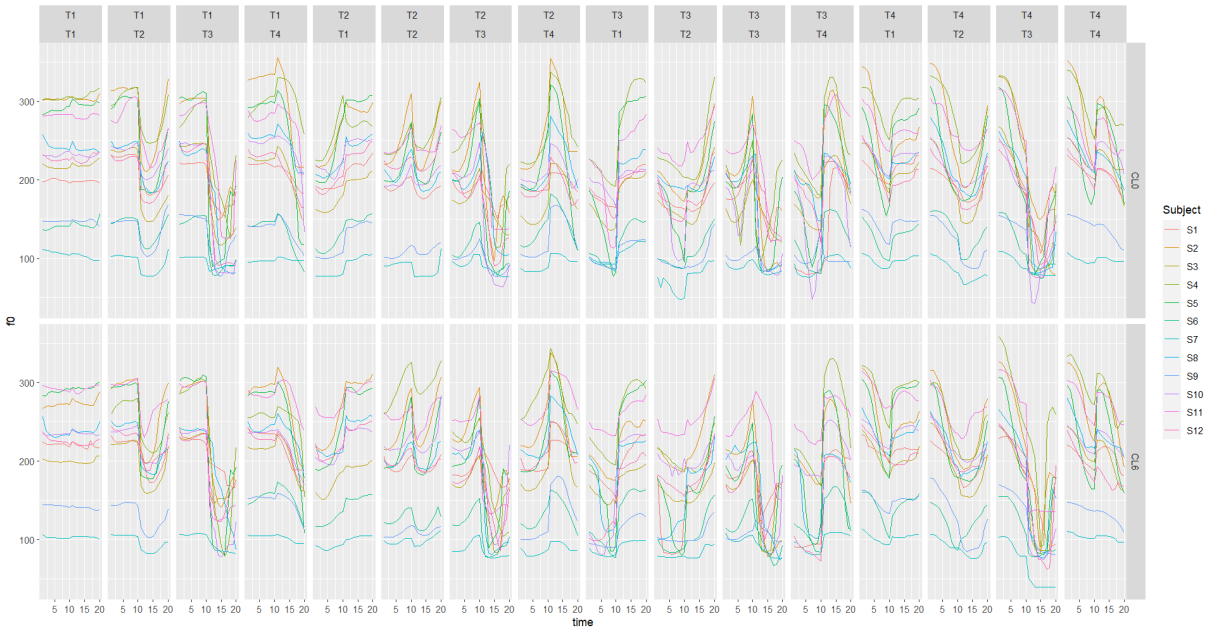

Figure 18: Frequency profile repetition 3.

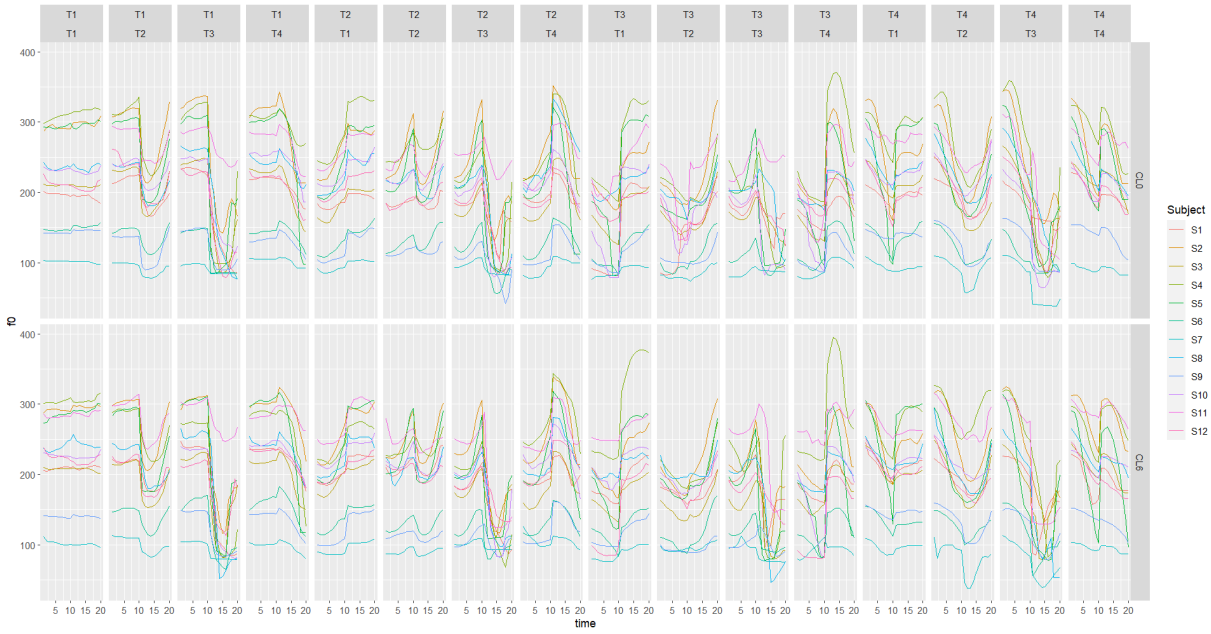

Figure 19: Frequency profile repetition 4.

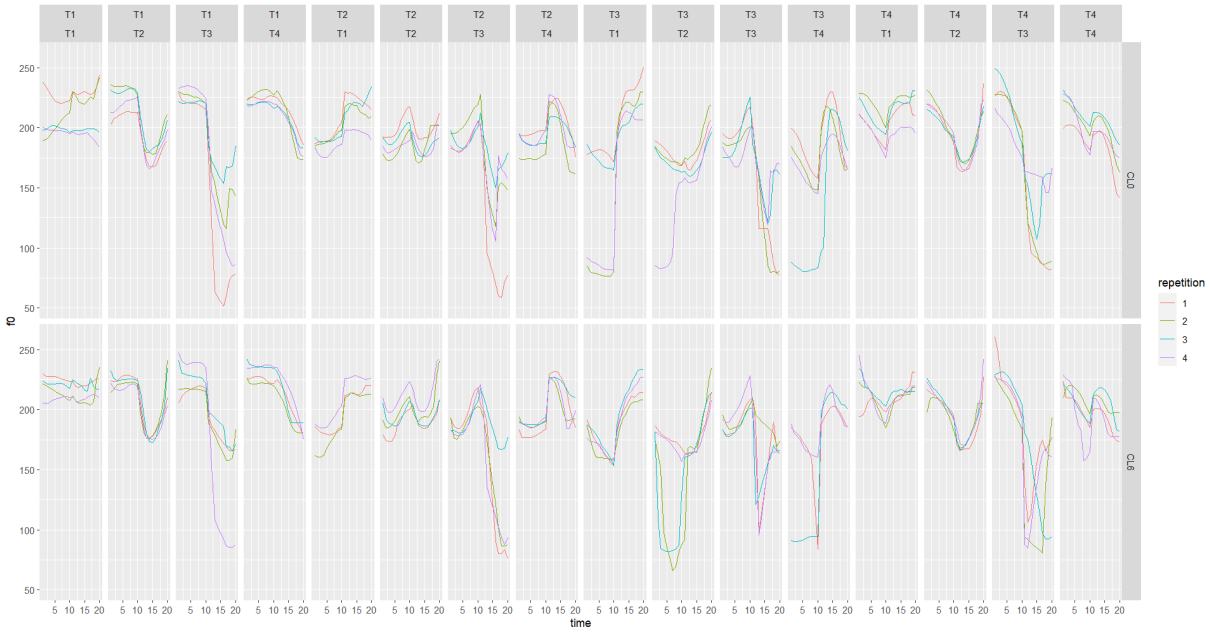

Figure 20: Frequency profile speaker 1.

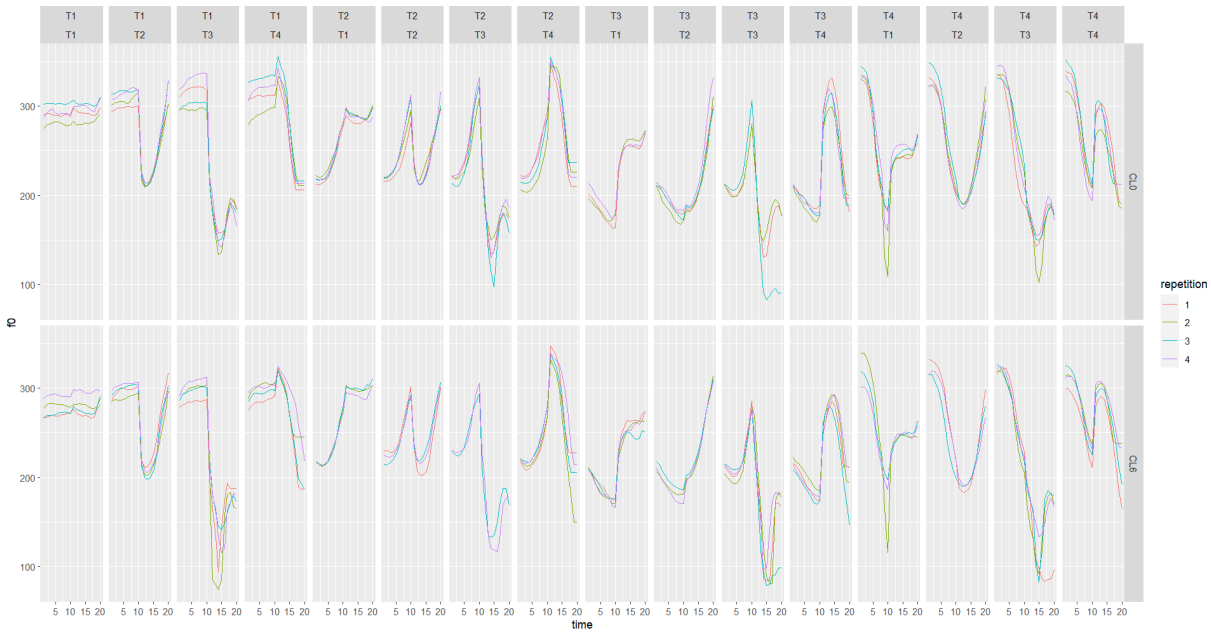

Figure 21: Frequency profile speaker 2.

