CH10: Analysis of Categorical Data

Other useful R functions for CH10 include:
1. prop.test() and binom.test() for large and small sample tests of a single proportion.
2. prop.test() and power.prop.test() for a large sample test comparing two proportions and corresponding power calculation.
3. mcnemar.test() for McNemar's test of paired proportions.

1 Creating Tables

Many times in these examples, we will start from summarized counts or tables. This is a handy, concise presentation of the data.

But in real life, it is much more common for data to start in a data.frame (or Excel spreadsheet!).

It is relatively easy to summarize into counts or tables using table() in R. Another option is to use Pivot Tables in Excel.

We will use the Birds data as an example of FET (later in these examples). For now, we simply create some summary tables.

```
Birds <- read.csv("C:/hess/STAT511_FA11/RData/CH10_Birds.csv")
head(Birds)
```

```
##   ID Type Disc
## 1  1 Blue  Yes
## 2  2 Blue  Yes
## 3  3 Blue  Yes
## 4  4 Blue  Yes
## 5  5 Blue   No
## 6  6 Blue   No
```

```
table(Birds$Disc)
```

```
##
##  No Yes
##  15   7
```

```
table(Birds$Type, Birds$Disc)
```

```
##
##        No Yes
##   Blue  6   4
##   Gold  9   3
```

2 Maize Example: Chi-square Goodness of Fit

The chi-square goodness of fit (GOF) test is used to compare observed proportions (or counts) for a single categorical variable to some expected probabilities under H0. The more common test is the chi-square test for contingency tables (see the next section).

In this example from Ott&Longnecker, we test H0: π_1=9/16, π_2=3/16, π_3=3/16, π_4=1/16. These null hypothesized probabilities are motivated by Mendel's laws.

Note: In practice, the hypothesized probabilities (or proportions) would be motivated by the research question.

```
chisq.test(c(773, 231, 238, 59), p = c(9/16, 3/16, 3/16, 1/16), correct = FALSE)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  c(773, 231, 238, 59)
## X-squared = 9.2714, df = 3, p-value = 0.02589
```

```
#Calculating Pearson Residuals
Counts <- c(773, 231, 238, 59)
Props <- c(9/16, 3/16, 3/16, 1/16)
Total <- sum(Counts)
Total
```

```
## [1] 1301
```

```
Exp <- Props*Total
Exp
```

```
## [1] 731.8125 243.9375 243.9375  81.3125
```

```
Resid <- Counts-Exp
SEResid <- sqrt(Total*Props*(1-Props))
PearsonResids <- Resid/SEResid
PearsonResids
```

```
## [1]  2.3018472 -0.9189659 -0.4217476 -2.5555474
```

```
rm(Counts, Props, Total, Exp, Resid, SEResid, PearsonResids)
```

3 Chi-square Test for Contingency Tables and Fisher's Exact Test

The chi-square test for contingency tables is used to test for an association between row and column variables. If sample size is small (see warning generated for the Birds Example), then Fisher's Exact test (FET) is preferred.

Notes:

1. For most of these examples, we start from a summarized table of counts (constructed using the matrix() function). But in practice, it is much more common for data to start in a data.frame. However, it is relatively easy to summarize into counts or tables using table() in R. See the Birds data for an example.
2. In these examples, I use correct = FALSE with the chisq.test function to match hand calculations from the notes. But in practice, I am fine with the default continuity correction!

3.1 French Skiers Example

```
Skiers <- matrix(c(109, 31, 122, 17), nrow = 2, byrow = TRUE)
colnames(Skiers) <- c("NoCold", "YesCold")
rownames(Skiers) <- c("Placebo", "VitC")
Skiers
```

```
##         NoCold YesCold
## Placebo    109      31
## VitC       122      17
```

```
prop.table(Skiers, 1)
```

```
##            NoCold   YesCold
## Placebo 0.7785714 0.2214286
## VitC    0.8776978 0.1223022
```

