MAST30027: Modern Applied Statistics

Week 2 Lab

1. The dataset `wbca` comes from a study of breast cancer in Wisconsin. There are 681 cases of potentially cancerous tumors of which 238 are actually malignant. Determining whether a tumor is really malignant is traditionally determined by an invasive surgical procedure. The purpose of this study was to determine whether a new procedure called fine needle aspiration, which draws only a small sample of tissue, could be effective in determining tumor status.

   (a) Load the data and read descriptions of the variables using

   ```
   library(faraway)
   data(wbca)
   ?wbca
   ```

   (b) Fit a binary regression model (logistic regression in this case) using `glm`. Include all the variables in your model (shorthand for this in an R model is $\sim$.).

   ```
   > model <- glm(cbind(Class, 1-Class)~., family=binomial, data=wbca)
   > summary(model)

   Call:
   glm(formula = cbind(Class, 1 - Class) ~ ., family = binomial,
       data = wbca)

   Deviance Residuals:
        Min          1Q      Median          3Q         Max
   -2.48282  -0.01179   0.04739   0.09678   3.06425

   Coefficients:
               Estimate Std. Error z value Pr(>|z|)
   (Intercept) 11.16678    1.41491   7.892 2.97e-15 ***
   Adhes       -0.39681    0.13384  -2.965  0.00303 **
   BNucl       -0.41478    0.10230  -4.055 5.02e-05 ***
   Chrom       -0.56456    0.18728  -3.014  0.00257 **
   Epith       -0.06440    0.16595  -0.388  0.69795
   Mitos       -0.65713    0.36764  -1.787  0.07387 .
   NNucl       -0.28659    0.12620  -2.271  0.02315 *
   Thick       -0.62675    0.15890  -3.944 8.01e-05 ***
   UShap       -0.28011    0.25235  -1.110  0.26699
   USize        0.05718    0.23271   0.246  0.80589
   ---
   Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

   (Dispersion parameter for binomial family taken to be 1)

       Null deviance: 881.388  on 680  degrees of freedom
   Residual deviance:  89.464  on 671  degrees of freedom
   AIC: 109.46

   Number of Fisher Scoring iterations: 8
   ```

   (c) Use the `step` function to search for a model with minimal AIC. Include all variables in the `scope` (type `?step` to see how to use `step`).

   You should end up with the model `cbind(Class, 1 - Class)` $\sim$ `Adhes + BNucl + Chrom + Mitos + NNucl + Thick + UShap`.

   ```
   > model2 <- step(model, scope=~.)
   ```

```
Start:  AIC=109.46
cbind(Class, 1 - Class) ~ Adhes + BNucl + Chrom + Epith + Mitos +
    NNucl + Thick + UShap + USize

        Df Deviance    AIC
- USize  1   89.523 107.52
- Epith  1   89.613 107.61
- UShap  1   90.627 108.63
<none>       89.464 109.46
- Mitos  1   93.551 111.55
- NNucl  1   95.204 113.20
- Adhes  1   98.844 116.84
- Chrom  1   99.841 117.84
- BNucl  1  109.000 127.00
- Thick  1  110.239 128.24

Step:  AIC=107.52
cbind(Class, 1 - Class) ~ Adhes + BNucl + Chrom + Epith + Mitos +
    NNucl + Thick + UShap

        Df Deviance    AIC
- Epith  1   89.662 105.66
- UShap  1   91.355 107.36
<none>       89.523 107.52
+ USize  1   89.464 109.46
- Mitos  1   93.552 109.55
- NNucl  1   95.231 111.23
- Adhes  1   99.042 115.04
- Chrom  1  100.153 116.15
- BNucl  1  109.064 125.06
- Thick  1  110.465 126.47

Step:  AIC=105.66
cbind(Class, 1 - Class) ~ Adhes + BNucl + Chrom + Mitos + NNucl +
    Thick + UShap

        Df Deviance    AIC
<none>       89.662 105.66
- UShap  1   91.884 105.88
+ Epith  1   89.523 107.52
+ USize  1   89.613 107.61
- Mitos  1   93.714 107.71
- NNucl  1   95.853 109.85
- Adhes  1  100.126 114.13
- Chrom  1  100.844 114.84
- BNucl  1  109.762 123.76
- Thick  1  110.632 124.63
```

