Lab 5 - Chisquare and Contingency Analysis

Contents

Lab Objectives

- Conduct Chi-Square goodness-of-fit tests
- Enter frequency data and assign weighting
- Calculate a Poisson distribution
- Conduct a Chi-square contingency test and Fisher's Exact test for contingency tables
- Use R to determine odds ratio and relative risk

Exercise 1: Chi-Square testing in R

A Chi-Square test is used to analyze frequency data for categorical or discrete numerical variables. It compares observed frequencies to expected or predicted frequencies. We will conduct Chi-square goodness-of-fit tests to analyze frequency distributions of a single variable, and Chi-Square contingency tests to analyze associations between two or more categorical variables.

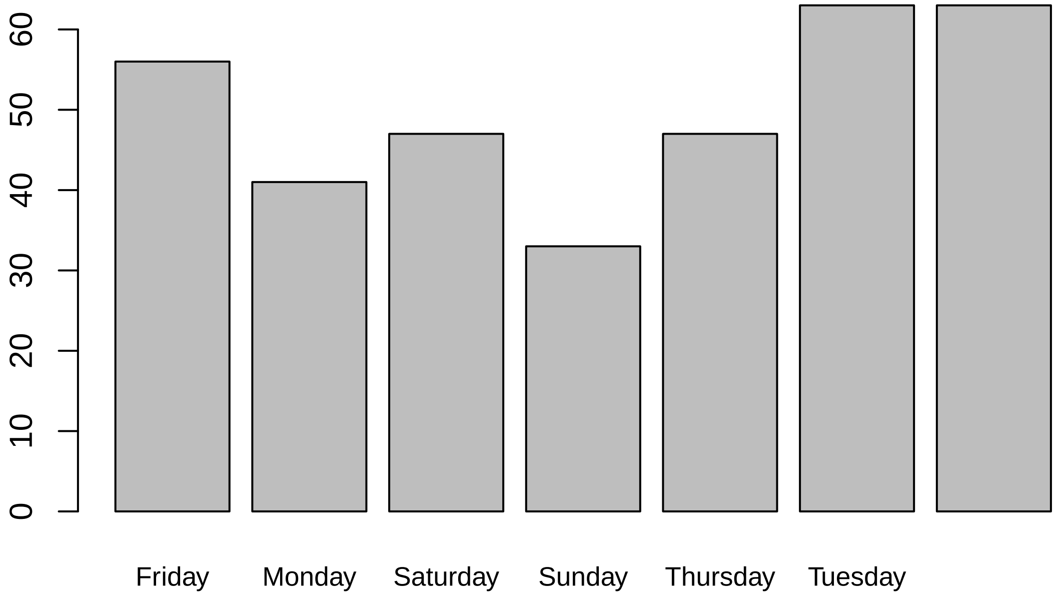
Note that a Statistical Table (χ^2 Distribution) can be found at the end of this file.

Days of the week

In class, we looked at a research study examining the number of births on each day of the week. We will examine this data set, but will consider a scenario where every day of the week has an equal probability (e.g. 1/7). This data is found in the file Days of the week.sav

Open Days of the week.csv (these data are straightforward enough they can be plotted with the plot(d) command:

```
d <- read.csv("days of the week.csv")
plot(d, cex.names = 0.8)
```



Tabulate the variable day, from within the d data.frame and perform a chi-square test:

```
x <- xtabs(~d$day)
chisq.test(x)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  x
## X-squared = 15.24, df = 6, p-value = 0.01847
```

By default, the chisq.test will assume that the null expectation is equally distributed across the categories, weighted by column sum or row sums. In this example, there are only 7 categories, so the null expectation is the sum of all observations divided by 7:

```
freq <- rep(sum(x)/7, 7)
p <- freq/sum(x)
chisq.test(x, p = p)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  x
## X-squared = 15.24, df = 6, p-value = 0.01847
```

which yields the same result. You might specify a different null by providing a unique set of frequencies. The

chisq.test() function requires that the frequencies be supplied as a vector of probabilities that sum to 1.

By default, the function provides only the essential statistical outputs. It provides the χ^2 value (X-squared), the degrees of freedom (df), and the significance value.