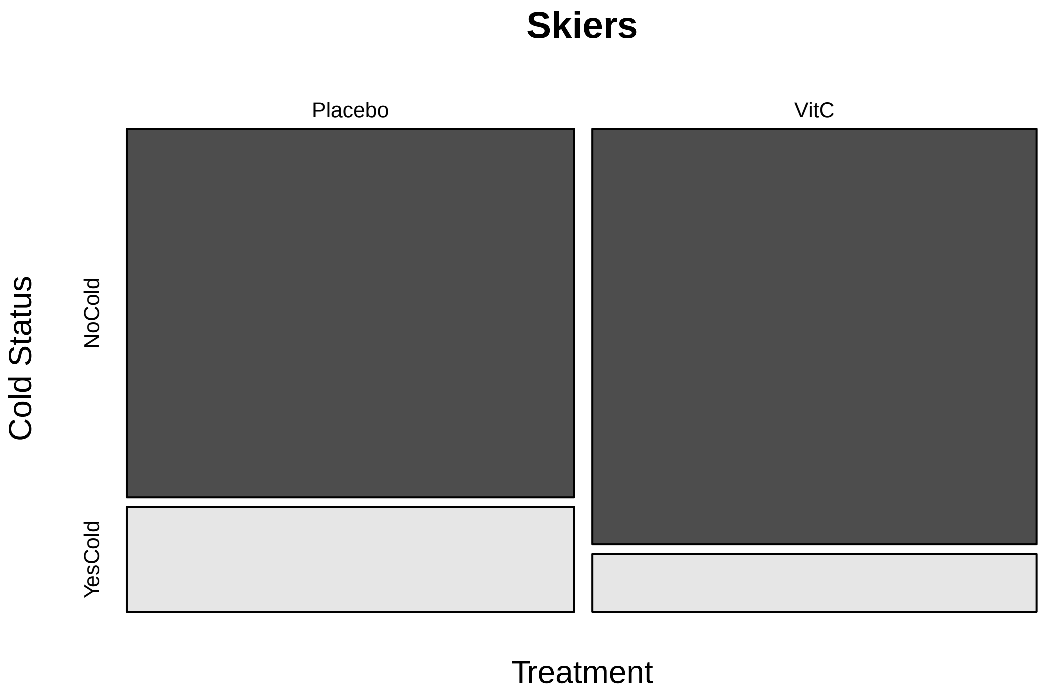
```
chisq.test(Skiers, correct = FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  Skiers
## X-squared = 4.8114, df = 1, p-value = 0.02827
```

```
#Look at Expected Values
SkierTest<-chisq.test(Skiers, correct = FALSE)
SkierTest$expected
```

```
##          NoCold  YesCold
## Placebo 115.914 24.08602
## VitC    115.086 23.91398
```

```
mosaicplot(Skiers, color = TRUE, xlab = "Treatment", ylab = "Cold Status")
```



3.2 Rat Tumor Example

```
Tumors <- matrix(c(90, 10, 81, 19, 86, 14), nrow = 3, byrow = TRUE)
colnames(Tumors) <- c("NoTumor", "SomeTumors")
rownames(Tumors) <- c("Ctrl", "High", "Low")
Tumors
```

```
##      NoTumor SomeTumors
## Ctrl      90         10
## High      81         19
## Low       86         14
```

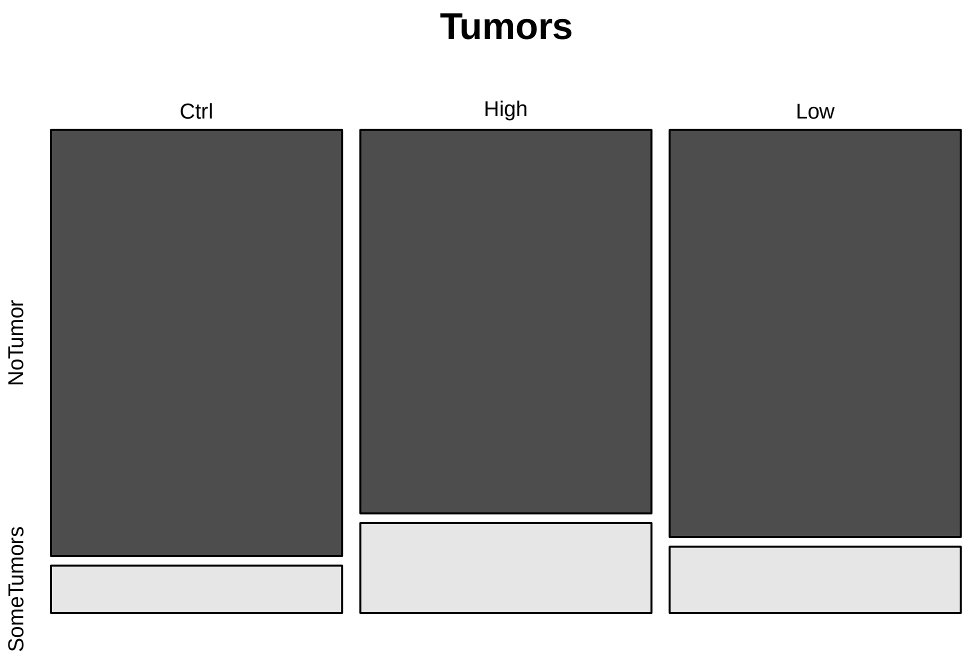
```
prop.table(Tumors, 1)
```

```
##      NoTumor SomeTumors
## Ctrl    0.90       0.10
## High    0.81       0.19
## Low     0.86       0.14
```

```
chisq.test(Tumors, correct = FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  Tumors
## X-squared = 3.3119, df = 2, p-value = 0.1909
```

```
mosaicplot(Tumors, color = TRUE)
```



