

# 431 Lab 05 Sketch and Grading Rubric

Instructor: Dr. Thomas E. Love

Lab Author: Mr. Wyatt P. Bensken

Due: ~~2021-10-18~~ 2021-10-21 | Last Edit: 2021-10-27 11:28:19

## Contents

<b>1</b>	<b>Loading Packages</b>	<b>2</b>
<b>2</b>	<b>Learning Objectives</b>	<b>2</b>
<b>3</b>	<b>Packages and Functions</b>	<b>2</b>
<b>4</b>	<b>Part A. County Health Rankings (Questions 1-6, 60 points)</b>	<b>3</b>
4.1	Question 1 (10 points) . . . . .	4
4.2	Question 2 (10 points) . . . . .	6
4.3	Question 3 (5 points) . . . . .	6
4.4	Question 4 (15 points) . . . . .	6
4.5	Question 5 (10 points) . . . . .	8
4.6	Question 6 (10 points) . . . . .	10
<b>5</b>	<b>Part B. An Observational Study (Questions 7-9, 40 points)</b>	<b>11</b>
5.1	Question 7 (10 points) . . . . .	11
5.2	Question 8 (10 points) . . . . .	14
5.3	Question 9 (20 points) . . . . .	15

# 1 Loading Packages

```
library(broom)
library(Epi)
library(janitor)
library(knitr)
library(magrittr)
library(naniar)
library(patchwork)
library(tidyverse)

source("data/Love-boost.R")

theme_set(theme_light())
```

## 2 Learning Objectives

1. Be able to take information about a dataset, and an associated visualization, to identify the appropriate inferential test when comparing means.
2. Develop and appropriately interpret a confidence interval, in context, as derived from an analysis of categorical variables.
3. Use some of the concepts developed in Chapter 6 of Spiegelhalter's *The Art of Statistics* to evaluate a model's performance.

## 3 Packages and Functions

This will be updated with each individual homework.

Packages:

- janitor
- Epi
- naniar
- patchwork
- broom
- magrittr
- tidyverse

Functions:

- read\_csv()
- filter()
- select()
- mutate()
- ggplot()
- lm()
- bootdif()
- tidy()
- twoby2()

## 4 Part A. County Health Rankings (Questions 1-6, 60 points)

We're going to revisit our County Health Rankings data, specifically using the same midwest counties we worked with in Lab 02, but starting from an expanded `lab05_counties.csv` data set.

First, you'll want to read in the `lab05_counties` data and filter to the states of Ohio (OH), Indiana (IN), Illinois (IL), Michigan (MI), and Wisconsin (WI). The main variables we'll be looking at in this Lab are `metro` and `access_to_exercise_opportunities`, but you'll go ahead and still want to keep `state` and `county_name` as well. Determine whether there are any missing values in that set of variables. We'll call the resulting tibble `midwest05`.

You've been asked by the principal investigator of a study to examine if there are differences in the mean percentage of adults with access to exercise opportunities between metropolitan and non-metropolitan counties, but we'll assume (for now) that you only have data within the five states (OH, IN, IL, MI and WI) we've identified above.

We'll need to start by reading in and filtering our dataset to the states and variables of interest.

```
midwest05 <- read_csv("data/lab05_counties.csv",
                      show_col_types = FALSE) %>%
  filter(state == "OH" | state == "IN" | state == "IL" |
         state == "MI" | state == "WI") %>%
  select(state, county_name, metro,
         access_to_exercise_opportunities)
```

Next, we'll go ahead and check to see if we have any missing values in our data.

```
miss_case_table(midwest05)
```

```
# A tibble: 1 x 3
  n_miss_in_case n_cases pct_cases
      <int>      <int>    <dbl>
1           0       437      100
```

We have no missing data.

Here's a quick summary of what we do have.

```
mosaic::inspect(midwest05)
```

```
categorical variables:
      name      class levels  n missing
1    state character     5 437        0
2 county_name character   320 437        0
      distribution
1 IL (23.3%), IN (21.1%), OH (20.1%) ...
2 Crawford County (1.1%) ...
```

```
quantitative variables:
      name      class  min    Q1 median    Q3
...1      metro numeric 0.0000 0.0000  0.000 1.0000
...2 access_to_exercise_opportunities numeric 0.0133 0.5835  0.703 0.8189
      max      mean      sd  n missing
...1 1.0000 0.3981693 0.4900818 437        0
...2 0.9963 0.6869918 0.1746083 437        0
```

