STA 414/2104, Spring 2011 — Assignment #1

Due at the start of class on March 1. 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 paper or electronic).

In this assignment, you will extend the R functions for linear basis function models demonstrated in class (available from the course web page) to handle two inputs, and try your functions out on two data sets (also available from the course web page).

Your program should be written in R, but should not use any of the built-in R functions for fitting linear models. (Ie, you should implement the computations yourself, using R's functions for matrix operations.) You should, however, use the R programs I wrote, modifying or replacing parts as needed. (Though some parts of my programs may not be relevant to this assignment.)

You should hand in a listing of your program, with suitable but not excessive comments. You should also hand in the output of your program (text or graphics), and your discussion of the results.

For each of the two datasets, there is a data file with 80 training cases, plus a data file with 1000 test cases (which you should look at only at the very end of the analysis). These data files start with a header line giving the names of the variables (x1, x2, and t) and then as many lines as cases, with three numbers per line. The first two numbers are the two inputs for that training case; the last number is the associated target value, which you are trying to predict from the inputs. The data file with the training cases for the first dataset can be read with commands like

```
d <- as.matrix(read.table ("ass1-train1.txt", head=TRUE))
xm <- d[,1:2]
tv <- d[,3]
```

This stores the two inputs for the 80 training cases in the 80×2 matrix `xm`, and the targets for these 80 cases in the 80-element vector `tv`. You can read the data file of test cases for the first dataset in the same way, except it's called `ass1-test1.txt`. The second dataset has training and test cases in files `ass1-train2.txt` and `ass1-test2.txt`.

In both dataset, both inputs are always between 0 and 1. The order of cases in a file is random.

For both datasets (separately), you will model the relationship of the target to the two inputs using a linear basis function model. The model using Gaussian basis functions that I used for the example with one input can be extended to two inputs in two ways (at least) — one way produces an additive model, the other a model with interactions between the inputs.

As in the example from lectures, we use a model with following general form:

$$\begin{aligned} t &= y(x,w) \;+\; \text{noise} \\ y(x,w) &= w_0 \;+\; \sum_{j=1}^{M-1} w_j \phi_j(x) \;=\; w^T\phi(x) \end{aligned}$$

We assume that the noise is normal with mean 0 and some variance σ^2.

For an additive model, the basis functions (apart from $\phi_0(x) = 1$, associated with the intercept, w_0) will have the form

$$\begin{aligned}\phi_j(x) &= \exp(-(x_1 - \mu_j)^2/(2s^2)) \\ \phi_{G+j}(x) &= \exp(-(x_2 - \mu_j)^2/(2s^2))\end{aligned}$$

Here, j ranges from 1 to G, with G being the number of points in a grid of values for μ_j. The total number of basis functions is $M = 1 + 2G$. We will use values of s for which $1/s$ is an integer, with $G = (1/s) + 5$. The μ_j values will be $-2s$, $-s$, 0, s, $\ldots$, $1 - s$, 1, $1 + s$, $1 + 2s$, or in other words, $\mu_j = (j - 3)s$ for $j = 1, \ldots, G$.

For the model with interactions, the basis functions other than ϕ_0 will have the form

$$\phi_j(x) = \exp(-((x_1 - \nu_{j1})^2 + (x_2 - \nu_{j2})^2)/(2s^2))$$

where j ranges from 1 to G^2. For these values of j, ν_{j1} and ν_{j2} take on all possible combinations of values from $\{-2s, -s, 0, s, \ldots, 1 - s, 1, 1 + s, 1 + 2s\}$.

You should fit both of these linear basis function models using both of the two approaches discussed in lectures — maximum penalized likelihood with choice of penalty factor and other choices made to minimize squared error from 5-fold cross validation, and Bayesian inference with choice of noise variance, prior standard deviation, and other choices made by maximizing marginal likelihood.

For the penalized likelihood approach, the penalty added to the log likelihood should have the form $\lambda \sum_{j=1}^{M-1} w_j^2$. Note that w_0 is not included in the penalty. You will use 5-fold cross validation to choose between additive and interactive models, to choose a value for s, and to choose a value for λ. Split the data into five parts in the obvious way without randomizing order (the data is already in random order), so that your split will be the same as other students (this makes marking easier).

For s, you should consider the values 0.1, 0.2, and 0.5. For λ, you should consider the values 0.002, 0.008, 0.032, 0.128, and 0.512.

For the Bayesian approach, the regression coefficients will be independent in the prior, with the prior for w_0 being $N(0, 10^2)$, and the priors for w_j with $j = 1, \ldots, M-1$ being $N(0, \omega^2)$. The the choice between additive and interactive models and the choice of values for ω, σ, and s is done by maximizing the marginal likeihood. (You should actually work in terms of the log of the marginal likelihood, to avoid possible overflow problems.)

For s, you should consider the values 0.1, 0.2, and 0.5. For ω, you should consider the values 0.2, 0.4, 0.8, 1.6, and 3.2. For σ, you should consider the values 0.04, 0.08, 0.16, 0.32, and 0.64.

For each dataset, and for both approaches, you should identify the single combination of additive/interactive model, s, and other choices that seems best. Note that the best choices may be different for the two datasets. You should then make predictions for the test cases with these choices, and see what the average squared error is with these predictions. This will give four final numbers — average squared error with penalized likelihood vs. Bayesian methods on dataset 1 and on dataset 2.

Finally, you should discuss these results, not just in terms of these four figures for squared error on the test set, but also referring to other things, such as (for example) how much the cross validation results varied with different choices.