Demonstrate Hosmer-Lemeshow Test

```r
birds <- read.table(file="http://users.humboldt.edu/rizzardi/Data.dir/bird.txt", header=T,skip=15)
head( birds )
  # sv=1 is alive, sv=2 is dead
  > # ag=1 is adult, ag=2 is juvenile
  > attach( birds )
  > library(ResourceSelection)
  >
  > # R requires 0 <= y <= 1 for logistic regression
  > alive <- ifelse( sv==1, 1, 0 )
  > fit1 <- glm( alive ~ ag + tl + ae + wt + bh, family="binomial" )
  > fit1

Call:  glm(formula = alive ~ ag + tl + ae + wt + bh, family = "binomial")

Coefficients:
(Intercept)           ag           tl           ae           wt           bh
5.7516       0.2961      -0.6905       0.3141      -0.5368       1.3063

Degrees of Freedom: 86 Total (i.e. Null); 81 Residual
Null Deviance: 118
Residual Deviance: 78.18  AIC: 90.18

> #help(hoslem.test)
> phat <- fitted(fit1)
> hoslem.test(alive,phat)

Hosmer and Lemeshow goodness of fit (GOF) test

data:  alive, phat
X-squared = 14.9863, df = 8, p-value = 0.05941

> #p-value depends a lot of number of groups
> hoslem.test(alive,phat,g=8)

Hosmer and Lemeshow goodness of fit (GOF) test

data:  alive, phat
X-squared = 5.8904, df = 6, p-value = 0.4356

> hoslem.test(alive,phat,g=12)

Hosmer and Lemeshow goodness of fit (GOF) test

data:  alive, phat
X-squared = 11.1829, df = 10, p-value = 0.3435

> 
```
> hout <- hoslem.test(alive, phat)
> hout$observed

cutyhat y0 y1
[0.0127,0.108] 9 0
(0.108,0.267] 8 1
(0.267,0.424] 2 6
(0.424,0.556] 8 1
(0.556,0.631] 3 6
(0.631,0.71] 2 6
(0.71,0.823] 3 6
(0.823,0.917] 1 7
(0.917,0.974] 0 9
(0.974,0.993] 0 9

> hout$expected

cutyhat yhat0 yhat1
[0.0127,0.108] 8.4894086 0.5105914
(0.108,0.267] 7.4543476 1.5456524
(0.267,0.424] 5.5112327 2.4887673
(0.424,0.556] 4.4952329 4.5047671
(0.556,0.631] 3.5506201 5.4493799
(0.631,0.71] 2.5136464 5.4863536
(0.71,0.823] 2.1340957 6.8659043
(0.823,0.917] 1.1011904 6.8988096
(0.917,0.974] 0.6110007 8.3889993
(0.974,0.993] 0.1392249 8.8607751

> # graph hosmer-lemeshow test
> # Need to figure out bins
> row.names(hout$expected)
[1] "[0.0127,0.108]" "(0.108,0.267]" "(0.267,0.424]" "(0.424,0.556]" "(0.556,0.631]"
"(0.631,0.71]" "(0.71,0.823]" "(0.823,0.917]" "(0.917,0.974]" "(0.974,0.993]"

