Asmt 6: Regression

Turn in through Canvas by 11:59pm:
Wednesday, March 24
100 points

Overview

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

You will use a few data sets for this assignment:

- Canvas –> Files –> Assignments –> Regression–> X.csv
- Canvas –> Files –> Assignments –> Regression–> y.csv
- Canvas –> Files –> Assignments –> Regression–> M.csv
- Canvas –> Files –> Assignments –> Regression–> W.csv

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

```
X = np.loadtxt('X.csv', delimiter=',')
y = np.loadtxt('y.csv', delimiter=',')
```

As usual, it is recommended that you use LaTeX for this assignment (or similar way to properly typeset math). If you do not, you may lose points if your assignment is difficult to read or hard to follow. Find a sample form in: Canvas –> Files –> Assignments –> Assignment_Latex_Template.zip.

1 Linear Regression & Cross-Validation (100 points)

We will find coefficients `alpha` to estimate `X*alpha ≈ y`, using the provided datasets `X` and `y`. We will compare two approaches *least squares* and *ridge regression*. (e.g., in python as)

Least Squares: Set `alpha = LA.inv(X.T @ X) @ X.T @ y.T`
Ridge Regression: Set `alphas = LA.inv(X.T @ X + s*np.identity(50)) @ X.T @ y.T`

A (30 points): Solve for the coefficients `alpha` (or `alphas`) using Least Squares and Ridge Regression with $s \in \{0.1, 0.3, 0.7, 0.9, 1.1, 1.3, 1.5\}$ (i.e. s will take on one of those 7 values each time you try, say obtaining `alpha04` for $s = 0.4$). For each set of coefficients, report the error in the estimate $\hat{y}$ of y as `norm(y - X*alpha,2)`.

B (30 points): Create four row-subsets of `X` and `Y`

- `X1 = X[:75,:]` and `Y1 = Y[:75]`
- `X2 = X[25:,:]` and `Y2 = Y[25:]`
- `X3 = np.vstack((X[:50,:], X[75:,:]))` and `Y3 = np.concatenate((Y[:50], Y[75:]))`
- `X4 = np.vstack((X[:25,:], X[50:,:]))` and `Y4 = np.concatenate((Y[:25], Y[50:]))`

Repeat the above procedure on these subsets and *cross-validate* the solution on the remainder of `X` and `Y`. Specifically, learn the coefficients `alpha` using, say, `X1 and Y1` and then measure `np.norm(Y[75:] - X[75:,:] @ alpha,2)`.

C (15 points): Which approach works best (averaging the results from the three subsets): Least Squares, or for which value of s using Ridge Regression?

D (15 points): Use the same 4 test / train splits, taking their average errors, to estimate the average squared error on each predicted data point.

What is problematic about the above estimate, especially for the best performing parameter value s?

E (10 points): Even circumventing the issue raised in part **D**, what *assumptions* about how the data set $(\texttt{X}, \texttt{y})$ is generated are needed in an assessment based on cross-validation?

2 Bonus: Matching Pursuit (5 points)

Consider a linear equation `W = M*S` where `M` is a measurement matrix filled with random values $\{-1, 0, +1\}$ (although now that they are there, they are no longer random), and `W` is the output of the sparse signal `S` when measured by `M`.

Use Matching Pursuit (as described in the book as **Algorithm 5.5.1**) to recover the non-zero entries from `S`. Record the order in which you find each entry and the residual vector after each step.