STA 414/2104, Spring 2013 — Assignment #1

Due at the start of class on February 14. Please hand it in on 8 1/2 by 11 inch paper, stapled in the upper left, with no other packaging.

This assignment is to be done by each student individually. You may discuss it in general terms with other students, but the work you hand in should be your own. In particular, you should not leave any discussion with someone else with any written notes (either on paper or in electronic form).

In some supervised learning situations, the input variables available for predicting the response variable come in two (or more) groups. For example, if we are trying to predicting the topic of a web document, we might look both at the number of times various words occur in the document, and at whether the document is linked to from certain popular websites. Sometimes both groups of inputs may be highly relevant, but other times, one or the other (or both) might not be of much use in predicting the response (at least if we already have the inputs in the other group).

If we have a huge amount of training data, perhaps we can get good results by just fitting a model that uses all the inputs, without any regularization. But with a limited amount of data, getting good results may require estimating parameters with a penalty added to the log likelihood, or using a Bayesian method with a prior distribution. When the inputs come in two groups, using separate penalties or prior distributions for regression coefficients in the two groups may be desirable — otherwise, we might "over regularize" inputs in one group, while "under regularizing" inputs in the other. If we can somehow automatically choose the appropriate penalty or prior for each group, we will have effectively learned the degree to which each group of inputs is relevant to the prediction problem.

In the lectures, we have seen how the magnitude of a penalty can be set by minimizing loss in a cross-validation assessment, and how the variance for a prior distribution can be set by maximizing marginal likelihood. In this assignment, you will apply these methods to the situation where there are two groups of inputs, and hence two penalty magnitudes, or two prior variances, which can jointly be set based on cross-validation loss or marginal likelihood.

For this assignment, you will use two regression datasets (with real-valued responses) that I generated synthetically. The first data set has 15 inputs in the first group, and 20 inputs in the second group (total 35 inputs). The second dataset has 20 inputs in both groups (total 40 inputs). For both datasets, I generated 50 training cases, and 1000 test cases. The test cases are supplied to allow you to see how well the methods performed. You should look at the responses in the test cases only at the very end, when computing the test errors. In a real application, you would of course not know the values of the responses in the test cases at the time when you were making predictions for them.

The datasets are available from the course webpage, at

`http://www.utstat.utoronto.ca/~radford/sta414/`

There are four text files for each dataset, containing inputs and responses for the training and test sets. The files of response values can be read with the R 'scan' function, which returns a vector. The files of inputs should be read with 'read.table', using the 'head=TRUE' option, since the first line of each file contains names for the inputs. The result of 'read.table' is a data frame,

which you should convert to a matrix with 'as.matrix' (this is likely to make your program run faster). Inputs in the first group will be the first so-many columns of this matrix, with the second group consisting of the remaining columns. You may consider the rows (that is, the cases) to be in random order (so you won't have to randomize when selecting validation sets).

The course webpage also has examples of R programs for doing penalized least squares with cross-validation and for fitting a Bayesian linear model with priors selected by marginal likelihood. You may use these as a starting point for your programs if you wish. Note that in this assignment, we will use only the original input values, not various functions of these inputs.

For the method of penalized least squares with cross-validation, you should write a function that takes as arguments the matrix of training inputs, the vector of training responses, the number of inputs in the first group, and a set of values to consider for the penalty magnitudes (λ_1 and λ_2), and returns a matrix of average squared errors from a 10-fold cross-validation assessment of how well each combination of lambda values for the two groups of inputs does. The penalty used should be of the form

$$\lambda_1 \sum_{j=1}^{p_1} \beta_j^2 \;+\; \lambda_2 \sum_{j=p_1+1}^{p} \beta_j^2$$

where p is the total number of inputs, of which p_1 are in the first group. Notice that the intercept, β_0, is not penalized.

You should also write a function that returns a matrix of average squared errors on test cases when using each combination of values for λ_1 and λ_2. It will take the same arguments as the cross-validation function, plus the matrix of inputs for test cases, and the vector of responses for test cases.

When trying out this method on the two datasets provided, you should consider values of 0, 0.25, 0.5, 1, 2, 4, 8, 16, and 32 for λ_1 and λ_2.

For the Bayesian method with prior variances selected using marginal likelihood, you should write a function that takes as arguments the matrix of training inputs, the vector of training responses, the number of inputs in the first group, a prior standard deviation for β_0, a set of values to consider for the prior standard deviations of β_j in each of the two groups (ω_1 and ω_2), and a set of values to consider for the residual standard deviation (σ), and returns an array of log marginal likelihoods for each combination of ω_1, ω_2, and σ. (Note that arrays with three dimensions can be created with the 'array' function in R).

You should also write a function that returns an array of average squared errors on test cases when using each combination of ω_1, ω_2, and σ, which takes the same arguments plus the matrix of inputs for test cases, and the vector of responses for test cases.

When trying out this method on the two datasets provided, you should consider values of 0.05, 0.1, 0.2, 0.4, 0.8, and 1.6 for ω_1 and ω_2, and values of 0.35, 0.5, 0.7, 1.0, 1.4, and 2.0 for σ.

You should hand in a listing of your functions and the R scripts used to apply them to the datasets, the output you obtained (cross-validation errors, marginal likelihoods, and test errors), and a discussion of what it means. Your discussion should address issues such as whether the two methods produced similar results, whether these results were better than when no regularization is used (ie, when $\lambda_1 = \lambda_2 = 0$), whether the results were better than when the inputs where not divided into two groups (ie, when $\lambda_1 = \lambda_2$ or $\omega_1 = \omega_2$), whether the two datasets differed, and whatever else you see that is interesting about the results.