BMC Bioinformatics

AliClu - Temporal sequence alignment for clustering longitudinal clinical data

--Manuscript Draft--

Manuscript Number:	BINF-D-19-00580
Full Title:	AliClu - Temporal sequence alignment for clustering longitudinal clinical data
Article Type:	Software
Section/Category:	Machine Learning and Artificial Intelligence in Bioinformatics
Funding Information:	Fundação para a Ciência e a Tecnologia (UID/CEC/50021/2019) — Prof. Susana Vinga Fundação para a Ciência e a Tecnologia (UID/EEA/50008/2019) — Prof. Alexandra M Carvalho Fundação para a Ciência e a Tecnologia (PTDC/CCI-CIF/29877/2017) — Prof. Alexandra M Carvalho Fundação para a Ciência e a Tecnologia (PTDC/EMS-SIS/0642/2014) — Prof. Susana Vinga Fundação para a Ciência e a Tecnologia (PTDC/EEI-SII/1937/2014) — Prof. Alexandra M Carvalho
Abstract:	Background: Patients stratification is a critical task in clinical decision since it can support physicians to choose treatments in a personalized way. Given the increasing availability of electronic medical records (EMR) with longitudinal data, one crucial problem is how to cluster the patients efficiently based on temporal information obtained in medical appointments. In this work, we propose to apply the Temporal Needleman-Wunsch algorithm to align discrete sequences with transitions time information between the symbols. The obtained pairwise scores are then used to perform hierarchical clustering. To find the best number of clusters and assess their stability, a resampling technique is applied. Results: We implemented AliClu for the combined analysis of rheumatoid arthritis EMRs obtained from Reuma.pt, the Portuguese database of rheumatologic patients visits. In particular, we applied AliClu for the analysis of therapy switches, coded as letters corresponding to biologic drugs, interspersed with their durations before each change occurs. The obtained optimized clusters allow to stratify the patients based on their temporal therapy profile and to support the identification of common features for those groups. Conclusions: AliClu is a promising computational strategy to analyze longitudinal patient data by providing validated clusters and by unraveling patterns that may be associated with clinical outcomes. Patient stratification is performed in an automatic or semi-automatic way, allowing to tune the alignment, clustering, and validation parameters. AliClu is freely available at https://github.com/sysbiomed/AliClu.
Corresponding Author:	Susana Vinga Instituto de Engenharia de Sistemas e Computadores Investigacao e Desenvolvimento em Lisboa Lisboa, PORTUGAL
Corresponding Author E-Mail:	susanavinga@gmail.com
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	Instituto de Engenharia de Sistemas e Computadores Investigacao e Desenvolvimento em Lisboa
Corresponding Author's Secondary Institution:	
First Author:	Kishan Rama
First Author Secondary Information:	
Order of Authors:	Kishan Rama Helena Canhão

	Alexandra M Carvalho
	Susana Vinga
Order of Authors Secondary Information:	
Author Comments:	

SOFTWARE

AliClu - Temporal sequence alignment for clustering longitudinal clinical data

Kishan Rama[1,3], Helena Canhão[2], Alexandra M. Carvalho[1] and Susana Vinga[3*]

*Correspondence:
susanavinga@tecnico.ulisboa.pt
[3] INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Rua Alves Redol 9, 1000-029 Lisboa, Portugal
Full list of author information is available at the end of the article

Abstract

Background: Patients stratification is a critical task in clinical decision since it can support physicians to choose treatments in a personalized way. Given the increasing availability of electronic medical records (EMR) with longitudinal data, one crucial problem is how to cluster the patients efficiently based on temporal information obtained in medical appointments. In this work, we propose to apply the Temporal Needleman-Wunsch algorithm to align discrete sequences with transitions time information between the symbols. The obtained pairwise scores are then used to perform hierarchical clustering. To find the best number of clusters and assess their stability, a resampling technique is applied.

Results: We implemented AliClu for the combined analysis of rheumatoid arthritis EMRs obtained from Reuma.pt, the Portuguese database of rheumatologic patients visits. In particular, we applied AliClu for the analysis of therapy switches, coded as letters corresponding to biologic drugs, interspersed with their durations before each change occurs. The obtained optimized clusters allow to stratify the patients based on their temporal therapy profile and to support the identification of common features for those groups.

Conclusions: AliClu is a promising computational strategy to analyze longitudinal patient data by providing validated clusters and by unraveling patterns that may be associated with clinical outcomes. Patient stratification is performed in an automatic or semi-automatic way, allowing to tune the alignment, clustering, and validation parameters. AliClu is freely available at https://github.com/sysbiomed/AliClu.

Keywords: temporal sequence alignment; clustering; bootstrap; clustering indices

Background

The increasing availability of clinical data and the growth of investment in healthcare is driving research towards building better clinical decision support systems for the effective personalization of treatment. In this context machine learning and data mining techniques are becoming ubiquitous, helping to provide high-quality care systems and improve the long-term health of the patients.

Patients health records are being stored in Electronic Medical Records (EMR) including a variety of data, such as demographics, medical history, laboratory test results, medication, and allergies. These EMR systems are designed to store patients data across time, providing large longitudinal cohorts. Exploring disease heterogeneity and patterns in these

datasets is a challenging task. Several issues contribute to this: the exponential number of all possible combinations in patients trajectories, their variability in the temporal scale, and the complexity of representing them.

We address the problem of learning temporal patterns in EMR data as a combined approach of (temporal) alignment and hierarchical clustering. More specifically, we propose to use the Temporal Needleman-Wunsch (TNW) algorithm [1] to align discrete sequences with time information between the symbols and, subsequently, perform hierarchical clustering using the obtained pairwise scores. The TNW algorithm is an extension of the traditional Needleman-Wunsch (NW) [2] for global sequence alignment. TNW takes into account not only the matches between symbols, as in the NW algorithm, but also adds a penalization term for the differences in the time values between two sequences. Other temporal alignment methods, such as dynamic time warping, are not adequate to deal with these type of data, just providing general trends for matching continuous-time signals [3, 4, 5, 6].

The TNW is particularly interesting when in the presence of data representing given events (coded as symbols) and their corresponding duration. Treatment switching provides us an excellent example of this type of temporal sequence data. Starting in instant 0, with Treatment A, its failure after $time_A$ may lead to switching to Treatment B. In this case, we would have a patient profile given by the sequence (0.A,$time_A$.B,$time_B$), which includes symbols and numeric values. Clustering patients with similar treatment profiles would allow identifying common features for those groups and delineate strategies to improve treatment outcome.

