

STA312f22 Assignment 9 Printout

```

> # Titanic data for STA312f22 Assignment 9. Use the table
> rm(list=ls()); options(scipen=999)
>
> # Prepare the data the way I did it in lecture for Berkeley
> ttable = as.data.frame(Titanic)
> ttable
  Class    Sex  Age Survived Freq
1   1st   Male Child       No    0
2   2nd   Male Child       No    0
3   3rd   Male Child       No   35
4  Crew   Male Child       No    0
5   1st Female Child       No    0
6   2nd Female Child       No    0
7   3rd Female Child       No   17
8  Crew Female Child       No    0
9   1st   Male Adult       No  118
10  2nd   Male Adult       No  154
11  3rd   Male Adult       No  387
12 Crew   Male Adult       No  670
13  1st Female Adult       No    4
14  2nd Female Adult       No   13
15  3rd Female Adult       No   89
16 Crew Female Adult       No    3
17  1st   Male Child      Yes    5
18  2nd   Male Child      Yes   11
19  3rd   Male Child      Yes   13
20 Crew   Male Child      Yes    0
21  1st Female Child      Yes    1
22  2nd Female Child      Yes   13
23  3rd Female Child      Yes   14
24 Crew Female Child      Yes    0
25  1st   Male Adult      Yes   57
26  2nd   Male Adult      Yes   14
27  3rd   Male Adult      Yes   75
28 Crew   Male Adult      Yes  192
29  1st Female Adult      Yes  140
30  2nd Female Adult      Yes   80
31  3rd Female Adult      Yes   76
32 Crew Female Adult      Yes   20
> 75/(75+387) # Proportion 3d class adult male survived = 0.1623377
[1] 0.1623377
>
> ttable = subset(ttable, Age=='Adult')
> # Assemble a data frame with one line per person
> adults = NULL
> nrows = dim(ttable)[1]
> for(j in 1:nrows)
+   {
+     oneline = ttable[j,c(1,2,4)]
+     people = ttable[j,5]
+     for(i in 1:people) adults = rbind(adults,oneline)
+   } # Next row of ttable
> c(dim(adults)[1],sum(ttable$Freq))
[1] 2092 2092
> nadults = dim(adults)[1]
> adults = within(adults,{
+ y = numeric(nadults); y[Survived == 'Yes'] = 1
+ })

```

```

> with(adults, table(Survived, y, useNA = 'ifany'))
      y
Survived  0    1
      No 1438    0
      Yes   0  654
>
> # a) Class (including crew) by survival for adult males.
> men = subset(adults, Sex=='Male'); dim(men)
[1] 1667    4
> with(adults, {table(Sex)})
Sex
  Male Female
1667    425
>
> # a i) LR test of class by survival from a contingency table
> with(men, {
+ classurv = table(Class, Survived); print(classurv)
+ cat('\n')
+ print(prop.table(classurv, margin=1))
+ cat('\n')
+ ctest = chisq.test(classurv, correct=FALSE)
+ muhat = ctest$expected
+ Gsq = 2 * sum(classurv*log(classurv/muhat))
+ dfree = ctest$parameter # 3
+ pval = 1-pchisq(Gsq, dfree)
+ output = c(Gsq, dfree, pval)
+ names(output) = c('G-squared', 'df', 'p-value')
+ print(output)
+ cat('\n')
+ })
      Survived
Class   No Yes
1st    118  57
2nd    154  14
3rd    387  75
Crew   670 192

      Survived
Class      No      Yes
1st  0.67428571 0.32571429
2nd  0.91666667 0.08333333
3rd  0.83766234 0.16233766
Crew 0.77726218 0.22273782

      G-squared      df      p-value
39.61020131810426  3.00000000000000  0.00000001288766
>
> # Contingency table
> #      G-squared      df      p-value
> # 39.61020131810426  3.00000000000000  0.00000001288766
>
> # a ii) Logistic regression with dummy vars
> # Making Crew reference
> men = within(men, { contrasts(Class) = contr.treatment(4, base=4) })
>

```

```

> boys = glm(y ~ Class, family=binomial, data=men)
> summary(boys)

Call:
glm(formula = y ~ Class, family = binomial, data = men)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.8878  -0.7099  -0.5952  -0.4172   2.2293

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.24978    0.08186 -15.268 < 0.0000000000000002 ***
Class1         0.52215    0.18088   2.887    0.00389 **
Class2        -1.14811    0.29079  -3.948    0.0000787 ***
Class3        -0.39115    0.15039  -2.601    0.00930 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1681.0  on 1666  degrees of freedom
Residual deviance: 1641.4  on 1663  degrees of freedom
AIC: 1649.4

Number of Fisher Scoring iterations: 4

> # ls(boys)
> G2 = boys$null.deviance - boys$deviance; G2
[1] 39.6102
> anova(boys, test='Chisq') # Same result.
Analysis of Deviance Table

Model: binomial, link: logit

Response: y

Terms added sequentially (first to last)

      Df Deviance Resid. Df Resid. Dev      Pr(>Chi)
NULL               1666      1681.0
Class  3       39.61      1663      1641.4 0.00000001289 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
> # a iii) and iv) Look at the summary.
>

