
Geodesic Clustering

Anonymous Author(s)

Affiliation

Address

email

Abstract

We propose a new class of distances for the purpose of data clustering, called the *geodesic* distance, and introduce a geodesic extension of K-medoids algorithm. We analyze the theoretical properties of the geodesic distance within a clustering framework and prove that the geodesic K-medoids algorithm converges to the correct clustering assignment in the asymptotic regime, even in the presence of outliers. We also present experimental evidence on the abilities of geodesic K-medoids to handle clustering problems involving outliers, nonlinearly separable clusters, and varying densities in the clusters. The results are compared to a few hierarchical and spectral clustering algorithms on several clustering problems.

1 Introduction

Data clustering is an important problem central to the field of unsupervised learning with various applications ranging from data mining and statistics [1] to biology [2]. Generally speaking, the purpose of clustering is to partition a given set of data points into groups such that the points in each group are “more similar” to each other than to points in other groups. The notion of similarity is collectively defined by a distance (or dissimilarity) measure over the data points, and a loss function which becomes the optimization objective of the clustering algorithm.

The K-means algorithm [1] as one of the most well-known clustering techniques is also known to fail for instance when clusters are not linearly separable. Many modifications and extensions to K-means algorithm have been devised over the years to address these issues, most of which are surveyed in [3]. K-medoids [1] clustering is one such modification which, unlike K-means, does not require a Euclidean representation of data points. This purely distance-based nature of K-medoids makes it a suitable candidate to pair with a non-Euclidean class of distances, as we will see shortly.

More recent developments in this area has given rise to modern clustering approaches such as hierarchical and spectral methods. Hierarchical clustering relies on a measure of similarity between disjoint subsets of data points to construct a hierarchy of clusters. Most commonly used among them are the agglomerative methods which take a bottom-up approach, iteratively merging clusters together until the desired number of clusters is reached. Depending on the specific similarity measure in use, there are various agglomerative clustering algorithms including single linkage, complete linkage, and average linkage to name a few [1]. Classical hierarchical clustering algorithms usually suffer from a lack of robustness in presence of outliers. More recent hierarchical techniques, such as CURE [4] and BIRCH [5], offer significant improvements in this regard, but still suffer from similar issues.

Spectral clustering [6, 7, 8] is another branch of work with remarkable success. These methods rely on weighted similarity graphs which they attempt to cut into a given number of clusters with the least weighted cuts possible, while maintaining the “size” of the clusters balanced. Intuitively, the similarity graph captures the local connectivity structure of the data points and the optimization algorithm tries to cut through the least similar regions of the graph to form the clusters. Despite

outstanding robustness of spectral clustering in dealing with outliers and non-convex clusters in data, there are aspects of these algorithms that limit their performance on certain classes of datasets. While not a valid assumption in general, the loss function in nearly all spectral methods implicitly attempts to evenly balance clusters in size (or volume). As we demonstrate in Section 4, normalized spectral clustering [6] can lead to incorrect clustering (for all values of σ) when clusters vary significantly in size, density, and proximity to each other.

Almost all clustering algorithms rely on some method of measuring the dissimilarity between pairs of data points. This distance measure, as we refer to it in this paper, should not only capture the connectivity of the points in each cluster, but also separate different clusters in presence of outliers between them. In this paper we focus on this important element at the core of clustering algorithms. Specifically, we introduce the class of *geodesic distances* which rely on a sparse neighborhood graph over the data points together with local density weights on its edges. Our claim is that the shortest weighted paths on such neighborhood graphs provide the desirable characteristics outlined above.

To validate our claim, we introduce geodesic K-medoids and analyze its properties theoretically and experimentally. While geodesic distance can be applied to any distance-based clustering algorithm, we choose K-medoids because of its simplicity in order to further emphasize the power of geodesic distance. As we will demonstrate empirically, the performance of geodesic K-medoids proves to be competitive with state-of-the-art clustering techniques such as spectral clustering, and in some cases even outperforms these algorithms. Our ultimate goal in this paper is to advance the understanding of how distance measures affect the behavior of clustering algorithms and how to build better distance metrics for clustering purposes.

The initial inspiration for our work comes from ISOMAP manifold learning [9]. However, our definition of geodesic distance incorporates local density scaling which does not appear in Tenenbaum’s work but is crucial for us to deal with outliers. In another work, Huo [10] has attempted to improve the performance of agglomerative clustering by projecting data points onto nearby smooth low-dimensional manifolds. However, when the data is not near any lower dimensional manifold, their approach would not provide any improvement.

