IS5 in R: Relationships Between Categorical Variables–Contingency Tables (Chapter 3)

Margaret Chien and Nicholas Horton (nhorton@amherst.edu)

June 16, 2018

Introduction and background

This document is intended to help describe how to undertake analyses introduced as examples in the Fifth Edition of *Intro Stats* (2018) by De Veaux, Velleman, and Bock. More information about the book can be found at http://wps.aw.com/aw_deveaux_stats_series. This file as well as the associated R Markdown reproducible analysis source file used to create it can be found at http://nhorton.people.amherst.edu/is5.

This work leverages initiatives undertaken by Project MOSAIC (http://www.mosaic-web.org), an NSF-funded effort to improve the teaching of statistics, calculus, science and computing in the undergraduate curriculum. In particular, we utilize the `mosaic` package, which was written to simplify the use of R for introductory statistics courses. A short summary of the R needed to teach introductory statistics can be found in the mosaic package vignettes (http://cran.r-project.org/web/packages/mosaic). A paper describing the mosaic approach was published in the *R Journal*: https://journal.r-project.org/archive/2017/RJ-2017-024.

Chapter 3: Relationships Between Categorical Variables–Contingency Tables

Section 3.1: Contingency Tables

```r
library(mosaic)
library(readr)
library(janitor)
OKCupid <- read_csv("http://nhorton.people.amherst.edu/is5/data/OKCupid_CatsDogs.csv", skip = 1) %>%
  clean_names()
```

```
## Parsed with column specification:
## cols(
##   CatsDogsBoth = col_character(),
##   Gender = col_character(),
##   `drugsY/N` = col_character(),
##   `smokesY/N` = col_character()
## )
```

```r
# If you look at the data set, you'll see that the first line is "Col1", "Col2", etc
tally(~ cats_dogs_both + gender, margin = TRUE, data = OKCupid)
```

```
##                gender
## cats_dogs_both     F     M Total
##       Has Both   897   577  1474
##       Has cats  3412  2388  5800
##       Has dogs  3431  3587  7018
##       <NA>     16377 29274 45651
##       Total    24117 35826 59943
```

```
tally(~ cats_dogs_both + gender, format = "percent", margin = TRUE, data = OKCupid)
```

```
##                gender
## cats_dogs_both           F           M       Total
##       Has Both   1.4964216   0.9625811   2.4590027
##       Has cats   5.6920741   3.9837846   9.6758587
##       Has dogs   5.7237709   5.9840182  11.7077891
##       <NA>      27.3209549  48.8363946  76.1573495
##       Total     40.2332216  59.7667784 100.0000000
```

```
tally(~ cats_dogs_both | gender, format = "percent", margin = TRUE, data = OKCupid)
```

```
##                gender
## cats_dogs_both          F          M
##       Has Both   3.719368   1.610562
##       Has cats  14.147697   6.665550
##       Has dogs  14.226479  10.012282
##       <NA>      67.906456  81.711606
##       Total    100.000000 100.000000
```

```
tally(~ gender | cats_dogs_both, format = "percent", margin = TRUE, data = OKCupid)
```

```
##        cats_dogs_both
## gender   Has Both  Has cats  Has dogs      <NA>
##   F      60.85482  58.82759  48.88857  35.87435
##   M      39.14518  41.17241  51.11143  64.12565
##   Total 100.00000 100.00000 100.00000 100.00000
```

Example 3.1: Exploring Marginal Distributions

```
SuperBowl <- read_csv("http://nhorton.people.amherst.edu/is5/data/Watch_the_Super_bowl.csv", skip = 1)
```

```
## Parsed with column specification:
## cols(
##   Plan = col_character(),
##   Sex = col_character()
## )
```

```
tally(~ Plan + Sex, data = SuperBowl)
```

```
##               Sex
## Plan          Female Male
##   Commercials    156   81
##   Game           200  279
##   Wont Watch     160  132
```