3.3 Gun Registration Example

```
Guns <- matrix(c(66, 311, 236, 784), nrow = 2, byrow = TRUE)
colnames(Guns)<-c("NoDP", "YesDP")
rownames(Guns)<-c("NoGR", "YesGR")
Guns
```

```
##       NoDP YesDP
## NoGR    66   311
## YesGR  236   784
```

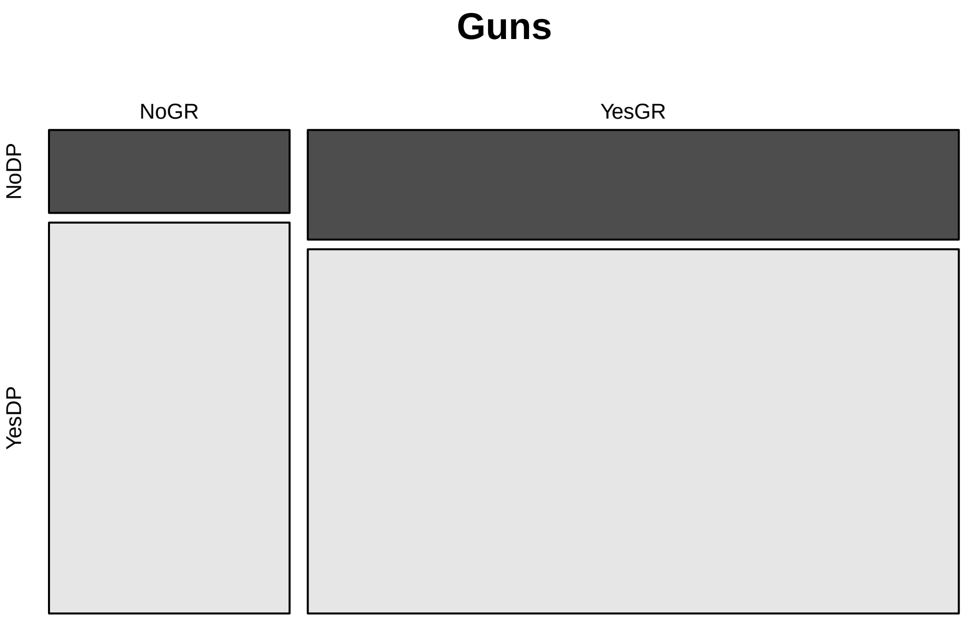
```
prop.table(Guns, 1)
```

```
##            NoDP     YesDP
## NoGR  0.1750663 0.8249337
## YesGR 0.2313725 0.7686275
```

```
chisq.test(Guns, correct = FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  Guns
## X-squared = 5.1503, df = 1, p-value = 0.02324
```

```
mosaicplot(Guns, color = TRUE)
```



```
rm(Guns)
```

