431 Class 12

https://thomaselove.github.io/431-2024/

2024-10-03

## Today’s Agenda

Contingency Tables (Sections 13-14 in Course Book)

* Some Reminders about 2 x 2 tables
* Building a J x K Table
* Chi-Square Tests of Independence
  + Cochran Conditions and Checking Assumptions

## Today’s Packages

library(janitor)  
library(patchwork)  
library(easystats)  
library(tidyverse)  
  
theme\_set(theme\_bw())  
  
source("c12/data/Love-431.R")

## TB Prevalence in IV Drug Users

Suppose now that we are investigating factors affecting tuberculosis prevalence among intravenous drug users.

* Among 97 individuals who admit to sharing needles,
  + 24 (24.7%) had a positive tuberculin skin test result.
* Among 161 drug users who deny sharing needles,
  + 28 (17.4%) had a positive test result.

What does the 2x2 table look like?

## TB Prevalence In IV Drug Users

The 2x2 Table is…

TB+ TB-  
 share 24 73  
 don't 28 133

* Standard Epidemiological Format
  + rows describe needle sharing
  + columns describe TB test result
* row 1 people who share needles: 24 TB+, and 97-24 = 73 TB-
* row 2 people who don’t share: 28 TB+ and 161-28 = 133 TB-

## twobytwo (with Bayesian Augmentation)

To start, we’ll test the null hypothesis that the population proportions of intravenous drug users who have a positive tuberculin skin test result are identical for those who share needles and those who do not.

$$
H\_0: \pi\_{share} = \pi\_{donotshare} \\
H\_A: \pi\_{share} \neq \pi\_{donotshare}
$$

We’ll use the Bayesian augmentation.

## twobytwo results (95% CI)

twobytwo(24+2, 73+2, 28+2, 133+2,   
 "Sharing", "Not Sharing",   
 "TB test+", "TB test-", conf.level = 0.95)

2 by 2 table analysis:   
------------------------------------------------------   
Outcome : TB test+   
Comparing : Sharing vs. Not Sharing   
  
 TB test+ TB test- P(TB test+) 95% conf. interval  
Sharing 26 75 0.2574 0.1816 0.3513  
Not Sharing 30 135 0.1818 0.1301 0.2482  
  
 95% conf. interval  
 Relative Risk: 1.4158 0.8910 2.2498  
 Sample Odds Ratio: 1.5600 0.8594 2.8318  
Conditional MLE Odds Ratio: 1.5572 0.8189 2.9511  
 Probability difference: 0.0756 -0.0244 0.1819  
  
 Exact P-value: 0.1638   
 Asymptotic P-value: 0.1438   
------------------------------------------------------

## Change to 90% confidence?

* What if we use a 90% confidence level instead?

2 by 2 table analysis:   
------------------------------------------------------   
Outcome : TB test+   
Comparing : Sharing vs. Not Sharing   
  
 TB test+ TB test- P(TB test+) 90% conf. interval  
Sharing 26 75 0.2574 0.1925 0.3351  
Not Sharing 30 135 0.1818 0.1375 0.2365  
  
 90% conf. interval  
 Relative Risk: 1.4158 0.9599 2.0884  
 Sample Odds Ratio: 1.5600 0.9458 2.5729  
Conditional MLE Odds Ratio: 1.5572 0.9023 2.6818  
 Probability difference: 0.0756 -0.0088 0.1646  
  
 Exact P-value: 0.1638   
 Asymptotic P-value: 0.1438   
------------------------------------------------------

# Larger Contingency Tables

## A contingency table

This table displays the count of patients who show *complete*, *partial*, or *no response* after treatment with either **active** medication or a **placebo** in a study of 100 patients…

| Group | None | Partial | Complete |
| --- | --- | --- | --- |
| Active | 8 | 24 | 20 |
| Placebo | 12 | 26 | 10 |

Is there a meaningful association here?

## The Pearson Chi-Square Test

* : Response Distribution is the same, regardless of Treatment.
* : There is an association between Treatment and Response.

The Pearson test assumes the null hypothesis is true (rows and columns are independent.) That is a model for our data. How does it work?

## Calculating Chi-Square

Here’s the table, with marginal totals added.

| – | None | Partial | Complete | **TOTAL** |
| --- | --- | --- | --- | --- |
| Active | 8 | 24 | 20 | **52** |
| Placebo | 12 | 26 | 10 | **48** |
| **TOTAL** | **20** | **50** | **30** | **100** |

The test needs to estimate the expected frequency in each of the six cells under the assumption of independence. If the rows and columns were independent, what is the expected count in the Active/None cell?