## 4.1 Question 1 (10 points)

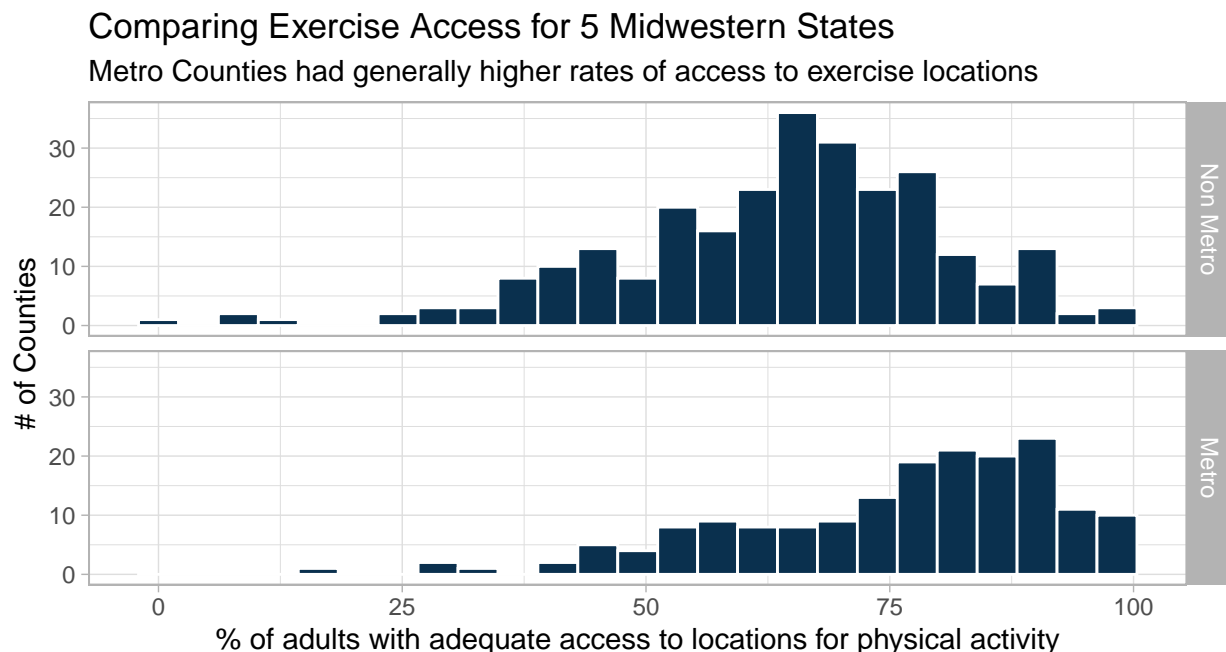
In Question 1, create a factor variable as part of the `midwest05` tibble which is better than the 0/1 that `metro` currently is, as well as adjust `access_to_exercise_opportunities` from a proportion to a percent and give it a shorter, more appealing name. Then please make an attractive and useful visualization which shows the distribution of the percent of adults with adequate access to exercise opportunities stratified by the county's metropolitan status. Finally, provide a couple of sentences describing your initial conclusions based on your visualization.

First, we'll create a factor for the `metro` data, then shorten the name of the `access_to_exercise_opportunities` variable and state it in terms of a percentage rather than a proportion.

```
midwest05 <- midwest05 %>%
  mutate(metro_f = fct_recode(as.factor(metro),
                              "Non Metro" = "0",
                              "Metro" = "1"),
         ex_access = access_to_exercise_opportunities*100)
```

As we've done a number of times now, there are numerous visualizations that may help us evaluate the distribution of `access_to_exercise_opportunities` between metropolitan and non-metropolitan counties.

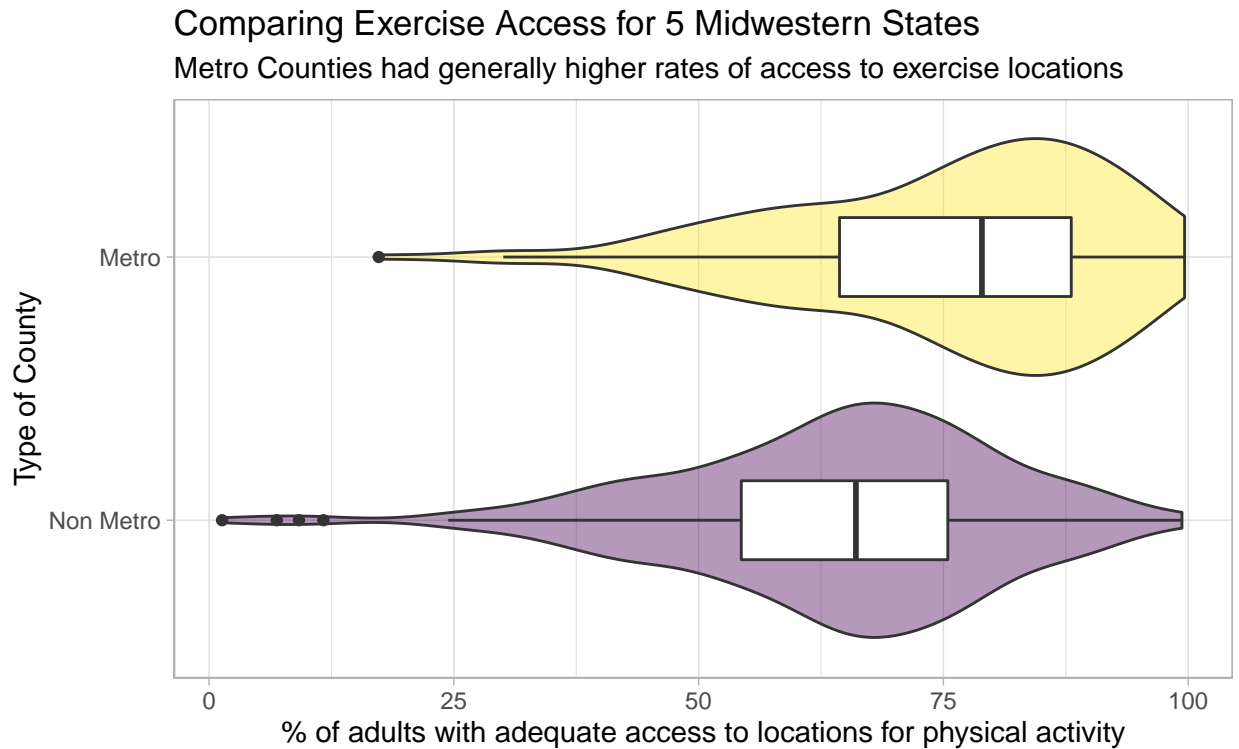
```
ggplot(midwest05, aes(x = ex_access)) +
  geom_histogram(bins = 25, fill = "#0a304e", col = "white") +
  facet_grid(metro_f ~ .) +
  labs(title = "Comparing Exercise Access for 5 Midwestern States",
       subtitle = "Metro Counties had generally higher rates of access to exercise locations",
       y = "# of Counties",
       x = "% of adults with adequate access to locations for physical activity")
```



