The World is not always Flat or Learning Curved Manifolds

Evgeni Begelfor ARISTO@CS.HUJI.AC.IL
School of Computer Science
Hebrew University of Jerusalem
Jerusalem 91904 Israel

Michael Werman WERMAN@CS.HUJI.AC.IL
School of Computer Science
Hebrew University of Jerusalem
Jerusalem 91904 Israel

Editor:

Abstract

Manifold learning and finding low-dimensional structure in data is an important task. Many algorithms for this purpose embed data in Euclidean space, an approach which is destined to fail on non-flat data. This paper presents a non-iterative algebraic method for embedding the data into hyperbolic and spherical spaces. We argue that these spaces are often better than Euclidean space in capturing the geometry of the data. The approach can be used to extend algorithms such as ISOMAP and SDE to the curved case. We also demonstrate the utility of these embeddings by showing how some of the standard clustering algorithms translate to these curved manifolds.

Keywords: Dimension reduction, embedding, hyperbolic spaces, PCA, clustering.

1. Introduction and motivation

Embedding data in low-dimensional space, also known as manifold learning is important for many applications such as visualization, data manipulation (smoothing / noise reduction etc.) and data exploration (clustering etc.). Most algorithms tackle the problem by searching for an embedding into a Euclidean space $\mathbb{R}^k$ such that k is as small as possible, while the data maintains some of its geometry (either locally or globally). The main advantage of the Euclidean space is its being simple and well understood. However, many times the inherent geometry of the data isn't flat, thus any embedding in $\mathbb{R}^k$ will necessarily have a big distortion. For example, embedding a regular tree with the shortest path metric in Euclidean space requires high dimension (Fig, 1). This is because the volume of a ball in $\mathbb{R}^k$ grows only polynomially, thus not having "enough space" for the tree. On the other hand, the volume in hyperbolic space grows exponentially, allowing a good embedding. For another example, points on a sphere, such as cities on Earth with the natural distances will be distorted as well by embedding in the Euclidean space, Figure 2.

This paper suggests an efficient non iterative algorithm for choosing the "right" space and embedding data points in it. We also provide a method of computing geodesic distances

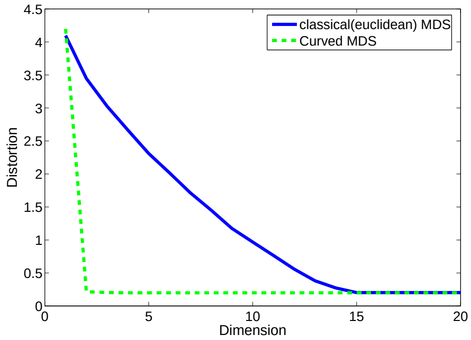

Figure 1: The distortion of embedding the nodes of a tree using the pairwise nodes path length as a metric versus the dimension of embedding space. Euclidean embedding requires much higher dimension than the curved one

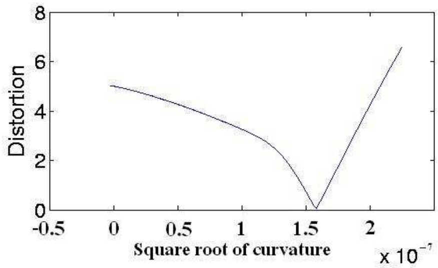

Figure 2: Embedding cities with their airline distances. The distortion of the embedding is shown versus the curvature of the host space. The best curvature is $\frac{1}{\sqrt{R}}$ where R is the radius of the Earth

and means in these spaces which are useful for data manipulation and understanding. Our algorithm can be used either for embedding data given its pairwise dissimilarity, or for dimensionality reduction through classical metric MDS.

A number of papers discussed the advantages of visualizing data using hyperbolic spaces (Walter, 2002; Lamping et al., 1995.; Munzner and Burchard, 1995). Our algorithm, described in section 4 can be used to embed the data for these visualizations.

Section 3 will review relevant properties of the hyperbolic and spheric spaces. Section 4 presents our algorithm to choose the "best" space and embed the data into it, based on a theorem of Blumenthal (1970), Menger and Schoenberg (1935). We will then show in section 5 how k-means and mean-shift clustering algorithms can be effectively implemented in H^n and S^n. The clustering is based on the possibility of computing geodesic distances and means in these spaces.

2. Previous work

Embedding data in low dimensional space has two variants. In dimensionality reduction or manifold learning we want to embed high dimensional data in a lower dimensional space while preserving some properties of the points geometry either globally or locally. In the second variant, called *Multidimensional Scaling (MDS)* (see Cox and Cox, 1995, for a general introduction), we only have the points pairwise dissimilarity measures, and our goal is to embed the data according to them.