Implementation

The pipeline of the proposed method, named AliClu, is illustrated in Figure 1. In the first step, raw data is pre-processed to obtain temporal sequences. Then, on the second step, pairwise temporal sequence alignment is performed, and a similarity matrix is obtained. The third step consists of converting the similarity matrix into distances. Agglomerative clustering is then performed with this distance matrix and, finally, the clustering results are validated via a bootstrapping approach. The obtained patient stratification can be graphically represented to ease clinical interpretation. Each step of this pipeline is detailed next.

Data Pre-Processing

This pre-processing step creates temporal sequences for each patient, from EMRs. Patients records are typically available in *panel data* format, where each patient is spread in different lines, one for each medical appointment, and the columns contain the features of interest measured over time. In this work, we consider each patient experiences a sequence of events spaced in time. Let A and B be events of interest, for a given patient, with time-distance t between them; a *prefix-encoded* (PE) sequence for that patient is defined as $0.A\ t.B$.

In this pre-processing phase, PE sequences are built for each patient, requiring information about the patient id, the event under study, and the time between two consecutive events. These features must be taken from the panel data. In there, time may appear formatted as a date or just a number in any time unit (e.g., seconds, minutes, days). Depending on the time format, two types of pre-processing are implemented. We refer the interested reader to the Additional File for further details.

Temporal Sequence alignment

After building the prefix-encoded (PE) sequences, it is possible to perform alignment between all patient pairs using the TNW algorithm [1]. TNW guarantees convergence to the optimal alignment, for a given scoring scheme, gap penalty g, and temporal penalty T_p. Notwithstanding, alignments can change drastically depending on the choice of these parameters, the reason why they should be carefully chosen.

The information of the retrieved alignments is summarized into a $N \times N$ similarity matrix S, where N is the number of patients in the data. In this matrix, the value at entry (i, j) gives the score of the alignment of i-th and j-th patients. Due to symmetry, only $N \times (N-1)/2$ entries need to be computed.

Distance matrix

Before using the agglomerative clustering algorithm, we need to convert the similarity matrix S, obtained in the previous step, into a distance matrix D. To this end, we take the symmetric value of each score and then we shift it by adding the maximum similarity score in matrix S. This shift is made in order to make all scores greater than or equal to zero. To sum up, the distance matrix is computed as:

$$a = \max_{i<j} S_{ij} \text{ with } i, j = 1, \ldots, N \quad \text{and} \quad D = -S + a\left(\mathbf{1} \cdot \mathbf{1}^T\right) \text{ with } \mathbf{1} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \in \mathbf{R}^N.$$

Clustering of temporal sequence alignments

The dissimilarity matrix obtained is then used to perform agglomerative hierarchical clustering [7]. The resulting groups can be depicted in a dendrogram, allowing to visualize a tree showing the order and distances of merges performed during the clustering procedure. Five different linkage functions are used, namely, single, complete, average, centroid, and Ward's method. Since hierarchical clustering methods do not explicitly set the number the clusters, AliClu additionally provides an automatic bootstrapping-based validation technique proposed by Mucha [8] that allows the selection of the best number according to several cluster indexes. They include *Rand* [9], *adjusted Rand* (AR) [10], *Fowlkes and Mallows* (FM) [11], *Jaccard*, and *adjusted Wallace* (AW) [12].

The pseudo-code of the cluster and validation procedure is given in Algorithm 1. As input the algorithm receives a distance matrix D for the agglomerative clustering algorithm, the number of bootstrap samples M, the linkage criterion L, and the minimum $K_{\min}$ and the maximum $K_{\max}$ number of clusters to be analysed.

Algorithm 1 Agglomerative clustering

1: Perform agglomerative clustering on distance matrix D, outputing a dendrogram Z.
2: **Repeat** M times:
3: - Bootstrap sample – randomly select $\frac{3}{4}$ patients from the original data.
4: - Create a new distance matrix D' for the bootstrap sample.
5: - Perform agglomerative clustering on D' with L which outputs a dendrogram Z'.
6: - Let $q = K_{\min}$.
7: **While** $q \leq K_{\max}$:
8: - Cut dendrograms Z and Z' in order to obtain q clusters.
9: - Compute Rand, AR, FM, Jaccard, and AW between the original and bootstrap partition.
10: - Let $q = q + 1$.
11: Evaluate statistics of the M computations for each analysed q.

The algorithm begins by performing agglomerative clustering on distance matrix D in Step 1. Then, an outer loop starts in Step 2, corresponding to a bootstrapping procedure.

From Step 3 to 5, a bootstrap sample is generated, and agglomerative clustering is performed on it. Then, an inner loop computes the clustering indices between the clustering of the original patients and the clustering of the bootstrap sample (Steps 6-10). In Step 8, the obtained dendrograms Z and Z' are cut to retrieve q clusters (in each), with $K_{\min} \leq q \leq K_{\max}$. After running the outer loop M times, the statistics of the clustering indices are computed then (Step 11).

The output of Algorithm 1 helps to select the best number of clusters in the data, herein k. A right candidate is the one that yields the higher number of maximum average values over the clustering indices. To corroborate the previous guess, the standard deviation of the clustering indices for each k can be taken into account. The choice of k can be automatic or semi-automatic. In this latter case, results composed by dendrograms, average and standard deviations values of the clustering indices obtained are displayed to the user for manual inspection and further selection.

Having the best number of clusters k according to these criteria, the stability of each cluster is then assessed individual in Algorithm 2, again via a bootstrap approach [8]. This algorithm receives as input the number of clusters k, as well as the clusters themselves $\{A_1, \ldots, A_k\}$, the linkage criterion L, and the number of bootstrap samples M.

Algorithm 2 Cluster stability assessment

1: **Repeat** M times:
2: - Bootstrap sample – randomly select $\frac{3}{4}$ patients from the original data.
3: - Create a new distance matrix D' for the bootstrap sample.
4: - Perform agglomerative clustering on D' with L, which outputs a dendrogram Z'.
5: - Obtain a collection of k clusters $\{B_1, \ldots, B_k\}$ by cutting the dendrogram Z'.
6: - Let $j = 1$.
7: **While** $j \leq k$:
8: - Let $\tau_j^* = \max_{i=1,\ldots,k} \tau(A_j, B_i)$.
9: - Let $\gamma_j^* = \max_{i=1,\ldots,k} \gamma(A_j, B_i)$.
10: - Let $\eta_j^* = \max_{i=1,\ldots,k} \eta(A_j, B_i)$.
11: - Let $j = j + 1$.
12: Evaluate statistics of the M computations for each analyzed cluster.

