Journal of Mathematics in Industry
a SpringerOpen Journal

RESEARCH Open Access

Pattern recognition in data as a diagnosis tool

Ana Carpio[1,2*], Alejandro Simón[1], Alicia Torres[1] and Luis F. Villa[3]

*Correspondence:
ana_carpio@mat.ucm.es
[1]Departamento de Matemática Aplicada, Universidad Complutense de Madrid, Plaza de Ciencias 3, 28040, Madrid, Spain
[2]Gregorio Millan Barbarny Institute for Modelling and Simulation in Fluid dynamics, Nanoscience and Industrial Mathematics, Avenida de la Universidad 30, 28911, Leganés, Spain
Full list of author information is available at the end of the article

Abstract

Medical data often appear in the form of numerical matrices or sequences. We develop mathematical tools for automatic screening of such data in two medical contexts: diagnosis of systemic lupus erythematosus (SLE) patients and identification of cardiac abnormalities. The idea is first to implement adequate data normalizations and then identify suitable hyperparameters and distances to classify relevant patterns. To this purpose, we discuss the applicability of Plackett-Luce models for rankings to hyperparameter and distance selection. Our tests suggest that, while Hamming distances seem to be well adapted to the study of patterns in matrices representing data from laboratory tests, dynamic time warping distances provide robust tools for the study of cardiac signals. The techniques developed here may set a basis for automatic screening of medical information based on pattern comparison.

Keywords: Pattern classification; Hyperparameter selection; Plackett-Luce models; Hamming distance; Dynamic time warping distance; Wasserstein distance; Medical diagnosis; Systemic lupus erythematosus; Electrocardiogram

1 Content

One of the most challenging and essential tasks performed by the healthcare professionals is diagnosing a disease or patient's condition. Medical diagnoses are based on symptoms, signs, medical history and complementary examinations, which define the patient's clinical picture. This information is collected in datasets combining different kinds of data. Laboratory analyses, for instance, are stored as matrices of numerical values, representing blood counts, metabolic, ionic, enzyme, hormone, protein, vitamin and antibody tests, or other variables of interest recorded at different times. Specific vital signals are recorded as time sequences, such is the case of electrocardiograms, for example. More sophisticated medical imaging devices (X-rays, magnetic resonance imaging, etc) visualize the status of organs or body parts by means of a series of images. As the amount of data collected grows in size, the development of algorithms allowing medical staff to automatically screen the information contained in the stored data becomes essential. This task faces additional challenges, since hospital records usually display non homogeneous data measured over irregular time periods.

The applicability of machine learning techniques in specific medical contexts involving large data amounts is an active area of research nowadays. Neural networks, for instance,

© The Author(s) 2022. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

are sometimes used for image-based diagnosis [1, 2], while unsupervised and supervised classification techniques [3] are nowadays exploited to study the role of genes in sickness [4, 5] and to investigate the response to treatment [6]. However, the amount of data available in many medical situations is scarce [7–9]. In the search for a diagnosis, one can resort to different procedures. Here, we will focus on diagnosis by pattern comparison. In principle, a diagnosis could be made by comparing the patient's clinical profile with that of different diseases and selecting the most similar ones. However, even if a patient has a disease, he does not need to display all symptoms, and many signs are common to different diseases. In this respect, the selection of key variables and the introduction of adequate comparison criteria become essential issues.

Here, we develop mathematical tools for automatic screening of data stored in the form of numerical time sequences or matrices and illustrate the results in two medical contexts: diagnosis of systemic lupus erythematosus (SLE) patients and identification of cardiac abnormalities. This article is organized as follows. First, we introduce distances which are helpful to compare different kinds of data. Then, we explain how to adapt Plackett-Luce models for rankings to select the most appropriate distances or hyperparameters when classifying data. We adapt these ideas to study data from anonymized SLE patients from two viewpoints. We initially take a clustering approach to find the onset of flares periods that require immediate medical attention. Next, we switch to a supervised classification approach to find daily patterns in the data representing a known sickness profile. Finally, we discuss applications to classify electrocardiogram patterns by comparing them with typical abnormal profiles and present our conclusions.

2 Distances for data

Consider a matrix $\mathcal{M} = (m_{i,j})$, $i = 1,\dots,I$, $j = 1,\dots,J$, containing data for I variables at J different times. We denote by $\mathbf{m}^{(k)}$, $k = 1,\dots K$, either the rows or the columns of this matrix, which we wish to compare. The *Euclidean distance* provides a standard tool to that purpose. Given two vectors $\mathbf{m}^{(1)}$ and $\mathbf{m}^{(2)}$ in a N dimensional space, their Euclidean distance is

$$d\big(\mathbf{m}^{(1)},\mathbf{m}^{(2)}\big) = \big\|\mathbf{m}^{(1)} - \mathbf{m}^{(2)}\big\|_2 = \left(\sum_{n=1}^{N}\big|m_n^{(1)} - m_n^{(2)}\big|^2\right)^{1/2}. \tag{1}$$

We can define the *Hamming distance* for simple vectors whose components are taken from alphabets with 2 or 3 elements. The Hamming distance between two vectors $\mathbf{m}^{(1)}$, $\mathbf{m}^{(2)}$ whose components are -1, 0 or 1, for instance, is the number of positions at which the symbols are different. This distance is often used in communications to detect and correct errors in codes [10].

The *dynamic time warping distance* (TWD) provides an alternative method to quantify the similarity between two general data vectors [11, 12]. It has become a standard tool for analysing video, audio, and graphics data, with applications such as speech or signature recognition. The TWD algorithm seeks an optimal match between two given vectors (time sequences, for instance), subject to some conditions:

- Every index from the first vector is matched with one or more indices from the second vector and vice versa.
- The first index from the first vector is matched with the first index from the second vector (it does not need to be a unique match).

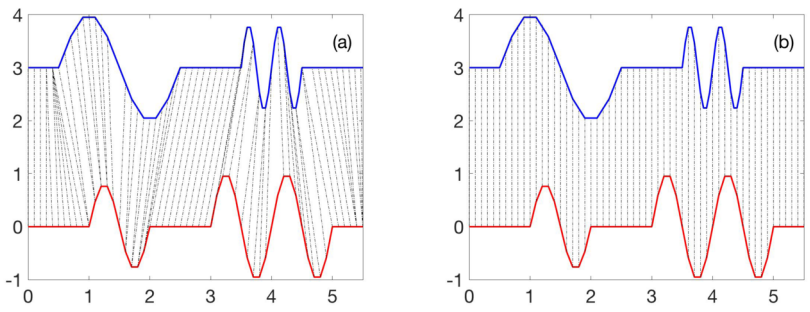

Figure 1 Dynamic time warping distance (a) versus Euclidean distance (b)

- The last index from the first vector is matched with the last index from the second vector (it does not need to be a unique match).
- The mapping of the indices from the first vector to indices from the second vector is monotonically increasing, and vice versa.

The idea is illustrated in Fig. 1. We define a cost by computing the sum of absolute differences of vector values for each matched pair of indices. The optimal match minimizes the cost subject to the above conditions.

The TWD can be calculated by dynamic programming based on cumulative distances as follows. Let $\mathbf{m}^{(1)}$ and $\mathbf{m}^{(2)}$ be two vectors with N and L components, respectively. Let $d(x,y)$ be a distance between real numbers, for instance, the absolute value of the difference $d(x,y) = |x-y|$. We create a matrix $\mathrm{TWD}(n,\ell)$ for $n = 0,\dots,N$, $\ell = 0,\dots,L$. We set $\mathrm{TWD}(n,0) = \infty$ for $n = 0,\dots,N$, $\mathrm{TWD}(0,\ell) = \infty$ for $\ell = 0,\dots,L$, and $\mathrm{TWD}(0,0) = 0$. Then, for n from 1 to N and ℓ from 1 to L, we calculate

$$\begin{aligned}\mathrm{TWD}(n,\ell) = {} & d\big(m_n^{(1)}, m_\ell^{(2)}\big) \\ & + \min\big\{\mathrm{TWD}(n-1,\ell), \mathrm{TWD}(n,\ell-1), \mathrm{TWD}(n-1,\ell-1)\big\},\end{aligned} \tag{2}$$

where $\mathrm{TWD}(n,\ell)$ represents the distance between subsequences $m_1^{(1)},\dots,m_n^{(1)}$ and $m_1^{(2)},\dots,m_\ell^{(2)}$. The final result $\mathrm{TWD}(N,L)$ defines the distance between the two vectors. Figure 1 illustrates the difference between the Euclidean (1) and the dynamic time warping (2) distances.

The *Earth Mover's distance* (EMD) is a more general concept, which quantifies the minimum cost of turning a collection of numeric values into another [13]. More precisely, the EMD between two vectors $\mathbf{m}^{(1)}$ and $\mathbf{m}^{(2)}$ formed by N and L real values, respectively, is

$$\mathrm{EMD}\big(\mathbf{m}^{(1)}, \mathbf{m}^{(2)}\big) = \frac{\sum_{n=1}^{N}\sum_{\ell=1}^{L} f_{n,\ell} d_{n,\ell}}{\sum_{\ell=1}^{L}\sum_{n=1}^{N} d_{n,\ell}}, \tag{3}$$

where $d_{n,\ell} = |m_n^{(1)} - m_\ell^{(2)}|$ is the ground distance, and $f_{n,\ell}$ minimizes the cost $\sum_{n=1}^{N}\sum_{\ell=1}^{L} f_{n,\ell} \times d_{n,\ell}$ subject to the constraints

$$f_{n,\ell} \geq 0, \quad 1 \leq n \leq N, 1 \leq \ell \leq L,$$

$$\sum_{n=1}^{N} f_{n,\ell} \leq 1, \quad 1 \leq \ell \leq L, \qquad \sum_{\ell=1}^{L} f_{n,\ell} \leq 1, \quad 1 \leq n \leq N,$$

$$\sum_{n=1}^{N}\sum_{\ell=1}^{L} f_{n,\ell} = \min(N, L).$$

This distance tracks patterns in the compared vectors, regardless of their location.

