**Chapter 5 practice problems**

The answers given here are sometimes only partial answers. Please see the answer keys for projects and tests for examples of full answers.

Note: Many of the practice problems are based on exercises given in Alan Agresti’s “Introduction to Categorical Data Analysis” book.

1. For the CH14TA08.csv data from Chapter 4’s practice problems, examine the deviance/df goodness-of-fit measure for the model that only uses X5 as an explanatory variable. Add the X4 explanatory variable to the model and examine if there is an improvement. Construct plots of the standardized Pearson residuals and discuss them relative to the goodness-of-fit for the models.

X5: There is a little evidence of potential problems with the model’s fit because deviance/df is 1.71 (larger than the 3 standard deviation threshold of 1.41) for the model with just X5 in it. A plot of the standardized Pearson residuals is not too bad but does include two values above the +3 threshold. These corresponding observations may be important contributing factors to the deviance/df larger value.

X4 and X5: Improvements occur when X4 is included in the model. The deviance/df is 1.40 and the 3 standard deviation threshold is 1.41. The standardized Pearson residuals are all reasonable in value.

Note that when a model has a deviance/df value greater than the 3 standard deviation threshold, this could be a sign of overdispersion (discussed in the second half of Chapter 5). This is because many residuals will need to be “larger than expected” for deviance/df to be larger than the threshold. If needed, you can relate this to a Pearson statistic because it directly contains Pearson residuals in its expression.

Below is some of the code used.

mod.fit <- glm(formula = y ~ x5, data = ch14ta08, family = poisson(link = log))

mod.fit$deviance / mod.fit$df.residual

1 + 2\*sqrt(2/mod.fit$df.residual)

1 + 3\*sqrt(2/mod.fit$df.residual)

mu.hat <- mod.fit$fitted.values

stand.resid <- rstandard(model = mod.fit, type = "pearson")

plot(x = mu.hat, y = stand.resid, xlab = expression(hat(mu)), ylab = "Standardized Pearson residuals",

ylim = c(min(c(-3, stand.resid)), max(c(3, stand.resid))))

abline(h = c(-3,-2,0,2,3), lty = "dotted", col = "red")

1. Continuing to examine the stoplight data from Chapter 4, examine if there is overdispersion. Whether or not overdispersion exists, estimate the quasi-Poisson and negative binomial models to the data. Compute confidence intervals for μ using these models and compare these intervals to those from Chapter 4.

Of course, there are no explanatory variables for this data, but this example provides a simple way to take advantage of the model estimation functions in R to still examine the data!

Poisson regression using Chapter 4 methods

> stoplight <- read.csv(file = "C:\\data\\stoplight.csv")

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

> # CIs using methods from Chapter 4

> mod.fit <- glm(formula = vehicles ~ 1, data = stoplight, family = poisson(link =

log))

> summary(mod.fit)

Call:

glm(formula = vehicles ~ 1, family = poisson(link = log), data = stoplight)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.78388 -1.05091 0.06316 0.54674 1.82985

Coefficients:

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

(Intercept) 1.35455 0.08032 16.86 <2e-16 \*\*\*

---

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

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 49.191 on 39 degrees of freedom

Residual deviance: 49.191 on 39 degrees of freedom

AIC: 172.73

Number of Fisher Scoring iterations: 4

The estimated Poisson regression model is . To examine the goodness-of-fit for the model, I calculate deviance/df:

> mod.fit$deviance / mod.fit$df.residual

[1] 1.261299

> round(c(1 + 2\*sqrt(2/mod.fit$df.residual), 1 + 3\*sqrt(2/mod.fit$df.residual)), 2)

[1] 1.45 1.68

The deviance/df value is close to 1 so there is not sufficient evidence of problems with the fit. Also, this means there is not evidence of overdispersion.