The algorithm starts with resampling: for each bootstrap sample, a dendrogram Z' is obtained by performing agglomerative clustering on it (Steps 2-4). Then, a collection of k clusters $\{B_1, \ldots, B_k\}$ is obtained by cutting the dendrogram Z' (Step 5). From Step 6 to 11, as proposed by Mucha [8], three different measures are computed for each cluster A_j, with $1 \leq j \leq k$, namely, τ_j^* (Jaccard), γ_j^* (rate of recovery) and η_j^* (Dice). These indices provide a measure of similarity between cluster A_j and its most similar cluster in $\{B_1, \ldots, B_k\}$. Finally, in Step 12, the stability of the retrieved clusters are assessed by computing the average values of τ_j^*, γ_j^* and η_j^*, and by analyzing the corresponding standard deviations.

As discussed in [8], it is difficult to fix an appropriate threshold to consider a cluster as stable. Therefore, we followed a rule of thumb and considered stable clusters the ones that yield high average values (close to one) and low standard deviations of τ_j^*, γ_j^* and η_j^*.

Algorithm 3 presents the overall proposed method to obtain clusters from PE sequences. It receives as input the raw data, the scoring system SS, temporal penalty T_p, and gap related parameters ($g_{\min}$, $g_{\max}$ and g_{istep}) required by the TNW, the number of bootstrap samples M, for Algorithm 1 and Algorithm 2, the linkage criterion L, the minimum $K_{\min}$ and the maximum $K_{\max}$ number of clusters.

Algorithm 3 AliClu

1: Pre-process raw data to obtain PE sequences.
2: Let $g = g_{\min}$.
3: **While** $g \leq g_{\max}$:
4: - Perform pairwise alignment using TNW algorithm with PE sequences, SS, T_p and g as input.
5: - Convert similarity matrix S into a distance matrix D.
6: - Run Algorithm 1 with D, M, L, $K_{\min}$, and $K_{\max}$ as input.
7: - Let $g = g + g_{istep}$.
8: Perform consensus decision on the number of clusters given the results from different gaps g.
9: Run Algorithm 2 to assess cluster stability with the best k clusters $\{A_1, \ldots, A_k\}$, L, and and M as input.

The initial step of the algorithm pre-processes the raw data to produce PE sequences (Step 1). The gap penalty of the TNW algorithm is then set to range over g_{min} to g_{max}, with incremental steps of g_{istep} (Step 2 and Step 7). For each value of the gap penalty g, pairwise temporal alignment using TNW is performed, which outputs a similarity matrix S (Step 4). Then, S is converted into a distance matrix D (Step 5). Clustering is then performed by running Algorithm 1 (Step 6).

When the cycle from Step 3 to 7 ends, there are several results to explore, one for each number of clusters ($K_{\min}$,..., $K_{\max}$) and gap penalties ($g_{\min}$ to $g_{\max}$ with g_{istep} incremental steps). In Step 8, the final number of clusters k is obtained from these results. As said before, if an automatic procedure is chosen, the final number of clusters k retrieved in this step is the one with the highest occurrence of higher average values over the clustering indices. In this case, the chosen gap penalty g is the one that yields the best average values of the clustering indices for the final number of clusters. In the semi-automatic option the full results for different k and g – including the dendrograms, average and standard deviations values of the clustering indices – are displayed to the user, which then decides the final number of clusters k and gap parameter g to be further used. In Step 9, the stability of the retrieved clusters is assessed by running Algorithm 2.

Results

Synthetic datasets

We first evaluate AliClu using synthetic datasets, which provides a proof of concept in a controlled scenario where the true cluster labels are known a priori, being easy to ascribe the merit of the method. Synthetic datasets consisted of temporal sequences generated by *continuous-time Markov chains* in a variety of parameter settings.

We concluded that AliClu successfully found the correct clusters in more than 80% of the cases, for datasets containing two well-separated clusters. Moreover, the linkage method that produced the best results for the agglomerative clustering was Ward's method, the reason why it was adopted in the remaining of the experimental results. The complete study of the AliClu behavior on each of the synthetic problems is available in the Additional File, along with all the details regarding sequence generation and clustering evaluation.

The Reuma.pt Database

We then assessed AliClu in biologic therapies switching in *rheumatoid arthritis* (RA) patients in a real-life longitudinal cohort – the Reuma.pt database [13].

Reuma.pt [13] is a Portuguese nationwide database developed by the Portuguese Society of Rheumatology. It stores EMRs of rheumatoid patients with structured and narrative data, with the goal of monitoring disease progression and assuring treatment effectiveness and

safety. In this study, we focus on patients with *rheumatoid arthritis* (RA) being treated with biologic therapies in one center. Retrieved data includes 426 patients diagnosed with RA, followed-up regularly, more or less every three/six months, in a total of 9305 medical appointments.

The RA is an immunomediated inflammatory rheumatic disease that causes pain and swelling in the wrist and small joints of the hand and feet. Treatments for RA can mitigate these symptoms, prevent joint damage, and provide a better quality of life to the patients. Traditional therapies consist of using conventional *disease-modifying antirheumatic drugs* (DMARD), used in monotherapy or in combination. When patients fail to respond to conventional DMARDs, modern biologic therapies are tried. Unlike conventional DMARDs, biologic ones are made using biotechnology. Biologics are genetically engineered to act like natural proteins in the human immune system.

The goals of RA treatment are to induce disease's remission by controlling inflammation. This would relieve symptoms, prevent joint and organ damage, improve physical function and overall well-being, and reduce long-term complications. It is crucial to identify the most effective RA treatment early in disease progression. In this regard, we used AliClu for the analysis of biologic therapy switching, where PE sequences are built by interspersing biologic drugs, coded as letters, along with their durations. The optimized clusters allow to stratify RA patients based on their temporal therapy profile and to identify common features for those groups. Patients entering new biologic therapies can then take profit from these insights.

Clustering of biologic therapy switches

Data of the 426 RA patients concerning biologic therapy switches was pre-processed from Reuma.pt database to build PE sequences. Figure 2 presents statistics regarding the number of biologic drugs taken by patients. Almost 60% of the patients had only one biologic drug recorded (no switches). Patients that have taken five or more drugs are rare; three patients have taken five, two have taken six, and other two seven different treatments. We stress that when switching therapies, a patient never goes back taking the previous biologics drug.

For this particular dataset, the following drugs were: A - Etanercept; B - Infliximab; C - Rituximab; D - Adalimumab; E - Anacinra; F - Abatacept; G - Tocilizumab; H - Golimumab. These correspond to distinct therapeutic active principles and are prescribed in different stages of the disease.

Having the PE sequences, Algorithm 3 is run with $K_{\max} = 30$; all other input parameters are set to their default values: the scoring system is 1 for match and -1.1 for mismatch of the drug representation, the temporal penalty $T_p = 0.25$, and the number of bootstrap samples $M = 1000$. Moreover, in this experiment, AliClu is used in a semi-automatic manner (Step 12 of Algorithm 1 and Step 8 of Algorithm 3 are prompted to user input).