The rest of this paper is organized as follows. Section 2 introduces some preliminaries, defines the geodesic class of distances, and outlines the geodesic K-medoids algorithm. Section 3 is devoted entirely to theoretical analysis of geodesic distance and geodesic K-medoids in particular. We present our experimental results in Section 4 comparing the performance of geodesic K-medoids to that of a few competitive spectral and hierarchical clustering methods on a number of clustering problems.

2 Algorithms

Consider a finite number of data points $X = \{x_1, x_2, \dots, x_n\}$ in \mathbb{R}^q , and suppose the number of clusters K is given. Here we define a class of geodesic distances and use it as a foundation for the algorithms that follow. In this paper, $\|\cdot\|$ always denotes Euclidean ℓ_2 norm.

2.1 Geodesic Distance

We first define two types of neighborhoods and their corresponding geodesic distances.¹

Definition. The ε -neighborhood of a point $x \in X$ is defined as $N_\varepsilon(x) = \{y \in X : \|x - y\| \leq \varepsilon\}$.

Definition. The k -neighborhood of a point $x \in X$ is the set of k closest points to x in the ℓ_2 norm sense: $N_k(x) \subseteq X$ such that $|N_k(x)| = k$ and $\max_{y \in N_k(x)} \|x - y\| \leq \min_{y \in X \setminus N_k(x)} \|x - y\|$.

Definition. The k -neighborhood graph of X is an undirected graph $G_k = (V, E)$ with $V = X$ and $(x_i, x_j) \in E$ iff $x_i \in N_k(x_j)$ or $x_j \in N_k(x_i)$. Similarly, the ε -neighborhood graph G_ε is defined with $N_\varepsilon(x)$.

Definition. Assume a symmetric matrix $W \in \mathbb{R}^{n \times n}$ of non-negative weights on the edges of G_k . The k -geodesic distance from any point x to any point y is defined as the total weight of the shortest weighted path from x to y on graph G_k , and is denoted by $d_{G_k, W}(x, y)$ (set to ∞ if there is no such path). Similarly, the ε -geodesic distance $d_{G_\varepsilon, W}$ is defined over the ε -neighborhood graph G_ε .

¹Readers familiar with manifold theory may notice that we are using the term “geodesic distance” for the distance approximated from data. This is to be consistent with the terminology of [9].

It is straightforward to show the following.

Theorem 2.1. *If G is an ε or k -neighborhood graph over X weighted by a symmetric matrix W of non-negative weights, then $d_{G,W}$ defines a distance on X .*

In practice, geodesic distance can be efficiently calculated by applying Dijkstra’s shortest path algorithm on graph G and weights W . The neighborhood graph G is constructed directly from X as defined above. However, the choice of W gives us a degree of freedom to define different classes of geodesic distances for different applications. The geodesic distance of Tenenbaum et al. [9] is equivalent to taking the weights W as the Euclidean distances between the neighboring points. This works well for their problem of interest—finding manifold embeddings. Here we are concerned with robustly separating different clusters in a data set in the presence of outliers. In most real world applications, it is usually safe to assume that the density of outliers is significantly lower than that of real data points. Therefore we define W in a way that captures the local density in addition to Euclidean distances.

Given a density estimator \hat{f} , we define the *density scaling weights* as

$$W_{ij} = \exp(1/2\sigma^2 \max\{\hat{f}(x_i), \hat{f}(x_j)\})\|x_i - x_j\|, \quad (1)$$

for all $(x_i, x_j) \in E$ with $G = (V, E)$ being the original neighborhood graph (either G_k or G_ε). The maximization in the above expression is to ensure weight symmetry required by Theorem 2.1. Intuitively, this choice of weights on the edges of our original neighborhood graph would introduce an exponential increase in the total weight of paths that pass through low density regions of the data set. Therefore, the intra-cluster shortest paths would tend to pass through the corresponding cluster, while the inter-cluster shortest paths would have much larger total weights compared to intra-cluster shortest paths. We present a more formal analysis of these effects in Section 3.

