DIMACS Technical Report 2006-09
May 2006

Probabilistic Distance Clustering

by

Adi Ben-Israel
RUTCOR–Rutgers Center for Operations Research
Rutgers University
640 Bartholomew Rd.
Piscataway, NJ 08854-8003, USA
adi.benisrael@gmail.com

Cem Iyigun
RUTCOR–Rutgers Center for Operations Research
Rutgers University
640 Bartholomew Rd.
Piscataway, NJ 08854-8003, USA
iyigun@rutcor.rutgers.edu

ABSTRACT

We present a new iterative method for probabilistic clustering of data. Given clusters, their **centers**, and the **distances** of data points from these centers, the **probability** of cluster membership at any point is assumed to depend on its distances from all centers. We state this assumption as a **principle** that probability is inversely proportional to distance.

The method is based on the above principle, and on the **joint distance function**, a measure of distance from all cluster centers, that evolves during the iterations, and captures the data in its low contours.

At each iteration, the distances (Euclidean, Mahalanobis, etc.) from the cluster centers are computed for all data points, and the centers are updated as stationary points of the joint distance function. The initial centers are arbitrary and computations stop when the centers stop moving.

The method is simple, fast (requiring a small number of cheap iterations) and gives a high percentage of correct classifications. It converges to the true cluster centers for all initial solutions, and is not sensitive to outliers.

Keywords: Distance clustering, probabilistic clustering, Euclidean distance, Mahalanobis distance

1 Introduction

A **cluster** is a set of data points that are similar, in some sense, and **clustering** is a process of partitioning a data set into disjoint clusters.

Clustering is a basic tool in statistics and machine learning, and has been applied in pattern recognition, medical diagnostics, data mining, biology, finance and other areas.

We take data points to be vectors $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$, and interpret "similar" as close in terms of a **distance** function $d(\mathbf{x}, \mathbf{y})$ in $\mathbb{R}^n$, such as

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|, \ \forall \, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \tag{1}$$

where the norm $\| \cdot \|$ is **elliptic**, defined for $\mathbf{u} = (u_i)$ by

$$\|\mathbf{u}\| = \langle \mathbf{u}, Q\mathbf{u} \rangle^{1/2}, \tag{2}$$

with $\langle \cdot, \cdot \rangle$ the standard inner product, and Q a positive definite matrix. In particular, we get the **Euclidean norm** for $Q = I$,

$$\|\mathbf{u}\| = \langle \mathbf{u}, \mathbf{u} \rangle^{1/2}, \tag{3}$$

and the **Mahalanobis distance** for $Q = \Sigma^{-1}$, where Σ is the covariance matrix of the data involved.

Example 1. A data set in $\mathbb{R}^2$ with $N = 200$ data points is shown in Figure 1. The data was simulated, from a normal distribution $N(\boldsymbol{\mu}, \Sigma)$, with:

$$\begin{aligned} \boldsymbol{\mu}_1 &= (0, 0), \ \Sigma_1 = \begin{pmatrix} 0.1 & 0 \\ 0 & 1 \end{pmatrix}, \ (100 \text{ points}) \ , \\ \boldsymbol{\mu}_2 &= (3, 0), \ \Sigma_2 = \begin{pmatrix} 1 & 0 \\ 0 & 0.1 \end{pmatrix}, \ (100 \text{ points}) \ . \end{aligned}$$

This data will serve as illustration in Examples 2–5 below.

Let the **dataset** $\mathcal{D}$ consist of N data points $\{\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_N\} \subset \mathbb{R}^n$, and let $\mathcal{D}$ be partitioned into **clusters** $\mathcal{C}_1, \ldots, \mathcal{C}_K$. With each cluster $\mathcal{C}_i$ we associate a **center** $\mathbf{c}_i$, such as its centroid.

Clusters are generated using a **clustering criterion**. **Distance clustering**, abbreviated **d–clustering**, is clustering with a metric criterion.

In d–clustering each data point is typically assigned to the cluster with the nearest center. After each such assignment, the cluster centers may change, resulting in further re-assignments. Such an algorithm will therefore iterate between centers and re–assignments.

A commonly used clustering criterion is the sum–of–squares of Euclidean distances from the cluster centers,

$$f(\mathcal{C}_1, \ldots, \mathcal{C}_K) := \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in \mathcal{C}_k} \| \, \mathbf{x}_i - \mathbf{c}_k \, \|^2, \tag{4}$$

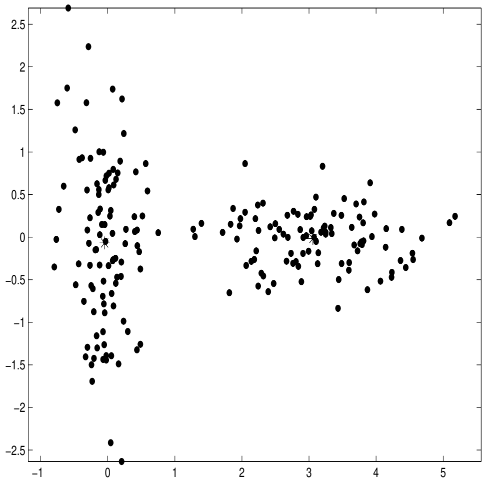

Figure 1: A data set in $\mathbb{R}^2$

to be minimized by the sought clusters $\mathcal{C}_1, \ldots, \mathcal{C}_K$. The well known k–**means clustering algorithm** [1] uses this criterion.

In **probabilistic clustering** the cluster membership is probabilistic, i.e. a data point $\mathbf{x}$ belongs to a cluster $\mathcal{C}_k$ with a **probability**, denoted $p_k(\mathbf{x})$.

To be specific, let a data set $\mathcal{D} \subset \mathbb{R}^n$ be partitioned into K clusters,

$$\mathcal{D} = \bigcup_{k=1}^{K} \mathcal{C}_k \tag{5}$$

and let $\mathbf{c}_k$ be the **center** of the k^{th} cluster. With each data point $\mathbf{x} \in \mathcal{D}$ we associate:
- a **distance** $d(\mathbf{x}, \mathbf{c}_k)$, also denoted $d_k(\mathbf{x})$, or just d_k if $\mathbf{x}$ is understood, and
- a **probability** of membership in $\mathcal{C}_k$, $p_k(\mathbf{x})$, abbreviated p_k.

