STATISTICAL MODELLING Part IIC

Practical 6: Binomial regression IAC/Lent 2010

Comments and corrections to ioana@statslab.cam.ac.uk

Start R and change the current working directory to `U:\Rwork`. Next, download the `AlloyData` from the course web page

`http://www.statslab.cam.ac.uk/~ioana/statsmod.html`

Save the file `alloy.txt` in the current working directory (`U:\Rwork`). The AlloyData studies the compressive strength of an alloy fastener used in aircraft construction. The pressure loads $x_1, ..., x_{10}$ range from 2500 pounds per square inch (psi) to 4300 psi. The number of fasteners tested and the number of failures are reported.

```
> AlloyData <- read.table("alloy.txt", header = TRUE)
> attach(AlloyData, warn.conflicts = FALSE)
```

Figure 1 shows a plot of the `AlloyData` showing the porportion of fasteners which failed as a function of pressure load.

It is natural to assume that the data $y_1, \ldots, y_n$ $(n = 10)$ are realisations of independent binomial random variables $Y_1, \ldots, Y_n$ with $Y_i \sim \text{Bin}(n_i, p_i)$ for $i = 1, \ldots, n$. We want to model the dependence of Y_i on the ith pressure load x_i, and we suppose that this dependence is in the way that p_i depends on x_i. Our initial model for the data is

$$\text{logit}(p_i) \equiv \log\Big(\frac{p_i}{1-p_i}\Big) = \alpha + \beta x_i, \quad i = 1, \ldots, n.$$

The `glm` function works in a similar way to the `lm` function, but when working with binomial regression models in R, we need to include an argument consisting of a vector of `weights`. Recalling that the dispersion parameter of the ith observation is $\sigma_i^2 = \sigma^2 a_i$, the ith weight is $1/a_i$, which is n_i in our model above. R uses these weights to form the matrix $\hat{W}_m$ used in the iterated weighted least squares algorithm.

```
> BinMod1 <- glm(y/n ~ x, family = binomial, weights = n)
```

As with the `lm` function, the output is stored as an object. You can find the many components of this object with

```
> plot(x, y/n, xlab = "pressure load", ylab = "proportion failed",
+     main = "Alloy Data, proportion of fasteners which failed")
```