Example 3.2: Exploring Percentages: Children and First-Class Ticket Holders First?

```
Titanic <- read_csv("http://nhorton.people.amherst.edu/is5/data/Titanic.csv")
```

```
## Parsed with column specification:
## cols(
##   Name = col_character(),
##   Survived = col_character(),
##   Boarded = col_character(),
##   Class = col_character(),
##   MWC = col_character(),
##   Age = col_double(),
##   Adut_or_Chld = col_character(),
##   Sex = col_character(),
##   Paid = col_double(),
##   Ticket_No = col_character(),
##   Boat_or_Body = col_character(),
##   Job = col_character(),
##   Class_Dept = col_character(),
##   Class_Full = col_character()
## )
```

```
tally(~ Class + Survived, format = "percent", margin = TRUE, data = Titanic)
```

```
##        Survived
## Class         Alive       Dead      Total
##   1        9.103261   5.570652  14.673913
##   2        5.389493   7.518116  12.907609
##   3        8.152174  24.003623  32.155797
##   Crew     9.601449  30.661232  40.262681
##   Total   32.246377  67.753623 100.000000
```

```
tally(~ Survived | Class, format = "percent", margin = TRUE, data = Titanic)
```

```
##         Class
## Survived         1         2         3      Crew
##    Alive  62.03704  41.75439  25.35211  23.84702
##    Dead   37.96296  58.24561  74.64789  76.15298
##    Total 100.00000 100.00000 100.00000 100.00000
```

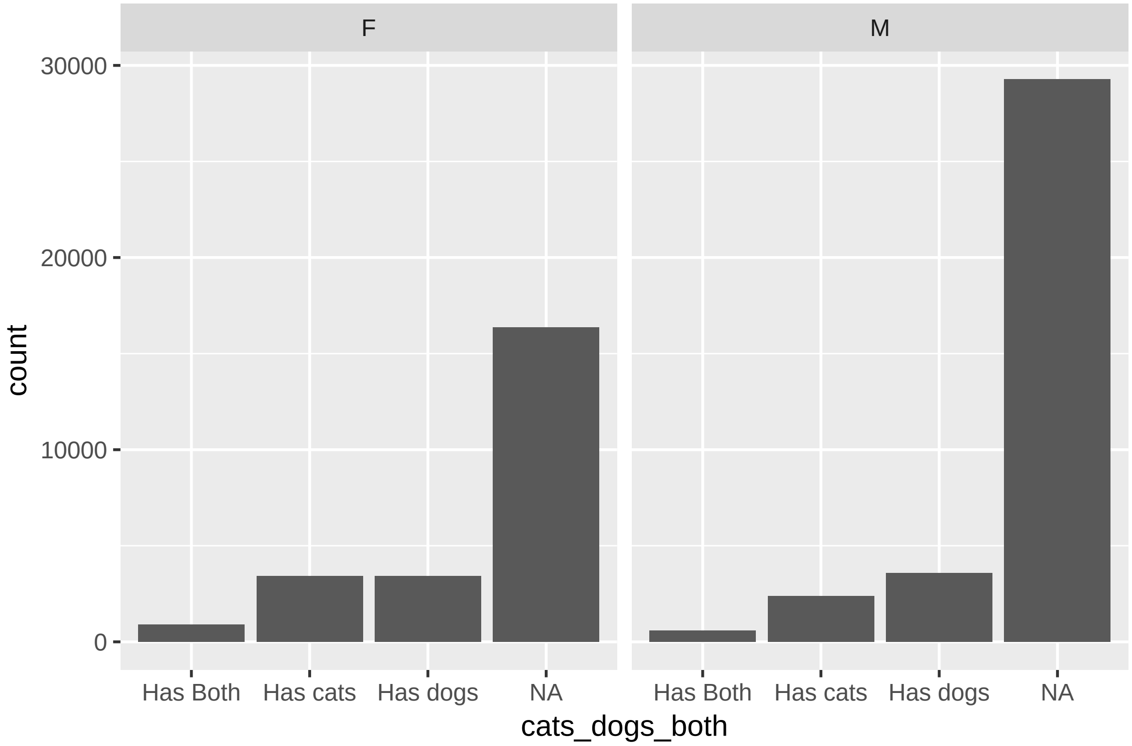
```
tally(~ Class | Survived, format = "percent", margin = TRUE, data = Titanic)
```

```
##        Survived
## Class         Alive       Dead
##   1       28.230337   8.221925
##   2       16.713483  11.096257
##   3       25.280899  35.427807
##   Crew    29.775281  45.254011
##   Total 100.000000 100.000000
```