Instead of tracking vectors, we may as well compare whole matrices. This can be done by a general family of distances, called Wasserstein type distances. As the EMD, their calculation is posed as optimal transport problems. Optimal transport plays crucial roles in many areas, including image processing and machine learning. Fast algorithms to calculate Wasserstein-1 distances between distributions defined on a grid are proposed in [14]. Given two 2D probability distributions, or two images, ρ^0 and ρ^1, they seek a transport plan $f(x)$ from one to the other with minimal cost

$$\min_f \int_x \left\| f(x) \right\|_p dx \tag{4}$$

such that $\text{divergence}_h(f) = \rho^0 - \rho^1$ under a zero-flux boundary condition. Here, p can be 1,2 or infinity, so that $\|f(x)\|_p$ represents the L^1, L^2 or L^∞ norms of $f(x)$, respectively, h is the grid step size and divergence_h a divergence operator defined on the grid, see [14] for details.

3 Distance and hyperparameter selection based on models for rankings

To evaluate which distance or hyperparameter choice performs better when classifying data of a different nature, we resort to Plackett-Luce type models for rankings. We can implement different approaches depending on the ranking structure.

3.1 Bayesian approach with Plackett-Luce ranking models

Consider a situation in which we have to judge the performance of N procedures $\{\text{proc}_1, \ldots, \text{proc}_N\}$ on D sets of data $\{\text{dat}_1, \ldots, \text{dat}_D\}$ of a similar nature. For instance, we wish to know which distance and hyperparameter choices perform best to extract specific information from the datasets through learning algorithms. To do so, we select D datasets for which the information we seek is known, apply the N procedures and quantify the error in the outcomes.

Excluding the possibility of ties, when we apply all the procedures to a dataset we obtain a ranking $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_N)$, that is, a permutation of $(1, \ldots, N)$ where $\rho_i = j$ indicates that the i-th procedure is ranked at the j-th position [15, 16]. The procedure ranked first performs better than the procedure ranked second, and so on. To each ranking, we associate another permutation of $(1, \ldots, N)$ which defines an ordering $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_N)$, where $\sigma_i = j$ indicates that the j-th procedure is ranked at the i-th position.[1] Orderings and rankings are related by $\sigma_{\rho_i} = i$ and $\rho_{\sigma_i} = i$, that is, the ordering is the inverse of the ranking.

We can describe the probability of observing a ranking $\boldsymbol{\rho}$ by means of the *Plackett-Luce* (PL) model, a distribution over rankings expressed in terms of the associated orderings $\boldsymbol{\sigma}$ and parametrized by a vector $\mathbf{w} = (w_1, \ldots, w_N)$, $w_i \geq 0$, $i = 1, \ldots N$ [15, 16]. The conditional

[1] Some authors use a different terminology [17], speaking of 'performance' and 'ranking', a 'ranking' being in fact an 'ordering'.

PL probability of observing $\boldsymbol{\sigma}$ given $\mathbf{w}$ is

$$P_{\mathrm{PL}}(\boldsymbol{\sigma}|\mathbf{w}) = \prod_{i=1}^{N} \frac{w_{\sigma_i}}{\sum_{k=i}^{N} w_{\sigma_k}}. \tag{5}$$

The probability of having $\sigma_1 = j$, that is, the j-th procedure being the best, is $\frac{w_j}{\sum_{k=1}^{N} w_k}$ [16]. Assuming $\sum_{k=1}^{N} w_k = 1$, the parameters of the model represent the probability of each of the different procedures under consideration being the best. Given a set of observations $\mathcal{O}$ formed by D independent orderings $\boldsymbol{\sigma}$, the conditional probability of observing this set is the product of the probabilities of each of the orderings:

$$P(\mathcal{O}|\mathbf{w}) = \prod_{\boldsymbol{\sigma}\in\mathcal{O}} P_{\mathrm{PL}}(\boldsymbol{\sigma}|\mathbf{w}). \tag{6}$$

We identify the model parameters $\mathbf{w}$ from the observations by Bayesian inference. By Bayes's theorem

$$P(\mathbf{w}|\mathcal{O}) = \frac{P(\mathbf{w}) \cdot P(\mathcal{O}|\mathbf{w})}{P(\mathcal{O})}, \tag{7}$$

where $P(\mathbf{w})$ is a probability distribution summarizing our prior knowledge on $\mathbf{w}$. By simplicity, we assume that the sum of its components is 1. We choose a Dirichlet distribution $P(\mathbf{w}) = \mathrm{Dir}(\mathbf{w};\boldsymbol{\alpha})$ [17], though other choices are possible [16]. The Dirichlet distribution of order $N \geq 2$ has a density function

$$f(w_1,\ldots,w_N;\alpha_1,\ldots,\alpha_N) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{i=1}^{N} w_i^{\alpha_i - 1}, \qquad B(\boldsymbol{\alpha}) = \frac{\prod_{i=1}^{N} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{N} \alpha_i)}, \tag{8}$$

where $\sum_{i=1}^{N} w_i = 1$, $w_i \geq 0$ for $i = 1,\ldots,N$, and Γ denotes the Γ function. In the absence of additional information, we choose a uniform distribution for $\boldsymbol{\alpha}$, that is, $\alpha_i = \alpha = 1$, $i = 1,\ldots,N$. This is the *flat Dirichlet distribution*, equivalent to a uniform distribution over the open standard $(N-1)$ - simplex [18].

With these definitions, we can sample the unnormalized posterior distribution $Q(\mathbf{w}) = P(\mathbf{w}) \cdot P(\mathcal{O}|\mathbf{w})$ using Markov Chain Monte Carlo (MCMC) methods [19]. Note that the unknown scaling factor $P(\mathcal{O})$ in (7) is not needed for MCMC sampling. From the samples, we obtain information on the most likely values for $\mathbf{w}$ with quantified uncertainty, thus, we infer which procedure is the best.

MCMC methods produce a chain of N-dimensional states $\boldsymbol{w}^{(0)} \longrightarrow \boldsymbol{w}^{(1)} \ldots \longrightarrow \boldsymbol{w}^{(k)} \ldots$ which evolve to be distributed according to the target distribution [19]. We first sample an initial state $\boldsymbol{w}^{(0)}$ from the prior distribution (8), and then move from one state $\boldsymbol{w}^{(k)}$ to the next $\boldsymbol{w}^{(k+1)}$ guided by a transition operator. We have selected a *Hamiltonian Monte Carlo transition* operator because it usually samples the distributions faster than other samplers, such as Metropolis-Hastings [20]. We set $U(\boldsymbol{w}) = Q(\mathbf{w})$ (the probability to be sampled), $K(\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^t\boldsymbol{p}$ and construct the Hamiltonian $H(\boldsymbol{w},\boldsymbol{p}) = U(\boldsymbol{w}) + K(\boldsymbol{p})$. We draw a random Gaussian moment $\boldsymbol{p}^{(0)} \in \mathbb{R}^N$ from a multivariate normal distribution $N(0,\mathcal{I})$, where the covariance matrix is the identity. Then, the transition proceeds as follows:

- Given $\boldsymbol{w}^{(k)}$ and $\boldsymbol{p}^{(k)}$, we use them as initial values to solve the Hamiltonian equations $\frac{\partial w_j}{\partial t} = \frac{\partial H}{\partial p_j}$, $\frac{\partial p_j}{\partial t} = -\frac{\partial H}{\partial w_j}$, $j = 1, \ldots, N$, by a leap frog algorithm for L steps of size δ. The final values define $\boldsymbol{w}^*$ and $\boldsymbol{p}^*$, the new proposed states for the chain.
- We calculate $\alpha = \min(1, \exp(-U(\boldsymbol{w}^*) + U(\boldsymbol{w}^{(k)}) - K(\boldsymbol{w}^*) + K(\boldsymbol{w}^{(k)}))$.
- We generate a random number u from a uniform distribution $\mathcal{U}(0,1)$. If $u \leq \alpha$, we accept the proposed states and set $\boldsymbol{w}^{(k+1)} = \boldsymbol{w}^*$. Otherwise, we keep $\boldsymbol{w}^{(k+1)} = \boldsymbol{w}^{(k)}$.

The final chain $\boldsymbol{w}^{(0)}, \boldsymbol{w}^{(1)}, \ldots, \boldsymbol{w}^{(M)}$ provides our set of samples, after discarding some initial states as a 'burn in' period. Once a large enough set of sampled weights is available, the sample with the highest probability furnishes the most likely value for $\mathbf{w}$. Histograms or Boxplots built with the additional samples quantify the uncertainty in this prediction.

The above procedure excludes the presence of ties in the rankings under study. When some of the items to be ranked perform equally well on a dataset, the ordering takes the form $\boldsymbol{\sigma} = \{C_1, C_2, \ldots, C_J\}$, containing sets C_j with one or more items equally ranked, the items in each set being ranked higher than the next. When the ranking matrix contains ties, we can break randomly the ties T times, to obtain T new ranking matrices without ties. We apply to each of them the previous procedure and average the results provided for each of them to obtain average weights w_i indicating the prevalence of each ranked item. This procedure quantifies the tendency of the ranked item to be not only the first, but to appear in relevant positions, see [21] for a detailed study.

3.2 Davidson-Luce models with ties

The *Davidson-Luce* model [22, 23] works directly with rankings that may contain tied items. As said above, the associated orderings take the form $\boldsymbol{\sigma} = \{C_1, C_2, \ldots, C_J\}$ where C_j are sets of tied items, the items of each set being ranked higher than the next. For instance, consider 6 choices labelled as $\{1, 2, 3, 4, 5, 6\}$. Choices $C_1 = \{1, 3\}$ perform equally well. Choices $C_3 = \{2, 5, 6\}$ too, but worse that C_1. An ordering $\boldsymbol{\sigma} = \{C_1, C_2, C_3\}$ with $C_2 = \{4\}$ would represent that observation when C_2 is better than C_3 one but worse than C_1.