You could report this in a scientific paper as: χ^2 (df=6) = 15; p=0.02

To obtain more detailed information, you should create an object that contains the results of the analysis:

```
results <- chisq.test(x, p = p)
str(results)
```

```
## List of 9
##  $ statistic: Named num 15.2
##   ..- attr(*, "names")= chr "X-squared"
##  $ parameter: Named num 6
##   ..- attr(*, "names")= chr "df"
##  $ p.value  : num 0.0185
##  $ method   : chr "Chi-squared test for given probabilities"
##  $ data.name: chr "x"
##  $ observed : 'xtabs' int [1:7(1d)] 56 41 47 33 47 63 63
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ d$day: chr [1:7] "Friday" "Monday" "Saturday" "Sunday" ...
##   ..- attr(*, "call")= language xtabs(formula = ~d$day)
##  $ expected : Named num [1:7] 50 50 50 50 50 50 50
##   ..- attr(*, "names")= chr [1:7] "Friday" "Monday" "Saturday" "Sunday" ...
##  $ residuals: 'xtabs' num [1:7(1d)] 0.849 -1.273 -0.424 -2.404 -0.424 ...
##   ..- attr(*, "call")= language xtabs(formula = ~d$day)
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ d$day: chr [1:7] "Friday" "Monday" "Saturday" "Sunday" ...
##  $ stdres   : 'xtabs' num [1:7(1d)] 0.917 -1.375 -0.458 -2.597 -0.458 ...
##   ..- attr(*, "call")= language xtabs(formula = ~d$day)
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ d$day: chr [1:7] "Friday" "Monday" "Saturday" "Sunday" ...
##  - attr(*, "class")= chr "htest"
```

```
results$expected
```

```
##    Friday    Monday  Saturday    Sunday  Thursday   Tuesday Wednesday
##        50        50        50        50        50        50        50
```

```
results$residuals
```

```
## d$day
##     Friday     Monday   Saturday     Sunday   Thursday    Tuesday
##  0.8485281 -1.2727922 -0.4242641 -2.4041631 -0.4242641  1.8384776
##  Wednesday
##  1.8384776
```

```
results$stdres
```

```
## d$day
##     Friday     Monday   Saturday     Sunday   Thursday    Tuesday
##  0.9165151 -1.3747727 -0.4582576 -2.5967929 -0.4582576  1.9857828
##  Wednesday
##  1.9857828
```

In the output above, you should see a table showing each day of the week, the expected N for each category, the residuals [(O-E)/sqrt(E)], and standardised residuals.

Answer questions 1 and 2 on sakai

Birds in a storm

There are two ways that R can work with frequency data. The first we have worked with extensively already: each row represents an individual or observation, and each column represents a variable. The value of a categorical variable in a given cell describes which group the individual in that row belongs to.

Alternatively, in some cases you may have frequency data that is already organized into a table, or summarized as such. For example, you may be interested in the sex of birds collected at a particular site following a wind storm. You have recorded that 49 females and 87 males were caught. In this case it would be quicker to input this data into R in summary format, rather than filling in 49 rows for females, and 87 rows for males.

The easiest way to do that for two categories to to create a vector using the **c()** function. With 2x2 tables or more complex data, you would need to use the **as.matrix()** function to convert the date into a 2 dimensional variable. Below we show both approaches:

```r
x <- c(49, 87)
chisq.test(x)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  x
## X-squared = 10.618, df = 1, p-value = 0.00112
```

```r
x <- as.matrix(c(Females = 49, Males = 87), nrow = 2)
chisq.test(x)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  x
## X-squared = 10.618, df = 1, p-value = 0.00112
```

Recently we focused on conducting binomial tests. A binomial test compares the observed number of successes to that expected under the null hypothesis, and calculates an exact P-value. A binomial test can only be used when there are only two categories. The Chi-Square goodness-of-fit test also works when there are only two categories and can be a quick substitute for the binomial test when calculating statistics by hand.

When calculated with a statistics program, the Chi-Square test is still an approximation based on the Chi-Square probability distribution with a particular number of degrees of freedom. It will provide a P-value, but this will not be as precise as the P-value calculated with the binomial test. Let's check the difference using the bird data:

```r
binom.test(x, p = 0.5, alternative = "two.sided")
```

```
##
##  Exact binomial test
##
## data:  x
## number of successes = 49, number of trials = 136, p-value =
## 0.001419
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.2798053 0.4470114
## sample estimates:
## probability of success
##              0.3602941
```

```
binom.test(x, p = 0.5, alternative = "two.sided")$p.value
```

```
## [1] 0.001418638
```

```
# compare to the p value from the chisq.test run earlier
chisq.test(x)$p.value
```

```
## [1] 0.001120135
```

Answer question 3 on Sakai

Not at all like me

Now let's try an example where the expected values are not equal across categories. The difference here is we have to specify the expected proportions or counts. Howell has a hypothesis that if you ask participants to sort one-sentence characteristics of themselves (such as "I eat too fast") into five piles ranging from "not at all like me", to "very much like me", the percentage of items placed in each pile will be approximately 10%, 20%, 40%, 20%, and 10% for the five piles. Let's test this hypothesis using data gathered from 50 sorted statements.

```
d <- read.csv("not like me.csv")
str(d)
```

```
## 'data.frame':    5 obs. of  3 variables:
##  $ Category   : Factor w/ 5 levels "Much like me",..: 3 4 2 1 5
##  $ Frequency  : int  7 11 21 7 4
##  $ PredPercent: int  10 20 40 20 10
```

```
levels(d$Category)
```

```
## [1] "Much like me"                  "Neither like me or unlike me"
## [3] "Not at all like me"            "Somewhat like me"
## [5] "Very much like me"
```

```
x <- d$Frequency
p <- d$PredPercent
# there are 5 levels in Category corresponding to the subject's answer.
# we need to rescale the p values to sum up to 1, not 100 when using
# the chisq.test function
```

Perform a chisq.test on the data above.