> hoslem.test

function (x, y, g = 10)
{
  DNAME <- paste(deparse(substitute(x)), deparse(substitute(y)),
    sep = "", ")
  METHOD <- "Hosmer and Lemeshow goodness of fit (GOF) test"
  yhat <- y
  y <- x
  qq <- unique(quantile(yhat, probs = seq(0, 1, 1/g)))
  cutyhat <- cut(yhat, breaks = qq, include.lowest = TRUE)
  observed <- xtabs(cbind(y0 = 1 - y, y1 = y) ~ cutyhat)
  expected <- xtabs(cbind(yhat0 = 1 - yhat, yhat1 = yhat) ~
    cutyhat)
  chisq <- sum((observed - expected)^2/expected)
  PVAL = 1 - pchisq(chisq, g - 2)
  PARAMETER <- g - 2
  names(chisq) <- "X-squared"
  names(PARAMETER) <- "df"
  structure(list(statistic = chisq, parameter = PARAMETER,
    p.value = PVAL, method = METHOD, data.name = DNAME, observed = observed,
    expected = expected), class = "htest")
}
```r
> yhat <- phat; y <- alive; g <- 10
> qq <- unique(quantile(yhat, probs = seq(0, 1, 1/g)))
> qq
[1] 0.01265196 0.10771889 0.26663120 0.42424692 0.55551343 0.63128451 0.71035246 0.82345758 0.91713840
[10] 0.97360352 0.99349068
> diff(qq)
[1] 0.09506693 0.15891232 0.15761572 0.13126651 0.07577108 0.07906795 0.11310512 0.09368083 0.05646512
[10] 0.01998716
> midpts <- qq[1:g]+diff(qq)/2
> midpts.logistic <- log( midpts / (1-midpts) )
> plot(x=range(predict(fit1)), y=c(0,1), type="n", xlab="eta",ylab="Prob."
> points(midpts.logistic, hout$expected[,2]/(hout$expected[,1]+hout$expected[,2]), pch=1)
> points(midpts.logistic, hout$observed[,2]/(hout$observed[,1]+hout$observed[,2]), pch=2)
> segments( x0=midpts.logistic, y0=hout$expected[,2]/(hout$expected[,1]+hout$expected[,2]),
+           x1=midpts.logistic, y1=hout$observed[,2]/(hout$observed[,1]+hout$observed[,2]) )
> abline( v=log(qq/(1-qq)), lty=2 )
> rug(predict(fit1)[alive==0],side=1)
> rug(predict(fit1)[alive==1],side=3)
```
# Demonstrate Hosmer-Lemeshow test

```r
birds <- read.table(file="http://users.humboldt.edu/rizzardi/Data.dir/bird.txt", header=T,skip=15)
head(birds)
# sv=1 is alive, sv=2 is dead
# ag=1 is adult, ag=2 is juvenile
attach(birds)
library(ResourceSelection)

# R requires 0 <= y <= 1 for logistic regression
alive <- ifelse(sv==1, 1, 0)
fit1 <- glm(alive ~ ag + tl + ae + wt + bh, family="binomial")
fit1
#help(hoslem.test)
phat <- fitted(fit1)
hoslem.test(alive,phat)

#p-value depends a lot of number of groups
hoslem.test(alive,phat,g=8)
hoslem.test(alive,phat,g=12)

hout <- hoslem.test(alive,phat)
hout$observed
hout$expected

# graph hosmer-lemeshow test
# Need to figure out bins
row.names(hout$expected)
hoslem.test
   yhat <- phat; y <- alive; g <- 10
   qq <- unique(quantile(yhat, probs = seq(0, 1, 1/g)))
   diff(qq)
   midpts <- qq[1:g]+diff(qq)/2
   midpts.logistic <- log( midpts / (1-midpts) )
   plot(x=range(predict(fit1)), y=c(0,1), type="n", xlab="eta",ylab="Prob.")
   points(midpts.logistic, hout$expected[,2]/(hout$expected[,1]+hout$expected[,2]),
          pch=1)
   points(midpts.logistic, hout$observed[,2]/(hout$observed[,1]+hout$observed[,2]),
          pch=2)
   segments( x0=midpts.logistic,
             y0=hout$expected[,2]/(hout$expected[,1]+hout$expected[,2]),
             x1=midpts.logistic,
             y1=hout$observed[,2]/(hout$observed[,1]+hout$observed[,2]) )
   abline( v=log(qq/(1-qq)), lty=2 )
   rug(predict(fit1)[alive==0],side=1)
   rug(predict(fit1)[alive==1],side=3)
```