Math3349 – Data Mining: Assignment 1

August 22, 2012

Marks will be given for layout and for clarity of exposition. Please email your results, in the form of a pdf file, to `john.maindonald@anu.edu.au`, by 5pm on Monday Sept 10.

This assignment will work with the data, on heavy metal concentrations in the vicinity of the river Meuse, that was considered in Subsection 4.4.1 of the notes. The termplots in that section suggested nonlinear effects for both `elev` and `dist`.

Here, we will in due course use smooth curves to model their effects. Run the code:

```
library(sp)
data(meuse)
meuse$ffreq <- factor(meuse$ffreq)
meuse$soil <- factor(meuse$soil)
```

Exercises

1. Given the model

$$\mathbf{y} = \mathbf{X}\,\beta + \epsilon \tag{1}$$

where the elements ϵ_i of ϵ are independent with variance σ^2, it can be shown that if $\mathbf{X}$ has p linearly independent columns, then:

$$\mathrm{E}[(\widehat{\mathbf{y}} - \mu)^T\,(\widehat{\mathbf{y}} - \mu)] = p\,\sigma^2$$

Now suppose training and test data sets, each of n observations, are drawn from the same population. Thus we have:

Training : $\mathbf{X}^{(1)}$ and $\mathbf{y}^{(1)}$; estimate $\beta^{(1)}$ of β

Test : $\mathbf{X}^{(2)}$ and $\mathbf{y}^{(2)}$; prediction $\widehat{\mathbf{y}^{te}}$ from training data

   (i) Explain why the elements of $\mathbf{y}^{(2)}$ are independent of $\widehat{\mathbf{y}^{te}}$

   (ii) The error mean square when the model that is derived using the training data is applied to the test data is:

$$(\mathbf{y}^{(2)} - \widehat{\mathbf{y}^{te}})^T\,(\mathbf{y}^{(2)} - \widehat{\mathbf{y}^{te}})\,/\,n$$

Show that this has expected value $(n + p)\,\sigma^2$

[Write

$$\mathbf{y}^{(2)} - \widehat{\mathbf{y}^{te}} = \mathbf{y}^{(2)} - \mu^{(2)} - (\widehat{\mathbf{y}^{te}} - \mu^{(2)})$$

]

[Thus, the mse for the test data is an inflated estimate of σ^2]

2. In the following, we simulate 1000 sets of observations from a specified model that has main effects only, and compare the model fit for main effects only (the 'true' model) with the model that fits a surface.

```
library(mgcv)
deltaDf <- F <- pval <- scale1 <- scale2 <- numeric(1000)
zSim <- with(meuse, 5 - 2*elev -3.3*dist + 1.6*dist^2 +
                 rnorm(nrow(meuse), sd=sqrt(0.16)))
zed.lm <- lm(zSim ~ elev + poly(dist,2) , data=meuse)
simY <- simulate(zed.lm, nsim=1000)
for(i in 1:1000){
   if((i%%250)==1)cat("\n")
   if((i%%20)==0)cat(i, " ")
z <- simY[,i]
z.gam <- gam(z ~ s(elev) + s(dist) , data=meuse, method="ML")
z2.gam <- gam(z ~ s(elev, dist), data=meuse, method="ML")
scale1[i] <- z.gam$sig2
scale2[i] <- z2.gam$sig2
aovtest <- anova(z.gam, z2.gam, test="F")
deltaDf[i] <- aovtest[2, "Df"]
F[i] <- aovtest[2, "Df"]
pval[i] <- aovtest[2, "Pr(>F)"]
}
cat("\n")
sum(!is/na(pval))
plot(pval ~ deltaDf, log="xy")
```

Notes:

a. A desirable preliminary step is to start by running a small number of simulations. For example, run the code with `for(i in 1:1000)` replaced by `for(i in 1:20)`. This will allow an estimate of the time that 1000 simulations are likely to take on your system.

b. Note that the analysis of variance comparison yields meaningless results when the difference in degrees of freedom for the two model fits is

very small. The uncertainty in the degrees of freedom is then of comparable magnitude, or larger, than the degrees of freedom estimate. The F-statistic is meaningless.