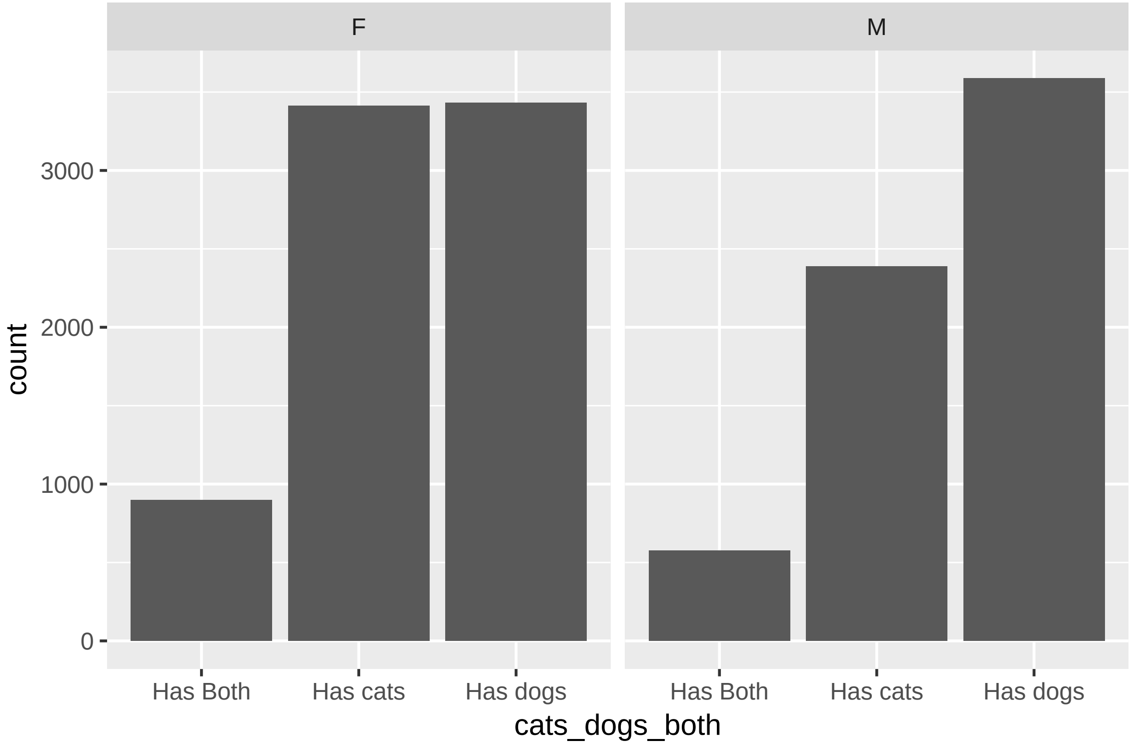
Section 3.2: Conditional Distributions

See displays on 68-69.

```r
gf_bar(~ cats_dogs_both | gender, data = OKCupid)
```



```r
#There are many who don't own either
gf_bar(~ cats_dogs_both | gender, data = filter(OKCupid, cats_dogs_both != "NA"))
```



Example 3.3: Finding Conditional Distributions: Watching the Super Bowl

```
tally(~ Plan + Sex, margin = TRUE, data = SuperBowl)
```