There are numerous approaches to MDS and we will briefly mention a few. Classical metric MDS (sometimes referred to as PCA, principal component analysis) (Torgerson, 1952) tries to minimize the L_2 norm of the differences between the inter-point distances given by a dissimilarity matrix and those calculated between the embedded points. The main advantage of classical metric MDS is that it is non iterative and guaranteed to minimize the L_2 norm functional. Non classical metric MDS, for example, Sammon (1969) tries to minimize different functionals at the price of using iterative methods that are both computationally expensive and not guaranteed to converge to a global minima. Non-metric MDS (Shepard, 1962; Kruskal, 1964) tries to preserve the order of the distances instead of their values. Often the metric space data (pairwise distances) has some missing values. There are a few methods to "fill in" the missing values, such as using geodesic distances and SDE (Weinberger and Saul, 2004).

One of the methods for solving the dimensionality reduction problem is to reduce it to MDS by constructing the inter-point dissimilarity matrix. PCA can be looked upon as MDS using the global Euclidean distance. Isomap(Tenenbaum et al., 2000) tries to capture the geometry of the data by taking the "geodesic" distances in the data between points. Some non-metric methods for dimensionality reduction are LLE(Roweis and Saul, 2000), HLLE(Grimes and Donoho, 2003) and Laplacian eigenmaps(Belkin and Niyogi, 2002), which try to preserve relationships in the local neighborhoods of the data. All these methods work well if the data lies on or close to a manifold which is isometric to a subset of $\mathbb{R}^k$.

Learning curved manifolds has been discussed in (V. de Silva, 2003), although the paper addresses non-flatness arising from the sampling of the manifold, and not from its own geometry. Pless and Simon (2001) deal with embedding in spheres, but they do not touch upon hyperbolic embedding.

There have been papers related to embedding finite metric spaces into the hyperbolic plane (Shavitt and Tankel, 2004), especially for visualization (Lamping et al., 1995.; Walter, 2002) and into spheres (Falissard, 1996). This is a highly non convex problem and all solutions up to date were based on iterative methods that are prone to local minima. In section 4 we demonstrate by experiments that iterative methods fail.

3. Geometrical Background

There are 3 types of Riemannian spaces with constant (sectional) curvature; spherical (curvature > 0), Euclidean (curvature $= 0$) and hyperbolic (curvature < 0). Formal definition of curvature is beyond the scope of this paper, see Carmo (1992); M.Berger (2003) for more information. Vaguely speaking, sectional curvature is the generalization of the Gauss curvature for surfaces. We shall now lay out the models of these spaces that use in this paper. All the spaces are defined as subsets of R^n for some n with a distance, which doesn't come from the ambient space. We will sometimes denote the m-dimenstional space of constant curvature k by X_k^m, uniting both the positive, negative and zero curvature cases.

3.1 Spherical Geometry

We model the m-dimensional sphere with curvature $k > 0$ in the following way:

$$S_k^m = \left\{ x = (x_0, \ldots, x_m), \sum_{i=0}^{m} x_i^2 = 1 \right\}$$

with the distance between points $x, y \in S_k^m$:

$$d(x, y) = \frac{\arccos(< x, y >)}{\sqrt{k}} = \frac{\arccos(\sum_{i=0}^{m} x_i y_i)}{\sqrt{k}}$$

As the distance between two points in a sphere with curvature k is bounded above by $\frac{\pi}{\sqrt{k}}$ it would be meaningless to try to embed a given metric space in a sphere having too high a curvature.

3.2 Hyperbolic Geometry

There are different models of the hyperbolic space and in this paper we shall use the hyperboloid model. For visualization matters (Munzner and Burchard, 1995; Walter, 2002) the Klein or the Poincare disk models may be more appropriate. On $\mathbb{R}^{m+1}$ we define a bilinear form $\prec, \succ$:

$$\prec u, v \succ = u_0 v_0 - \sum_{i=1}^{m} u_i v_i$$

The hyperboloid consisting of all vectors $v \in \mathbb{R}^{m+1}$ such that $\prec v, v \succ = 1$ has two sheets. For $k < 0$ we take H_k^m to be the one containing points with positive v_0, with the distance:

$$d(u, v) = \frac{\arccos(\prec u, v \succ)}{\sqrt{k}} = \frac{\operatorname{arccosh}(\prec u, v \succ)}{\sqrt{-k}}$$

where $\cosh(x) = \frac{1}{2}(e^x + e^{-x})$ and arccosh is the inverse function of cosh.