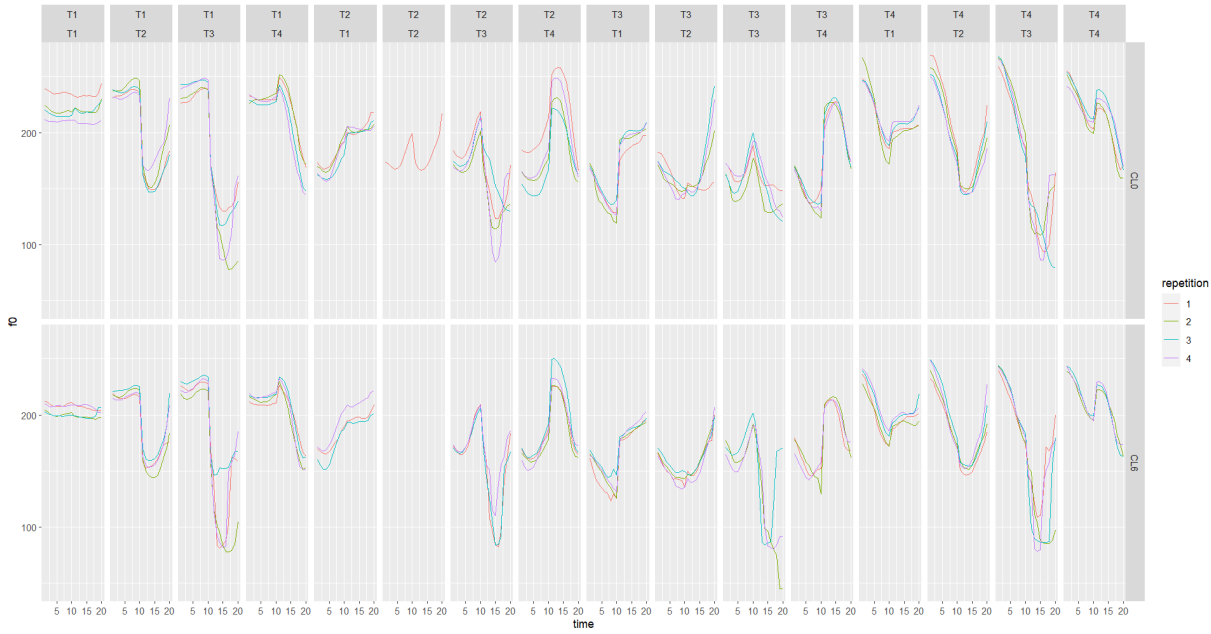

Figure 22: Frequency profile speaker 3.

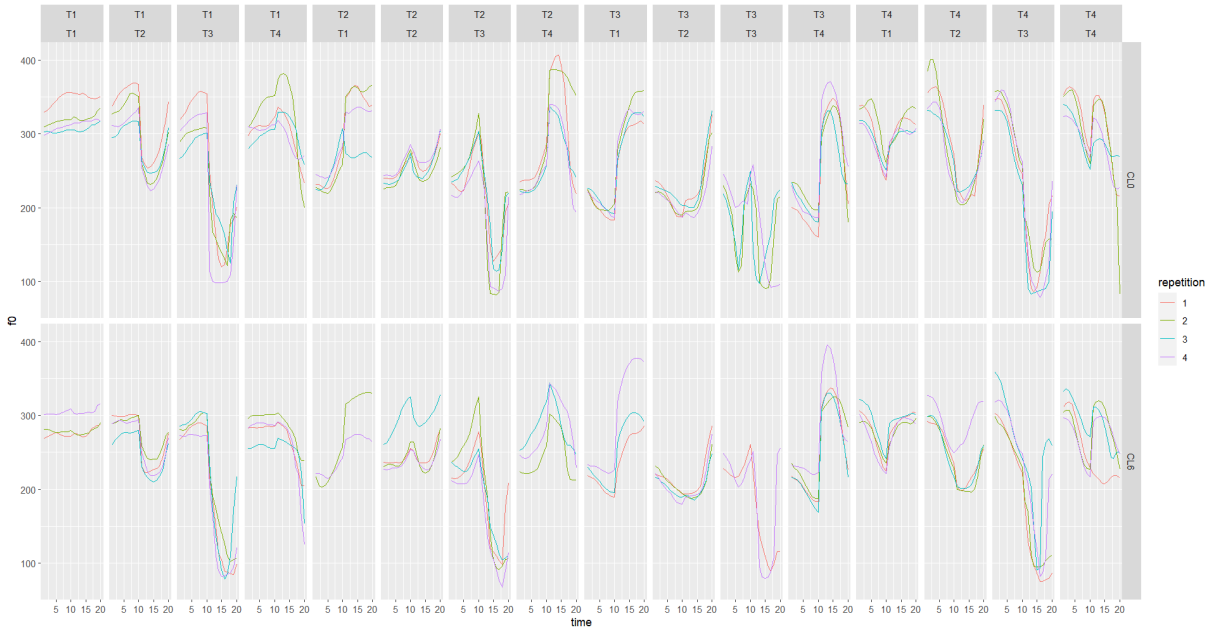

Figure 23: Frequency profile speaker 4.

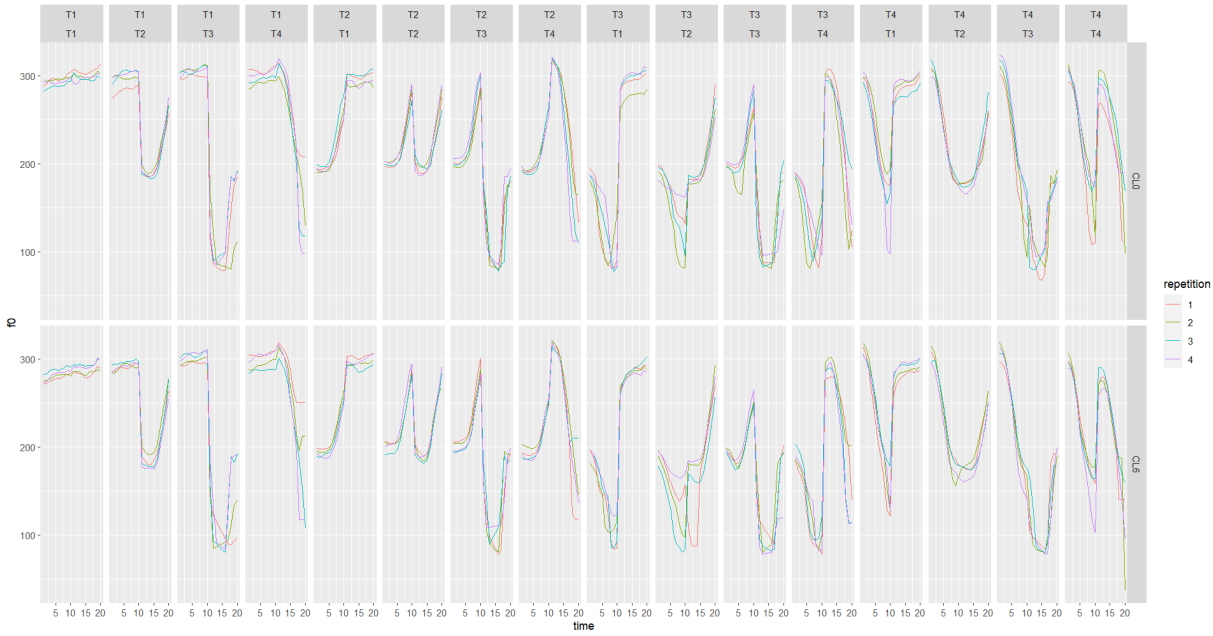

Figure 24: Frequency profile speaker 5.

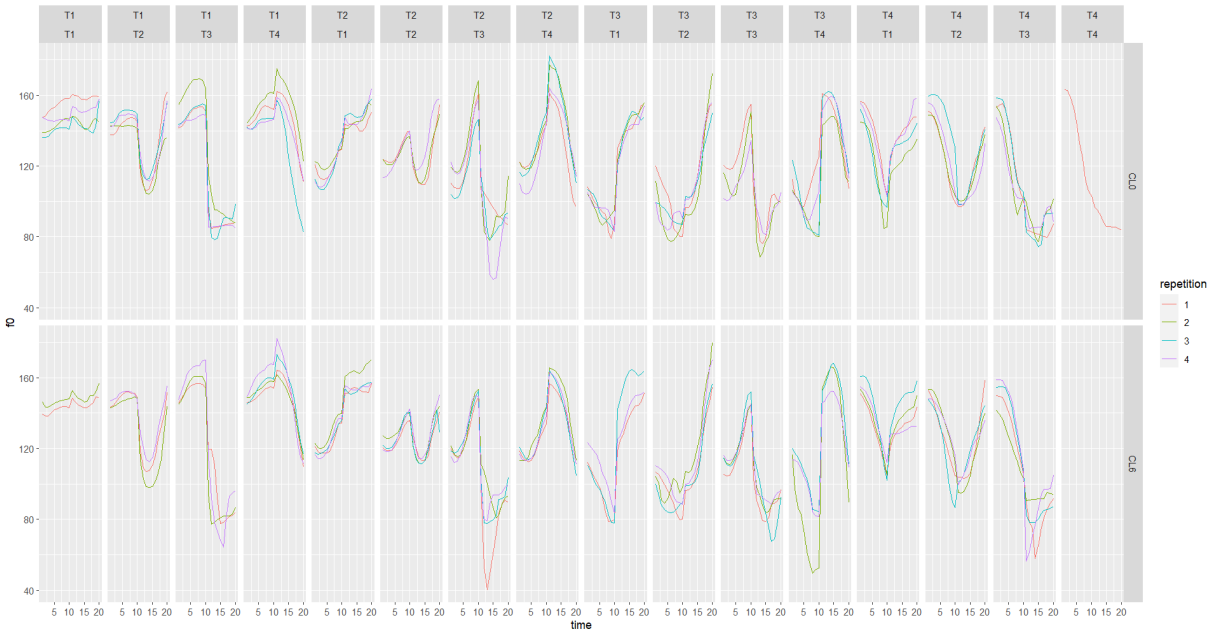

Figure 25: Frequency profile speaker 6.

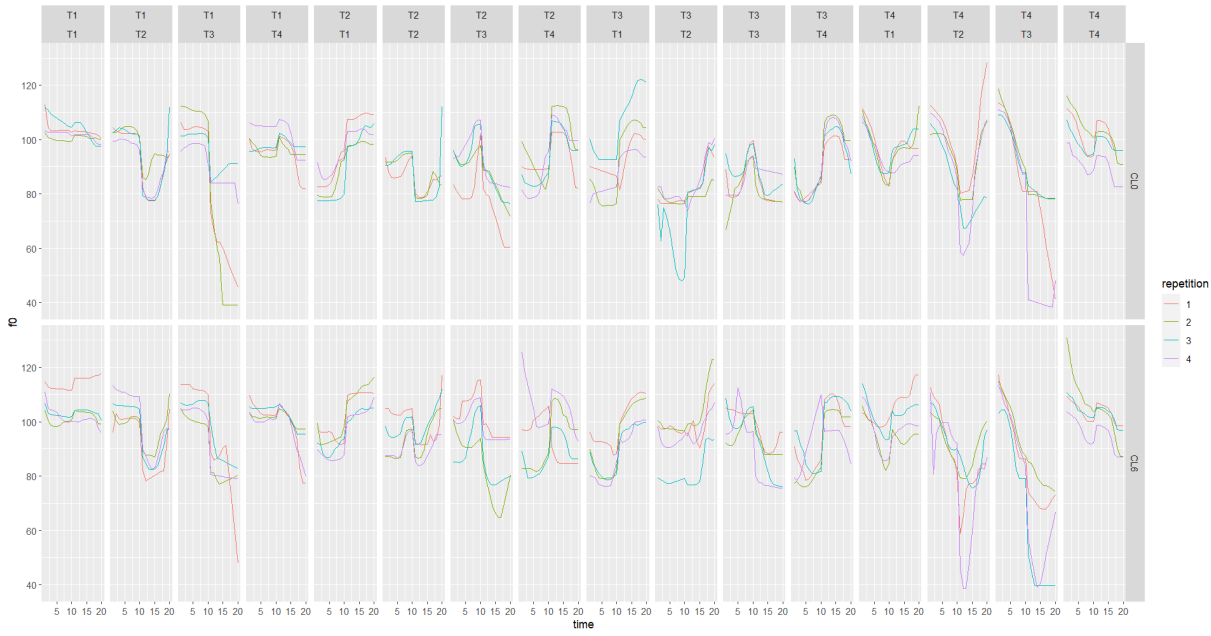

Figure 26: Frequency profile speaker 7.

