

# Regression Course – Additional Notes I & II

John Maindonald

June 13, 2007

These consolidate the earlier separate sets of additional notes – I & II

## 1 Balanced vs Unbalanced Data – What is the Effect?

This section illustrates, in a very simple case, confounding effects that may arise when data are unbalanced. It demonstrates, also, how the estimate for a term may change when a further term is included in the model.

### 1.1 Cricket bowling averages example

In a game of cricket the bowler's aim is to give away as few runs as possible, while taking the maximum possible number of wickets. (A wicket is taken when a batsman is given out). The game is divided into two innings, and the conditions can be very different between the two innings. Here the interest is in comparing the performances of different bowlers.

Bowler A			Bowler B		
1st Innings	2nd Innings	Overall	1st Innings	2nd Innings	Overall
50 runs	70 runs	<b>120 runs</b>	240 runs	40 runs	<b>280 runs</b>
1 w	5 w	<b>6 w</b>	6 w	4 w	<b>10 w</b>
Runs per wicket					
50 r/w	14 r/w	<b>20 r/w</b>	40 r/w	10 r/w	<b>28 r/w</b>

Thus Bowler A does better than Bowler B in both innings (gives away fewer runs for each wicket taken), but ends up giving away more runs per wicket overall. Observe that Bowler A did more of the bowling in the first innings, when there were more runs per wicket. Thus Bowler A's runs per wicket average is skewed towards the first innings runs per wicket rate.

Here is the 2 by 2 table of runs per wicket information. Adding the total of runs for each bowler, and dividing in each case by the corresponding total of wickets is equivalent to taking averages for each bowler of runs per wicket, with number of wickets as weight.

	Innings 1	Innings 2	Average
A	50 (1 w)	14 (5 w)	
B	40 (6 w)	10 (4 w)	$25 = 32 - 7$
Average	45	$12 = 45 - 33$	28.5

The default contrasts used by the function `lm()` take the baseline for all effects (the "intercept") as the outcome for Innings 1 and Bowler A. With innings 1 is taken as the baseline, the innings effect (Innings 2 - Innings 1) appears as -33, as in the above table. With Bowler A as the baseline, the bowler effect (Bowler B - Bowler A) appears as -7.

## 1.2 Analysis Using `lm()`

```
> cricket <- data.frame(Innings=factor(rep(c("Inn1","Inn2"),2)),
+                       Bowler=factor(rep(c("A","B"), c(2,2))))
> cricket$rw <- c(50,14,40,10)
> cricket$w <- c(1,5,6,4)
> cricket
  Innings Bowler rw w
1   Inn1      A  50 1
2   Inn2      A  14 5
3   Inn1      B  40 6
4   Inn2      B  10 4

> cricket.lm <- lm(rw ~ Innings+Bowler, data=cricket)
> round(coef(cricket.lm), 2)
(Intercept) InningsInn2      BowlerB
      48.5         -33.0         -7.0
```

Now omit consideration of the innings effect:

```
> cricket1.lm <- lm(rw ~ Bowler, data=cricket)
> round(coef(cricket1.lm), 2)
(Intercept)      BowlerB
          32           -7
```

Observe that the estimate of the bowler effect is unchanged.

### Weighted analysis

```
> cricket.wlm <- lm(rw ~ Bowler+innings, weight=w, data=cricket)
> round(coef(cricket.wlm), 2)
(Intercept)      BowlerB InningsInn2
      46.29         -5.67      -31.55
```

Observe that the estimate of the bowler effect has changed, but is still fair to Bowler B.

Now omit consideration of the innings effect:

```
> cricket1.wlm <- lm(rw ~ Bowler, weight=w, data=cricket)
> round(coef(cricket1.wlm), 2)
(Intercept)      BowlerB
          20           8
```

This is equivalent to dividing the overall number of runs, for each bowler, by the overall number of wickets. As above, Bowler B is estimated as costing 8 runs per wicket more than Bowler A. The combination of unequal weights and omission of a relevant factor generates this misleading result.