Assuming different relations between the probabilities $p_k(\mathbf{x})$ and distances $d_k(\mathbf{x})$ results in different ways of clustering the data. We use here two principles,

$$p_k(\mathbf{x})\, d_k(\mathbf{x}) = \text{constant, depending on } \mathbf{x}\ , \tag{6}$$

giving the **probabilistic d–clustering** of § 2, and

$$p_k(\mathbf{x}) e^{d_k(\mathbf{x})} = \text{constant, depending on } \mathbf{x}\ , \tag{31}$$

for **probabilistic exponential d–clustering** in § 3.

The probabilistic d–clustering algorithm is presented in § 4. It is based on the **joint distance function**, see § 2.2 and (34), a distance function that captures the data in its low contours. The centers are updated by the algorithm as stationary points of the joint distance function.

We end with a small problem, of clustering 9 judges, in § 5.

For other approaches to probabilistic clustering see the surveys in [2], [5].

2 Probabilistic d–clustering

There are several ways to model the relationship between distances and probabilities. The simplest model is as follows.

Principle 1. For each $\mathbf{x} \in \mathcal{D}$, and each cluster $\mathcal{C}_k$, the probability $p_k(\mathbf{x})$ is inversely proportional to $d_k(\mathbf{x})$, i.e.,

$$p_k(\mathbf{x})\, d_k(\mathbf{x}) = \text{constant, depending on } \mathbf{x}\ . \tag{6}$$

Cluster membership is thus more probable the closer the data point is to the cluster center.

2.1 Probabilities

From Principle 1 and the fact that probabilities add to 1 we get

Theorem 1. Let the cluster centers $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K\}$ be given, let $\mathbf{x}$ be a data point, and let $\{d_k(\mathbf{x}) : k = 1, \ldots, K\}$ be its distances from the given centers. Then the membership probabilities of $\mathbf{x}$ are

$$p_k(\mathbf{x}) = \frac{\prod\limits_{j \neq k} d_j(\mathbf{x})}{\sum\limits_{i=1}^{K} \prod\limits_{j \neq i} d_j(\mathbf{x})}\ , \ k = 1, \ldots, K. \tag{7}$$

Proof. Using (6) we write for i, k

$$p_i(\mathbf{x}) = \left(\frac{p_k(\mathbf{x}) d_k(\mathbf{x})}{d_i(\mathbf{x})} \right)\ .$$

Since $\sum\limits_{i=1}^{K} p_i(\mathbf{x}) = 1$,

$$p_k(\mathbf{x}) \sum_{i=1}^{K} \left(\frac{d_k(\mathbf{x})}{d_i(\mathbf{x})} \right) = 1.$$

$$\therefore\ p_k(\mathbf{x}) = \frac{1}{\sum\limits_{i=1}^{K} \left(\frac{d_k(\mathbf{x})}{d_i(\mathbf{x})} \right)} = \frac{\prod\limits_{j \neq k} d_j(\mathbf{x})}{\sum\limits_{i=1}^{K} \prod\limits_{j \neq i} d_j(\mathbf{x})} \qquad \square$$

In particular, for K=2,

$$p_1(\mathbf{x}) = \frac{d_2(\mathbf{x})}{d_1(\mathbf{x}) + d_2(\mathbf{x})}\ , \ p_2(\mathbf{x}) = \frac{d_1(\mathbf{x})}{d_1(\mathbf{x}) + d_2(\mathbf{x})}\ , \tag{8}$$

and for $K = 3$,

$$p_1(\mathbf{x}) = \frac{d_2(\mathbf{x})d_3(\mathbf{x})}{d_1(\mathbf{x})d_2(\mathbf{x}) + d_1(\mathbf{x})d_3(\mathbf{x}) + d_2(\mathbf{x})d_3(\mathbf{x})} \text{ , etc.} \tag{9}$$

2.2 The joint distance function

We denote the constant in (6) by $D(\mathbf{x})$, a function of $\mathbf{x}$. Then

$$p_k(\mathbf{x}) = \frac{D(\mathbf{x})}{d_k(\mathbf{x})} \text{ , } k = 1, \ldots, K$$

Since the probabilities add to 1 we get,

$$D(\mathbf{x}) = \frac{\prod_{j=1}^{K} d_j(\mathbf{x})}{\sum_{i=1}^{K} \prod_{j \neq i} d_j(\mathbf{x})} \text{ .} \tag{10}$$

$D(\mathbf{x})$ has the dimension of distance, and measures the distance of $\mathbf{x}$ from all cluster centers, which is why we call it the **joint distance function** of $\mathbf{x}$.

Special cases of (10), for $K = 2$,

$$D(\mathbf{x}) = \frac{d_1(\mathbf{x})\, d_2(\mathbf{x})}{d_1(\mathbf{x}) + d_2(\mathbf{x})} \text{ ,} \tag{11}$$

and for $K = 3$,

$$D(\mathbf{x}) = \frac{d_1(\mathbf{x})\, d_2(\mathbf{x})\, d_3(\mathbf{x})}{d_1(\mathbf{x})\, d_2(\mathbf{x}) + d_1(\mathbf{x})\, d_3(\mathbf{x}) + d_2(\mathbf{x})\, d_3(\mathbf{x})} \text{ .} \tag{12}$$

Example 2. Figure 2 shows level sets of the joint distance function (11), with Mahalanobis distances

$$d(\mathbf{x}, \mathbf{c}_k) = \sqrt{(\mathbf{x} - \mathbf{c}_k)^T \Sigma_k^{-1} (\mathbf{x} - \mathbf{c}_k)} \text{ ,} \tag{13}$$

$\mathbf{c}_1 = \boldsymbol{\mu}_1,\ \mathbf{c}_2 = \boldsymbol{\mu}_2$, and $\Sigma_1,\ \Sigma_2$ as in Example 1.