> alpha <- 0.05

> n <- length(stoplight$vehicles)

> mu.hat <- mean(stoplight$vehicles)

> wald1 <- mu.hat + qnorm(p = c(alpha/2, 1 - alpha/2))\*sqrt(mu.hat/n)

> score1 <- (mu.hat + qnorm(p = c(alpha/2, 1 - alpha/2))/(2\*n)) + qnorm(p =

c(alpha/2, 1 - alpha/2)) \* sqrt((mu.hat + qnorm(p = 1 - alpha/2)/(4\*n))/n)

> # Match Wald CI for illustrative purposes

> mod.pred <- predict(object = mod.fit, newdata = stoplight[1,], type = "response",

se = TRUE)

> mod.pred$fit + qnorm(p = c(alpha/2, 1-alpha/2))\*mod.pred$se

[1] 3.264983 4.485017

> #Function to find Wald confidence interval for mu

> ci.mu<-function(newdata, mod.fit.obj, alpha) {

lin.pred.hat<-predict(object = mod.fit.obj, newdata = newdata, type = "link",

se = TRUE)

lower<-exp(lin.pred.hat$fit-qnorm(1-alpha/2)\*lin.pred.hat$se)

upper<-exp(lin.pred.hat$fit+qnorm(1-alpha/2)\*lin.pred.hat$se)

list(lower = lower, upper = upper)

}

> #Wald on log-scale

> wald2 <- ci.mu(newdata = stoplight[1,], mod.fit.obj = mod.fit, alpha = 0.05)

> beta.ci.pois <- confint(object = mod.fit, level = 0.95)

Waiting for profiling to be done...

> profile.lr1 <- exp(beta.ci.pois)

> #Interval summary

> data.frame(name = c("Wald", "Wald-log scale", "Score", "Profile LR"),

lower = c(wald1[1], wald2$lower, score1[1], profile.lr1[1]),

upper = c(wald1[2], wald2$upper, score1[2], profile.lr1[2]))

name lower upper

1 Wald 3.264966 4.485034

2 Wald-log scale 3.310575 4.535655

3 Score 3.239503 4.510497

4 Profile LR 3.296533 4.517434

Quasi-Poisson

> mod.fit.quasi <- glm(formula = vehicles ~ 1, data = stoplight, family =

quasipoisson(link = log))

> summary(mod.fit.quasi)

Call:

glm(formula = vehicles ~ 1, family = quasipoisson(link = log),

data = stoplight)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.78388 -1.05091 0.06316 0.54674 1.82985

Coefficients:

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

(Intercept) 1.35455 0.08478 15.98 <2e-16 \*\*\*

---

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

(Dispersion parameter for quasipoisson family taken to be 1.114206)

Null deviance: 49.191 on 39 degrees of freedom

Residual deviance: 49.191 on 39 degrees of freedom

AIC: NA

Number of Fisher Scoring iterations: 4

> ci.mu(newdata = stoplight[1,], mod.fit.obj = mod.fit.quasi, alpha = 0.05)

$lower

1

3.281746

$upper

1

4.575499

The interval is similar to the intervals from the Poisson regression model. This should be expected because there was not evidence of overdispersion.

Negative binomial

> library(package = MASS)

> mod.fit.nb <- glm.nb(formula = vehicles ~ 1, data = stoplight, link = log)

> summary(mod.fit.nb)

Call:

glm.nb(formula = vehicles ~ 1, data = stoplight, link = log,

init.theta = 37.71600332)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.71610 -1.00925 0.06012 0.51842 1.71817

Coefficients:

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

(Intercept) 1.35455 0.08435 16.06 <2e-16 \*\*\*

---

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

(Dispersion parameter for Negative Binomial(37.716) family taken to be 1)

Null deviance: 45.145 on 39 degrees of freedom

Residual deviance: 45.145 on 39 degrees of freedom

AIC: 174.56

Number of Fisher Scoring iterations: 1

Theta: 37.7

Std. Err.: 98.8

2 x log-likelihood: -170.565

> #Wald interval

> beta.ci.nb1 <- confint.default(object = mod.fit.nb, level = 0.95)

> exp(beta.ci.nb1)

2.5 % 97.5 %

(Intercept) 3.284545 4.5716

> #Profile LR interval

> beta.ci.nb2 <- confint(object = mod.fit.nb, level = 0.95)

Waiting for profiling to be done...

> exp(beta.ci.nb2)

2.5 % 97.5 %

3.272003 4.555178

The intervals are similar to the intervals from the Poisson regression model. This should be expected because there was not evidence of overdispersion.