Supplementary material

Smoothed distance

If $\{d_i\}$ is the sequence of the distance function, where the index i takes values from 1 to the maximum of data N, then the values for the smoothed distance function sequence $\{\bar{d}_i\}$, are determined as follows for a radius of K frames around each value. For $K < i < N - K$,

$$\bar{d}_i = \frac{1}{2K+1} \sum_{j=i-K}^{i+K} d_j.$$

The first K terms of $\{\bar{d}_i\}$, for $1 \leq i \leq K$, are determined by

$$\bar{d}_i = \frac{1}{K+i} \sum_{j=i}^{i+K} d_j.$$

While the last K terms of $\{\bar{d}_i\}$, for $N - K \leq i \leq N$, are defined by the formula

$$\bar{d}_i = \frac{1}{K+N-i} \sum_{j=i-K}^{N} d_j.$$

Recurrence

The recurrence plot is defined by symmetric matrix $A = [a_{ij}]$ with dimensions $N \times N$, in which the inputs are determined by the following function:

$$a_{ij} = \begin{cases} \text{black}, & \text{if } P_j \in R_k \text{ and } P_i \in R_k, \\ \text{white}, & \text{if } P_j \notin R_k \text{ and } P_i \in R_k, \end{cases}$$

where $i = 1,2,\dots,N$; $j = 1,2,\dots,N$; $k = 1,2,\dots,100$. That is, the matrix A (recurrence plot) is a matrix of time (frame=.2 sec) per time (frame=.2 sec), from i to j. P_j is a given frame and R_k is a given region (one of hundred regions). If given frames P_i & P_j coincide in R_k, then the value a_{ij} =black in the intersection between P_i & P_j in the matrix A. If given frames P_i & P_j do not coincide in R_k, then the value a_{ij} =white in the intersection between P_i & P_j in the matrix A.

Entropy

The concept of entropy was introduced by Shannon (1948). Entropy is a measure associated with a discrete random variable, which indicates variability within a distribution. Thus, Shannon entropy is a continuous, monotonic, and linear indicator of how different the distribution elements are from each other (Carcassi, Aidala & Barbour, 2021). In our work a statistical distribution.

Formally, given a discrete random variable X with possible outcomes $\{x_i\}$, each with probability $P(x_i)$, the entropy $H(X,P)$ is as follows

$$H(X,P) = -\sum P(x_i)\, ln\big(P(x_i)\big).$$

It can be proven that the entropy of a discrete random variable is a non-negative number, $H(X,P) \geq 0$, and its measure should be maximal if all the outcomes are equally likely (uncertainty is highest when all possible events are equiprobable).

To analyze the displacement pattern of individuals in each session, the discrete random variables $\{x_i\}$ are the permanence in each defined zone from a configuration of 10×10 defined zones and $P(x_i)$ is accumulated time (standardized) at it.

Divergence

The Kullback-Leibler divergence was introduced by Kullback & Leibler (1951) and discussed by Kullback (1959). The Kullback-Leibler divergence (or relative entropy) is a measure of difference from first one probability distribution to second one.

For discrete probability distributions P and Q defined on the same random variable X, with possible outcomes $\{x_i\}$, the Kullback–Leibler divergence from Q to P is defined as

$$D_{KL}(P \parallel Q) = \sum_i P(x_i) ln\left(\frac{P(x_i)}{Q(x_i)}\right).$$

The Kullback-Leibler divergence is defined only if for all x, $Q(x) = 0$ implies $P(x) = 0$. Whenever $P(x)$ is zero, the contribution of the corresponding term is interpreted as zero.

The KL divergence $D_{KL}(P \parallel Q)$ can be thought of as something like a measurement of how far the distribution Q is from the distribution P, because it is always non-negative ($D_{KL}(P \parallel Q) \geq 0$) and a result known is $D_{KL}(P \parallel Q)$ zero if and only if $P = Q$, this is a Kullback–Leibler divergence of 0 indicates that the two distributions in question are identical. However, it is not symmetric; that is, $D_{KL}(P \parallel Q) \neq D_{KL}(Q \parallel P)$.

To analyze the displacement pattern of individuals in consecutive sessions, the discrete random variables $\{x_i\}$ are the permanence in each square region from a configuration of 10×10 defined zones and $Q(x_i)$ is the accumulated time (standardized) at the first session and $P(x_i)$ is the accumulated time (standardized) at the second one.