Among many methods proposed for nonparametric density estimation [11], we are especially interested in the ones that are based on neighborhood graphs. Such methods fit naturally into our framework in this paper. First we introduce some notation for the purpose of local density estimation. For some integer k_d , suppose we have the k_d -neighborhood graph G_{k_d} over X . For any $x_i \in X$, let $R(N_{k_d}(x_i))$ denote the distance from x_i to its farthest neighbor in its neighborhood $N_{k_d}(x_i)$, i.e., $R(N_{k_d}(x_i)) = \max_{x_j \in N_{k_d}(x_i)} \|x_i - x_j\|$. Moreover, let $Vol(r)$ represent the volume (Lebesgue measure) of a Euclidean ball of radius r in \mathbb{R}^q , i.e., $Vol(r) = 2r^q \pi^{q/2} / q\Gamma(q/2)$, where Γ is the gamma function. Then the k -NN density estimator at any point $x_i \in X$ is given by

$$\hat{f}(x_i) = \frac{k_d - 1}{n \cdot Vol(R(N_{k_d}(x_i)))}. \quad (2)$$

The k -NN estimator provides pointwise convergence (in probability) if $k_d \rightarrow \infty$ and $k_d/n \rightarrow 0$ as $n \rightarrow \infty$ [12]. This is sufficient for most practical purposes. However, our theoretical analysis of Section 3 requires stronger convergence guarantees for the density estimator. Therefore we also consider a special case of variable kernel estimator with a rectangular kernel, given by

$$\hat{f}(x_i) = \frac{1}{n} \sum_{\{x_j | x_i \in N_{k_d}(x_j)\}} \frac{1}{Vol(R(N_{k_d}(x_j)))}. \quad (3)$$

This estimator converges to the true density function almost surely if $k_d/n \rightarrow 0$ and $k_d/\log n \rightarrow \infty$ as $n \rightarrow \infty$ [13]. We will make use of this strong convergence property in the analysis of Section 3.

2.2 Geodesic K-medoids Clustering

1. Construct the (ε - or k -) neighborhood graph $G = (V, E)$ over the set of data points X .
2. Set the weights $W_{ij} = \exp(1/2\sigma^2 \max\{\hat{f}(x_i), \hat{f}(x_j)\})\|x_i - x_j\|$ for all $(x_i, x_j) \in E$, in which $\hat{f}(x_i)$ is a estimated density at point x_i .
3. Form the matrix of geodesic distances $D \in \mathbb{R}^{n \times n}$ by $D_{ij} = d_{G,W}(x_i, x_j)$.
4. Apply K-medoids to the distance matrix D to find K clusters.

In the above algorithm σ is a parameter that controls the importance of density in the clustering process. For example in the ideal case where no outliers exist in the data points large values of σ

work well. As we will demonstrate in the simulations the performance of our algorithm is not very sensitive on the value of σ and it finds correct clusters for a range of values of σ .

The K-medoids algorithm is very similar to K-means except that, instead of using the means (centroids) of the clusters, it represents each cluster by its medoid point, i.e., the data point that is assigned to that cluster and has the minimum sum of distances to the rest of the points in the same cluster. Therefore, K-medoids itself suffers from drawbacks similar to those of K-means. For instance, it is unable to handle non-convex clusters as we demonstrate in Section 4. However, as we show in the theoretical analysis of Section 3, geodesic distance empowers the algorithm to capture the coherence of each cluster even in the lack of convexity. K-medoids is also known to be more robust in the presence of outliers [1].

The geodesic distance derived from the neighborhood graph G and this specific choice of weights W allows two data points to have long range connectivity if there exists a path between them that passes through regions of higher density relative to the density of surrounding regions. It is also worth noting that since the large geodesic distance between two points means that they are in different clusters, this helps us to use informed initialization to make the algorithm converge faster and in fewer iterations. We propose one such informed initialization algorithm here. Choose the first medoid at random from the data points. Find p (usually around 5 percent of the cardinality of the data) points that have the maximum distance from the first chosen medoid and put them in set M . Randomly select one of the points in M as the second medoid. To choose the r th medoid, find p points that have the maximum sum of distances from the $r - 1$ previously chosen medoids and put them in M . Select one of the points in M at random as the r th medoid.

Although we have used the K-medoids algorithm to show the efficiency of the geodesic distance we have introduced, the class of geodesic distances can be applied to any distance based (or similarity based) clustering algorithm such as hierarchical and spectral clustering methods.