## The Independence Model

| – | None | Partial | Complete | **TOTAL** |
| --- | --- | --- | --- | --- |
| Active | – | – | – | **52** |
| Placebo | – | – | – | **48** |
| **TOTAL** | **20** | **50** | **30** | **100** |

If the rows and columns were independent, then:

* 20/100 of subjects would have response = “None”
  + That’s 20% of the 52 Active, and 20% of the 48 Placebo
* 50% would have a “Partial” response, and
* 30% would have a “Complete” response in each group.

## Observed (*Expected*) Cell Counts

So, can we fill in the expected frequencies under our independence model?

| – | None | Partial | Complete | **TOTAL** |
| --- | --- | --- | --- | --- |
| Active | 8 (*10.4*) | 24 (*26.0*) | 20 (*15.6*) | **52** |
| Placebo | 12 (*9.6*) | 26 (*24.0*) | 10 (*14.4*) | **48** |
| **TOTAL** | **20** | **50** | **30** | **100** |

## General Formula for Expected Frequencies under Independence

This assumes that the independence model holds: the probability of being in a particular column is exactly the same in each row, and vice versa.

## Chi-Square Assumptions

* Expected Frequencies: We assume that the expected frequency, under the null hypothesized model of independence, will be **at least 5** (and ideally at least 10) in each cell. If that is not the case, then the test is likely to give unreliable results.
* The *Cochran conditions* require us to have no cells with zero counts **and** at least 80% of the cells in our table with expected counts of 5 or higher. That’s what R uses to warn you of trouble.
* Don’t meet the standards? Consider collapsing categories.

## Observed (**Expected**) Cell Counts (again)

| – | None | Partial | Complete | **TOTAL** |
| --- | --- | --- | --- | --- |
| Active | 8 (**10.4**) | 24 (**26.0**) | 20 (**15.6**) | 52 |
| Placebo | 12 (**9.6**) | 26 (**24.0**) | 10 (**14.4**) | 48 |
| TOTAL | 20 | 50 | 30 | 100 |

* Do we meet the Cochran conditions in this case?

## Getting the Table into R

We’ll put the table into a matrix in R. Here’s one approach…

T1 <- matrix(c(8, 24, 20, 12, 26, 10),   
 ncol=3, nrow=2, byrow=TRUE)  
rownames(T1) <- c("Active", "Placebo")  
colnames(T1) <- c("None", "Partial", "Complete")  
T1

None Partial Complete  
Active 8 24 20  
Placebo 12 26 10

chisq.test(T1)

Pearson's Chi-squared test  
  
data: T1  
X-squared = 4.0598, df = 2, p-value = 0.1313

## Chi-Square Test Results in R

* : Response Distribution is the same, regardless of Treatment.
  + Rows and Columns of the table are *independent*
* : There is an association between Treatment and Response.
  + Rows and Columns of the table are *associated*.
* For our T1, the results were: = 4.0598, df = 2, *p* = 0.1313

What is the conclusion?

## Does Sample Size Affect The Test?

* T1 results were: = 4.0598, df = 2, *p* = 0.1313
* What if we had the same pattern, but twice as much data?

T1\_doubled <- T1\*2  
T1\_doubled

None Partial Complete  
Active 16 48 40  
Placebo 24 52 20

chisq.test(T1\_doubled)

Pearson's Chi-squared test  
  
data: T1\_doubled  
X-squared = 8.1197, df = 2, p-value = 0.01725

## Fisher’s exact test instead?

Yes, but … if the Pearson assumptions don’t hold, then the Fisher’s test is not generally an improvement.

fisher.test(T1)

Fisher's Exact Test for Count Data  
  
data: T1  
p-value = 0.1358  
alternative hypothesis: two.sided

* Intended for small-ish square tables, with the same number of rows as columns.

## Returning to the DM-464 data

* We discussed these data (other variables) in Classes 5-7.

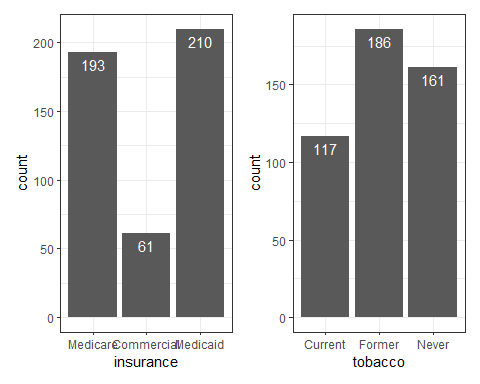
dm1 <- read\_csv("c12/data/dm464\_class12.csv", show\_col\_types = FALSE) |>  
 janitor::clean\_names() |>  
 mutate(tobacco = fct\_relevel(tobacco, "Current", "Former"),  
 insurance = fct\_relevel(insurance, "Medicare",   
 "Commercial", "Medicaid"))  
  
dm1 |> tabyl(tobacco, insurance) |>   
 adorn\_totals(where = c("row", "col"))

tobacco Medicare Commercial Medicaid Total  
 Current 33 8 76 117  
 Former 92 23 71 186  
 Never 68 30 63 161  
 Total 193 61 210 464