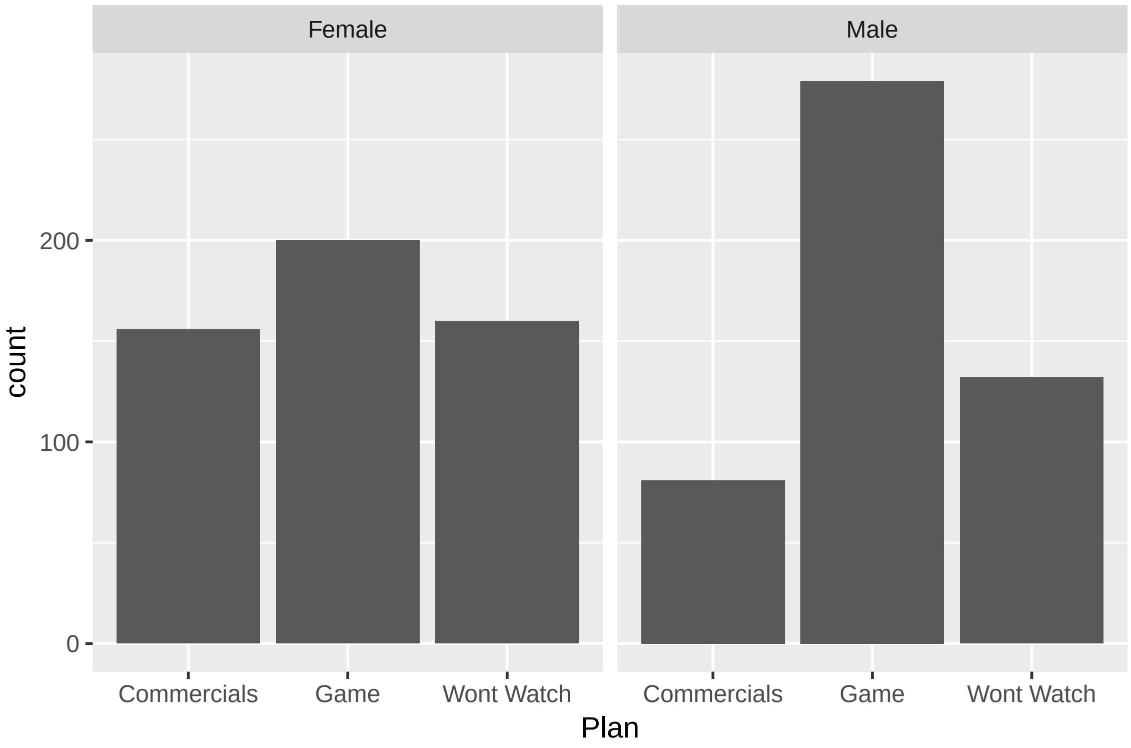
```
##               Sex
## Plan          Female Male Total
##   Commercials    156   81   237
##   Game           200  279   479
##   Wont Watch     160  132   292
##   Total          516  492  1008
```

```
tally(~ Plan | Sex, format = "percent", data = SuperBowl)
```

```
##               Sex
## Plan             Female     Male
##   Commercials 30.23256 16.46341
##   Game        38.75969 56.70732
##   Wont Watch  31.00775 26.82927
```

Example 3.4: Looking for Associations Between Variables: Still Watching the Super Bowl

```
gf_bar(~ Plan | Sex, format = "percent", data = SuperBowl)
```



Examining Contingency Tables

See displays on page 72.

```
FishDiet <- read_csv("http://nhorton.people.amherst.edu/is5/data/Fish_diet.csv", skip = 1) %>%
  clean_names()
```

```
## Parsed with column specification:
## cols(
##   `Diet:Counts` = col_character(),
##   `Cancer:Counts` = col_character()
## )
```

```
tally(~ diet_counts + cancer_counts, data = FishDiet)
```

```
##             cancer_counts
## diet_counts   No  Yes
##    Large     507   42
##    Moderate 2769  209
##    Never     110   14
##    Small    2420  201
```

Random Matters

Seee display on page 74.

```
Nightmares <- read_csv("http://nhorton.people.amherst.edu/is5/data/Nightmares.csv", skip = 1)
```

```
## Parsed with column specification:
## cols(
##   Side = col_character(),
##   Dream = col_character()
## )
```

```
Nightmares <- Nightmares %>%
  mutate(Dream = ifelse(Dream == "N", "Nightmare", "SweetDreams"))
tally(~ Side + Dream, data = Nightmares)
```

```
##     Dream
## Side Nightmare SweetDreams
##    L         9          13
##    R         6          35
```