For further information on hyperbolic geometry the reader is referred to Iversen (1992).

3.3 Euclidean Geometry

Euclidean space E^m of dimension has the well known definition

$$E^m = I\!R^m \ mboxwith \ d(x,y) = \sqrt{< x - y, x - y >}$$

The Euclidean space E^m can also be seen as the limit of the hyperbolic spaces H_k^m or the spherical spaces S_k^m as k tends to zero. This statement can be precise in the following way. For any two metric spaces (X,d) and (Y,ρ) and a mapping $f : X \to Y$ we define the distortion of the mapping f as

$$|f| = \sup_{x,x' \in X} \frac{\rho(f(x), f(x'))}{d(x,x')}$$

We define the similarity of the spaces X and Y as

$$s(X,Y) = \inf_{f:X \to Y \quad 1:1} |f||f^{-1}|$$

One sees that X and Y are isometric iff $s(X,Y) = 1$.

Proposition 1 *If $B(m,r,k)$ is the ball of radius r in the m-dimensional space with constance curvature k, then*

$$\lim_{k \to 0} s(B(m,R,k), B(m,r,0)) = 1$$

4. C-MDS: Curved Multidimensional Scaling

The goal of the well-known classical MDS is to embed a given metric space in $I\!R^n$ equipped with the Euclidean metric minimizing the average distortion of the distances. We shall describe a procedure generalizing MDS to non-flat spaces with constant curvature (spheres and hyperbolic spaces), which is very similar in spirit to the original MDS and also depends on a generalization of the PCA/SVD method. We need a theorem connecting the distances in a metric space X and its isometric embeddability in a space with constant curvature. The theorem goes back to Blumenthal (1970), Menger and Schoenberg (1935) and the proof reproduced here was taken from Kokkendorff (2004).

Let X be a finite space of size n with $d : X \times X \to I\!R^+$ a metric. Denote by D the distance matrix with entries $d_{ij} = d(x_i, x_j)$. For $k \neq 0$ define an $n \times n$ matrix $C^{X,k}$ as follows

$$C_{ij}^{X,k} = \cos(\sqrt{k} d_{ij})$$

Theorem 1 *Let $X, d, k, C^{X,k}$ as above.*
$k < 0$. The metric space X can be isometrically embedded in H_k^m iff the signature of $C^{X,k}$ is $(1, q, n-q-1)$ with $q \leq m$.
$k > 0$. The metric space X can be isometrically embedded in S_k^m iff the signature of $C^{X,k}$ is $(q, 0, n-q)$ with $q \leq m+1$.

Signature (a, b, c) means that a matrix has a positive eigenvalues, b negative eigenvalues and c zero eigenvalues.

Proof The hyperbolic case, $k < 0$. $\Rightarrow$ Let $\{y^i\}_{i=1}^n$ be points in H_k^m. We look at the hyperboloid model of H_k^m, thus y^i is a vector in $\mathbb{R}^{m+1}$. As defined above, the distances between y^i are

$$d(y^i, y^j) = \frac{\operatorname{arccos}(\prec y^i, y^j \succ)}{\sqrt{k}}$$

thus $C_{ij}^{X,k} = \prec y^i, y^j \succ$ In other words, if we write y^i as rows of matrix A and define an $(m+1) \times (m+1)$ matrix $J = \begin{pmatrix} 1 & & & \\ & -1 & & \\ & & \cdots & \\ & & & -1 \end{pmatrix}$ then one has $C^{X,k} = AJA^T$ and the condition on the signature follows. If A is of full column rank, then the signature of $C^{X,k}$ is $(1, m, n-m-1)$. Linear dependence of the columns of A will change the signature of A to $(1, q, n-q-1)$ for $q < m$).

In the other direction, if $C^{X,k}$ has signature $(1, q, n-q-1)$ then there exists a $n \times m+1$ matrix A such that $C^{X,k} = AJA^T$. Now the rows of A will be points in H_k^m having the right distances between them.

The spheric case, $k > 0$. $\Rightarrow$ Let $\{y^i\}_{i=1}^n$ be points in S_k^m, thus every y^i is a vector in $\mathbb{R}^{m+1}$. As defined above, the distances between y^i are

$$d(y^i, y^j) = \frac{arccos(< y^i, y^j >)}{\sqrt{k}}$$

thus $C_{ij}^{X,k} = < y^i, y^j >$ In other words, if we write y^i as rows of matrix A then one has $C^{X,k} = AA^T$ and the condition on the signature follows. If A is of full column rank, then the signature of $C^{X,k}$ is $(m+1, 0, n-m-1)$. Linear dependence of the columns of A will change the signature of A to $(q, 0, n-q)$ for $q \leq m+1$).