3 Theoretical Analysis

Here we present a theoretical analysis of the properties of the geodesic distance introduced in the previous section. Moreover, we investigate the behavior of the geodesic K-medoids algorithm in the asymptotic regime where the number of data points is very large. To that end, we first formulate a theoretical framework to model the clustering task. Then we employ that model to derive conditions under which geodesic K-medoids clustering would succeed with high probability as the number of data points grows. To prove the latter, we will show that under certain conditions, the geodesic distance between points from different clusters is larger than the geodesic distance between any two points in the same cluster.

3.1 Framework

Suppose that the data points X are samples drawn from a true distribution f over \mathbb{R}^q . We define a theoretical notion of cluster regions and outlier region as follows.

Definition. The *outlier region* is defined as the set of points $\tilde{O} \in \mathbb{R}^q$ such that $f(x) \leq f_{max}^o$ for all $x \in \tilde{O}$ and for some positive f_{max}^o . Each *cluster region* \tilde{C} is then defined as one of the connected components of \mathbb{R}^q / \tilde{O} which we assume to be open and totally bounded. We also assume that between any two points in \tilde{C} , there exists some path of finite length in the same cluster region.

Definition. The *minimum inter-cluster distance* for a set of cluster regions $\tilde{C}_1, \tilde{C}_2, \dots, \tilde{C}_K$ is defined as:

$$\ell_{ic} = \min_{i < j} \inf_{x \in \tilde{C}_i, y \in \tilde{C}_j} \|x - y\|_2. \quad (4)$$

Definition. The *geodesic diameter* of a cluster region \tilde{C} is defined as the supremum length of all the shortest paths between every pair of points in \tilde{C} . More precisely,

$$\rho(\tilde{C}) = \sup_{x, y \in \tilde{C}} \inf_{\substack{\gamma: [0,1] \rightarrow \tilde{C} \\ \gamma(0)=x, \gamma(1)=y}} L(\gamma). \quad (5)$$

Here, $L(\gamma)$ denotes the length of a C^1 -smooth (differentiable) path $\gamma(t) : [0, 1] \rightarrow \mathbb{R}^q$ as

$$L(\gamma) = \int_0^1 \|\dot{\gamma}(t)\| dt .$$

For a piecewise smooth path, its length is defined as the sum of the lengths of its smooth pieces.

3.2 Analysis

We first derive bounds on intra-cluster and inter-cluster geodesic distances in two lemmas that follow. Then we show that the conditions required by these lemmas hold under simple assumptions on the density estimation \hat{f} . In the end, we conclude with conditions under which the correct clustering assignment is a local optimum of geodesic K-medoid. For the analysis presented here, we mainly consider the case of ε -neighborhood graph and briefly discuss the case of k -neighborhood.

Lemma 3.1. *Consider a cluster region \tilde{C} and assume that for any point $y \in \tilde{C}$, there exist a data point $m \in \tilde{C} \cap X$ such that $\|y - m\| \leq \delta \leq \varepsilon/3$. Also suppose that $\exp(1/2\sigma^2 \hat{f}(x_i)) \leq \exp(1/2\sigma^2 f(x_i)) + \gamma$ for any $x_i \in X$ and some positive γ , where $\hat{f}(x_i)$ is an estimate of f at x_i . Then, the ε -geodesic distance between any two data points in $\tilde{C} \cap X$ is bounded above by $3(\exp(1/2\sigma^2 f_{min}^{\tilde{C}}) + \gamma)\rho(\tilde{C})$, in which $f_{min}^{\tilde{C}} = \min_{x_i \in \tilde{C} \cap X} f(x_i)$.*

Proof. Take two arbitrary data points x and z in $\tilde{C} \cap X$ and suppose that there is a path of finite length ℓ between them in \tilde{C} . Divide the path into $s = \lceil \ell/\delta \rceil$ segments, such that each segment has a length less than or equal to δ . Denote the endpoints of these path segments by y_1, y_2, \dots, y_{s-1} . According to our hypothesis, for each y_i there exist an $m_i \in \tilde{C} \cap X$ such that $\|y_i - m_i\| \leq \delta$. We also know that $\|y_i - y_{i+1}\| \leq \delta$. Therefore, by triangle inequality, the distance between m_i and m_{i+1} is less than $\|y_i - m_i\| + \|y_i - y_{i+1}\| + \|y_{i+1} - m_{i+1}\| \leq 3\delta \leq \varepsilon$ and the neighborhood graph G_ε will have an edge between these two data points. Thus, there exists a path through $x, m_1, \dots, m_{s-1}, z$ on G_ε which gives an upper bound on the discrete geodesic distance between x and z :