In this framework, the probability of the choices $A = \{a_{i_1}, \ldots, a_{i_\ell}\}$ being equally preferred to the remaining choices in a set of r total choices, $1 \leq \ell \leq r$, is proportional to

$$q(A) = \delta_{|A|} \left(\prod_{j \in A} \alpha_j \right)^{1/|A|}, \tag{9}$$

where $|A| = \ell$ is the number of elements in A, α_j is the worth of choice a_j and $\delta_{|A|} \geq 0$ is a parameter representing the prevalence of ties of order $|A|$ for $|A| > 1$. When $|A| = 1$, δ_1 can be set arbitrarily equal to 1. The remaining worths and prevalences are parameters to be fitted. The worths α_j are interpreted as follows: conditional upon the outcome being an outright win for one choice, the probabilities for each choice to be the winner are in the ratios indicated by the α_j [22, 23]. When $\sum_j \alpha_j = 1$, the worths represent the probability of each choice being the best assuming there is no tie in the first position. The probability of preferring an item a_j from a set S is $\alpha_j / \sum_{k \in S} \alpha_k$, $j \in S$. The parameters representing tie prevalences $\delta_{|A|}$ are interpretable in terms of tie probabilities between items of equal worth. For instance, δ_2 is related to the probability that two items of equal worth tie in the first position, conditional upon not having 3-way or higher ties for first place. To be more precise, that probability is $\delta_2/(2 + \delta_2)$, see [22, 23].

Then, the probability of an ordering $\boldsymbol{\sigma} = \{C_1, C_2, \ldots, C_J\}$ allowing for ties up to order K is given by

$$p_{\mathrm{DL}}(\boldsymbol{\sigma}) = \prod_{j=1}^{J} \frac{q(C_j)}{\sum_{k=1}^{\min(|A_j|,K)} \sum_{S\in\binom{A_j}{k}} q(S)}, \tag{10}$$

where $q(S)$ is defined in (9), A_j is the set of items from which C_j is chosen and $\binom{A_j}{k}$ is the set of all possible choices of k items from A_j. The parameter K equals the maximum number of ties encountered, so that $\delta_n = 0$ when $n > K$. The Plackett-Luce model is a particular case in which ties are forbidden, $\delta_{|A|} = 0$ for $|A| > 1$. Given a matrix of independent observations $\mathcal{O}$, its probability is defined as the product of the probabilities (10) of each observation.

In this case, the worths and tie prevalences are fitted by Broyden-Fletcher-Goldfarb-Shanno quasi-Newton optimization algorithms (BFGS, L-BFGS) for functions involving a large number of parameters, more complicated to implement than the Bayesian approach. Moreover, some parameter tuning or additional prior information may be needed to converge to a fit [22, 23]. Once they are calculated, we know the probabilities that an item is best conditional upon being one absolute win or also including different kinds of ties. It is the sum of the probabilities of all the rankings placing that item in the top set. For instance, considering $\{a_1, a_2, a_3\}$ we have 7 possible outcomes for the first position: 3 absolute winners ($\{a_1\}$, $\{a_2\}$, $\{a_3\}$), 3 double ties ($\{a_1, a_2\}$, $\{a_1, a_3\}$, $\{a_2, a_3\}$) and one triple tie ($\{a_1, a_2, a_3\}$), with probabilities proportional to α_1, α_2, α_3, $\delta_2(\alpha_1\alpha_2)^{1/2}$, $\delta_2(\alpha_1\alpha_3)^{1/2}$, $\delta_2(\alpha_2\alpha_3)^{1/2}$, $\delta_3(\alpha_1\alpha_2\alpha_3)^{1/3}$. The DL model provides 7 probabilities that sum up to 1 in those proportions. The probability of a_1 being best including ties is the probability of having the outcomes $\{a_1\}$, $\{a_1, a_2\}$, $\{a_1, a_3\}$ or $\{a_1, a_2, a_3\}$. This strategy gives more value to being placed in the first position most of the time than to being usually placed at good positions. The R package [23] implements this scheme and provides standard errors indicating the precision of the estimates, which are not always useful due to overlaps.

4 Application to laboratory data from systemic lupus erythematosus

Laboratory tests constitute a cornerstone of the diagnosis process for many health disorders: blood counts, enzyme, hormone, metabolic, ionic, protein, vitamin and antibody levels are measured at different times. Developing automatic tools to trace key features in the resulting numeric matrices can help to interpret the data, specially when dealing with disorders which are difficult to diagnose. Consider systemic lupus erythematosus (SLE), for instance, a chronic autoimmune disease in which the immune system attacks healthy tissues by mistake [24]. This attack causes inflammation and, in some cases, permanent tissue damage. SLE is difficult to identify and treat appropriately because many symptoms are non specific and change throughout the course of the disease. Lupus patients go through periods of illness, called flares, and periods of wellness, called remission. The symptoms during flares vary. It is essential to be able to distinguish early when the patient is transitioning from remission to flares, and what factors are causing it, to timely administer adequate treatments. As it happens with other diseases, SLE diagnosis relies heavily on alterations observed in laboratory tests.

Next, we propose a procedure to extract automatically relevant features from time series of laboratory tests. The idea is as follows. First, we apply clustering techniques to locate

relevant time frames and then seek specific patterns in the data to select a possible diagnosis, resorting to adequate distances. Finally, we will illustrate the process on anonymized data from SLE patients.

4.1 Clustering time records

Let us consider a matrix $\mathcal{M}$ collecting the results of laboratory tests for a patient during a series of days. We monitorize I variables (rows) at J times (columns). To analyze time variations, we normalize the data as follows. For each variable $i = 1, \ldots, I$, we calculate the mean μ_i and the standard deviation σ_i of the corresponding row $m_{i,j}$, $j = 1, \ldots, J$. Then, we construct the normalized matrix with elements

$$\tilde{m}_{i,j} = \frac{m_{i,j} - \mu_i}{3\sigma_i}, \quad i = 1, \ldots, I, j = 1, \ldots, J, \tag{11}$$

most of which lie in $[-1, 1]$, see Fig. 2. This procedure defines Normalization 1. This normalization is useful to visualize time periods in the data. Figure 2, for instance, suggests a period of strong instability between days 810-868.

Datasets are often incomplete, as Fig. 2 shows. For day clustering purposes, we adopt the convention of suppressing rows (variables) for which more than half the recordings are missing. Otherwise, we fill the gaps with the average of the two contiguous values. We also remove rows corresponding to variables which do not change noticeably during the studied period [21]. The outcome is represented in Fig. 3 for the remaining 60 variables. The 5 variables comprised between Glucose and Nitrites are removed because they remain essentially zero. Nevertheless, their presence does not affect the clustering results. One could additionally remove days for which less than half the variables are recorded, as it happens for day 850. However, the measurements recorded just before and after it are taken in a narrow time gap compared to other measurements present in the datasets, thus, interpolating between the two neighbouring days seems reasonable. For days 526 and 565, the variation between the values recorded before and after them is small. Keeping or removing these days does not change significantly the analysis either.

We identify automatically different time stages in the evolution of the patient by means of clustering techniques such as K-Means (KM) [25], Hierarchical clustering (HC) [26] and density-based spatial clustering of applications with noise (DBSCAN) [27], applied to the normalized matrix (11) with the Euclidean distance and different hyperparameter selections. We set as initial day of a flares period the first day of the shortest cluster for KM and HC and the smallest outlier for DBSCAN with different hyperparameters [21]. For ease of the reader, we briefly describe these clustering algorithms in Appendix A.

To select the most adequate clustering procedure for the kind of data we are working with we first study a collection of datasets which have already been diagnosed. We have considered anonymized data from 19 patients containing one flares period whose starting day is known. The strategy is as follows:

- For each dataset, we normalize the raw data matrix according to (11).
- We apply to the columns KM and HC, setting different numbers of clusters, and also DBSCAN for different choices of ε (smallest distance for points to be considered neighbours) and MP (minimum number of points to be considered a cluster).
- For each dataset, we compare the prediction obtained for the onset of the flares period with the known onset day. Then, we rank the procedures according to the difference between both.

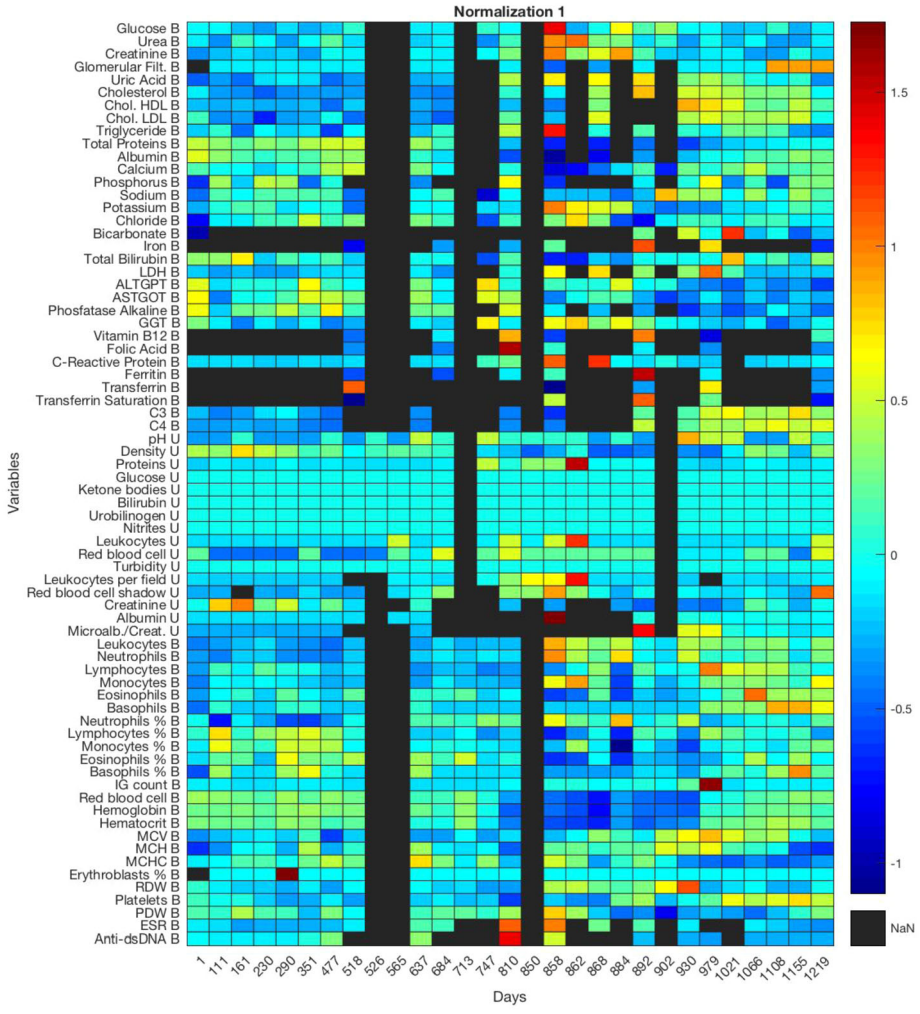

Figure 2 Heatmap representing the outcome of Normalization 1, which allows us to establish stages in the time evolution. Cold and warm colors indicate variables below and above the average values for this patient, respectively. Black boxes indicate missing data. Variables ending in B correspond to blood tests, while those ending in U correspond to urine tests

- If the resulting matrix of rankings does not contain ties, we use the Plackett-Luce model (6)-(8) to obtain the worths $\mathbf{w}$ representing the probability of each clustering procedure to provide the best estimate of the onset of flares and to quantify the uncertainty in this result.
- If the resulting matrix of rankings contains ties, we can break the ties randomly and average the results provided by (6)-(8) for a large number of attempts, as detailed in [21]. Alternatively, we can use the Davidson-Luce model (9)-(10) to obtain the worths $\mathbf{w}$ and tie prevalences δ. Then we use them to estimate the probability of each clustering procedure being the best counting all the possible outcomes (alone or tied).