2.3 An extremal principle

For simplicity we consider here the case of 2 clusters, but the results are easily extended to the general case.

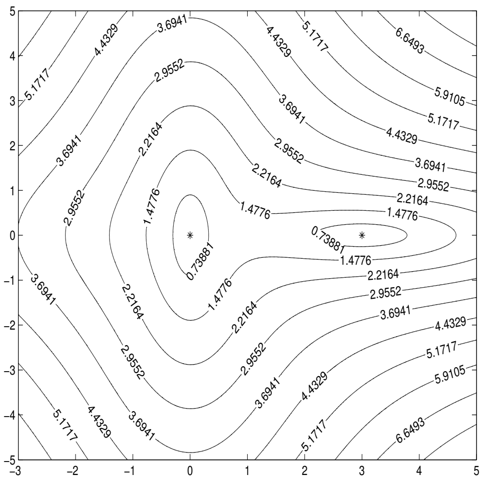

Figure 2: Level sets of a joint distance function

Let $\mathbf{x}$ be a given data point with distances $d_1(\mathbf{x})$, $d_2(\mathbf{x})$ to the cluster centers. Then the probabilities in (8) are the optimal solutions of the extremal problem

$$\begin{aligned} \min \quad & d_1(\mathbf{x})\, p_1^2 + d_2(\mathbf{x})\, p_2^2 \\ \text{s.t.} \quad & p_1 + p_2 = 1 \\ & p_1,\, p_2 \geq 0 \end{aligned} \tag{14}$$

Indeed, the Lagrangian of this problem is

$$L(p_1, p_2, \lambda) = d_1(\mathbf{x})\, p_1^2 + d_2(\mathbf{x})\, p_2^2 - \lambda(p_1 + p_2 - 1) \tag{15}$$

and zeroing the gradient (with respect to p_1, p_2) gives the principle (6),

$$p_1\, d_1(\mathbf{x}) = p_2\, d_2(\mathbf{x})$$

Substituting the probabilities (8) in the Lagrangian (15) we get the optimal value of (14),

$$L^*(p_1(\mathbf{x}), p_2(\mathbf{x}), \lambda) = \frac{d_1(\mathbf{x})\, d_2(\mathbf{x})}{d_1(\mathbf{x}) + d_2(\mathbf{x})} \tag{16}$$

which is again the joint distance function (11).

The extremal problem for a data set $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\} \subset \mathbb{R}^n$, using the above notation, is

$$\begin{aligned} \min \quad & \sum_{i=1}^{N} \left(d_1(\mathbf{x}_i)\, p_1(i)^2 + d_2(\mathbf{x}_i)\, p_2(i)^2 \right) \\ \text{s.t.} \quad & p_1(i) + p_2(i) = 1 \\ & p_1(i),\, p_2(i) \geq 0 \end{aligned} \tag{17}$$

where $p_1(i), p_2(i)$ are the cluster probabilities at $\mathbf{x}_i$. The problem separates into N problems like (14), and its optimal value is

$$\sum_{i=1}^{N} \frac{d_1(\mathbf{x}_i)\, d_2(\mathbf{x}_i)}{d_1(\mathbf{x}_i) + d_2(\mathbf{x}_i)} \tag{18}$$

the sum of the joint distance functions of all points.

2.4 Centers

We rewrite (18) as a function of the cluster centers $\mathbf{c}_1, \mathbf{c}_2$,

$$f(\mathbf{c}_1, \mathbf{c}_2) = \sum_{i=1}^{N} \frac{d(\mathbf{x}_i, \mathbf{c}_1)\, d(\mathbf{x}_i, \mathbf{c}_2)}{d(\mathbf{x}_i, \mathbf{c}_1) + d(\mathbf{x}_i, \mathbf{c}_2)} = \sum_{i=1}^{N} \frac{\|\mathbf{x}_i - \mathbf{c}_1\|\, \|\mathbf{x}_i - \mathbf{c}_2\|}{\|\mathbf{x}_i - \mathbf{c}_1\| + \|\mathbf{x}_i - \mathbf{c}_2\|} \tag{19}$$

and look for centers $\mathbf{c}_1, \mathbf{c}_2$ minimizing f. The function f is not convex, not even quasi–convex (although each of its summands is), and the sought minimum may be local.

Theorem 2. A stationary point of the function f of (19), with Euclidean norm $\| \cdot \|$, is

$$\mathbf{c}_1 = \sum_{i=1}^{N} \left(\frac{u_1(\mathbf{x}_i)}{\sum\limits_{j=1}^{N} u_1(\mathbf{x}_j)} \right) \mathbf{x}_i \ , \quad \mathbf{c}_2 = \sum_{i=1}^{N} \left(\frac{u_2(\mathbf{x}_i)}{\sum\limits_{j=1}^{N} u_2(\mathbf{x}_j)} \right) \mathbf{x}_i \ , \tag{20}$$

where

$$u_1(\mathbf{x}_i) = \frac{d_2(\mathbf{x}_i)^2 / d_1(\mathbf{x}_i)}{(d_1(\mathbf{x}_i) + d_2(\mathbf{x}_i))^2} \ , \quad u_2(\mathbf{x}_i) = \frac{d_1(\mathbf{x}_i)^2 / d_2(\mathbf{x}_i)}{(d_1(\mathbf{x}_i) + d_2(\mathbf{x}_i))^2} \ . \tag{21}$$

or equivalently, in terms of the probabilities (8),

$$u_1(\mathbf{x}_i) = \frac{p_1(\mathbf{x}_i)^2}{d_1(\mathbf{x}_i)} \ , \quad u_2(\mathbf{x}_i) = \frac{p_1(\mathbf{x}_i)^2}{d_2(\mathbf{x}_i)} \ . \tag{22}$$

Proof. The gradient of $d(\mathbf{x}, \mathbf{c}) = \|\mathbf{x} - \mathbf{c}\|$ with respect to $\mathbf{c}$ is

$$\nabla_{\mathbf{c}} \|\mathbf{x} - \mathbf{c}\| = -\frac{\mathbf{x} - \mathbf{c}}{\|\mathbf{x} - \mathbf{c}\|} = -\frac{\mathbf{x} - \mathbf{c}}{d(\mathbf{x}, \mathbf{c})} \ .$$