$$\begin{aligned} d_{G,W}^\varepsilon(x, z) &\leq \sum \|m_{i+1} - m_i\| \exp(1/2\sigma^2 \max\{\hat{f}(m_i), \hat{f}(m_{i+1})\}) \\ &= \sum \|m_{i+1} - y_{i+1} + y_{i+1} - y_i + y_i - m_i\| \exp(1/2\sigma^2 \max\{\hat{f}(m_i), \hat{f}(m_{i+1})\}) \\ &\leq \sum (\|m_{i+1} - y_{i+1}\| + \|y_{i+1} - y_i\| + \|y_i - m_i\|) (\exp(1/2\sigma^2 \max\{\hat{f}(m_i), \hat{f}(m_{i+1})\}) + \gamma) \\ &\leq 3\ell(\exp(1/2\sigma^2 f_{min}^{\tilde{C}}) + \gamma) . \end{aligned} \tag{6}$$

Since $\ell \leq \rho(\tilde{C})$ according to the definition, the lemma is established. \square

Lemma 3.2. *Suppose that our density estimation has the property $\exp(1/2\sigma^2 \hat{f}(x_i)) \geq \exp(1/2\sigma^2 f(x_i)) - \gamma$ for any $x_i \in X$ and for some positive γ . Then, the ε -geodesic distance between any two data points from different cluster regions is bounded below by $(\exp(1/2\sigma^2 f_{max}^o) - \gamma)\ell_{ic}$, in which $f_{max}^o = \sup_{x \in \tilde{O} \cap X} f(x)$.*

Proof. Take any two data points x and z from two different clusters. Assume that there exist a path through m_1, m_2, \dots, m_s between these two points. Without loss of generality, let m_1, m_2, \dots, m_r lie in the outlier region and m_{r+1}, \dots, m_s in cluster regions. Thus we have:

$$\begin{aligned} d_{G,W}^\varepsilon(x, z) &\geq \sum_{i=1}^s \|m_{i+1} - m_i\| \exp(1/2\sigma^2 \max\{\hat{f}(m_i), \hat{f}(m_{i+1})\}) \geq \\ &\quad \sum_{i=1}^r \|m_{i+1} - m_i\| (\exp(1/2\sigma^2 f_{max}^o) - \gamma) \geq (\exp(1/2\sigma^2 f_{max}^o) - \gamma)\ell_{ic} . \end{aligned}$$

\square

Now the following theorem facilitates the conditions needed in the previous two lemmas through the almost surely convergence of density estimation to the true distribution.

Theorem 3.3. *For any cluster region \tilde{C} assume that $\inf_{x \in \tilde{C}} f(x) > 0$. If $\hat{f}(x_i) \rightarrow f(x_i)$ almost surely, then the conditions of Lemma 3.1 and Lemma 3.2 hold with probability one as $n \rightarrow \infty$.*

Proof. Let $\inf_{x \in \tilde{C}} f(x) = \beta \geq 0$. Consider any point x in \tilde{C} and let $B(x, \delta)$ represent a ball of radius δ centered at x . Finally, define A to be the event “ $B(x, \delta)$ includes some data point $m \in \tilde{C} \cap X$ ”, V to be the event “ $\exp(1/2\sigma^2 \hat{f}(x_i)) \leq \exp(1/2\sigma^2 f(x_i)) + \gamma$ for all $x_i \in X$ ”, and U to be the event “ $\exp(1/2\sigma^2 \hat{f}(x_i)) \geq \exp(1/2\sigma^2 f(x_i)) - \gamma$ for all $x_i \in X$.” We have,

$$\begin{aligned} P(A \cap V \cap U) &= 1 - P(A^c \cup V^c \cup U^c) \geq 1 - P(A^c) - P(V^c) - P(U^c) \\ &\geq 1 - (1 - \beta \text{Vol}(B(x, \delta) \cap \tilde{C}))^n - P(V^c) - P(U^c) \xrightarrow{n \rightarrow \infty} 1, \end{aligned}$$

in which n is the total number of points and $\text{Vol}(S)$ represents the volume (Lebesgue measure) of $S \subset \mathbb{R}^q$. Notice that $\text{Vol}(B(x, \delta) \cap \tilde{C}) > 0$ since \tilde{C} is an open set. Furthermore, $P(V^c)$ and $P(U^c)$ shrink to zero because Theorem 3.4 implies that $\exp(1/2\sigma^2 \hat{f}(x_i))$ converges to $\exp(1/2\sigma^2 f(x_i))$ almost surely. \square