We have implemented this scheme setting the number of clusters equal to 3, 4, 5 in KM and HC, and DBSCAN with $\varepsilon = 3, 3.5$ and $MP = 3$. The resulting ranking matrices contain ties. However, our ranking analysis tends to select HC with 4 clusters as the best strategy, followed by HC with 3 clusters. This study is further detailed in Appendix B. For the data

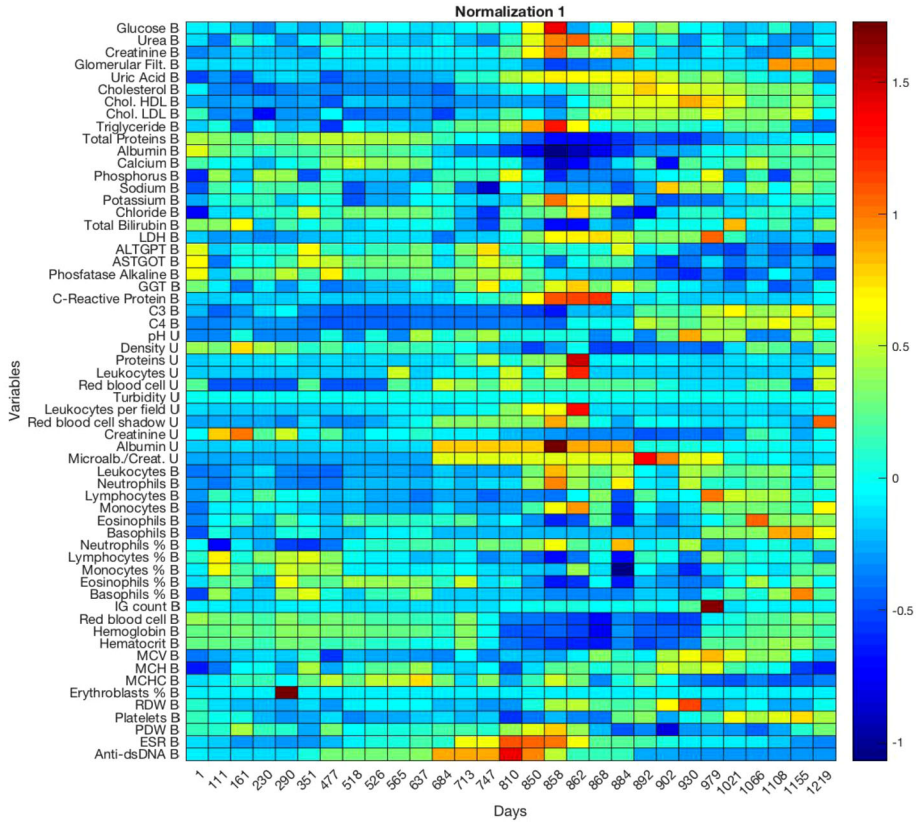

Figure 3 Heatmap representing the 60 variables remaining in Normalization 1 after suppressing 7 variables for which more than half the recordings are missing and 5 variables which do not change noticeably during the studied period, interpolating to fill the remaining gaps

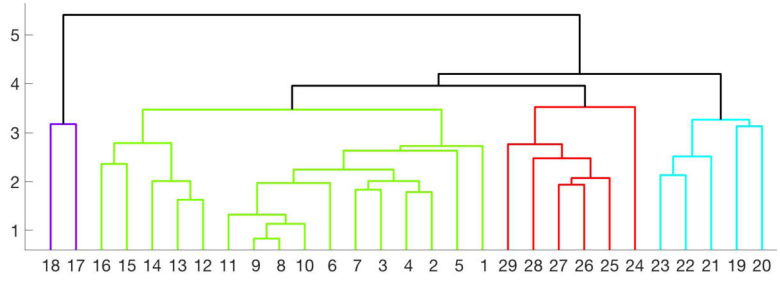

Figure 4 Dendrogram constructed by hierarchical clustering using the Euclidean distance and average linkage. Numbers 1-29 correspond to the 29 observations reported in Fig. 2, keeping the same order

represented in Fig. 2, this strategy singles out automatically days 858-862 (flares crisis), and splits the remaining days into three periods 1-850 (pre-flares), 868-930 (treatment), 930-1219 (post-flares), see Fig. 4.

Detecting distinguished days, or blocks of similar days, allows us to focus our study on the status of the variables on the specific days which mark transitions from remission to flares, or on specific periods to identify types of flares and observe the response to treatment. For example, if we focus in the suspected flares period in Fig. 2, we notice a sharp

increase in anti-dsDNA antibodies on day 810 and a sharp decrease of the red blood cell counts, hemoglobin and hematocrit measurements after it. We also observe an increase in creatinine in blood and the presence of leukocytes in urine. Searching for specific patterns in the data during the selected days may help us to propose a diagnosis, as we explain in the next section.

4.2 Daily pattern search

To this purpose, we renormalize the raw data matrix $\mathcal{M}$ in a different way. Assuming normality ranges for the variables are known, we construct a normalized matrix with elements $\hat{m}_{i,j}$ replacing $m_{i,j}$ with 1 or −1 depending on whether the original values are above or below the normality ranges of the i-th variable, and 0 when they are in the normality range, see Fig. 5. When the information stored is binary, either 'true' or 'false', we replace it by either 1 (positive, true) or 0 (negative, false). This defines Normalization 2.

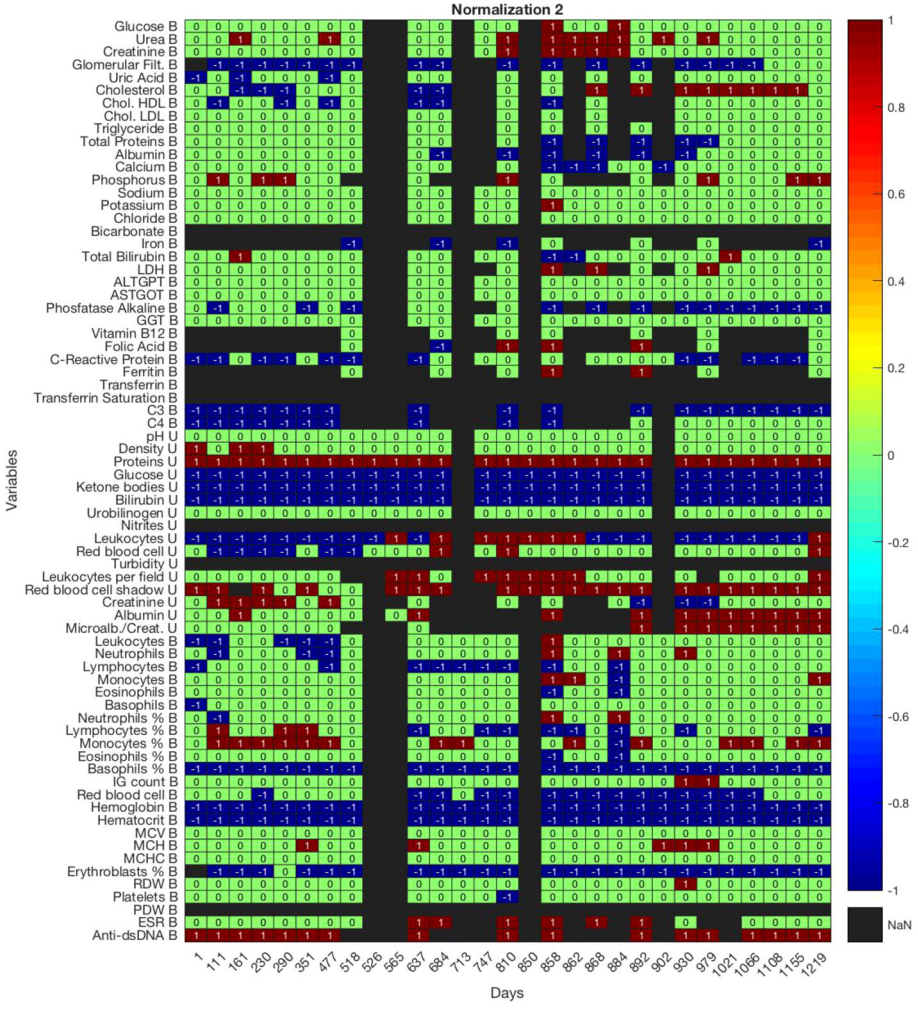

Figure 5 Heatmap representing the outcome of Normalization 2, allowing us to identify relevant illness patterns. We appreciate which variables usually lie outside laboratory ranges of normal values (1 above, −1 below, 0 inside) and which ones exit this range occasionally. Black boxes indicate missing data and variables for which we do not have the normality range (such is the case when the entire row is black)

Many disorders can be characterized as simple −1, 0, 1 patterns, or combinations thereof:

- Normocytic anemia: −1 Hemoglobine in blood, 0 Mean corpuscular velocity.
- Hypocomplementemia C3, −1 C3 in blood, and Hypocomplementemia C4, −1 C4 in blood. They usually indicate active SLE.
- Leukopenia: −1 Leukocytes in blood. It usually indicates active SLE or infection or medication effects.
- Neutropenia: −1 Neutrophiles in blood. It usually indicates active SLE or infection or medication effects.
- Lymphopenia: −1 Lymphocytes in blood. It usually indicates active SLE or infection.
- High Anti-dsDNA antibodies: +1 Anti-dsDNA, the most relevant antibody for SLE.
- High Creatinin: +1 Creatinin in blood. It usually indicates kidney failure.
- Pathological proteinuria: +1 Proteins in urine. It usually indicates inflammation of the renal glomerulus or tubular damage.
- Hyperglycemia: +1 Glucose in blood. It may be due to the use of corticosteroids, or a sign of Diabetes Mellitus.
- Hypercholesterolemia: +1 Cholesterol LDL in blood. Risk factor to be treated, as associated with hypertension and corticosteroids increases cardiovascular risk.
- Suspicion of hemophagocytic syndrome: −1 Hemoglobin or −1 Leukocytes or −1 Neutrophils or −1 Lymphocytes or −1 Platelets in blood, plus large ferritin counts, and large triglycerid or small fibrinogen counts in blood. It is a complication of sustained severe inflammatory states, including SLE.