In the other direction, if $C^{X,k}$ has signature $(q, 0, n-q)$ with $q \leq m+1$) then there exists a $n \times m+1$ matrix A such that $C^{X,k} = AA^T$. Now the rows of A will be points in S_k^m having the right distances between them.

$\blacksquare$

What if the metric space (X, d) cannot be embedded exactly in H_k^m or S_k^m? We would like to obtain the best solution in some sense. Recall the Frobenius matrix norm:

$$||A|| = \sqrt{\operatorname{tr}(AA^T)}$$

In the classic MDS we look for $v_1 \ldots, v_n \in \mathbb{R}^m$ minimizing the frobenius distance (norm of the difference) between the original squared distance matrix $D._{ij}^2 = d_{i,j}^2$ and the squared distance matrix $D'._{ij}^2 = \|v_i - v_j\|^2$. In the curved case we look for $u_1 \ldots, u_n$ in X_k^m minimizing the frobenius distance between the matrix $C^{X,k}$ as defined above and $C'_{ij} = \cos(\sqrt{k}d(u_i, u_j))$. Plainly put, we approximate $\cos(\sqrt{k}d_{ij})$ instead of d_{ij}^2 itself. The reason we choose this distortion is the algorithm described below.

For k going to zero the results coincide with MDS, the proof follows. When the curvature k tends to zero we have $C^{X,k} = \cos(\sqrt{k}d_{ij}) \approx 1 - \frac{kd_{ij}^2}{2}$. For any k we get the embedding inside X_k^m minimizing the frobenius distance between C and C', but from proposition 1 we know that for k tending to zero the space X_k^m goes to $I\!R^m$. Thus in the limit we get the embedding in $I\!R^m$ minimizing the frobenius distance between $1 - \frac{kd_{ij}^2}{2}$ and $1 - \frac{kd(u_i,u_j)^2}{2}$ which is the MDS embedding.

As in classic MDS we use spectral decomposition to embed the space with minimal distortion, taking the dominant eigenvalues. The algorithms for the spherical and hyperbolic cases are outlined below. The bottleneck of the running time is the eigenvalue decomposition. Fortunately, we do not have to compute the entire decomposition, but only the leading $m+1$ eigenvectors. Thus the running time for computing the embedding in m-dimensional space is $O(|X|^2 m)$, which makes the algorithm realistic for very large data sets.

Input: $D, m, k < 0$
Output $\{v^i\}_{i=1}^n \in H_k^n$

$$C_{ij} = \cos(\sqrt{k}d_{ij})$$

$C = UEU^T$ eigenvalue decomposition, eigenvalues in **increasing order**
$U = [U_m U_{n-m}]$
E_m consists of the m most negative eigenvalues

$$A = U_m\sqrt{-E_m} = \begin{pmatrix} w_1 \\ \dots \\ w_n \end{pmatrix}$$

$$v^i = \frac{w_i}{\sqrt{1 - \|w^i\|^2}}$$

Input: $D, m, k > 0$
Output $\{v^i\}_{i=1}^n \in S_k^n$

$$C_{ij} = \cos(\sqrt{k}d_i)$$

$C = UEU^T$ eigenvalue decomposition
$U = [U_{m+1} U_{n-m-1}]$

$$E = \begin{pmatrix} E_{m+1} & \\ & E_{n-m-1} \end{pmatrix}$$

$$A = U\sqrt{E} = \begin{pmatrix} w_1 \\ \dots \\ w_n \end{pmatrix}$$

$$v^i = \frac{w^i}{\|w^i\|}$$

To show that curved manifolds appear naturally in applications, we've built the distortion versus curvature graph for different datasets, Fig.[3,4,5] coming from diverse areas. The zero curvature corresponds to the Euclidean case, the result obtained with classical PCA (MDS). The distortion of the embedding ϕ of X, d into Y, ρ is defined to be

$$dis(\phi) = \frac{1}{|X|}\sqrt{\sum_{ij}(d(x_i, x_j) - \rho(\phi(x_i), \phi(x_j)))^2}$$

It is clearly seen in these examples that the datasets do not have an Euclidean structure, and thus call for learning curved manifolds.