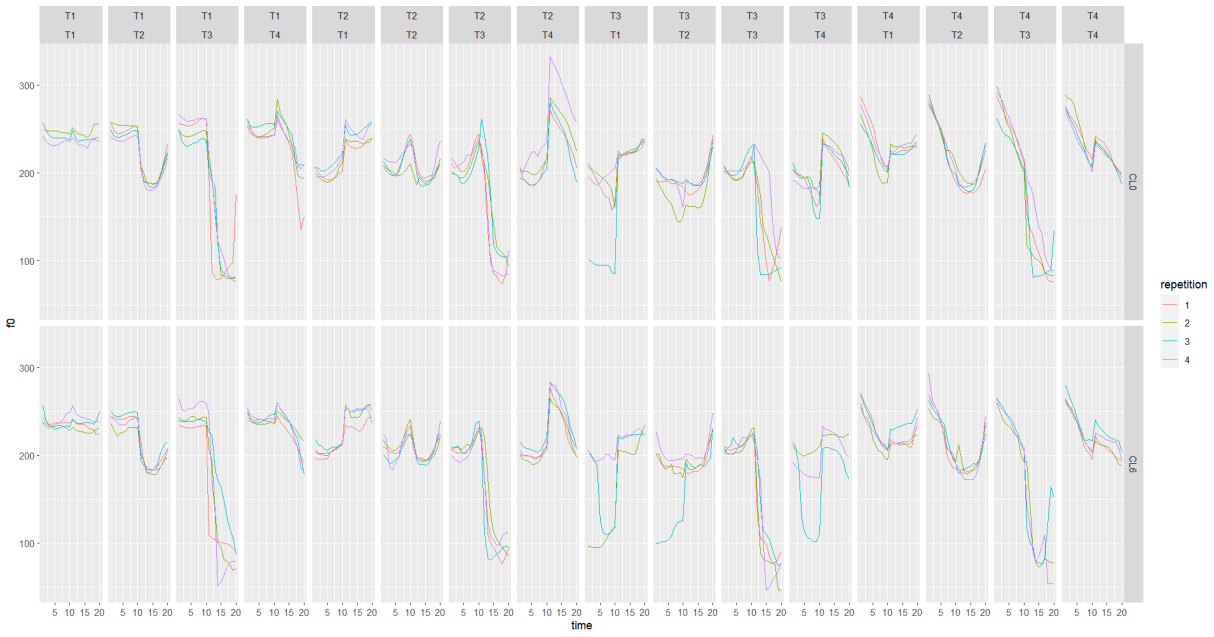

Figure 27: Frequency profile speaker 8.

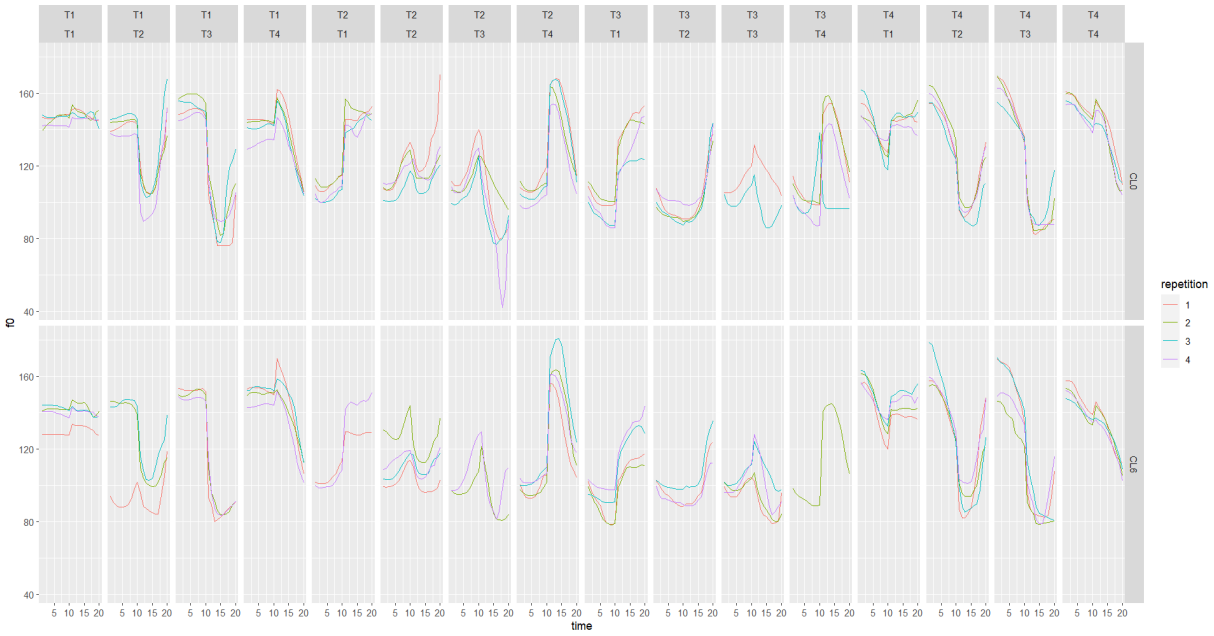

Figure 28: Frequency profile speaker 9.

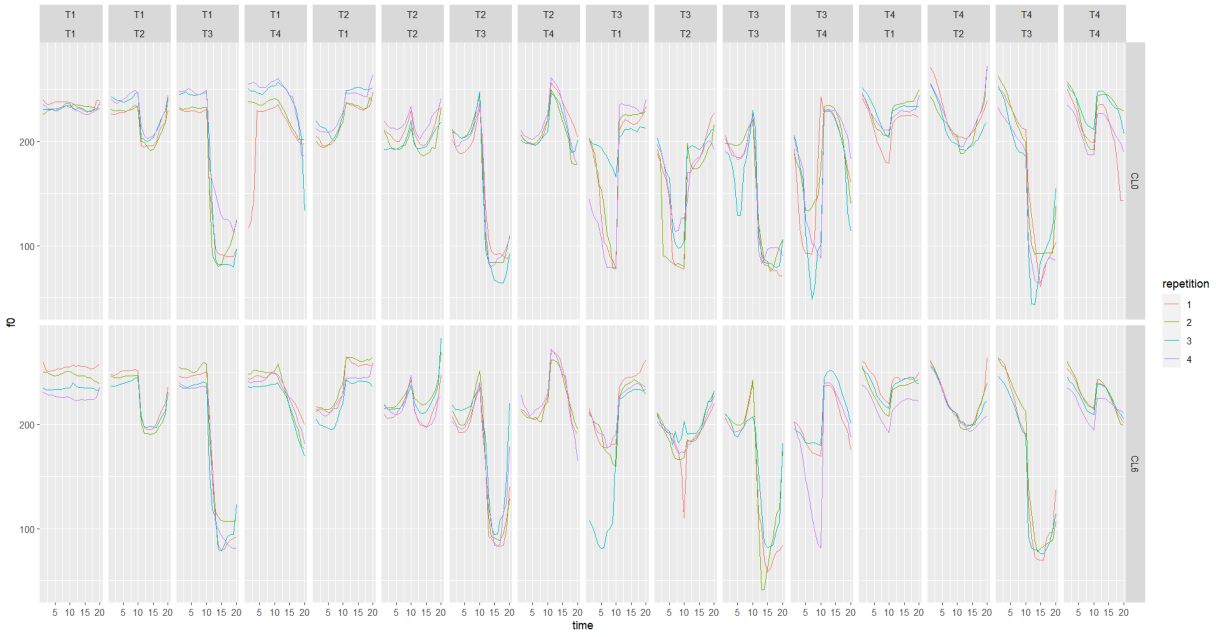

Figure 29: Frequency profile speaker 10.

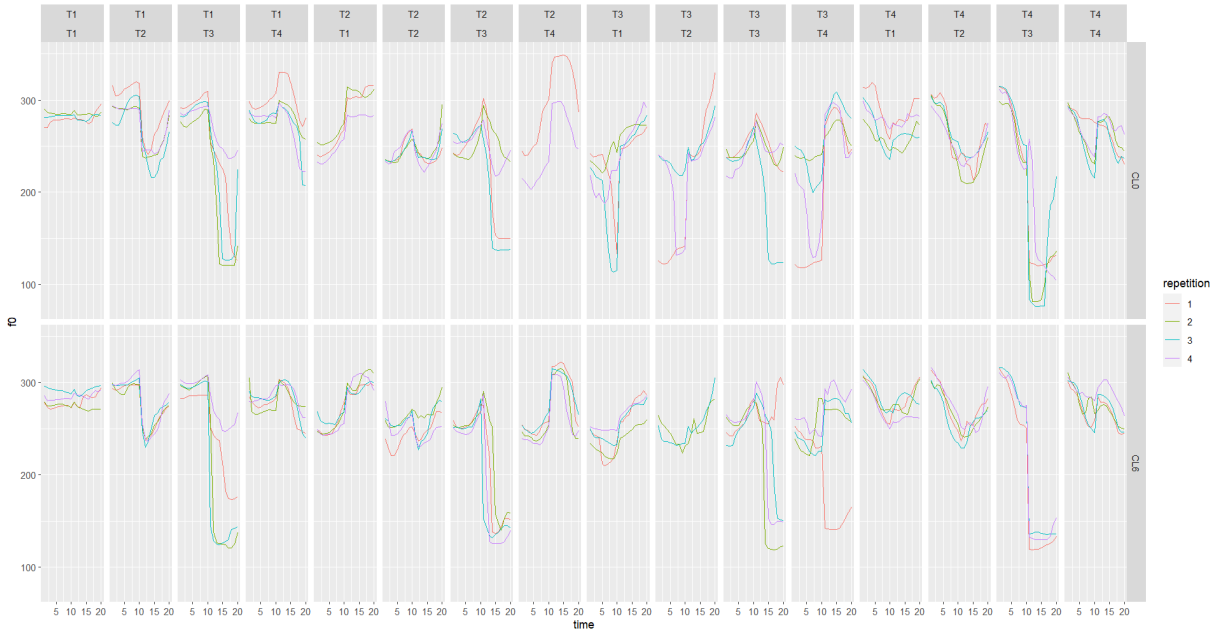

Figure 30: Frequency profile speaker 11.