Theorem 3.4. *if $f_n(x) \xrightarrow{n \rightarrow \infty} f(x)$ almost surely or in probability, and g is a continuous function, then $g(f_n(x)) \xrightarrow{n \rightarrow \infty} g(f(x))$ almost surely or in probability respectively. In other words, a continuous mapping preserves the convergence in probability and almost surely.*

Proof. Refer to [14] for the proof. \square

Notice that Theorem 3.3 does not hold when $\hat{f}(x) \rightarrow f(x)$ only in probability. In that case, it is straightforward to show that the bounds of lemmas 3.2 and 3.1 are still valid if f is replaced with \hat{f} .

Finally, we put everything together in one theorem on the geodesic K-medoids algorithm.

Theorem 3.5. *Suppose $\hat{f}(x_i) \rightarrow f(x_i)$ almost surely, $\inf_{x \in \tilde{C}} f(x) > 0$, and $(\exp(1/2\sigma^2 f_{max}^o) - \gamma)\ell_{ic} > 3(\exp(1/2\sigma^2 f_{min}^{\tilde{C}}) + \gamma)\rho(\tilde{C})$ for every $\tilde{C} \in \{\tilde{C}_1, \dots, \tilde{C}_K\}$ and some $\gamma > 0$, then the correct clustering of X is a local optimum of the geodesic K-medoids algorithm with ε -geodesic distance.*

Proof. Let $m_i \in \tilde{C}_i \cap X$ be the medoid of the i th cluster in some correct clustering assignment $\zeta : X \rightarrow \{1, 2, \dots, K\}$. For any data point $x \in \tilde{C}_i \cap X$, lemmas 3.2 and 3.1 allow us to write

$$d_{G,W}^\varepsilon(x, m_i) \leq 3(\exp(1/2\sigma^2 f_{min}^{\tilde{C}_i}) + \gamma)\rho(\tilde{C}_i) < (\exp(1/2\sigma^2 f_{max}^o) - \gamma)\ell_{ic} \leq d_{G,W}^\varepsilon(x, m_j),$$

for any $j \neq i$. Therefore, ζ is a stable local optimum for geodesic K-medoids. \square

The case of the k -geodesic distance is more involved to analyze, we only briefly sketch the intuition behind the proofs. Here we only considers uniform distribution in each cluster. The results can be extended to the case of piecewise constant densities in each cluster, and then to general distributions. Consider a cluster region \tilde{C} and let $B(x, r) \subset \tilde{C}$ be a ball of radius r centered at $x \in \tilde{C}$. The average number of data points in this ball would be $k/n \sim \alpha r^q / \text{Vol}(\tilde{C})$. Therefore, if we take $k/n \text{Vol}(\tilde{C}) = \gamma^q$, the average radius would be equal to γ . As n goes to infinity, the probability that ε is different from its mean decays exponentially and so this problem reduces to the case of ε -neighborhood and the same conclusions apply.

4 Experiments

In the previous section we showed theoretically that in the asymptotic regime where the number of points goes to infinity the geodesic k-medoids is able to detect the clusters correctly. The goal of this section is to investigate the performance of the geodesic k-medoids on a number of data sets through simulations and to compare its performance with a number of other algorithms. These include K-medoids, single linkage clustering and the normalized spectral clustering of Ng et al. [6] with different types of similarity graphs, namely k -nearest neighbors, and fully connected with Gaussian similarity function [7]. The datasets are constructed such that they reveal some of the major limitations of the spectral clustering algorithms.