> summary(model2)

```
Call:
glm(formula = cbind(Class, 1 - Class) ~ Adhes + BNucl + Chrom +
    Mitos + NNucl + Thick + UShap, family = binomial, data = wbca)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.44161  -0.01119   0.04962   0.09741   3.08205

Coefficients:
```

```
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  11.0333     1.3632   8.094 5.79e-16 ***
Adhes        -0.3984     0.1294  -3.080  0.00207 **
BNucl        -0.4192     0.1020  -4.111 3.93e-05 ***
Chrom        -0.5679     0.1840  -3.085  0.00203 **
Mitos        -0.6456     0.3634  -1.777  0.07561 .
NNucl        -0.2915     0.1236  -2.358  0.01837 *
Thick        -0.6216     0.1579  -3.937 8.27e-05 ***
UShap        -0.2541     0.1785  -1.423  0.15461
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 881.388  on 680  degrees of freedom
Residual deviance:  89.662  on 673  degrees of freedom
AIC: 105.66

Number of Fisher Scoring iterations: 8
```

(d) Using the reduced model, use `predict` to estimate the outcome for a new patient with predictors 1, 1, 3, 1, 1, 4, 1. You will need to put `newdata = list(Adhes=1, BNucl=1, Chrom=3, Mitos=1, NNucl=1, Thick=4, UShap=1)` and `type="response"`.

To get a 95% CI for your estimate, use `predict` with `type="link"` and `se.fit=TRUE`, to obtain the estimate and its standard error *on the linear scale*. Use these to get a symmetric CI on the linear scale, which you can then transform back to the response scale.

```
> predict(model2, newdata = list(Adhes=1, BNucl=1, Chrom=3, Mitos=1, NNucl=1,
+                                Thick=4, UShap=1), type="response")
        1
0.9921115
> (x <- predict(model2, newdata = list(Adhes=1, BNucl=1, Chrom=3, Mitos=1, NNucl=1,
+                                      Thick=4, UShap=1), type="link", se.fit=TRUE))
$fit
       1
4.834428

$se.fit
[1] 0.5815185

$residual.scale
[1] 1
> ilogit(c(x$fit-2*x$se.fit, x$fit, x$fit+2*x$se.fit))
        1         1         1
0.9751901 0.9921115 0.9975211
```

(e) Suppose that a cancer is classified as benign if $p \geq 0.5$ and malignant if $p < 0.5$. Compute the number of errors of both types that will be made if this method is applied to the current data with the reduced model.

```
> pfit <- predict(model2, type="response")
> (false_neg <- sum(pfit >= 0.5 & !wbca$Class)/sum(!wbca$Class))
[1] 0.04621849
> (false_pos <- sum(pfit < 0.5 & wbca$Class)/sum(wbca$Class))
[1] 0.02031603
```

(f) Suppose we change the cutoff to 0.9 so that $p < 0.9$ is classified as malignant and $p \geq 0.9$ as benign. Compute the number of errors in this case.

Consider how you might determine the cutoff in practice.

```
> pfit <- predict(model2, type="response")
> (false_neg <- sum(pfit >= 0.9 & !wbca$Class)/sum(!wbca$Class))
[1] 0.004201681
> (false_pos <- sum(pfit < 0.9 & wbca$Class)/sum(wbca$Class))
[1] 0.03611738
```

Clearly there is a trade-off between false positives and false negatives. Where you choose the cut-off depends on the relative costs (individial and societal) in each case. For medical tests we usually prefer to reduce the false negative rate at the expense of increasing the false positive rate, especially for a screening test, where there is the opportunity for further testing following a positive result.

2. The National Institute of Diabetes and Digestive and Kidney Diseases conducted a study on 768 adult female Pima Indians living near Phoenix. The purpose of the study was to investigate factors related to diabetes. The data may be found in the the dataset `pima`. Read the help file (`?pima`) to get a description of the predictor and response variables. There are missing observations for many variables, which have been recorded as zeros. The easiest (not necessarily the best) way to deal with these is to remove the relevant observations from the data set.

```
> missing <- with(pima, missing <- glucose==0 | diastolic==0 | triceps==0 | bmi == 0)
> pima <- pima[!missing,]
```

