Discussion 5

Sam Reeves

Linear Models with R 13.3

The pima dataset contains information on 768 adult female Pima Indians living near Phoenix.

```
head(pima <- faraway::pima %>% mutate_all(as.numeric))
```

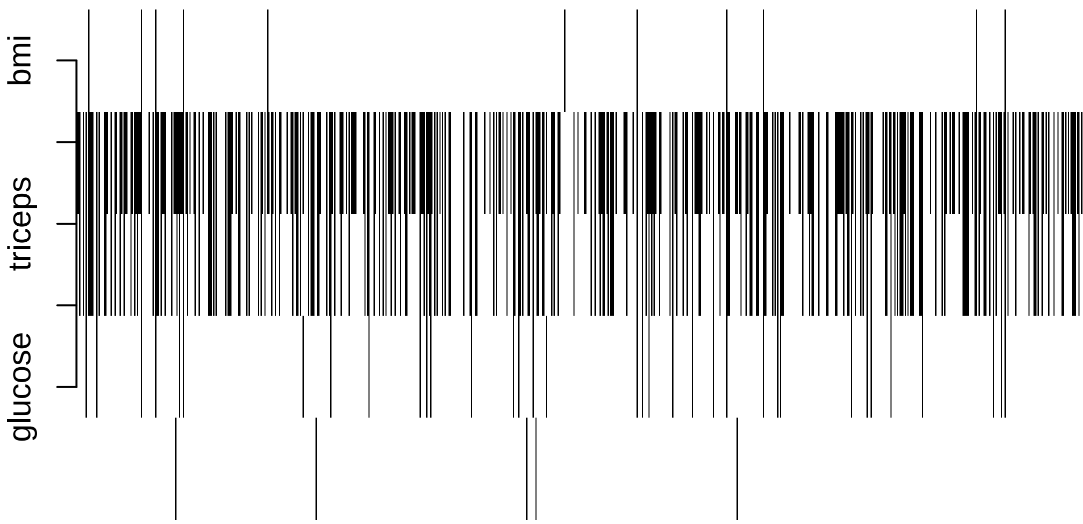
```
##   pregnant glucose diastolic triceps insulin  bmi diabetes age test
## 1        6     148        72      35       0 33.6    0.627  50    1
## 2        1      85        66      29       0 26.6    0.351  31    0
## 3        8     183        64       0       0 23.3    0.672  32    1
## 4        1      89        66      23      94 28.1    0.167  21    0
## 5        0     137        40      35     168 43.1    2.288  33    1
## 6        5     116        74       0       0 25.6    0.201  30    0
```

(a) The analysis in Chapter 1 sugests that zero has been used as a missing value code for several of the variables. Replace these values with NA. Describe the distribution of missing values in the data.

```
na.zero <- c('glucose', 'diastolic', 'triceps', 'insulin', 'bmi')
filled <- pima[na.zero]

filled[filled == 0] <- NA
pima[na.zero] <- filled

image(is.na(filled),axes=FALSE,col=gray(1:0))
axis(2, at = 1:5/5, labels=colnames(filled))
```



It seems likely that if one piece of information is missing, then another may also be missing. All information for 'test', 'age', and 'diabetes' is present. It seems that 'triceps', 'insulin', and 'diastolic' are the most commonly missing.

(b) Fit a linear model with diastolic as the response and the other variables as predictors. Summarize the fit.

```r
summary(lm1 <- lm(diastolic ~ ., pima))
```

```
##
## Call:
## lm(formula = diastolic ~ ., data = pima)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.420  -6.956  -0.604   7.432  29.268
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 41.004185   4.043536  10.141  < 2e-16 ***
## pregnant     0.183487   0.247575   0.741 0.459064
## glucose      0.047134   0.025848   1.824 0.069003 .
## triceps     -0.005719   0.074506  -0.077 0.938851
## insulin     -0.008268   0.006027  -1.372 0.170913
## bmi          0.532806   0.112798   4.724 3.26e-06 ***
```

```
## diabetes    -3.213760   1.722406  -1.866 0.062826 .
## age          0.284048   0.081494   3.485 0.000548 ***
## test         0.047652   1.508849   0.032 0.974822
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.38 on 383 degrees of freedom
##   (376 observations deleted due to missingness)
## Multiple R-squared:  0.1882, Adjusted R-squared:  0.1712
## F-statistic:  11.1 on 8 and 383 DF,  p-value: 3.94e-14
```

Looks like BMI and age have a large effect on the target variable.