## 1.3 Calculations using Householder reflections

First, form the matrix on which the calculations will be performed.

```
> Xy <- cbind(model.matrix(~ Bowler + Innings, data=cricket),
+             RunsPerW = cricket$rw)
> Xy
(Intercept) BowlerB InningsInn2 RunsPerW
1           1         0           0       50
2           1         0           1       14
```

```
3      1      1      0      40
4      1      1      1      10
```

Now use my function `house()` for the calculation.

```
> house(Xy, showzeros=3)
(Intercept) BowlerB InningsInn2 RunsPerW
1           2         1           1        57
2           0         1           0        -7
3           0         0           1       -33
4           0         0           0         3
```

To obtain estimates of the model parameters, solve:

$$\mathbf{X} = \begin{bmatrix} 2 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{b} = \begin{bmatrix} 57 \\ -7 \\ -33 \end{bmatrix}$$

Here are the calculations for the weighted analysis:

```
> house(Xy*sqrt(cricket$w), showzeros=3)
(Intercept) BowlerB InningsInn2 RunsPerW
1           4 2.500000  2.2500000 100.000000
2           0 1.936492 -0.8391464  15.491933
3           0 0.000000  1.7981472 -56.725056
4           0 0.000000  0.0000000  -4.718903
```

Now solve:

$$\mathbf{X} = \begin{bmatrix} 4 & 2.500000 & 2.2500000 \\ 0 & 1.936492 & -0.8391464 \\ 0 & 0 & 1.7981472 \end{bmatrix} \mathbf{b} = \begin{bmatrix} 100.000000 \\ 15.491933 \\ -56.725056 \end{bmatrix}$$

A key difference is that the R matrix no longer has a zero in the (2, 3) position. The consequence is that the estimate of the bowler effect changes (and changes in sign) if there is no allowance for Innings.

## 1.4 A further example

The table shows, separately for males and females, the effect of pentazocine on post-operative pain profiles (average VAS scores), with (mbac and fbac) and without (mpl and fpl) preoperatively administered baclofen. Pain scores are recorded every 20 minutes, from 10 minutes to 170 minutes. Results are shown for 30 minutes and 50 minutes only.

15 females were given baclofen, as against 3 males. 7 females received the placebo, as against 16 males. Averages for the two treatments (baclofen/placebo), taken over all trial participants and ignoring sex, are misleading. Notice that, both for males and for females, the scores are lower when baclofen is

Time (min)	mbac	mpl	fbac	fpl	avbac	avplac
30	1.31	1.65	3.48	4.15	3.12	2.74
50	0.05	0.67	3.13	3.66	2.61	1.98

administered. Reliance on the overall average would suggest that the scores are higher when baclofen is administered.

## 2 Bootstrap & Permutation Resampling

Here our interest will be in comparing residual sums of squares for models that are not nested. The methodology will be demonstrated using the `fruitohms` data in the `DAAG` package.

## 2.1 Strategy

For this initial discussion, denote the models as M1 and M2. We first fit the model M1 to the original data, then obtaining fitted values and residuals. Also, we fit M2, and note the change as measured by a statistic  $\tilde{F}_{12}$  that is calculated just as for the usual  $F$ -statistic.

If M2 is not giving an improvement, then  $\tilde{F}_{12}$  will differ only by statistical error from the  $F_{12}$  observed when a random set of residuals are added. It will, in effect, be a random sample from the distribution obtained when repeated random sets of residuals are added on to the fitted values, and  $F_{12}$  is calculated for each such new set of “observations”. We can then locate  $\tilde{F}_{12}$  within this distribution. If  $\tilde{F}_{12}$  falls above its 95th percentile then we will judge that, at the 5% significance level, M2 improves on M1. The change when the residuals are correctly attached is, on this criterion, more than statistical variation.

With slight modification, the argument applies if the residuals are obtained by permuting the original residuals. If model M2 is not improving on M1, then the change when new “observations” are derived by adding randomly permuted residuals back onto the fitted values will be attributable to statistical variation.