3.4 Birds Example

Approach 1: Since the data is in a data.frame, we can construct a summary table and then pass the table to the fisher.test() function. This is the approach we used in previous examples.

```
Birds <- read.csv("C:/hess/STAT511_FA11/RData/CH10_Birds.csv")
str(Birds)
```

```
## 'data.frame':    22 obs. of  3 variables:
##  $ ID  : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Type: Factor w/ 2 levels "Blue","Gold": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Disc: Factor w/ 2 levels "No","Yes": 2 2 2 2 1 1 1 1 1 1 ...
```

```
BirdTable <- with(table(Type, Disc), data = Birds)
BirdTable
```

```
##       Disc
## Type   No Yes
##   Blue  6   4
##   Gold  9   3
```

```
fisher.test(BirdTable)
```

```
##
```

```
##  Fisher's Exact Test for Count Data
##
## data:  BirdTable
## p-value = 0.6517
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.05404558 4.33256378
## sample estimates:
## odds ratio
##  0.5163825
```

Approach 2: We can run FET and chi-square test without first creating the summary table. Note however that the with() function is handy here. Also since the sample sizes are small, the chi-square test is just for illustration here!

```
with(fisher.test(x = Type, y = Disc), data = Birds)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  Type and Disc
## p-value = 0.6517
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.05404558 4.33256378
## sample estimates:
## odds ratio
##  0.5163825
```

```
with(chisq.test(x = Type, y = Disc, correct = FALSE), data = Birds)
```

```
## Warning in chisq.test(x = Type, y = Disc, correct = FALSE): Chi-squared
## approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  Type and Disc
## X-squared = 0.56571, df = 1, p-value = 0.452
```

```
rm(Birds, BirdTable)
```

4 Odds Ratios

Odds ratio and corresponding CI are an alternative to presenting and testing proportions. This is especially useful for case-control studies. We return to the some of the same examples that we used to illustrate the chi-square test for contingency tables. In addition, we look at the Birth control data representing a case-control study.

Notes:
1. When using the oddsratio() function in R, it helps to have (1) reference/control group in first row and (2) "event" in last column. For more details see the function help.
2. Notice that the oddsratio() function provides the results of the chi-square test and Fisher's Exact Test.

4.1 French Skiers Example

```
library(epitools)
oddsratio(Skiers, method = "wald")
```

```
## $data
##         NoCold YesCold Total
## Placebo    109      31   140
## VitC       122      17   139
## Total      231      48   279
##
## $measure
##                          NA
## odds ratio with 95% C.I.  estimate     lower     upper
##                  Placebo 1.0000000        NA        NA
##                  VitC    0.4899524 0.2569419 0.9342709
##
## $p.value
##          NA
## two-sided midp.exact fisher.exact chi.square
##   Placebo         NA           NA         NA
##   VitC    0.02951602   0.03849249 0.02827186
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "Unconditional MLE & normal approximation (Wald) CI"
```

4.2 Rat Tumor Example

```
oddsratio(Tumors, method = "wald")
```

```
## $data
##       NoTumor SomeTumors Total
## Ctrl       90         10   100
## High       81         19   100
## Low        86         14   100
## Total     257         43   300
##
```

```
## $measure
##                          NA
## odds ratio with 95% C.I. estimate     lower    upper
##                       Ctrl 1.000000        NA       NA
##                       High 2.111111 0.9275180 4.805071
##                       Low  1.465116 0.6177245 3.474957
##
## $p.value
##           NA
## two-sided midp.exact fisher.exact chi.square
##      Ctrl         NA           NA         NA
##      High 0.07510514    0.1069786 0.07069593
##      Low  0.39553616    0.5146243 0.38408825
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "Unconditional MLE & normal approximation (Wald) CI"
```

4.3 Birth Control Example

```
BC <- matrix(c(132,35,34,23), nrow = 2, byrow = TRUE)
colnames(BC) <- c("NoMI", "YesMI")
rownames(BC) <- c("NoBC", "YesBC")
oddsratio(BC, method = "wald")
```

```
## $data
##       NoMI YesMI Total
## NoBC   132    35   167
## YesBC   34    23    57
## Total  166    58   224
##
## $measure
##                          NA
## odds ratio with 95% C.I. estimate    lower    upper
##                      NoBC  1.000000       NA       NA
##                      YesBC 2.551261 1.335615 4.873357
##
## $p.value
##           NA
## two-sided  midp.exact fisher.exact  chi.square
##     NoBC           NA           NA          NA
##     YesBC 0.005478672  0.005190049 0.003902078
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "Unconditional MLE & normal approximation (Wald) CI"
```

```
rm(Skiers, Tumors, BC)
```

5 Berkeley Gender Discrimination Example

We use a classic data set where we consider combining 2x2 contingency tables (sex by admission for several departments). The Berkeley admissions study is an example of Simpson's Paradox. We use Breslow-Day test to test for equality of odds ratios comparing across departments.