Section 3.3: Displaying Contingency Tables

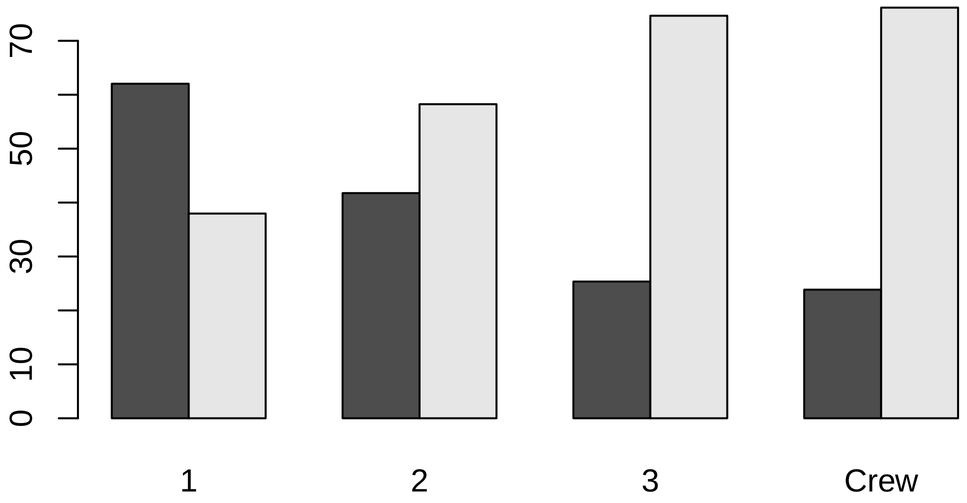
```
tally(~ Class + Survived, format = "count", data = Titanic)
```

```
##       Survived
## Class  Alive Dead
##   1       201  123
##   2       119  166
##   3       180  530
##   Crew    212  677
```

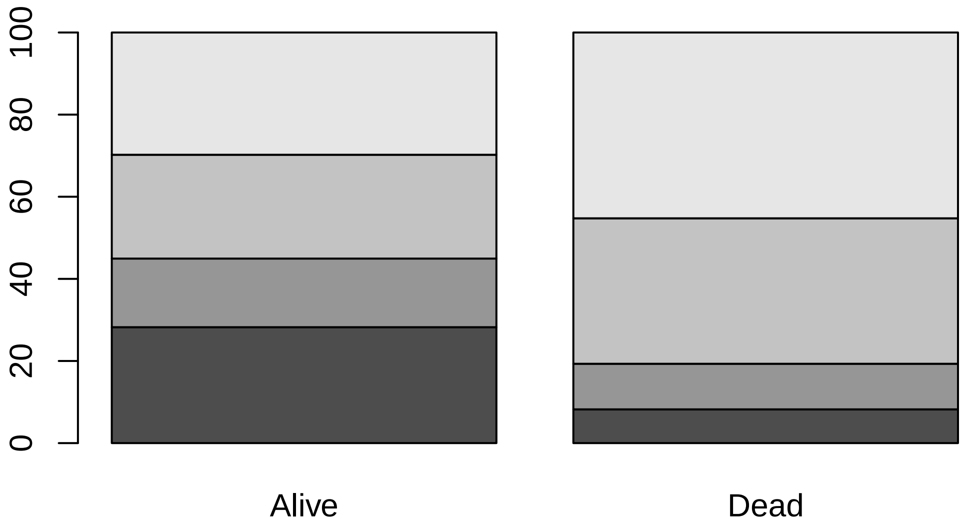
```
tally(~ Class + Survived, format = "percent", data = Titanic)
```

```
##       Survived
## Class       Alive      Dead
##   1      9.103261  5.570652
##   2      5.389493  7.518116
##   3      8.152174 24.003623
##   Crew   9.601449 30.661232
```