It also applies, again in slightly modified form, if repeated bootstrap samples of the original residuals are used. Note that we are not, here, deriving bootstrap estimates of some parameter. While there are theoretical issues even with this use of the bootstrap, we do not have the issues of bias that commonly arise when the bootstrap is used to estimate variance-like statistics.

Note that the three distributions contemplated above are different. There will, accordingly, be differences of power between the three approaches.

The methodology can be varied or extended in various ways:

- We can choose M2 if it appears, on average, to improve on M1, i.e., choose M2 if the  $p$ -value is less than 0.5.
- This same style of approach can be used if M2 is selected from a wider class of models. The model selection step must then be repeated for each new bootstrap or permutation sample.

## 2.2 Code

Here are functions that can handle the calculations:

```

'funF' <-
function(mmat1, mmat2, y,
        showstats=FALSE, divide=10^6){
  ## mmat1 & mmat2 must be model matrices
  M1 <- mmat1
  M2 <- mmat2
  n <- dim(M1)[1]
  qrM1 <- qr(M1)
  qrM2 <- qr(M2)
  ss <- c(sum(qr.resid(qrM1, y)^2),
        sum(qr.resid(qrM2, y)^2))
  df <- dim(mmat1)[1] - c(qrM1$rank, qrM2$rank)
  ssd <- ss[1] - ss[2]
  F12 <- ssd/ss[2]*df[2]    # F-like statistic
  if(showstats){
    print(paste("Estimates of sigma^2 (Xply by ", divide, ")", sep=""))
    names(ss) <- paste(df, "df", sep="")
    print(ss/df/divide)
    print(paste("'F-statistic' = ", round(F12,4),
                " (df=", df[1]-df[2], " & ", df[2],")", sep=""))
  }
}

```

```
invisible(F12)
}
```

First try running `funF()` with the initial data:

```
> M1.lm <- lm(ohms ~ poly(juice, 3), data=fruitohms)
> M2.lm <- lm(ohms ~ ns(juice, 4), data=fruitohms)
> M1 <- model.matrix(M1.lm)
> M2 <- model.matrix(M2.lm)
> funF(M1, M2, y=fruitohms$ohms, showstats=TRUE)
[1] "Estimates of sigma^2 (Xply by 1e+06)"
    124df    123df
0.9723468 0.9343716
[1] "'F-statistic' = 6.0397 (df=1 & 123)"
```

The function `bootF` now shown can be used for the calculations. The default is to use bootstrap samples of the residuals (`type="ordinary"`). Other options are `type="permutation"` and `type="parametric"`.

```
'bootF' <-
function(data=fruitohms, statistic=funF, R=999,
         form1= ohms ~ poly(juice, 3), form2= ohms ~ ns(juice,4),
         type=c("ordinary", "parametric", "permutation")){
  ## By default, type[1]="ordinary" is used.
  ## Alternatives are type="parametric" or type="permutation"
  M1mod <- lm(form1, data=data)
  sigma <- summary(M1mod)$sigma
  n <- dim(data)[1]
  M1 <- model.matrix(M1mod)
  M2 <- model.matrix(form2, data=data)
  M1fit <- fitted(M1mod)
  M1resid <- resid(M1mod)
  R2 <- R+1
  y <- M1fit+M1resid
  boot.out <- numeric(R2)
  boot.out[1] <- funF(mmat1=M1, mmat2=M2, y=y, showstats=TRUE)
  for(i in 2:R2){
    if(type[1] == "permutation")
      index <- resid <- M1resid[sample(1:n)] else
    if(type[1] ==
       "ordinary") resid <- M1resid[sample(1:n, replace=TRUE)] else
    if(type[1] == "parametric") resid <- rnorm(n, sd=sigma)
    y <- M1fit+resid
    boot.out[i] <- statistic(mmat1=M1, mmat2=M2, y=y)
  }
  testval <- boot.out[1]
  pval <- sum(boot.out >= testval)/R2
  print(c("p-value" = round(pval,5)))
  invisible(boot.out)
}
```