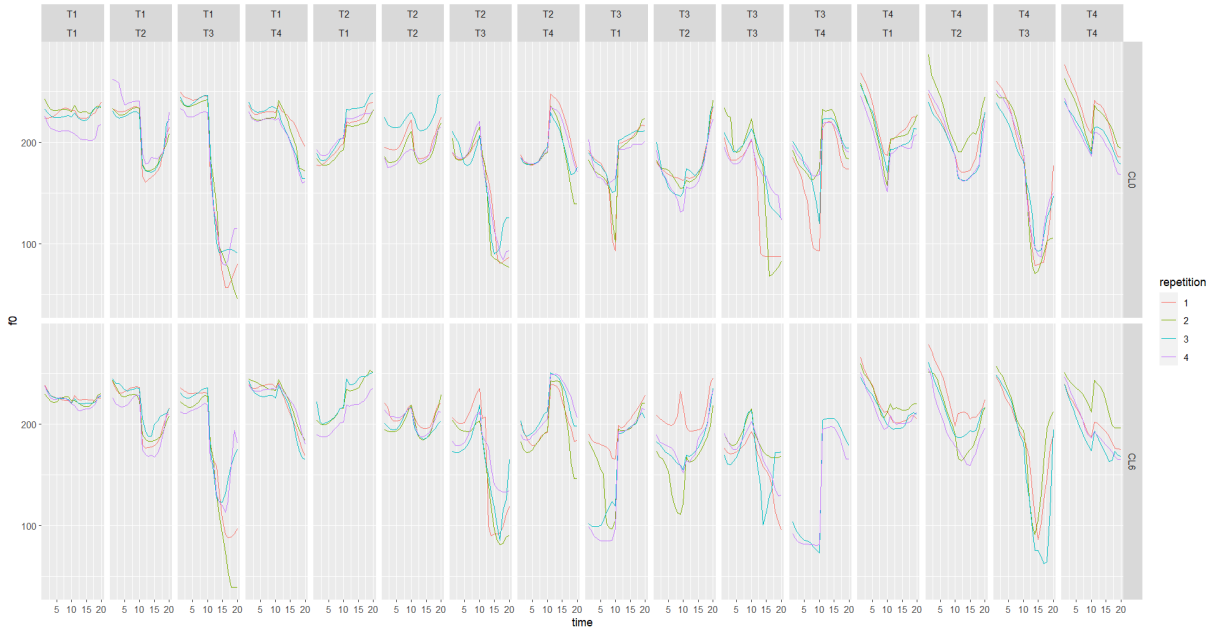

Figure 31: Frequency profile speaker 12.

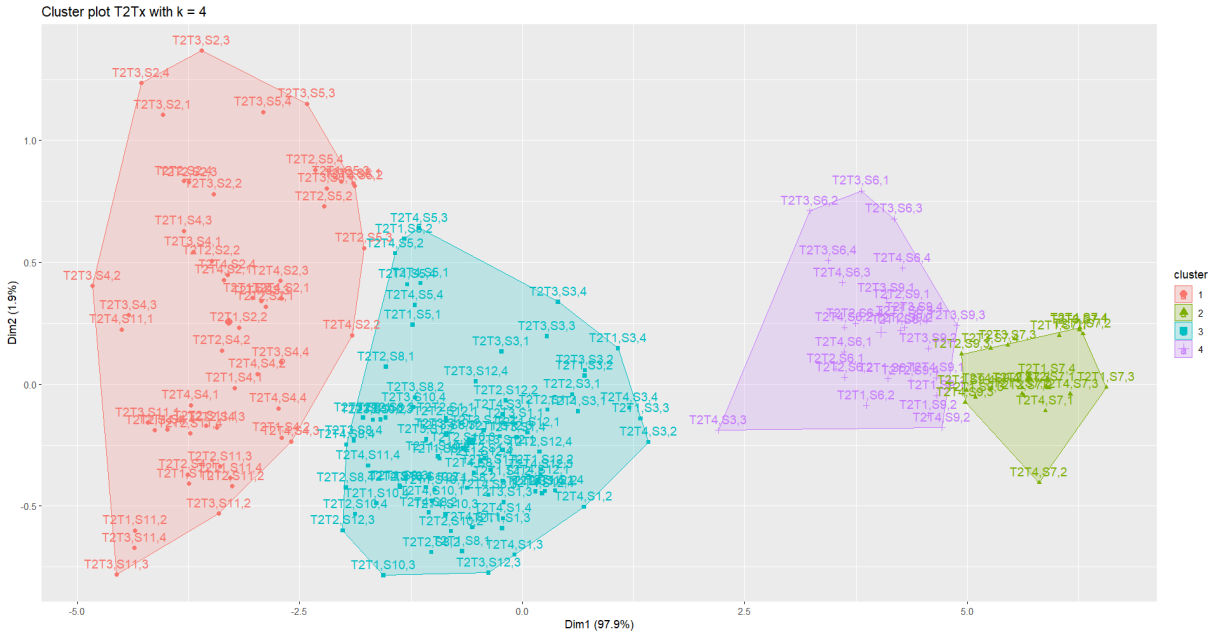

Figure 32: Clusters for $T2Tx$ using kmeans() from the stats package with $k = 4$ and where all speakers are considered.

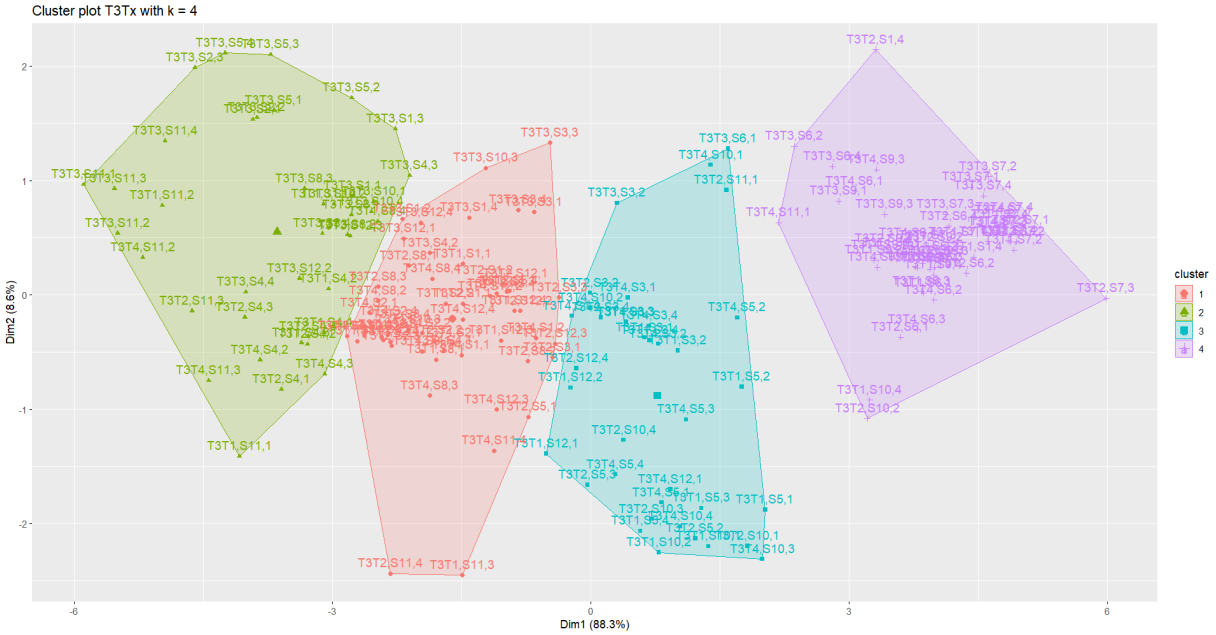

Figure 33: Clusters for $T3Tx$ using kmeans() from the stats package with $k = 4$ and where all speakers are considered.

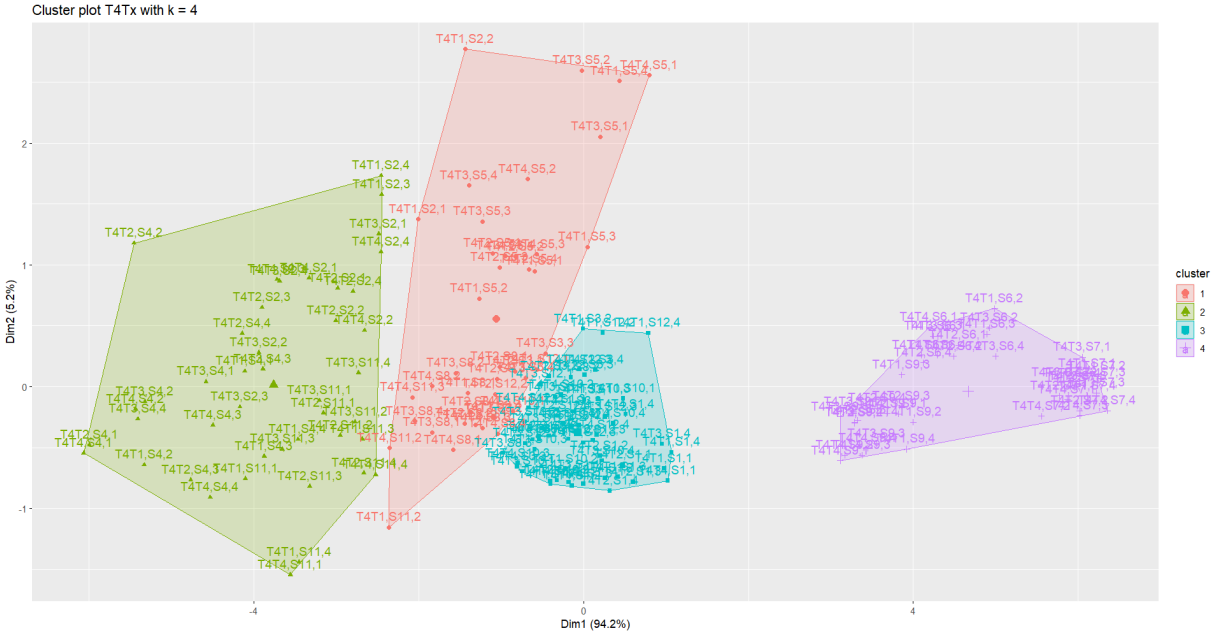

Figure 34: Clusters for $T4Tx$ using kmeans() from the stats package with $k = 4$ and where all speakers are considered.

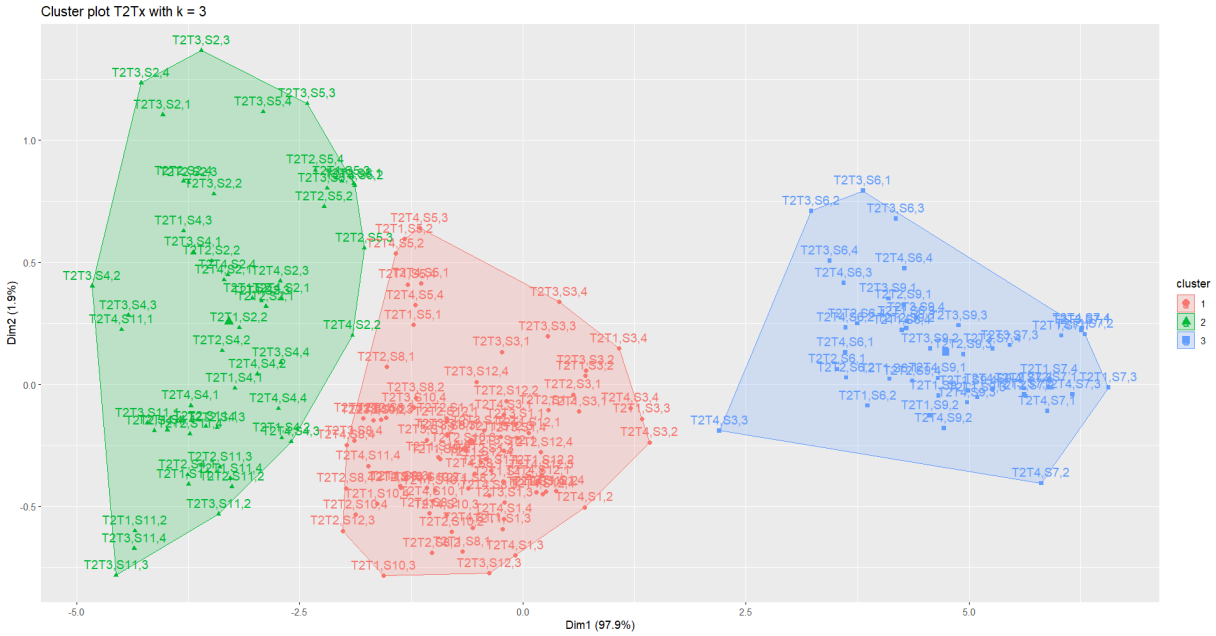

Figure 35: Clusters for $T1Tx$ using kmeans() from the stats package with $k = 3$ and where all speakers are considered.