## dm1: Bar Plots with Counts

p1 <- ggplot(dm1, aes(x = insurance)) + geom\_bar() +   
 geom\_text(aes(label = ..count..), stat = "count",   
 vjust = 1.5, col = "white")  
  
p2 <- ggplot(dm1, aes(x = tobacco)) + geom\_bar() +   
 geom\_text(aes(label = ..count..), stat = "count",   
 vjust = 1.5, col = "white")  
  
p1 + p2

Warning: The dot-dot notation (`..count..`) was deprecated in ggplot2 3.4.0.  
ℹ Please use `after\_stat(count)` instead.



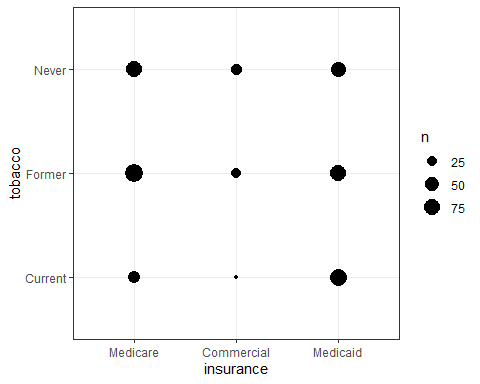
## A table with the dm1 data

dm1 |>   
 tabyl(insurance, tobacco) |>  
 adorn\_totals(where = c("row", "col"))

insurance Current Former Never Total  
 Medicare 33 92 68 193  
 Commercial 8 23 30 61  
 Medicaid 76 71 63 210  
 Total 117 186 161 464

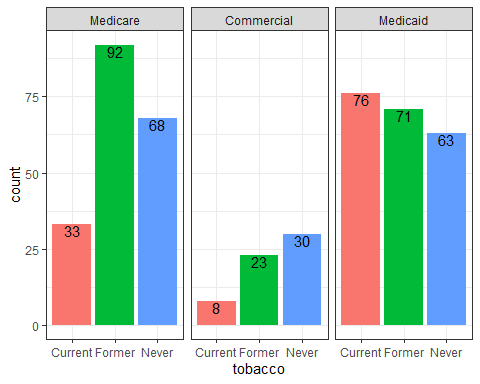
## Plotting a Cross-Tabulation?

ggplot(dm1, aes(x = insurance, y = tobacco)) +  
 geom\_count()



## Tobacco Bar Chart faceted by Insurance

ggplot(dm1, aes(x = tobacco, fill = tobacco)) +   
 geom\_bar() + facet\_wrap(~ insurance) +  
 guides(fill = "none") +   
 geom\_text(aes(label = ..count..), stat = "count",   
 vjust = 1, col = "black")



## Tobacco Status and Insurance in dm1

* : Insurance type and Tobacco status are independent
* : Insurance type and Tobacco status have a detectable association

Pearson results?

dm1 |> tabyl(insurance, tobacco) |> chisq.test()

Pearson's Chi-squared test  
  
data: tabyl(dm1, insurance, tobacco)  
X-squared = 28.574, df = 4, p-value = 9.542e-06

Can we check our expected frequencies?

## Checking Expected Frequencies

res <- dm1 |> tabyl(insurance, tobacco) |> chisq.test()  
  
res$observed

insurance Current Former Never  
 Medicare 33 92 68  
 Commercial 8 23 30  
 Medicaid 76 71 63

res$expected

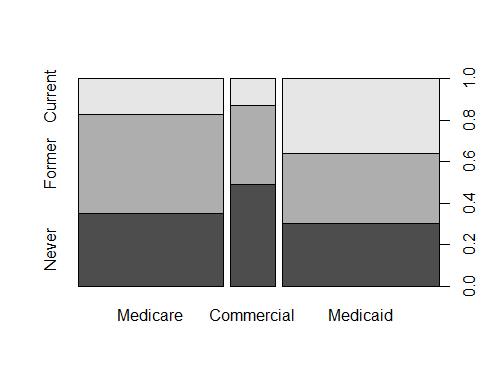
insurance Current Former Never  
 Medicare 48.66595 77.36638 66.96767  
 Commercial 15.38147 24.45259 21.16595  
 Medicaid 52.95259 84.18103 72.86638

Any problems with Cochran conditions?