```

```

> #####
>
> # b) Test class by sex interaction, just adult passengers
> apass = subset(adults, Class != 'Crew'); dim(apass)
[1] 1207      4
> with(adults, table(Class))
Class
 1st  2nd  3rd Crew
 319  261  627  885
> 319 + 261 + 627 # 1207, okay
[1] 1207
> # Effect coding and a combination variable too
> apass = within(apass, {
+ Class = factor(Class) # Getting rid of ghost category crew.
+ contrasts(Class) = contr.sum
+ contrasts(Sex) = contr.sum
+ combo = paste(Sex, Class, sep='')
+ }) # End with adult passengers
> # Checking apass
> head(apass)
  Class Sex Survived y   combo
9    1st Male      No 0 Male1st
91   1st Male      No 0 Male1st
92   1st Male      No 0 Male1st
93   1st Male      No 0 Male1st
94   1st Male      No 0 Male1st
95   1st Male      No 0 Male1st
> with(apass, {
+ print(contrasts(Sex)); cat('\n')
+ print(contrasts(Class)); cat('\n')
+ print(table(Class, combo, useNA = 'ifany'))
+ }) # End checking apass
      [,1]
Male      1
Female   -1

      [,1] [,2]
1st      1    0
2nd      0    1
3rd     -1   -1

      combo
Class Female1st Female2nd Female3rd Male1st Male2nd Male3rd
1st      144      0      0      175      0      0
2nd       0      93      0      0      168      0
3rd       0      0     165      0      0     462
>
> # b i) Take a look at survival proportions using tapply
> with(apass, {
+ stable = tapply(y, list(Sex, Class), FUN=mean)
+ print(round(stable, 3)); cat('\n')
+ })
      1st    2nd    3rd
Male  0.326 0.083 0.162
Female 0.972 0.860 0.461

```

```

>
> # b ii) Test interaction with effect coding
> ecmodel = glm(y ~ Sex*Class, family=binomial, data=apass)
> summary(ecmodel)

Call:
glm(formula = y ~ Sex * Class, family = binomial, data = apass)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6771  -0.5952  -0.5952   0.3931   2.2293

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.07468    0.11676   0.640   0.52247
Sex1        -1.66350    0.11676 -14.247 < 0.0000000000000002 ***
Class1       1.33918    0.19295   6.940   0.000000000000391 ***
Class2      -0.36509    0.16607  -2.198   0.02792 *
Sex1:Class1 -0.47799    0.19295  -2.477   0.01324 *
Sex1:Class2 -0.44399    0.16607  -2.674   0.00751 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1585.8  on 1206  degrees of freedom
Residual deviance: 1066.6  on 1201  degrees of freedom
AIC: 1078.6

Number of Fisher Scoring iterations: 6

> anova(ecmodel, test='Chisq')
Analysis of Deviance Table

Model: binomial, link: logit

Response: y

Terms added sequentially (first to last)


```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
NULL			1206	1585.8	
Sex	1	359.68	1205	1226.1	< 0.00000000000000022 ***
Class	2	95.40	1203	1130.7	< 0.00000000000000022 ***
Sex:Class	2	64.07	1201	1066.6	0.0000000000000122 ***

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # Get that Gsq = 64.07 with full-reduced
> ecred = glm(y ~ Sex + Class, family=binomial, data=apass)
> anova(ecred,ecmodel, test='Chisq')
Analysis of Deviance Table

Model 1: y ~ Sex + Class
Model 2: y ~ Sex * Class
    Resid. Df Resid. Dev Df Deviance      Pr(>Chi)
1      1203      1130.7
2      1201      1066.6  2    64.074 0.0000000000000122 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>

```

```

> # Wald test of the interaction
> # For Wald tests: Wtest = function(L,Tn,Vn,h=0) # H0: L theta = h
> source("http://www.utstat.utoronto.ca/brunner/Rfunctions/Wtest.txt")
> Linter = rbind(c(0,0,0,0,1,0),
+               c(0,0,0,0,0,1))
> Wtest(Linter,coefficients(ecmodel),vcov(ecmodel))

```

	W	df	p-value
	52.26095656290280544	2.000000000000000000	0.000000000000448408

