STAT 185 - Problem Set #6

November 19, 2019

Instructions

This assignment covers nonlinear manifold learning (ISOMAP and LLE) and the Johnson-Lindenstrauss Lemma through a mixture of theory and coding. You will need the file *pset6.Rdata* available on Canvas. The commands for loading the data assume this file, *pset6.Rmd*, and the Rdata file are in the same folder.

You can use the command *out<-embed(data, "Isomap", ndim = d, knn = k, get_geod = TRUE)* to compute the d dimensional representation of a data matrix *data* using the ISOMAP algorithm with k-nearest neighbors. The d-dimensional representation of the data is stored in *out@data@data* and the estimated geodesic distances are stored in *out@other.data$geod*.

You can use the command *out<-embed(data, "LLE", ndim = d, knn = k)* to compute the d dimensional representation of a data matrix *data* using the LLE algorithm with k-nearest neighbors. The d-dimensional representation of the data is stored in *out@data@data*.

You can add the option *.mute = c("message","output")* to each call of *embed* to supress the messages.

Complete the following questions and submit your solutions as a *pdf* via Canvas by 11:59 PM on Tuesday November 19th.

Problems

1. The variable *faces* contains 33 images (112×92 pixels) of a person taken at different angles. The data were obtained (and modified slightly) from the github repository maintained by Yuan Yao. Three images are shown below.

   a. Compute a one-dimensional represenation of the faces using ISOMAP with $k = 3$ nearest neighbors. Plot the first 32 images (ordered from the lowest 1-dimensional coordinate to the largest 1-dimensional coordinate) in an 8×4 grid. What feature of the images is the ISOMAP embedding capturing?

   b. Repeat part a, using LLE.

   c. The original ordering of the images (1 = face-forward, 33 = profile) is stored in the variable *faces.order*. Compare the ordering given by ISOMAP and LLE with the true ordering of the images.

2. The variable *digits* contains 28×28 pixel images (saved as 784-dimensional vectors) in the array *digits$pixels* and their associated labels *digits$labels*.

   a. Extract the first 1000-ones from the handwritten digits and the first 1000-sevens from the *digits* data and combine into a single data matrix. Generate a two-dimensional representation of the data saved in this new data matrix using ISOMAP with $k = 10, 50, 100, 500$ nearest neighbors. Show the two-dimensional configurations with the points corresponding to ones colored in blue and the points corresponding to sevens colored in red. Which choice of k nearest neighbors shows the best separation between the ones and sevens?

b. Repeat part a using LLE.

c. Repeat parts a and b using the first 1000-fours and 1000-nines. How does the performance compare to the results for a/b.

d. Discuss the different performance for the choice of algorithm and number of nearest neighbors. In particular, discuss the different performance between the 1/7 and 4/9 studies.

3. Extract the first 1000-twos from the *digits* images.

a. Generate and two-dimensional representation of the subset of twos using ISOMAP. Discuss your choice of k-nearest neighbors. Plot the two-dimensional configuration.

b. Show the images of the four handwritten twos from the most extremal points (leftmost,rightmost,upmost,downmost) of the two-dimensional embedding. Do the horizontal and vertical axes appear to have any association with features of the twos?

4. The swiss roll can be generated by mapping the rectangle

$$\left[\frac{3\pi}{2}, \frac{9\pi}{2}\right] \times [0, 15] \subset \mathbb{R}^2$$

to $\mathbb{R}^3$ through the map

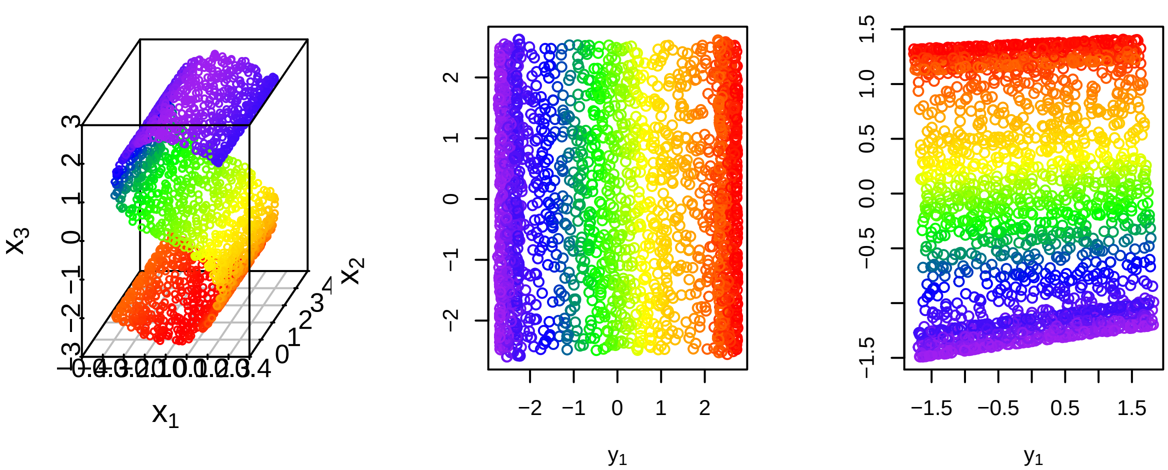
$$\psi : \mathbb{R}^2 \longmapsto \mathbb{R}^3, \qquad \psi((s,t)) = (s\cos(s), s\sin(s), t)^T$$

