Stat 557 Homework Assignment 3 Fall 2015

Yeng M. Chang

1. (a) Change the table like so:

	IM Cases	Controls
Tonsillectomy	40	235
No Tonsillectomy	145	420

We have

$$\hat{\phi} = \frac{40/145}{235/420} = 0.493030081,\ S_{\log(\hat{\phi})} = 0.196297792,\ z_{1-\alpha/2} = 1.96,$$

so that a 95% confidence interval for $\log(\phi)$ is given by

$$\log(\hat{\phi}) \pm 1.96 S_{\log(\hat{\phi})} = (-1.091928763, -0.322441418).$$

Exponentiate both sides to get $(0.335568637, 0.724378366)$ as an approximate 95% confidence interval for ϕ. With 95% confidence, the odds of getting IM are between 66% and 28% less among people who have had their tonsils removed.

In R, I use the following code for the percentile bootstrap method:

```r
library(boot)
x <- matrix(c(40, 145, 235, 420), nrow = 2, ncol = 2)
odds <- function(x, inds){
odds_matrix <- x[inds,1]/x[inds,2]
ratio <- odds_matrix[1]/odds_matrix[2]
return(ratio)
}
results <- boot(x, statistic=odds, R=2000)
boot.ci(results, type = 'perc', index = 1)

## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = results, type = "perc", index = 1)
##
## Intervals :
## Level     Percentile
## 95%   ( 0.4930,  2.0283 )
## Calculations and Intervals on Original Scale
```

With 95% confidence, the odds of getting IM are between 50% less and 100% greater among people who have had their tonsils removed.

(b) I use the method of p. 606 in Agresti, equation (16.27). That is, using the probability mass function

$$f(t \mid n_{1+}, n_{+1}, n, \theta) = \frac{\binom{n_{1+}}{t}\binom{n-n_{1+}}{n_{+1}-t}\theta^t}{\sum_{u=m_-}^{m_+}\binom{n_{1+}}{u}\binom{n-n_{1+}}{n_{+1}-u}\theta^u}$$

find θ_0 (the lower bound of the confidence interval) and θ_1 (the upper bound) such that

$$0.025 = \sum_{t \geq n_{11}} f(t \mid n_{1+}, n_{+1}, n, \theta_1) = \sum_{t \leq n_{11}} f(t \mid n_{1+}, n_{+1}, n, \theta_0),$$

where $m_- = \max(0, n_{1+} + n_{+1} - n)$ and $m_+ = \min(n_{1+}, n_{+1})$. I start by using Python to generate the necessary polynomial equations with respect to θ_0 and θ_1. For example:

```
from sympy import binomial, Symbol
import numpy as np
def confidence_odds(matrix, alpha):
    x = Symbol('x')
    n_11 = matrix[1,1]
    n_1_plus = matrix.sum(axis = 1)[1]
    n_plus_1 = matrix.sum(axis = 0)[1]
    n = matrix.sum()
    m_minus = max(0, n_1_plus + n_plus_1 - n)
    m_plus = min(n_1_plus, n_plus_1)
    denominator = []
    for u in range(m_minus, m_plus+1):
        denominator.append(binomial(n_1_plus, u)*
        binomial(n-n_1_plus, n_plus_1 - u)*x**u)
    numerator = []
    for t in range(n_11, n_1_plus+1):
        numerator.append(binomial(n_1_plus, t)*
        binomial(n-n_1_plus, n_plus_1 - t)*x**t)
    equation = sum(numerator)/sum(denominator) - alpha/2
    numerator_less_than = []
    for t in range(0, n_11+1):
        numerator_less_than.append(binomial(n_1_plus, t)*
        binomial(n-n_1_plus, n_plus_1 - t)*x**t)
    equation_2 = sum(numerator_less_than)/sum(denominator) - alpha/2
    return equation, equation_2

matrix = np.array([[4, 36], [5, 39]])
confidence_odds(matrix, 0.05)
```

(Lines involving products are split into two lines so as not to be cut off.) Running this program gives me two functions in terms of x, where $x = \theta_1$ for the first equation and θ_0 for the second equation. After getting the equations, I run the following code in `R` for each polynomial I get:

```
library(rootSolve)
fun <- function(x) ##insert polynomial in terms of x here
uniroot(fun, c(0.01, 50)) ##looks between 0.01 and 50 for the roots
```