The staight-forward iterative solution to the problem of finding the best embedding problem is bound to get stuck because of the abundance of the local minima. To emphasize,

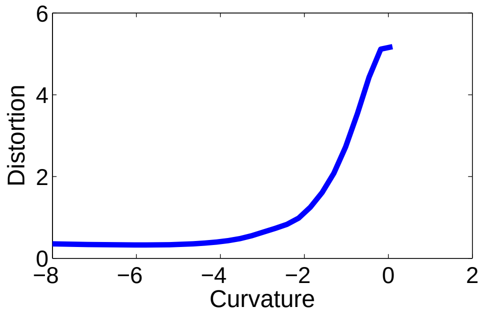

Figure 3: The distortion of embedding the internet in dimension two as a function of the curvature of the embedding space (negative values are hyperbolic spaces and positive are spherical). The distance between hubs was taken to be the minimal number of hops between them. The dataset contained 10000 hubs. Data from the DIMES project supplied by Scott Kirkpatrick.

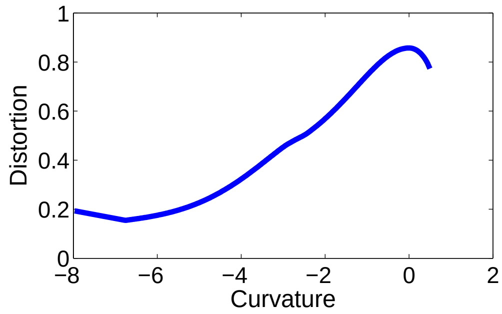

Figure 4: The distortion of the embedding a set of 4000 actors in dimension two as a function of the curvature of the embedding space. The distance between close actors is inversely proportional to the number of movies they jointly acted in, the further distances where computed by shortest paths. The data is taken from the UCI KDD archive

we've done the following experiment: We took random 100 points in the hyperbolic space, computed the distance matrix and then tried to find the embedding using iterative methods. Even in this case, when the distortion of the best embedding is zero, brute-force optimization results in embedding with high distortion.

4.1 Volume and expanders

So what makes data more 'hyperbolic' or 'spheric' ? One huristic answer is that while in the Euclidean space E^n the volume of a ball of radius r is $\Theta(r^n)$, the volume in the hyperbolic space grows exponentially. Specifically, the volume of a ball of radius r in $H^m_{-k^2}$ is M.Berger (2003)

$$V(r, m, -k^2) = C \int_0^r \sinh(kt)^{m-1} dt = \Theta(e^{(m-1)kr}) \text{for constant } k \text{and} m$$

So data which 'expands' fast would embedd better in the hyperbolic space. In graph theory, graphs with high expansion are called *expanders*, Hoory et al.. A graph $G = (V, E)$ is said to have h vertex expansion if for every set $S \subset$ with $|S| \leq |V|/2$ the number of neighbors of S in G is at least $h|S|$. It follows that in an h-expander up to some R, the volume (number of vertices) in a ball with radius r is at least $(1 + h)^r$, that is, exponentially growing. There are reasons to believe that natural graphs, such as aquaitance graph and the internet behave like expanders **?**.

We shall demonstrate that expanders embedd better in curved spaces by the following example. Pick a prime p. For every $0 < k < p$ we put edges $(k - 1, k)$, $(k, k + 1)$ and $(k, k^{-1} \; mod \; p)$. This graphs constitute a family of expanders Lubotzky et al. (1988). In

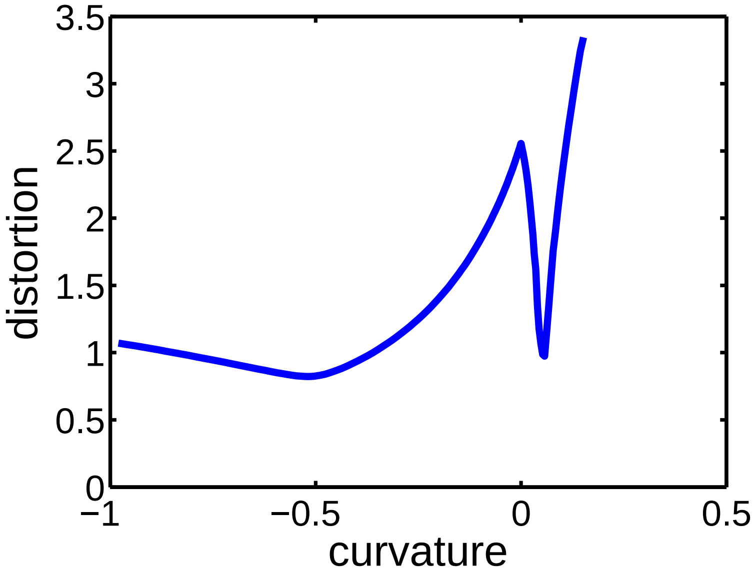

Figure 5: Embedding an expander: distortion vis curvature

Figure 4.1 we show the distortion of the embedding as a function of curvature for the graph otained with $p = 397$.