5.1 Analysis Ignoring Department (or with Departments combined)

```
library(epitools)
library(lawstat)
library(metafor)
data(UCBAdmissions)
UCBAdmissions
```

```
## , , Dept = A
##
##           Gender
## Admit      Male Female
##   Admitted  512     89
##   Rejected  313     19
##
## , , Dept = B
##
##           Gender
## Admit      Male Female
##   Admitted  353     17
##   Rejected  207      8
##
## , , Dept = C
##
##           Gender
## Admit      Male Female
##   Admitted  120    202
##   Rejected  205    391
##
## , , Dept = D
##
##           Gender
## Admit      Male Female
##   Admitted  138    131
##   Rejected  279    244
##
## , , Dept = E
##
##           Gender
## Admit      Male Female
##   Admitted   53     94
##   Rejected  138    299
##
## , , Dept = F
##
##           Gender
## Admit      Male Female
##   Admitted   22     24
##   Rejected  351    317
```

```
class(UCBAdmissions)
```

```
## [1] "table"
```

```
CombineDepts <- margin.table(UCBAdmissions, c(1, 2))
CombineDepts
```

```
##           Gender
## Admit      Male Female
##   Admitted 1198    557
##   Rejected 1493   1278
```

```
prop.table(CombineDepts, 2)
```

```
##           Gender
## Admit           Male    Female
##   Admitted 0.4451877 0.3035422
##   Rejected 0.5548123 0.6964578
```

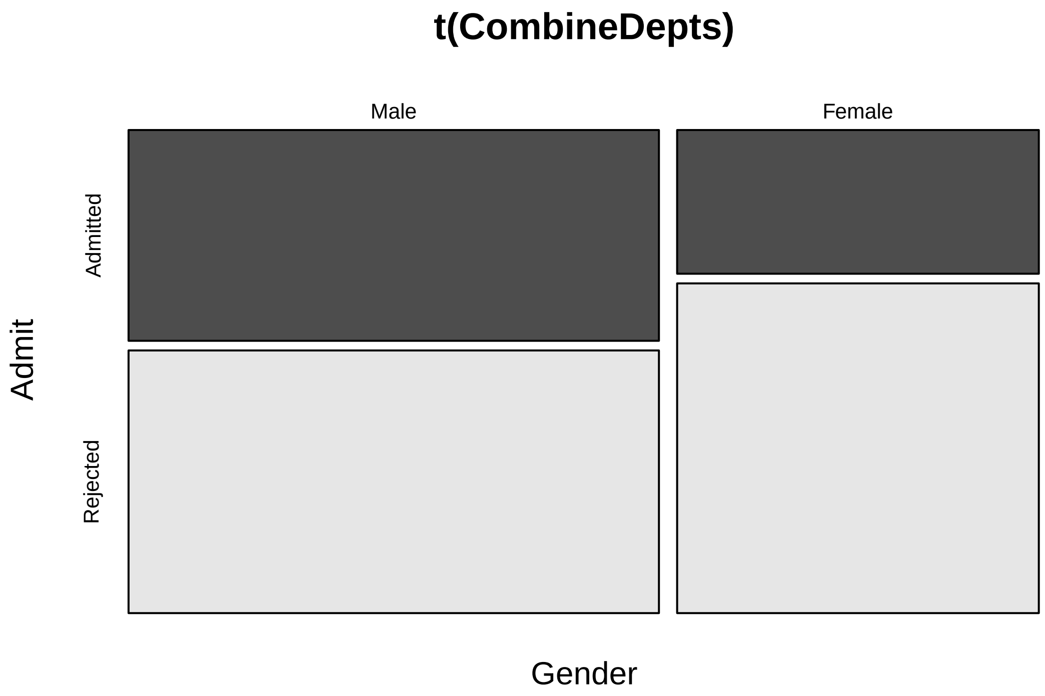
```
chisq.test(CombineDepts)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  CombineDepts
## X-squared = 91.61, df = 1, p-value < 2.2e-16
```

```
oddsratio(CombineDepts, method = "wald")
```

```
## $data
##           Gender
## Admit      Male Female Total
##   Admitted 1198    557  1755
##   Rejected 1493   1278  2771
##   Total    2691   1835  4526
##
## $measure
##           odds ratio with 95% C.I.
## Admit      estimate    lower    upper
##   Admitted  1.00000       NA       NA
##   Rejected  1.84108 1.624377 2.086693
##
## $p.value
##           two-sided
## Admit      midp.exact fisher.exact chi.square
##   Admitted         NA           NA         NA
##   Rejected          0 4.835903e-22 7.8136e-22
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "Unconditional MLE & normal approximation (Wald) CI"
```

```
mosaicplot(t(CombineDepts), color = TRUE)
```



