Introduction

The Hosmer-Lemeshow test (HL test) is a goodness of fit test for binary classification models which tells how well data fits a given model. Specifically, the HL test calculates if the observed event rates match the expected event rates in population subgroups and could be used as a supporting diagnostic as to whether to accept or reject a given model.

The idea is to group observations into categories on the basis of their predicted probabilities into a number of groups and it is calculated with the following formula:

$$G_{HL} = \sum_{j=1}^{g} \frac{(o_j - e_j)^2}{e_j}$$

Where o_j is the number of observed successes and failures and e_j is the number of expected successes and failures for a given group j. If model of interest fits the data well $G_{HL} \sim \chi^2_{g-2}$. This means that when using this test we are aiming to have a large p-value to accept our model. In this blog post, I am going to explore the trivial question of proving that the Hosmer-Lemeshow Statistic follows a Chi-Square distribution for a given GLM function with a simulation study.

"Are you just proving the obvious?"

Long story short- yes, but I feel like there is what to share from this. Namely- demonstrating this follows a χ^2_{g-2} distribution with a simulation study. On a personal note, this question was quite annoying also to work on for my GLMs course so finally getting the hang of it was cathartic. So I'm sharing this so others hopefully won't have to work as hard on dealing with a problem like this (should it catch the attention of search engines).

A note about choosing number of groups

As far as choosing the number of groups g, I found a conflict in the resources I have looked into:

1. According to the text which I used to study GLMS (An Introduction to Generalized Linear Models by Dobson), usually 10 groups are used with approximately equal number of observations in each group.
2. According to StatisticsHowTo.com the number of subgroups, g, is usually calculated using the formula $g > P + 1$, where P is the number of parameters in a given model.

Overall, there is very little guidance as far as choice of number of groups is concerned. For the simulation I will be using the recommendation I was taught in my GLMs course- albeit it being just as questionable as the second option.

Running the Simulation

For running this simulation I will be making use of two libraries: `ResourceSelection` for the `hoselem.test` function and the `snpStats` package for the `qq.chisq` function. `snpStats` was not available for my R version, so I installed the package from github with the `devtools` package.

While it was possible to probably write a more vectorized version of this code, using a for-loop was suitable for visualizing this problem. There are only a total of 991 simulations done in this example so speed is not a major concern here (the strange number is due to the code).

I simulated the logistic regression problem by generating random data together that is functionally related and then modeled with logistic regression.The model is tested with the Hosmer-Lemeshow test.This is then repeated multiple times with varying sampling size, starting from a sample of size 10 and iterating to size 991. To determine if the statistics generated by the HL test are Chi-Square distributed, the test

statistics are plotted on a QQ-plot.

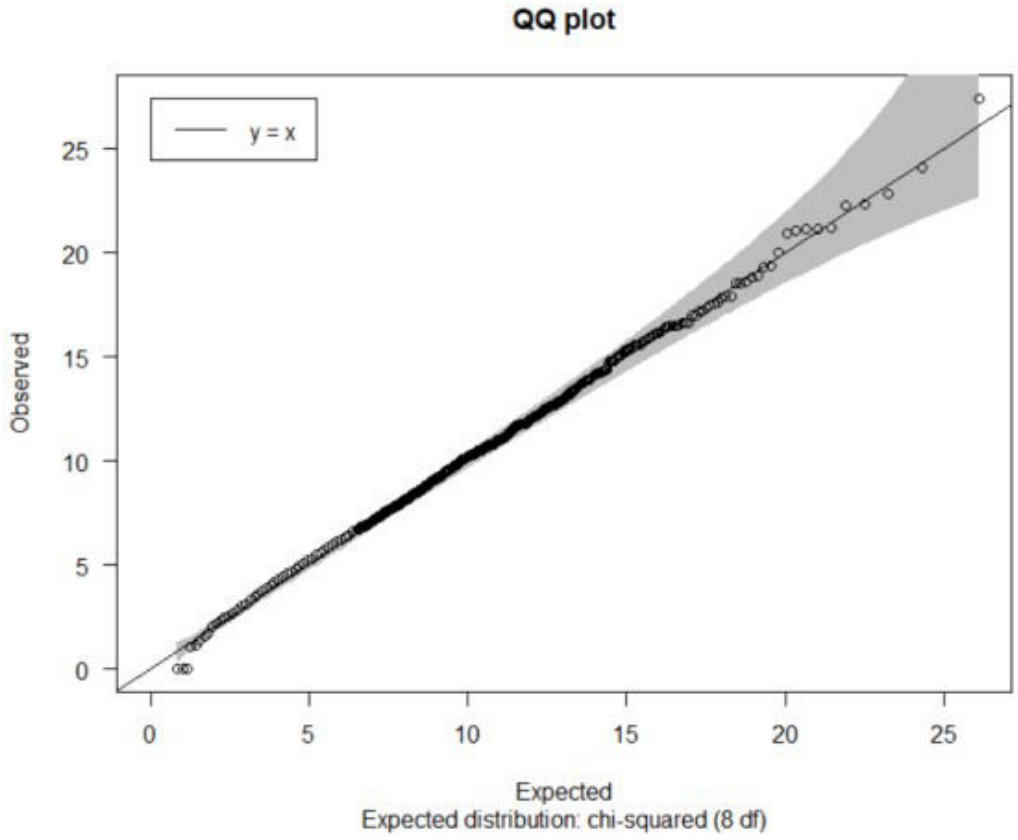
The simulation was conducted with the code below.

```
# Set Seed for reproducibility
set.seed(1234)

####################
## Load Libraries ##
####################

library("ResourceSelection")


# Define where we will be storing the generated HL statistics
# (Yes this is spaghetti code, if you have better code please send it over!)
hlvals<-c()

# Run the simulation
for (i in 10:1000){

  # Set sample size
  n<-i
  # Making data dependent on each other
  x1<-rnorm(n)
  x2<-0.5*x1+rnorm(n)
  xb<-x1-x2

  # Link function
  pr<- exp(xb)/(1+exp(xb))

  # Response variable
  y <- 1*(runif(n)
```

Now to verify the HL test statistics with the QQ-plot.

(Drum roll, please)

```
# To make a Q-Q plot for the chi-square distribution we use the snpStats
package
# Because its not available in the current builds of R. We install the package
directly from Github
# devtools::install_github("NikNakk/snpStats")
library(snpStats)

# g= 10 so our degrees of freedom are 8
qq.chisq(hlvals,df=8,slope.one = TRUE)
```




```
##           N    omitted     lambda
## 991.000000   0.000000   1.035931
```

….And there you have it! Does it look like the Hosmer-Lemeshow test follows a χ^2_{g-2} distribution? Certainly looks like it! **Water is wet!**

Conclusion

Well, that was cool diagnostic to see if the Hosmer-Lemeshow does what it claims it does. But does that mean I should use it? I was initially very impressed with this result and I wondered why I didn't see it more commonly used in Statistics and Data Analysis. After looking at StatisticsHowTo we can note the following:

- Like we said earlier, **there is very little guidance as far as choice of number of groups is concerned for conducting this test.** It seems like this is based more on heuristic than an actual rigorous prescription like other tests stats such as Wald and Student t.
- **The HL test statistic doesn't take overfitting into account and tends to have low power.** This is evident if we look at the summary of the of the

HL statistic p-values, the average and median are very low.

```
summary(1-pchisq(hlvals,8))
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.0005921 0.2410906 0.4860173 0.4870918 0.7246510 1.0000000
```