## Mosaic Plot for Cross-Tabulation

Each rectangle’s area is proportional to the number of cases in that cell.

plot(dm1$insurance, dm1$tobacco, ylab = "", xlab = "")



## Session Information

xfun::session\_info()

R version 4.4.1 (2024-06-14 ucrt)  
Platform: x86\_64-w64-mingw32/x64  
Running under: Windows 11 x64 (build 22631)  
  
Locale:  
 LC\_COLLATE=English\_United States.utf8   
 LC\_CTYPE=English\_United States.utf8   
 LC\_MONETARY=English\_United States.utf8  
 LC\_NUMERIC=C   
 LC\_TIME=English\_United States.utf8   
  
Package version:  
 askpass\_1.2.0 backports\_1.5.0 base64enc\_0.1.3   
 bayestestR\_0.14.0 bit\_4.0.5 bit64\_4.0.5   
 blob\_1.2.4 broom\_1.0.6 bslib\_0.8.0   
 cachem\_1.1.0 callr\_3.7.6 cellranger\_1.1.0   
 cli\_3.6.3 clipr\_0.8.0 cmprsk\_2.2-12   
 coda\_0.19-4.1 codetools\_0.2-20 colorspace\_2.1-1   
 compiler\_4.4.1 conflicted\_1.2.0 correlation\_0.8.5   
 cpp11\_0.5.0 crayon\_1.5.3 curl\_5.2.2   
 data.table\_1.16.0 datasets\_4.4.1 datawizard\_0.12.3   
 DBI\_1.2.3 dbplyr\_2.5.0 digest\_0.6.37   
 dplyr\_1.1.4 dtplyr\_1.3.1 easystats\_0.7.3   
 effectsize\_0.8.9 emmeans\_1.10.4 Epi\_2.55   
 estimability\_1.5.1 etm\_1.1.1 evaluate\_1.0.0   
 fansi\_1.0.6 farver\_2.1.2 fastmap\_1.2.0   
 fontawesome\_0.5.2 forcats\_1.0.0 fs\_1.6.4   
 gargle\_1.5.2 generics\_0.1.3 ggplot2\_3.5.1   
 glue\_1.7.0 googledrive\_2.1.1 googlesheets4\_1.1.1   
 graphics\_4.4.1 grDevices\_4.4.1 grid\_4.4.1   
 gtable\_0.3.5 haven\_2.5.4 highr\_0.11   
 hms\_1.1.3 htmltools\_0.5.8.1 httr\_1.4.7   
 ids\_1.0.1 insight\_0.20.4 isoband\_0.2.7   
 janitor\_2.2.0 jquerylib\_0.1.4 jsonlite\_1.8.9   
 knitr\_1.48 labeling\_0.4.3 lattice\_0.22-6   
 lifecycle\_1.0.4 lubridate\_1.9.3 magrittr\_2.0.3   
 MASS\_7.3-61 Matrix\_1.7-0 memoise\_2.0.1   
 methods\_4.4.1 mgcv\_1.9-1 mime\_0.12   
 modelbased\_0.8.8 modelr\_0.1.11 multcomp\_1.4-26   
 munsell\_0.5.1 mvtnorm\_1.3-1 nlme\_3.1-164   
 numDeriv\_2016.8-1.1 openssl\_2.2.1 parallel\_4.4.1   
 parameters\_0.22.2 patchwork\_1.3.0 performance\_0.12.3   
 pillar\_1.9.0 pkgconfig\_2.0.3 plyr\_1.8.9   
 prettyunits\_1.2.0 processx\_3.8.4 progress\_1.2.3   
 ps\_1.8.0 purrr\_1.0.2 R6\_2.5.1   
 ragg\_1.3.2 rappdirs\_0.3.3 RColorBrewer\_1.1.3   
 Rcpp\_1.0.13 RcppArmadillo\_14.0.2.1 readr\_2.1.5   
 readxl\_1.4.3 rematch\_2.0.0 rematch2\_2.1.2   
 report\_0.5.9 reprex\_2.1.1 rlang\_1.1.4   
 rmarkdown\_2.28 rstudioapi\_0.16.0 rvest\_1.0.4   
 sandwich\_3.1-1 sass\_0.4.9 scales\_1.3.0   
 see\_0.9.0 selectr\_0.4.2 snakecase\_0.11.1   
 splines\_4.4.1 stats\_4.4.1 stringi\_1.8.4   
 stringr\_1.5.1 survival\_3.7-0 sys\_3.4.2   
 systemfonts\_1.1.0 textshaping\_0.4.0 TH.data\_1.1-2   
 tibble\_3.2.1 tidyr\_1.3.1 tidyselect\_1.2.1   
 tidyverse\_2.0.0 timechange\_0.3.0 tinytex\_0.53   
 tools\_4.4.1 tzdb\_0.4.0 utf8\_1.2.4   
 utils\_4.4.1 uuid\_1.2.1 vctrs\_0.6.5   
 viridisLite\_0.4.2 vroom\_1.6.5 withr\_3.0.1   
 xfun\_0.47 xml2\_1.3.6 xtable\_1.8-4   
 yaml\_2.3.10 zoo\_1.8-12