(a) Fit a model with `test` as the response and all the other variables as predictors.

```
> model <- glm(cbind(test, 1-test)~., family=binomial, data=pima)
> summary(model)

Call:
glm(formula = cbind(test, 1 - test) ~ ., family = binomial, data = pima)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8627  -0.6639  -0.3672   0.6347   2.4942

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -9.677562   1.005400  -9.626  < 2e-16 ***
pregnant     0.121235   0.043926   2.760 0.005780 **
glucose      0.037439   0.004765   7.857 3.92e-15 ***
diastolic   -0.009316   0.010446  -0.892 0.372494
triceps      0.006341   0.014853   0.427 0.669426
insulin     -0.001053   0.001007  -1.046 0.295651
bmi          0.085992   0.023661   3.634 0.000279 ***
diabetes     1.335764   0.365771   3.652 0.000260 ***
age          0.026430   0.013962   1.893 0.058371 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 676.79  on 531  degrees of freedom
Residual deviance: 465.23  on 523  degrees of freedom
AIC: 483.23

Number of Fisher Scoring iterations: 5
```

(b) Do women who test positive have higher diastolic blood pressures? Is the diastolic blood pressure significant in the regression model? Explain the distinction between the two questions and discuss why the answers are only apparently contradictory.

Recall that `model` used all the predictor variables.

```
> cor(pima)
              pregnant   glucose   diastolic    triceps      insulin
pregnant   1.000000000 0.1253296 0.204663421 0.09508511 -0.006568130
glucose    0.125329647 1.0000000 0.219177950 0.22659042  0.459904606
diastolic  0.204663421 0.2191779 1.000000000 0.22607244  0.007051676
triceps    0.095085114 0.2265904 0.226072440 1.00000000  0.126240293
insulin   -0.006568130 0.4599046 0.007051676 0.12624029  1.000000000
bmi        0.008576282 0.2470793 0.307356904 0.64742239  0.191167600
diabetes   0.007435104 0.1658174 0.008047249 0.11863557  0.151531103
age        0.640746866 0.2789071 0.346938723 0.16133614  0.081126066
test       0.252585511 0.5036139 0.183431874 0.25487371  0.212204307
                 bmi    diabetes        age      test
pregnant  0.008576282 0.007435104 0.64074687 0.2525855
glucose   0.247079294 0.165817411 0.27890711 0.5036139
diastolic 0.307356904 0.008047249 0.34693872 0.1834319
triceps   0.647422386 0.118635569 0.16133614 0.2548737
insulin   0.191167600 0.151531103 0.08112607 0.2122043
bmi       1.000000000 0.151107136 0.07343826 0.3009007
diabetes  0.151107136 1.000000000 0.07165413 0.2330739
age       0.073438257 0.071654133 1.00000000 0.3150968
test      0.300900748 0.233073898 0.31509683 1.0000000
```

```
> summary(model)

Call:
glm(formula = cbind(test, 1 - test) ~ ., family = binomial, data = pima)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8627  -0.6639  -0.3672   0.6347   2.4942

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -9.677562   1.005400  -9.626  < 2e-16 ***
pregnant     0.121235   0.043926   2.760 0.005780 **
glucose      0.037439   0.004765   7.857 3.92e-15 ***
diastolic   -0.009316   0.010446  -0.892 0.372494
triceps      0.006341   0.014853   0.427 0.669426
insulin     -0.001053   0.001007  -1.046 0.295651
bmi          0.085992   0.023661   3.634 0.000279 ***
diabetes     1.335764   0.365771   3.652 0.000260 ***
age          0.026430   0.013962   1.893 0.058371 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 676.79  on 531  degrees of freedom
Residual deviance: 465.23  on 523  degrees of freedom
AIC: 483.23

Number of Fisher Scoring iterations: 5
```

`diastolic` is not significant in the presence of the other variables.

There is positive correlation between `diastolic` and `test`, yet in the model `diastolic` has a negative coefficient. This is possible because `diastolic` is correlated with other (more significant) variables: the test *is* more likely to be positive when `diastolic` is large, but this is because `glucose`, `triceps`, `bmi` and `age` are all more likely to be large, and these all have the effect of increasing the chance of a positive test.

(c) Predict the outcome for a woman with predictor values 1, 99, 64, 22, 76, 27, 0.25, 25 (same order as in the dataset). Give a confidence interval for your prediction.

```
> x <- predict(model, newdata = list(pregnant=1, glucose=99, diastolic=64, triceps = 22, in
+              type="link", se.fit=TRUE)
> ilogit(c(x$fit-2*x$se.fit, x$fit, x$fit+2*x$se.fit))

         1          1          1
0.02632817 0.04435407 0.07378636
```

3. Consider the binomial regression model with logit link fitted to the Challenger data in class. Using the log likelihood ratio, plot a 95% confidence region for (α, β).