### 2.3 Bootstrap samples of the residuals

We will now test the use of ordinary bootstrap samples in a situation where we pretty much know the answer. For this purpose, we take the models to be `ohms$poly(juice,2)` and `ohms$poly(juice,3)`, so that the models are nested.

```
> poly45stats <- bootF(form1=ohms ~ poly(juice,4), form2 = ohms ~ poly(juice,5))
[1] "Estimates of sigma^2 (Xply by 1e+06)"
      123df      122df
0.8878564 0.8859704
[1] "'F-statistic' = 1.2618 (df=1 & 122)"
p-value
 0.263
```

Here, this statistic is an analysis of variance  $F$ -statistic. Thus, we may, provided iid normality assumptions are acceptable, refer it to the relevant theoretical  $F$ -distribution. This gives a  $p$ -value that is essentially the same as the bootstrap distribution  $p$ -value.

```
> 1-pf(1.2618, 1,122)
[1] 0.2635162
```

Saving the values in `poly45stats` allows us, if we wish, to examine other percentiles of the bootstrap distribution.

For the comparison between `poly(juice,3)` and `ns(juice, 4)` we have:

```
> poly3ns4ord <- bootF(form1=ohms ~ poly(juice,3), form2 = ohms ~ ns(juice, 4))
[1] "Estimates of sigma^2 (Xply by 1e+06)"
      124df      123df
0.9723468 0.9343716
[1] "'F-statistic' = 6.0397 (df=1 & 123)"
p-value
 0.015
```

## 2.4 The parametric bootstrap

This is not, strictly, a bootstrap. Rather it is a simulation that is based on a theoretical distribution. Here, the residuals are sampled from the theoretical normal, with the standard deviation taken to be the square root of the mean residual sum of squares from fitting the model M1.

```
> poly3ns4sim <- bootF(form1=ohms ~ poly(juice,3),
+                      form2 = ohms ~ ns(juice, 4), type="parametric")
[1] "Estimates of sigma^2 (Xply by 1e+06)"
      124df      123df
0.9723468 0.9343716
[1] "'F-statistic' = 6.0397 (df=1 & 123)"
p-value
 0.011
```

## 2.5 The permutation distribution

```
> poly3ns4perm <- bootF(form1=ohms ~ poly(juice,3),
+                      form2 = ohms ~ ns(juice, 4), type="permutation")
[1] "Estimates of sigma^2 (Xply by 1e+06)"
      124df      123df
0.9723468 0.9343716
[1] "'F-statistic' = 6.0397 (df=1 & 123)"
p-value
 0.021
```

The permutation distribution is widely useful in contexts where the asymptotic or other theory is in doubt, and where the null hypothesis implies that permuting the  $y$ -values, or permuting the values of

an explanatory variable, should on average not affect the model's fitted values. It can be useful in the fitting of logistic regression models.

### 3 Logistic Regression vs Multi-way Tabulation

Models for multi-way tables that allow or all interactions, at all levels, are said to be “saturated”. Fitted values from a logistic regression, with a model that is thus “saturated”, equal the probabilities that may alternatively be derived from the equivalent multi-way table. Code will be given that may be used to verify the equivalence of the multi-way table to fitted values from a logistic model. (Actually, it is for this purpose immaterial what link is used with the binomial or quasibinomial model.)

#### 3.1 US Accident Mortality Data

Data, in the data frame `nassCDS` in the `DAAGxtras` package, were collected according to a sampling design where different cells had different weights. Whether using the logistic regression or tabulating the proportions, it is necessary to take account of the weights.

The data have had a central role in a controversy, debated in three articles in the journal *Chance*, on the effectiveness of airbags. References, both to these articles and to relevant web pages, are given on the help page for `nassCDS`.