We concluded that Ward's linkage leads to superior results in terms of clustering indexes and clinical information, and also that a gap penalty of $g = 0.7$ and a temporal penalty of $T_p = 0.25$ corresponds to balanced choice with respect to the other input parameters. It is noteworthy that these choices are data dependent and have a proof-of-concept principle since a full analysis and optimization of the clustering parameters would be out of the scope of the present work.

Figure 3 shows the dendrogram obtained when using this parameter set, i.e., $g = 0.7$ and temporal penalty $T_p = 0.25$. The average values of the five clustering indices obtained with Algorithm 1 are presented in Table 1.

Three of the measures, namely AR, FM, and Jaccard, indicate the existence of 26 clusters; AW favours $k = 25$ and AR ties $k = 25$, 26 and 27. In this case, not all average values point to the same number of clusters k and, therefore, a more careful and refined analysis is required.

We complemented this analysis with the standard deviation of the AR presented in Figure 4. The minimum AR standard deviation value is achieved for $k = 25$, which, combined with the information provided in Table 1 and Figure 4, leads to the selection of 25 clusters

The stability of the 25 clusters was then assessed through the median, average and standard deviation of η^*, τ^* and γ^* (Table 2). As expected, the three statistic values of η^* are always smaller than those of τ^* and γ^*. For some clusters, the medians and averages of the three measures are not as high as desirable to consider the clusters stable. Moreover, the median and averages of τ^* and γ^* do not agree in all clusters. Notwithstanding, in clusters $20, 21, 22, 23, 24$, and 25 (also those with more observations), those values agree and they are relatively high to consider them stable.

Clusters visualization

Visualization is an essential task in any clustering process, providing an intuitive way for validating clusters. Due to the characteristics of the clustered PE sequences, we propose a graph representation that allows summarizing the information regarding the sequences that belong to a given cluster. Therein, each node represents a biologic drug symbol ("A" to "H", and "Z" described above) and each edge represents a therapy switching (from one biologic drug to another). A special symbol "Z" marks the end of the sequence, signaling that from that point on there is no information regarding the therapy success or failure. A value is given on top of an edge amounting for the median of the times between the corresponding drug switches in that cluster.

The color of an edge elicits the transition probability from one biologic drug to another. This probability is computed by counting the number of times a switch occurs divided by the total number of transitions in that cluster. A gray scale is used in the edges in this regard. A darker edge means that the switch between the linked biologic drugs appeared frequently on that cluster.

The clusters with higher stability correspond to easily interpretable therapy profiles, including monotherapies (no switches). For example, clusters with only Etanercept (A; cluster 25 – 101 patients), and Infliximab (B; cluster 24 – 46 patients), but also with minor or no switches for the majority of the patients in that group. This includes Adalimumab (D; cluster 23 – 37 patients) where some patients have a switch to Golimumab (H), and vice-versa (cluster 20 – 19 patients). These clusters are represented in Figure 5. Less stable clusters may also provide relevant clinical information regarding the longitudinal profile of the therapy. For example, Cluster 14 (with 10 patients), defines a more complex structure of therapy switches form an initial Etanercept (A) treatment and a final administration of Tocilizumab (G), with some patients being administered with Adalimumab (D) and Rituximab (C).

Conclusions

We propose AliClu, a method that combines temporal sequence alignment and agglomerative hierarchical clustering to find groups in longitudinal data given by sequences of symbols and numeric values. AliClu includes a clustering validation strategy based on

bootstrap and in several clustering indexes, such as (adjusted) Rand, Fowlkes–Mallows, Jaccard, and adjusted Wallace, to choose the best number of groups to consider for each particular dataset. The stability of the obtained clusters is then assessed through resampling and using the Jaccard, the rate of recovery, and Dice indexes. AliClu can either be run entirely automatically or in a semi-automatic way, which requires user input regarding the chosen parameters. The final clusters are depicted in graphs where each node represents a symbol; each edge (a state switch) has one number corresponding to the median time, and a weight representing the switching estimated conditional probability.

AliClu was tested in synthetic data generated with continuous-time Markov chain models, being able to separate the sequences generated with different parameters. AliClu was run in the Rheumatic Diseases Portuguese Register (Reuma.pt), the national database for all the rheumatic patients treated with biological agents. In particular, rheumatoid arthritis (RA) patients' therapy information, including the sequence of drugs taken and their duration, was used as an input. The procedure allowed to stratify RA patients in a clinically relevant way by creating groups of similar treatment profiles. The clusters obtained depict the treatment switches between different drugs along with their median duration times and probabilities.

AliClu provides a strategy, validation, and visualization procedure for the automatic clustering of temporal sequence data, with promising applications for patient stratification using Electronic Medical Records (EMR) data.

Availability and requirements

Project name: AliClu
Project home page: https://github.com/sysbiomed/AliClu
Operating system(s): Platform independent
Programming language: Python
Other requirements: Python3 (in Linux or Windows) and Anaconda (in Mac OS)
License: Free
Any restrictions to use by non-academics: None

Abbreviations

EMR: Electronic Medical Records **TNW:** Temporal Needleman-Wunsch **PE:** prefix-encoded **AR:** adjusted Rand **FM:** Fowlkes and Mallows **AW:** adjusted Wallace **RA:** rheumatoid arthritis **DMARD:** disease-modifying antirheumatic drugs.

Declarations

Ethics approval and consent to participate

Reuma.pt was approved by the National Data Protection Board (Comissão Nacional de Proteção de Dados – CNPD, Portugal) and by the Ethics Committee of Centro Hospitalar Lisboa Norte (CHLN) - Hospital de Santa Maria (HSM), Lisbon, Portugal. Patients signed Reuma.pt's written consent.

Consent for publication

Not applicable.

Availability of data and material

AliClu is available at https://github.com/sysbiomed/AliClu. Data from Reuma.pt are not publicly available. Synthetic data is provided along with AliClu to ease its use.

Competing interests

SV is member of the Editorial Board of BMC Bioinformatics. KR, HC, and AMC declare that they have no competing interests.

Funding
The authors acknowledge funding the Portuguese Foundation for Science and Technology (Fundação para a Ciência e a Tecnologia - FCT) under contracts INESC-ID (UID/CEC/50021/2019) and IT (UID/EEA/50008/2019), projects PREDICT (PTDC/CCI-CIF/29877/2017), PERSEIDS (PTDC/EMS-SIS/0642/2014) and NEUROCLINOMICS2 (PTDC/EEI-SII/1937/2014). The funders had no role in the design of the study, collection, analysis and interpretation of data, or writing the manuscript.

Author's contributions
KR implemented the algorithms, performed the computational experiments and wrote the first draft of the manuscript (all authors made the required updates). HC provided the data, clinical insights and interpretation. AMC and SV conceived the study, supervised the research, generated the final results and manuscript. All authors contributed to the final draft, read and approved the final version of the manuscript.