In order to find the best value of σ for the variants of the normalized spectral clustering algorithms, we search over a range of values for the optimum value that minimizes a cost function. In our experiments we considered both Ncut [7] and K-means clustering distortion (within cluster weighted

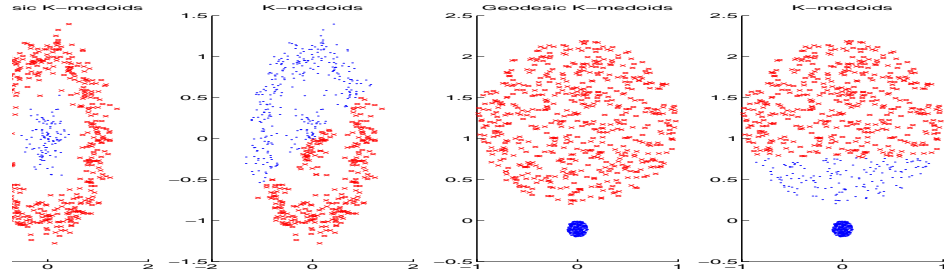


Figure 1: Some common problems of K-medoids and the performance of the geodesic k-medoids on these datasets. $k = 5$ and $\sigma = .15$ are chosen for the geodesic k-medoids in both cases.

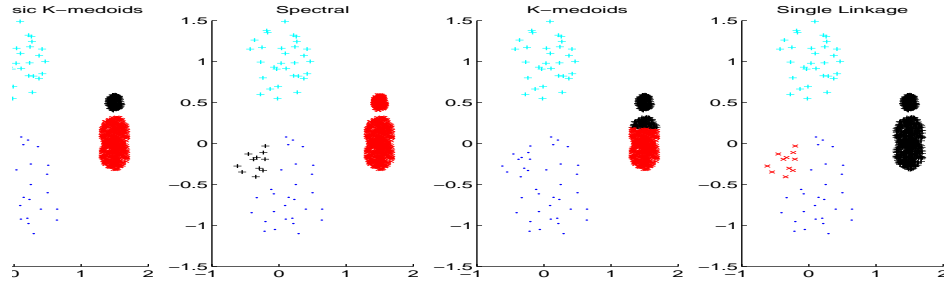


Figure 2: (a) Geodesic K-medoids algorithm with k -geodesic distance, $k=6$, and $\sigma = 0.15$, (b) K-medoids, (c) Single Linkage (d) Normalized spectral clustering of Ng, et al. with Gaussian similarity matrix, $\sigma = .03$.

mean square error [1]) as the cost function. Finally after choosing the best value of the parameter by these automated methods, we run the algorithm on a number of values of the tuning constant and compare the performance visually with the automated method and choose the best. In order to avoid local minima, we run 100 iterations of each algorithm for each value of their tuning parameter and choose the best automatically. We do the same thing on the geodesic K-medoids algorithm, although it has been seen that the performance of this algorithm is less sensitive to the values of the parameters and we usually get the correct answer for a range of values. In order to initialize the geodesic K-medoids algorithm, we use the method explained in 2.2. In all these simulations k -nn is used for the density estimation with the same k as the neighborhood graph.

Figure (1) shows two different datasets that reveal some of the basic problems of the K-medoids (or K-means) algorithm. The first two data sets are not linearly separable and therefore K-medoids cannot distinguish the clusters. In the second dataset since the diameter of one cluster is larger than half of the distance between the medoids of the two clusters, K-medoids cannot distinguish the clusters correctly. In both cases geodesic K-medoids can discriminate the two clusters.

Figure (2) shows the best performance of different algorithms on a data set of four clusters. In this data set, two of the clusters have very high density and are located close to each other while the other two have much lower densities. The problem of K-medoids is not surprising and was explained in Figure 1. Single linkage clustering cannot perform well here either, since it uses Euclidean distances

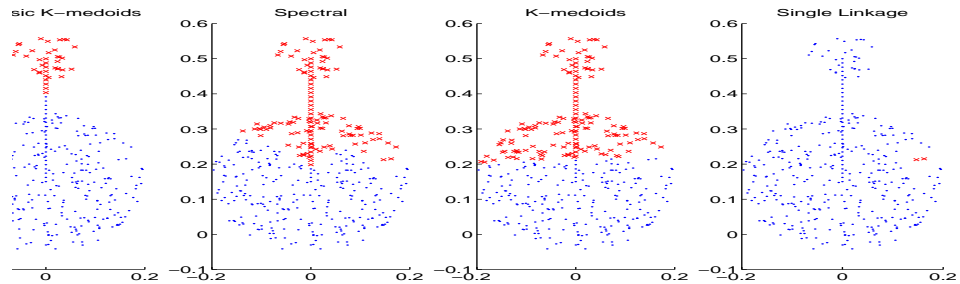


Figure 3: (a) Geodesic K-medoids $k = 10$, $\sigma = .01$, (b) K-medoids, (c) single linkage, (d) spectral clustering with k -nearest neighbor similarity matrix, $k = 10$ and $\sigma = 0.03$