This gives me roots which occur between 0.01 and 50. I do this fourteen times, two per age table. Using this method, we get the following 95% confidence intervals for each table:

Age (in years)	Confidence Interval for Odds Ratio	$\hat{\phi}$
18	$(0.1808937, 2.215413)$	0.664359861592
19	$(0.03809784, 0.7492263)$	0.207100591716
20	$(0.3926584, 2.199275)$	0.949290060852
21	$(0.1460894, 0.9426745)$	0.390350877193
22	$(0.2042759, 2.814585)$	0.811111111111
23	$(0.03487985, 2.1377)$	0.364532019704
24	$(0.1591823, 4.38628)$	0.866666666667

Using the same idea as I did in (a), I transpose the matrix and compute the odds ratio using $\hat{\phi} = \dfrac{Y_1/(n_1 - Y_1)}{Y_2/(n_2 - Y_2)}$ (as shown in the table above).

(c) In SAS, I run the following code:

```
Data set1;
Input I J K X;
Label I = Tonsillectomy
  J = Disease
  K = Age;
Datalines;
1 1 1 6
1 2 1 17
2 1 1 17
2 2 1 32
1 1 2 3
1 2 2 26
2 1 2 39
2 2 2 70
1 1 3 12
1 2 3 34
2 1 3 29
2 2 3 78
1 1 4 8
1 2 4 48
2 1 4 38
2 2 4 89
1 1 5 5
1 2 5 48
2 1 5 38
2 2 5 73
```

```
1 1 6 2
1 2 6 29
2 1 6 7
2 2 6 37
1 1 7 4
1 2 7 36
2 1 7 5
2 2 7 39
run;
proc sort data=set1; by K I J; run;

Proc print data=set1;
title 'The Mantel-Haenszel IM Data';
run;
Proc format;
value IFMT 1='Tonsillectomy'
       2='No Tonsillectomy';
value JFMT 1='IM Cases'
           2='Controls';
value KFMT 1='18'
           2='19'
           3='20'
           4='21'
           5='22'
           6='23'
           7='24';
run;
Proc freq data=set1;
Tables K*I*J / CHISQ ALL NOPERCENT NOROW;
Weight X;
Format I IFMT. J JFMT. K KFMT.;
RUN;
```

```
                      The Mantel-Haenszel IM Data

                          The FREQ Procedure

                     Summary Statistics for I by J
                           Controlling for K

                 Common Odds Ratio and Relative Risks

Statistic                 Method                Value   95% Confidence Limits
-----------------------------------------------------------------------------
Odds Ratio                Mantel-Haenszel      0.4404       0.3001       0.6463
                          Logit                0.4713       0.3173       0.7000

Relative Risk (Column 1)  Mantel-Haenszel      0.5249       0.3824       0.7206
                          Logit                0.5856       0.4263       0.8045

Relative Risk (Column 2)  Mantel-Haenszel      1.1839       1.1041       1.2696
                          Logit                1.1756       1.1002       1.2561
```

The M-H estimator for the odds ratio is 0.4404, with a 95% confidence interval of $(0.3001, 0.6463)$. With 95% confidence, the odds of getting IM are between 70% less and 35.3% less among people who have their tonsils removed.

(d) I run the following code in SAS:

```
Proc freq data=set1;
Tables K*I*J ;
EXACT COMOR;
Weight X;
Format I IFMT. J JFMT. K KFMT.;
RUN;
```

```
                    The Mantel-Haenszel IM Data

                         The FREQ Procedure

                  Summary Statistics for I by J
                         Controlling for K

                         Common Odds Ratio
               ------------------------------------
               Mantel-Haenszel Estimate      0.4404

               Asymptotic Conf Limits
               95% Lower Conf Limit          0.3001
               95% Upper Conf Limit          0.6463

               Exact Conf Limits
               95% Lower Conf Limit          0.2927
               95% Upper Conf Limit          0.6539
```

The exact confidence interval is given by $(0.2927, 0.6539)$. With 95% confidence, the odds of getting IM are between 70.7% less and 35.3% less among people who have their tonsils removed. The exact confidence interval is quite different from the confidence interval in (a). It is essentially the confidence interval in (a) translated left.

(e)

(f) (i)

```
                        Breslow-Day Test for
                  Homogeneity of the Odds Ratios
                  ------------------------------
                  Chi-Square                9.3165
                  DF                             6
                  Pr > ChiSq                0.1565
```

At the 5% level of significance, we do not have evidence that the odds ratios are not homogeneous.