Therefore,

$$\begin{aligned} \nabla_{\mathbf{c}_1} f(\mathbf{c}_1, \mathbf{c}_2) &= \sum_{i=1}^{N} \frac{(d_1(\mathbf{x}_i) + d_2(\mathbf{x}_i))\, d_2(\mathbf{x}_i) \left(-\frac{\mathbf{x}_i - \mathbf{c}_1}{d_1(\mathbf{x}_i)} \right) + d_1(\mathbf{x}_i)\, d_2(\mathbf{x}_i) \left(\frac{\mathbf{x}_i - \mathbf{c}_1}{d_1(\mathbf{x}_i)} \right)}{(d_1(\mathbf{x}_i) + d_2(\mathbf{x}_i))^2} \\ &= \sum_{i=1}^{N} \frac{-d_2(\mathbf{x}_i)^2 \left(\frac{\mathbf{x}_i - c_1}{d_1(\mathbf{x}_i)} \right)}{(d_1(\mathbf{x}_i) + d_2(\mathbf{x}_i))^2} \end{aligned} \tag{23}$$

Zeroing the gradient (23) we get

$$\sum_{j=1}^{N} \frac{\dfrac{d_2(\mathbf{x}_j)^2}{d_1(\mathbf{x}_j)}\,\mathbf{c}_1}{(d_1(\mathbf{x}_j)+d_2(\mathbf{x}_j))^2} = \sum_{i=1}^{N} \frac{\dfrac{d_2(\mathbf{x}_i)^2}{d_1(\mathbf{x}_i)}\,\mathbf{x}_i}{(d_1(\mathbf{x}_i)+d_2(\mathbf{x}_i))^2}\ , \tag{24}$$

which can be written, using (21), as

$$\sum_{j=1}^{N} u_1(\mathbf{x}_j)\,\mathbf{c}_1 = \sum_{i=1}^{N} u_1(\mathbf{x}_i)\,\mathbf{x}_i$$

proving $\mathbf{c}_1$ in (20). The equivalence of (21) and (22) follows from (8).

The rest follows similarly from $\nabla_{\mathbf{c}_2} f(\mathbf{c}_1,\mathbf{c}_2) = \mathbf{0}$. □

The same formulas for the centers $\mathbf{c}_1, \mathbf{c}_2$ hold if the norm used in (19) is elliptic, (2), instead of the Euclidean norm.

Corollary 1. Consider the function $f(\mathbf{c}_1,\mathbf{c}_2)$

$$f(\mathbf{c}_1,\mathbf{c}_2) = \sum_{i=1}^{N} \frac{d(\mathbf{x}_i,\mathbf{c}_1)\,d(\mathbf{x}_i,\mathbf{c}_2)}{d(\mathbf{x}_i,\mathbf{c}_1)+d(\mathbf{x}_i,\mathbf{c}_2)} = \sum_{i=1}^{N} \frac{\|\mathbf{x}_i-\mathbf{c}_1\|\,\|\mathbf{x}_i-\mathbf{c}_2\|}{\|\mathbf{x}_i-\mathbf{c}_1\|+\|\mathbf{x}_i-\mathbf{c}_2\|}\ , \tag{19}$$

using the distance function,

$$d(\mathbf{x},\mathbf{c}) = \langle(\mathbf{x}-\mathbf{c}), Q(\mathbf{x}-\mathbf{c})\rangle^{1/2}, \tag{25}$$

for some positive–definite matrix Q. Then a stationary point $(\mathbf{c}_1,\mathbf{c}_2)$ of f is given by (20)–(21).

Proof. The proof of Theorem 2 can be adapted. The result analogous to (24) is

$$\sum_{i=1}^{N} \frac{\dfrac{d_2(\mathbf{x}_i)^2}{d_1(\mathbf{x}_i)}\,Q\,\mathbf{c}_1}{(d_1(\mathbf{x}_i)+d_2(\mathbf{x}_i))^2} = \sum_{i=1}^{N} \frac{\dfrac{d_2(\mathbf{x}_i)^2}{d_1(\mathbf{x}_i)}\,Q\,\mathbf{x}_i}{(d_1(\mathbf{x}_i)+d_2(\mathbf{x}_i))^2}$$

where Q can be cancelled. □

Corollary 1 applies, in particular, to the Mahalanobis distance

$$d(\mathbf{x},\mathbf{c}_k) = \sqrt{(\mathbf{x}-\mathbf{c}_k)^T\Sigma_k^{-1}(\mathbf{x}-\mathbf{c}_k)} \tag{13}$$

where Σ_k is the (given or computed) covariance matrix of the cluster $\mathcal{C}_k$.

The probabilistic form (22) is convenient for the general case of K clusters.

Corollary 2. Let the sum, for all data points, of the joint distance functions (10), be written as a function of the cluster centers $\mathbf{c}_k$,

$$f(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K) = \sum_{i=1}^{N} \frac{\prod_{j=1}^{K} \|\mathbf{x}_i - \mathbf{c}_j\|}{\sum_{k=1}^{K} \prod_{j \neq k} \|\mathbf{x}_i - \mathbf{c}_j\|} . \tag{26}$$

Then the point $(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K)$ with

$$\mathbf{c}_k = \sum_{i=1}^{N} \left(\frac{u_k(\mathbf{x}_i)}{\sum_{j=1}^{K} u_j(\mathbf{x}_i)} \right) \mathbf{x}_i \ , \quad \text{with } u_k(\mathbf{x}_i) = \frac{p_k(\mathbf{x}_i)^2}{d_k(\mathbf{x}_i)} \ , \ k = 1, \ldots, K, \tag{27}$$

is stationary for f.

Proof. Similar to the proof of Theorem 2. □

2.5 Other principles

There are alternative ways of modelling the relations between distances and probabilities. For example:

Principle 2. For each $\mathbf{x} \in \mathcal{D}$, and each cluster $\mathcal{C}_k$, the probability $p_k = p_k(\mathbf{x})$ and distance $d_k = d_k(\mathbf{x})$ are related by

$$p_k^{\alpha}\, d_k = \text{constant, depending on } \mathbf{x} \, . \tag{28}$$

where the exponent α is nonzero.