Acknowledgments
We acknowledge all Reuma.pt contributors.

Author details
[1] Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, Avenida Rovisco Pais, 1 - Torre Norte Piso 10. 1049-001 Lisboa, Portugal. [2] CEDOC, EpiDoC Unit, NOVA Medical School, National School of Public Health, Universidade NOVA de Lisboa, Rua do Instituto Bacteriológico, nº 5 Lab 2.9., 1150-082 Lisboa, Portugal. [3] INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Rua Alves Redol 9, 1000-029 Lisboa, Portugal.

References
1. Syed, H., Das, A.K.: Temporal Needleman-Wunsch. In: IEEE International Conference on Data Science and Advanced Analytics, DSAA, pp. 1–9 (2015)
2. Needleman, S.B., Wunsch, C.D.: A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins. Journal of Molecular Biology **48**, 443–453 (1970)
3. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech and Signal Processing **26**, 43–49 (1978)
4. Zhou, F., la Torre, F.D.: Canonical time warping for alignment of human behavior. In: Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009, pp. 2286–2294 (2009)
5. Kulkarni, K., Evangelidis, G., Cech, J., Horaud, R.: Continuous action recognition based on sequence alignment. International Journal of Computer Vision **112** (2014)
6. Fischer, B., Roth, V., Buhmann, J.M.: Time-series alignment by non-negative multiple generalized canonical correlation analysis. BMC Bioinformatics **8**(10), 4 (2007)
7. Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O.P., Tiwari, A., Er, M.J., Ding, W., Lin, C.-T.: A review of clustering techniques and developments. Neurocomputing **267**, 664–681 (2017)
8. Mucha, H.-J.: On validation of hierarchical clustering. In: Decker, R., Lenz, H.-J. (eds.) Advances in Data Analysis, pp. 115–122. Springer, Berlin, Heidelberg (2007)
9. M. Rand, W.: Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association **66**, 846–850 (1971)
10. Hubert, L., Arabie, P.: Comparing partitions. Journal of Classification **2**(1), 193–218 (1985)
11. B. Fowlkes, E., Mallows, C.: A method for comparing two hierachical clusterings. Journal of The American Statistical Association **78**, 553–569 (1983)
12. Wallace, D.L.: A method for comparing two hierachical clusterings: Comment. Journal of The American Statistical Association **78**, 569–576 (1983)
13. Canhão, H., Faustino, A., et al., F.M.: Reuma.pt - The Rheumatic Diseases Portuguese Register. Acta Reumatologica Portuguesa **36(1)**, 45–56 (2011)

Figures

Figure 1 The proposed approach. First, raw data is pre-processed to obtain PE sequences. Then, pairwise sequence alignment is performed and a similarity matrix S is obtained. Next, S is converted into a distance matrix D. Agglomerative clustering is then performed with this distance matrix D. Validation of the clustering results is accomplished via a bootstrapping approach. In the end, retrieved clusters are analysed by the clinicians.
[width=0.9]images/Figure1.pdf

Figure 2 Percentage of biologic drugs taken by Rheumatoid Arthritis (RA) patients. Almost 60% of the patients only had one biologic drug. Patients that have taken more than five biologic drugs are rare; three patients have taken five, two patients have taken six, and other two seven biologic drugs.
[width=0.9]images/Figure2.png

Additional File
Additional File — Supplementary Information

Figure 3 Dendrogram of the agglomerative hierarchical clustering of Rheumatoid Arthritis (RA) patients. Dendrogram of Ward's method hierarchical clustering with gap penalty $g = 0.7$ and temporal penalty $T_p = 0.25$. Twenty five clusters were selected based on the analysis of the clustering indices and clinical interpretation.

[width=0.9]images/Figure3

Figure 4 Standard deviation of AR versus the number of clusters. Standard deviation of AR versus number of clusters for dendrogram in Figure 3. There is a downward trend of the standard deviation when increasing the number of clusters. The minimum value is attained with 25 clusters.

[width=0.9]images/Figure4

Figure 5 Cluster Visualization. Graph representation of selected clusters based on stability measures and clinical interpretation. Drug codes: A - Etanercept; B - Infliximab; C - Rituximab; D - Adalimumab; E - Anacinra; F - Abatacept; G - Tocilizumab; H - Golimumab. Z - Follow-up/end.

[width=0.9]images/Figure5.pdf

Table 1 Average values of five clustering indices for the dendrogram of Figure 3.

k	Rand	AR	FM	Jaccard	AW
2	0.876	0.744	0.897	0.827	0.704
3	0.852	0.675	0.789	0.658	0.661
4	0.872	0.689	0.780	0.644	0.644
5	0.897	0.705	0.773	0.632	0.759
6	0.920	0.751	0.802	0.672	0.768
7	0.935	0.780	0.820	0.699	0.771
8	0.931	0.753	0.796	0.662	0.700
9	0.950	0.801	0.830	0.712	0.782
10	0.966	0.855	0.875	0.779	0.861
11	0.969	0.863	0.881	0.789	0.857
12	0.973	0.876	0.892	0.805	0.878
13	0.975	0.883	0.897	0.814	0.883
14	0.979	0.897	0.909	0.833	0.914
15	0.982	0.910	0.920	0.852	0.917
16	0.985	0.925	0.933	0.875	0.931
17	0.987	0.932	0.940	0.887	0.937
18	0.988	0.936	0.943	0.893	0.939
19	0.989	0.940	0.946	0.899	0.944
20	0.988	0.937	0.943	0.893	0.933
21	0.989	0.938	0.945	0.895	0.939
22	0.990	0.942	0.948	0.901	0.940
23	0.991	0.946	0.951	0.907	0.961
24	0.992	0.953	0.958	0.919	0.965
25	**0.993**	0.958	0.962	0.926	**0.966**
26	**0.993**	**0.959**	**0.963**	**0.929**	0.964
27	**0.993**	0.958	0.962	0.928	0.960
28	0.992	0.955	0.959	0.923	0.952
29	0.992	0.952	0.957	0.920	0.945
30	0.991	0.940	0.947	0.903	0.924

Table 2 Stability of the 25 clusters for Ward's method, $g = 0.7$, and $T_p = 0.25$.