(ii)

(g) In R:

```
array <- array(c(6, 17, 17, 32, 3, 39, 26, 70, 12, 29, 34, 78, 8, 38,
48, 89, 5, 10, 45, 73, 2, 7, 29, 37, 4, 5, 36, 39), c(2, 2, 7))
mantelhaen.test(array, conf.level=0.95)

##
##  Mantel-Haenszel chi-squared test with continuity correction
##
## data:  array
## Mantel-Haenszel X-squared = 8.4754, df = 1, p-value = 0.0036
## alternative hypothesis: true common odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.3708204 0.8142302
## sample estimates:
## common odds ratio
##         0.5494845
```

At the 5% level of significance, we have evidence that tonsillectomy rates differ for IM cases and controls within every age group.

2. (a) Assuming ab corresponds to Tall/Cut, aB corresponds to Tall/Potato, Ab corresponds to Dwarf/Cut, and AB corresponds to Dwarf/Potato, here is the corresponding R code:

```
data <- matrix(c(926,467,693,288,151,234,293,150,219,104,47,70),nrow=3,ncol=4)
proportions <- matrix(rep(c(9/16,3/16,3/16,1/16),3),byrow=T,nrow=3,ncol=4)
row_totals <- matrix(c(rep(sum(data[1,]),4), rep(sum(data[2,]),4),
rep(sum(data[3,]),4)), nrow = 3, ncol = 4, byrow = T)
expected <- row_totals * proportions
sum(data*log(data/expected))*2 ##deviance

## [1] 3.143858

pchisq(sum(data*log(data/expected))*2, df = 1, lower.tail = FALSE)

## [1] 0.07621335
```

Degrees of freedom are 1, and the p-value is 0.076, so at the 5% level of significance, we do not have evidence that the data follow the general alternative.

(b) Degrees of freedom are 2. Using maximum likelihood estimation, suppose $Y_i = (Y_{i1}, \ldots, Y_{i4})$ is a single row and $n = \sum_j Y_{ij}$. Then

$$L(p_a, p_b) = \frac{n!}{\prod_j Y_{ij}!} p_a^{Y_{i1}+Y_{i3}} p_b^{Y_{i1}+Y_{i2}} (1-p_a)^{Y_{i2}+Y_{i4}} (1-p_b)^{Y_{i3}+Y_{i4}}.$$

Setting $c = \dfrac{n!}{\prod_j Y_{ij}!}$, we have

$$\ell(p_a, p_b) = \log(c) + (Y_{i1}+Y_{i3})\log(p_a) + (Y_{i1}+Y_{i2})\log(p_b) + (Y_{i2}+Y_{i4})\log(1-p_a) + (Y_{i3}+Y_{i4})\log(1-p_b).$$

Taking the first partial with respect to p_a and setting it equal to 0, it can be shown that

$$p_a = \frac{Y_{i1} + Y_{i3}}{n}.$$

Similarly,

$$p_b = \frac{Y_{i1} + Y_{i2}}{n}.$$

Using R to compute these quantities, we have the following:

```
proportions <- matrix(0, nrow = 3, ncol = 4)
for (i in 1:3){
n <- sum(data[i,])
p_a <- (data[i,1]+data[i,3])/n
p_b <- (data[i,1]+data[i,2])/n
proportions[i,1] <- p_a * p_b
proportions[i,2] <- (1-p_a)*p_b
proportions[i,3] <- p_a * (1-p_b)
proportions[i,4] <- (1-p_a)*(1-p_b)
cat('Region', i, ': ','\n')
cat('p_a: ',p_a, '\n')
cat('p_b: ',p_b, '\n')
}
```

```
## Region 1 :
## p_a:  0.7566729
## p_b:  0.7535692
## Region 2 :
## p_a:  0.7570552
## p_b:  0.7582822
## Region 3 :
## p_a:  0.75
## p_b:  0.7623355
```

```
proportions
```

```
##           [,1]      [,2]      [,3]       [,4]
## [1,] 0.5702054 0.1833638 0.1864675 0.05996330
## [2,] 0.5740615 0.1842207 0.1829937 0.05872408
## [3,] 0.5717516 0.1905839 0.1782484 0.05941612
```

```
expected <- row_totals * proportions
sum(data*log(data/expected))*2 ##deviance
```

```
## [1] 1.133048
```

```
pchisq(sum(data*log(data/expected))*2, df = 2, lower.tail = FALSE)
```

```
## [1] 0.5674946
```

Degrees of freedom are 2, and the p-value is 0.57, so at the 5% level of significance, we do not have evidence that the data follow the general alternative.