For the case of 2 clusters we get, in analogy with (8) and (17) respectively, the probabilities

$$p_1(\mathbf{x}) = \frac{d_2(\mathbf{x})^{1/\alpha}}{d_1(\mathbf{x})^{1/\alpha} + d_2(\mathbf{x})^{1/\alpha}} \ , \ p_2(\mathbf{x}) = \frac{d_1(\mathbf{x})^{1/\alpha}}{d_1(\mathbf{x})^{1/\alpha} + d_2(\mathbf{x})^{1/\alpha}} \ , \tag{29}$$

and an extremal principle,

$$\begin{aligned} \min \quad & \sum_{i=1}^{N} \left(d_1(\mathbf{x}_i)\, p_1(i)^{\alpha+1} + d_2(\mathbf{x}_i)\, p_2(i)^{\alpha+1} \right) \\ \text{s.t.} \quad & p_1(i) + p_2(i) = 1 \\ & p_1(i),\ p_2(i) \geq 0 \end{aligned} \tag{30}$$

where $p_1(i), p_2(i)$ are the cluster probabilities at $\mathbf{x}_i$.

Another relation between probabilities and distances is studied in the next section.

3 Probabilistic exponential d–clustering

The following relation has probabilities decaying exponentially as distances increase.

Principle 3. For each $\mathbf{x} \in \mathcal{D}$, and each cluster $\mathcal{C}_k$, the probability $p_k = p_k(\mathbf{x})$ and distance $d_k = d_k(\mathbf{x})$ are related by

$$p_k e^{d_k} = C, \text{ where } C \text{ is a constant, depending on } \mathbf{x} \; . \tag{31}$$

The analog of the probabilities (8) for Principle 3 is

$$p_1(\mathbf{x}) = \frac{e^{d_2(\mathbf{x})}}{e^{d_1(\mathbf{x})} + e^{d_2(\mathbf{x})}} \;,\; p_2(\mathbf{x}) = \frac{e^{d_1(\mathbf{x})}}{e^{d_1(\mathbf{x})} + e^{d_2(\mathbf{x})}} \;, \tag{32}$$

or equivalently

$$p_1(\mathbf{x}) = \frac{e^{-d_1(\mathbf{x})}}{e^{-d_1(\mathbf{x})} + e^{-d_2(\mathbf{x})}} \;,\; p_2(\mathbf{x}) = \frac{e^{-d_2(\mathbf{x})}}{e^{-d_1(\mathbf{x})} + e^{-d_2(\mathbf{x})}} \;. \tag{33}$$

We denote the constant in (31) by $E(\mathbf{x})$, a function of $\mathbf{x}$. Then

$$p_k(\mathbf{x}) = \frac{E(\mathbf{x})}{e^{d_k(\mathbf{x})}}, \; k = 1, 2 \; .$$

Since the probabilities add to 1 we get

$$E(\mathbf{x}) = \frac{e^{d_1(\mathbf{x}) + d_2(\mathbf{x})}}{e^{d_1(\mathbf{x})} + e^{d_2(\mathbf{x})}} \;, \tag{34}$$

called the **exponential joint distance function**.

3.1 An extremal principle

The probabilities (32) are the optimal solutions of the problem

$$\min_{p_1, p_2} \quad \left\{ \sum_{k=1}^{2} e^{d_k} \, p_k^2 \; : \; p_1 + p_2 = 1 \; , \; p_1, \, p_2 \geq 0 \right\} \;, \tag{35}$$

whose optimal value, obtained by substituting the probabilities (32), is again the exponential joint distance function (34).

The extremal problem for a data set $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\} \subset \mathbb{R}^n$, partitioned into 2 clusters, is the following analog of (17)

$$\begin{aligned} \min \quad & \sum_{i=1}^{N} \left(e^{d_1(\mathbf{x}_i)} \, p_1(i)^2 + e^{d_2(\mathbf{x}_i)} \, p_2(i)^2 \right) \\ \text{s.t.} \quad & p_1(i) + p_2(i) = 1 \\ & p_1(i), \, p_2(i) \geq 0 \end{aligned} \tag{36}$$

where $p_1(i), p_2(i)$ are the cluster probabilities at $\mathbf{x}_i$. The problem separates into N problems like (35), and its optimal value is

$$\sum_{i=1}^{N} \frac{e^{d_1(\mathbf{x}_i)+d_2(\mathbf{x}_i)}}{e^{d_1(\mathbf{x}_i)} + e^{d_2(\mathbf{x}_i)}} \ , \tag{37}$$

the sum of the joint distances of all points.

Alternatively, (31) follows from the extremal principle

$$\min_{p_1,p_2} \quad \left\{ \sum_{k=1}^{2} p_k \, d_k + \sum_{k=1}^{2} p_k \, \log p_k \ : \ p_1 + p_2 = 1 \ , \ p_1, \, p_2 \geq 0 \right\} \ , \tag{38}$$

obtained by adding an entropy term to $\sum p_k \, d_k$. Indeed the Lagrangian of (38) is

$$L(p_1, p_2, \lambda) = \sum_{k=1}^{2} p_k \, d_k + \sum_{k=1}^{2} p_k \, \log p_k - \lambda \, (p_1 + p_2 - 1) \ .$$

Differentiation with respect to p_k, and equating to 0, gives

$$d_k + 1 + \log p_k - \lambda = 0$$

which is (31).