The combinations of −1, 0, 1 patterns observed in distinguished days mark specific types of evolution.

To select the most adequate distance to this purpose, we have compared the performance of the Euclidean distance, the Earth Mover's distance and the Hamming distance to identify definite patterns formed combining 1, 0 and −1 in columns containing those digits. We propose a collection of P patterns and distribute them in the columns of a $I \times J$ matrix, filling the remaining positions with the digits −1, 0, 1. Then we calculate the distances between the columns and the reference patterns, and rank the distances according to their performance. Implementing a Plackett-Luce model ranking analysis, we conclude that the Hamming distance is by far the most efficient one. This strategy is detailed in Appendix C.

Once this fact has been established, the 'automatic diagnosis' strategy is as follows:

- For a given dataset, we normalize the raw data matrix to obtain a matrix involving only −1, 0, 1 according to the normality ranges of the observed variables.
- We compare the status of the variables in given columns corresponding to specific days with known sickness patterns defined by −1, 0, 1 sequences using the Hamming distance.

In this way, we can screen large datasets selecting patient profiles requesting immediate attention, while providing at the same time a simplified diagnosis that should receive further consideration from a specialist. For instance, for the dataset considered in Figs. 2 and 5, we identify SLE activity indicated by elevated anti-dsDNA and low C3 and C4 complements. We identify low hemoglobin and hematocrit levels suggesting an anemization (normocytic anemia) process as a result of illness. Alterations in the glomerular filtration rate and presence of proteins in urine indicate alterations in the kidney function too.

High creatinine values at days 810 and 858 indicate kidney damage and require immediate treatment to restore normal values.

5 Application to electrocardiogram interpretation

When a specific variable is recorded for an interval of time, the data take the form of curves. Often, typical patterns representing normal stages and abnormal stages are known. Developing automatic screening tools can assist doctors in processing such data. Consider electrocardiograms (ECG) for instance. An electrocardiogram is a graph of the electrical activity of the heart [28] representing voltage versus time, see Fig. 6. They are recorded placing electrodes on the skin, which detect small electrical changes resulting from cardiac muscle depolarization and repolarization during each heartbeat (cardiac cycle). Alterations in the ECG pattern indicate cardiac abnormalities. Numerous diagnoses are based on the observed patterns. Producing effective tools to automatically identify cardiac patterns would allow for the proper use of defibrillators when untrained emergency staff assists people going into cardiac arrest at work or leisure centers. Currently, it is hard to succeed in the absence of an expert doctor who interprets the signs.

Our goal in this section is to introduce an automatic procedure to identify alterations in patterns described by one dimensional curves. The idea is as follows. We first define a set of normalized reference curves corresponding to different known pathologies. Given another curve normalized in the same way, we compare it with the reference patterns by means of an adequate distance. The smallest distance selects a possible diagnosis. We will illustrate the process on examples taken from electrocardiography.

5.1 Electrocardiogram structure and basic alterations

Electrocardiograms repeat periodically the structure represented in Fig. 6. Different regions are designed by P, Q, R, S and T [29]:

- *P wave:* The first wave in the ECG represents atrial depolarization. Usual length is shorter than 0.11 s in adults. Usual amplitude is smaller than 0.25 mV. Its shape is smooth and rounded.

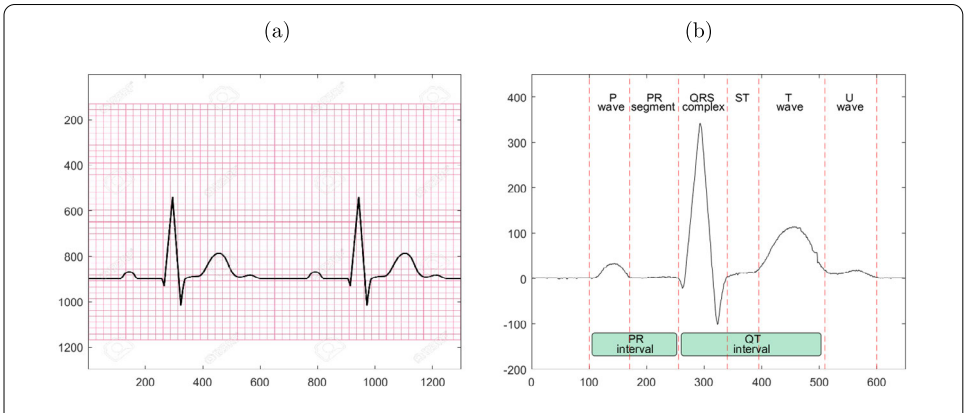

Figure 6 (a) Original ECG, represented in the standard red grid paper. Ten boxes represent 10 mm = 1 mV in the vertical direction and 0.4 s in the horizontal direction. (b) Representation of an ECG extracted from (a). The labels in the vertical and horizontal axes represent positions in the matrix defining both images and indicate the number of points defining the curve

- *QRS complex:* It is formed by a sequence of waves representing ventricular depolarization. Usual duration is about 0.06 s - 0.10 s. A small negative wave Q is followed by a large positive R wave and a small negative S wave.
- *T wave:* The next wave represents ventricular repolarization. Usual length is shorter than 0.20 s in adults. Usual amplitude is between 0.2 and 0.3 mV. Its shape is smooth and rounded.
- *U wave:* This last tiny wave is believed to represent papillary muscle repolarization. It may not be seen, and is often ignored.

The P wave and the QRS complex are separated by the PR segment, while the ST segment separates the QRS complex and the T wave. The P wave with the PR segment form the PR interval. The QRS complex, the ST segment and the T wave form the QT interval. Variations in the heart's structure and its environment (blood composition included) alter these entities. The presence, absence and size of the different ECG parts characterize different cardiac abnormalities, comprising cardiac rhythm disturbances (such as ventricular tachycardia and atrial fibrillation), perturbed coronary artery blood flow (such as myocardial infarction and myocardial ischemia), as well as electrolyte disturbances (such as hyperkalemia and hypokalemia).

We select a few representative patterns to illustrate the method:

- A *normal ECG* repeats the basic P-Q-R-S-T structure corresponding to one heartbeat at a rate between 60 and 100 beats per minute (bpm)
- *Sinus tachycardia* is a heart rate greater than 100 beats per minute. This is normal with exercise, but abnormal otherwise.
- *Sinus bradycardia* is a sinus node dysfunction with a rate under 60 beats per minute.
- *Sinus arrhythmia* is an irregular heartbeat that can be either too fast or too slow. It is characterized by variations in the P-P intervals greater than 0.12 s (from one beat to the next).
- *QT prolongation* is a sign of delayed ventricular repolarisation. This means that the heart muscle takes longer than usual to recharge between beats. It is a known side effect of a wide range of medicines. Excessive QT prolongation can trigger tachycardias and torsades de pointes (TdP).
- *Torsades de Pointes* is a ventricular tachycardia, fast and polymorphic, characterized by fluctuation of the QRS complexes around the electrocardiographic baseline. It may lead to life-threatening ventricular fibrillation.

To compare ECGs we must normalize them somehow. ECGs are usually recorded in red grid paper, see Fig. 6. Panel (a) displays a standard printed ECG. There is an established correspondence between the number of squares and the units (mm, mV, s). Such an image, or portions of the image, can be read as a matrix of a size adjusted to the resolution. From that matrix, we extract only the locations corresponding to black points forming the curve. When several ordinates correspond to the same abscissa, we keep only one ordinate: their average value. Panel (b) represents the outcome of this procedure applied to part of panel (a), indicating the different ECG regions. Additionally, we have normalized heights by setting the first point of the curve at zero. Figure 7 collects reference patterns for the abnormalities under consideration obtained from diagnosed ECGs, available in open access image datasets, such as [30]. We have considered regions defined by the same number of red squares and extracted the ECG curves by the same procedure, choosing a

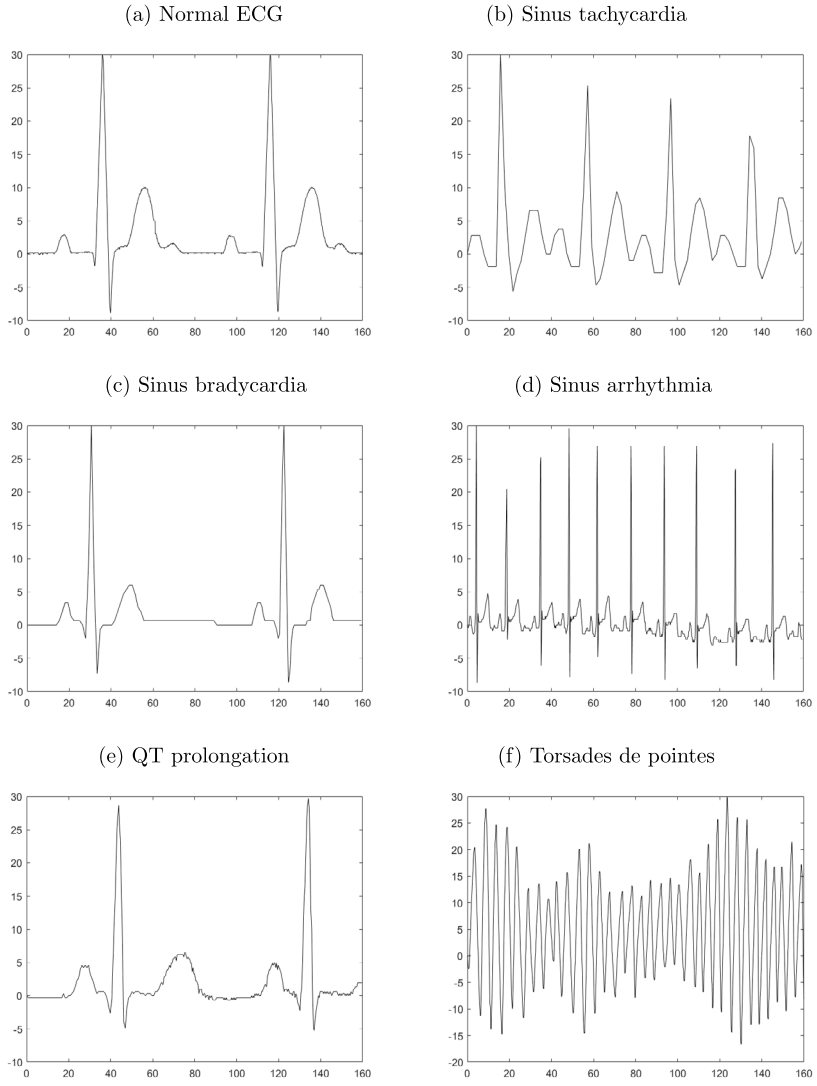

Figure 7 Normalized and nondimensionalized reference ECG patterns

smaller matrix size (a factor 10 smaller). One this is done, we need to choose appropriate distances to compare patterns.