The points $\vec{x}_1 = (0, -3\pi/2, 0)$ and $\vec{x}_2 = (8\pi, 0, 15)$ are points on the swiss roll.

a. Estimate the geodesic distance from $\vec{x}_1$ to $\vec{x}_2$. Explain your work.

b. (BONUS) Compute the geodesic distance between these points analytically.

5. In this problem, we will investigate the effect of noise on the performance of ISOMAP and LLE. The variable *S\$points* contains 2000 samples from a S-shaped sheet in $\mathbb{R}^3$. To assist with plotting, associated colors for each point are stored in the variable *S\$colors*. For reference, the manifold is shown below with two-dimensional ISOMAP and LLE representations obtained using $k = 50$ nearest neighbors.



a. Add independent $N(0,1)$ noise to each entry of *S\$points*. Generate a two-dimensional representation of the data using ISOMAP with $k = 5$ nearest neighbors. Repeat with $N(0,1/4)$ and $N(0,1/10)$ noise.

b. Repeat part a using LLE.

c. Repeat parts a and b with $k = 50$ nearest neighbors.

d. Discuss the performance of each algorithm for $k = 5, 50$ nearest neighbors and each value of the noise.

6. In this problem, we are going to compare some of the features of the Johnson-Lindenstrauss Theorem (after some probability).

   a. Consider a matrix $U \in \mathbb{R}^{54,613 \times 54,613}$ with independent $N(0,1)$ entries. Find $E[UU^T]$.

   b. Let U be as in part a. Find an upper bound on

   $$P(\|U\|_F \geq a)$$

   where $\|U\|_F = \sqrt{\sum_{jk} U_{jk}^2}$ is the Frobenius norm.

   c. The variable *sun$x* contains 180 observations of random vectors in $\mathbb{R}^{54613}$. Each vector corresponds to the expression level of 54,613 genes in one of 180 cells (157 Glioma, 23 nontumor) in the experiments from Sun et al 2006. These data were obtained from the github repository datamicroarray. Suppose we map the gene expression vectors to k dimensional space using the first k columns of U, i.e.

   $$f : \mathbb{R}^{54613} \longmapsto \mathbb{R}^k, \qquad f(\vec{x}) = U_k^T \vec{x}, \qquad U_k = [\vec{u}_1, \ldots, \vec{u}_k] \in \mathbb{R}^{54613 \times k}.$$

   Here U is as in part a. Compare all of the pairwise Euclidean distances in the original data with the pairwise Euclidean distances after applying the random map for $k = 50, 100, 500, 1000$. Note, memory restrictions will likely prevent you from generating $U \in \mathbb{R}^{54613 \times 54613}$ but you should be able to generate U_{1000} then use its first 50, 100, 500 columns for U_{50}, U_{100}, and U_{500} respectively.

      i. Plot the original distances on the horizonal axis and the k-dimensional distances on the y axis.

      ii. For each k, generate histograms of the relative error between $\|\vec{x}_i - \vec{x}_j\|$ and $\|U_k^T \vec{x}_i - U_k^T \vec{x}_j\|/\sqrt{k}$ for $1 \leq i < j \leq 180$.

   d. From Johnson-Lindenstrauss, how large must k be so that the relative error between $\|\vec{x}_i - \vec{x}_j\|$ and $\|U_k^T \vec{x}_i - U_k^T \vec{x}_j\|/\sqrt{k}$ is no more than 20% for all $1 \leq i < j \leq 180$ with probability at least 99%?