We see generally higher percentages of adults with access to exercise opportunities in metropolitan counties, when compared to non-metropolitan counties. Notably, the distribution is fairly-left skewed and is certainly not well approximated by the Normal model.

Of course, a boxplot could also show these distributions.

```
ggplot(midwest05, aes(x = ex_access, y = metro_f)) +  
  geom_violin(aes(fill = metro_f)) +  
  geom_boxplot(width = 0.3) +  
  guides(fill = "none") +  
  scale_fill_viridis_d(alpha = 0.4) +  
  labs(title = "Comparing Exercise Access for 5 Midwestern States",  
       subtitle = "Metro Counties had generally higher rates of access to exercise locations",  
       y = "Type of County",  
       x = "% of adults with adequate access to locations for physical activity")
```



#### 4.1.1 Grading Rubric

- If the student has produced a visually pleasing figure, with a reasonable title and axes labels, which are not just the variable names, that allows one to examine this relationship, they should receive 10 points.
- If any of the following occur the student should lose 2 points for each:
  - No title, or a title which does not convey what the figure represents
  - Axes remain unlabeled or with the default labels
- If the figure has *substantial* issues beyond the above issues and is not interpretable, but a figure is provided, the student should lose up to 7 points and the TA should provide *very* detailed comments.

## 4.2 Question 2 (10 points)

In a few sentences, comment on the sampling approach used to create the data you used in Question 1. Are the data a random sample from the population(s) of interest? Is there at least a reasonable argument for generalizing from the sample to the population of all US counties? Or is there insufficient information provided on this point? How do you know?

Clearly this is **not** a random sample from the population of interest, assuming that the population of interest all US counties. Instead, we have selected specific states which fall in a narrow geographic region. There doesn't seem to be much of an argument to be made that this could be generalizable to the United States. We certainly do have enough information to determine this as part of our analysis was to select those counties.

### 4.2.1 Grading Rubric

- If a student correctly states that this is (a) not a random sample and (b) is not generalizable to all US Counties, they should receive the full 10 points.
  - They should lose 3 points for each of these items not mentioned.

## 4.3 Question 3 (5 points)

Are the data you developed in Question 1 matched / paired samples or independent samples? How do you know?

These data are quite clearly independent samples, there is no matching or pairing, but rather just the counties split into metropolitan and non-metropolitan status.

### 4.3.1 Grading Rubric

- If a student correctly states that these are independent samples they should receive all 5 points.
- If they state that these are paired samples they should lose all 5 points.

## 4.4 Question 4 (15 points)

Answer either part a or b of this question, based on your response to Question 3.

- a. If we have paired samples, what does the distribution of sample paired differences tell us about which inferential procedure to use? Display an appropriate visualization that motivates your conclusions, and then describe those conclusions in complete English sentences.
- b. If we have independent samples, what does the distribution of each individual sample tell us about which inferential procedure to use? Display an appropriate visualization that motivates your conclusions, and then describe those conclusions in complete English sentences.

