Clustering using a random walk based distance measure

Luh Yen[1], Denis Vanvyve, Fabien Wouters, François Fouss[1],
Michel Verleysen * [2] & Marco Saerens[1]

1- Université catholique de Louvain, ISYS, IAG
Place des Doyens 1, B-1348 Louvain-la-Neuve, Belgium
{yen, fouss, saerens}@isys.ucl.ac.be

2- Université catholique de Louvain, DICE, FSA
Place de Levant 3, B-1348 Louvain-la-Neuve, Belgium
verleysen@dice.ucl.ac.be

Abstract. This work proposes a simple way to improve a clustering algorithm. The idea is to exploit a new distance metric called the "Euclidian Commute Time" (ECT) distance, based on a random walk model on a graph derived from the data. Using this distance measure instead of the usual Euclidean distance in a k-means algorithm allows to retrieve well-separated clusters of arbitrary shape, without working hypothesis about their data distribution. Experimental results show that the use of this new distance measure significantly improves the quality of the clustering on the tested data sets.

1 Introduction

In clustering, the data distribution has an important impact on the classification results. However, in most clustering problems, there is few prior information available about the underlying statistical model, and the decision maker must make some arbitrary assumptions. For instance, the k-means algorithm, in its basic form, can fail on data sets containing clusters of arbitrary or even non-convex shape, even if they are well-separated.

In this work, we propose the use of a new distance measure, the **Euclidean Commute Time distance** (ECT distance, see reference [11] and [12]), in order to improve the clustering performance. The ECT distance is based on a random walk model on a graph derived from the data. More precisely, the ECT distance is a distance measure between the nodes of a weighted graph and presents the interesting property of decreasing when the number of paths connecting two nodes increases or when the "length" of any path decreases, which makes it well-suited for clustering tasks.

At first sight, the proposed method seems similar to the classical "shortest path" distance on a graph (also called Dijkstra or geodesic distance [2]). Actually our distance metric differs about the fact that it takes the connectivity between nodes into account: Two nodes are "close" according to this distance if they are highly connected. Notice that the idea of exploiting random walks concept

*Michel Verleysen is a Senior Research Associate of the F.N.R.S.

for clustering has already been proposed by Koren and Harel [7], by using the notion of escape probabilities to find separating edges of a graph. The difference between the two works is that our method is based on a distance measure and has a nice geometric interpretation in terms of a Mahalanobis distance (see Equation 2).

The paper is organized as follows. An introduction to the ECT distance is provided in Section 2. Section 3 shows how the ECT distance can be computed from the Laplacian matrix of the graph derived from the data. Section 4 presents the clustering algorithm based on ECT distance. Section 5 provides experimental results on an artificial data set and on a digital characters clustering problem.

2 Distance measure based on a random walk model

The essential of the theory justifying the defined distance is developed in papers [11] and [12]. Only a short overview is provided here.

2.1 A random walk model on a weighted graph

In a first step, the data (N observations in total) are linked to form a connected graph in the following way: Each observation is represented by a node of the graph and is connected to his k nearest neighbors, according to the Euclidean distance. In addition, the minimum spanning tree [3] (minimizing the sum of the Euclidian distances) is computed and its edges are added to the graph in order to obtain a connected graph : each node can be reached from any other node of the graph through at least one path. Following the definition of this graph, we expect that two points in the same cohesive cluster are connected by a large number of short paths.

The weight $w_{ij} \geq 0$ of the edge connecting node i and node j is set to some meaningful value, representing the closeness of observations i and j. It is chosen here to be inversely proportional to the Euclidean distance between the two observations.

Based on the constructed graph it is possible to compute the associated adjacency matrix $\mathbf{A}$ in the standard way, with elements $a_{ij} = w_{ij}$ if node i is connected to node j, and 0 otherwise.

Then we associate the state of a Markov chain to every node of the graph (N in total). To any state or node i, we associate a probability of jumping to an adjacent node (a nearest neighbor) : $p_{ij} = \frac{a_{ij}}{a_{i.}}$, with $a_{i.} = \sum_{j=1}^{N} a_{ij}$.

2.2 The average commute time

Based on this Markov chain, two important quantities are defined : the average first-passage time and the average commute time.