Various biases may affect the result. There are alternatives to the style of analysis discussed here. Farmer (2006) discusses an approach that is preferred by the National Highway and Traffic Safety Administration (NHTSA), and which gives an answer that is favourable to the use of airbags.

#### Strategy issues

Notwithstanding care to consider all relevant effects, it remains possible that there will be relevant factors of interactions that have not been considered. The relevant information may not be included in the data. A useful strategy, with data such as these, may be:

- Account first for those effects that on prior grounds (relevant science, previous experience with related data), seem certain to have a role. Such arguments justify use, for the present data, of the factors `seatbelt`, `airbag` and `dvcat`.
- Investigate addition of other possible effects one at a time.

Stepwise and best subsets automatic variable selection procedures have a much more limited usefulness than older textbooks on regression may have suggested. Cross-validation or bootstrap approaches should be used to check out the stability of the selection with respect to statistical variation. Note that the use of automatic selection procedures invalidates, in general, estimates that classical linear model theory gives for the standard errors of parameters. Bootstrap estimates of the standard errors, perhaps obtained along with the procedure used to check out the stability of the selection, may provide a workaround.

#### 3.2 Factors that affect mortality

The analysis here will be limited to the factors `seatbelt` and `airbag`, leaving as an exercise extension to account for the force of impact measure (`dvcat`). Such a more extended analysis makes it clear that any defensible analysis in the style of the analysis discussed here must, as a minimum, include these three factors and their interactions.

Almost certainly, there are other factors, not considered in any of the analyses presented in the *Chance* articles, that have affected results. A more complete analysis will require consideration of further possible effects for which data are available. If more than one or two of those factors are included there is a risk, even with this relatively large data set, that it will become impossible to distinguish the likely effects of airbags from those of other factors.

Here is the code for the limited (and misleading!) tabulations presented here:

```
tot <- xtabs(weight ~ seatbelt+airbag, data=nassCDS)
dead <- xtabs(I(weight*(unclass(dead)-1)) ~ seatbelt+airbag, data=nassCDS)
```

Here is code for the glm analysis:

```
nass.glm <- glm(dead ~ seatbelt*airbag, weight=weight, family=binomial,
               data=nassCDS)
## Reconcile with tabulated result
df <- with(nassCDS, expand.grid(seatbelt=factor(levels(seatbelt)),
                              levels=levels(seatbelt)),
          airbag=factor(levels(airbag)),
          levels=levels(airbag)))
df$tot <- as.vector(xtabs(weight ~ seatbelt+airbag, data=nassCDS))
nasshat <- predict(nass.glm, newdata=df, type="response", se=TRUE)
df$estdead <- nasshat$fit*df$tot
xtabs(tot ~ seatbelt+airbag, data=df)      # Table of totals
xtabs(estdead ~ seatbelt+airbag, data=df)  # Table of dead
```

Bootstrap estimates of the excess risk from airbags can be obtained thus:

```
xtra <- matrix(0, nrow=2, ncol=1000)
nass <- nassCDS[nassCDS$weight>0,]
prob=with(nass, weight/sum(weight))
for(i in 1:1000){
  nrows <- sample(1:dim(nass)[1], prob=prob, replace=TRUE)
  xtra[,i] <- excessRisk(form = weight ~ seatbelt + airbag,
                        data=nass[nrows, ])[, 8]
}
```

Calculations can take a long time. Run this to calculate 10 or 100 bootstrap samples before running it for the full 1000 samples. Percentile estimates of the confidence limits may in this instance be satisfactory.

## 4 Propensities

Propensities are one of a number of devices that may be used in the attempt to reduce the number of explanatory variables that need to be considered in a regression. The method is intended for a very specific, but important, context where there are two (or, potentially, more) levels of a treatment factor. The aim is to investigate treatment effects, after adjustment for covariate effects.

The attempt to adjust for multiple potential covariate effects has a variety of complications. The correct functional form must be used – it may not be adequate to assume additive linear effects, even after transformation of covariates in cases where this seems desirable. Diagnostic checking may be difficult; failure to account adequately for the effect of one or more variable may lead to misleading diagnostics for other variables.

