Asmt 7: Dimensionality Reduction

Turn in through Canvas by 2:45pm:
Wednesday, April 8
100 points

Overview

In this assignment you will explore regression techniques on high-dimensional data.

You will use a data set for this assignment:

- http://www.cs.utah.edu/~jeffp/teaching/cs5140/A7/A.csv

and a file stub:

- http://www.cs.utah.edu/~jeffp/teaching/cs5140/A7/FD.py

For python, you can use the following approach to load the data:

```
A = np.loadtxt('A.csv', delimiter=',')
```

As usual, it is recommended that you use LaTeX of some other tool that can properly format math for this assignment. If you do not, you may lose points if your assignment is difficult to read or hard to follow. Find a sample form in this directory: http://www.cs.utah.edu/~jeffp/teaching/latex/

1 Singular Value Decomposition (70 points)

First we will compute the SVD of the matrix `A` we have loaded

```
import numpy as np
from scipy import linalg as LA
U, s, Vt = LA.svd(A, full_matrices=False)
```

Then take the top k components of `A` for values of $k = 1$ through $k = 10$ using

```
Uk = U[:,:k)
Sk = S[:k,:k]
Vtk = Vt[:k,:]
Ak = Uk @ Sk @ Vtk
```

A (40 points): Compute and report the L_2 norm of the difference between `A` and `Ak` for each value of k using

```
LA.norm(A-Ak,2)
```

B (10 points): Find the smallest value k so that the L_2 norm of `A-Ak` is less than 10% that of `A`; k might or might not be larger than 10.

C (20 points): Treat the matrix as 4000 points in 20 dimensions. Plot the points in 2 dimensions in the way that minimizes the sum of residuals squared, and describe briefly how you did it.

2 Frequent Directions and Random Projections (30 points)

Use the stub file `FD.py` to create a function for the Frequent Directions algorithm (**Algorithm 16.2.1**). Consider running this code on matrix `A`.

A (30 points): Measure the error $\max_{\|x\|=1} |\|Ax\|^2 - \|Bx\|^2|$ as `LA.norm(A.T @ A - B.T @ B)`

- How large does `l` need to be for the above error to be at most $\|A\|_F^2/10$?
- How does this compare to the theoretical bound (e.g. for $k = 0$).
- How large does `l` need to be for the above error to be at most $\|A - A_k\|_F^2/10$ (for $k = 2$)?

Note: you can calculate $\|A\|_F^2$ as `LA.norm(A, 'fro')^2`.

3 BONUS

A (10 points): Create another `l x d` matrix B, but using random projections. You can do this by creating an `l x n` matrix `S`, and letting `B = SA`. Fill each entry of `S` by an independent normal random variable $S_{i,j} = \frac{1}{\sqrt{1}} N(0, 1)$.

Estimate how large should `l` be in order to achieve $\max_{\|x\|=1} |\|Ax\|^2 - \|Bx\|^2| \leq \|A\|_F^2/10$. To estimate the relationship between `l` and the error in this randomized algorithm, you will need to run multiple trials. Be sure to describe how you used these multiple trials, and discuss how many you ran and why you thought this was enough trials to run to get a good estimate.

B (2 points) Consider any $n \times d$ matrix A. For some parameter $t \geq 1$, assume that $n > d > 100t^2$. Without knowing A, explain at most how many squared singular values of A could be larger than $\|A\|_F^2/t$.