Data Mining: 36-462/36-662
Homework 5

Due Thursday April 11 2013
(at the beginning of lecture)

Append your R code to the end of your homework. In your solutions, you should just present your R output (e.g. numbers, table, figures) or snippets of R code as you deem it appropriate. Make sure to present your results (i.e., your R ouput) in a clear and readable fashion. Careless or confusing presentations will be penalized.

Problem 1

(a) Given $y \in \mathbb{R}^n$, consider ridge regression with predictor matrix $X = I_{n\times n}$, i.e.,

$$\begin{aligned}\hat{\beta}^{\text{ridge}} &= \underset{\beta\in\mathbb{R}^n}{\text{argmin}} \ \|y-\beta\|_2^2 + \lambda\|\beta\|_2^2 \\ &= \underset{\beta\in\mathbb{R}^n}{\text{argmin}} \ \sum_{i=1}^n (y_i-\beta_i)^2 + \lambda\sum_{i=1}^n \beta_i^2.\end{aligned}$$

Show that the solution is

$$\hat{\beta}_i^{\text{ridge}} = \frac{y_i}{1+\lambda}, \quad i = 1,\ldots n.$$

(b) For the lasso with identity predictor matrix,

$$\begin{aligned}\hat{\beta}^{\text{lasso}} &= \underset{\beta\in\mathbb{R}^n}{\text{argmin}} \ \|y-\beta\|_2^2 + \lambda\|\beta\|_1 \\ &= \underset{\beta\in\mathbb{R}^n}{\text{argmin}} \ \sum_{i=1}^n (y_i-\beta_i)^2 + \lambda\sum_{i=1}^n |\beta_i|,\end{aligned}$$

the solution (bonus problem) is

$$\hat{\beta}_i^{\text{lasso}} = \begin{cases} y_i + \lambda/2 & y_i < -\lambda/2 \\ 0 & |y_i| \le \lambda/2 \\ y_i - \lambda/2 & y_i > \lambda/2 \end{cases}, \quad i = 1,\ldots n.$$

For a fixed value of λ (e.g., you can take $\lambda = 1$), draw $\hat{\beta}_i^{\text{ridge}}(y_i)$ and $\hat{\beta}_i^{\text{lasso}}(y_i)$ as functions of y_i. Describe the difference between these two coefficient functions.

(c) Suppose that $X \in \mathbb{R}^{n \times p}$ is orthogonal, i.e., $X^T X = I_{p \times p}$. Consider the ridge regression and lasso problems:

$$\hat{\beta}^{\text{ridge}} = \underset{\beta \in \mathbb{R}^p}{\text{argmin}} \ \|y - X\beta\|_2^2 + \lambda \|\beta\|_2^2$$
$$\hat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\text{argmin}} \ \|y - X\beta\|_2^2 + \lambda \|\beta\|_1.$$

Let $X_i \in \mathbb{R}^n$ denote the ith column of X. Show that the solutions $\hat{\beta}^{\text{ridge}}, \hat{\beta}^{\text{lasso}}$ are given by the same formulas as in parts (a) and (b), but with $X_i^T y$ in place of y_i (and p in place of n).

(Hint 1: if $O \in \mathbb{R}^{n \times n}$ is an orthogonal and square matrix, recall that it preserves distances, i.e., $\|Oz\|_2 = \|z\|_2$ for any $z \in \mathbb{R}^n$.)

(Hint 2: using Hint 1 and the fact that X has orthonormal columns, show that

$$\|y - X\beta\|_2^2 = \|X^T y - \beta\|_2^2 + c,$$

where c is a constant, meaning that it doesn't depend on β.)

(d) If $X \in \mathbb{R}^{n \times p}$ is orthogonal, what are the linear regression coefficients $\hat{\beta}^{\text{LS}}$ of y on X? Given your answers for the ridge regression and lasso coefficients in part (c) (and the picture you drew in part (b)), give a few sentences interpreting the ridge and lasso coefficients as a functions of the linear regression coefficients.

Problem 2

Consider the linear regression of predictors $y \in \mathbb{R}^n$ on predictors $X \in \mathbb{R}^{n \times p}$. Let $x_i \in \mathbb{R}^p$, $i = 1, \ldots n$ denote the predictors measurements, i.e., these are the rows of X. Recall that the linear regression estimator is given by

$$\hat{f}(x_i) = x_i^T \hat{\beta}^{\text{LS}} = x_i^T (X^T X)^{-1} X^T y.$$

In this problem you will prove the formula

$$y_i - \hat{f}^{-i}(x_i) = \frac{y_i - \hat{f}(x_i)}{1 - S_{ii}}, \tag{1}$$

where $S = X(X^T X)^{-1} X^T$, and $\hat{f}^{-i}$ is the linear regression estimator fit to all but the ith training pair (x_i, y_i). This gives us big savings when computing the leave-one-out cross-validation error:

$$\frac{1}{n} \sum_{i=1}^n \left(y_i - \hat{f}^{-i}(x_i)\right)^2 = \frac{1}{n} \sum_{i=1}^n \left[\frac{y_i - \hat{f}(x_i)}{1 - S_{ii}}\right]^2, \tag{2}$$

because we don't have to actually compute $\hat{f}^{-i}$ for each i.