Variable ranking

The variable ranking is a lenient version of feature selection, which consists of ordering a set of features (input variables) normally by the value of a scoring function that measures the relevance of each feature according to a target (or output variable) as a predicting tool. Feature selection could be too rigid for some applications where determining the variables

that must be removed or preserved is not clear, given that all features are required to explain or predict a target value. Still, the relevance of some features is higher for the investigation. Thus, even in some cases, variable ranking is the first phase for variable selection.

The algorithms for variable ranking are divided into two main classes: wrappers and filters. On the one hand, we can find several wrappers algorithms; most of them rely on training a machine learning algorithm like SVM or Decision Tree to estimate the quality of a subset of features of the dataset to predict the target variable. On the other hand, this approach could consume a large amount of computing resources for high-dimensional datasets when looking for a minimal optimal feature set.

On the other hand, filter algorithms are the most widely used due to the low computing resources used to apply them, even on high-dimensional datasets. There are versions for single or multiple variable classifications, and their implementation also depends on whether the target variable is numerical or categorical. Since, in our experiments, the data sets have only categorical targets, we opted for this approach. Three theoretical information filter algorithms, conducted using Orange© 3.26, were used for the classification of a single variable ranking: information gain, mean decrease Gini, and $\boldsymbol{\chi^2}$.

We will denote by T the set of training samples in the form of tuples $(x, y) = (x_1, x_2, x_3, \ldots, x_n, y)$, where x_a is the value of the a^{th} feature (e.g., mean distance to the relevant object, mean velocity, entropy, and so on) of the example x and y is the corresponding target variable to a class label (e.g., water deprivation; food deprivation).

t-SNE

The t-distributed Stochastic Neighbor Embedding is a Machine Learning algorithm commonly used to visualize high-dimensional datasets into a 2D o 3D space. In broad terms, t-SNE performs a non-linear dimensionality reduction task for embedding datasets and obtaining low dimension transformations. This reduction of dimensionality is suitable for visualization where data entries with similar values for the features taken as inputs would be closer to each other than entries with dissimilar values. The relations between inputs that might be impossible to observe due to a large number of variables would be distinguished after transforming them into a space with reduced dimension. The t-SNE analysis was conducted using Orange© 3.26

The t-SNE algorithm performs two phases to transform input data. First, it builds a probability distribution with each pair of high-dimension data entries looking for similar data to obtain a higher probability and place them closer together, and a lower probability for data with values that have a significant difference between them. The second phase involves defining a probability distribution over a reduced dimension space (2D or 3D) and minimizing the Kullbak-Leibler Divergence between them through the gradient descent algorithm, normally training a neural network. By reducing the differences between probability distributions, a high dimension space is transformed to a low dimension space, preserving the relationships among data entries given the values of its features.

For calculating the t-SNE, given a dataset X with a length N, for each pair of data entries xi and xj, the probability distribution in the first stage is calculated by

$$P(i,j) = \frac{P(i|j) + P(j|i)}{2N},$$

where the conditional probability $P(i|j)$ is computed by using

$$P(i|j) = \begin{cases} \dfrac{\exp\left(-\dfrac{\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq i} \exp\left(-\dfrac{\|x_i - x_k\|^2}{2\sigma_i^2}\right)}, & if\ i \neq j, \\ 0, & if\ i = j. \end{cases}$$

In the last expression, $\|x_i - x_j\|$ denotes the Euclidian distance between data entries, but another distance metric could be used. For the second phase, points Y = (y_0, y_1, y_2, …, y_N) are found in such a way that minimizes the Kullback-Leibler Divergence between $P(i|j)$ and $Q(i|j)$, where if $i \neq j$ then

$$Q(i|j) = \frac{\dfrac{1}{1 + \|y_i - y_j\|^2}}{\sum_k \sum_{k \neq l} \exp(\dfrac{1}{1 + \|y_k - y_l\|^2})}$$

and it is equal to 0 in another case.