to connect the points and this will lead to serious mistakes as you may see. In the case of Gaussian similarity matrix, spectral clustering requires larger values of σ in order to find the low density clusters correctly. On the other hand, large values of σ prevent the algorithm from distinguishing between the clusters that are close to each other. As mentioned before our algorithm is not very sensitive to the value of the parameters. For example, by changing k from 5 to 15 the resulting clusters do not change in (a). Also, in the cases where the clusters are well separated as in this data set, the value of σ does not play much role. We also noticed that spectral clustering with k-nearest neighbor performs very well in this case (not shown in the figure). This is due to the fact that for certain values of k , the similarities between the points from different clusters vanish. In the next experiment, however, we show the situation where k-nearest neighbor does not perform so well.

Now we turn our attention to a different situation in which the clusters vary significantly in size and are not very well separated. Figure 3 presents the results for different clustering algorithms on such a dataset. Again, K-medoids and single linkage suffer from their usual limitations. Spectral clustering algorithms show a different type of error in these figures. Since they try to balance the volume of different clusters [7], when two clusters in close proximity have very different sizes, spectral algorithms tend to assign some of the points in the larger cluster to the smaller one. This phenomenon is observed to different degrees in all of the clusterings produced by spectral algorithms.

5 Concluding Remarks

Toward a better class of clustering algorithms, we proposed a class of geodesic distances as the basis. These geodesic distances can be used with any clustering method that works with distances or similarities of the points rather than their actual Euclidean coordinates. We used the simplest instance of such algorithms, K-medoids and introduced the geodesic K-medoids. The superior performance of this algorithm on a number of difficult clustering problems is an encouraging indication that our approach has significant potential toward building better performing algorithms. The theoretical analysis of the geodesic K-medoids algorithm also indicates that in the asymptotic regime where the number of data points is very large the algorithm will find clusters with very high probability.

References

- [1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of the Statistical Learning*. Springer, 2001.
- [2] T. Sorlie, C. M. Perou, R. Tibshirani, T. Aas, and S. Geisler, "Gene expression pattern of breast carcinomas distinguish tumor subclass with clinical implications," *Proceedings of National Academy of Science*, vol. 98, pp. 10869–10874, 2001.
- [3] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE Trans. Neural Networks*, vol. 16, pp. 645–678, 2005.
- [4] S. Guha, R. Rastogi, and K. Shim, "Cure: An efficient clustering algorithms for large databases," *Proceedings of the ACM SIGMOD Conference on Management of Data*, pp. 73–84, 1998.
- [5] T. Zhang, R. Ramakrishnan, and M. Livny, "Birch: an efficient data clustering method for very large databases," *Proceedings of the ACM SIGMOD Conference on Management of Data*, vol. 25, no. 2, pp. 103–114, 1996.
- [6] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering analysis and an algorithm," pp. 849–856, NIPS, MIT Press, 2002.
- [7] U. V. Luxburg, "A tutorial on spectral clustering," *Technical Report*, pp. 1–30, 2006.
- [8] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 888–905, 2000.
- [9] J. B. Tenenbaum, V. D. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, 2000.
- [10] X. Hue, "A local smoothing and geodesic distance based clustering algorithm for high dimensional noisy data; utilizing embedded geometric structures," *Workshop on Clustering High Dimensional Data and its Applications, in conjunction with SIAM International Conference on Data Mining*, 2003.
- [11] A. J. Izenman, "Recent developments in nonparametric density estimation," *Journal of the American Statistical Association*, vol. 86, no. 413, pp. 205–224, 1991.
- [12] D. O. Loftsgaarden and C. P. Quesenberry, "A nonparametric estimate of multivariate density function," *Annals of Mathematical Statistics*, vol. 36, pp. 1049–1051, 1965.
- [13] L. Devroye and C. S. Penrod, "The strong uniform convergence of multivariate variable kernel estimates," *The Canadian Journal of Statistics*, vol. 14, no. 3, pp. 211–219, 1986.
- [14] S. R. S. Varadhan, *Probability Theory*. American Mathematical Society, 2001.