Assignment 5 - Regression*

Due: Monday, April 2
Late assignments accepted (with full credit) until Wednesday, April 4
Turn in a hard copy at the start of class

Overview

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

You will use a few data sets for this assignment:

- `http://www.cs.utah.edu/~jeffp/teaching/cs5955/A5/M.dat`
- `http://www.cs.utah.edu/~jeffp/teaching/cs5955/A5/X.dat`
- `http://www.cs.utah.edu/~jeffp/teaching/cs5955/A5/Y.dat`

This data sets are in matrix format and can be loaded into MATLAB or OCTAVE. By calling
`load filename` (for instance `load M.dat`)
it will put in memory the the data in the file, for instance in the above example the matrix `M`. You can then display this matrix by typing
`M`

As usual, it is highly recommended that you use LaTeX 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 (4 points)

First we will first computer the SVD of the matrix M we have loaded
```
[U,S,V] = svd(M)
```
Then take the top k columns components of M for values of $k = 1$ through $k = 10$ using
```
Uk = U(:,1:k)
Sk = S(1:k,1:k)
Vk = V(:,1:k)
Mk = Uk*Sk*Vk'
```
Compute and report the L_2 norm of the difference between M and Mk for each value of k using
```
norm(M-Mk,2)
```
Find the value k so that the L_2 norm of `M-Mk` is 10% that of `M`; k may be larger than 10.

2 Column Sampling (8 points)

Select t (for t from 1 to 30) columns $\{c1, c2, \ldots, c30\}$ using the two types of column sampling from the matrix data set `M`.

Type 1: For each column j `M(:,j)` calculate the squared norm `sj = norm(M(:,j))^2`, and select t columns proportional to the values `sj`.

*CS 6955 Data Mining; Spring 2012 Instructor: Jeff M. Phillips, University of Utah

Type 2: Calculate the SVD of M: `[U, S, V] = svd(M)`. For each column j calculate the squared norm projected onto the column space of the top k-singular vectors: `wj = norm(Uk*Uk'*M(:,j))^2`, and select t columns proportional to the values `wj`. (Use $k = 5$.)

We now need to measure how accurate a subspace these columns represent. Construct a matrix with the sampled columns `C = [c1 c2 c3 ... c30]`. Then create a projection matrix onto the column space of `C` as `P = C*inverse(C'*C)*C'`. Finally calculate the L_2 norm of the difference between M and M projected on to the column space of C as `norm(M - P*M,2)`.

If in the `inverse` returns `NaN`, then try `pinv`.

A (4 points): Report this error for each choice of t. Since this is a randomized algorithm, the values may vary. You should repeat this experiment several times to get good representative values. Also the nice plotting functions of MATLAB/OCTAVE may be useful as a replacement for presenting this data instead of reporting a series of numbers.

B (2 points): For both types of column sampling, estimate how large t need to be to reach the same error as the SVD approach with $k = 5$.

C (2 points): Using the values of t found in part **B**, for both types of column sampling, estimate the number of non-zero entries in these t columns sampled. Compare this value to the number of non-zero entries in `U5` constructed using the SVD.

3 Linear Regression (4 points)

We will find coefficients `A` to estimate `X*A = Y`. We will compare two approaches *least squares* and *ridge regression*.

Least Squares: Set `A = inverse(X' * X)*X'*Y`

Ridge Regression: Set `As = inverse(X'*X + s*eye(6))*X'*Y`

A (2 points): Solve for the coefficients A (or As) using Least Squares and Ridge Regression with $s = \{0.1, 0.3, 0.5, 1.0, 2.0\}$. For each set of coefficients, report the error in the estimate $\hat{Y}$ of Y as `norm(Y - X*A,2)`.

B (2 points): Create three row- subsets of `X` and `Y`

- `X1 = X(1:8,:)` and `Y1 = Y(1:8)`
- `X2 = X(3:10,:)` and `Y2 = Y(3:10)`
- `X3 = [X(1:4,:); X(7:10,:)]` and `Y3 = [Y(1:4); Y(7:10)]`

Repeat the above procedure on these subsets and *cross-validate* the solution on the remainder of `X` and `Y`. Specifically, learn the coefficients `A` using, say, `X1 and Y1` and then measure `norm(Y(9:10) - X(9:10,:)*A,2)`.

Which approach works best (averaging the results from the thee subsets): Least Squares, or for which value of s using Ridge Regression?

4 BONUS) (5 points)

The Lasso Regression technique takes as input a matrix X and an vector Y and for some parameter t finds the coefficient vector A that minimizes

$$\|Y - XA\|_2 + t\|A\|_1.$$

The optimal values of A can be found as follows. Start with $t = 0$ and for all $a_j \in A$ with $a_j = 0$. It then finds the column of X, corresponding with a coefficient $a \in A$, that has the most correlation with Y. Then as we increase t, it allows the associated coefficient a to increase. It then determines certain *break points* in the value t, where it becomes beneficial to make other coefficients non-zero, placing them in the *active set* of non-zero coefficients. Between each pair of consecutive break points, only coefficients in the *active set* change. Show that each coefficient changes *linearly* with respect to t between any pair of break points.