As we have independent samples, we'll proceed that route and start by visualizing these data using Normal Q-Q plots, and boxplots with violins.

```
p1 <- ggplot(midwest05, aes(sample = ex_access)) +  
  geom_qq(aes(col = metro_f)) + geom_qq_line(col = "black") +  
  facet_wrap(~ metro_f, labeller = "label_both") +  
  scale_color_viridis_d(option = "C", begin = 0.4, end = 0.7) +  
  theme(aspect.ratio = 1) +  
  labs(y = "Exercise Opportunity Access",  
       x = "Expectations Under Normal Model") +  
  guides(col = "none")  
  
p2 <- ggplot(midwest05, aes(x = metro_f, y = ex_access)) +  
  geom_violin() +  
  geom_boxplot(aes(fill = metro_f), notch = TRUE, width = 0.3) +
```

```

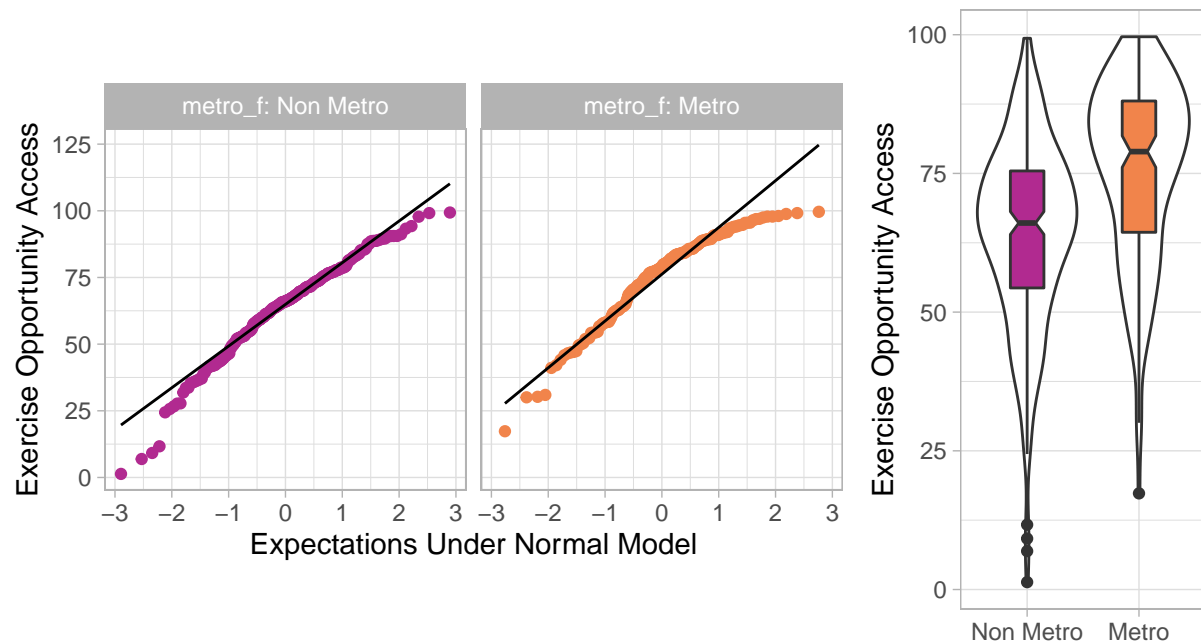
scale_fill_viridis_d(option = "C", begin = 0.4, end = 0.7) +
guides(fill = "none") +
labs(y = "Exercise Opportunity Access",
     x = "")

p1 + p2 + plot_layout(widths = c(3,1)) +
  plot_annotation(title = "Comparing Exercise Opportunity Access",
                 subtitle = "Rural and Urban Counties in 5 States (OH, IN, IL, MI, WI)")

```

## Comparing Exercise Opportunity Access

Rural and Urban Counties in 5 States (OH, IN, IL, MI, WI)



We can also use `group_by` and `summarize` to obtain useful numeric summaries.

```

midwest05 %>%
  group_by(metro_f) %>%
  summarize(n = n(), mean = mean(ex_access),
            median = median(ex_access),
            sd = sd(ex_access), variance = var(ex_access)) %>%
  kable(digits = 3)

```

metro_f	n	mean	median	sd	variance
Non Metro	263	64.052	66.06	16.723	279.673
Metro	174	75.724	78.93	16.197	262.346

Based on these figures, and our quick numeric summary, there is not much of a case to be made to treat these as being well-described by a Normal distribution and model. We see some modest left skew in each group.

- A bootstrap is likely to be a stronger choice for comparing means here than an approach based on a t-distribution.

- A rank-based approach is problematic because it's hard to assume symmetry, and because we want to be comparing means, ideally.

#### 4.4.1 Grading Rubric

- If a student has produced visualization and appropriately discussed that these distributions are certainly not Normal and likely not symmetrical they should receive 10 points.
  - If the student arrives at these conclusions, but does not include Normal Q-Q plots, they should lose 1 point and this should be noted.
- The remaining 5 points will be given if a student has appropriately decided that a bootstrap is likely the best approach for comparing means.
  - If a student states a t-based procedure is best, they should lose all 5 points and this should be explicitly noted in their feedback.

### 4.5 Question 5 (10 points)

Produce an appropriate 95% confidence interval for a relevant population mean that addresses the key question from the study. Be sure to show and describe the R code that led to your selected confidence interval, and describe how your responses to prior Questions led you to select this approach. Save your interpretation of the results for Question 6.

As stated, we will use the bootstrap approach, which can be done using `bootdif()` which comes from `Love-boost.R`. As noted, we've set our seed to 4312021.

