Alex Lemon
EE263 (Introduction to Linear Dynamical Systems)
Homework 7
August 8, 2013

1 A power method for computing the matrix norm

Given a matrix $A \in \mathbb{R}^{m\times n}$, the following method can be used to compute the largest singular value, σ_1, and the corresponding left and right singular vectors, u_1 and v_1. For simplicity, you can assume that the largest singular value is isolated: that is, $\sigma_1 > \sigma_2$. Choose any nonzero $z(0) = a \in \mathbb{R}^n$, and repeat the iteration

$$w(t) = Az(t), \qquad z(t+1) = A^{\mathsf{T}} w(t).$$

For large t, we usually have that

$$\frac{w(t)}{\|w(t)\|} \approx u_1 \qquad \text{and} \qquad \frac{z(t)}{\|z(t)\|} \approx v_1.$$

Explain why this method usually works, and how it can fail if we are unlucky.

2 Drawing a representation of a graph

Consider an undirected graph with n nodes, and m edges. We want to draw a representation of this graph in the plane, which means assigning coordinates $(x_i, y_i) \in \mathbb{R}^2$ to node i for $i = 1, \ldots, n$. Let $x \in \mathbb{R}^n$ be the vector of x-coordinates of the nodes, and $y \in \mathbb{R}^n$ be the vector of y-coordinates of the nodes. One desirable property of our representation is that nodes that are connected by an edge should not be too far apart. We can try to find a representation that has this property by minimizing the objective

$$J = \sum_{(i,j)\in E} ((x_i - x_j)^2 + (y_i - y_j)^2),$$

where E is the edge set of the graph. Note that J is the sum of the squares of the lengths of all of the edges in the graph. Just minimizing J does not usually yield a sensible solution, so we need to introduce some additional constraints. Since the objective J is not affected by shifting all of the coordinates by some fixed amount, we will assume that the coordinates are centered: that is,

$$\sum_{i=1}^n x_i = 0, \qquad \text{and} \qquad \sum_{i=1}^n y_i = 0.$$

Another problem with just minimizing J is that there is a trivial solution: set $x = y = 0$. To force the nodes to spread out, we impose the constraints

$$\sum_{i=1}^n x_i^2 = 1, \qquad \sum_{i=1}^n y_i^2 = 1, \qquad \text{and} \qquad \sum_{i=1}^n x_i y_i = 0.$$

The first constraint says that the x-coordinates have unit variance, while the second constraint says that the y-coordinates have unit variance; the third constraint says that the

x-coordinates and y-coordinates are uncorrelated. Even with all of these additional constraints, there is still not a unique set of coordinates that minimize J. For example, if x and y are some set of coordinates satisfying the constraints, and $Q \in \mathbb{R}^{2\times 2}$ is orthogonal, then

$$\begin{bmatrix} \tilde{x}_i \\ \tilde{y}_i \end{bmatrix} = Q \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \qquad i = 1, \ldots, n$$

is another set of coordinates that also satisfies the constraints, and achieves the same value of J. Intuitively, we can rotate or reflect any set of coordinates to obtain another set of coordinates that is just as good. We will live with this ambiguity.

(a) Explain how to find coordinates x and y that minimize J subject to the centering and spreading constraints.

(b) The file `draw_graph_representation_data.m` defines the following variables.

- `A`, the adjacency matrix of the graph
- `x_circ`, a vector of x-coordinates obtained using a simple method that places the nodes at equally spaced points on a circle (the radius of the circle is chosen to satisfy the spreading constraints)
- `y_circ`, a vector of y-coordinates corresponding to `x_circ`

Apply your method to the graph described by `A`. Report the optimal value of J, and draw a corresponding representation of the graph. For comparison, compute the value of J associated with `x_circ` and `y_circ`, and draw the corresponding representation of the graph.