```

>
> # b iii) Take a look at the odds ratios
>
> with(apass,{
+ print(table(Sex,Survived,Class)); cat('\n')
+ stable = tapply(y,list(Sex,Class), FUN=mean)
+ # print(stable); cat('\n') # Did this above
+ om = stable[1,]/(1-stable[1,])
+ of = stable[2,]/(1-stable[2,])
+ print(of/om)
+ })
, , Class = 1st

      Survived
Sex      No Yes
Male    118  57
Female   4  140

, , Class = 2nd

      Survived
Sex      No Yes
Male    154  14
Female  13   80

, , Class = 3rd

      Survived
Sex      No Yes
Male    387  75
Female  89   76

      1st      2nd      3rd
72.456140 67.692308 4.406292
> # First class manually from table
> of = 140/144 / (4/144); of
[1] 35
> om = 57/118; om
[1] 0.4830508
> of/om # 72.45614
[1] 72.45614
>

```

```

> # b iv) Cell means model for pairwise comparison of odds ratios.
> cmmmodel = glm(y ~ 0 + combo, family=binomial, data=apass)
> summary(cmmmodel)

Call:
glm(formula = y ~ 0 + combo, family = binomial, data = apass)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6771  -0.5952  -0.5952   0.3931   2.2293

Coefficients:
            Estimate Std. Error z value      Pr(>|z|)
comboFemale1st   3.5553    0.5071   7.011 0.000000000000236 ***
comboFemale2nd   1.8171    0.2990   6.076 0.000000000122887 ***
comboFemale3rd  -0.1579    0.1562  -1.011    0.312
comboMale1st    -0.7276    0.1613  -4.511 0.000000645253117 ***
comboMale2nd    -2.3979    0.2791  -8.590 < 0.0000000000000002 ***
comboMale3rd    -1.6409    0.1262 -13.006 < 0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1673.3  on 1207  degrees of freedom
Residual deviance: 1066.6  on 1201  degrees of freedom
AIC: 1078.6

Number of Fisher Scoring iterations: 6

> # First get those estimated odds ratios from (b iii) another way.
> betahat = coefficients(cmmmodel); betahat
comboFemale1st comboFemale2nd comboFemale3rd  comboMale1st  comboMale2nd
comboMale3rd
    3.5553481      1.8170773      -0.1579030      -0.7276334      -2.3978953
-1.6409366
> or1 = exp(betahat[1]-betahat[4])
> or2 = exp(betahat[2]-betahat[5])
> or3 = exp(betahat[3]-betahat[6])
> c(or1,or2,or3)
comboFemale1st comboFemale2nd comboFemale3rd
    72.456140      67.692308      4.406292
> # Now the pairwise comparisons.
> one = c(1, 0, 0,-1, 0, 0)
> two = c(0, 1, 0, 0,-1, 0)
> three = c(0, 0, 1, 0, 0,-1)
> oneVstwo = rbind(one-two); oneVstwo
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1    -1     0    -1     1     0
> oneVsthree = rbind(one-three); oneVsthree
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1     0    -1    -1     0     1
> twoVsthree = rbind(two-three); twoVsthree
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    0     1    -1     0    -1     1
> Wtest(oneVstwo,coefficients(cmmmodel),vcov(cmmmodel)) # 1 vs. 2
      W      df      p-value
0.01026673 1.00000000 0.91929258
> Wtest(oneVsthree,coefficients(cmmmodel),vcov(cmmmodel)) # 1 vs. 3
      W      df      p-value
24.2362479853382 1.00000000000000 0.00000008521334
> Wtest(twoVsthree,coefficients(cmmmodel),vcov(cmmmodel)) # 2 vs. 3
      W      df      p-value
35.941553797028597 1.000000000000000 0.000000002033258

```

```

> # This might look fishy, but the standard error for Female 1st
> # class is quite large.
> diag(vcov(cmmmodel)) # This will affect all comparisons involving 1st class.
comboFemale1st comboFemale2nd comboFemale3rd  comboMale1st  comboMale2nd
comboMale3rd
0.25714077      0.08942308      0.02439385      0.02601844      0.07792208
0.01591731
>
> # b v) See summary. Those tests are comparing survival to 50%.
>
> # b vi) Recover the estimated survival probabilities
> exp(betahat)/(1+exp(betahat))
comboFemale1st comboFemale2nd comboFemale3rd  comboMale1st  comboMale2nd
comboMale3rd
0.97222222      0.86021505      0.46060606      0.32571429      0.08333333
0.16233766
>
>

```