3.2 Centers

We rewrite (37) as a function of the cluster centers $\mathbf{c}_1, \mathbf{c}_2$,

$$f(\mathbf{c}_1, \mathbf{c}_2) = \sum_{i=1}^{N} \frac{e^{d(\mathbf{x}_i,\mathbf{c}_1)+d(\mathbf{x}_i,\mathbf{c}_2)}}{e^{d(\mathbf{x}_i,\mathbf{c}_1)} + e^{d(\mathbf{x}_i,\mathbf{c}_2)}} = \sum_{i=1}^{N} \frac{e^{\|\mathbf{x}_i-\mathbf{c}_1\|+\|\mathbf{x}_i-\mathbf{c}_2\|}}{e^{\|\mathbf{x}_i-\mathbf{c}_1\|} + e^{\|\mathbf{x}_i-\mathbf{c}_2\|}} \tag{39}$$

and look for centers $\mathbf{c}_1, \mathbf{c}_2$ minimizing f. As in Theorem 2 we can prove that a stationary point of f is

$$\mathbf{c}_1 = \sum_{i=1}^{N} \left(\frac{u_1(\mathbf{x}_i)}{\sum\limits_{j=1}^{N} u_1(\mathbf{x}_j)} \right) \mathbf{x}_i \ , \quad \mathbf{c}_2 = \sum_{i=1}^{N} \left(\frac{u_2(\mathbf{x}_i)}{\sum\limits_{j=1}^{N} u_2(\mathbf{x}_j)} \right) \mathbf{x}_i \ , \tag{40}$$

where

$$u_1(\mathbf{x}_i) = \frac{e^{-d_1(\mathbf{x}_i)}/d_1(\mathbf{x}_i)}{(e^{-d_1(\mathbf{x}_i)} + e^{-d_2(\mathbf{x}_i)})^2} \ , \quad u_2(\mathbf{x}_i) = \frac{e^{-d_2(\mathbf{x}_i)}/d_2(\mathbf{x}_i)}{(e^{-d_1(\mathbf{x}_i)} + e^{-d_2(\mathbf{x}_i)})^2} \ , \tag{41}$$

or equivalently, using the probabilities (33),

$$u_1(\mathbf{x}_i) = \frac{p_1(\mathbf{x}_i)^2\, e^{d_1(\mathbf{x}_i)}}{d_1(\mathbf{x}_i)} \ , \quad u_2(\mathbf{x}_i) = \frac{p_2(\mathbf{x}_i)^2\, e^{d_2(\mathbf{x}_i)}}{d_2(\mathbf{x}_i)} \ , \tag{42}$$

As in Corollary 1 we can prove that (40)–(41) are correct also for elliptic distance (25), in particular, for the Mahalanobis distance (13).

In the case of $K \geq 2$ clusters we have the following analog of Corollary 2: The point $(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K)$ with,

$$\mathbf{c}_k = \sum_{i=1}^{N} \left(\frac{u_k(\mathbf{x}_i)}{\sum\limits_{j=1}^{K} u_j(\mathbf{x}_i)} \right) \mathbf{x}_i \ , \ \text{and } u_k(\mathbf{x}_i) = \frac{p_k(\mathbf{x}_i)^2\, e^{d_1(\mathbf{x}_i)}}{d_k(\mathbf{x}_i)} \ , \tag{43}$$

for $k = 1, \ldots, K$, is a stationary point of the exponential joint distance function, summed over all data points $\mathbf{x}_i$.

4 A probabilistic d–clustering algorithm

The above ideas are implemented in the following algorithm for the unsupervised clustering of data, presented – for simplicity – for the case of 2 clusters.

A schematic description, valid for both §§ 2–3, follows.

Initialization:	given data $\mathcal{D}$, any two points $\mathbf{c}_1, \mathbf{c}_2$, and $\epsilon > 0$
Iteration:	
Step 1	**compute** distances $d_1(\mathbf{x})$, $d_2(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{D}$
Step 2	**update** the centers $\mathbf{c}_1^+, \mathbf{c}_2^+$
Step 3	**if** $\|\mathbf{c}_1^+ - \mathbf{c}_1\| + \|\mathbf{c}_2^+ - \mathbf{c}_2\| < \epsilon$ **stop**
	return to step 1

The algorithm iterates between the cluster **centers**, (20) or (40), expressed as stationary points for the joint distance function (19) or (39), respectively, and the **distances** of the data points to these centers. The cluster **probabilities**, (8) or (32), are not used explicitly.

Notes:

(a) The distance used in Step 1 can be Euclidean or elliptic (the formulas (20)–(21), and (40)–(41), are valid in both cases.)

(b) In particular, if the Mahalanobis distance

$$d(\mathbf{x}, \mathbf{c}_k) = \sqrt{(\mathbf{x} - \mathbf{c}_k)^T \Sigma_k^{-1} (\mathbf{x} - \mathbf{c}_k)} \tag{13}$$

is used, the covariance matrix Σ_k of the k^{th}–cluster, can be estimated at each iteration by

$$\Sigma_k = \frac{\sum_{i=1}^{N} u_k(\mathbf{x}_i)(\mathbf{x}_i - \mathbf{c}_k)(\mathbf{x}_i - \mathbf{c}_k)^T}{\sum_{i=1}^{N} u_k(\mathbf{x}_i)} \tag{44}$$

with $u_k(\mathbf{x}_i)$ given by (21) or (41)

(c) If the joint distance function (11) is used, then the centers are updated in Step 3 using (20)–(21). If the exponential joint distance function (34) is used, the centers are updated with (40)–(41).

(d) The computations stop (in Step 3) when the centers stop moving, at which point the cluster membership probabilities may be computed by (8) or (32). These probabilities are not used in the algorithm, but are needed for classifying the data.

Example 3. We apply the algorithm, using d–clustering as in § 2 and Mahalanobis distance, to the data of Example 1. Figure 3 shows the evolution of the joint distance function, represented by its level sets. The initial function, shown in the top-left pane, corresponds to the (arbitrarily chosen) initial centers and initial covariances $\Sigma_1 = \Sigma_2 = I$. The covariances are updated at each iteration using (44), and by iteration 8 the function is already very close to its final form, shown in the bottom-right pane. The algorithm terminated in iteration 12.

Example 4. In Figure 4 we illustrate the movement of the cluster centers for different initial centers. The centers at each run are shown with the final level sets of the joint distance function found in Example 3.

The algorithm gives the correct cluster centers, for all initial starts. In particular, the two initial centers may be arbitrarily close, as shown in the top–left pane of Fig. 4.