4.2 Choosing the curvature K

Although we do not have an analytical proof of the fact, in the experiments we made (Fig. 3,4,5) the variation of distortion with curvature is well behaved. This suggests using binary search to find the optimal curvature. Another approach, at least for the hyperbolic case, is to estimate the growth of the volume of a ball of radius r in the data as a function $f(r)$ of the radius. Knowing the formula for the growth of volume in H_k^m we can estimate the best curvature k.

4.3 Semi-definite Embedding

When doing manifold learning one wants to "trust" only the local information and use it to obtain the global structure of the manifold. The approach of *semi-definite embedding* Weinberger and Saul (2004) is to use the local distances to build the global distance matrix by semi-definite programming and then use classical MDS.

We have a partial distance matrix D, that is, d_{ij} is defined only for $(i, j) \in S$ for some set S.

Maximize $Tr(A)$ s.t. $A \succeq 0$ and $A_{ii} + A_{jj} - 2A_{ij} = D_{ij}^2$ for $(i, j) \in S$
Use MDS/PCA with the Gramm matrix A.

The same method can be used for spherical embeddings: we again replace the distance matrix D with $C^{X,k}$.

Maximize $\sum_{i,j} A_{ij}$ s.t. $A \succeq 0$ and $A_{ij} = \cos(\sqrt{k} D_{ij})$ for $(i, j) \in S$

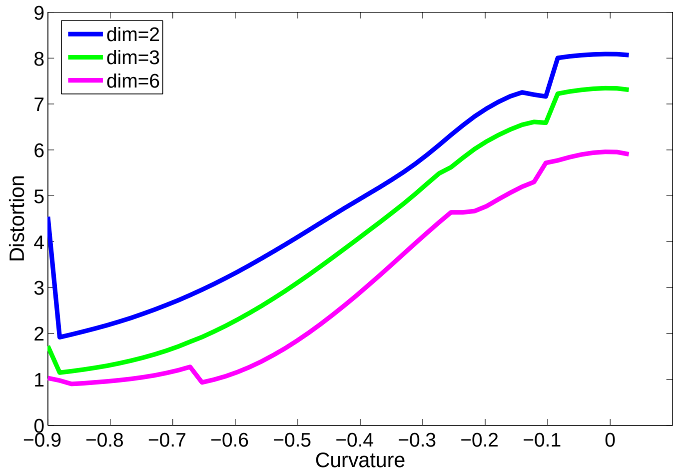

Figure 6: The distortion of the embedding of ORFs (potential genes) in the M. tuberculosis bacterium in dimension two (blue), three (green) and six (magenta) with respect to the curvature of the embedding space. The distances where computed by BLAST and the data taken from King et al. (2000).

Use C-MDS with the distance matrix $\frac{1}{\sqrt{k}} arccos(A)$.

Note that instead of maximazing the sum of the lengths of the vectors, we directly maximize the sum of all pairwise distances, as the vectors' length is constance in the spherical case. Direct usage of this algorithm for the hyperbolic embedding is impossible in the hyperbolic case the set of $C^{X,d}$ when X is a subset of the hyperbolic space with curvature k is not convex.

5. After the embedding

Embedding the data in a low dimensional Euclidean space is useful: there are many algorithms for handling data that work in the Euclidean space. We show that some of these algorithms can be transferred to the spherical/hyperbolic case. For example, linear separation of points in $\mathbb{R}^k$ by a hyperplane translates into separation by a geodesic manifold of codimension 1.

Clustering in Euclidean spaces is often done using k-means or mean-shift. In order to carry out these algorithms in the curved spaces S^m or in H^m for k-means or mean-shift we need to compute means and distances.