The **average first-passage time** $m(k|i)$ is defined as the average number of steps a random walker, starting in state $i \neq k$, will take to enter state k for the first time. Formally, $m(k|i)$ is defined as (see for instance [10]) :

$$\begin{cases} m(k|k) = 0 \\ m(k|i) = 1 + \sum\limits_{\substack{j=1 \\ j \neq k}}^{N} p_{ij}\, m(k|j), \text{ for } i \neq k. \end{cases} \tag{1}$$

These equations can be used in order to iteratively compute the first-passage times.

The second quantity is the **average commute time**, $n(i,j)$, which is defined as the average number of steps a random walker, starting in state $i \neq j$, will take before entering a given state j for the first time, and go back to i. That is, $n(i,j) = m(j|i) + m(i|j)$. It was shown by several authors [6], [8] that the average commute time is a **distance measure** between any nodes of the graph.

3 Computation of the basic quantities by means of $\mathbf{L}^{+}$

The Laplacian matrix of the graph is defined by $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{A}$ is the adjacency matrix of the graph and $\mathbf{D} = \text{diag}(a_{i.})$ (with $a_{i.} = \sum_{j=1}^{N} a_{ij}$) is the degree matrix. It is shown in [11] that the computation of the average commute time can be obtained from the Moore-Penrose pseudoinverse [1] of $\mathbf{L}$, denoted by $\mathbf{L}^{+}$:

$$n(i,j) = V_G\, (\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}} \mathbf{L}^{+} (\mathbf{e}_i - \mathbf{e}_j), \tag{2}$$

where $\mathbf{e}_i = [\underset{1}{0}, \ldots, \underset{i-1}{0}, \underset{i}{1}, \underset{i+1}{0}, \ldots, \underset{N}{0}]^{\mathrm{T}}$ is a basis vector and where $V_G = \sum_{i,j} a_{ij}$ is the volume of the graph.

We easily observe from Equation 2 that $[n(i,j)]^{1/2}$ is a distance, since it can be shown [11] that $\mathbf{L}^{+}$ is symmetric and positive semidefinite. It is therefore called the **Euclidean Commute Time** (ECT) **distance**.

If the matrices are too large, the computation by pseudoinverse becomes cumbersome; in this case, it is still possible to compute the ECT distance iteratively using Equation 1.

4 K-means based on ECT distances

Of course, any clustering algorithm (hierarchical clustering, k-means, etc) could be used in conjunction with the ECT distance. In this work, we illustrate its potential usefulness by using a k-means algorithm. To this end, we implemented a k-means method working directly on the distance matrix (see for instance [14]).

Let us denote as $\{\mathbf{x}_k\}$, $k = 1, ..., N$, the set of observations to be clustered into c different clusters. We define the ECT distance matrix, $\mathbf{\Delta}$, where element $[\mathbf{\Delta}]_{ij} = \delta(\mathbf{x}_i, \mathbf{x}_j) = n(i,j)$ is the squared ECT distance between observations $\mathbf{x}_i$ and $\mathbf{x}_j$.

Each cluster C_l, $l = 1, ..., c$, is represented by one prototype, $\mathbf{p}_l$, which is chosen among the observations (it is therefore not the centroid, as it is usually the case with the k-means algorithm). The distance between an observation $\mathbf{x}_k$ and a cluster C_l is defined as the distance to the prototype : $\text{dist}[\mathbf{x}_k, C_l] = \delta(\mathbf{x}_k, \mathbf{p}_l)$

The within-cluster variance for cluster C_l is defined by

$$J_l = \sum_{\mathbf{x}_k \in C_l} \text{dist}^2[\mathbf{x}_k, C_l]. \tag{3}$$

The optimization criterion J is simply the sum of the within-cluster variances J_l of each cluster C_l :

$$J = \sum_{l=1}^{c} J_l = \sum_{l=1}^{c} \sum_{\mathbf{x}_k \in C_l} \text{dist}^2[\mathbf{x}_k, C_l]. \tag{4}$$

Criterion J depends on two elements: the allocation of the observations to a cluster and the position of the prototypes. It is quite difficult in terms of computing time to find the best, global, minimum of J. Most of the algorithms only compute a local minimum of J; this is the case for our ECT distance k-means algorithm, which iterates the two basics steps:

(1) Allocation of the observations. The prototypes are fixed. Each observation $\mathbf{x}_k$ is allocated to its nearest cluster; that is, $\mathbf{x}_k$ is assigned to cluster C_l such that

$$l = \arg\min_j \text{dist}^2[\mathbf{x}_k, C_j] = \arg\min_j \delta^2(\mathbf{x}_k, \mathbf{p}_j); \tag{5}$$

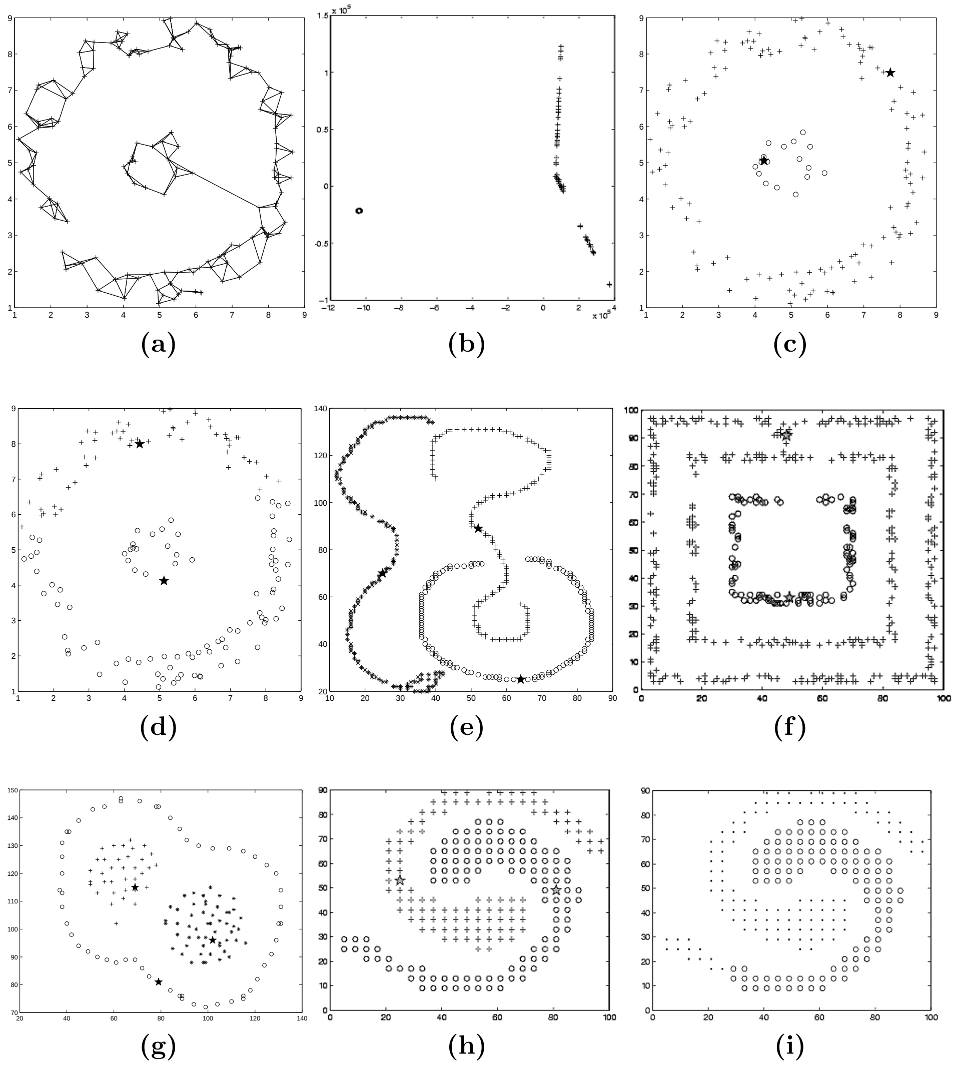
(2) Computation of the prototypes. We now consider that the allocation of the observations is fixed (each $\mathbf{x}_k$ is assigned to a cluster). For each cluster C_l, we choose a new prototype, $\mathbf{p}_l$, among the observations so that it minimize the within-cluster variance (3) of this cluster. More precisely, the prototype of each cluster C_l is chosen according to:

$$\mathbf{p}_l = \arg\min_{\mathbf{x}_j} \left\{ \sum_{\mathbf{x}_k \in C_l} \delta^2(\mathbf{x}_k, \mathbf{x}_j) \right\}. \tag{6}$$

The clustering algorithm aims to repeat steps **(1)** and **(2)** until convergence of J to a local minimum. It can be shown that J decreases at each such step [14]. This clustering procedure based on the ECT distance will be called the ECT distance k-means.