5.2 Analysis BY Department

Here we use the cmh.test() function from the lawstat package to calculate odds ratios by department. But notice we can also use the oddsratio() function to calculate the odds ratio for a single department.

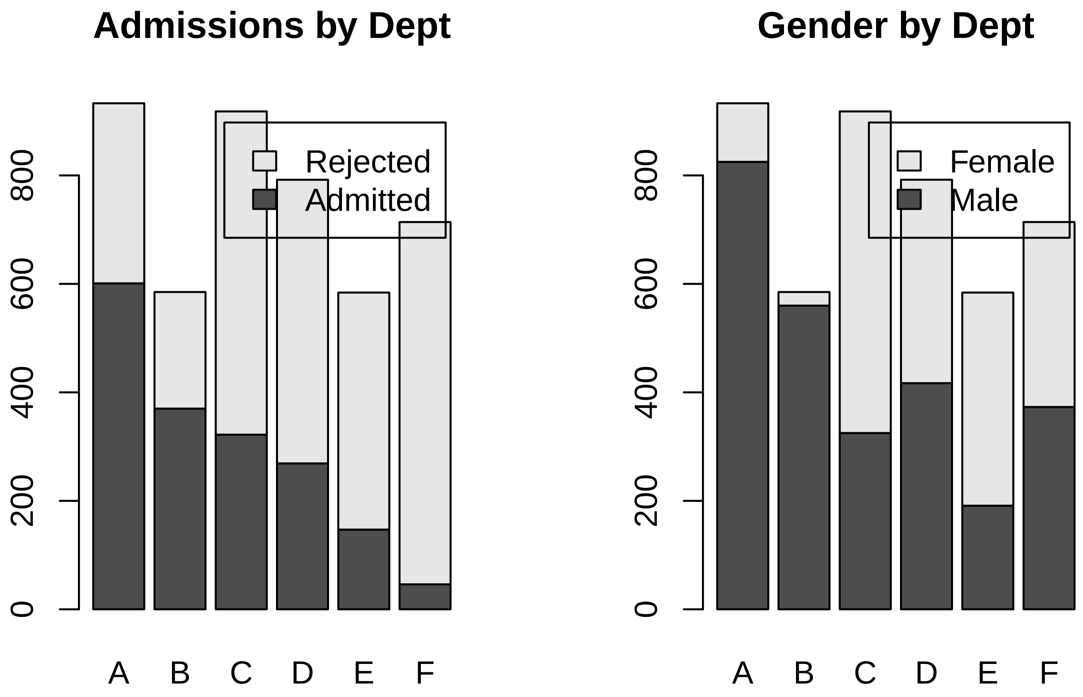
```
cmh.test(UCBAdmissions)
```

```
##
##  Cochran-Mantel-Haenszel Chi-square Test
##
## data:  UCBAdmissions
## CMH statistic = 1.52460, df = 1.00000, p-value = 0.21692, MH
## Estimate = 0.90470, Pooled Odd Ratio = 1.84110, Odd Ratio of level
## 1 = 0.34921, Odd Ratio of level 2 = 0.80250, Odd Ratio of level 3
## = 1.13310, Odd Ratio of level 4 = 0.92128, Odd Ratio of level 5 =
## 1.22160, Odd Ratio of level 6 = 0.82787
```

```
oddsratio(UCBAdmissions[,,1], method = "wald")
```

```
## $data
##           Gender
## Admit      Male Female Total
##   Admitted  512     89   601
##   Rejected  313     19   332
##   Total      825    108   933
##
## $measure
##           odds ratio with 95% C.I.
## Admit      estimate     lower     upper
##   Admitted 1.000000        NA        NA
##   Rejected 0.349212 0.2086756 0.5843954
##
## $p.value
##           two-sided
## Admit        midp.exact fisher.exact   chi.square
##   Admitted           NA           NA           NA
##   Rejected 1.534042e-05 1.669189e-05 3.280404e-05
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "Unconditional MLE & normal approximation (Wald) CI"
```

```r
par(mfrow=c(1,2))
Admit.by.Dept <- margin.table(UCBAdmissions, c(1, 3))
barplot(Admit.by.Dept, legend = T, main = "Admissions by Dept")
Gender.by.Dept <- margin.table(UCBAdmissions, c(2,3))
barplot(Gender.by.Dept, legend = T, main = "Gender by Dept")
```