The notion of the mean of a set of points in a metric space Y can be defined in different ways. One of them, usually called the Karcher mean (Buser and Karcher, Asterisque 1981)[1] comes from noticing that the mean of a set of points in the Euclidean space $\mathbb{R}^k$ minimizes the sum of the squared distances to the points in the set. This is still well defined in any metric space, thus we set:

1. although originally defined and studied by Cartan

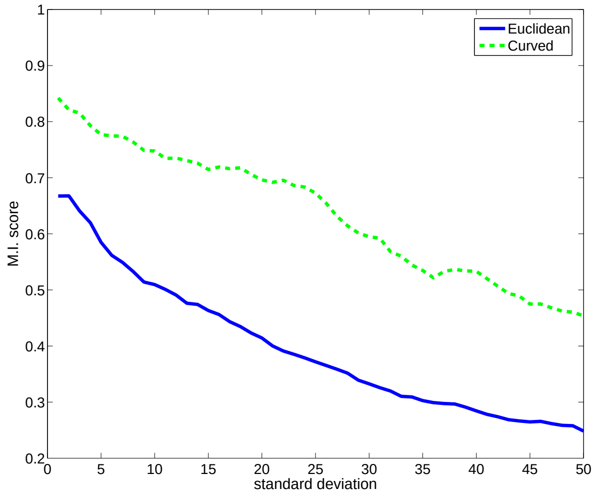

Figure 7: Comparison of clustering with k-means after embedding in euclidean (solid line) and hyperbolic (dashed line) plane.

$$Mean(y_1, \dots, y_n) = \arg\min_{x \in Y} \sum_{i=1}^{n} d(x, y_i)^2$$

The mean is not unique for a general set of points, consider two antipodal points on a sphere. The mean is unique for a set lying in an open half sphere [2]. It was shown by Cartan back in the 1920's that every set in a Riemannian manifold with non-positive curvature (such as hyperbolic space) has a unique mean. Summarizing: to compute the mean of a set of points in S^m or in H^m we minimize the sum of the squared distances. There is a unique local minima, thus any gradient decent type minimization will succeed. As we can analytically obtain the gradient and the Hessian of the function we minimize, the convergence to the minimum is quadratic.

The notion of Gaussian distributions can be generalized to Riemannian manifolds, in particular to spheres and hyperbolic spaces. Obtaining these, one can use EM for mixture of Gaussians. The computation of the covariance matrix of data is straightforward (Pennec, 1999), but estimating the parameters of the Gaussian distributions is more technical and will be published elsewhere.

In Figure 6 we show the results of a clustering experiment. We take a random set of points $\{c_i\}_{i=1}^k$ in the two-dimensional hyperbolic plane. A set of n points is drawn randomly with std σ around each of the c_i-s, giving us a subspace X consisting of nk points. The goal is to cluster X having only its distance matrix.We use a staight-forward implementation of k-means algorithm and compare the results of the clustering after embedding in $I\!R^2$ and H^2. To evaluate the clustering we used the mutual information score: if $f : X \to \{1, \dots, k\}$

2. If S lies in a convex set in a Riemannian manifold, the mean is unique, see M.Berger (2003)

is a clustering and $g : X \to \{1, \ldots, k\}$ is the ground truth, then $\text{score}(f) = \frac{1}{k} 2^{I(f;g)}$ where $I(f; g)$ is the mutual information between f and g.

5.1 Supervised Learning

Learning with labeled examples is classicaly done by separation of the given examples with hypersurfaces in E^m, or generalization of the idea through the kernel trick. The notion of hyperplanes and separation can be naturally extended to the non-zero curvature case as well, leading to curved version of the learning algorithm. Here we shall present the perceptron and its learning algorithm for the non-euclidean case.

We define the notion of a hyperplace in the particular case of spaces with constant curvature.

Definition 1 *A hyperplane H in a $X = X_k^m$ is an $m-1$ dimensional complete subspace of X such that for every two points y, z in$H$ the geodesic between x and y lies in H.*

It easy to see that for the zero curvature $X = E^m$ the definition above agrees with the usual notion of hyperplanes in the euclidean space. What happens in the other cases? On the sphere S_k^m one gets that any hyperplace is determined by a vector $v \in \mathbb{R}^{m+1}$ with

$$H_v = \{x \in S_k^m :< x, v >= 0\}$$

Thus the standard (euclidean) perceptron learning algorithm can be directly applied.

The hyperbolic case is more complicated, and a different model due to Klein of the hyperbolic space has to be introduced. The ambient space in this model is the m-dimensional open ball of radius 1.

$$D_k^m = \{x \in \mathbb{R}^m :< x, x >< 1\} \text{ with } d(x, y) = \arccos \frac{1- < x, y >}{\sqrt{k(1- < x, x >)(1- < y, y >)}}$$

an ismometry between the two models is given by the map ϕ:

$$\phi : D_k^m \to H_k^m \quad \phi(x) = \frac{(1, x)}{\sqrt{1- < x, x >}}$$

The surprising fact is that in the Klein model geodesics are straight lines and hyperplanes are intersections of hyperplanes in $\mathbb{R}^m$ with D_k^m. Thus, the standard perceptron learning algorithm can be applied without any change to obtain the hyperbolic separation.

