**Section 2.2.3 – Odds ratios**

A logistic regression model can be written as



where the left-side of the model is the log odds of a success. Using a similar interpretation as for normal linear regression models, we can look at βr then to interpret the effect that xr has on this log odds of a success. We can then form odds ratios by looking at these odds at different values of xr.

For ease of presentation, consider the logistic regression with only one explanatory variable x:

 

We can re-write this model as



where I replaced π/(1 – π) with Oddsx to help with the notation. For example, the odds of success are  for the placekicking example. When distance is 30 yards, we have .

If we increase x by c-units, the odds of success become



To interpret the effect of increasing x by c-units, we can form an odds ratio:



Notice that x falls out!!! Thus, it does not matter what the value of x is, the odds ratio remains the same for a c-unit increase. This is one of the main reasons why logistic regression is the most used way to model binary responses.

On your own, verify  for the model given at the beginning of this section.

There are a number of ways to interpret the odds ratio in the context of logistic regression. I recommend using the following:

The odds of success are  times as large for a c-unit increase in x.

Another correct interpretation is

The odds of success change by  times for every c-unit increase in x.

The interpretations here are worded a little different from what we had in Chapter 1. The reason is due to x not necessarily having ONLY two levels.

Suppose x did only have two levels coded as 0 or 1 as commonly done for indicator variables in normal linear regression. This leads to



and



as the only possible odds. The odds ratio becomes

 

In this situation, you could say

The odds of success are  times as large for x = 1 than for x = 0.

To find the estimated odds ratio corresponding to xr, simply replace the parameter with its corresponding estimate:



The interpretation of the odds ratio now needs to have an “estimated” inserted in the appropriate location. This estimate is the MLE.

Confidence intervals for OR

Wald confidence intervals are the easiest to calculate. First, an interval for cβ1 needs to be found:



where  is obtained from the estimated covariance matrix for the parameter estimates. Notice where c is located in the interval calculation. The second c comes about through .

You may have seen in another class that for a random variable Y and constant a, Var(aY) = a2Var(Y).

To find the Wald confidence interval for OR, we use the exponential function:



A better interval is a profile likelihood ratio interval. For this interval, we find the set of β1 values such that



is satisfied. On the left-side, we have the usual -2log(Λ) form, but without a specified value of β1. The  is an estimate of β0 for a fixed value of β1. Iterative numerical procedures can be used to find the β1 values which satisfy the above equation.

An ad-hoc approach would be to simply try a large number of possible values of β1 and obtain  for each β1. Next, the -2log(Λ) equation would be calculated to see which β1’s satisfy . The smallest and largest β1 values that satisfy the inequality would be used as the lower and upper confidence interval limits. Please see my Placekick.R program for an example.

The profile LR interval for OR is then



using the lower and upper limits found for β1 in the above equation.

A standard interpretation for the interval is

With (1 – α)100% confidence, the odds of success are between <lower limit> to <upper limit> times as large for a c-unit increase in x.

Another version is

With (1 – α)100% confidence, the odds of success change by an amount between <lower limit> to <upper limit> times for every c-unit increase in x.

Comments:

1. Inverting odds ratios less than 1 can be helpful for interpretation purposes.
2. An appropriate value of c should be chosen in the context of the explanatory variable. For example, if 0.1 < x < 0.2, a value of c = 1 would not be appropriate. Additionally, if 0 < x < 1000, a value of c = 1 may not be appropriate as well.
3. When there is more than one explanatory variable, the same interpretation of the odds ratio *generally* can be made with the addition of “holding the other explanatory variables constant” added. This is basically the same as what is done in normal linear regression.
4. If there are interaction, quadratic, or other transformations of the explanatory variables contained within the model, the odds ratio is not simply  as given previously, because the odds ratio is no longer constant for every c-unit increase in x. We will examine what the odds ratio is later.
5. A categorical explanatory variable represented by multiple indicator variables does not have the same type of interpretation as given previously. We will examine this later.