5.2 Electrocardiogram classification

This section illustrates how to use TWD, EMD and Wasserstein distances to classify ECG patterns. Figure 8 shows a set of normalized ECGs we want to classify as displaying a kind of abnormality or another. To do so, we select a distance and calculate the distances between these ECG and the reference patterns displayed in Fig. 7. Then, we propose a diagnosis based on the smallest value obtained for each of them.

Tables 1 and 2 reproduce the distance matrices obtained for the TWD and the one dimensional EMD. Tables 3 and 4 reproduce the distance matrices for the 2D image Warssentein-1 distance with Algorithm 1M and Algorithm 2M from [14], respectively, and the 1-norm ($p = 1$). Choosing the norms for $p = 2$ or $p = \infty$ we find the same classification.

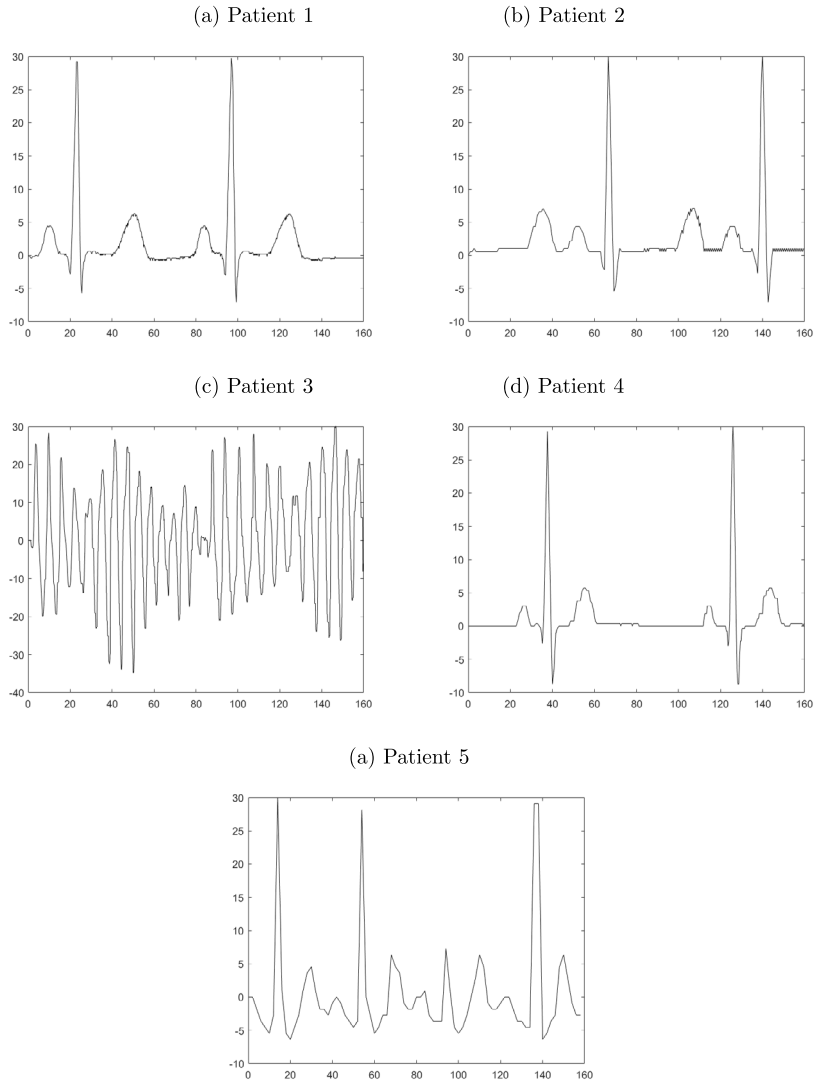

Figure 8 Normalized and rescaled ECGs to be diagnosed by comparison with the reference patterns collected in Fig. 7

To visualize the strategy's performance, we create Table 5, whose entries are 1 when patient i is correctly diagnosed by distance j and 0 otherwise. Notice that the TWD performs better, and classifies all the patients correctly. Then the Wasserstein-1 distance correctly classifies four out of five patients. The ECG of the patient who is misclassified displays bradycardia and is classified as QT prolongation, which could be confused with slow rhythms.

To better evaluate the performance of different distances, we have studied a larger collection of 100 electrocardiograms obtained by perturbing the reference patterns. We can construct ranking and observation matrices by comparing each of them with the reference ECG patterns using the TWD, EMD and Wasserstein distances. Then, we check whether the proposed diagnosis is correct or not. Implementing a Plackett-Luce model ranking analysis, we conclude that the dynamic time warping distance would be the most efficient one for this dataset. However, this synthetic dataset is quite limited. Testing the method

Table 1 Distances between ECG patterns and ECGs from patients calculated using TWD and proposed diagnosis, based on the smallest distance

Pattern\Patient	1	2	3	4	5
Normal ECG	772	414	9974	327	663
Sinus tachycardia	1807	539	10,046	413	197
Sinus bradycardia	1086	259	10,381	113	277
Sinus arrythmia	2099	1120	8550	1105	975
QT prolongation	1379	236	10,174	313	378
Torsades de pointes	10,170	9852	5939	9862	8947
Diagnosis	Normal	long QT	TdP	Bradycardia	Tachycardia

Table 2 Distances between ECG patterns and ECGs from patients calculated using EMD and proposed diagnosis, based on the smallest distance

Pattern\Patient	1	2	3	4	5
Normal ECG	175.9249	95.9059	273.9574	108.5616	79.0234
Sinus tachycardia	65.2876	79.0439	139.7507	74.7188	77.9492
Sinus bradycardia	60.6457	76.1689	172.1750	72.2178	67.4229
Sinus arrythmia	191.0266	121.2034	290.5680	108.8583	76.6855
QT prolongation	135.7548	116.9913	194.5916	104.3210	72.2278
Torsades de pointes	473.4429	309.0894	542.9015	278.8493	131.6373
Diagnosis	Bradycardia	Bradycardia	Tachycardia	Bradycardia	Bradycardia

Table 3 Distances between ECG patterns and ECGs from patients calculated using Wassertein-1 distances (Algorithm 1M in [14] with $p = 1$) and proposed diagnosis, based on the smallest distance

Pattern\Patient	1	2	3	4	5
Normal ECG	0.2543	0.2703	0.1886	0.2635	0.4157
Sinus tachycardia	0.4474	0.2907	0.3775	0.3028	0.0070
Sinusal bradycardia	0.4240	0.1817	0.3483	0.1982	0.2033
Sinus arrythmia	0.3225	0.2087	0.2032	0.2088	0.3347
QT prolongation	0.3959	0.0588	0.3167	0.0732	0.2271
Torsades de pointes	0.2796	0.1671	0.1317	0.1639	0.241
Diagnosis	Normal	long QT	TdP	long QT	Tachycardia

Table 4 Distances between ECG patterns and ECGs from patients calculated using Wassertein-1 distances (Algorithm 2M in [14] with $p = 1$) and proposed diagnosis, based on the smallest distance

Pattern\Patient	1	2	3	4	5
Normal ECG	0.2615	0.2761	0.1867	0.2678	0.4197
Sinus tachycardia	0.4596	0.2885	0.3757	0.3033	0.0070
Sinus bradycardia	0.4349	0.1824	0.3481	0.1974	0.2043
Sinus arrythmia	0.3313	0.2082	0.2024	0.2090	0.3351
QT prolongation	0.4069	0.0601	0.3153	0.0745	0.2252
Torsades de pointes	0.2791	0.1673	0.1313	0.1646	0.2410
Diagnosis	Normal	long QT	TdP	long QT	Tachycardia

on more general curves should be the object of further work, but it would require the previous obtention of an extensive collection of diagnosed authentic EGC images.

Table 5 Correct diagnosis using TWD versus correct diagnosis using EMD and Wasserstein distances

Patient	TWD	EMD	Wasserstein-1
1	1	0	1
2	1	0	1
3	1	0	1
4	1	1	0
5	1	0	1
Success Score	5	1	4

6 Conclusions

The design of automated tools for detecting abnormalities in a patient's clinical picture offers an excellent opportunity to improve clinical case management and clinical research by developing new algorithms and software.

Medical data are usually stored in the form of matrices or sequences. Time sequences of results of Laboratory tests, for instance, take the form of numerical matrices. We have discussed two different types of normalization, one related to average values for unsupervised learning, and the other related to normal ranges for supervised classification. First, we show how to use clustering techniques to distinguish periods of medical relevance in the time evolution of a SLE patient. In particular, we locate flares periods which require immediate medical attention. Then, we illustrate how to propose a possible diagnosis by seeking specific numeric patterns in each period according to the second normalization. Using Hamming distances to compare the outcome of Laboratory analyses at different days, or for different patients, one could automatically classify patient's profiles.