In order to evaluate the advantage of the perceptron in curved space, we have done the following experiment: We took the Isolet data from the UCI repository and chose pairs of letters which are naturally hard to distinguish, for example 'n' and 'm'. The data was embedded into $\mathbb{R}^3$ and H^3. We used the given labels of a part of the data to learn the dividing hyperplane with the perceptron algorithm and used the rest as a testing set. The proportion of the rightly classified test samples is shown in Table 5.1.

An additional reason for the low-dimensional embedding is visualization. We chose not to add figures demonstrating advantages of hyperbolic visualization, as it is documented in literature: (Walter, 2002; Lamping et al., 1995.; Munzner and Burchard, 1995). The algorithm presented in this paper merely gives us a way to compute the embedding needed for the visualization.

Algorithm	n/m	d/t
Euclidean	0.65	0.72
Hyperbolic	0.76	0.81

Table 1: Using euclidean/hyperbolic perceptron to classify pairs of sounds from Isolet database

6. Discussion

In this work we proposed an algorithm for faithfully embedding data in low dimensional hyperbolic and spheric spaces. Our embedding algorithm is a generalization of the classical metric MDS and is non iterative and fast. As demonstrated by the examples the geometry of many datasets is better captured by these curved manifolds than by the Euclidean space. A lot of tools currently used for the Euclidean space can be transferred to the hyperbolic and the spherical case, thus enlarging the possibilities of data mining, understanding and visualizing complex datasets.

References

M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering, 2002.

L.M. Blumenthal. Theory and applications of distance geometry, 1970.

P. Buser and H. Karcher. Gromov's almost flat manifolds, Asterisque 1981.

M.P. Do Carmo. *Riemannian Geometry.* Birkuser, Boston, 1992.

T. F. Cox and M. A. A. Cox. *Multidimensional scaling.* London, 1995.

B. A. Falissard. A spherical representation of a correlation matrix. *Journal of Classification*, 13(2):267–280, 1996.

C. Grimes and D. Donoho. Hessian eigenmaps: New locally-linear embedding techniques for high-dimensional data. Technical report, Stanford University, 2003.

S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applications. Bull. AMS, to appear.

B. Iversen. *Hyperbolic Geometr.* Cambridge University Press, 1992.

R. King, A. Karwath, A. Clare, and L. Dehaspe. Accurate prediction of protein functional class in the m. tuberculosis and e. coli genomes using data mining. *Comparative and Functional Genomic*, 17, 2000.

S. L. Kokkendorff. Gram matrix analysis of finite distance spaces in constant curvature. *Discrete and Computational Geometry*, 31:515–543, 2004.

J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29:1–27, 1964.

J. Lamping, R. Rao, and P. Pirolli. A focus+context technique based on hyperbolic geometry for viewing large hierarchies. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 401–408, 1995.

A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988. ISSN 0209-9683.

M.Berger. *A Panoramic View of Riemannian Geometry*. Springer, Berlin, 2003.

T. Munzner and P. Burchard. Visualizing the structure of the world wide web in 3d hyperbolic space. In *Proceedings of VRML 1995*, pages 33–38, 1995.

X. Pennec. Probabilities and statistics on riemannian manifolds: Basic tools for geometric measurements. In *Proc. of Nonlinear Signal and Image Processing*, pages 194–198, 1999.

R. Pless and I. Simon. Embedding images in non-flat spaces. Technical report, Washington University, 2001.

S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

J. W. Sammon. A nonlinear mapping algorithm for data structure analysis. *IEEE Transactions on Computers*, 18(5:401–409, 1969.

I.J. Schoenberg. A remark to maurece frechets article. *Ann. of Math*, 36(3), 1935.

Y. Shavitt and T. Tankel. On the curvature of the internet and its usage for overlay construction and distance estimation. In *In Proceedings of IEEE INFOCOM 2004*, 2004.

R. N Shepard. The analysis of proximities: multidimensional scaling with an unknown distance function. *Psychometrika*, 27:125–140, 1962.

J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

W. S. Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 1952.

J. Tenenbaum V. de Silva. Unsupervised learning of curved manifolds. In *Proceedings of the MSRI workshop on nonlinear estimation*, 2003.

J. A. Walter. On interactive visualization of high dimensional data using the hyperbolic plane. In *Proceedings of the eighth ACM SIGKDD*, pages 123 – 132, 2002.

K. Q. Weinberger and L. K. Saul. Unsupervised learning of image manifolds by semidefinite programming. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-04)*, volume 2, pages 988–995, Washington D.C., 2004.