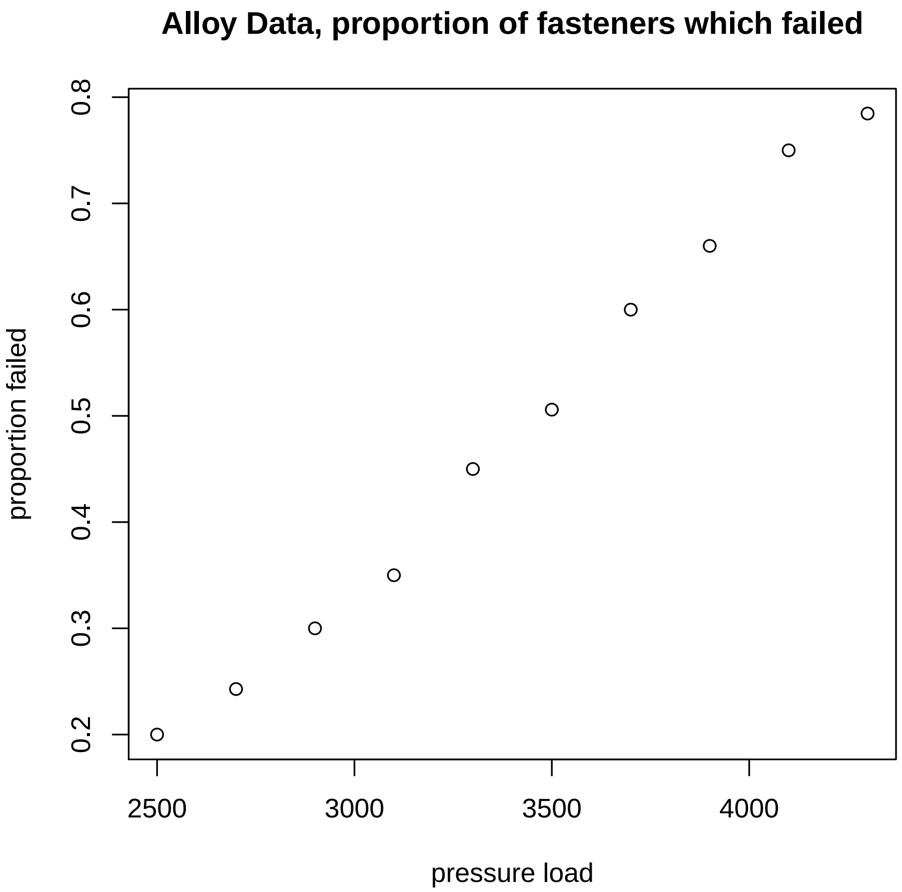

Figure 1: `AlloyData`: proportion of fasteners which failed plotted as a function of pressure load

```
> names(BinMod1)
```

```
 [1] "coefficients"      "residuals"         "fitted.values"
 [4] "effects"           "R"                 "rank"
 [7] "qr"                "family"            "linear.predictors"
[10] "deviance"          "aic"               "null.deviance"
[13] "iter"              "weights"           "prior.weights"
[16] "df.residual"       "df.null"           "y"
```

```
[19] "converged"          "boundary"           "model"
[22] "call"               "formula"            "terms"
[25] "data"               "offset"             "control"
[28] "method"             "contrasts"          "xlevels"
```

but most of the relevant information can be accessed simultaneously with

```
> summary(BinMod1)


Call:
glm(formula = y/n ~ x, family = binomial, weights = n)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.29475  -0.11129   0.04162   0.08847   0.35016

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -5.3397115  0.5456932  -9.785   <2e-16 ***
x            0.0015484  0.0001575   9.829   <2e-16 ***
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.1 âĂŸ âĂŹ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 112.83207  on 9  degrees of freedom
Residual deviance:   0.37192  on 8  degrees of freedom
AIC: 49.088

Number of Fisher Scoring iterations: 3
```

An example sheet question asks you to verify the calculations leading to the values given for standard errors, z-values and residual deviance in the summary. What approximation is used to compute the standard errors?

Is pressure load significant in explaining the failure of alloy fasteners? The quantity $p/(1-p)$ is called the *odds of success*, where, in this example, success is the failure of a fastener. How does the log odds change as the pressure load increases by 1 psi?

The null deviance compares the unrestricted model in which $Y_1, \dots, Y_n$ are independent with $Y_i \sim \text{Bin}(n_i, p_i)$, with the 'null' model, with p_i constant for $i = 1, \dots, n$, i.e., $\log\{p_i/(1-p_i)\} = \alpha$.

For binomial regression, the log-likelihood function is

$$\ell(p, \sigma^2) = \sum_{i=1}^{n} \log\left\{ \binom{n_i}{y_i} p_i^{y_i} (1-p_i)^{n_i - y_i} \right\},$$

where $p = (p_1, \dots, p_n)$ and $\sigma^2 = 1$. In the unrestricted model, the m.l.e. of p_i is $\tilde{p}_i = y_i/n_i$, whereas in the null model, it is $\bar{p} = \sum y_i / \sum n_i$. So the null deviance is defined as

$$D(\tilde{p}, \bar{p}) = 2\sigma^2 \left[\ell(\tilde{p}, \sigma^2) - \ell(\bar{p}, \sigma^2) \right].$$

In R, this is computed by

```
> 2 * (sum(dbinom(y, n, y/n, log = TRUE)) - sum(dbinom(y, n,
+     mean(y)/mean(n), log = TRUE)))
```

```
[1] 112.8321
```

The residual deviance (simply called the deviance in lectures) compares the unrestricted model with the model we are primarily interested in, namely the logistic regression model with $\log\{p_i/(1-p_i)\} = \alpha + \beta x_i$. Let $\hat{p}_i$ denote the m.l.e. in this logistic model. Then the residual deviance is given by

$$D(\tilde{p}, \hat{p}) = 2\sigma^2 \left[\ell(\tilde{p}, \sigma^2) - \ell(\hat{p}, \sigma^2) \right],$$

and it is computed in R by

```
> beta <- coef(BinMod1)
> X <- model.matrix(y/n ~ x)
> p.hat <- exp(X %*% beta)/(1 + exp(X %*% beta))
> 2 * (sum(dbinom(y, n, y/n, log = TRUE)) - sum(dbinom(y, n,
+     p.hat, log = TRUE)))
```

```
[1] 0.3719169
```

Recall that, since the dispersion parameter $\sigma^2 = 1$ for this binomial situation, the residual deviance is the likelihood ratio statistic for testing our logistic regression model against the unrestricted model, and has an approximate χ^2_{n-p} distribution (cf. the discussion of small dispersion asymptotics) if our logistic regression model is correct. In this example $n - p = 8$. Is the fit of the model satisfactory? What p-value do you obtain?

The penultimate piece of information in the summary is the Akaike information criterion (AIC). This is defined up to an additive constant as

$$\text{AIC} = -2\ell(\hat{p}, \hat{\sigma}^2) + 2p,$$

where p is the dimension of the parameter space in the model (here $p = 2$ as we have two unknown parameters, α and β), and $\hat{p}$, $\hat{\sigma}$ are the m.l.e. estimates returned by Iterated Weighted Least Squares algorithm. In comparing different models, one criterion is to seek to minimise the AIC – notice the trade-off between maximising the log-likelihood and keeping the dimension of the parameter space small.