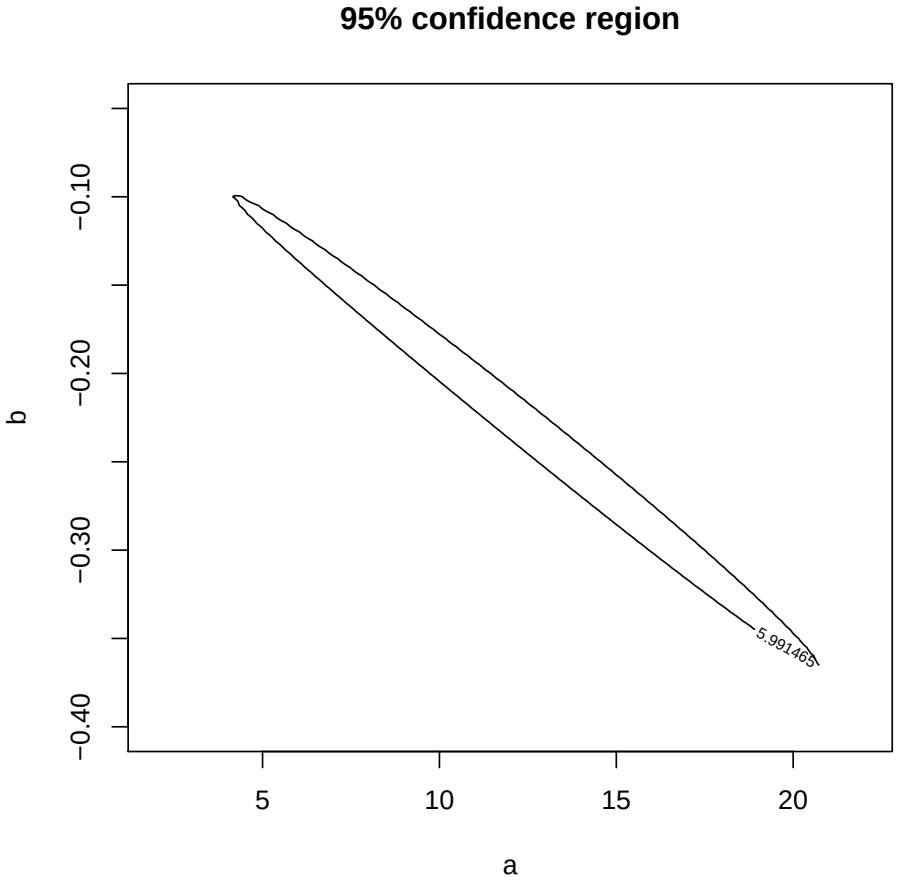
   One way of doing this is to use the function `contour`:

   (a) Let $(\hat{\alpha}^*, \hat{\beta}^*)$ be the MLE, then for a grid of α and β values calculate $2l(\hat{\alpha}^*, \hat{\beta}^*) - 2l(\alpha, \beta)$.

   (b) The contour line with value $\chi^2_2(0.95)$ will delineate the confidence region.

   Solution:

```
> # load data and fit model
> library(faraway)
> data(orings)
> logitmod <- glm(cbind(damage,6-damage) ~ temp, family=binomial, orings)
> # log-likelihood function
> logL <- function(beta, orings) {
+   eta <- cbind(1, orings$temp) %*% beta
+   return( sum(orings$damage*eta - 6*log(1 + exp(eta))) )
+ }
> # log-likelihood ratio for beta = c(a, b) against beta = betafit
> logLR <- function(a, b, betafit, orings) 2*logL(betafit, orings) - 2*logL(c(a, b), orings)
> # interested in c(a, b) such that f(a, b, ...) <= qchisq(0.95, 2)
> a_vec <- seq(2, 22, 0.1)
> b_vec <- seq(-0.4, -0.05, .005)
> z <- matrix(0, nrow = length(a_vec), ncol = length(b_vec))
> for (i in 1:length(a_vec)) {
+   for (j in 1:length(b_vec)) {
+     z[i,j] <- logLR(a_vec[i], b_vec[j], logitmod$coefficients, orings)
+   }
+ }
> # a vectorised alternative for R afficionados
> # z <- outer(a_vec, b_vec, Vectorize(logLR, c("a", "b")),
> #            betafit = logitmod$coefficients, orings = orings)
> contour(a_vec, b_vec, z, levels = qchisq(0.95, 2),
+         xlab="a", ylab="b", main="95% confidence region")
```


95% confidence region
b
−0.10
−0.20
−0.30
−0.40
5
10
15
20
a
5.991465