Example: Placekicking (Placekick.R, Placekick.csv)

Consider the model with only the distance of the placekick as the explanatory variable:



To estimate the odds ratio, we can simply use the exp() function:

> exp(mod.fit$coefficients[2])

 distance

 0.8913424

> exp(-10\*mod.fit$coefficients[2])

 distance

 3.159035

The first odds ratio is for a 1-yard (c = 1) increase in distance. This is not very meaningful in the context of the problem! Instead, c = 10 would be much more meaningful because 10 yards are needed for a first down in football. Also, it is more meaningful to look at a 10-yard *decrease* (another first down) rather than a 10-yard increase. Thus, use c = -10.

Interpretation:

* The estimated odds of success are  times as large for a 10-yard decrease in the distance of the placekick.
* The estimated odds of success change 3.16 times for every 10-yard decrease in the distance of the placekick.

Because the odds of success are larger for a 10-yard decrease, a football coach would prefer to go for a field goal at the shorter distance (if possible). While this result is not surprising for football fans, this research quantifies the amount of benefits, which was previously unknown before my research. Also, remember that the 3.16 holds for comparing 30 to 20-yard placekicks as well as 55 to 45-yard placekicks or any other 10-yard decrease.

To account for the variability in the odds ratio estimator, we would like to calculate a confidence interval for the actual odds ratio itself. Below is the code for the profile likelihood ratio interval:

> beta.ci <- confint(object = mod.fit, parm = "distance",

 level = 0.95)

Waiting for profiling to be done...

> beta.ci

 2.5 % 97.5 %

 -0.13181435 -0.09907103

> rev(exp(-10\*beta.ci)) #Invert OR C.I. and c=10

 97.5 % 2.5 %

 2.693147 3.736478

> #Remove labels with as.numeric()

> as.numeric(rev(exp(-10\*beta.ci))

[1] 2.693147 3.736478

The confint() function first finds an interval for β1 itself. This function actually uses a method function in the stats package for the computation. We then use the exp() function to find the confidence interval for OR. The 95% profile likelihood ratio confidence interval is 2.69 < OR < 3.74. Pay special attention to how this was found with the rev() function and the beta.ci object. The standard interpretation of the interval is

Interpretation:

* With 95% confidence, the odds of success are between 2.69 to 3.74 times as large for a 10-yard decrease in the distance of the placekick.
* With 95% confidence, the odds of a success change by an amount between 2.69 to 3.74 times for every 10-yard decrease in the distance of the placekick.

Because the interval is entirely above 1, there is sufficient evidence that a 10-yard decrease in distance increases the odds of a successful placekick.

While a profile likelihood ratio interval is usually preferred, it is instructive to see how to calculate a Wald interval as well, because there will be instances later where we will not be able to calculate a profile likelihood ratio interval.

> beta.ci <- confint.default(object = mod.fit, parm =

 "distance", level = 0.95)

> beta.ci

 2.5 % 97.5 %

distance -0.1313709 -0.0986824

> rev(exp(-10\*beta.ci)) #Invert OR C.I. with c=10

[1] 2.682701 3.719946

I used the confint.default() function for part of the calculations, because there is no “Wald” like option in confint(). The 95% Wald interval is 2.68 < OR < 3.72, which is similar to the profile likelihood ratio interval due to the large sample size.

To see how these calculations are performed without the confint.default() function, below is an example of how to program into R the corresponding formula:

> beta.ci <- mod.fit$coefficients[2] + qnorm(p = c(0.025,

 0.975))\*sqrt(vcov(mod.fit)[2,2])

> beta.ci

[1] -0.1313709 -0.0986824

> rev(-10\*beta.ci))

[1] 2.682701 3.719946

I show in the corresponding program how to find the profile likelihood ratio interval without using confint(). It is instructive to see how this done because it shows how to use the uniroot() function and how to estimate a logistic regression model with a fixed constant in the formula argument of glm().