These results suggest a point estimate for the (metro - non-metro) mean difference in exercise access percentage of 11.67 percentage points with a 95% confidence interval of (8.57, 14.82).

```
set.seed(4312021)
```

```
bootdif(midwest05$ex_access, midwest05$metro_f, conf.level = 0.95)
```

Mean Difference	0.025	0.975
11.672560	8.572354	14.824930

#### 4.5.1 What if we had calculated Non Metro - Metro instead?

The mean difference will just be the inverse and the 95% CI will be nearly identical (again, inverse) as well.

```
set.seed(4312021)
```

```
bootdif(midwest05$ex_access, fct_relevel(midwest05$metro_f, "Metro", "Non Metro"),
        conf.level = 0.95)
```

Mean Difference	0.025	0.975
-11.672560	-14.826146	-8.521418

#### 4.5.2 What if I used a different B.rep?

The default number of bootstrap samples in the `bootdif()` function is 2,000. However, it's possible that some of you may have set the number to something else. Generally, we don't want to do less than 1,000 without some compelling reason.

If we were instead to do 5,000 we'd get:

```
set.seed(4312021)
```

```
bootdif(midwest05$ex_access, midwest05$metro_f, conf.level = 0.95, B.reps = 5000)
```



Mean Difference	0.025	0.975
11.672560	8.530718	14.816487

And now 10,000

```
set.seed(4312021)

bootdif(midwest05$ex_access, midwest05$metro_f, conf.level = 0.95, B.reps = 10000)
```

Mean Difference	0.025	0.975
11.672560	8.454167	14.755090

And finally if we did 1000

```
set.seed(4312021)

bootdif(midwest05$ex_access, midwest05$metro_f, conf.level = 0.95, B.reps = 1000)
```

Mean Difference	0.025	0.975
11.672560	8.568322	14.777382

### 4.5.3 Comments

With such large differences in sample size and sample variance between metropolitan and non-metropolitan counties, if forced to do a t test, I would be thinking about the Welch t, instead of a pooled t test.

```
mosaic::favstats(ex_access ~ metro_f, data = midwest05)
```

	metro_f	min	Q1	median	Q3	max	mean	sd	n	missing
1	Non Metro	1.33	54.355	66.06	75.445	99.36	64.05152	16.72342	263	0
2	Metro	17.33	64.385	78.93	88.055	99.63	75.72408	16.19710	174	0

```
midwest05 %>%
  t.test(ex_access ~ metro_f, conf.level = 0.95)
```

Welch Two Sample t-test

```
data: ex_access by metro_f
t = -7.2795, df = 378.7, p-value = 1.951e-12
alternative hypothesis: true difference in means between group Non Metro and group Metro is not equal to 0
95 percent confidence interval:
 -14.825388 -8.519731
sample estimates:
mean in group Non Metro      mean in group Metro
      64.05152              75.72408
```

Again, a t-based procedure is simply not as accurate as the bootstrap here. However, we have included it just to show what it gives us. Here's the pooled t test, as well.

```
model2 <- lm(ex_access ~ metro_f, data = midwest05)

tidy(model2, conf.int = TRUE, conf.level = 0.95) %>%
  select(term, estimate, conf.low, conf.high) %>%
  kable(digits = 3)
```

term	estimate	conf.low	conf.high
(Intercept)	64.052	62.05	66.053
metro_fMetro	11.673	8.50	14.845

#### 4.5.4 Grading Rubric

- If a student has successfully run the bootstrap, they should receive all 10 points.
  - **Note:** if the student chose a different number of B.reps, their answer may slightly differ. This is **OK** and no points should be deducted assuming the mean difference is the same and the answer is reasonable.
- If, in the question above, the student concluded the t-test was appropriate and they properly ran the t-test, they should receive 5 points.
  - If, however, the t-test is also not correct the student should only receive between 0 and 5 points.

#### 4.6 Question 6 (10 points)

Interpret your confidence interval from Question 5 in the context of the request by the project's principal investigator using complete English sentences.

The two-sided 95% confidence interval would have a lower bound of 8.6 percent and an upper bound of 14.8 percent for the population mean percentage of adults who have access to physical activity opportunities difference comparing rural to urban counties. Specifically we are estimating  $\mu_{Metro} - \mu_{Non-Metro}$  here, and we know this from the listed Mean Difference.

This means, essentially, that we have 95% confidence in the interval which places our population mean difference in `access_to_exercise_opportunities` between 8.6% and 14.8%.

##### 4.6.1 Grading Rubric

- If the student appropriately states the meaning of the confidence interval that they generated in Question 5, then they should receive all 10 points on Question 6.
- If their interpretation of their CI is incorrect, the student should lose 5-10 points, depending on how serious the problem is.

## 5 Part B. An Observational Study (Questions 7-9, 40 points)

The `lab05_lind.Rds` dataset provided on our 431-data page comes from an observational study of 996 patients receiving an initial Percutaneous Coronary Intervention (PCI) at Ohio Heart Health, Christ Hospital, Cincinnati in 1997 and followed for at least 6 months by the staff of the Lindner Center.