On the other side, recordings of vital signals, such as electrocardiograms, take the form of time sequences, usually represented as curves. We propose a strategy to identify abnormalities in recorded ECGs by comparing recorded curves to patterns representing typical anomalies. First, we introduce a normalization procedure. Then, we investigate the potential of time warping, Earth mover's and Wasserstein distances to correctly classify basic abnormality patterns, finding a better performance for the dynamic time warping distance on a synthetic dataset. Further studies would require the previous obtention of an extensive collection of diagnosed authentic EGC images. Being able to correctly classify abnormal ECG patterns in an automated way would increase the chance of survival in out-of-hospital treatments. It would allow for the proper use of defibrillators by untrained emergency staff when assisting cardiac arrest cases at work or leisure centers.

In these studies, it is essential to select the distances and hyperparameters best adapted to the datasets under consideration. We have introduced a Plackett-Luce ranking based analysis as a tool to select the most adequate distances and hyperparameters to analyze datasets with a specific structure. The techniques developed here may set a basis for automatic screening of medical information based on pattern comparison.

Appendix A: Clustering techniques

In this section we briefly recall the clustering techniques used to classify SLE data.

K-means [25] clusters data in groups in order to minimize the total intra-cluster variation, which measures the cluster compactness. Given a data cloud formed by points $\mathbf{x}_i = (x_{i,1}, \ldots, x_{i,M})$ in a M dimensional space, the intra-cluster total variation is given by $\sum_{j=1}^{J} W(C_j) = \sum_{j=1}^{J} \sum_{\mathbf{x}_i \in C_j} d(\mathbf{x}_i, \boldsymbol{\mu}_j)$, where C_j is a cluster of such points and $\boldsymbol{\mu}_j$ is the cluster centroid, for $j = 1, \ldots, J$. Each term $W(C_j)$ represents the variation within a cluster.

Here, the distance d stands for the Euclidean distance $d(\mathbf{x}_i, \boldsymbol{\mu}_j)^2 = \sum_{m=1}^{M}(x_{i,m} - \mu_{j,m})^2$. The K-means algorithm proceeds in the following steps. We fix the number of clusters k to be formed and initialize the centroids $\boldsymbol{\mu}_j$ by randomly generating k points. Next, each datum $\mathbf{x}_i$ is assigned to the centroid minimizing the Euclidean distance. Within each cluster, we set the average of the cluster points as the new centroid. These steps are repeated until the clusters do not change. K-means needs one hyperparameter to proceed: the number of clusters. There are specific criteria such as the Elbow or Silhouette methods to propose a tentative cluster number.

Hierarchical clustering produces a multilevel hierarchy, in which clusters at one level coalesce at the next level [26]. The agglomerative algorithm starts from as many clusters as data points. Nearby clusters merge to create larger ones until all the data points form a single cluster. This procedure is schematized in a dendrogram, a graph visualizing how the clusters join until they form a tree that comprises all, see Fig. 4. To evaluate the proximity of clusters and merge them, the algorithm employs 'linkage functions': 'single', 'average', 'complete', 'weighted', 'centroid', 'median', 'ward'. Both the linkage functions and the distance are hyper parameters to be selected. Here, we have fixed the Euclidean distance and implemented the linkage providing the biggest cophenetic correlation coefficient for these datasets, that is, the biggest correlation between the original distance and the cophenetic distance (the height at which clusters coalesce). Large correlation indicates that the tree is representative of our dataset. The remaining hyperparameter, that is, the height, determines the number of clusters. Cutting the tree at different heights, we select specific numbers of clusters.

The *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) algorithm [27] defines clusters in high density regions, leaving observations in low density regions outside, which eventually become anomalies. The process starts with an arbitrary data point that has not yet been classified. We find the points at a distance smaller than ε (the ε-neighbourhood). When it contains more than a minimum number of points MP, we create a new cluster with them. Otherwise, we consider that point as noise. However, this point might become part later of the ε-neighbourhood of a different point containing enough points, and, thus, belong to that cluster. If not, it remains an outlier. Thus, this algorithm can identify non convex clusters and outliers. However, finding good values for the two hyperparameters ε and MP is a nontrivial task, strongly dependant on the dataset's structure. In principle, the distance has to be selected too, but we have fixed the Euclidean distance.

Appendix B: Stages in SLE patients' medical records

This appendix summarizes some results obtained when applying the strategy described in Section 4 to 19 time series of anonymized laboratory data of patients diagnosed with SLE according to criteria of the European League against Rheumatism/American College of Rheumatology [31]. The number of variables and days varies slightly within them. All the datasets considered contained only one transition, and Hopkins criterion for them indicated the presence of relevant clusters [32]: the Hopkins statistics $H > 0.5$. Table 6 quantifies the distance between the column of the normalized heatmap selected as onset of flares after the clustering procedure and the true column corresponding to the diagnosed day. HC stands for hierarchical clustering (3C with 3 clusters, 4C with 4 clusters) and KM by K-means (with k clusters). DBSCAN uses parameters $MP = 3$ and $\varepsilon = 3$. Next,

Table 6 Distances to the transition day for the different algorithms

Transition days	KM $k=3$	KM $k=4$	KM $k=5$	HC 3C	HC 4C	DBSCAN 3,3
D15	0	2	4	0	2	0
D4	0	1	1	0	0	0
D9	0	6	6	9	9	2
D3	2	2	2	0	0	0
D1	2	2	0	2	2	0
D1	6	10	10	0	0	0
D16	6	6	9	19	19	15
D14	0	0	11	0	0	0
D5	0	1	1	0	0	0
D9	0	5	5	3	5	5
D19	0	0	16	17	0	17
D50	41	41	41	4	0	0
D1	3	0	0	0	0	0
D23	22	22	41	0	0	21
D17	0	9	9	0	0	12
D19	14	14	14	0	7	18
D11	6	0	0	0	0	4
D3	21	14	0	19	21	0
D12	11	0	0	11	0	10

Table 7 Rankings generated from the distances to transition days in Table 6

Dataset	KM $k=3$	KM $k=4$	KM $k=5$	HC 3C	HC 4C	DBSCAN 3,3
1	1	4	6	1	4	1
2	1	5	5	1	1	1
3	1	3	3	5	5	2
4	4	4	4	1	1	1
5	3	3	1	3	3	1
6	4	5	5	1	1	1
7	1	1	3	5	5	4
8	1	1	6	1	1	1
9	1	5	5	1	1	1
10	1	3	3	2	3	3
11	1	1	4	5	1	5
12	4	4	4	3	1	1
13	6	1	1	1	1	1
14	4	4	6	1	1	3
15	1	4	4	1	1	6
16	3	3	3	1	2	6
17	6	1	1	1	1	5
18	5	3	1	4	5	1
19	5	1	1	5	1	4

we build the ranking presented in Table 7. We assign a higher position in the ranking to smaller distances. The smallest possible distance is $D = 0$. Smallest distances rank first. Ties here are represented assigning the same position to tied algorithms and freeing the next positions in equal number. Following this convention, we obtain Table 7.

Finally, we illustrate the use of PL methods to determine the probability of each algorithm being the best, as well as the uncertainty in our choice of algorithm.

Table 8 represents the results obtained combining the PL method with random tiebreaking and averaging. These results indicate that hierarchical clustering with 4 clusters is the algorithm performing best, with a probability of 21.30%. Next, it follows hierarchical clustering with 3 clusters, with a 20.36% probability. DBSCAN appears with probability

Table 8 Weights (probabilities) obtained for each method applying the Placket-Luce method to the ranking in Table 7, after random triebreakers, and averaging the results. We average 100 runs undoing ties randomly

KM $k=3$	KM $k=4$	KM $k=5$	HC 3C	HC 4C	DBSCAN MinPts = 3, $\varepsilon = 3$
0.148702	0.137208	0.11005	0.20365	0.213026	0.187354

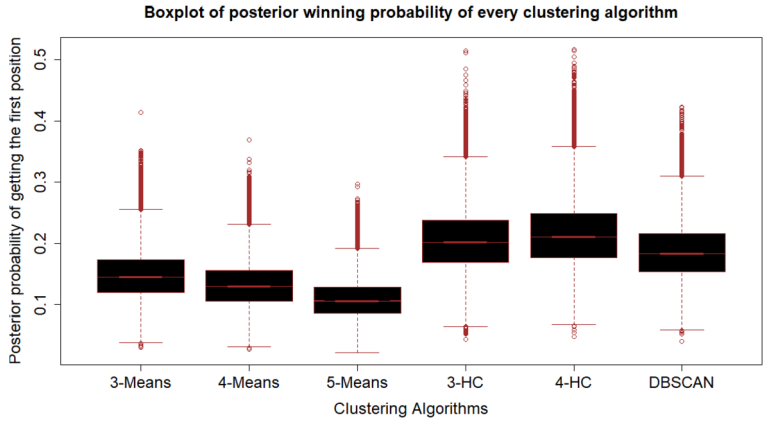

Figure 9 Boxplots illustrating uncertainty in the weight predictions given by Table 8 constructed with 10^6 samples

Table 9 Patterns selected by each distance for different trial sequences

Trial sequence	Reference pattern	d_1 selection	d_2 selection	d_3 selection
1	1	1	1	1
2	2	2	1	2
3	5	5	5	2
4	3	3	2	3
5	4	5	4	1

18.73%. Figure 9 quantifies uncertainty using boxplots constructed from samples from the 100 runs (1000 samples from each run).

A similar study can be performed allowing for ties by means of the Davidson-Luce model. Next, we illustrate this procedure on distance selection.

Appendix C: Distance selection

This appendix illustrates how to select the best distance to compare sequences formed by the digits −1, 0, 1. We consider 5 patterns containing 125 digits and up to 300 sequences obtained changing randomly a few digits in such patterns. Thus, the underlying reference pattern (the 'diagnosis') is known for each of them. Next, we compare each sequence (each 'patient') with the 5 reference patterns using the Hamming distance d_1, the Euclidean distance d_2 and the Earth Mover's distance d_3. Table 9 illustrates the outcome for $M = 5$ patients.

