

# Homework 5

Yuki Joyama

2024-03-18

```
# data prep
df_crab = read.table("/Users/yukijoyama/Library/CloudStorage/GoogleDrive-jikeyu1995@gmail.com/My Drive/")
df_para = read.table("/Users/yukijoyama/Library/CloudStorage/GoogleDrive-jikeyu1995@gmail.com/My Drive/
```

1

(a)

I will fit a Poisson model (M1) with log link with carapace width (W) as the single predictor.

```
# M1: Poisson model with log link
m1_fit <- glm(Sa ~ W, family = poisson(link = "log"), data = df_crab)
summary(m1_fit)

##
## Call:
## glm(formula = Sa ~ W, family = poisson(link = "log"), data = df_crab)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.30476    0.54224  -6.095  1.1e-09 ***
## W            0.16405    0.01997   8.216  < 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: 632.79  on 172  degrees of freedom
## Residual deviance: 567.88  on 171  degrees of freedom
## AIC: 927.18
##
## Number of Fisher Scoring iterations: 6
```

```
exp(m1_fit$coefficients)
```

```
## (Intercept)          W
## 0.03670812  1.17826744
```

The expected number of satellites (Sa) given carapace width (W) = 0 is 0.037. With every unit increase in W, the expected number of Sa has multiplicative effect of 1.178 on  $\mu = E(Y)$ .

```
# Goodness of fit
G = sum(residuals(m1_fit, type = "pearson") ^ 2)
G
```

```
## [1] 544.157
```

```
1 - pchisq(G, 171)
```

```
## [1] 0
```

Thus, we reject the null, which indicates that the model does not have a good fit.

(b)

Now, I will fit a Poisson model (M2) with log link with carapace width (W) and weight (Wt) as predictors.

```
# M2: W and Wt as predictors
m2_fit <- glm(Sa ~ W + Wt, family = poisson(link = "log"), data = df_crab)
summary(m2_fit)
```

```
##
## Call:
## glm(formula = Sa ~ W + Wt, family = poisson(link = "log"), data = df_crab)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.29168    0.89929  -1.436  0.15091
## W           0.04590    0.04677   0.981  0.32640
## Wt          0.44744    0.15864   2.820  0.00479 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
## Null deviance: 632.79  on 172  degrees of freedom
## Residual deviance: 559.89  on 170  degrees of freedom
## AIC: 921.18
##
## Number of Fisher Scoring iterations: 6
```

```
exp(m2_fit$coefficients)
```

```
## (Intercept)          W          Wt
##    0.274809    1.046968    1.564296
```

The expected number of Sa given W = 0 and Wt = 0 is 0.275. All else being equal, with every unit increase in W, the expected number of Sa has multiplicative effect of 1.047 on  $\mu = E(Y)$ . Similarly, holding other

variables constant, with every unit increase in Wt, the expected number of Sa has multiplicative effect of 1.564 on  $\mu = E(Y)$ .

M1 is nested within M2, so I will perform ANOVA to compare the two models.

```
# compare with M1
anova(m1_fit, m2_fit)
```

```
## Analysis of Deviance Table
##
## Model 1: Sa ~ W
## Model 2: Sa ~ W + Wt
##   Resid. Df Resid. Dev Df Deviance
## 1         171      567.88
## 2         170      559.89  1    7.9934
```

The residual deviance appears to be reduced in M2 compared to M1, indicating that M2 has a better fit by adding Wt as a predictor. We need to note that the coefficient in W in M2 is not statistically significant unlike M1.