Answer question 4 and 5 on Sakai

Exercise 2: Poisson Distribution

The Poisson distribution describes the frequency distribution of successes, when successes happen independently and with equal probability over time or space. Given a mean number of events observed in a given time or space, the Poisson distribution can be used to determine the probability of observing X number of events in that same unit of time or space. If successes occur "randomly" in time or space, then the distribution of values should follow the Poisson distribution.

Emergency

Let's check the Poisson distribution for the number of people admitted to the ER between 7 and 8 pm on Friday night, given an average admittance of 2.4 people. We will assume that admissions are independent of one another, and are just as likely to land in one instant in time as another on a Friday night. Your boss at the hospital is adjusting staff schedules and is interested in the probability that 0 – 10 patients will be admitted during this time period.

ppois() returns a cumulative probability that the stated X will be <= to a threshold value. So if you want the probability of a specific whole number value, you will have to calculated two probabilities and subtract one from the other.

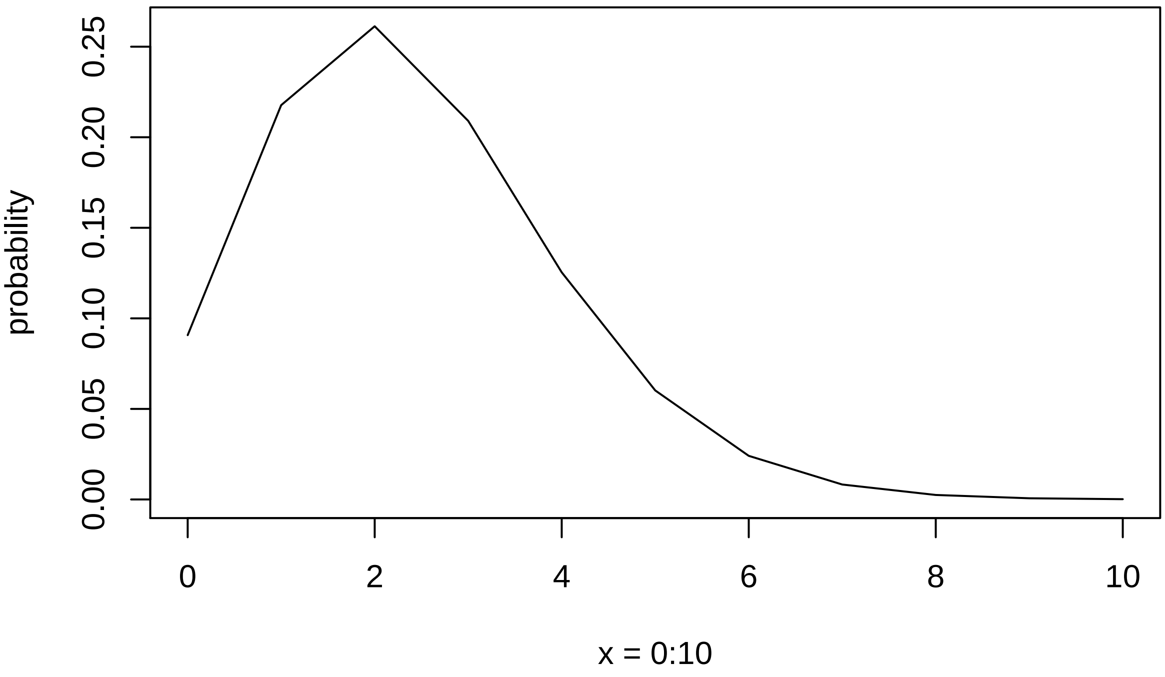
For example, if you wanted to know the probabilty of 5 people being admitted, calculate the cumulative probabilty of 5 or less, and subtract the cumulative probability of 4 or less:

```
ppois(q = 5, lambda = 2.4) - ppois(q = 4, lambda = 2.4)
```

```
## [1] 0.06019608
```

To do this for every discrete value can be tedious to code, but there is another function, **dpois()** that helps. If you supply it with a vector of numbers, it will evaluate the probabilty density at each of those values:

```
probs <- dpois(0:10, lambda = 2.4)
plot(x = 0:10, y = probs, type = "l", xlab = "x = 0:10", ylab = "probability")
```



Note that these values indicate the probability of 0, 1, 2, 3, … 10 people admitted during this time period, given a mean number of admittances of 2.4, and each admittance being independent and occurring with equal probability.