Cluster Nb. (# patients)	τ^* median	η^* median	γ^* median	τ^* average	η^* average	γ^* average	τ^* std	η^* std	γ^* std
1 (4)	0.475	0.298	0.625	0.475	0.298	0.625	0.389	0.185	0.177
2 (4)	0.750	0.429	0.750	0.750	0.429	0.750	0.000	0.000	0.000
3 (5)	0.083	0.077	0.200	0.083	0.077	0.200	0.000	0.000	0.000
4 (5)	0.400	0.271	0.600	0.400	0.271	0.600	0.283	0.147	0.000
5 (5)	0.275	0.215	0.500	0.275	0.215	0.500	0.035	0.022	0.141
6 (6)	0.833	0.455	0.833	0.833	0.455	0.833	0.000	0.000	0.000
7 (7)	0.741	0.423	0.786	0.741	0.423	0.786	0.164	0.054	0.101
8 (7)	0.307	0.233	0.500	0.307	0.233	0.500	0.080	0.047	0.101
9 (7)	0.643	0.390	0.643	0.643	0.390	0.643	0.101	0.037	0.101
10 (8)	0.688	0.407	0.688	0.688	0.407	0.688	0.088	0.031	0.088
11 (9)	0.542	0.347	0.611	0.542	0.347	0.611	0.177	0.075	0.079
12 (9)	0.389	0.269	0.444	0.389	0.269	0.444	0.236	0.124	0.157
13 (10)	0.352	0.256	0.400	0.352	0.256	0.400	0.145	0.080	0.141
14 (10)	0.489	0.311	0.550	0.489	0.311	0.550	0.337	0.156	0.354
15 (13)	0.513	0.330	0.577	0.513	0.330	0.577	0.254	0.112	0.163
16 (13)	0.472	0.321	0.577	0.472	0.321	0.577	0.039	0.018	0.054
17 (14)	0.571	0.358	0.571	0.571	0.358	0.571	0.202	0.082	0.202
18 (16)	0.719	0.416	0.719	0.719	0.416	0.719	0.133	0.045	0.133
19 (17)	0.309	0.235	0.353	0.309	0.235	0.353	0.084	0.049	0.083
20 (19)	0.716	0.416	0.737	0.716	0.416	0.737	0.119	0.041	0.149
21 (20)	0.791	0.440	0.825	0.791	0.440	0.825	0.154	0.048	0.106
22 (32)	0.696	0.410	0.719	0.696	0.410	0.719	0.056	0.019	0.088
23 (37)	0.791	0.441	0.811	0.791	0.441	0.811	0.104	0.032	0.076
24 (46)	0.728	0.420	0.728	0.728	0.420	0.728	0.108	0.036	0.108
25 (101)	0.777	0.437	0.777	0.777	0.437	0.777	0.007	0.002	0.007