The derivation and use of propensity scores can simplify the model fitting process. The complications that arise from the attempt to adjust for multiple covariates are limited to the modeling used to predict the propensity scores. Having derived propensity scores, the regression model that incorporates the treatment effect has two terms – a treatment effect, and a single covariate adjustment term.

### 4.1 What is a propensity?

Discussion will be limited to the case where there is a binary treatment assignment. For the purposes of the data that will be studied here, this will be a control and a treatment. A propensity is the conditional

probability  $\lambda(\mathbf{x})$  of assignment to a particular treatment given a vector of observed covariates  $\mathbf{x}$ . The methodology requires that treatment assignment should be ignorable given the propensity, i.e., treatment assignment should be unrelated to potential outcomes within strata defined by  $\lambda(\mathbf{x})$ . Conditional on the propensity score, the distributions of the observed covariates are independent of the binary treatment assignment.<sup>1</sup> This allows use of the propensity score as a balancing score. See Rosenbaum (2002, pp.296-297), perhaps supplemented by Rosenbaum (1999) and Rosenbaum and Rubin (1983), for a discussion.

The propensity score, or a monotone function of the score, can be estimated using discriminant analysis methodology, independently of the outcome  $y_i$ . The regression equation becomes

$$y_i = t_i + \beta\phi(\lambda_i) + \epsilon_i$$

where the functional form of  $\phi()$  has to be estimated or guessed. Scores from use of the logit transformation are often used as a starting point.

Compare this with the use of regression adjustments of the form

$$y_i = t_i + f(x_1, x_2, \dots, x_k) \quad (1)$$

where in the simplest situation it might be hoped that

$$f(x_1, x_2, \dots, x_k) = a_1x_1 + a_2x_2 + \dots + a_kx_k$$

This requires the stronger condition that treatment assignment should be ignorable given the observed covariates  $\mathbf{x}$ . i.e., treatment assignment should be unrelated to potential outcomes within strata defined by  $\mathbf{x}$ .

The propensity score approach reduces the regression equation that is of primary interest to a simple form. Decisions on which variables and interactions to include, and on transformation and/or modeling using spline terms where this seems required, is relegated to the earlier discriminant function calculations. Diagnostics for the model for  $y_i$  need be studied for one covariate only.

## Scores calculated using a linear discriminant

In Maindonald & Braun (2007, pp.412-419), we discuss the use, as propensity scores, of functions  $f(x_1, x_2, \dots, x_k)$  that are linear in covariates  $x_1, x_2, \dots, x_k$ . This may be too restrictive. Even after use of spline terms (how many d.f.? what interactions, if any, should be included?) the model may be unable to capture well the nuances of the regression dependence in cases where there are more than one or two explanatory variables.

Here is code for the calculations:

```
common <- multilap(maxf=30) # Mild preliminary filtering on
                          # variables educ, re74 and re75
## Calculate propensity scores: data frame nsw74psidA (DAAG)
disc.glm <- glm(formula = trt ~ age + educ + black + hisp + marr +
                re74 + re75, family = binomial,
                data = nsw74psid1, subset=common)
Pscores <- predict(disc.glm)
## Now filter further, based on values of Pscores
xchop <- with(subset(nsw74psid1, common),
              overlapDensity(Pscores[trt==0], Pscores[trt==1],
                             compare.numbers=FALSE,
                             ratio=c(1/30, 30)))

overlap <- common
overlap[common] <- Pscores > xchop[1] & Pscores < xchop[2]
nsw74psidC <- subset(nsw74psid1, overlap)
Pscores <- Pscores[Pscores > xchop[1] & Pscores < xchop[2]]
```

<sup>1</sup>The ignorability assumption seems to me implausible for the present data.