Answer question 6 on Sakai

Truffles

Truffles are a great delicacy. A set of plots of equal size in an old-growth forest in Northern California was surveyed to count the number of truffles per plot. The resulting distribution is found in the file Truffles.csv. The data include a total of 288 plots, and have a mean value of 0.604 truffles per plot. You are interested in determining whether truffles are randomly located around the forest. If not, they may be either clumped or dispersed.

```
d <- read.csv("truffles.csv")
str(d)
```

```
## 'data.frame':    5 obs. of  2 variables:
##  $ TrufflesPerPlot: int  0 1 2 3 4
##  $ Frequency      : int  203 39 18 13 15
```

```
d$Frequency
```

```
## [1] 203  39  18  13  15
```

You might be tempted to perform a chi-squared test on these 5 categories, but remember what the default parameters are when testing chi-squared:

```
result <- chisq.test(d$Frequency)
print(result)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  d$Frequency
## X-squared = 466.31, df = 4, p-value < 2.2e-16
```

```
result$expected
```

```
## [1] 57.6 57.6 57.6 57.6 57.6
```

Calculate the probabilities of 0 – 4 truffles per plot using the **dpois()** distribution function.

```
# Here is the probability for 0 and 1 truffles per plot.
dpois(0, lambda = 0.604)
```

```
## [1] 0.5466208
```

```
dpois(1, lambda = 0.604)
```

```
## [1] 0.3301589
```

Calculate the expected frequencies for each number of truffles per plot (0 – 4) according to the expected probabilities of the Poisson distribution and the total number of truffles collected. Try using R to perform this calculation directly.

Hint: Once you have a vector of probabilities, you can multiply it out by the sum of the total observations. Create a new variable called freq to contain this output.

Note that we cannot run a Chi-square test on this data, because one of our expected values is too low. In a Chi-square test, no expected value can be less than 1. To get around this, you will have to combine the data for 3 and 4 truffles per plot into a single value. Think about your categories as 0, 1, 2, and (3 to 4).

Run a Chi-square test on the data. Enter each of the expected frequencies you computed.

Answer question 7 on Sakai

Exercise 3: Chi-Square for Contingency Analysis

The Chi-Square test can also be used to analyze whether two categorical variables are associated. That is, whether the two variables are independent, or whether the outcome of one variable depends on the other.

The gnarly worm gets the bird

Example 9.4 in your book examines the life cycle of a parasite that is transferred from snails, to fish, to birds. The researchers noticed that infected fish spend more time near the water surface, and wanted to examine whether this influenced their chances of being ingested by a bird. An outdoor tank was stocked with uninfected, lightly infected, and heavily infected fish. The number of each that were eaten by birds was recorded. This data could be presented in a contingency table as follows:

	Uninfected	Lightly Infected	Highly Infected	Total
Eaten	1	10	37	48
Not Eaten	49	35	9	93
Total	50	45	46	191

Remember that the response variable (eaten/not eaten in this case) is displayed in rows, while the explanatory variable is displayed in columns. This data is contained in the files worms.csv. We can use a Chi-Square test to determine whether parasite infection and being eaten are independent (H_0) or not independent (H_a).

Note that in this case we want to use Chi-Square to compare associations between two categorical variables.

To be consistent with the contingency table, add the fate variable to the rows and the infection variable to the columns. You can use the built-in **xtabs()** to visualise the table:

```
d <- read.csv("worms.csv")
str(d)
```

```
## 'data.frame':    141 obs. of  2 variables:
##  $ infection: Factor w/ 3 levels "highly","lightly",..: 3 2 2 2 2 2 2 2 2 2 ...
##  $ fate     : Factor w/ 2 levels "eaten","not eaten": 1 1 1 1 1 1 1 1 1 1 ...
```

```
xtabs(~d$fate + d$infection)
```

```
##            d$infection
## d$fate      highly lightly uninfected
##   eaten         37      10          1
##   not eaten      9      35         49
```

To do a crosstabulation in R, you will need to install the packge “gmodels”, using the install.packages(“gmodels”) command.

Run this line without the comment (#) if you don't have gmodels installed:

```
# install.packages('gmodels')
```