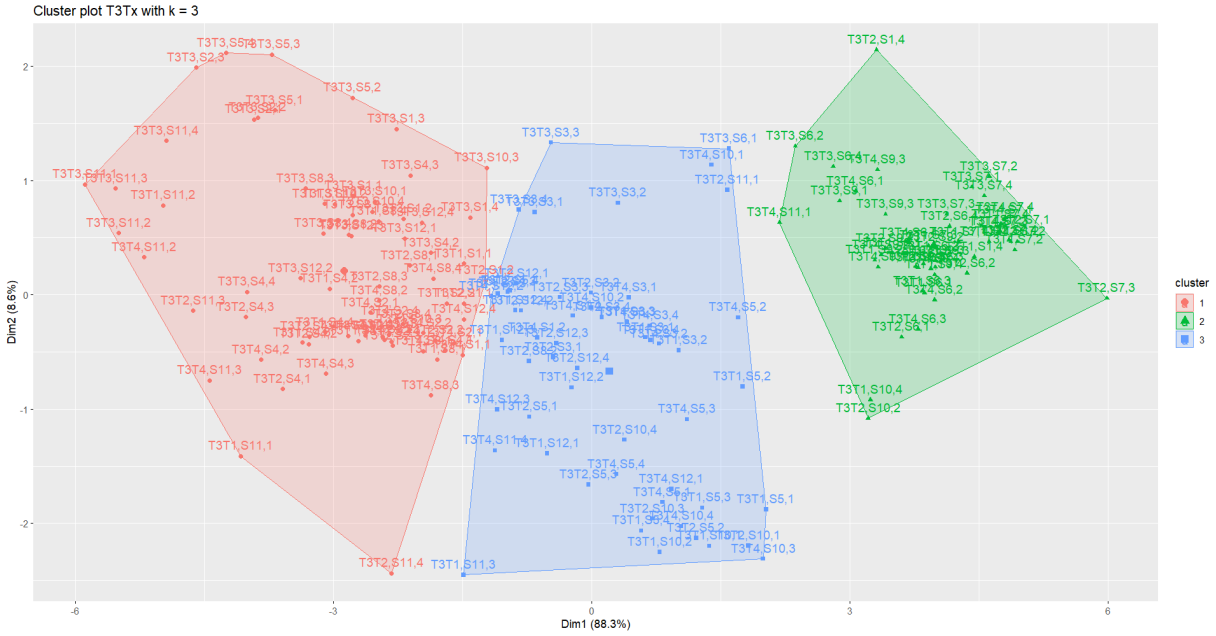

Figure 36: Clusters for $T1Tx$ using kmeans() from the stats package with $k = 3$ and where all speakers are considered.

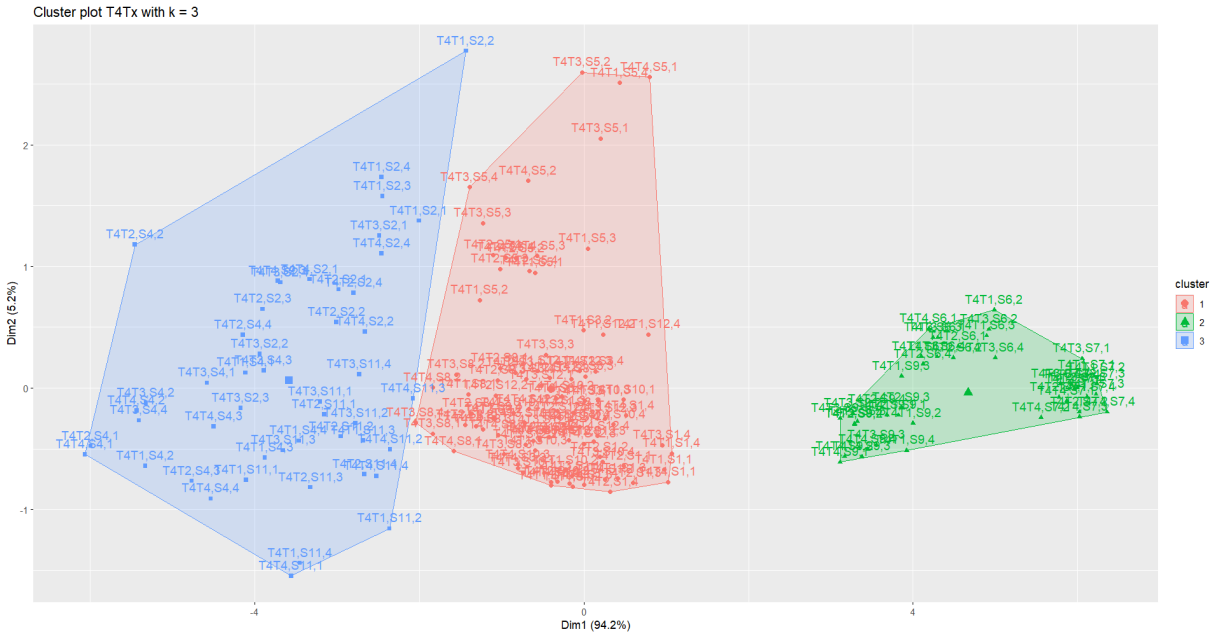

Figure 37: Clusters for $T1Tx$ using kmeans() from the stats package with $k = 3$ and where all speakers are considered.

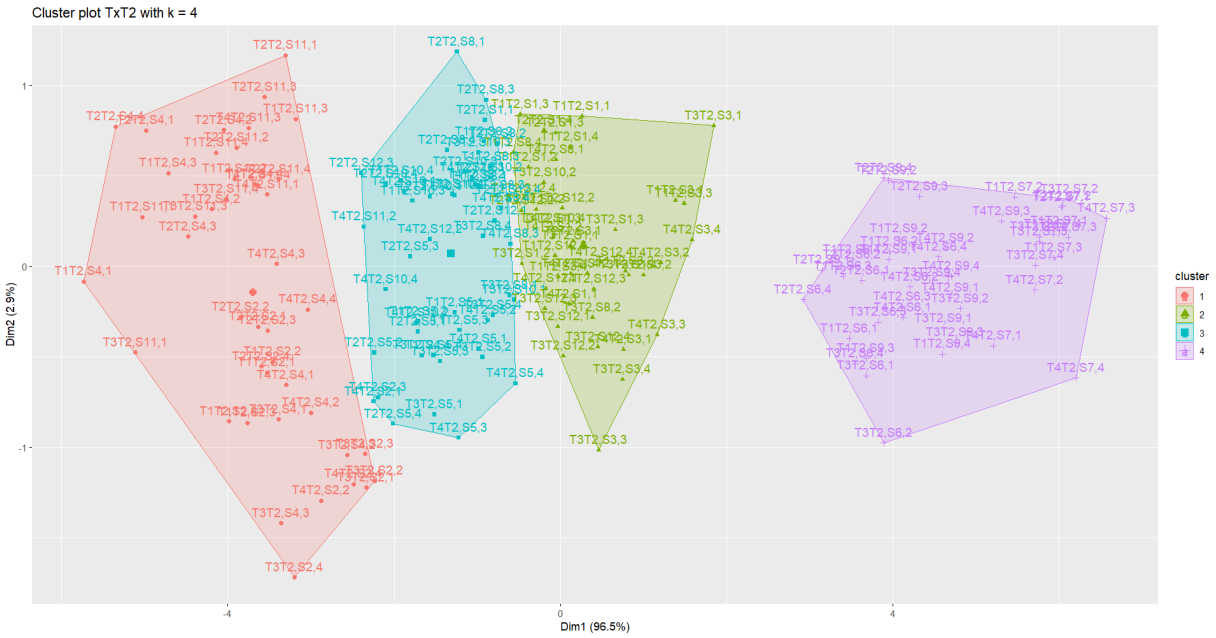

Figure 38: Clusters for $TxT2$ using kmeans() from the stats package with $k = 4$ and where all speakers are considered.

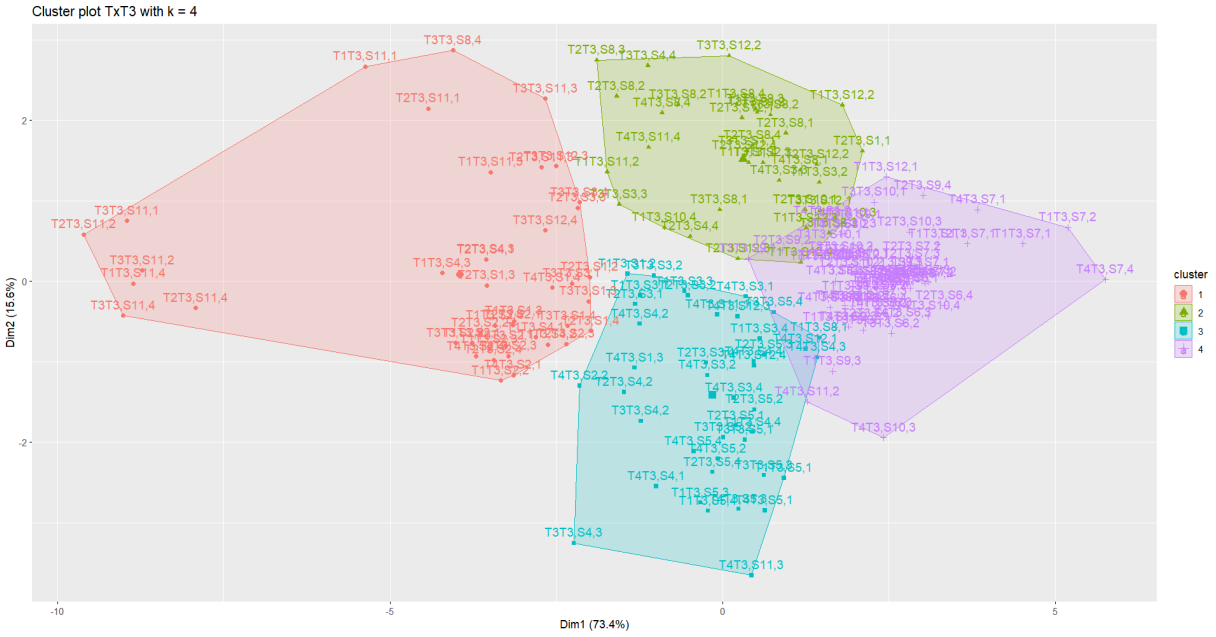

Figure 39: Clusters for $TxT3$ using kmeans() from the stats package with $k = 4$ and where all speakers are considered.