5 Experiments

In order to evaluate the ECT distance k-means algorithm, it is applied to two clustering problems, and compared to the classical k-means based on the Euclidean distance. Five artificial data sets (inspired by [9]) are used to illustrate the ability to detect clusters with arbitrary shapes. We also compare our method to the normalized cuts [13], since we established in [12] several similarities between the normalized cuts and the ECT distance. The second experiment aims to cluster digital characters.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 1: **Clustering using ECT distance k-means.** (a) Rings data set and its associated connected graph. (b) The multidimensional scaling projection of the ECT distance matrix on the two first principal axis. (c) Clustering results using the ECT distance k-means. Clusters are indicated by different symbols and prototypes by stars. (d) Clustering results using the Euclidean distance k-means. (e) – (h) Other clustering examples using ECT distance k-means on artificial data sets. (i) Clustering results using Shi and Malik's algorithm.

5.1 Experiments on artificial data sets

Figure 1a shows an example of graph construction. We made the arbitrary choice for every experiments of this paper to link each observation (node) of the data set to its three nearest neighbors, in addition to the links provided by the computation of the minimum spanning tree. Actually we observed that three neighbors are enough to get satisfactory results, in addition to reduce the computation complexity.

For illustration, the multidimensional scaling projection of the ECT distance matrix on the two first principal axis is shown in Figure 1b. We observe that the two clusters are well separated with the ECT distance metric.

The resulting partition obtained by using the ECT distance and the Euclidean distance are shown respectively in Figure 1c and Figure 1d. Both clustering algorithms are run twenty times with two prototypes (two clusters) and various random seeds; only the clustering with the minimal total within-class variance J is retained.

The same experiment is realized with four other artificial data sets (Figures 1e, 1f, 1g and 1h). Figure 1i shows an example of the clustering result obtained by using Shi and Malik's spectral clustering algorithm [13].

5.2 Digital characters clustering

The second experiment concerns a digital character clustering problem where the word "DENIS" is digitalized; the objective here is to retrieve the letters from the two-dimensional image.

Three data sets are constructed from the digitalized "DENIS", with various letter interspaces (see Figure 2a). An example of clustering on medium interspace set, obtained by ECT distance k-means, is shown in Figure 2b.

For each of the three data sets the ECT distance k-means and the classical k-means are respectively repeated twenty times. For each of the twenty clusterings, the quality of the obtained partition is assessed by comparing it to the optimal partition where each letter is a cluster (in this case, there are five clusters: the five letters of "DENIS"). Therefore, the *adjusted rand index* is computed, measuring the quality of the clustering (see for instance [5]). Then the adjusted rand indexes obtained by the twenty clusterings are averaged, in order to obtain the *averaged adjusted rand index*.

Figure 2c shows the values of the averaged adjusted rand index for the three "DENIS" data sets and the two k-means procedures, based on ECT and Euclidean distances. The first data set (label 1 in Figure 2a) contains small letter interspaces; the second data set (label 2) contains medium letter interspaces, and the third data set (label 3) contains large letter interspaces.

5.3 Discussion of the results

We observe that the algorithm based on the ECT distances provides good clustering results, both for the artificial data and the character clustering problems.

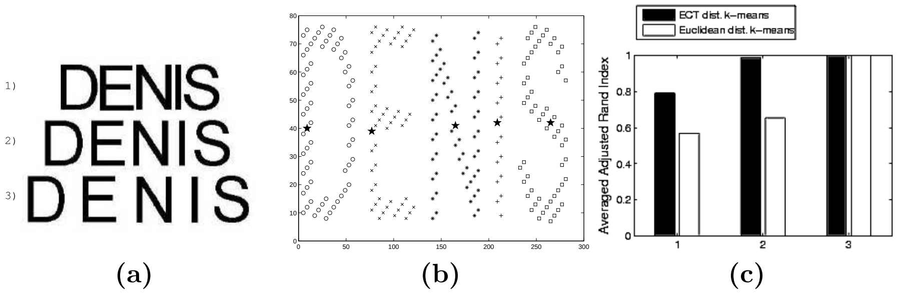


Fig. 2: **Digital characters clustering.** (a) Three "DENIS" sets with various interspace between letters. (b) Clustering results using the ECT distance k-means for medium interspace. (c) Comparisons of the averaged adjusted rand index for the three "DENIS" sets and the two clustering methods.