The final part of the summary tells us the number of Fisher scoring iterations required for the difference between successive iterations to be satisfactorily small. The `boot` library contains logit and inverse logit functions, so we can see our fitted model.

```
> plot(x, y/n, xlab = "pressure load", ylab = "proportion failed",
+     main = "Fitted Alloy Data, proportion of fasteners which failed")
> library(boot)
> lines(x, inv.logit(coef(BinMod1)[[1]] + coef(BinMod1)[[2]] *
+     x))
> lines(x, fitted.values(BinMod1), col = 2)
```

The last line above should have the same effect. Figure 2 shows the result.

EXERCISES:

- Write down the equation of the curve being plotted in the last line.
- The probit link function is $g(\mu) = \Phi^{-1}(\mu)$, while the complementary log-log function is $g(\mu) = \log\{-\log(1-\mu)\}$. Write down the models that are being fitted with the following commands, compare the summaries with the first model, and add the fitted lines to your plots, with dotted and dashed lines respectively.

  ```
  > BinMod2 <- glm(y/n ~ x, family = binomial(link = probit),
  +     weights = n)
  > BinMod3 <- glm(y/n ~ x, family = binomial(link = cloglog),
  +     weights = n)
  ```

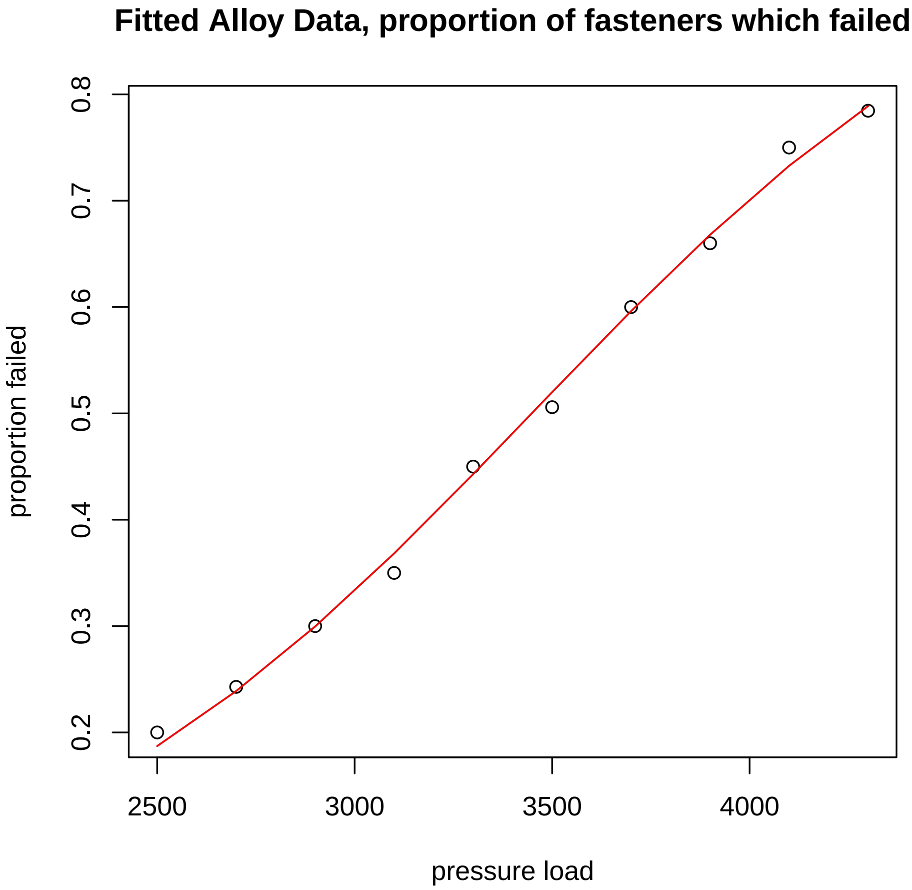

Figure 2: Fit of the `AlloyData`.

- The next data set is `SpaceData`, contained in `space.txt` on the course web page. Download the data, read the information about the data and then read the table into R with `read.table`. Plot the points and fit a logistic regression model to the data (you will need to define a vector of length 23 with each component equal to 6). Add the fitted line to your plot. Although the experimental design is far from perfect (how would you improve it?), what would you conclude?

- Look at the discussion in the original data file of the conclusions reached by the NASA staff. How is your model affected if you omit the points with no failures using the `subset=(y>0)` argument to the `glm` function?