2-Way Cross Tabulation

```
library(gmodels)
CrossTable(d$fate, d$infection, expected = TRUE, prop.c = FALSE, prop.r = FALSE,
    fisher = TRUE, format = "SAS")
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |              Expected N |
## | Chi-square contribution |
## |           N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  141
##
##
##              | d$infection
##       d$fate |     highly |    lightly | uninfected |  Row Total |
## -------------|------------|------------|------------|------------|
##        eaten |         37 |         10 |          1 |         48 |
##              |     15.660 |     15.319 |     17.021 |            |
##              |     29.082 |      1.847 |     15.080 |            |
##              |      0.262 |      0.071 |      0.007 |            |
## -------------|------------|------------|------------|------------|
##    not eaten |          9 |         35 |         49 |         93 |
##              |     30.340 |     29.681 |     32.979 |            |
##              |     15.010 |      0.953 |      7.783 |            |
##              |      0.064 |      0.248 |      0.348 |            |
## -------------|------------|------------|------------|------------|
## Column Total |         46 |         45 |         50 |        141 |
## -------------|------------|------------|------------|------------|
##
##
## Statistics for All Table Factors
##
##
## Pearson's Chi-squared test
## ------------------------------------------------------------
## Chi^2 =  69.75571     d.f. =  2     p =  7.124282e-16
##
##
##
## Fisher's Exact Test for Count Data
## ------------------------------------------------------------
## Alternative hypothesis: two.sided
## p =  1.369809e-17
```

```
##
##
```

Review the crosstabulation table. This is similar to a contingency table, but shows both the observed and expected counts. If fisher=TRUE, the results display the p value for a Fisher Exact test, which is required when any expected frequency is low (typically < 5).

In the Chi-Square test table we are interested in the Pearson Chi-Square results (what are labelled as the Chi-squared contributions, which are summed to obtain the actual Chi-square value).

Answer question 8 on sakai

Small fry

A study by Miller et al examined the survival of rainbow trout fry (babies) in Lake Superior, comparing those that came from a government hatchery on the lake, and those that came from wild trout. The data is contained in fry.csv.

Open this file and conduct a Chi-Square test through the crosstab function, as above. Add frysource to the columns and survival to the rows. Display the observed and expected counts.

Note that in your output, since this is a 2 x 2 comparison, the results for Fisher's Exact Test can also be displayed if the expected values are too low. The P-value for Fisher's exact test is an exact P-value, rather than a close approximation as obtained with Chi-Square. Fisher's Exact test must be used whenever the expected frequencies are too low for the Chi-Square test, but it is difficult to compute by hand.

Answer question 9 on sakai

Exercise 4: Odds Ratio & Relative Risk

An odds ratio measures the magnitude of association between two categorical variables when each variable has only two categories. It is calculated from the odds of a focal outcome in one group, divided by the odds of the same outcome in a second group. Odds are calculated by the probability of the focal outcome (success) divided by the probability of the alternate outcome (failure). A similar measure used in medical studies is relative risk, which is equal to the probability of an undesired outcome in the treatment group, divided by the probability of the same undesired outcome in the control group.

Postnatal depression

Postnatal depression affects approximately 8 – 15% of new mothers. Patel et al (2005) examined whether the rates of postnatal depression differed between mothers who delivered vaginally (control), compared to mothers who delivered by C-section (treatment). The data is found in delivery.csv.

Open the file delivery.csv.

Perform a cross tabulation analysis.

```r
d <- read.csv("delivery.csv")
str(d)
```

```
## 'data.frame':    10934 obs. of  2 variables:
##  $ Delivery  : Factor w/ 2 levels "C-section","vaginal  ": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Depression: Factor w/ 2 levels "Depression   ",..: 2 2 2 2 2 2 2 2 2 2 ...
```

```r
counts <- table(d$Delivery, d$Depression)
counts
```

```
##
##              Depression    No depression
##   C-section            48           341
##   vaginal            1025          9520
```

For the graph below, the rownames will be too large to show on the plot, so we will change to abbreviations:

```r
rownames(counts) <- c("CS", "V")
addmargins(counts)
```

