Chapter 5 Workshop

Table of contents

Dataset `Toxaemia`

This dataset is from the `vcdExtra` package. Two signs of *toxaemia*, an abnormal condition during pregnancy characterized by high blood pressure (hypertension) and high levels of protein in the urine. If untreated, both the mother and baby are at risk of complications or death. The dataset `Toxaemia` represents 13384 expectant mothers in Bradford, England in their first pregnancy, who were also classified according to social class and the number of cigarettes smoked per day.

The dataset is a 5 x 3 x 2 x 2 contingency table, with 60 observations on the following 5 variables:

`class` - Social class of mother, a factor with levels: 1, 2, 3, 4, 5

`smoke` - Cigarettes smoked per day during pregnancy, a factor with levels: 0, 1-19, 20+

`hyper` - Hypertension level, a factor with levels: Low, High

`urea` - Protein urea level, a factor with levels: Low, High

`Freq` - frequency in each cell, a numeric vector

Exercise 5.1

Obtain relevant graphical displays for this dataset.

Bar charts

```
library(tidyverse)
library(vcdExtra)
```

```
data(Toxaemia)
```

```
Toxaemia |>
  ggplot() +
  aes(x=smoke, y=Freq, fill=hyper) +
  geom_bar(stat='identity')
```

```
Toxaemia |>
  ggplot() +
  aes(x=smoke, y=Freq, fill=hyper) +
  geom_bar(stat='identity',
           position = "dodge"
           )
```

```
Toxaemia |>
  ggplot() +
  aes(x=smoke, y=Freq, fill=hyper) +
  geom_bar(stat ='identity',
           position = "dodge") +
  facet_grid(urea ~ ., scales = "free")
```

Mosaic type charts

```
tab.data <- xtabs(Freq ~ smoke + hyper + urea, data=Toxaemia)

plot(tab.data)
```

```
mosaic(tab.data, shade=TRUE, legend=TRUE)
```

```
assoc(tab.data, shade=TRUE)
```

```
strucplot(tab.data)
```

```
sieve(tab.data)
```

The full dataset is a 5 x 3 x 2 x 2 contingency table, with 60 observations on the following 5 variables. For this question we will focus on two categorical variables from this dataset, `hyper` and `urea`. This forms a 2 x 2 contingency table since these variables each have two levels.

```
# subset the data
tox_2 <- Toxaemia |>
  dplyr::select(hyper, urea, Freq)
```

```
# the tidyverse way
tox_display <- tox_2 |>
  pivot_wider(names_from = urea,
              values_from = Freq,
              values_fn = sum) |>
  column_to_rownames( var = "hyper") # make values of hyper column row names

tox_display
```

```
     High  Low
High  665 2715
Low   589 9415
```

```
# xtabs()
```

Two signs of *toxaemia*, are high blood pressure (hypertension) and high levels of protein in the urine. We want to ask if in our sample of expectant mothers in Bradford, England, is high

blood pressure related to high protein levels? If these two variables are associated this may indicate the presence of toxaemia in the sample, if they are independent toxaemia may not be present.

We can test this question using a Chi-squared test.

The null hypothesis of the chi-squared these is that the two variables are independent and the alternative hypothesis is that the two variables are not independent.

Our null hypothesis is that Hypertension level and the Protein urea level in expectant mothers in Bradford, England are independent.

Our alternative hypothesis that Hypertension level and the Protein urea level in expectant mothers in Bradford, England are *not* independent.

Set our alpha = 0.05

```
chisq.test(tox_display)
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  tox_display
X-squared = 563.9, df = 1, p-value < 2.2e-16
```

Since our p-value is less than our alpha level we reject the null hypothesis and conclude that the two variables (hyper & urea) are not independent. We found evidence of an association between hypertension levels and protein in urine levels in our sample of expectant mothers in in Bradford, England.

We can see the expected counts

```
chisq.test(tox_display)$expected
```

```
         High      Low
High 316.6856 3063.314
Low  937.3144 9066.686
```

```
# compared to our observed
tox_display
```

```
     High  Low
High  665 2715
Low   589 9415
```

```
# total counts 13384
```

Exercise 5.2

The genetic information of an organism is stored in its Deoxyribonucleic acid (DNA). DNA is a double stranded helix made up of four different nucleotides. These nucleotides differ in which of the four bases Adenine (A), Guanine (G), Cytosine (C), or Thymine (T) they contain. Nucleotides combine to form amino acids which are the building blocks of proteins. Simply put, three nucleotides form an amino acid and the specific order of a combination dictates what amino acid is formed. A simple pattern that we may want to detect in a DNA sequence is that of the nucleotide at position `i+1` based on the nucleotide at position `i`. The nucleotide positional data collected by a researcher in a particular case is given in the following table:

`i\(i+1)`	A	C	G	T
A	622	316	328	536
C	428	262	204	306
G	354	294	174	266
T	396	330	382	648