The 698 patients thought to be more severely diseased were assigned to treatment with **abciximab** (an expensive, high-molecular-weight IIb/IIIa cascade blocker); while the remaining 298 patients received **usual care** with their initial PCI. Additional information on the lindner data set is available here <sup>1,2</sup>

<sup>1</sup> Rdocumentation. (n.d.). lindner: Lindner Center Data On 996 PCI Patients Analyzed By Kereiakes Et Al. (2000). Retrieved from <https://www.rdocumentation.org/packages/MatchLinReg/versions/0.7.0/topics/lindner>

<sup>2</sup> Kereiakes DJ, Obenchain RL, Barber BL, et al. Abciximab provides cost effective survival advantage in high volume interventional practice. Am Heart J 2000; 140: 603-610.

### 5.1 Question 7 (10 points)

Ingest the `lab05_lind.Rds` data into R, and use them to develop an appropriate comparison of the relative risk of an `acutemi` for those receiving abciximab compared to those receiving usual care. Be sure to provide your code, and interpret your results in context in one or two English sentences. Use a 90% confidence level.

A couple of hints for Question 7:

1. You should be changing the variable type and labels to make the results more interpretable (perhaps with `fct_recode()`), as well as change the levels so we are obtaining the probability or odds of a myocardial infarction for those who received abciximab compared to those who received usual care in a contingency table with abciximab status in the rows and acute MI status in the columns.
2. An appropriate contingency table will have the value for subjects who have an acute MI and who are receiving abciximab in the top left, and that cell should contain between 100 and 150 subjects.

First, we'll start by reading our data into R and making some adjustments to our variables of interest here.

```
lab05_lind <- readRDS("data/lab05_lind.Rds")

lab05_lind <- lab05_lind %>%
  mutate(diabetic_f = fct_recode(factor(diabetic),
                                   "No Diabetes" = "0",
                                   "Diabetes" = "1"),
         acutemi_f = fct_recode(factor(acutemi),
                                   "No Acute MI" = "0",
                                   "Acute MI" = "1"),
         acutemi_f2 = fct_relevel(acutemi_f,
                                   "Acute MI", "No Acute MI"),
         diabetic_f2 = fct_relevel(diabetic_f,
                                   "Diabetes", "No Diabetes"),
         abcix_f = fct_recode(factor(abcix),
                                   "Usual Care" = "0",
                                   "Abciximab" = "1"),
         abcix_f2 = fct_relevel(abcix_f,
                                   "Abciximab", "Usual Care"))
```

Now, it's helpful to show our 2 x 2 table. We see there were 125 acute MIs among 573 people with receiving abciximab, and 18 among 180 among those without diabetes. We can also include row percentages, using the code below.

```
lab05_lind %>%
  tabyl(abcix_f2, acutemi_f2) %>%
  adorn_totals("row") %>%
  adorn_percentages("row") %>%
  adorn_pct_formatting() %>%
  adorn_ns(position = "front") %>%
  kable()
```

abcix_f2	Acute MI	No Acute MI
Abciximab	125 (17.9%)	573 (82.1%)
Usual Care	18 (6.0%)	280 (94.0%)
Total	143 (14.4%)	853 (85.6%)

So, in our data, we see 17.9% of those on abciximab having an acute MI compared to just 6.0% of those receiving usual care.

Finally, we can obtain our estimate of relative risk, by pushing the relevant data into the Epi package's `twoby2` function, using a 90% confidence interval, as requested.

```
lab05_lind %$% twoby2(abcix_f2, acutemi_f2,
  conf.level = 0.90)
```

2 by 2 table analysis:

```
-----
Outcome      : Acute MI
Comparing    : Abciximab vs. Usual Care

      Acute MI No Acute MI    P(Acute MI) 90% conf. interval
Abciximab      125      573      0.1791    0.1564    0.2042
Usual Care      18      280      0.0604    0.0413    0.0875

                                90% conf. interval
      Relative Risk: 2.9648    1.9899    4.4174
      Sample Odds Ratio: 3.3934    2.2038    5.2253
Conditional MLE Odds Ratio: 3.3900    2.1647    5.4945
      Probability difference: 0.1187    0.0835    0.1501

      Exact P-value: 0.0000
      Asymptotic P-value: 0.0000
-----
```

We obtain a relative risk of 2.96 with a 90% confidence interval of (1.99, 4.42). Patients who received abciximab, compared to those who received usual care, had 2.96 (95% CI: 1.99, 4.42) times the risk of an acute myocardial infarction.

- If you'd mistakenly ran a 95% CI, you'd get: (1.84, 4.77) instead.

### 5.1.1 On a Bayesian Augmentation

We didn't ask for a Bayesian augmentation here, but you might have decided to include one any way. This might look like this:

```
table_aug <- lab05_lind %>% table(abcix_f2, acutemi_f2) + 2
```

```
table_aug
```

```
      acutemi_f2
abcix_f2  Acute MI No Acute MI
  Abciximab      127      575
  Usual Care       20      282
```

```
twoby2(table_aug, conf.level = 0.90)
```

2 by 2 table analysis:

```
-----
Outcome      : Acute MI
Comparing    : Abciximab vs. Usual Care
```

	Acute MI	No Acute MI	P(Acute MI)	90% conf. interval
Abciximab	127	575	0.1809	0.1582 0.206
Usual Care	20	282	0.0662	0.0462 0.094

	90% conf. interval
Relative Risk: 2.7318	1.8697 3.9913
Sample Odds Ratio: 3.1143	2.0598 4.7085
Conditional MLE Odds Ratio: 3.1113	2.0246 4.9267
Probability difference: 0.1147	0.0790 0.1467

```
Exact P-value: 0.0000
Asymptotic P-value: 0.0000
-----
```

and the resulting relative risk has point estimate 2.73 and 90% CI (1.87, 3.99).