(ii) Annotate the code, explaining briefly what each line does.

(ii) Which is closest to the null model – `z.gam` or `z2.gam`?

(iii) Plot `pval` against `deltaDf`, using a logarithmic scale for both axes (specify `log="xy"`). The points in the graph are likely to separate into two clusters. Which are the points for which the p-value is meaningful? What does the graph suggest might be a reasonable cut-off value for `deltaDf` to be meaningful? Draw a line or lines across the graph to mark off the points that correspond to a p-value that should be regarded as significant at the 5% level.

3. Now move to work with the outcome values in the `meuse` dataset:

(i) Examine fits to the outcome values in the `meuse` dataset. Fit and compare the following models:

```
m.gam <- gam(log(lead) ~ s(elev) + s(dist) + ffreq + soil,
             data=meuse, method="ML")
mx.gam <- gam(log(lead) ~ s(elev, dist) + ffreq + soil,
              data=meuse, method="ML")
mxx.gam <- gam(log(lead) ~ s(elev, dist, by=ffreq) + ffreq + soil,
               data=meuse, method="ML")
(aovcomp <- anova(m.gam, mx.gam, mxx.gam, test='F'))
obsF <- aovcomp["F"][2,]
  ## Compares mx.gam with m.gam, and mxx.gam with mx.gam
```

Do you notice a problem for the comparison between mxx.gam and mx.gam?

(ii) Now do a simulation to check the comparison between the mx.gam model when errors are iid normal.

```
hat <- predict(m.gam)
simY <- simulate(m.gam, nsim=1000)
scale1 <- scale2 <- f1000 <- p1000 <- deltaDf <- numeric(1000)
for(i in 1:1000){
   if(i%%20==0)cat(i)
   ms.gam <- gam(simY[,i] ~ s(elev) + s(dist) + ffreq + soil,
                 data=meuse, method="ML")
   mxs.gam <- gam(simY[,i] ~ s(elev, dist) + ffreq + soil,
```

```
                    data=meuse, method="ML")
    aovcomp <- anova(ms.gam, mxs.gam, test="F")
    f1000[i] <- aovcomp["F"][2,]
    p1000[i] <- aovcomp["Pr(>F)"][2,]
    deltaDf[i] <- aovcomp[2, "Df"]
    scale1[i] <- m.gam$sig2
    scale2[i] <- mx.gam$sig2
}
## Now refer the observed F to this null sampling distribution
## NAs will mostly indicate no real difference:
##   a -ve deviance change or a -ve df change
plot(p1000 ~ deltaDf, log="xy")
## Modify the following to require some minimum value for \texttt{delta
## print(sum(!is.na(f1000) & f1000>obsF)/1000)
```

From the plot, decide on a reasonable cutoff value for `deltaDf`.

(iii) Compare also `mean(scale1)` and `mean(scale2)` with the scale estimate from `m.gam`. What does this suggest?

4. Another approach is to get cross-validation estimates of the mean square error for each of the two models, and compare these estimates. We can use the function `CVgam()` from the package *modregR* for this purpose. The same set of cross-validation folds should be used for the two models, in order to get an accurate comparison:

```
library(modregR)
foldnum <- sample(1:10, size=nrow(meuse), replace=TRUE)
CVgam(log(lead) ~ s(elev) + s(dist) + ffreq + soil, data=meuse,
      method="ML", cvparts=foldnum)
CVgam(log(lead) ~ s(elev, dist) + ffreq + soil, data=meuse,
      method="ML", cvparts=foldnum)
```

Repeat this comparison several times. Comment on the comparison between the scale estimates from the GAM model. Do the simulation results in Exercise 3(i) (the means of the respective scale parameter estimates) have useful light to shed on what you observe here? Do these results from cross-validation seem broadly consistent with the simulation results?

Note: Include the R code that you have used in an appendix to your assignment. Be careful to identify the question for which the code has been used.