The classical k-means usually fails to cluster properly when the separation border between clusters is not trivial. On the contrary, the ECT distance k-means algorithm overcomes the difficulty and manages to separate the different clusters for the non-linearly separable, but nevertheless well separated, data sets. The visualization of the ECT distance matrix projected in a two-dimensional space by multidimensional scaling (Figure 1b) shows a interesting characteristic of the ECT distance metric : observations with strong internal cohesion move closer to their nearest neighbors. On the contrary, observations with few connections between them tend to be drawn aside.

But what happens if the subgroups are really close ? In this case, many connections can be built between close observations of different groups and can alter the performances. Indeed, as expected, the clustering performances decrease in the second experiment when the interspaces between letters get smaller (Figure 2c). Actually, this experiment illustrates one advantage of using the ECT distance compared to Euclidean distance: two points, which are close in the Euclidian space, can nevertheless have a large ECT distance if there are few paths connecting them. On the other hand, two points that are distant in the Euclidean space can nevertheless be close in terms of ECT distance if there are many paths connecting them.

Notice that the application of the normalized cuts proposed by Shi and Malik on our data sets gives slightly worse results when clusters are close (e.g., Figure 1i).

6 Conclusions and further work

We introduced a new distance measure, called the Euclidean commute time distance, which allows to retrieve well-separated clusters of arbitrary shapes. Experiments show that the ECT distance k-means is less sensitive to the shape of the cluster than the standard k-means based on the Euclidean distance. It is also interesting to notice that the ECT distance k-means is easy to use since there is no need to make assumption on the data distribution nor to fix some parameter values.

The main drawback of this method is that it does not scale well for large data sets. The distance matrix size is determined by the number of data and its estimation can be time consuming. However, the Laplacian matrix is usually sparse: only the information about links between nearest neighbors is kept.

Further work will extend the application of the ECT distance k-means to more sophisticated clustering problems. We will also continue our comparisons and investigations of the links between ECT distance k-means and spectral clustering (see [12]).

References

[1] S. Barnett. *Matrices: Methods and Applications.* Oxford University Press, 1992.

[2] F. Buckley and F. Harary. *Distance in graphs.* Addison-Wesley Publishing Company, 1990.

[3] T. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, 2nd ed.* Carnegie Mellon University, September 2001.

[4] P. G. Doyle and J. L. Snell. *Random Walks and Electric Networks.* The Mathematical Association of America, 1984.

[5] B. S. Everitt, S. Landau, and M. Leese. *Cluster Analysis.* Edward Arnold, London, 2001.

[6] F. Gobel and A. Jagers. Random walks on graphs. *Stochastic Processes and their Applications*, 2:311–336, 1974.

[7] D. Harel and Y. Koren. On clustering using random walks. *Lecture Notes in Computer Science*, 2245:18–41, 2001.

[8] D. J. Klein and M. Randic. Resistance distance. *Journal of Mathematical Chemistry*, 12:81–95, 1993.

[9] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems (NIPS)*, volume 14, pages 849–856, Vancouver, Canada, 2001. MIT Press.

[10] J. Norris. *Markov Chains.* Cambridge University Press, 1997.

[11] M. Saerens and F. Fouss. Computing similarities between nodes of a graph: Application to collaborative filtering. *Submitted for publication*, 2004. Available from http://www.isys.ucl.ac.be/staff/marco/Publications/.

[12] M. Saerens, F. Fouss, L. Yen, and P. Dupont. The principal components analysis of a graph, and its relationships to spectral clustering. *Proceeding of the 15th European Conference on Machine Learning (ECML)*, pages 371–383, 2004. Lecture Notes in Artificial Intelligence, Vol. 3201, Springer-Verlag, Berlin, 2004, pp 371-383.

[13] J. Shi and J. Malik. Normalised cuts and image segmentation. *IEEE Transactions on Pattern Matching and Machine Intelligence*, 22:888–905, August 2000.

[14] H. Spath. *Cluster analysis algorithms for data reduction and classification of objects.* Ellis Horwood, 1980.