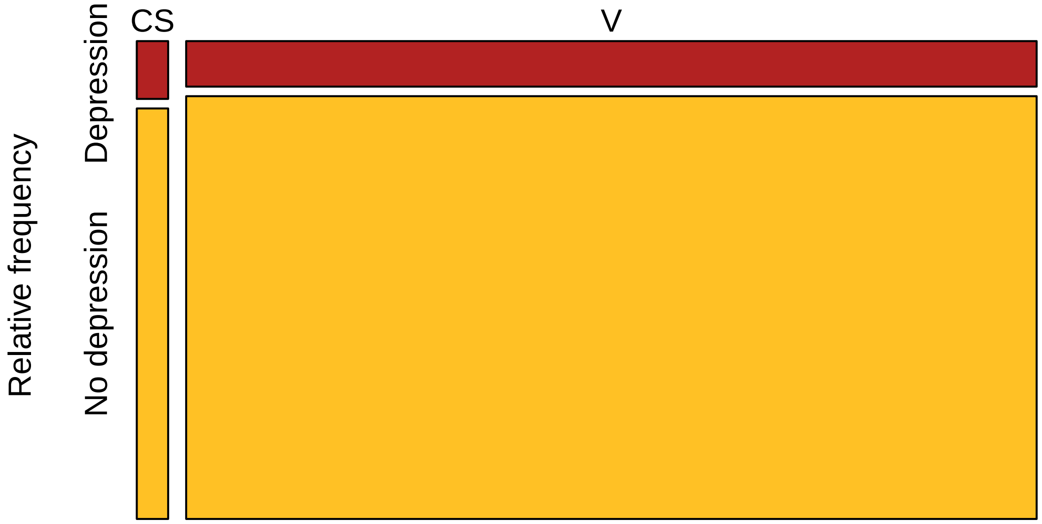
```
##
##        Depression    No depression   Sum
##   CS             48            341   389
##   V            1025           9520 10545
##   Sum          1073           9861 10934
```

```r
CrossTable(counts, expected = TRUE, prop.c = FALSE, prop.r = FALSE, format = "SAS")
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |              Expected N |
## | Chi-square contribution |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  10934
##
##
##              |
##              | Depression    | No depression |     Row Total |
## -------------|---------------|---------------|---------------|
##           CS |            48 |           341 |           389 |
##              |        38.174 |       350.826 |               |
##              |         2.529 |         0.275 |               |
##              |         0.004 |         0.031 |               |
## -------------|---------------|---------------|---------------|
##            V |          1025 |          9520 |         10545 |
##              |      1034.826 |      9510.174 |               |
##              |         0.093 |         0.010 |               |
##              |         0.094 |         0.871 |               |
## -------------|---------------|---------------|---------------|
## Column Total |          1073 |          9861 |         10934 |
## -------------|---------------|---------------|---------------|
##
##
## Statistics for All Table Factors
##
##
## Pearson's Chi-squared test
## ------------------------------------------------------------
## Chi^2 =  2.907728     d.f. =  1     p =  0.08815598
##
## Pearson's Chi-squared test with Yates' continuity correction
## ------------------------------------------------------------
## Chi^2 =  2.619329     d.f. =  1     p =  0.1055691
##
##
```

We can create a Mosaic plot of the association:

```
mosaicplot(counts, col = c("firebrick", "goldenrod1"), cex.axis = 1, ylab = "Relative frequency",
    main = "")
```



You can display the proportion of those experiencing the outcome, but within a row category using the **prop.table()** function:

```
prop.table(counts, margin = 1)
```

```
##
##      Depression   No depression
##   CS    0.12339332    0.87660668
##   V     0.09720247    0.90279753
```

```
addmargins(prop.table(counts, margin = 1))
```

```
##
##       Depression   No depression       Sum
##   CS     0.12339332    0.87660668 1.00000000
##   V      0.09720247    0.90279753 1.00000000
##   Sum    0.22059578    1.77940422 2.00000000
```

```
prop.table(counts, margin = 1) * 100  # multiply the proportions * 100 for percent
```

```
##
##      Depression   No depression
##   CS    12.339332     87.660668
##   V      9.720247     90.279753
```

Setting margin=1 means that the proportion will be calculated within each row, separately from each other row. Thus, each row should add up to 1, as demonstrated with the **addmargins()** function. For percentages, you can multiple the entire prop.table by 100.

Note that for determining odds and relative risk in R, the risk factor (delivery method) must be a row item, and the outcome we are interested in (depression) must be a column item (columns are often the 'disease' or outcome states). This was achieved in the **CrossTable()** function by the order of the variables.

For future reference, you could build the numbers into your own variable if you don't have them saved in a file, but it requires knowing how the **matrix()** function operates:

```
counts <- matrix(c(48, 1025, 341, 9520), nrow = 2, dimnames = list(c("C-Section",
    "vaginal"), c("Depression", "No Depression")))
counts
```

```
##           Depression No Depression
## C-Section         48           341
## vaginal         1025          9520
```

Next, calculate the odds ratio using the **epitools** package. Install the package if you haven't already done so, using **'install.packages("epitools", dependencies=TRUE)'**

```
# install.packages('epitools', dependencies = TRUE)
library(epitools)
```

This **oddratio()** function expects the following table struture for your contingency table, with the reference condition usually in the first row, and disease states labeled according to reference of interest, where 0 is usually the state being compared to:

	disease=0	disease=1
exposed=0 (ref)	n00	n01
exposed=1	n10	n11
exposed=2	n20	n21
exposed=4	n30	n31

The reason for this is because each level of exposure is compared to the reference level (row 1).

This layout looks a little different from what our **counts** data look like, but the **oddsratio()** function allows you to reverse the order of the rows and columns by setting rev="both".