(a) Recall that $S = X(X^T X)^{-1} X^T$, and let $Z \in \mathbb{R}^{(n-1) \times p}$ denote the predictor matrix X but with its ith row removed. Argue that

$$S_{ii} = x_i^T (X^T X)^{-1} x_i \quad \text{and} \quad \hat{f}^{-i}(x_i) = x_i^T (Z^T Z)^{-1} Z^T y_{-i},$$

where $y_{-i} \in \mathbb{R}^{n-1}$ denotes the observation vector y but with its ith component removed. Argue also that

$$X^T X = Z^T Z + x_i x_i^T \quad \text{and} \quad X^T y = Z^T y_{-i} + x_i y_i.$$

(b) For a matrix $A \in \mathbb{R}^{k \times k}$ and vectors $u, v \in \mathbb{R}^k$, the Sherman-Morrison update formula gives the inverse of $A + uv^T$ in terms of the inverse of A:

$$(A + uv^T)^{-1} = A^{-1} - \frac{A^{-1} u v^T A^{-1}}{1 + v^T A^{-1} u}.$$

Using this formula, prove that $\hat{f}^{-i}(x_i)$ can be expressed as

$$\hat{f}^{-i}(x_i) = \frac{\hat{f}(x_i) - S_{ii} y_i}{1 - S_{ii}}.$$

Rearrange this to conclude the result in (1).

(Hint: use your results from (a), and the Sherman-Morrison formula to express $(Z^T Z)^{-1} = (X^T X - x_i x_i^T)^{-1}$ in terms of $(X^T X)^{-1}$.)

(c) Prove the result (1) when $\hat{f}$ is the ridge regression estimator, $\hat{f}(x_i) = x_i^T \hat{\beta}^{\text{ridge}}$, at any arbitrary tuning parameter value $\lambda \geq 0$.

(Hint: your proof shouldn't be longer than a few lines! Recall the result from homework 4, problem 2(a).)

Problem 3

Download the file "splines.Rdata" used in lecture 19, from the class website. Using the code "19-val2.R" (also provided on the class website) as a starting point, perform leave-one-out cross-validation for the smoothing spline estimator on the splines data, over the same range of degrees of freedom values as those in the file (i.e., `dfs = 2:30`). Plot the CV error curve. What value of degrees of freedom is selected here (by the usual rule)?

Now compute the right-hand side of the formula in (2). At each the tuning parameter value (each value of degrees of freedom), for the diagonal elements of S, take the `lev` component of the object returned by the `smooth.spline` function. Are these values (the right-hand sides in (2), over the various degrees of freedom values) different from the leave-one-out CV errors you computed above? Plot them and compare.

Problem 4

In this problem, you'll investigate using linear discriminant analysis (LDA) to label handwritten digits as a "0", "1", or "4". Download the file "zip.014.Rdata" from the class website and load it into your R session. (This data is taken from the ESL website, `http://www-stat.stanford.edu/~tibs/ElemStatLearn/`.) Now you should have `x.014.tr`, `y.014.tr`: the predictors and class labels for the training set, and `x.014.te`, `y.014.te`: the predictors and class labels for the test set. Recall that each row of the predictor matrix

is a digitized 16×16 image that has been unraveled into a vector (of length 256). You can use the function `plot.digit` from the "plot.digit.R" file provided on the class website for homework 2 to plot these images. For example, try `plot.digit(x.014.tr[1,])` to plot the first image in the training set.

(a) Perform LDA using the `lda` function from the `MASS` package. What is the dimension p of the original feature space? What is the dimension used by LDA in the transformed space, i.e., the real dimension used by LDA in order to classify between $0, 1, 4$? Plot the data in the transformed space, according to the `scaling` matrix returned by the function `lda`. To emphasize the differences between classes, you can use `col=y.014.tr+1` and/or `pch=as.character(y.014.tr)`.

(b) Draw decision boundaries between the classes $0, 1, 4$ on your plot from (a).

(c) Predict the labels of the test set data `x.014.te` using the LDA model fit from (a). (Hint: use the `predict` function; to read the relevant documentation, type `?predict.lda`.) What is the misclassification rate?

(d) Using the LDA model from (a), draw the class centroids and the decision boundaries in the transformed space. Now draw the test observations on top of this plot. Pick out a few misclassified observations, and plot these digits using `plot.digit`. Are these mistakes "excusable", i.e., are they digits that you would have had trouble classifying by eye?

(e) Install the package `class` using `install.packages("class")`. Use the function `knn` to perform 1-nearest-neighbor classification on the test set `x.014.te`—that is, for every point in `x.014.te`, label this observation according to the class of its nearest neighbor in the training set `x.014.tr`. What is the misclassification rate of this rule?

(f) What can you say in terms of comparing the rules (LDA and 1-nearest-neighbors)? Which has a lower misclassification rate? Which is easier to visualize? What is the effective dimension used by each rule?

Bonus problem 1

Prove the formula for the lasso solution in the identity predictor matrix case, $X = I$, given in Problem 1(b).

Bonus problem 2

Suppose $\hat{f}$ is a linear estimator fit to the training data (x_i, y_i), $i = 1, \ldots n$, meaning that the vector of fitted values $\hat{y} = (\hat{f}(x_1), \ldots \hat{f}(x_n))$ can be written as

$$\hat{y} = Sy$$

for some matrix $S \in \mathbb{R}^{n \times n}$ (here S can depend on the inputs x_i). You proved in problem 2 that this formula held when $S = X(X^T X)^{-1} X^T y$. In general (beyond this case), for what kinds of matrices S does the leave-one-out shortcut (1) hold?