Perform a test of association and then obtain the symmetric plot.

```
tabledata <- data.frame(
  A = c(622, 428, 354, 396),
  C = c(316, 262, 294, 330),
  G = c(328, 204, 174, 382),
  T = c(536, 306, 266, 648),
  row.names = c("A", "C", "G", "T")
  )
```

```
chisq.test(tabledata)$exp
```

```
          A        C        G        T
A 554.8409 370.5104 335.3705 541.2781
C 369.4834 246.7328 223.3322 360.4516
G 334.9983 223.7044 202.4879 326.8094
T 540.6774 361.0523 326.8094 527.4608
```

```r
chisq.test(tabledata)
```

```
    Pearson's Chi-squared test

data:  tabledata
X-squared = 153.21, df = 9, p-value < 2.2e-16
```

```r
chisq.test(tabledata, simulate.p.value = T)
```

```
    Pearson's Chi-squared test with simulated p-value (based on 2000
    replicates)

data:  tabledata
X-squared = 153.21, df = NA, p-value = 0.0004998
```

```r
# if there is an association we can examine patterns
library(MASS)
corresp(tabledata)
```

```
First canonical correlation(s): 0.1443355

 Row scores:
         A          C          G          T
-0.1921802 -0.8894387 -1.0334109  1.4453224

 Column scores:
         A          C          G          T
-1.1304512 -0.6952989  0.8139424  1.1304056
```

```r
plot(corresp(tabledata, nf=2))
abline(v=0)
abline(h=0)
```

```r
#or
library(FactoMineR)
CA(tabledata)
```

```
**Results of the Correspondence Analysis (CA)**
The row variable has  4  categories; the column variable has 4 categories
The chi square of independence between the two variables is equal to 153.2146 (p-value =  1.9
*The results are available in the following objects:

   name              description
1  "$eig"            "eigenvalues"
2  "$col"            "results for the columns"
3  "$col$coord"      "coord. for the columns"
4  "$col$cos2"       "cos2 for the columns"
5  "$col$contrib"    "contributions of the columns"
6  "$row"            "results for the rows"
7  "$row$coord"      "coord. for the rows"
8  "$row$cos2"       "cos2 for the rows"
9  "$row$contrib"    "contributions of the rows"
10 "$call"           "summary called parameters"
11 "$call$marge.col" "weights of the columns"
12 "$call$marge.row" "weights of the rows"
```

Exercise 5.3

The `diamonds` dataset contains the prices and other attributes of almost 54,000 diamonds. Use `?diamonds` to see information for each variable.

We are interested in whether there is an association between cut and color. Perform a test of association and then obtain the symmetric plot.

```
data("diamonds")
names(diamonds)
```

```
[1] "carat"   "cut"     "color"   "clarity" "depth"   "table"   "price"
[8] "x"       "y"       "z"
```

```
## Some EDA plots
ggplot(diamonds, aes(color))+geom_bar() + facet_wrap(~cut)
```

```
ggplot(diamonds, aes(color))+geom_bar(aes(fill=cut))
```

```
ggplot(diamonds, aes(color))+geom_bar(aes(fill=cut))+ facet_wrap(~clarity)
```

```
# alternative coding for making a table of data to count observations of each category
cont.table <- table(diamonds$cut, diamonds$color)

# EDA
tab.data <- xtabs( ~ cut+color, data = diamonds)


plot(tab.data)
```

```
# A test of association
#
chisq.test(tab.data)
```

```
    Pearson's Chi-squared test

data:  tab.data
X-squared = 310.32, df = 24, p-value < 2.2e-16
```

```
chisq.test(tab.data)$expected
```

```
           color
cut                 D         E         F         G         H         I
  Fair       202.2201  292.4207  284.8094  337.0434  247.8576  161.8357
  Good       616.2060  891.0657  867.8727 1027.0403  755.2730  493.1467
  Very Good 1517.5297 2194.4263 2137.3089 2529.2908 1860.0098 1214.4717
  Premium   1732.1844 2504.8281 2439.6315 2887.0592 2123.1083 1386.2588
  Ideal     2706.8599 3914.2593 3812.3775 4511.5664 3317.7513 2166.2870
           color
cut                 J
  Fair        83.81313
  Good       255.39577
  Very Good  628.96285
  Premium    717.92970
  Ideal     1121.89855
```

```
chisq.test(tab.data)$stdres
```

```
           color
cut                 D          E          F          G          H          I
  Fair      -2.9944825 -4.4904291  1.8029978 -1.4331431  3.8660384  1.1077538
  Good       2.0691800  1.6287194  1.6139239 -5.7432258 -2.2103621  1.4368735
  Very Good -0.1411592  5.5067817  0.7223909 -5.8458749 -1.0304594 -0.3596617
  Premium   -3.8474818 -4.2965348 -2.8098650  0.8961959  6.4786345  1.3701384
  Ideal      3.3724986 -0.2567250  0.3138264  8.0472595 -4.9385571 -2.1425443
           color
cut                 J
  Fair       4.0078721
  Good       3.4785185
  Very Good  2.2797544
  Premium    4.0019148
  Ideal     -8.9392779
```

```r
chisq.test(tab.data, simulate.p.value = TRUE)
```

```
    Pearson's Chi-squared test with simulated p-value (based on 2000
    replicates)

data:  tab.data
X-squared = 310.32, df = NA, p-value = 0.0004998
```

```r
# plots below are expecting an xtabs object, additional arguments would have to be added f
mosaic(tab.data, shade=TRUE, legend=TRUE)
```

```r
assoc(tab.data, shade=TRUE)
```

```r
strucplot(tab.data, core = struc_assoc, )
```

```r
sieve(tab.data)
```

```r
library(gplots)
gplots::balloonplot(tab.data, main ="Balloon Plot", xlab ="", ylab="",
                    label = FALSE, show.margins = FALSE)
```

- More R code examples are here