Table 10 Success score and calculation of intermediate success averages

Case	d_1	d_2	d_3
1	1	1	1
2	1	0	1
3	1	1	0
4	1	0	1
5	0	1	0
sum	4	3	3

Table 11 Success averages representing distance performance for $N = 10$ collections of $M = 5$ runs

Block	d_1	d_2	d_3
1	4	3	3
2	5	2	1
3	4	2	3
4	4	4	3
5	3	4	3
6	3	3	3
7	5	3	2
8	4	1	3
9	4	5	2
10	5	3	3

Table 12 Distance rankings

Block	d_1	d_2	d_3
1	1	2	2
2	1	2	3
3	1	3	2
4	1	1	3
5	2	1	2
6	1	1	1
7	1	2	3
8	1	3	2
9	2	1	3
10	1	2	2

Now, we create an intermediate $NM \times 3$ success score table R:

$$R_{ij} = \begin{cases} 1 & \text{when distance } j \text{ selects the correct reference pattern for patient } i, \\ 0 & \text{otherwise,} \end{cases}$$

and reduce dimensionality calculating N intermediate averages for consecutive blocks of M patients. For the first 5 patients, we find Table 10, where the last row is obtained adding up each column. The last row becomes the first row of a $N \times 3$ performance matrix. Repeating this process for $N = 10$ consecutive blocks of $M = 5$ patients we find the performance matrix collected in Table 11.

From the performance matrix, we construct the table of rankings, see Table 12. For each row, the distance that scores a higher number of correct assignations ranks first, the next one ranks second, and so on. Distances with the same number of correct guesses are tied.

Table 13 Probabilities for each distance

Hamming	Euclidean	EMD
0.7452619	0.3127958	0.10000000

This ranking[2] is introduced in the R package [23] to find the worths and tie prevalences involved in formulas (9) and (10). We have implemented this procedure for different choices of M and N. Setting $M = 20$ and $N = 15$, for instance, we have found the worths $\alpha_1 = 0.79739001$, $\alpha_2 = 0.13612835$ $\alpha_3 = 0.06648164$ and tie prevalences $\delta_2 = 0.57440580$, $\delta_3 = 0.79125103$. The probability of each distance being the best is the sum of the probabilities of that distance being the first alone, tied with another one or in a triple tie. We obtain for each of them the probabilities listed in Table 13.

Notice that the probabilities in Table 13 do not add to one because the probabilities of 2-ties and 3-ties are counted twice and three times. In any case, the Hamming distance clearly outperformed the rest for the synthetic dataset considered.

Alternatively, we can proceed as in the previous appendix, resorting to the PL method with random tiebreaking and averaging. The resulting probabilities add up to one, and the Hamming distance is still placed first.

Acknowledgements
The authors thank Hospital Universitario Puerta de Hierro (Madrid, Spain) for providing anonymized SLE clinical data.

Funding
This research has been partially supported by the FEDER /Ministerio de Ciencia, Innovación y Universidades - Agencia Estatal de Investigación grants No. MTM2017-84446-C2-1-R and PID2020-112796RB-C21.

Abbreviations
SLE, Systemic Lupus Erythematosus; TWD, dynamic time warping distance; EMD, Earth movers' distance; KM, K Means; HC, Hierarchical clustering; DBSCAN, Density-Based Spatial Clustering of Applications with Noise; PL, Plackett-Luce; DL, Davidson-Luce; MCMC, Markov Chain Monte Carlo; ECG, Electrocardiogram; TdP, Torsades de pointes.

Availability of data and materials
Anonymized data could be available upon request. The Plackett-Luce model with Dirichlet prior distribution and MCMC samplers are implemented in the library https://github.com/b0rxa/scmamp. The Plackett-Luce model with ties is implemented in https://cran.r-project.org/web/packages/PlackettLuce/index.html. Codes for Wasserstein distances are available at https://github.com/liujl11git/multilevelOT. Implementations of the TWD are available at https://github.com/talcs/simpledtw/blob/master/simpledtw.py for instance.

Declarations

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
AS and AT were responsible for data curation, visualization and software devolopment. AC and LFV conceptualized and supervised the work, provided resources and acquired funds. AC, AS and AT selected the methodology and wrote the paper, with edits from LFV. All authors carried out the investigation, formal analysis and validation of results. All authors read and approved the final manuscript.

Author details
[1]Departamento de Matemática Aplicada, Universidad Complutense de Madrid, Plaza de Ciencias 3, 28040, Madrid, Spain. [2]Gregorio Millan Barbarny Institute for Modelling and Simulation in Fluid dynamics, Nanoscience and Industrial Mathematics, Avenida de la Universidad 30, 28911, Leganés, Spain. [3]Hospital Universitario Puerta de Hierro de Madrid, Manuel de Falla s/n, 28220, Majadahonda, Spain.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

[2]Notice that we have written 1,1,3 and not 1,1,2. This choice does not alter the outcome.

Received: 8 November 2021 Accepted: 6 January 2022 Published online: 13 January 2022

References

1. Rajpurkar P, Irvin J, Zhu K, Yang B, Mehta H, Duan T, Ding D, Bagul A, Ball RL, Langlotz C, Shpanskaya K, Lungren MP, Ng AY. CheXNet: radiologist-level pneumonia detection on chest X-rays with deep learning. 2017. https://arxiv.org/abs/1711.05225.
2. Wu E, Hadjiiski LM, Samala RK, Chan HP, Cha KH, Richter C, Cohan RH, Caoili EM, Paramagul C, Alva A, Weizer AZ. Deep learning approach for assessment of bladder cancer treatment response. Tomography. 2019;5(1):201–8.
3. Vogt W, Nagel D. Cluster analysis in diagnosis. Clin Chem. 1992;38(2):182–98.
4. Benjamin JR. Making connections: using networks to stratify human tumors. Nat Methods. 2013;10(11):1077–8.
5. Soul J, Dunn SL, Anand S, Serracino-Inglott F, Schwartz JM, Boot-Handford RP, Hardingham TE. Stratification of knee osteoarthritis: two major patient subgroups identified by genome-wide expression analysis of articular cartilage. Ann Rheum Dis. 2017;0:1–8.
6. Pouryahya M, Oh JH, Mathews JC, Deasy JO, Tannenbaum AR. Characterizing cancer drug response and biological correlates: a geometric network approach. Sci Rep. 2018;8:6402.
7. Carlier A, Vasilevich A, Marechal M, de Boer J, Geris L. In silico clinical trials for pediatric orphan diseases. Sci Rep. 2018;8:2465.
8. Saeb S, Lonini L, Jayaraman A, Mohr DC, Kording KP. The need to approximate the use-case in clinical machine learning. GigaScience. 2017;6:1–9.
9. Valladares-Rodriguez S, Pérez-Rodriguez R, Fernandez-Iglesias JM, Anido-Rifón LE, Facal D, Rivas-Costa C. Learning to detect cognitive impairment through digital games and machine learning techniques. Methods Inf Med. 2018;57:197–207.
10. Waggener B. Pulse code modulation techniques. Berlin: Springer; 1995. p. 206. https://books.google.com/books?id=8l_o6kl3760C&pg=PA206.
11. Yingmin L, Huiguo C, Zheqian W. Dynamic time warping distance method for similarity test of multipoint ground motion field. Math Probl Eng. 2010;2010:749517.
12. Gold O, Sharir M. Dynamic time warping and geometric edit distance: breaking the quadratic barrier. J Assoc Comput Mach. 2018;14(4):50.
13. Yossi R, Carlo T, Leonidas JG. A metric for distributions with applications to image databases. In: Proceedings ICCV. 1998. p. 59–66. https://doi.org/10.1109/ICCV.1998.710701.
14. Jialin L, Wotao Y, Wuchen L, Yat TC. A fast approximation of Wasserstein-1 distances. SIAM J Sci Comput. 2021;43(1):A193–220.
15. Marden J. Analyzing and modeling rank data. London: Chapman & Hall; 1995.
16. Guiver J, Snelson E. Bayesian inference for Plackett-Luce ranking models. In: Proceedings of the 26th annual international conference on machine learning - ICML 09. 2009. p. 1–8.
17. Calvo B, Ceberio J, Lozano JA. Bayesian inference for algorithm ranking analysis. In: GECCO genetic and evolutionary computation conference companion, Kyoto, Japan. New York: ACM; 2018.
18. Kotz S, Balakrishnan N, Johnson NL. Continuous multivariate distributions. Volume 1: models and applications. New York: Wiley; 2000.
19. Gilks WR, Richardson S, Spiegelhalter D. Markov chain Monte Carlo in practice. Boca Raton: CRC Press; 1995.
20. Neal RM. MCMC using Hamiltonian dynamics. In: Brooks S, Gelman A, Jones GL, Meng XL, editors. Handbook of Markov chain Monte Carlo. London: Chapman & Hall; 2011.
21. Carpio A, Simón A, Villa LF. Clustering methods and Bayesian inference for the analysis of the time evolution of immune disorders. 2020. https://arxiv.org/abs/2009.11531.
22. Firth D, Kosmidis I, Turner HT. Davidson-Luce model for multi-item choice with ties. 2019. https://arxiv.org/abs/1909.07123.
23. Turner HL, van Etten J, Firth D, Kosmidis I. Modelling rankings in R: the PlackettLuce package. Comput Stat. 2020;35:1027–57.
24. Arnaud L, van Vollenhoven R. Advanced handbook of systemic lupus erythematosus. Berlin: Springer; 2018.
25. MacQueen J. Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, volume 1: statistics. Berkeley: University of California Press; 1967. p. 281–97.
26. Kaufman L, Rousseeuw PJ. Finding groups in data: an introduction to cluster analysis. Hoboken: Wiley-Interscience; 1990.
27. Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the second international conference on knowledge discovery and data mining. Portland: AAAI Press; 1996. p. 226–31.
28. Lilly LS, editor. Pathophysiology of heart disease: a collaborative project of medical students and faculty. 6th ed. Baltimore: Williams & Wilkins; 2016.
29. Klabunde RE. Electrical activity of the heart. Cardiovascular physiology concepts. Baltimore: Williams & Wilkins; 2005.
30. https://data.mendeley.com/datasets/gwbz3fsgp8/2.
31. Aringer M, Costenbader K, Daikh D, Ralph Brinks R, Mosca M, Ramsey-Goldman R et al. European league against rheumatism/American college of rheumatology classification criteria for systemic lupus erythematosus. Ann Rheum Dis. 2019;78:1151–9.
32. Hopkins B, Skellam JG. A new method for determining the type of distribution of plant individuals. Ann Bot. 1954;18(2):213–27.