The hope is that, conditional on values of Pcores, controls and treated are now relatively similar with respect to the various covariates. This can be checked directly, by splitting the data set up in to, e.g., 5 parts, based on values of Pcores.

```
cut5 <- cut(Pcores, breaks=5)
for (cutlev in levels(cut5)){
  print(cutlev)
  nsw74 <- subset(nsw74psidC, cutlev==cut5)
  print(
  sapply(nsw74[, c("black","hisp","marr","nodeg")],
        function(x){tab <- table(nsw74[, "trt"], x)
                        tab[1,]/apply(tab, 2, sum)})
  )
}
```

The balancing is, in most cases, reasonable. There is however a big disparity in the numbers of hispanics in one of the categories, and none of the treated group in the final category were married. Similar comparisons can be done for the continuous variables.

Now try fitting the models:

```
nsw.lm <- lm(log(re78+25) ~ trt + propensity, data=nsw74psidC)
nsw.glm <- glm(I(re78>0) ~ trt + propensity,
              family=binomial, data=nsw74psidC)
```

## 5 Scores Calculated from randomForest Analysis

This is at the other extreme, relative to linear models. These models builds in as little structure as possible. There is no insistence on continuous forms of dependence on continuous variables. Remarkably as it seems to me, these models can, in some classification problems, do very well. It may be that the loss from ignoring of obvious structure is more than compensated by the ability to handle complex interactions and non-linear responses.

The following is exploratory. As a technique for revealing structure in data, it can work well. Its use for deriving propensity scores requires futher exploration and theoretical elucidation.

Here, we use the larger data set in which not all observations have information on re75. We therefore omit this variable from consideration.

```
nsw <- rbind(psid1, nswdemo[nswdemo$trt==1,])
nsw$trt <- factor(nsw$trt)
nswx <- nsw[, c(1:7,9)]
nsw.rf <- randomForest(trt ~ ., data=nswx, proximity=TRUE)
distmat <- 1-nsw.rf$proximity
distmat[distmat==0] <- 0.001 # Half minimum of non-zero distances
## Apply arcsine transformation to stretch the scale out at both ends
distmat <- asin(distmat)
## Start with classical multi-dimensional scaling (Euclidean distances)
nsw.cmd <- cmdscale(distmat)
plot(nsw.cmd, col=unclass(nsw$trt))
## Apply Sammon (semi-metric) scaling
nsw.sam <- sammon(distmat, nsw.cmd)
plot(nsw.sam$points, col=unclass(nsw$trt))
```

The plot suggests that the two groups are not well matched. There may however be subgroups that overlap substantially. One might try to separate out those regions of the plot where both controls and (especially, as there are many fewer of these) treatment observations are reasonably well represented.

A more direct approach is to use the predict method for `randomForest` objects to return a matrix of class probabilities, with one row for each data point.

```
## Take the second column; the first would do equally well.
prob <- predict(nsw.rf, type="prob")[,2]
prob[prob==0] <- 0.5*min(prob[prob>0])
prob[prob==1] <- 1-0.5*min(1-prob[prob<1])
scores <- log(prob/(1-prob))
```

Now find the range of values of scores where the ratios of the densities are in the range (0.025, 40), and try the regressions with attention limited to data where the scores are in that range:

```
z <- with(nsw, overlapDensity(ratio=c(.025,40),
                             scores[trt==0], scores[trt==1]))
retain <- scores>z[1] & scores<z[2]
nsw.lm <- lm(log(re78+25) ~ trt + scores, data=nsw, subset=retain)
termpplot(nsw.lm, par=T, smooth=panel.smooth)
nsw.glm <- glm(I(re78>0) ~ trt + scores,
               family=binomial, data=nsw, subset=retain)
```

The results depend on the chosen range of values of the scores. The estimates from `lm()` do not reproduce the results from the experimental comparison; in fact they suggest a negative effect from training. The estimates from `glm()` do favour the treatment group, providing the range of ratios is set small enough. The difference does not, however, reach the 5% level of statistical significance.

## 6 Errors in Variables – Further Notes

The model in Section 8.1 is

$$y = \alpha + \beta x + \epsilon$$

We measure, not  $x$ , but  $w = x + u$ , where  $u$  is “measurement error”.