5.3 Breslow-Day Test for Equality of Odds Ratios

First we run the CMH test using rma.mh() from the metafor package. Then "extract" the results for the Breslow Day Test. Since p-value $<$ alpha $= 0.05$, we reject H0 and conclude that the odds ratio of admission (by gender) varies by department.

```r
cmh <- rma.mh(ai = UCBAdmissions[1,1,],
              bi = UCBAdmissions[1,2,], ci = UCBAdmissions[2,1,],
              di = UCBAdmissions[2,2,])
cmh$BD
```

```
## [1] 18.82551
```

```r
cmh$BDp
```

```
## [1] 0.00207139
```

6 Drug Clinic Three Way Example

Another example of combining 2x2 tables. This time using data from a designed experiment looking at response (improvement or no improvement) versus drug (active or placebo) at 3 study locations. We use Breslow-Day test to test for equality of odds ratios comparing across study locations. We use the Cochran-Mantel-Haenszel test to combine information across locations.

6.1 Create the Data Array

```
library(metafor)
library(lawstat)
library(epitools)
Drugs <- array(c(40,15,10,35,
                     35,20,15,30,
                 43,31,7,19),
      dim = c(2, 2, 3),
      dimnames = list(  Trt = c("Drug", "Plac"),
                Response = c("Imp", "NoImp"),
                Clinic = c("1", "2", "3")))
Drugs
```

```
## , , Clinic = 1
##
##       Response
## Trt    Imp NoImp
##   Drug  40    10
##   Plac  15    35
##
## , , Clinic = 2
##
##       Response
## Trt    Imp NoImp
##   Drug  35    15
##   Plac  20    30
##
## , , Clinic = 3
##
##       Response
## Trt    Imp NoImp
##   Drug  43     7
##   Plac  31    19
```

6.2 Breslow-Day Test for Equality of Odds Ratios

```
cmh <- rma.mh(ai = Drugs[1,1,], bi = Drugs[1,2,],
              ci = Drugs[2,1,], di = Drugs[2,2,])
cmh
```

```
##
## Fixed-Effects Model (k = 3)
##
## Test for Heterogeneity:
```

```
## Q(df = 2) = 2.7958, p-val = 0.2471
##
## Model Results (log scale):
##
## estimate      se    zval    pval   ci.lb   ci.ub
##   1.5888  0.2636  6.0267  <.0001  1.0721  2.1055
##
## Model Results (OR scale):
##
## estimate   ci.lb   ci.ub
##   4.8981  2.9216  8.2116
##
## Cochran-Mantel-Haenszel Test:    CMH = 37.4598, df = 1, p-val < 0.0001
## Tarone's Test for Heterogeneity: X^2 =  2.8085, df = 2, p-val = 0.2456
```

```
cmh$BD
```

```
## [1] 2.816384
```

```
cmh$BDp
```

```
## [1] 0.2445851
```