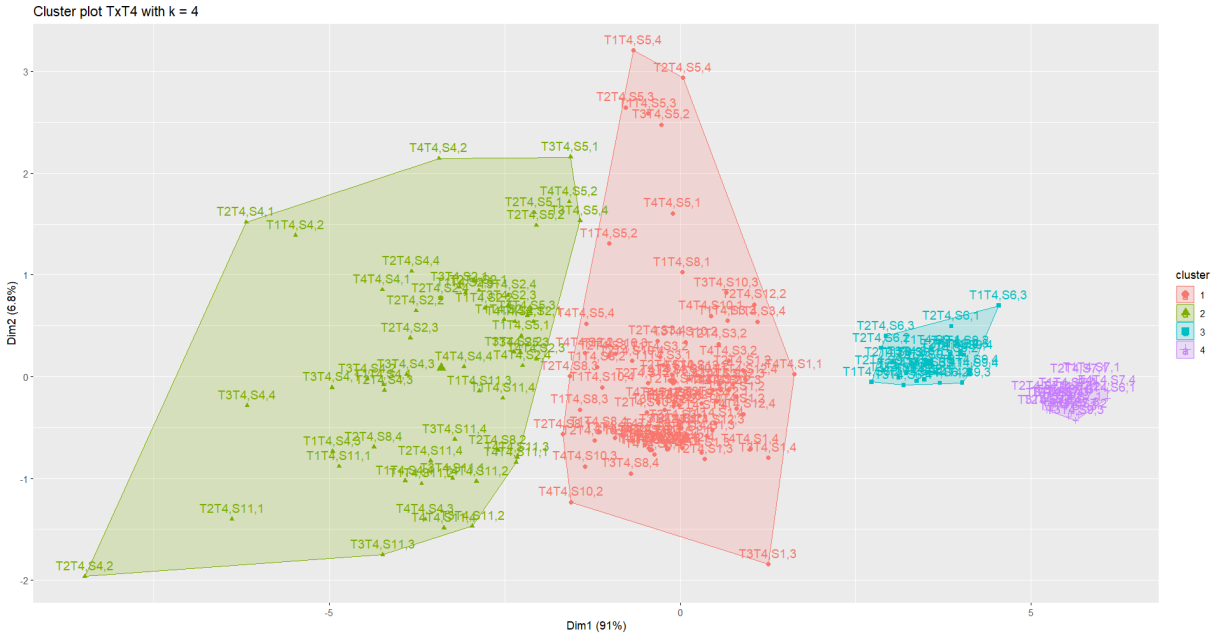

Figure 40: Clusters for $TxT4$ using kmeans() from the stats package with $k = 4$ and where all speakers are considered.

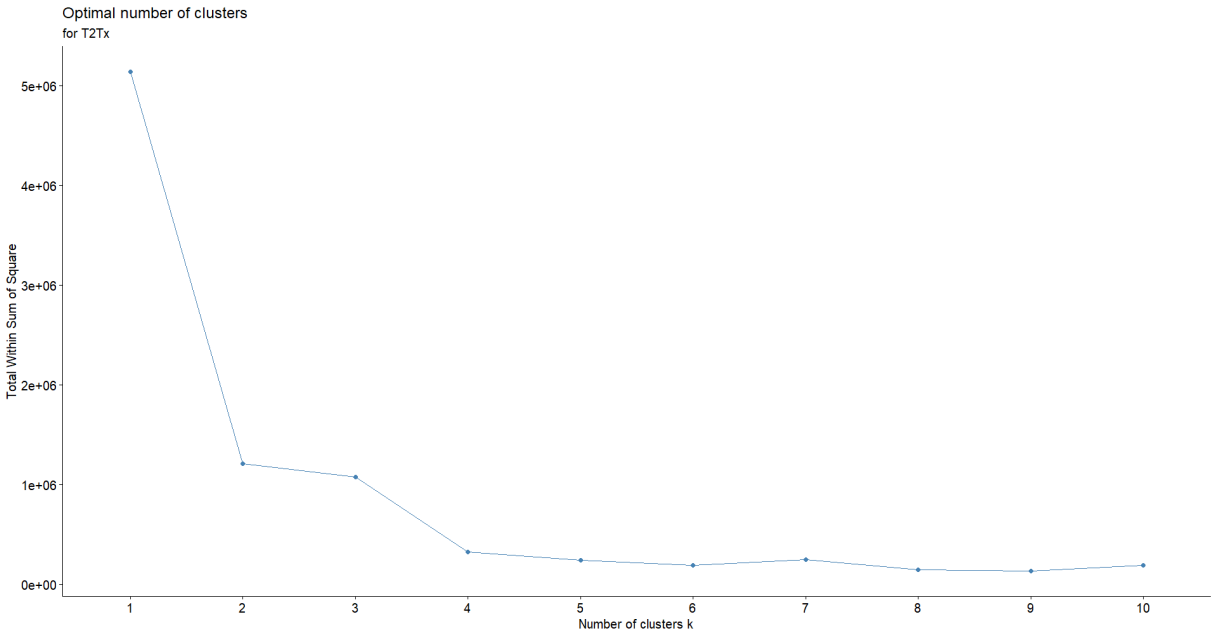

Figure 41: Optimal number of clusters for the tonal combination $T2Tx$.

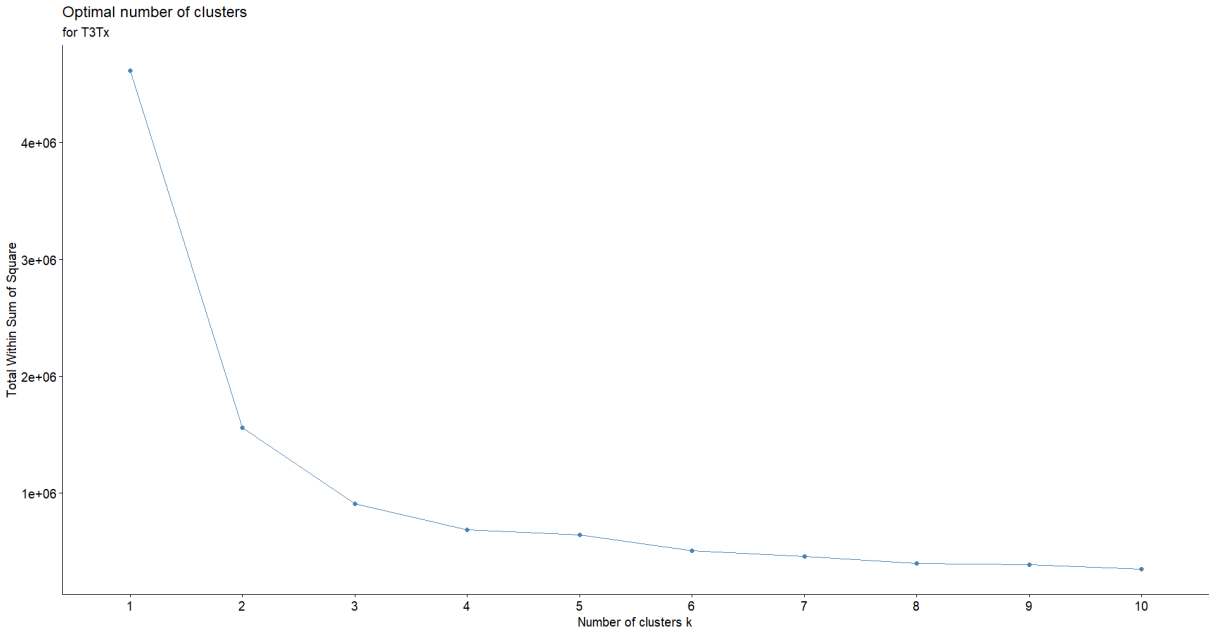

Figure 42: Optimal number of clusters for the tonal combination $T3Tx$.

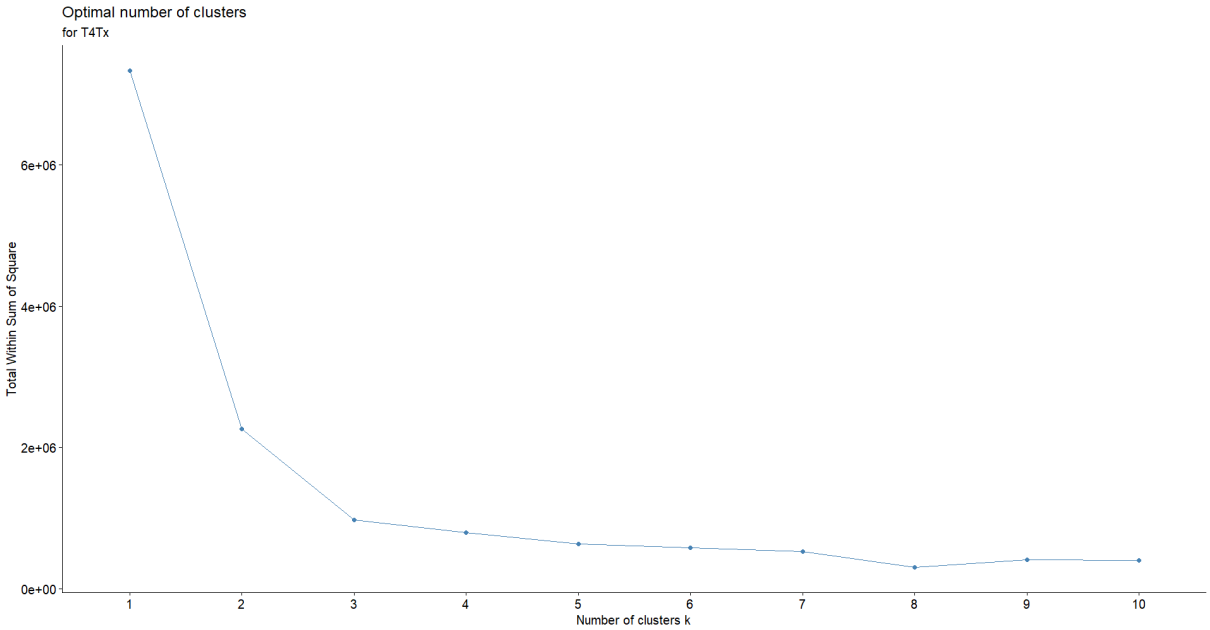

Figure 43: Optimal number of clusters for the tonal combination $T4Tx$.

Figure 44: Clusters for $T2Tx$ for speaker 1 with the Euclidean distance using the kmeans() function with $k = 4$.

