Statistical Methods for Big Data

Worksheet 13 Logistic Regression

In January 1986, the space shuttle Challenger exploded shortly after launch. An investigation was launched into the cause of the crash and attention focused on the rubber O-ring seals in the rocket boosters. At lower temperatures, rubber becomes more brittle and is a less effective sealant. At the time of the launch, the temperature was 31oF

In the 23 previous shuttle mission, some evidence of damage was recorded on some O-rings. For each mission, we know the number of O-rings out of six showing some damage and the launch temperature. We aim to model the relationship between launch temperature and damage to the O-rings.

We look at two cases:

Case A: where we model the response variable as a binary variable, where 1 represents the outcome that damage occurred to at least one of the O-rings.

Case B: where we model the response variable as the proportion of O-rings damaged out of the total of 6.

Case A

Install the faraway package.

The orings data set is included in the faraway package, to include in the workspace use:

data(orings)

orings<-orings

The table has two variables, temp and damage.

We will model the response variable as a binary variable with two possible outcomes:

* 1 represents the outcome that there is damage to at least one of the six O-rings
* 0 represents the event that there is no damage on any of the six O-rings.

As it stands now, the data in the damage variable represents how many of the O-rings were damaged, so we need to create a new binary variable indicating whether there was damage present to at least one of the O-rings or not. Call the new variable Binary.

Binary<-orings$damage

Binary[orings$damage == 0] <- 0

Binary[orings$damage > 0] <- 1

Plot the variable Binary as a function of the launch temperature.

We will use the function glm to fit a logistic model to the data. Find out about the glm function:

?glm

We see that the glm function is of the form:

glm(formula, family = Gaussian)

The formula argument indicates the relationship between the variables and the family argument indicates which link function R should use.

Since we are fitting a logistic model to the data, the family argument is binomial.

If the response variable is binary (0 or 1) then the response variable in the formula argument is simply the column containing the binary data.

If the response variable is a proportion then the response variable in the formula argument consists of a matrix with two columns, where the first column represents the number of “successes” and the second column represents the number of “failures”.

To fit the logistic regression model to the binary data, use:

Ch\_model\_binary<-glm(Binary~orings$temp, family=binomial)

Recall, the logistic regression equation is given by:

where pi represents the probability of damage occurring to at least one of the O-rings.

Summarise the model and write down the equation representing the model.

What does the model tell you about the relationship between launch temperature and the probability of damage to at least one of the O-rings?

To plot the model on the graph showing the relationship between damage and launch temperature we can use:

plot(Binary~orings$temp, xlim=c(40,90), ylim=c(0,1), xlab="Temperature", ylab="p")

x<-seq(40, 90, 1)

This creates a vector starting at the value 40 and increasing to 90 in steps of size 1.

lines(x,ilogit(15.04-0.23\*x),type="l")

Since the logit function is of the form:

The ilogit function plots the inverse of the logit function i.e. it plots the logistic function

The plot shows that as the temperature drops below 50oF the probability of damage occurring to at least one of the O-rings approaches 1. At 65⁰F the probability of damage occurring to at least one of the O-rings is approximately 0.5 and as the temperature increases above 800F the probability of damage occurring to at least one of the O-rings approaches 0.

Next we wish to examine the diagnostic graphs to assess the fit of the model. The plot(model) command works for generalized linear models, just as it does for linear models. The residuals are calculated differently for generalized linear models but R does this for us.

Note that for data where *ni* = 1, (as is the case for this example) the diagnostic plots are not informative.

Let’s demonstrate the command anyway (to be used on other data sets).

plot(Ch\_model\_Binary)



The plot of the residuals versus fitted values shows a distinct pattern, this is because of the binary nature of the data and it does not indicate a lack of fit. So, one string of points corresponds to the observations with a 1, and the other to observations with a 0. This graph is still useful as it allows you to check that there are no samples with a score of 1 in the string of points for the 0's, and vice versa.

Let’s, look at the predicted values

pred<-predict(Ch\_model\_Binary)

Our aim is to model the relationship between launch temperature and damage to at least one of the O-rings. The response variable pi represents the probability that damage has occurred to at least one of the O-rings. What do you notice about the predicted values?

The predicted values are clearly not probabilities. The predict function has predicted values of the *log of the odds* of these probabilities, . To calculate the predicted probabilities we can take the inverse.

pred\_prob<-exp(pred)/(1+exp(pred))

Alternatively, we can set type=response, or use the fitted command

pred<-predict(Ch\_model\_Binary, type=”response”)

fitted(Ch\_model\_Binary)

Since, in this example, *ni* = 1 testing the model residual deviance directly using the chi-squared test is not possible. We can assess the goodness of fit based on prediction errors. Suppose we were to use the fitted model to predict `success' if the fitted probability exceeds 0.5 and `failure' otherwise. We can then compare the observed and predicted responses, and calculate the proportion of cases predicted correctly.

First we need to create a vector of predicted values:

We create a new vector containing the predicted probabilities (pred\_binary) then assign 1 to observations where the probability is > 0.5 and assign 0 to observations where the probability is < 0.5

pred\_binary <- predprob

pred\_binary [pred\_binary < 0.5] <- 0

pred\_binary [pred\_binary > 0.5] <- 1

we can compare the predicted outcomes to the observed values contained in the Damage vector

diff<- Binary – pred\_binary

we can calculate the proportion of cases predicted correctly as:

1-sum(diff)/length(pred\_binary)

Whether the proportion of correct predictions is acceptable, depends on the accuracy required of the model.

Case B

Next, model the response variable as the proportion of O-rings damaged out of the total of 6.

To create the matrix containing the response variable we use the command:

y<-cbind(orings$damage, 6-orings$damage)

then

Ch\_model\_prom<-glm(y~orings$temp, family=binomial)

Note that in this case, the number of trials ni = 6 since there are six O-rings for each temperature recorded.

**Exercise**

Next, analyse the proportional data from the insect gender ratio experiment. You need to import the genderratio.txt file from Blackboard.

Note that for this data it is possible to calculate the goodness of fit using the residual deviance of the model since we have ni > 5 for all but two of the observations.

To test the model fit using the residual deviances use the pchisq command.

For the insect gender ratio example

To compare the fitted model with the Null model

pchisq(71.1593-5.6739,1, lower=F)

The p-value is < 0.05 i.e. we find that the fitted model explains a significant

amount of the variation in the data compared to the null model

We can also test to see whether the fitted model is an adequate fit (null)

vs not an adequate fit (alternative) by comparing the fitted model to the saturated model.

pchisq(5.6739,6, lower=F)

The non significant p-value indicates that this model is an adequate fit