- If you'd mistakenly ran a 95% CI while including this augmentation, you'd get: (1.74, 4.29) instead.

### 5.1.2 Grading Rubric

- If the student has successfully set up the contingency table, calculate, and interpreted the relative risk they should receive all 10 points.
  - If the student has correctly built the contingency table but misinterpreted the relative risk they should score 8 of 10 points.
- If the contingency table is incorrect, but the calculation and interpretation are consistent with their results they should receive up to 7 of 10 points.
- If the contingency table is incorrect, and the calculation or interpretation is incorrect they should receive no more than 5 of 10 points.
- No penalty or bonus should be given for using or failing to use the Bayesian augmentation.- If their results are consistent with a 95% confidence level instead of 90%, they should lose 3 points.

## 5.2 Question 8 (10 points)

Now develop an appropriate comparison of the difference in probability of an acute MI for those who have diabetes as compared to those who do not have diabetes. Again, use a 90% confidence level for this Question, provide all necessary code, and interpret your result in context using one or two English sentences.

Hint for Question 8: Make sure the top left cell of your contingency table includes subjects who have an acute MI and also have diabetes, with diabetes status in the rows of your table, thus following standard epidemiological format.

Since we've already set up our variables, we simply can make another contingency table and again run the `twoby2` function. We see that 13.0% ( $n = 29$ ) of our patients with diabetes had an acute MI compared to 14.7% ( $n = 114$ ) of our patients without diabetes.

```
lab05_lind %>%
  tabyl(diabetic_f2, acutemi_f2) %>%
  adorn_totals("row") %>%
  adorn_percentages("row") %>%
  adorn_pct_formatting() %>%
  adorn_ns(position = "front") %>%
  kable()
```

diabetic_f2	Acute MI	No Acute MI
Diabetes	29 (13.0%)	194 (87.0%)
No Diabetes	114 (14.7%)	659 (85.3%)
Total	143 (14.4%)	853 (85.6%)

The diabetes - no diabetes difference in probability of having an acute MI is estimated to be 0.0174 with a 90% confidence interval of (-0.0569, 0.0286).

```
lab05_lind %$% twoby2(diabetic_f2, acutemi_f2,
  conf.level = 0.90)
```

2 by 2 table analysis:

Outcome : Acute MI

Comparing : Diabetes vs. No Diabetes

	Acute MI	No Acute MI	P(Acute MI)	90% conf. interval
Diabetes	29	194	0.1300	0.0973 0.1718
No Diabetes	114	659	0.1475	0.1277 0.1697

		90% conf. interval
Relative Risk:	0.8818	0.6413 1.2124
Sample Odds Ratio:	0.8641	0.5984 1.2479
Conditional MLE Odds Ratio:	0.8643	0.5791 1.2687
Probability difference:	-0.0174	-0.0569 0.0286

Exact P-value: 0.5880

Asymptotic P-value: 0.5134

- The result without a Bayesian augmentation, but (incorrectly) using a 95% CI is (-0.0639, 0.0381).
- The result with a Bayesian (+2) augmentation is a probability difference of -0.0127 and a 90% CI of (-0.0527, 0.0335). If we'd mistakenly used a 95% CI, it would be (-0.0598, 0.0431).

### 5.2.1 Grading Rubric

- If the student has successfully setup the contingency table, calculate, and interpreted the relative risk they should receive all 10 points.
  - If the student has correctly built the contingency table but misinterpreted the relative risk they should score 8 of 10 points.
- If the contingency table is incorrect, but the calculation and interpretation are consistent with their results they should receive 7 of 10 points.
- If the contingency table is incorrect, and the calculation or interpretation is incorrect they should receive no higher than 5 of 10 points.
- No penalty or bonus should be given for using or failing to use the Bayesian augmentation.
- If their results are consistent with a 95% confidence level instead of 90%, they should lose 3 points.

### 5.3 Question 9 (20 points)

Suppose that in a new test sample of 495 patients receiving an initial PCI (like those described in the Lindner Center data) that we obtain the following results for a model we have developed to predict six-month survival using information available at baseline.

- 405 were predicted to survive at least 6 months, and actually survived at least 6 months.
- 74 were predicted not to survive at least 6 months, but did actually survive at least 6 months.
- 9 were predicted not to survive at least 6 months and did not actually survive at least 6 months.

Specify the appropriate cross-tabulation for predicted and actual survival to 6 months, and then calculate and interpret the accuracy, sensitivity and specificity for the model described here.