3 Worst and best directions of excitation for a suspension system

A suspension system is connected at one to a base (which can move or vibrate), and at the other end to a load (which is supposed to be isolated from the vibration of the base). Suitably discretized, the system is described by the linear dynamical system

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \quad x(0) = 0, \\ y(t) &= Cx(t), \end{aligned}$$

where $u(t) \in \mathbb{R}^3$ is the displacement of the base, and $y(t) \in \mathbb{R}^3$ is the displacement of the load. The input u has the form $u(t) = qv(t)$, where $q \in \mathbb{R}^3$ is a constant vector with $\|q\| = 1$, and $v(t) \in \mathbb{R}$ is the displacement amplitude as as function of time. In other words, the driving displacement u is always in the direction q, with amplitude given by the scalar signal v. The response of the system is judged by the RMS deviation of the load over a time interval of length 100: that is,

$$D = \left(\frac{1}{100} \sum_{t=1}^{100} \|y(t)\|^2 \right)^{\frac{1}{2}}.$$

The file `suspension_system_excitation_data.m` defines A, B, C, and $v(0), \ldots, v(99)$. Explain how to find the direction $q_{\max} \in \mathbb{R}^3$ that maximizes D, and the direction $q_{\min} \in \mathbb{R}^3$ that minimizes D. Report the directions $q_{\max}$, and $q_{\min}$, and the corresponding values of D.

4 Uncovering a hidden linear explanation

Consider a set of vectors $y_1, \ldots, y_N \in \mathbb{R}^n$, which might represent a collection of measurements or other data. Suppose we have

$$y_i \approx Ax_i + b, \qquad i = 1, \ldots, N,$$

where $A \in \mathbb{R}^{n \times m}$, $x_i \in \mathbb{R}^m$, $b \in \mathbb{R}^n$, and $m < n$. Then, we say that $y = Ax + b$ is a linear explanation of the data y. We refer to x as the vector of factors or underlying causes. For example, suppose $N = 500$, $n = 30$, and $m = 5$. In this case, we have 500 measurements $y_1, \ldots, y_{500}$, and each vector y_i consists of 30 scalar measurements. But this 30-dimensional data can actually be explained by a much smaller set of 5 factors (the components of x_i). In other words, we are given $y_1, \ldots, y_N$, and the goal is to find m, A, b, and $x_1, \ldots, x_N$ such that $y_i \approx Ax_i + b$. We judge the accuracy of a proposed explanation using the RMS error:

$$J = \left(\frac{1}{N} \sum_{i=1}^{N} \|y_i - (Ax_i + b)\|^2 \right)^{\frac{1}{2}}.$$

A trivial explanation of the data is explained with $x_i = y_i$, $A = I$, and $b = 0$. In other words, the data explains itself. In this case, we have $y_i = Ax_i + b$, so the RMS error is zero. But such a model misses the point of our analysis. To be a useful explanation, we want to have m substantially smaller than n: that is, the high-dimensional data is explained by comparatively few factors. Typically such a useful explanation will have a value of J that is nonzero, but still hopefully small. Thus, we want the number of factors m to be small subject to the constraint that the RMS error J is not too large.

Even if we fix the number of factors m, a linear explanation of the data is not unique. Suppose A, b, and $x_1, \ldots, x_N$ give a linear explanation of our data, with $x_i \in \mathbb{R}^m$. Then, for instance, we can multiply the matrix A by 2, and divide each vector x_i by 2, and we have another linear explanation of the data. More generally, for any invertible matrix $F \in \mathbb{R}^{m \times m}$, and any vector $g \in \mathbb{R}^m$, we have that

$$Ax_i + b = (AF^{-1})(Fx_i + g) + (b - AF^{-1}g).$$

Thus,

$$\tilde{A} = AF^{-1}, \qquad \tilde{b} = b - AF^{-1}g, \qquad \tilde{x}_i = Fx_i + g,$$

is another linear explanation of the data that does just as well as the linear explanation with A, b, and $x_1, \ldots, x_N$. In order to remove this ambiguity, we assume that $x_1, \ldots, x_N$ are standardized such that

$$\frac{1}{N} \sum_{i=1}^{N} x_i = 0, \qquad \text{and} \qquad \frac{1}{N} \sum_{i=1}^{N} x_i x_i^{\mathsf{T}} = I.$$

In other words, $x_1, \ldots, x_N$ have mean zero, and unit covariance.