(c)

```
# over-dispersion in M2
# calculate dispersion parameter
G.stat = sum(residuals(m2_fit, type = 'pearson', data = df_crab) ^ 2) # pearson chisq
G.stat
```

```
## [1] 536.5963
```

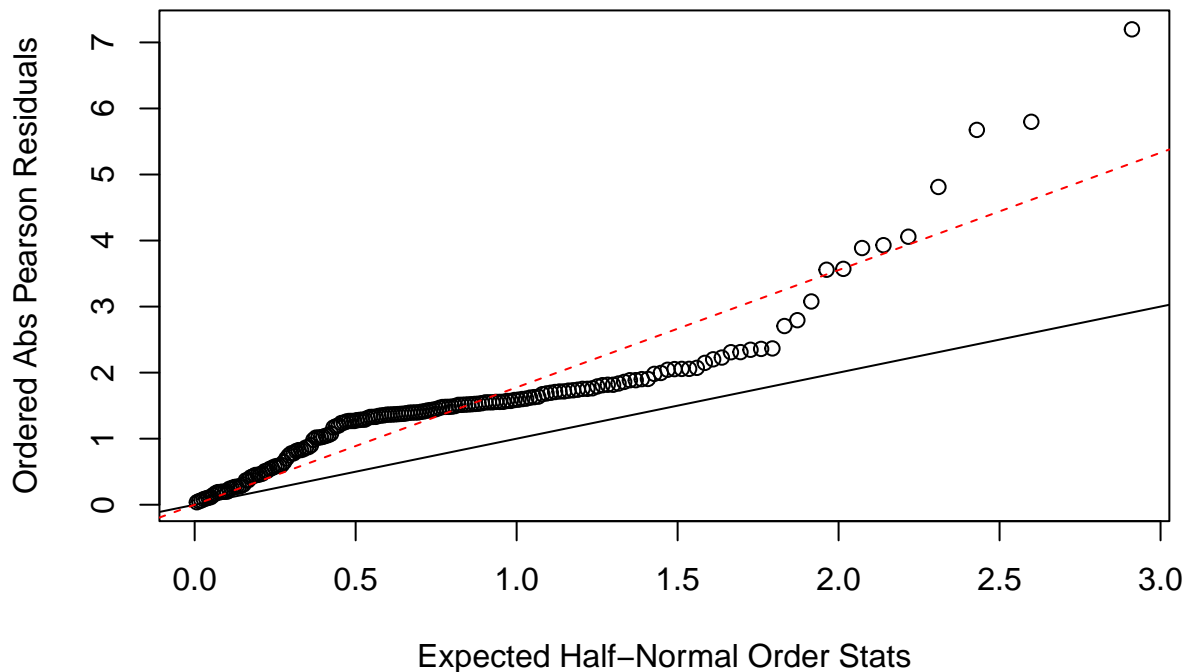
```
phi = G.stat / m2_fit$df.residual
phi
```

```
## [1] 3.156449
```

```
tilde.phi = m2_fit$deviance / m2_fit$df.residual
tilde.phi
```

```
## [1] 3.293442
```

```
# test over-dispersion (half normal plot)
res = residuals(m2_fit, type = 'pearson')
plot(qnorm((173 + 1: 173 + 0.5)/(2 * 173 + 1.125)), sort(abs(res)), xlab = 'Expected Half-Normal Order Statistic',
     abline(a = 0, b = 1)
     abline(a = 0, b = sqrt(phi), lty = 2, col = 'red'))
```



There is a linear deviation from the reference line in the half normal plot, suggesting that the response variance of the data exceeds the  $\mu$  assumed by the model.

Hence, we can say that there is over-dispersion in the original model.

The estimate of dispersion parameter:  $\hat{\phi} = 3.16$

```
# adjust for over-dispersion
summary(m2_fit, dispersion = phi)
```

```
##
## Call:
## glm(formula = Sa ~ W + Wt, family = poisson(link = "log"), data = df_crab)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.29168    1.59771  -0.808   0.419
## W             0.04590    0.08309   0.552   0.581
## Wt            0.44744    0.28184   1.588   0.112
##
## (Dispersion parameter for poisson family taken to be 3.156449)
##
##      Null deviance: 632.79  on 172  degrees of freedom
## Residual deviance: 559.89  on 170  degrees of freedom
## AIC: 921.18
##
## Number of Fisher Scoring iterations: 6
```

After adjusting for over-dispersion, The coefficient values remain the same. However, the standard errors of each variable differ from the unadjusted model. Now, all coefficients are not statistically significant.

**2**

**a**

**b**

**c**