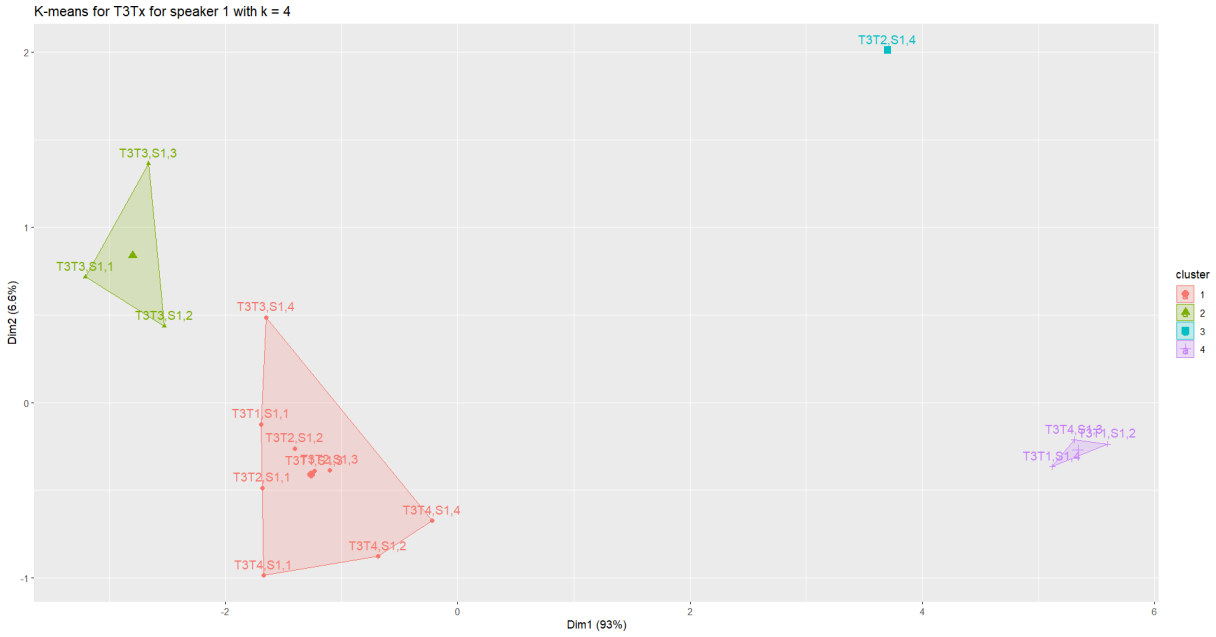

Figure 45: Clusters for $T3Tx$ for speaker 1 with the Euclidean distance using the kmeans() function with $k = 4$.

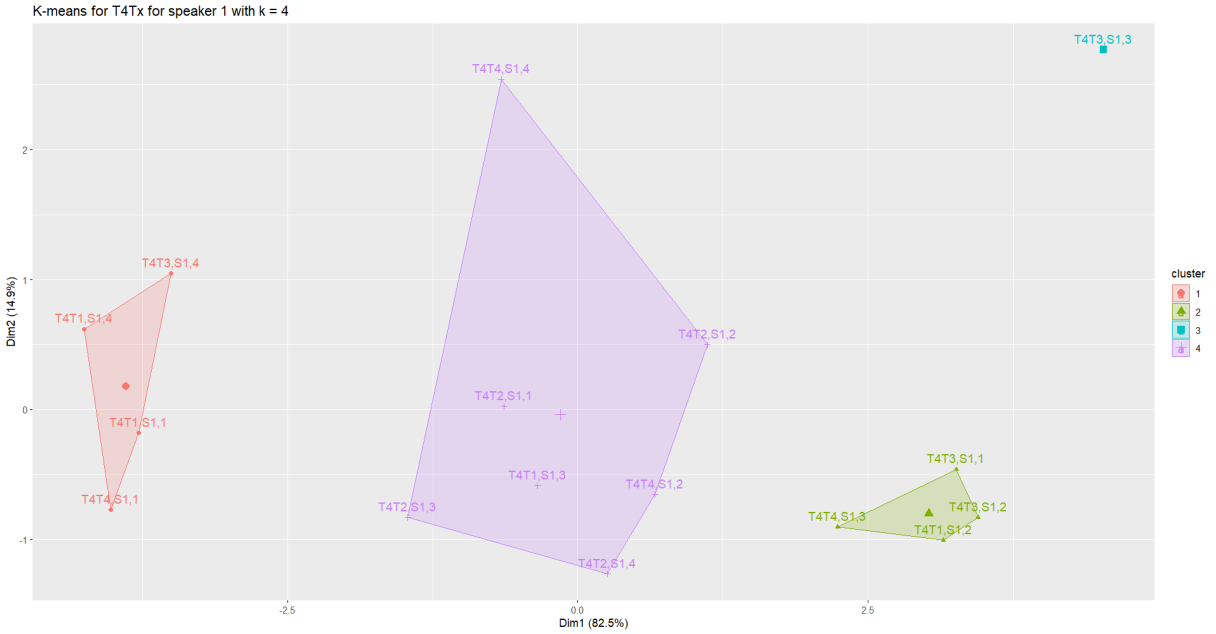

Figure 46: Clusters for $T4Tx$ for speaker 1 with the Euclidean distance using the kmeans() function with $k = 4$.

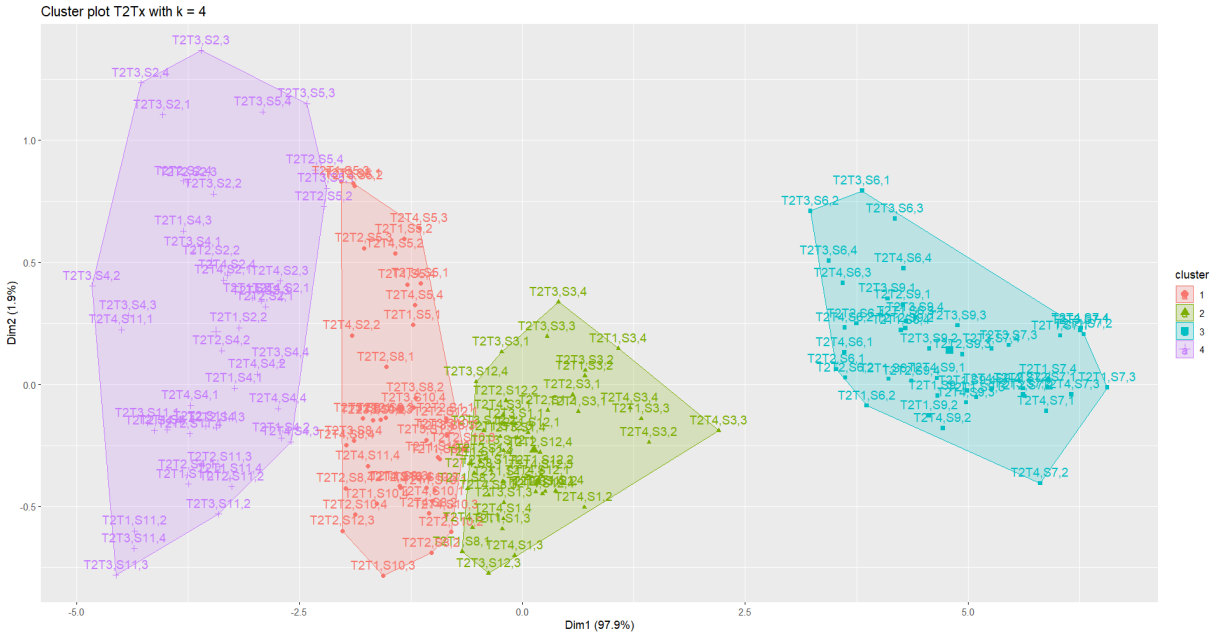

Figure 47: Clusters for $T2Tx$ for speaker 1 with the Euclidean distance using the KMeans() function with $k = 4$.

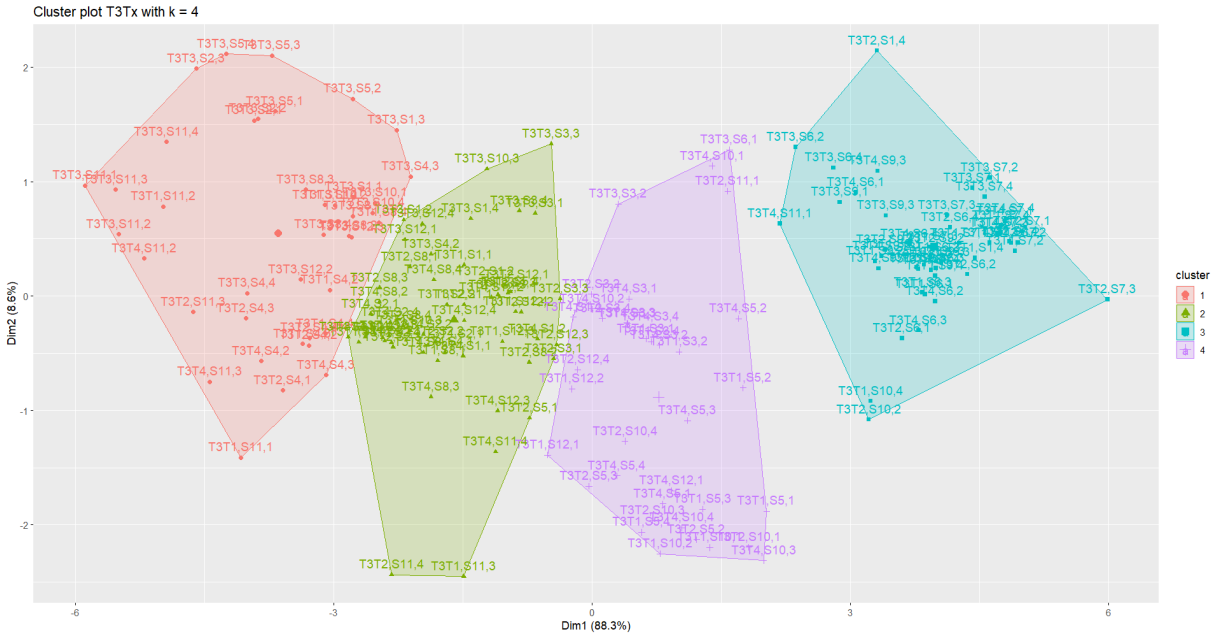

Figure 48: Clusters for $T3Tx$ for speaker 1 with the Euclidean distance using the KMeans() function with $k = 4$.

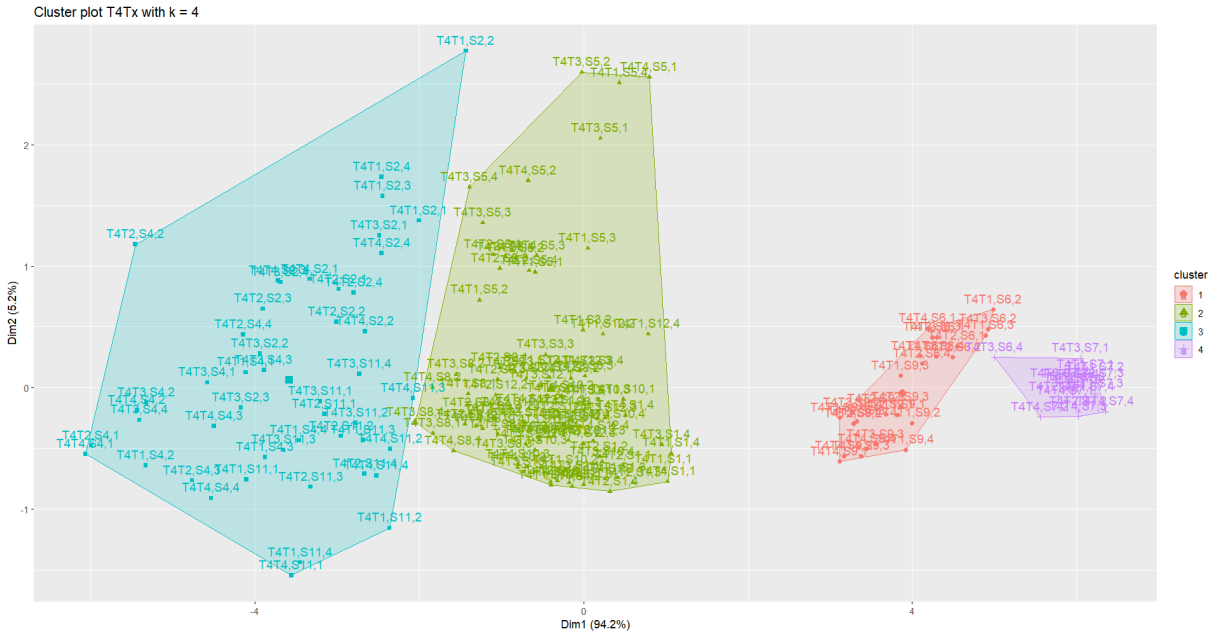

Figure 49: Clusters for $T4Tx$ for speaker 1 with the Euclidean distance using the KMeans() function with $k = 4$.