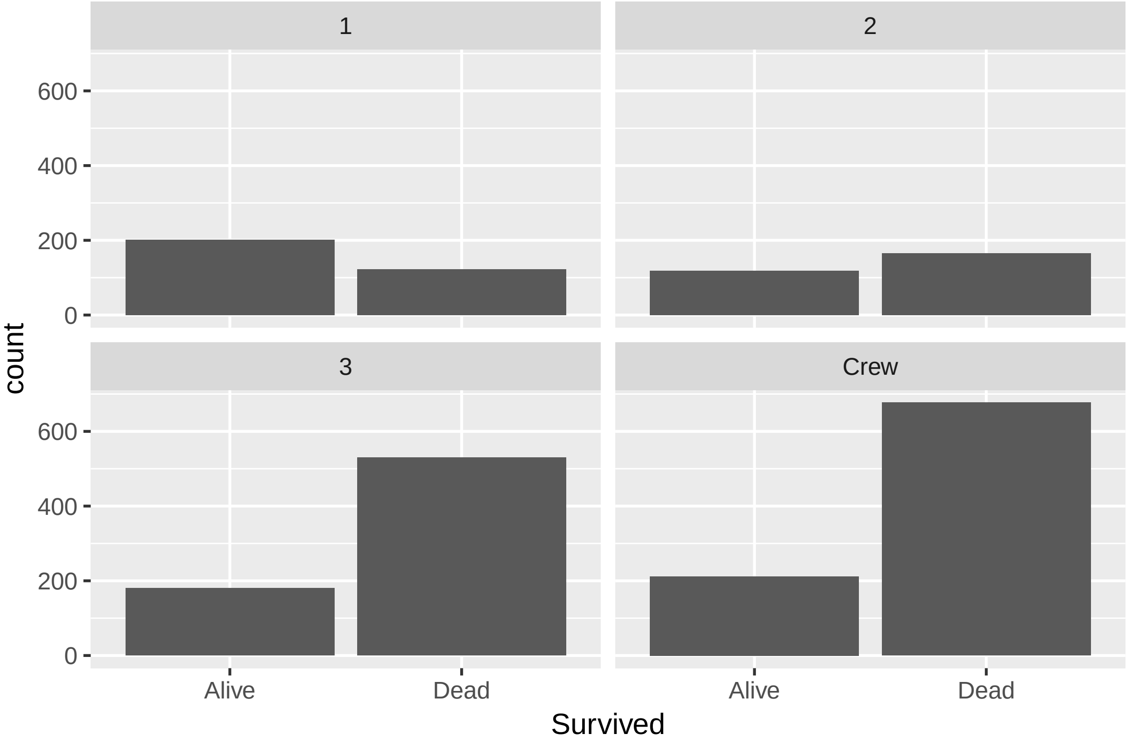
```
barplot(tally(~ Survived | Class, format = "percent", data = Titanic), beside = TRUE)
```



```
barplot(tally(~ Class | Survived, format = "percent", data = Titanic))
```

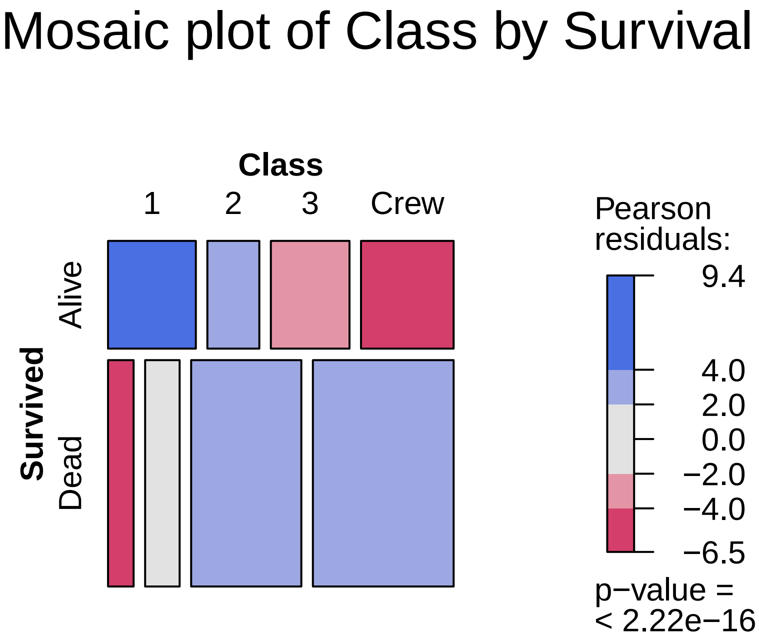


```
gf_bar(~ Survived | Class, format = "percent", data = Titanic)
```



```
#gf_bar automatically facets
```

```
vcd::mosaic(tally(~ Survived + Class, data = Titanic),
            main = "Mosaic plot of Class by Survival",
            shade = TRUE)
```



See the mosaic plots on page 77.