Hint: I expect that a close reading of Chapter 6 in Spiegelhalter will be helpful here.

We can start by setting up our two by two table.

- The two groups we need to create in the rows are the predictions from the model.
  - Predicted to survive 6 months or more, which I'll abbreviate as (Pred\_6mos)
  - Predicted to survive less than 6 months, (Pred\_shorter)
- The two groups we need to create in the columns are the actual survival states.
  - Actually survived 6 months or more (Actual\_6mos)
  - Actually survived less than 6 months (Actual\_shorter)

So the cross-tabulation we want to fill in should look like this:

	Actual_6mos	Actual_shorter	Total
Pred_6mos	A	B	A+B
Pred_shorter	C	D	C+D
Total	A+C	B+D	A+B+C+D

and we need to work our way through the provided summaries to identify A, B, C and D.

- We are told that there are 495 patients in the new sample, so  $A+B+C+D = 495$ .
- We are told that 405 patients were predicted to survive 6 months and actually did, so  $A = 405$ .
- We are told that 74 were predicted to survive  $< 6$  months, but actually survived 6 months or more, so  $C = 74$
- We are told 9 were predicted to survive less than 6 months, and actually survived less than 6 months, so  $D = 9$

So far, we have...

	Actual_6mos	Actual_shorter	Total
Pred_6mos	405	B	405+B
Pred_shorter	74	9	83
Total	479	B+9	495

That's enough to fill in the whole table, as we now know that  $B = 7$ , since  $A + C + D = 405 + 74 + 9 = 488$ , which is 7 less than 495.

```
q9 <- matrix(c(405, 7, 74, 9), byrow = TRUE, nrow = 2)
rownames(q9) <- c("Pred 6 mos. or more", "Pred less than 6 mos.")
colnames(q9) <- c("Survived 6 mos. or more", "Survived less than 6 mos.")
addmargins(q9)
```

	Survived 6 mos. or more	Survived less than 6 mos.	Sum
Pred 6 mos. or more	405	7	412
Pred less than 6 mos.	74	9	83
Sum	479	16	495

Now, we can calculate our accuracy, sensitivity, and specificity.

- **Accuracy** is the number of correct predictions divided by the total number of patients. Here, we have  $405 + 9 = 414$  correct predictions, and a total of 495 patients in the sample. So the accuracy is 0.8364. This just means that 83.64% of the predictions were accurate.
- **Sensitivity** describes the 479 patients who actually survived at least 6 months, and we're calculating the proportion of those patients who were correctly classified, so this is  $405/479$ , and the sensitivity is 0.8455. The sensitivity tells us that 84.55% of those who survived six months or more were correctly predicted to fall into that category by the model.
- **Specificity** describes the 16 patients who actually did not survive at least 6 months, and we're calculating the proportion of those patients who were correctly classified, so this is  $9/16$ , and the specificity is 0.5625. The specificity tells us that 56.25% of those who failed to survive at least six months were correctly predicted to fall into that category by the model.

### 5.3.1 With a Bayesian augmentation

We didn't ask for a Bayesian augmentation, but you might have done it.

```
q9_aug <- matrix(c(405+2, 7+2, 74+2, 9+2), byrow = TRUE, nrow = 2)
rownames(q9_aug) <- c("Pred 6 mos. or more", "Pred less than 6 mos.")
colnames(q9_aug) <- c("Survived 6 mos. or more", "Survived less than 6 mos.")
addmargins(q9_aug)
```

	Survived 6 mos. or more	Survived less than 6 mos.	Sum
Pred 6 mos. or more	407	9	416
Pred less than 6 mos.	76	11	87
Sum	483	20	503

- So under the Bayesian augmentation, we have:
  - Accuracy =  $(11 + 407)/503 = 0.831$ , or 83.1%.
  - Sensitivity =  $407/483 = 0.8427$  or 84.27%.
  - Specificity =  $11/20 = 0.55$  or 55%.



### 5.3.2 Grading Rubric

- 11 points will be awarded for a correctly set up (as shown above) contingency table.
  - Variations from this, but that are still correctly aligned, should be awarded 7 points.
  - Incorrect contingency tables should be awarded at most 5 points.
- 9 points will be awarded for a correct accuracy, sensitivity, and specificity.
  - Each of these is worth 3 points, with 1 being for the calculation and 2 being for a correct interpretation.

### 5.3.3 Update as of 2021-10-27

In our grading, we deducted points where the contingency table was not set up correctly. However, some students who lost points had instead followed the Spiegelhalter setup which reverses the rows and columns from what we show above. This is still correct. As such we re-reviewed the assignments and awarded points back to a number of students where this was the case. The Spiegelhalter way would look something like this:

```
q9_alt <- matrix(c(9, 7, 74, 405), byrow = TRUE, nrow = 2)
rownames(q9_alt) <- c("Did not survive", "Survived")
colnames(q9_alt) <- c("Predicted not to survive", "Predicted to survive")
addmargins(q9_alt)
```

	Predicted not to survive	Predicted to survive	Sum
Did not survive	9	7	16
Survived	74	405	479
Sum	83	412	495