(c) Since each row has an independent multinomial distribution, we can write in particular (where n_i is the total number in the ith row)

$$\begin{aligned}
L(p_a, p_b) &= \frac{\prod_i n_i!}{\prod_i \prod_j Y_{ij}!} \prod_i p_a^{Y_{i1}+Y_{i3}} p_b^{Y_{i1}+Y_{i2}} (1-p_a)^{Y_{i2}+Y_{i4}} (1-p_b)^{Y_{i3}+Y_{i4}} \\
&= \frac{\prod_i n_i!}{\prod_i \prod_j Y_{ij}!} p_a^{\sum_i (Y_{i1}+Y_{i3})} p_b^{\sum_i (Y_{i1}+Y_{i2})} (1-p_a)^{\sum_i (Y_{i2}+Y_{i4})} (1-p_b)^{\sum_i (Y_{i3}+Y_{i4})} \\
&\propto p_a^{\sum_i (Y_{i1}+Y_{i3})} p_b^{\sum_i (Y_{i1}+Y_{i2})} (1-p_a)^{\sum_i (Y_{i2}+Y_{i4})} (1-p_b)^{\sum_i (Y_{i3}+Y_{i4})}.
\end{aligned}$$

Furthermore,

$$\frac{\partial \ell}{\partial p_a} = \frac{\sum_i (Y_{i1}+Y_{i3})}{p_a} - \frac{\sum_i (Y_{i2}+Y_{i4})}{1-p_a} = 0$$

implies that

$$p_a = \frac{\sum_i (Y_{i1}+Y_{i3})}{N}$$

where $N = \sum_{i,j} Y_{ij}$. Similarly,

$$p_b = \frac{\sum_i (Y_{i1}+Y_{i2})}{N}.$$

In R:

```r
proportions <- matrix(0, nrow = 3, ncol = 4)
N <- sum(data)
p_a <- (sum(data[,1])+sum(data[,3]))/N
p_b <- (sum(data[,1])+sum(data[,2]))/N
for (i in 1:3){
proportions[i,1] <- p_a * p_b
proportions[i,2] <- (1-p_a)*p_b
proportions[i,3] <- p_a * (1-p_b)
proportions[i,4] <- (1-p_a)*(1-p_b)
cat('Region', i, ': ','\n')
cat('p_a: ',p_a, '\n')
cat('p_b: ',p_b, '\n')
}
```

```
## Region 1 :
## p_a:  0.7545305
## p_b:  0.7575508
## Region 2 :
## p_a:  0.7545305
## p_b:  0.7575508
## Region 3 :
## p_a:  0.7545305
## p_b:  0.7575508
```

```r
expected <- row_totals * proportions
sum(data*log(data/expected))*2 ##deviance
```

```
## [1] 1.628537

pchisq(sum(data*log(data/expected))*2, df = 3, lower.tail = FALSE)

## [1] 0.6529368
```

At the 5% level of significance, we do not have evidence that the data follow the general alternative.

(d)

Comparison	d.f.	deviance value	p-value
Model A vs. General	1	3.143858	0.076
Model B vs. General	2	1.133048	0.567
Model C vs. General	3	1.628537	0.653

(e) It looks like Model C has the best fit to the data.

3. We have $\ell(\mu) = -n\mu + \log(\mu)\sum Y_i - \sum \log(Y_i!)$. The score function is

$$u = \frac{\partial \ell}{\partial \mu} = -n + \frac{\sum Y_i}{\mu} = \frac{\sum Y_i - n\mu}{\mu}.$$

The second partial is $\dfrac{-\sum Y_i}{\mu^2}$, giving an expected information of $\sum E[Y_i]/\mu^2 = n/\mu$. Thus,

$$\mu^{(1)} = \mu^{(0)} + \frac{\mu^{(0)}}{n}\left(\frac{\sum y_i - n\mu^{(0)}}{\mu^{(0)}}\right) = \frac{\sum_i y_i}{n} = \bar{y}.$$