Example 5. The class membership probabilities (8) were then computed using the centers determined by the algorithm. The level sets of the probability $p_1(\mathbf{x})$ are shown in Figure 5. The curve $p_1(\mathbf{x}) = 0.5$, the thick curve shown in the left pane of Fig. 5, may serve as the clustering rule. Alternatively, the 2 clusters can be defined as

$$\mathcal{C}_1 = \{\mathbf{x} : \, p_1(\mathbf{x}) \geq 0.6\}, \; \mathcal{C}_2 = \{\mathbf{x} : \, p_1(\mathbf{x}) \leq 0.4\} \; ,$$

with points $\{\mathbf{x} : \, 0.4 < p_1(\mathbf{x}) < 0.6\}$ left unclassified, see the right pane of Fig. 5.

5 The Liberal-Conservative Divide of the Rehnquist Court

The **Rehnquist Supreme Court** was analyzed by Hubert and Steinley in [3], where the justices were ranked as follows, from most liberal to most conservative.

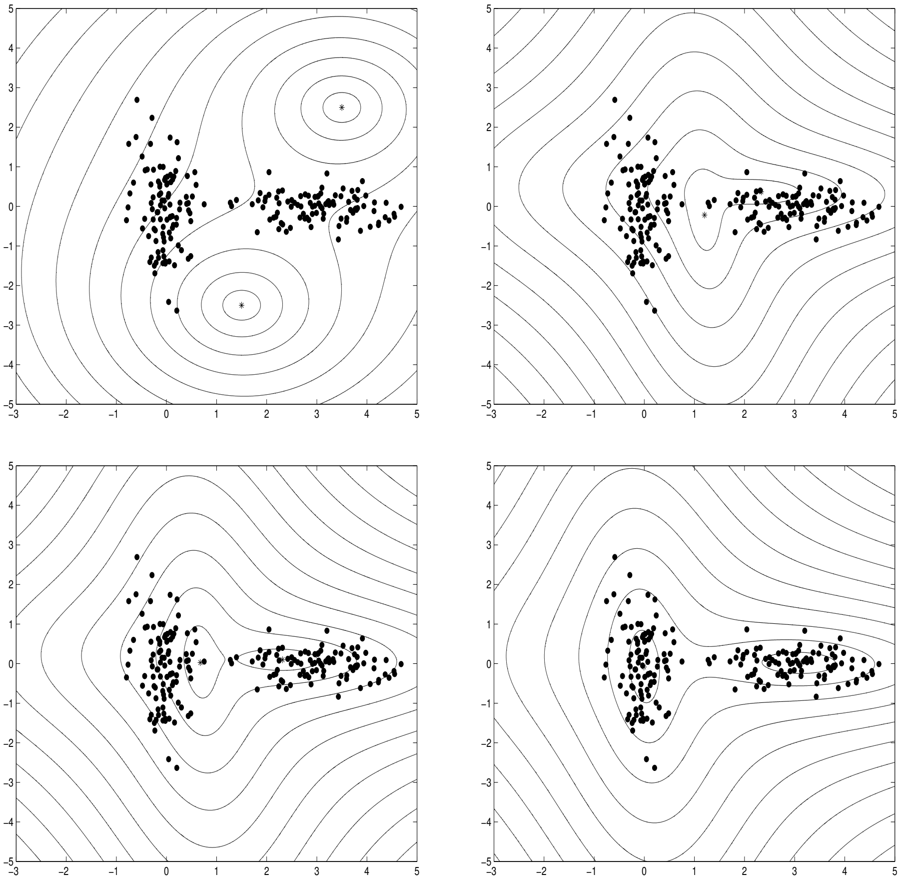

Figure 3: The level sets of the evolving **joint distance function** at **iteration 0** (top left), **iteration 1** (top right), **iteration 2** (bottom left) and **iteration 12** (bottom right)

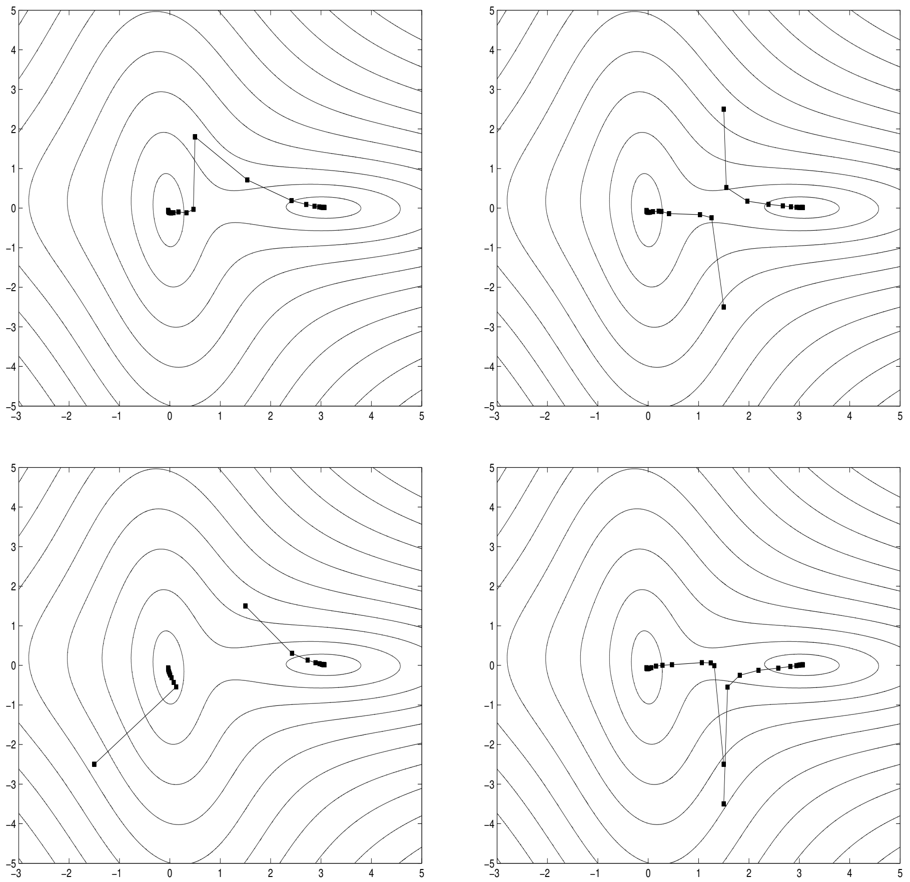

Figure 4: Movements of the cluster centers for different starts. The top–right pane shows the centers corresponding to Fig. 3. The top–left pane shows very close initial centers.