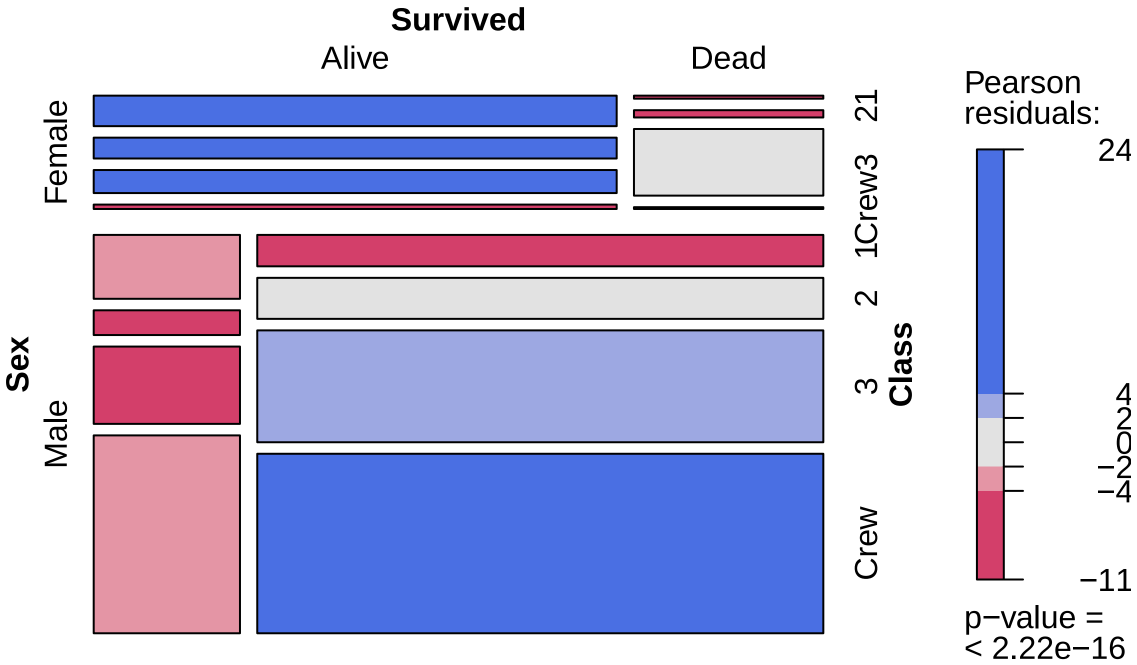
Section 3.4: Three Categorical Variables

```
tally(~ gender + cats_dogs_both + drugs_y_n, format = "percent", data = OKCupid)
```

```
## , , drugs_y_n = No
##
##       cats_dogs_both
## gender   Has Both   Has cats   Has dogs       <NA>
##      F  1.0243064  3.4199156  3.9437466 18.0187845
##      M  0.5922293  2.0819779  3.7769214 30.0719016
##
## , , drugs_y_n = Yes
##
##       cats_dogs_both
## gender   Has Both   Has cats   Has dogs       <NA>
##      F  0.2085314  0.8941828  0.6272626  2.9794972
##      M  0.1901807  0.8658225  0.9041923  6.9132342
##
## , , drugs_y_n = NA
##
##       cats_dogs_both
## gender   Has Both   Has cats   Has dogs       <NA>
##      F  0.2635837  1.3779757  1.1527618  6.3226732
##      M  0.1801712  1.0359842  1.3029044 11.8512587
```

Example 3.7: Looking for Associations Among Three Variables at Once

```
vcd::mosaic(tally(~ Sex + Survived + Class, data = Titanic), shade = TRUE)
```



Example 3.8: Simpson's Paradox: Gender Discrimination?