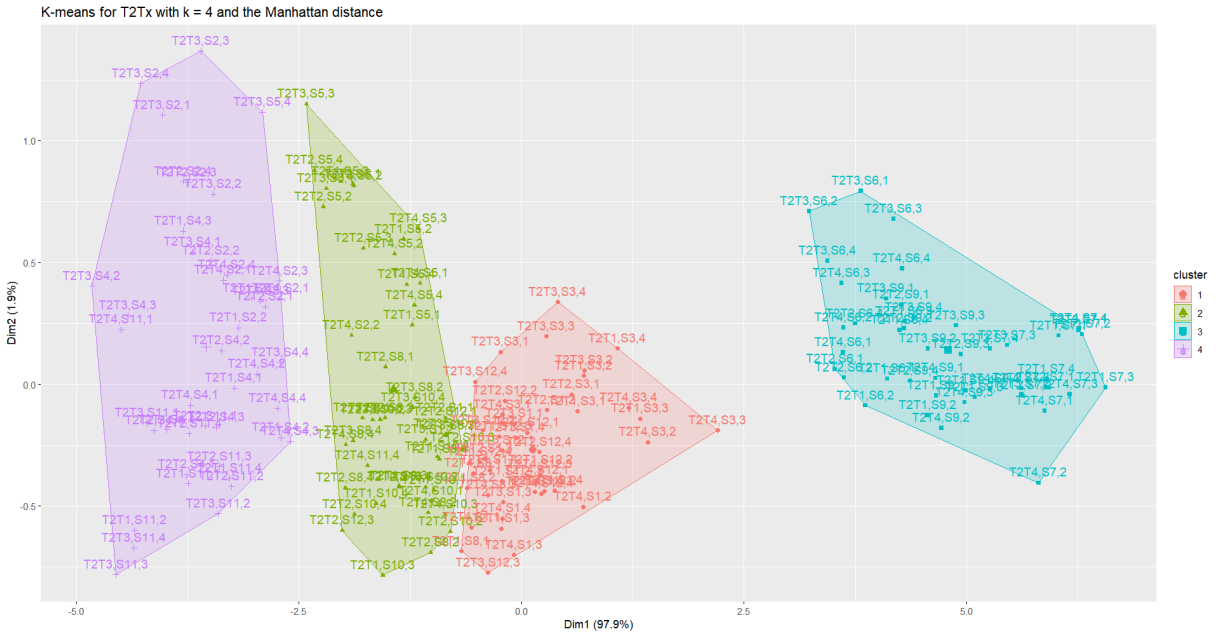

Figure 50: Clusters for $T2Tx$ with the Manhattan distance with the KMeans() function with $k = 4$ and where all speakers are considered.

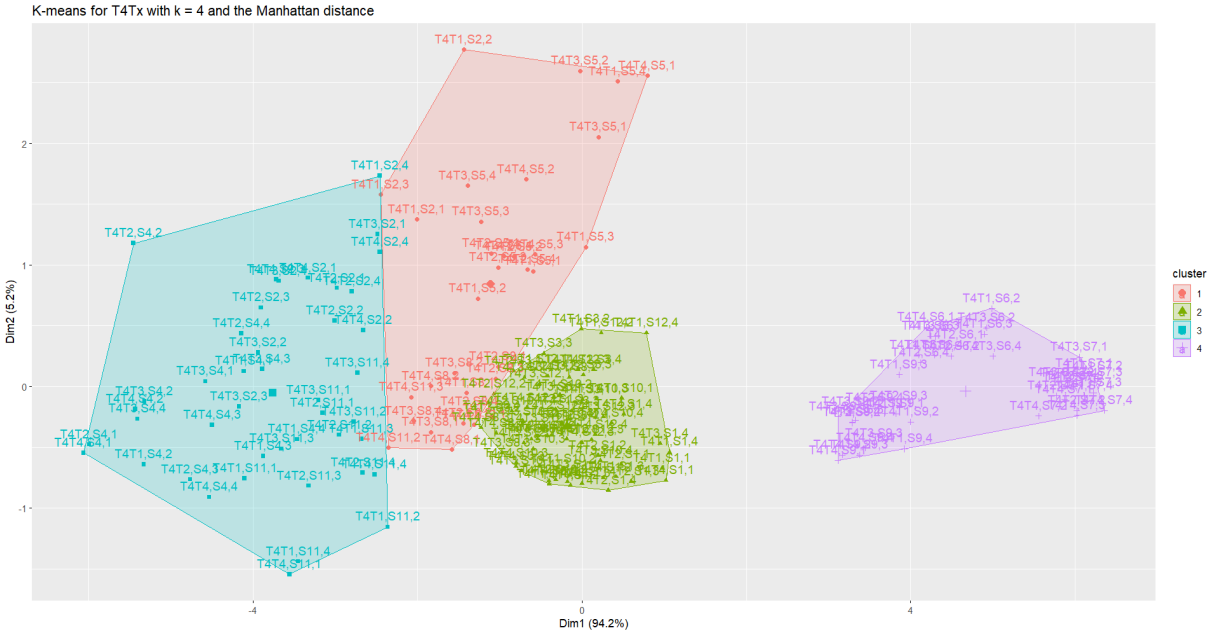

Figure 51: Clusters for $T4Tx$ with the Manhattan distance with the KMeans() function with $k = 4$ and where all speakers are considered.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
1	4148.225	4189.434	4213.550	4219.655	4216.637	4215.961	4222.622	4239.717	4261.898	4278.916
2	4189.434	4240.170	4269.483	4278.790	4276.753	4276.464	4283.558	4301.165	4324.015	4342.089
3	4213.550	4269.483	4303.510	4315.833	4314.863	4314.932	4322.709	4340.927	4364.441	4383.446
4	4219.655	4278.790	4315.833	4331.003	4331.568	4332.367	4340.831	4359.401	4383.145	4402.485
5	4216.637	4276.753	4314.863	4331.568	4334.257	4336.239	4345.490	4364.455	4388.217	4407.505
6	4215.961	4276.464	4314.932	4332.367	4336.239	4339.319	4349.355	4368.714	4392.431	4411.752
7	4222.622	4283.558	4322.709	4340.831	4345.490	4349.355	4360.532	4380.745	4404.878	4424.464
8	4239.717	4301.165	4340.927	4359.401	4364.455	4368.714	4380.745	4402.149	4427.056	4446.756
9	4261.898	4324.015	4364.441	4383.145	4388.217	4392.431	4404.878	4427.056	4453.206	4473.599
10	4278.916	4342.089	4383.446	4402.485	4407.505	4411.752	4424.464	4446.756	4473.599	4495.524

Figure 52: Covariance matrix for $T1T1$. The rows as well as the columns correspond with the time points $t = 1, 2, ..., 10$.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
1	4636.289	4640.023	4660.390	4678.493	4718.915	4759.296	4803.414	4848.727	4881.762	4908.940
2	4640.023	4653.917	4678.613	4698.062	4738.375	4778.504	4821.915	4866.129	4898.717	4925.664
3	4660.390	4678.613	4707.577	4730.357	4772.503	4812.731	4855.707	4899.547	4931.972	4958.685
4	4678.493	4698.062	4730.357	4759.897	4807.569	4849.594	4893.496	4937.749	4969.918	4996.284
5	4718.915	4738.375	4772.503	4807.569	4861.135	4906.003	4951.821	4996.883	5028.727	5054.893
6	4759.296	4778.504	4812.731	4849.594	4906.003	4954.027	5002.103	5047.917	5079.522	5105.738
7	4803.414	4821.915	4855.707	4893.496	4951.821	5002.103	5052.693	5099.566	5130.961	5157.240
8	4848.727	4866.129	4899.547	4937.749	4996.883	5047.917	5099.566	5148.188	5180.676	5207.710
9	4881.762	4898.717	4931.972	4969.918	5028.727	5079.522	5130.961	5180.676	5215.011	5243.433
10	4908.940	4925.664	4958.685	4996.284	5054.893	5105.738	5157.240	5207.710	5243.433	5273.982

Figure 53: Covariance matrix for $T1T2$. The rows as well as the columns correspond to the time points $t = 1, 2, ..., 10$.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
1	4135.412	4156.895	4190.190	4220.242	4238.797	4266.949	4301.755	4336.571	4377.316	4419.599
2	4156.895	4190.288	4230.111	4263.135	4282.694	4311.505	4346.639	4380.515	4420.610	4462.501
3	4190.190	4230.111	4276.529	4313.281	4334.419	4364.436	4400.042	4433.623	4473.767	4516.048
4	4220.242	4263.135	4313.281	4354.185	4377.779	4409.945	4446.502	4480.321	4520.591	4563.158
5	4238.797	4282.694	4334.419	4377.779	4404.024	4438.543	4476.393	4510.939	4551.488	4594.077
6	4266.949	4311.505	4364.436	4409.945	4438.543	4476.115	4515.917	4551.510	4592.653	4635.519
7	4301.755	4346.639	4400.042	4446.502	4476.393	4515.917	4557.630	4594.367	4636.203	4679.406
8	4336.571	4380.515	4433.623	4480.321	4510.939	4551.510	4594.367	4632.647	4675.763	4719.626
9	4377.316	4420.610	4473.767	4520.591	4551.488	4592.653	4636.203	4675.763	4721.162	4766.640
10	4419.599	4462.501	4516.048	4563.158	4594.077	4635.519	4679.406	4719.626	4766.640	4814.579

Figure 54: Covariance matrix for $T1T3$. The rows as well as the columns correspond to the time points $t = 1, 2, ..., 10$.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
1	4707.998	4712.376	4667.729	4487.055	4491.352	4493.127	4519.661	4562.136	4612.218	4628.337
2	4712.376	4729.943	4696.128	4529.328	4535.295	4537.629	4564.460	4606.598	4657.272	4674.013
3	4667.729	4696.128	4680.799	4559.766	4567.698	4570.733	4598.060	4639.812	4690.507	4707.595
4	4487.055	4529.328	4559.766	4591.251	4600.658	4603.460	4631.144	4673.331	4724.508	4742.627
5	4491.352	4535.295	4567.698	4600.658	4612.956	4617.715	4646.172	4688.312	4739.017	4756.983
6	4493.127	4537.629	4570.733	4603.460	4617.715	4624.345	4653.554	4695.835	4746.203	4764.040
7	4519.661	4564.460	4598.060	4631.144	4646.172	4653.554	4683.892	4727.000	4777.646	4795.742
8	4562.136	4606.598	4639.812	4673.331	4688.312	4695.835	4727.000	4771.926	4823.763	4842.443
9	4612.218	4657.272	4690.507	4724.508	4739.017	4746.203	4777.646	4823.763	4877.817	4898.068
10	4628.337	4674.013	4707.595	4742.627	4756.983	4764.040	4795.742	4842.443	4898.068	4920.359

Figure 55: Covariance matrix for $T1T4$. The rows as well as the columns correspond to the time points $t = 1, 2, ..., 10$.