Figure1

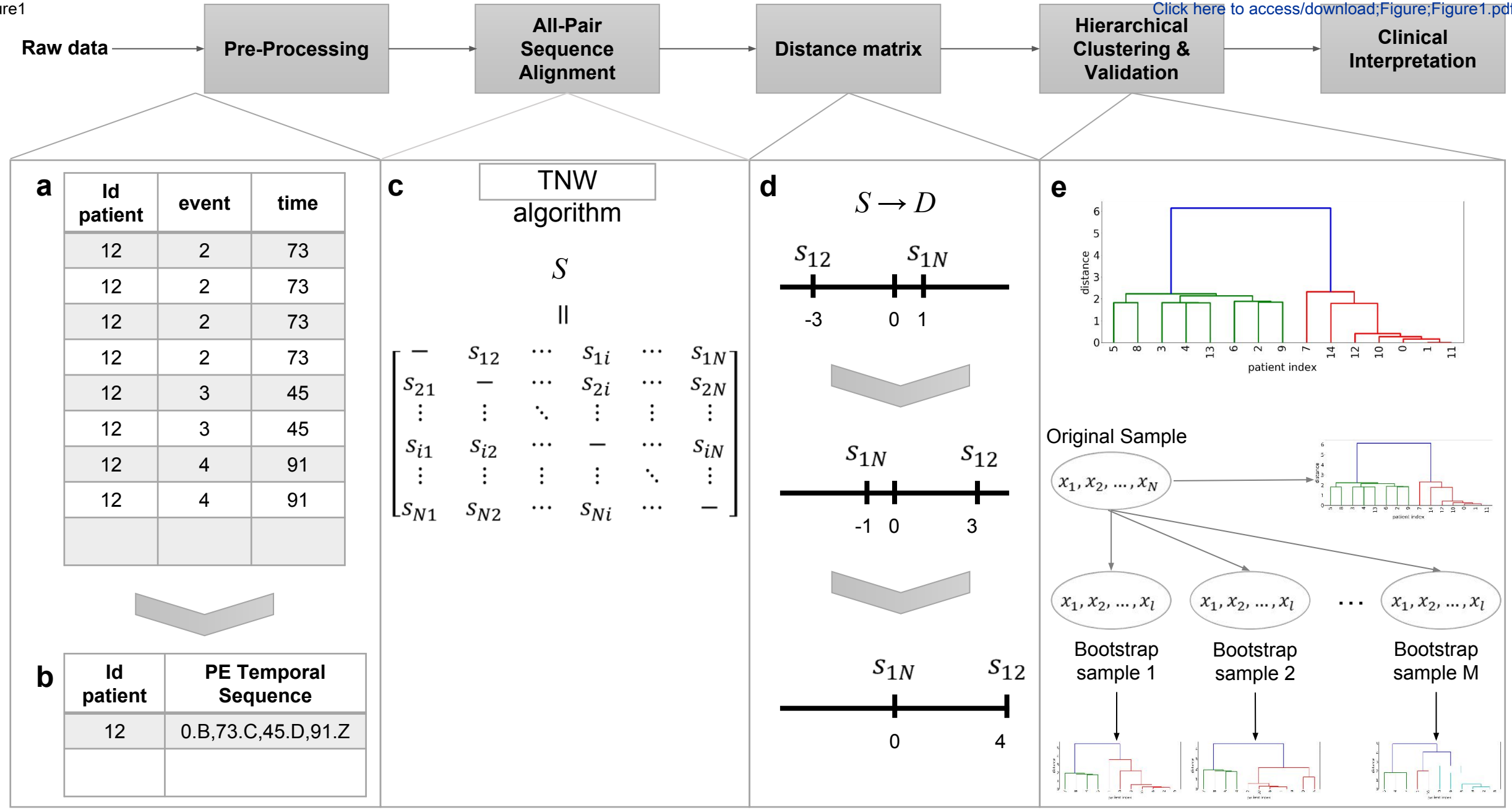
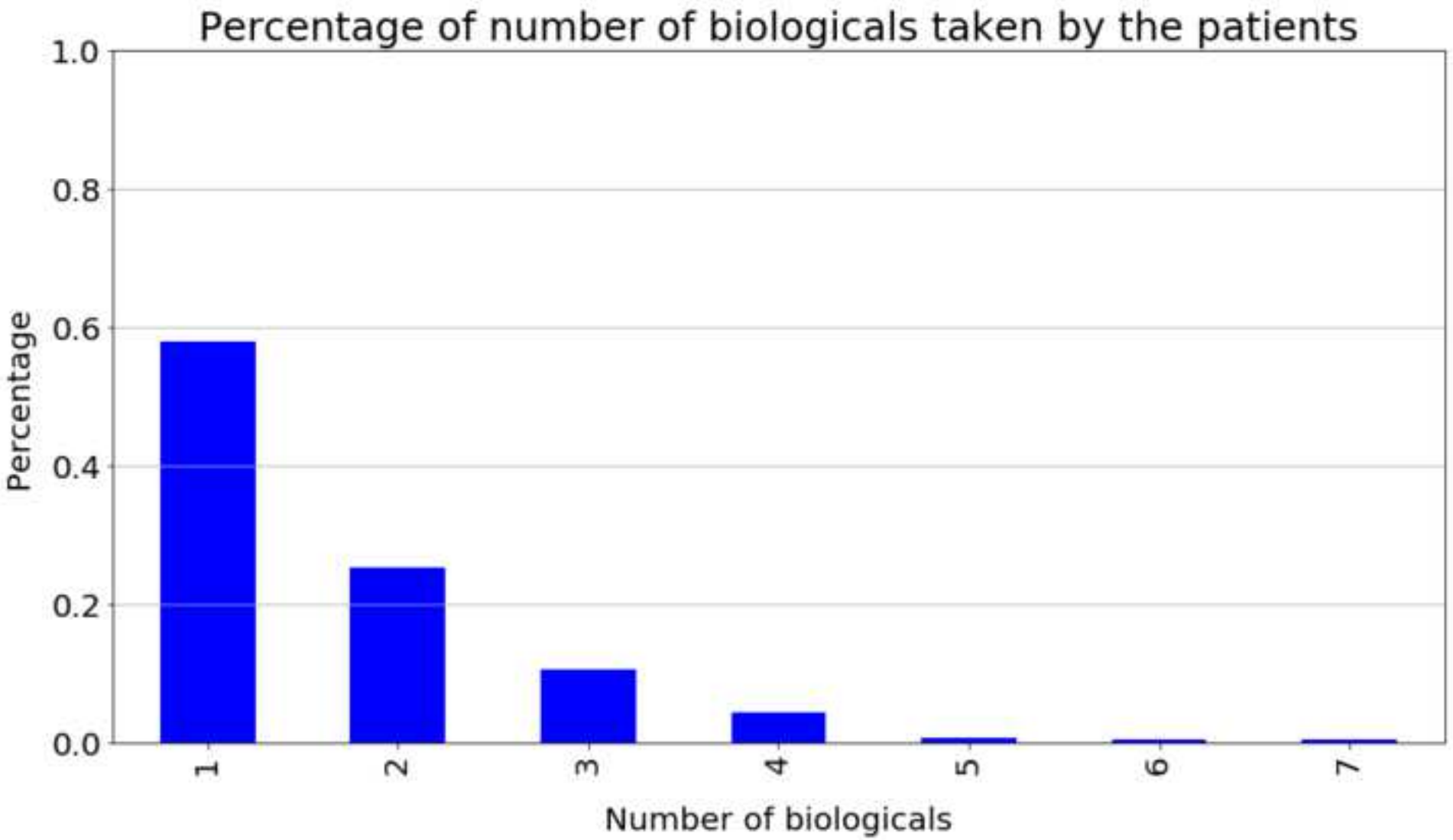
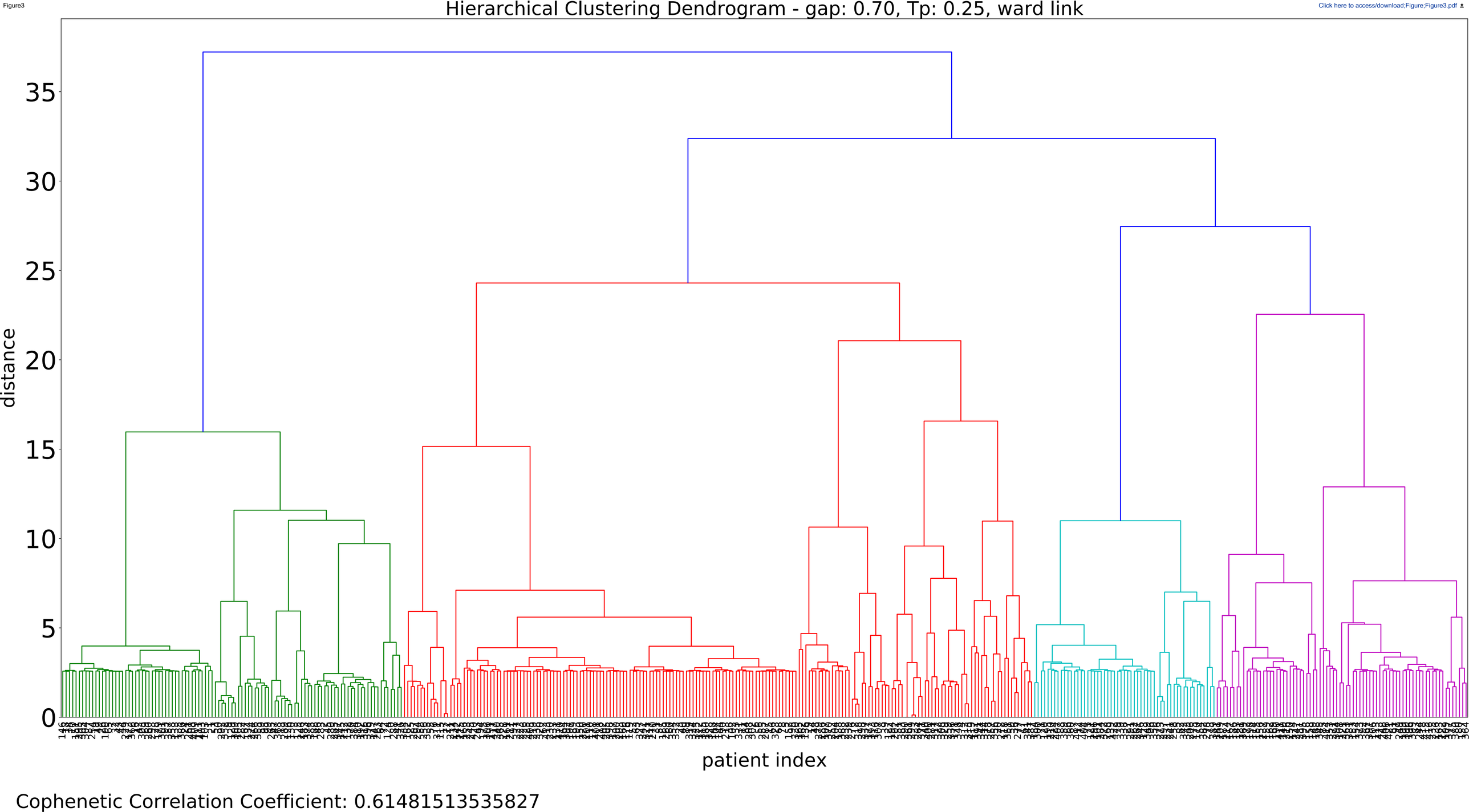
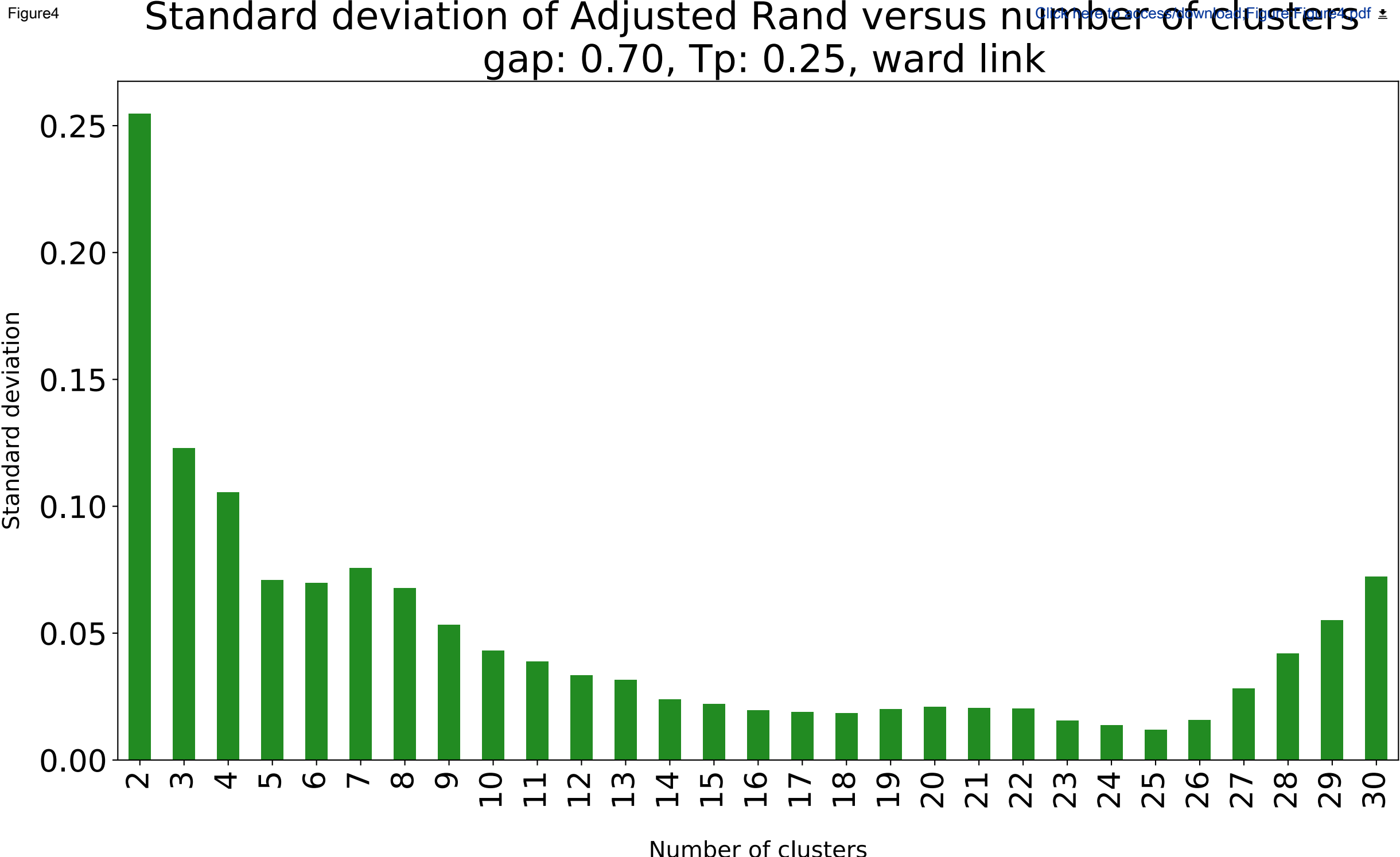