6.3 CMH Test

There are several ways to run the CMH test. Above we used rma.mh() from the metafor package. Now we will use cmh.test() from lawstat package and mantelhaen.test() from base R. The slight discrepencies between the methods is due to the fact that some of the approaches use a continuity correction.

```
cmh.test(Drugs)
```

```
##
##  Cochran-Mantel-Haenszel Chi-square Test
##
## data:  Drugs
## CMH statistic = 3.8943e+01, df = 1.0000e+00, p-value = 4.3630e-10,
## MH Estimate = 4.8981e+00, Pooled Odd Ratio = 4.6932e+00, Odd Ratio
## of level 1 = 9.3333e+00, Odd Ratio of level 2 = 3.5000e+00, Odd
## Ratio of level 3 = 3.7650e+00
```

```
mantelhaen.test(Drugs, correct = FALSE)
```

```
##
##  Mantel-Haenszel chi-squared test without continuity correction
##
## data:  Drugs
## Mantel-Haenszel X-squared = 38.943, df = 1, p-value = 4.363e-10
## alternative hypothesis: true common odds ratio is not equal to 1
## 95 percent confidence interval:
##  2.921595 8.211577
## sample estimates:
## common odds ratio
##          4.898051
```

6.4 Analysis with Clinics Combined

```
CombineClinics <- margin.table(Drugs, c(1, 2))
CombineClinics
```

```
##       Response
## Trt    Imp NoImp
##   Drug 118    32
##   Plac  66    84
```

```
oddsratio(CombineClinics, method = "wald")
```

```
## $data
##        Response
## Trt     Imp NoImp Total
##   Drug  118    32   150
##   Plac   66    84   150
##   Total 184   116   300
##
## $measure
##       odds ratio with 95% C.I.
## Trt    estimate    lower    upper
##   Drug 1.000000       NA       NA
##   Plac 4.693182 2.828134 7.788159
##
## $p.value
##       two-sided
## Trt      midp.exact fisher.exact   chi.square
##   Drug           NA           NA           NA
##   Plac 5.501279e-10 9.127684e-10 7.052752e-10
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "Unconditional MLE & normal approximation (Wald) CI"
```

7 Mulekick Example: Poisson GOF Test

Poisson goodness of fit (GOF) test is a special case of the chi-square GOF test, used to test whether data is from a Poisson distribution. Note that we lose 1 df by estimating the mean for the Poisson distribution hence we cannot use the chisq.test function directly.

Since the resulting p-value > alpha = 0.05, we fail to reject H0. No evidence against the Poisson distribution.

```
#Observed Data
Obs <- c(109, 65, 22, 3, 1)
Y <- seq(from = 0, to = 4, by = 1)
cbind(Y, Obs)
```

```
##      Y Obs
## [1,] 0 109
## [2,] 1  65
## [3,] 2  22
## [4,] 3   3
## [5,] 4   1
```

```
#Calculate the mean
Muhat <- sum(Obs*Y)/sum(Obs)
Muhat
```

```
## [1] 0.61
```

```
#Calculate the corresponding Poisson Probabilities
Prob <- dpois(Y, Muhat)
Prob
```

```
## [1] 0.543350869 0.331444030 0.101090429 0.020555054 0.003134646
```

```
length(Prob)
```

```
## [1] 5
```

```
sum(Prob)
```

```
## [1] 0.999575
```

```
#"Fix" the final entry so that the probabilities sum to 1
Prob[5] <- 1-sum(Prob[1:4])
Prob
```

```
## [1] 0.543350869 0.331444030 0.101090429 0.020555054 0.003559618
```

```
sum(Prob)
```

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

```
#Calculate Expected values and Contributions to Chisquare TS
Exp <- Prob*200
X2 <- (Obs-Exp)^2/Exp
cbind(Y, Obs, Prob, Exp, X2)
```

```
##      Y Obs        Prob         Exp         X2
## [1,] 0 109 0.543350869 108.6701738 0.00100106
## [2,] 1  65 0.331444030  66.2888060 0.02505734
## [3,] 2  22 0.101090429  20.2180858 0.15704840
## [4,] 3   3 0.020555054   4.1110108 0.30025340
## [5,] 4   1 0.003559618   0.7119235 0.11656877
```

```r
#Run GOF Test
ChiSqTS <- sum(X2)
ChiSqTS
```

```
## [1] 0.599929
```

```r
pval <- 1-pchisq(ChiSqTS, 5-2)
pval
```

```
## [1] 0.8964486
```