Hence the algorithm converges. Using Newton-Raphson,

$$\mu^{(1)} = \mu^{(0)} + \frac{(\mu^{(0)})^2}{\sum Y_i}\left(\frac{\sum Y_i - n\mu^{(0)}}{\mu^{(0)}}\right) = \mu^{(0)} + \frac{\mu^{(0)}}{\bar{y}}\left(\bar{y} - \mu^{(0)}\right) = 2\mu^{(0)} + \left(\mu^{(0)}\right)^2/\bar{y}.$$

4. $f(y,k,\mu) = \exp\left[\underbrace{\log\left(\frac{\Gamma(y+k)}{\Gamma(k)\Gamma(y+1)}\right)}_{c(y,\theta)} + k\underbrace{\log\left(\frac{k}{y+k}\right)}_{-b(\theta)} + y\underbrace{\log\left(\frac{\mu}{\mu+k}\right)}_{\theta}\right]$. If k is not known, then f is not of the exponential family.

5. (a)

```
teggll <- read.table("C:/Users/Yeng Chang/Desktop/Stat 557/Homework 3/teggll.dat"
header = FALSE, col.names=c("Box", "Temperature", "Females", "Males", "Total"))
teggll$female_hatching[teggll$Females >= 1] <- 1
teggll$female_hatching[teggll$Females < 1] <- 0
teggll$female_hatching <- factor(teggll$female_hatching)
tegl <- glm(female_hatching ~ Temperature, data = teggll,
weights = Females, x = TRUE, trace = TRUE, family = binomial(link = logit))

## Deviance = 4.337061 Iterations - 1
## Deviance = 1.567939 Iterations - 2
## Deviance = 0.5732229 Iterations - 3
## Deviance = 0.2103978 Iterations - 4
## Deviance = 0.07733653 Iterations - 5
## Deviance = 0.0284418 Iterations - 6
## Deviance = 0.01046198 Iterations - 7
## Deviance = 0.003848587 Iterations - 8
## Deviance = 0.001415794 Iterations - 9
## Deviance = 0.0005208387 Iterations - 10
## Deviance = 0.0001916055 Iterations - 11
## Deviance = 7.048765e-05 Iterations - 12
## Deviance = 2.593095e-05 Iterations - 13
## Deviance = 9.539463e-06 Iterations - 14
## Deviance = 3.509372e-06 Iterations - 15
## Deviance = 1.291026e-06 Iterations - 16
## Deviance = 4.749419e-07 Iterations - 17
## Deviance = 1.747214e-07 Iterations - 18
## Deviance = 6.427641e-08 Iterations - 19
## Deviance = 2.364597e-08 Iterations - 20
## Deviance = 8.698868e-09 Iterations - 21
## Deviance = 3.200138e-09 Iterations - 22
## Deviance = 1.177273e-09 Iterations - 23
## Deviance = 4.330984e-10 Iterations - 24
```

```
summary(tegl)
```

```
##
## Call:
## glm(formula = female_hatching ~ Temperature, family = binomial(link = logit),
##     data = teggll, weights = Females, x = TRUE, trace = TRUE)
##
## Deviance Residuals:
##       Min         1Q     Median         3Q        Max
## 0.000e+00  4.969e-06  5.549e-06  5.950e-06  8.168e-06
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 1.842e+01  1.497e+06       0        1
## Temperature 2.921e-01  5.256e+04       0        1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance:       NaN  on 13  degrees of freedom
## Residual deviance: 4.331e-10  on 12  degrees of freedom
## AIC: 4
##
## Number of Fisher Scoring iterations: 24
```

We have $\hat{\beta}_0 = 1.84$ and $\hat{\beta}_1 = 0.29$, with standard errors $1.497 \cdot 10^6$ and $5.256 \cdot 10^4$

respectively.

(b)

```
b <- coef(tegl)
bcov <- vcov(tegl)
bse <- sqrt(diag(bcov))
z975 <- qnorm(0.975)
bci <- matrix(c(b-z975*bse, b+z975*bse), ncol=2)
bci
```

```
##              [,1]      [,2]
## [1,] -2933193.6 2933230.4
## [2,]  -103015.2  103015.8
```

```
or <- exp(b)
or
```

```
##  (Intercept)  Temperature
## 1.001347e+08 1.339245e+00
```

```
cior <- exp(bci)
cior
```

```
##      [,1] [,2]
## [1,]    0  Inf
## [2,]    0  Inf
```