See tables on page 80.

```
# Two ways to create a data frame
Berk_2 <- rbind(
  do(512) * data.frame(admit = TRUE, sex = "M", school = "A"),
  do(313) * data.frame(admit = TRUE, sex = "M", school = "B"),
  do(120) * data.frame(admit = TRUE, sex = "M", school = "C"),
  do(138) * data.frame(admit = TRUE, sex = "M", school = "D"),
  do(53) * data.frame(admit = TRUE, sex = "M", school = "E"),
  do(22) * data.frame(admit = TRUE, sex = "M", school = "F"),
  do(316) * data.frame(admit = FALSE, sex = "M", school = "A"),
  do(206) * data.frame(admit = FALSE, sex = "M", school = "B"),
  do(205) * data.frame(admit = FALSE, sex = "M", school = "C"),
  do(278) * data.frame(admit = FALSE, sex = "M", school = "D"),
  do(138) * data.frame(admit = FALSE, sex = "M", school = "E"),
  do(350) * data.frame(admit = FALSE, sex = "M", school = "F"),
  do(89) * data.frame(admit = TRUE, sex = "F", school = "A"),
  do(17) * data.frame(admit = TRUE, sex = "F", school = "B"),
  do(202) * data.frame(admit = TRUE, sex = "F", school = "C"),
  do(131) * data.frame(admit = TRUE, sex = "F", school = "D"),
  do(94) * data.frame(admit = TRUE, sex = "F", school = "E"),
  do(24) * data.frame(admit = TRUE, sex = "F", school = "F"),
  do(16) * data.frame(admit = FALSE, sex = "F", school = "A"),
  do(8) * data.frame(admit = FALSE, sex = "F", school = "B"),
  do(391) * data.frame(admit = FALSE, sex = "F", school = "C"),
  do(245) * data.frame(admit = FALSE, sex = "F", school = "D"),
  do(300) * data.frame(admit = FALSE, sex = "F", school = "E"),
  do(318) * data.frame(admit = FALSE, sex = "F", school = "F")
)  # some numbers are off but it's really annoying to fix

tally(~ sex + admit, data = Berk_2)
```

```
##    admit
## sex TRUE FALSE
##   M 1158  1493
##   F  557  1278
```

```
tally(~ sex + school + admit, data = Berk_2)
```

```
## , , admit = TRUE
##
##    school
## sex   A   B   C   D   E   F
##   M 512 313 120 138  53  22
##   F  89  17 202 131  94  24
##
## , , admit = FALSE
##
##    school
## sex   A   B   C   D   E   F
##   M 316 206 205 278 138 350
##   F  16   8 391 245 300 318
```

```r
admit <- c(rep("TRUE", 1158), rep("FALSE", 1493), rep("TRUE", 557), rep("FALSE", 1278))
sex <- c(rep("M", 1158 + 1493), rep("F", 557 + 1278))
school <- c(rep("A", 512), rep("B", 313), rep("C", 120),
            rep("D", 138), rep("E", 53), rep("F", 22),
            rep("A", 512/.621 - 512), rep("B", 313/.602 - 313),
            rep("C", 120/.369 - 120), rep("D", 138/.331 - 138),
            rep("E", 53/.277 - 53), rep("F", 22/.059 - 22),
            rep("A", 89), rep("B", 17), rep("C", 202),
            rep("D", 131), rep("E", 94), rep("F", 24),
            rep("A", 89/.824 - 89 + 1), rep("B", 17/.68 - 17 + 1),
            rep("C", 202/.341 - 202 + 1), rep("D", 131/.349 - 131 + 1),
            rep("E", 94/.239 - 94 + 1), rep("F", 24/.07 - 24)) # + 1 because the rows weren't matching
Berk <- data.frame(sex, admit, school)
tally(~ sex + admit, data = Berk)
```

```
##    admit
## sex FALSE TRUE
##   F  1278  557
##   M  1493 1158
```

```r
tally(~ sex + school + admit, data = Berk)
```

```
## , , admit = FALSE
##
##    school
## sex   A   B   C   D   E   F
##   F  16   8 391 245 300 318
##   M 316 206 205 278 138 350
##
## , , admit = TRUE
##
##    school
## sex   A   B   C   D   E   F
##   F  89  17 202 131  94  24
##   M 512 313 120 138  53  22
```