Here is the default layout:

```
counts
```

```
##           Depression No Depression
## C-Section         48           341
## vaginal         1025          9520
```

The layout the oddratio function wants, where row 1 and 2 are swapped and column 1 and 2 are swapped:

```
counts[2:1, 2:1]
```

```
##           No Depression Depression
## vaginal            9520       1025
## C-Section           341         48
```

You can specify the reversal method when you call the oddratio function:

```
oddsratio(counts, rev = "both", method = "wald")
```

```
## $data
##            No Depression Depression Total
## vaginal            9520       1025 10545
## C-Section           341         48   389
## Total              9861       1073 10934
##
## $measure
##                          NA
## odds ratio with 95% C.I. estimate    lower    upper
##                vaginal   1.000000       NA       NA
##                C-Section 1.307374 0.959902 1.780627
##
## $p.value
##            NA
## two-sided   midp.exact fisher.exact chi.square
##   vaginal           NA           NA         NA
##   C-Section 0.09620641   0.09847188 0.08815598
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "Unconditional MLE & normal approximation (Wald) CI"
```

In the $measure section, it shows the odds of developing depression with a C-section delivery compared to a vaginal delivery. If this number is greater than 1, then the odds of developing depression are higher with a C-section delivery. If this value is less than 1, then the odds of developing depression are lower with a C-section delivery. The first row odds ratio is usually the reference state, so it will not have confidence limits and will be equal to 1.

The resulting output of oddsratio includes a lot of incidental computations. To get clean output only for the odds ratio, including the 95% confidence interval, use the following instead:

```
oddsratio(counts, rev = "both", method = "wald")$measure[-1, ]
```

```
## estimate    lower    upper
## 1.307374 0.959902 1.780627
```

Do the confidence intervals exclude or include 1?

Now, calculate relative risk using the **epitools** package. To use the command with our counts contingency table, we need to reverse the order of the rows and columns, by setting the rev option = “both”. We can do this all at once with the following arguments to the riskratio function.

```
library(epitools)
riskratio(counts, rev = "both", method = "wald")
```

```
## $data
##            No Depression Depression Total
## vaginal            9520       1025 10545
## C-Section           341         48   389
## Total              9861       1073 10934
##
## $measure
##                         NA
## risk ratio with 95% C.I. estimate     lower    upper
##                vaginal   1.000000        NA       NA
##                C-Section 1.269446 0.9679264 1.664893
##
## $p.value
##            NA
## two-sided   midp.exact fisher.exact chi.square
##   vaginal           NA           NA         NA
##   C-Section 0.09620641   0.09847188 0.08815598
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "Unconditional MLE & normal approximation (Wald) CI"
```

The $measure section shows the relative risk for the outcome depression, with a C-section compared to vaginal birth.

Return to crosstabs, and conduct a Chi-Square test on the data to see if there is a significant difference in these outcomes. This test will tell you if the two variables (delivery method and depression) are independent.

Answer questions 10 and 11 on Sakai.

Exercise 5: Additional practice:

Heart failure

The file heart failure.csv contains data on the day of the week that patients were admitted to a hospital with heart failure. Analyze this data using a Chi-Square test to determine if patients are admitted in equal proportions on each day of the week.

Feline high rise syndrome

A more recent study of feline high-rise syndrome (FHRS) included data on the month in which each of 119 cats fell. The data are found in the file Rainingcats.csv. Do a hypothesis test to determine whether the probability of FHRS is the same every month.

MS and CCSVI

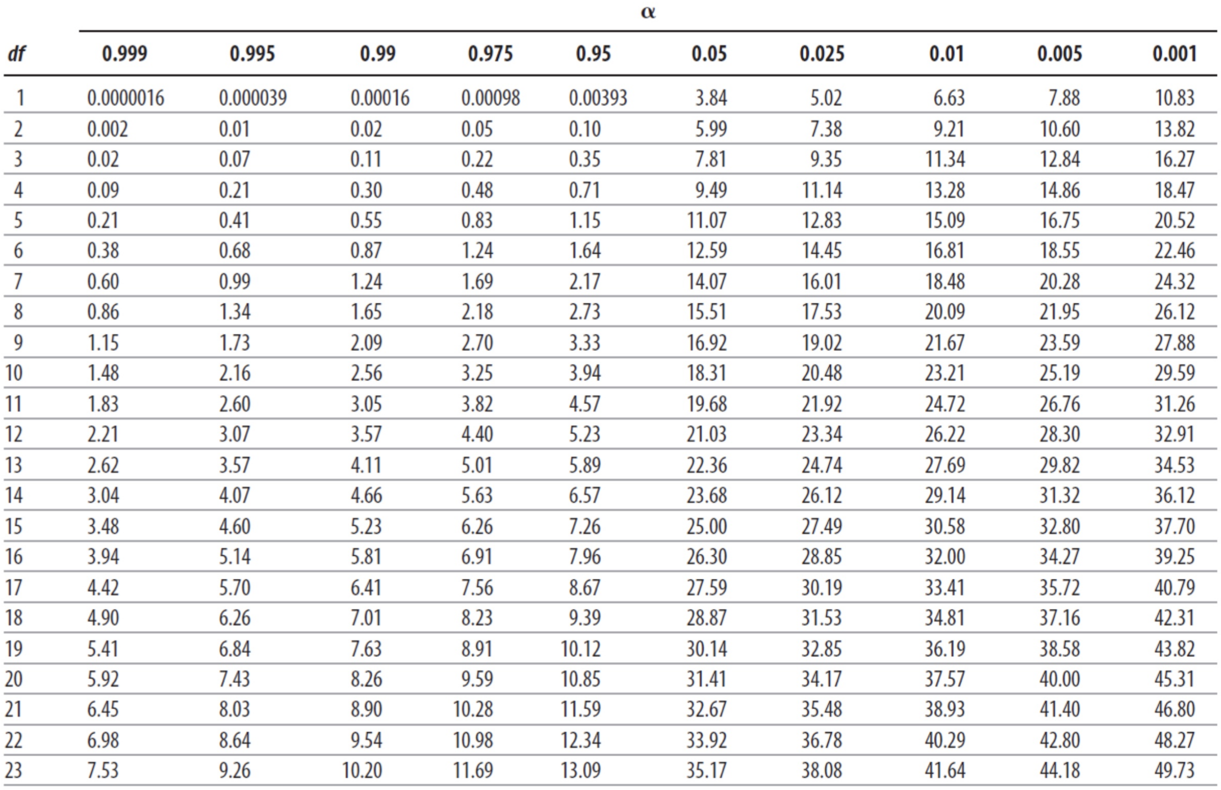
In 2012, a research group examined the association between MS and a vein condition known as chronic cerebrospinal venous insufficiency (CCSV). This data is found in the file CCSVI.csv. Conduct a Chi-Square test to determine whether there is a significant association between MS and CCSVI.