Now assume that  $w$  is unbiased for  $x$  and that  $u$  is independent of  $x$  and  $\epsilon$ .

Then, conditional on  $x$ , instead of

$$\hat{\beta} = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sum (x - \bar{x})^2}$$

we have

$$\beta^* = \frac{\sum (w - \bar{w})(y - \bar{y})}{\sum (w - \bar{w})^2}$$

Then

$$E[\sum (w - \bar{w})(y - \bar{y})] = \sum (x - \bar{x})(y - \bar{y})$$

This happens because  $y$  is independent of  $u$ .

$$\begin{aligned} E[\sum (w - \bar{w})^2] &= E[\sum (x - \bar{x} + u - \bar{u})^2] \\ &= \sum (x - \bar{x})^2 + \sum (u - \bar{u})^2 \\ &= (n-1)\sigma_x^2 + (n-1)\sigma_u^2 \end{aligned}$$

Then  $\beta^*$  is a consistent estimate of  $\lambda\beta$ , where

$$\lambda = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_u^2}$$

## Simulation

Use the function `g6.17`, included in the image file `figs6.RData` that is available from:  
<http://www.maths.anu.edu.au/~johnm/r-book/2edn/figures/>

### 6.1 One covariate measured without error

The function `errorsINxGP()`, available from <http://www.maths.anu.edu.au/~johnm/r/functions/>, simulates the effect when the variables that are measured without error code for a categorical effect.

## 7 Variable Selection

In addition to the notes in Maindonald & Braun (2007, p.186), note the following:

- Consider data where one variable, or a small number of variables jointly, have effects that, in the preferred model, are large relative to statistical error, while other variables have effects that, at best, are marginally detectable. Then classical selection techniques (stepwise regression, etc.) are likely to find those variables that have large effects, and their coefficients will be estimated without selection bias.
- In contexts where automatic selection techniques are tested more severely, they may not do much better than chance.
- To see the potential, with automatic selection algorithms, to get highly significant effects from random data, run the function `bestset.noise()`, from the *DAAG* package.
- There may be no unique “best” set of explanatory variables.
- The paper by Zhu and Chipman (2006) is interesting. The key here seems to be the incorporation of a random element. I suspect that a bootstrap approach, used in a similar way as in the random forests algorithm, would do as well or better.
- The selection problem is fraught with further hazards when one or more of the variables is measured with substantial error.

Attempts to interpret regression coefficients raise further hazards. Conditions that may make coefficients interpretable include: a) It is possible to identify a few variables that have large effects; b) the data allow their contribution to the regression to be estimated accurately; c) there is good reason to believe that no variables or interactions with substantial effects have been left out; d) there is a context of scientific understanding that supports any interpretations that are proposed. Note further the Bradford Hill criteria, in Hill (1965).

## References

- Farmer, C.H. 2006. Another look at Meyer and Finney’s ‘Who wants airbags?’. *Chance* 19:15-22.
- Hill, A.B. 1965. The environment and disease. Association or causation? *Proceedings of the Royal Society of Medicine* 58: 295-300.
- Maindonald, J. H. and Braun, W.J. 2007. *Data Analysis and Graphics Using R – An Example-Based Approach*. 2<sup>nd</sup> edition, Cambridge University Press.  
 URL:<http://wwwmaths.anu.edu.au/~johnm/r-book.html>
- Rosenbaum, P. and Rubin, D., 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70:41–55.

Rosenbaum, P. R., 1999. Choice as an alternative to control in observational studies. *Statistical Science*, 14:259–278. With following discussion, pp.279–304.

Rosenbaum, P. R., 2002. *Observational Studies*. Springer-Verlag, 2 edition.

Zhu, M. and Chipman, H.A. 2006 *Darwinian Evolution in Parallel Universes: A Parallel Genetic Algorithm for Variable Selection*. *Technometrics* 48, 491–502.