Figure2

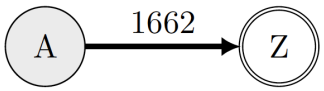
Click here to access/download;Figure;Figure2.png



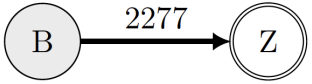




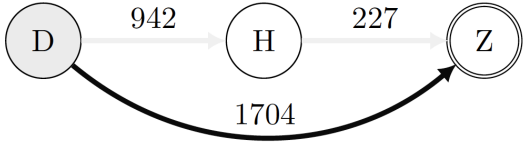
Cluster 25 - 101 patients:



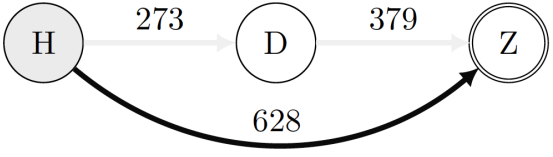
Cluster 24 - 46 patients:



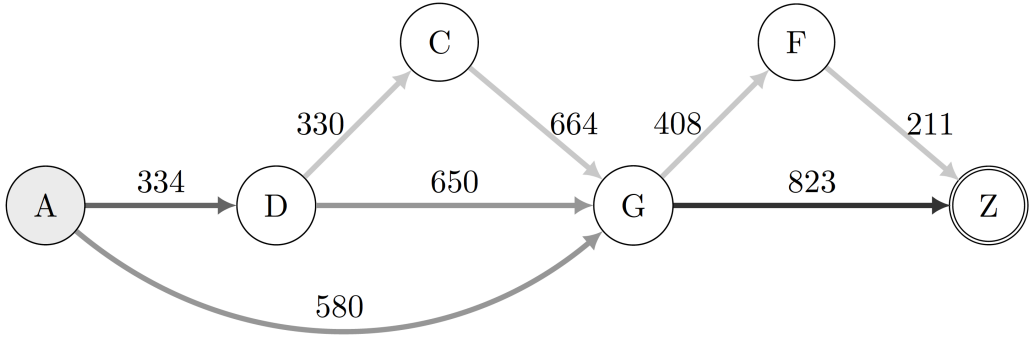
Cluster 23 - 37 patients:



Cluster 20 – 19 patients:



Cluster 14 – 10 patients:



Additional File