Smoking Fingers

Researchers examined the presence of finger defects (fused fingers, extra fingers, or less than five fingers) in births of mothers who smoked (risk factor) during pregnancy, compared to control mothers who did not smoke during pregnancy. The data is found in the file Smoking fingers.csv. Calculate the Odds ratio and 95% confidence interval of the odds ratio. Use a Chi-Square test to determine if this difference is significant. Don't forget that the risk factor must go in the row, and the outcome in the column.

Answer questions 12-16 on sakai

Chi Squared Statistical Table:



	α									
df	**0.999**	**0.995**	**0.99**	**0.975**	**0.95**	**0.05**	**0.025**	**0.01**	**0.005**	**0.001**
1	0.0000016	0.000039	0.00016	0.00098	0.00393	3.84	5.02	6.63	7.88	10.83
2	0.002	0.01	0.02	0.05	0.10	5.99	7.38	9.21	10.60	13.82
3	0.02	0.07	0.11	0.22	0.35	7.81	9.35	11.34	12.84	16.27
4	0.09	0.21	0.30	0.48	0.71	9.49	11.14	13.28	14.86	18.47
5	0.21	0.41	0.55	0.83	1.15	11.07	12.83	15.09	16.75	20.52
6	0.38	0.68	0.87	1.24	1.64	12.59	14.45	16.81	18.55	22.46
7	0.60	0.99	1.24	1.69	2.17	14.07	16.01	18.48	20.28	24.32
8	0.86	1.34	1.65	2.18	2.73	15.51	17.53	20.09	21.95	26.12
9	1.15	1.73	2.09	2.70	3.33	16.92	19.02	21.67	23.59	27.88
10	1.48	2.16	2.56	3.25	3.94	18.31	20.48	23.21	25.19	29.59
11	1.83	2.60	3.05	3.82	4.57	19.68	21.92	24.72	26.76	31.26
12	2.21	3.07	3.57	4.40	5.23	21.03	23.34	26.22	28.30	32.91
13	2.62	3.57	4.11	5.01	5.89	22.36	24.74	27.69	29.82	34.53
14	3.04	4.07	4.66	5.63	6.57	23.68	26.12	29.14	31.32	36.12
15	3.48	4.60	5.23	6.26	7.26	25.00	27.49	30.58	32.80	37.70
16	3.94	5.14	5.81	6.91	7.96	26.30	28.85	32.00	34.27	39.25
17	4.42	5.70	6.41	7.56	8.67	27.59	30.19	33.41	35.72	40.79
18	4.90	6.26	7.01	8.23	9.39	28.87	31.53	34.81	37.16	42.31
19	5.41	6.84	7.63	8.91	10.12	30.14	32.85	36.19	38.58	43.82
20	5.92	7.43	8.26	9.59	10.85	31.41	34.17	37.57	40.00	45.31
21	6.45	8.03	8.90	10.28	11.59	32.67	35.48	38.93	41.40	46.80
22	6.98	8.64	9.54	10.98	12.34	33.92	36.78	40.29	42.80	48.27
23	7.53	9.26	10.20	11.69	13.09	35.17	38.08	41.64	44.18	49.73