Linear Projection for Visualization of multidimensional datasets

Data Visualization is a powerful tool for data analysis in several stages of data processing. It plays a critical role in exploratory data analysis and data mining tasks. However, the visualization of datasets with more than two or three dimensions (variables) represents a challenge using conventional plotting 2D or 3D. A series of methods have been developed to solve this problem, in which linear projections are made to reduce multidimensional data presented in two or three dimensions. Several methods exist for performing those linear projections; examples are Linear Discriminant Analysis, Principal Components Analysis, Orthogonal Projection, Linear Regression, among others.

In general, a linear projection could be described as a linear transformation P from a vector space to the same vector space that satisfies $P2=P$. Thus, several projections could be performed for a given dataset, resulting in various graphs expressing different facets of the dataset. When a class separation is a purpose for the projection, choosing the appropriate parameters and method is crucial, as could be appreciated in the image below.

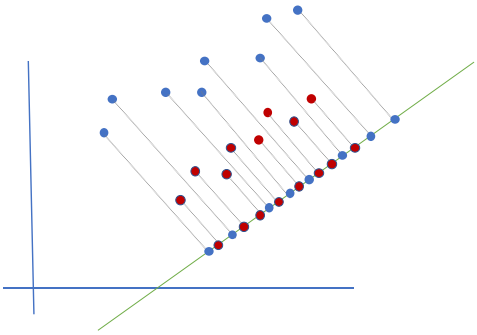

Fig. A

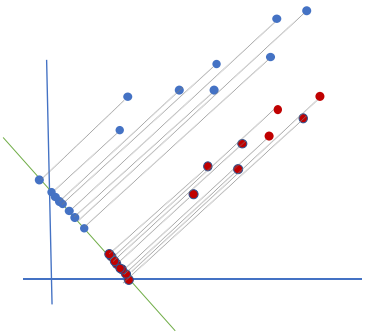

Fig. B

In the figure, we observe the same class-labeled multidimensional dataset being projected over two different linear models. In Fig. A, the resulting projection combines the entries from the two classes and does not help to separate them. While in figure B, the same dataset is correctly separated by classes. Two linear models have been created for the projection, being different in the parameters used to construct them, showing the benefits of an adequate linear projection for analyzing multidimensional datasets. In this work, the Linear Projection was conducted using Orange© 3.26

K-means

K-means is an algorithm that aims to generate clusters of an observed dataset $\{x_i\}$ of n observations into k groups. Each group is represented by the average of the points that compose it. Thus, the centroid is the representative of each group. The number of groups to discover, k, is a parameter that must be configured a priori. The clustering method starts with k randomly located centroids and assigns each observation to the closest centroid. The centroids, once assigned, are moved to the average location of all assigned data; then, the points are again assigned according to the location of the new centroids.

The objective of K-means is to group the observations in such a way that all those that are in the same group are the most similar to each other and that those that belong to different groups are the most dissimilar to each other. Distance measures, such as Euclidean, are used to measure similarities and differences. A measure of how well the centroids $M^t = (m_1^t, m_2^t, \dots, m_k^t)$, at iteration t, represent the members of your group is the sum of the squared errors. In each iteration t, K-means tries to reduce the value of the sum of the errors squared. The measure consists of the sum of the squared distances of each observation $x_i$from the centroid of its group

$$M^t = argmin \sum_{j=1}^{k} \sum_{i=1}^{n} || x_i - m_j^{t-1} ||.$$

The algorithm is iterative and always concludes, either by a fixed number of iterations or when it converges to a solution (total error due to assignations remains the same as in the previous iteration). Since it does not necessarily find the most optimal

configuration, the one corresponding to the minimum of the objective function is shown. Finding a minimum of the function, even if it is not the absolute minimum, guarantees a grouping in which the groups are sparsely dispersed and separated from each other. The algorithm is significantly sensitive to centroids that are initially randomly selected. This effect can be reduced by performing multiple runs of the method. The clustering by K-means was conducted using Orange© 3.26

References

Carcassi, G., Aidala, C. A., & Barbour, J. (2021). Variability as a better characterization of Shannon entropy. *European Journal of Physics*, 42(4), 045102. http://doi.org/10.1088/1361-6404/abe361

Kullback, S., and Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1), 79–86. https://doi.org/10.1214/aoms/1177729694

Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(4), 623–656. https://doi.org/10.1002/j.1538-7305.1948.tb00917.x

Universidad de Ljubljana. (1996). *Orange: Data Mining Fruitful & Fun*. (version 3.26) [software]. Bioinformatics Lab.