(c) Use mean value imputation to the missing cases and refit the model comparing to fit found in the previous question.

```r
(means <- colMeans(filled, na.rm=TRUE))
```

```
##   glucose diastolic   triceps   insulin       bmi
## 121.68676  72.40518  29.15342 155.54822  32.45746
```

```r
mvi <- pima

for (i in c(1:5)) {
  vec <- filled[,i]
  vec[is.na(vec)] <- mean(vec[!is.na(vec)])
  mvi[,i+1] <- vec
}

summary(lm2 <- lm(diastolic ~ ., mvi))
```

```
##
## Call:
## lm(formula = diastolic ~ ., data = mvi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -48.879  -6.599  -0.694   6.369  56.998
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 43.205265   2.620456  16.488  < 2e-16 ***
## pregnant     0.157970   0.141327   1.118  0.26402
## glucose      0.048453   0.016310   2.971  0.00306 **
## triceps      0.006457   0.054022   0.120  0.90489
## insulin     -0.007388   0.005139  -1.438  0.15095
## bmi          0.476441   0.071163   6.695 4.19e-11 ***
## diabetes    -2.127135   1.221251  -1.742  0.08195 .
## age          0.285792   0.041421   6.900 1.10e-11 ***
## test        -0.868070   1.002583  -0.866  0.38686
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 10.92 on 759 degrees of freedom
## Multiple R-squared:  0.1938, Adjusted R-squared:  0.1853
## F-statistic:  22.8 on 8 and 759 DF,  p-value: < 2.2e-16
```

Error is a bit lower, p-value is considerably lower, and now glucose and diabetes play a bigger role in the result!

(d) Use regression-based imputation using the other four geographic predictors to fill in the missing values in the data. Fit the same model and compare to previous fits.

```r
lm3 <- lm(glucose ~ diabetes + age + test + pregnant, pima)
```

The book offers two methods for this. . . . The first is a normal linear regression.

```r
pima[is.na(pima$glucose),]
```

```
##     pregnant glucose diastolic triceps insulin  bmi diabetes age test
## 76         1      NA        48      20      NA 24.7    0.140  22    0
## 183        1      NA        74      20      23 27.7    0.299  21    0
## 343        1      NA        68      35      NA 32.0    0.389  22    0
## 350        5      NA        80      32      NA 41.0    0.346  37    1
## 503        6      NA        68      41      NA 39.0    0.727  41    1
```

```r
predict(lm3, pima[is.na(pima$glucose),])
```

```
##       76      183      343      350      503
## 106.1448 106.3480 107.2806 141.2591 144.3697
```

The other method is by logit transformation.

```r
lm4 <- lm(logit(glucose/100) ~ diabetes + age + test + pregnant, pima)
ilogit(predict(lm4, pima[is.na(pima$glucose),]))*100
```

```
##       76      183      343      350      503
## 91.59940 91.41115 91.17365 91.72546 90.63433
```

Both seem pretty bad to me.

(e) Use multiple imputation to handle missing values and fit the same model again. Compare to previous fits.

```r
set.seed(1337)
pima_imp <- amelia(pima, m = 25)
```

```
## -- Imputation 1 --
##
##   1  2  3  4  5  6  7  8  9 10
##
```

```
## -- Imputation 2 --
##
##   1  2  3  4  5  6  7
##
## -- Imputation 3 --
##
##   1  2  3  4  5  6  7  8  9
##
## -- Imputation 4 --
##
##   1  2  3  4  5  6  7  8  9
##
## -- Imputation 5 --
##
##   1  2  3  4  5  6  7  8  9 10 11
##
## -- Imputation 6 --
##
##   1  2  3  4  5  6  7  8  9 10
##
## -- Imputation 7 --
##
##   1  2  3  4  5  6  7  8  9 10 11 12 13 14
##
## -- Imputation 8 --
##
##   1  2  3  4  5  6  7  8  9 10
##
## -- Imputation 9 --
##
##   1  2  3  4  5  6  7  8  9 10
##
## -- Imputation 10 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 11 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 12 --
##
##   1  2  3  4  5  6  7  8  9 10
##
## -- Imputation 13 --
##
##   1  2  3  4  5  6  7  8  9
##
## -- Imputation 14 --
##
##   1  2  3  4  5  6  7  8  9 10 11
##
## -- Imputation 15 --
##
```

```
##   1  2  3  4  5  6  7  8  9
##
## -- Imputation 16 --
##
##   1  2  3  4  5  6  7  8  9 10 11 12 13
##
## -- Imputation 17 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 18 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 19 --
##
##   1  2  3  4  5  6  7  8  9 10
##
## -- Imputation 20 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 21 --
##
##   1  2  3  4  5  6  7  8  9 10
##
## -- Imputation 22 --
##
##   1  2  3  4  5  6
##
## -- Imputation 23 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 24 --
##
##   1  2  3  4  5  6  7  8  9 10
##
## -- Imputation 25 --
##
##   1  2  3  4  5  6  7  8
```

```
betas <- NULL
ses <- NULL

for (i in 1:pima_imp$m) {
  lmod <- lm (diastolic ~ diabetes + age + test + pregnant, pima_imp$imputations[[i]])
  betas <- rbind(betas, coef(lmod))
  ses <- rbind(ses, coef(summary(lmod))[,2])
}
```

```
(cr <- mi.meld(q=betas,se=ses))
```

```
## $q.mi
```

```
##      (Intercept)   diabetes      age     test  pregnant
## [1,]    61.24715 -0.8953743 0.305828 2.618151 0.1111107
##
## $se.mi
##      (Intercept) diabetes        age     test  pregnant
## [1,]    1.419607 1.313738 0.04395895 0.954869 0.1536927
```

```
# t-statistics
cr$q.mi/cr$se.mi
```

```
##      (Intercept)   diabetes      age     test  pregnant
## [1,]    43.14373 -0.6815471 6.957129 2.741895 0.7229407
```

I'm really not sure how to use these functions.