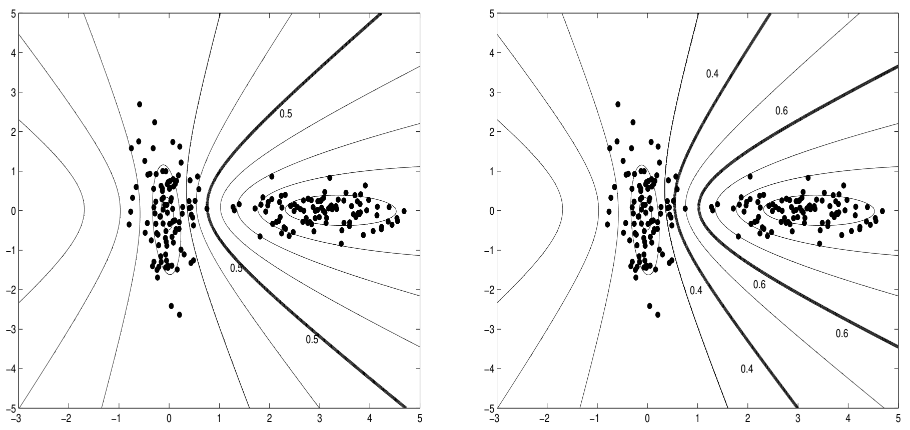

Figure 5: The level sets of the probabilities $p_1(\mathbf{x})$ and two clustering rules.

1. John Paul Stevens (St)
2. Stephen G.Breyer (Br)
3. Ruth Bader Ginsberg (Gi)
4. David Souter (So)
5. Sandra Day O'Connor (Oc)
6. Anthony M. Kennendy (Ke)
7. William H. Rehnquist (Re)
8. Antonin Scalia (Sc)
9. Clarence Thomas (Th)

The data used in the analysis are the percentages of non-unanimous cases in which the justices *agreed*, listed in Table 1 (a mirror image of Table 1 in [3], listing the disagreements.)

	St	Br	Gi	So	Oc	Ke	Re	Sc	Th
1 St	1.00	.62	.66	.63	.33	.36	.25	.14	.15
2 Br	.62	1.00	.72	.71	.55	.47	.43	.25	.24
3 Gi	.66	.72	1.00	.78	.47	.49	.43	.28	.26
4 So	.63	.71	.78	1.00	.55	.50	.44	.31	.29
5 Oc	.33	.55	.47	.55	1.00	.67	.71	.54	.54
6 Ke	.36	.47	.49	.50	.67	1.00	.77	.58	.59
7 Re	.25	.43	.43	.44	.71	.77	1.00	.66	.68
8 Sc	.14	.25	.28	.31	.54	.58	.66	1.00	.79
9 Th	.15	.24	.26	.29	.54	.59	.68	.79	1.00

Table 1. Similarities among the nine Supreme Court justices

Hubert and Steinley used two methods, *unidimensional scaling* (projecting the data from $\mathbb{R}^9$ to $\mathbb{R}$), and *hierarchical classification*, see [3] for details.

We applied our clustering method to the Rehnquist Court, with Justices represented by points $\mathbf{x}$ in $\mathbb{R}^9$ (the columns of Table 1), using the Euclidean distance in $\mathbb{R}^9$. Our results are given in the following table, listing the clusters and their membership probabilities.

Cluster	Justice	Membership Probability
Liberal	Ruth Bader Ginsburg	0.8685
	David Souter	0.8390
	Stephen Breyer	0.7922
	John Paul Stevens	0.7144
Conservative	William Rehnquist	0.8966
	Anthony Kennedy	0.7540
	Clarence Thomas	0.7220
	Antonin Scalia	0.7173
	Sandra Day O'Connor	0.6740

Table 2. The liberal-conservative divide of the Rehnquist Court

The membership probability of a Justice in a cluster is, by (6), proportional to the proximity to the cluster center, and is thus a measure of the agreement of the Justice with others in the cluster.

Since not all non-unanimous cases were equally important, or equally revealing of ideology, we should not read into these probabilities more than is supported by the data. For example, Justice Kennedy (probability 0.7540) is not “more conservative” than Justice Scalia (probability 0.7173), but perhaps “more conformist” with the “conservative center”.

Similarly, Justice Stevens, ranked “most liberal” in [3], is in our analysis the “least conformist” in the liberal cluster.

Overall, the liberal cluster is tighter, and more conformist, than the conservative cluster.

6 Conclusions

The **probabilistic distance clustering algorithm** presented here is simple, fast (requiring a small number of cheap iterations), robust (insensitive to outliers), and gives a high percentage of correct classifications.

It was tried on hundreds of problems with both simulated and real data sets. In all cases the algorithm, starting at random initial centers, converged to the true cluster centers.

The results of our numerical experiments, and the comparison with other distance–based clustering algorithms, will be reported elsewhere. Our purpose here was to present the main idea, which will find applications in other areas, such as facility location, pattern recognition, and data mining.

References

[1] HARTIGAN J. (1975), *Clustering Algorithms*, J. Wiley.

[2] HÖPPNER F., KLAWONN F., KRUSE R., RUNKLER T. (1999), *Fuzzy Cluster Analysis*. John Wiley & Sons.

[3] HUBERT L., STEINLEY D.(2005), ”Agreement Among Supreme Court Justices: Categorical vs. Continuous Representation”, *SIAM News*, 38(7).

[4] JAIN A.K., DUBES R.C. (1988) *Algorithms for Clustering Data*, Prentice Hall.

[5] TAN P., STEINBACH M., KUMAR V. (2006), *Introduction to Data Mining*, Addison Wesley.