(a) Explain how to find a hidden linear explanation for the data set $y_1, \ldots, y_N$. In particular, you need to specify how to determine m, A, b, and $x_1, \ldots, x_N$.

(b) Apply your method to the data in `hidden_linear_explanation_data.m`. This file defines the given data as a matrix

$$Y = \begin{bmatrix} y_1 & \cdots & y_N \end{bmatrix}.$$

Report your values of m, A, and b. Verify that $y_i = Ax_i + b$ by computing the norms of error vectors $\|y_i - (Ax_i + b)\|$ for $i = 1, \ldots, N$. Sort these norms in descending order, and make a stem plot of them.

5 Alternating projections for low-rank matrix completion

In the low-rank matrix completion problem, you are given some of the entries of a matrix, along with an upper bound on its rank; you are to guess or estimate the remaining entries. This arises in several applications. In this problem, you will use a heuristic method for low-rank matrix completion.

You are told that $A \in \mathbb{R}^{m \times n}$ has rank at most r, and that $A_{ij} = A_{ij}^{\text{known}}$ for $(i,j) \in \mathcal{K}$, where $\mathcal{K} \subseteq \mathbb{N}_m \times \mathbb{N}_n$ is the set of indices of the known entries of A. Your job is to estimate A_{ij} for $(i,j) \notin \mathcal{K}$.

You will use an alternating projection method to find an estimate $\hat{A}$ of A. After choosing an initial point $\hat{A}^{(0)}$ that has the correct known entries, you will alternate between the following two projections.

- *Project to the closest matrix satisfying the rank constraint.* Set $\tilde{A}^{(k)}$ to be the matrix of rank at most r that is closest to $\hat{A}^{(k)}$ in the Frobenius norm: that is, choose $\tilde{A}^{(k)}$ to be the matrix that minimizes

$$\|\tilde{A}^{(k)} - \hat{A}^{(k)}\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} (\tilde{A}_{ij}^{(k)} - \hat{A}_{ij}^{(k)})^2}$$

  subject to the constraint that $\text{rank}(\tilde{A}^{(k)}) \leq r$.

- *Project to the closest matrix with the known entries.* Set $\hat{A}^{(k+1)}$ to be the matrix with correct known entries that is closest to $\tilde{A}^{(k)}$ in the Frobenius norm: that is, choose $\hat{A}^{(k+1)}$ to be the matrix that minimizes

$$\|\hat{A}^{(k+1)} - \tilde{A}^{(k)}\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} (\hat{A}_{ij}^{(k+1)} - \tilde{A}_{ij}^{(k)})^2}$$

  subject to the constraint that $\hat{A}_{ij}^{(k+1)} = A_{ij}^{\text{known}}$ for all $(i,j) \in \mathcal{K}$.

This is only a heuristic method: it can fail to converge, or it can converge to different limits depending on the initial point. However, the method often works well in practice.

(a) Explain to perform each of these projections.

(b) The file `matrix_completion_data.m` defines the following variables.

- `K`, a $|\mathcal{K}| \times 2$ matrix whose rows are the indices (i,j) of the known entries of A

- `Aknown`, a vector of length $|\mathcal{K}|$ whose components are the known entries of A
- `A`, the matrix A (this can only be used for evaluating the quality of your estimates)

Use $k_{\max} = 300$ steps of the alternating-projections algorithm to find a low-rank matrix completion. For your initial estimate, take

$$\hat{A}_{ij}^{(0)} = \begin{cases} A_{ij}^{\text{known}} & (i,j) \in \mathcal{K}, \\ \bar{A} & (i,j) \notin \mathcal{K}, \end{cases}$$

where $\bar{A}$ is the average value of the known entries of A:

$$\bar{A} = \frac{1}{|\mathcal{K}|} \sum_{(i,j) \in \mathcal{K}} A_{ij}^{\text{known}}.$$

Plot $\|\hat{A}^{(k)} - A\|_F$ for $k = 1, \ldots, k_{\max}$.