> # R logistic regression to evaluate dogs in the 1st and 2nd rounds of the NCAA tournamen

>

> # Data from the Excel document came from ncaa.org

>

> # Entering the data and defining the variables:

>

>

> ##########

> ##

> # Reading the data into R:

>

> hoops <- read.csv("C:/Documents and Settings/Desktop/desktop/hoops/dogdata.csv", header=T)

>

> attach(hoops)

>

> # The data frame called hoops is now created,

> # with X variables, "Y", "income".

> #

> #########

> ####################################################################

>

> # fitting the regression model:

>

> hoopslogit1<- glm(X1stdogwin~RPI+SOS+winper+ft\_per+score\_margin+ft\_per+reb\_margin+to\_margin+asst\_to, + family=binomial)

>

> # getting the summary regression output:

>

> summary(hoopslogit1)

Call:

glm(formula = X1stdogwin ~ RPI + SOS + winper + ft\_per + score\_margin +

ft\_per + reb\_margin + to\_margin + asst\_to, family = binomial)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.0420 -0.6291 -0.4916 -0.3119 2.5157

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 6.839731 5.036011 1.358 0.17441

RPI -0.035334 0.014465 -2.443 0.01458 \*

SOS 0.015213 0.005975 2.546 0.01089 \*

winper -14.398293 5.271880 -2.731 0.00631 \*\*

ft\_per 4.025077 4.941256 0.815 0.41531

score\_margin -0.071790 0.092842 -0.773 0.43938

reb\_margin 0.112248 0.082541 1.360 0.17386

to\_margin -0.270509 0.119091 -2.271 0.02312 \*

asst\_to -0.674658 1.125228 -0.600 0.54879

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 278.73 on 323 degrees of freedom

Residual deviance: 257.97 on 315 degrees of freedom

AIC: 275.97

Number of Fisher Scoring iterations: 5

>

> # obtain confidence intervals:

>

> confint(hoopslogit1)

>

Waiting for profiling to be done...

2.5 % 97.5 %

(Intercept) -2.926584131 16.885600613

RPI -0.065261755 -0.008534305

SOS 0.003917552 0.027398467

winper -25.197948243 -4.473783797

ft\_per -5.586036870 13.866926390

score\_margin -0.255864368 0.109381369

reb\_margin -0.047909534 0.276798050

to\_margin -0.510702712 -0.041830579

asst\_to -2.945518439 1.482024247

> exp(hoopslogit1$coefficients)

(Intercept) RPI SOS winper ft\_per score\_margin reb\_margin to\_margin asst\_to

9.342375e+02 9.652829e-01 1.015329e+00 5.583425e-07 5.598462e+01 9.307266e-01 1.118790e+00 7.629908e-01 5.093306e-01

> lines(xrange, inv.logit(city.reg$coef[1]+ city.reg$coef[2]\*xrange))

>



>

> ############### INFERENCE IN LOGISTIC REGRESSION ######################

>

>

>

>

> # Getting the LR test statistic and P-value in R (simple logistic regression):

> anova(city.reg)

Analysis of Deviance Table

Model: binomial, link: logit

Response: Y

Terms added sequentially (first to last)

Df Deviance Resid. Df Resid. Dev

NULL 49 69.235

income 1 15.569 48 53.666

> LR.test.stat <- anova(city.reg)[2,2]; LR.test.stat

[1] 15.56875

> LR.test.df <- anova(city.reg)[2,1]

>

> LR.test.Pvalue <- 1 - pchisq(LR.test.stat, df=LR.test.df); LR.test.Pvalue

[1] 0.00007955886

>

> # Output: R provides a likelihood-ratio test of H0: beta\_1 = 0. Since the P-value

> # is very small ( < .0001), we reject H0, conclude beta\_1 is not zero, and conclude

> # that income has a significant effect on the probability a city uses TIF.

>

> est.odds.ratio <- exp(summary(city.reg)$coef["income","Estimate"])

> print(est.odds.ratio)

[1] 2.723401

>

> # getting an approximate 95% CI for the odds ratio

> # associated with Y (an indirect way):

>

> conf.level <- 0.95

> alpha <- 1 - conf.level

> b1 <- summary(city.reg)$coef["income","Estimate"]

> s.b1 <- summary(city.reg)$coef["income","Std. Error"]

> lower <- exp(b1 - qnorm(1-alpha/2)\*s.b1)

> upper <- exp(b1 + qnorm(1-alpha/2)\*s.b1)

> print(paste(100\*(1-alpha), "percent CI for odds ratio:", lower, upper))

[1] "95 percent CI for odds ratio: 1.52642800228199 4.85899894744525"

>

> # The estimates beta\_0-hat and beta\_1-hat are -11.347 and 1.002.

> # Estimated odds ratio = 2.723, and 95% CI for odds ratio is (1.526, 4.858).

>

>

> # To get, say, a 99% CI, just change the specified conf.level to 0.99.

>

> #################################################################################

>

>

> ############ Goodness of Fit: ##############

>

> ####

> # A function to do the Hosmer-Lemeshow test in R.

> # R Function is due to Peter D. M. Macdonald, McMaster University.

> #

> hosmerlem <-

+ function (y, yhat, g = 10)

+ {

+ cutyhat <- cut(yhat, breaks = quantile(yhat, probs = seq(0,

+ 1, 1/g)), include.lowest = T)

+ obs <- xtabs(cbind(1 - y, y) ~ cutyhat)

+ expect <- xtabs(cbind(1 - yhat, yhat) ~ cutyhat)

+ chisq <- sum((obs - expect)^2/expect)

+ P <- 1 - pchisq(chisq, g - 2)

+ c("X^2" = chisq, Df = g - 2, "P(>Chi)" = P)

+ }

> #

> ######

>

> # Doing the Hosmer-Lemeshow test

> # (after copying the above function into R):

>

> hosmerlem(Y, fitted(city.reg))

X^2 Df P(>Chi)

6.262964 8.000000 0.617802

>

> # The P-value will not match SAS's P-value perfectly but should be close.