431 Lab 02 Sketch and Grading Rubric

Instructor: Dr. Thomas E. Love Lab Author: Mr. Wyatt P. Bensken

Due: 2021-09-20 | Last Edit: 2021-08-13 08:35:35

Contents

1 Learning Objectives

1. Be comfortable using R to import and manage data.
2. Become familiar with the `tidyverse` packages and their functions
3. Be able to build and interpret a figure using R.
4. Use figures to contextualize specific data points of interest.

2 Packages and Functions

In Lab 02 we were hoping you would become more familiar with the following packages and functions:

Packages: `tidyverse`

Functions:

- `%>%` (pipe)
- `filter()`
- `count()`
- `select()`
- `ggplot()`

3 Setup

```
library(tidyverse)
```

4 Obtaining the Data for Lab 02

For this Lab, we have prepared a CSV (comma-separated version) file which contains a small subset of data from the 2021 County Health Rankings. The County Health Rankings data provide some useful information on how the health of US residents is affected by where they live, and we will use data from these Rankings several times this semester.

You can find the CSV file for Lab 02, called `lab02_counties.csv` in our class Data and Code repository. This data file contains 3,142 rows (each row is a county) and 5 variables including:

Variable Name	Description
`state`	Two-letter postal abbreviation of the state name
`county_name`	Name of the county
`metro`	Whether or not the county is in a metropolitan area
`some_college`	Percentage of county residents who have completed some college
`female_pct`	Percentage of county residents who are female

Note that the `some_college` estimates come from American Community Survey 5-year estimates from 2015-19, and the `female_pct` estimates come from Census Population Estimates published in 2019.

4.1 Import the `lab02_counties.csv` data

```
lab02_data <- read_csv("data/lab02_counties.csv")
```

```
-- Column specification --------------------------------------------------------
cols(
  state = col_character(),
  county_name = col_character(),
  metro = col_double(),
  some_college = col_double(),
  female_pct = col_double()
)
```

5 Question 1 (10 points)

> Write a piece of R code that filters the observations (counties) in the data set to only the following midwest states: Ohio (OH), Indiana (IN), Illinois (IL), Michigan (MI), and Wisconsin (WI). Specifically, take our `lab02_data` and create `midwest_data` which contains counties in these states. Hint: the pipe `%>%` and `filter` function should be a large part of your code. **The rest of the assignment will use this smaller set of counties.**

Here we create a smaller dataset, which we call "midwest_data", from our full data ("lab02_data") and use a pipe and filter command to only select those states which we have listed. It is important to use the `|` operator to say "or".

```
midwest_data <- lab02_data %>%
    filter(state == "OH" | state == "IN" | state == "IL" |
             state == "MI" | state == "WI")
```

5.1 Grading Rubric

- If a student was able to successfully filter, into a new dataset, the 5 states of interest then they will receive all 10 points.
- Any errors in this step will likely result in the loss of all 10 points, but that will be at the discrection of the TA grading the assignment.

6 Question 2 (10 points)

Write a piece of R code that counts the number of observations (counties) in our midwestern states data that you created in Question 1, within each of the five states in which we are interested. Hint: The `count` function and the pipe `%>%` should be a big part of your code.

```
midwest_data %>%
    count(state)
```

```
# A tibble: 5 x 2
  state     n
  <chr> <int>
1 IL      102
2 IN       92
3 MI       83
4 OH       88
5 WI       72
```

So, for example, there are **88** counties from the state of Ohio in the data set, and the states represented also include Illinois, Indiana, Michigan and Wisconsin.

6.1 Comments

1. It's important to realize that each row in the data set represents a single county.

- You can verify this by looking at the data:
  - perhaps with the command `View(midwest_data)`. This will bring up a spreadsheet of the data in a new R Studio window.
  - **or** perhaps by typing the name of the data set `midwest_data` into the Console in R Studio to get a listing.

The listing of the data set (which is a tibble) displays the first few variables for the first ten rows of the data set, like this.

```
midwest_data
```

```
# A tibble: 437 x 5
   state county_name      metro some_college female_pct
   <chr> <chr>            <dbl>        <dbl>      <dbl>
 1 IL    Adams County         0         66.5       50.7
 2 IL    Alexander County     1         41.6       51.4
 3 IL    Bond County          1         64.8       48.1
 4 IL    Boone County         1         57.6       50.0
 5 IL    Brown County         0         45.9       35.7
 6 IL    Bureau County        0         59.3       51.0
 7 IL    Calhoun County       1         66.7       49.7
 8 IL    Carroll County       0         60.4       50.2
 9 IL    Cass County          0         46.7       49.9
10 IL    Champaign County     1         79.1       50.3
# ... with 427 more rows
```

3. To obtain the number of counties (rows) in each state, I need only to obtain a count of the number of rows associated with each state. That's the command I used above, with `midwest_data %>% count(state)`.

4. There are some other things I used R to count in this case, for example:

   - To count the number of rows (counties) in the data as a whole, I used the command `nrow(midwest_data)`. The result is 437.

    - If I’d wanted to count the number of columns (variables) I’d have used `ncol(midwest_data)`. The result is 5.
    - I can get a count of both the rows and the columns by either listing the tibble with `midwest_data` or by capturing its dimensions (the size of the rectangle of data) with:

```
dim(midwest_data)
```

```
[1] 437   5
```

5. I did some (slightly) more sophisticated counting to understand:
    - The number of (different) states in the data, using `n_distinct(midwest_data %>% select(state))`, which yields 5.
    - The number of counties in the state of Ohio, using `nrow(midwest_data %>% filter(state == "OH"))`, which yields 88.

6.2 Grading Rubric

- If a student has specified the 5 states and the number of counties by state, then they should receive all 10 points.
- If a student has done any of the following then they should lose 5 points:
    - Reported only the number of states
    - Reported the total number of counties (instead of counties per state)
    - Done this analysis on the entire dataset, rather than our midwestern counties from Question 1.
- If there are problems with the number of reported counties, please help identify why that may be and deduct 5 points.

7 Question 3 (10 points)

Use the `filter()` and `select()` functions in R to obtain a result which specifies the `some_college` and `metro` status of Cuyahoga County in the state of Ohio.

Here's the approach we used

```
midwest_data %>%
    filter(state == "OH") %>%
    filter(county_name == "Cuyahoga County")
```

```
# A tibble: 1 x 5
  state county_name     metro some_college female_pct
  <chr> <chr>           <dbl>        <dbl>      <dbl>
1 OH    Cuyahoga County     1         69.4       52.3
```

This displays the tibble, but restricted to the county of Cuyahoga in the state of OH (Ohio).

So we conclude that 69.38% of the adult residents of Cuyahoga County have completed a college degree, and that Cuyahoga County is identified as being part of a metropolitan area.

7.1 Comments

1. Did we need the `filter(state == "OH")` line in our code? What happens if we leave this out?

```
midwest_data %>%
#    filter(state == "OH") %>%
    filter(county_name == "Cuyahoga County")
```

```
# A tibble: 1 x 5
  state county_name     metro some_college female_pct
  <chr> <chr>           <dbl>        <dbl>      <dbl>
1 OH    Cuyahoga County     1         69.4       52.3
```

Looks like there is only one county in the data set with the name Cuyahoga County, so we're OK.

2. What if we tried this approach (not specifying the state) with Adams county?

```
midwest_data %>%
  filter(county_name == "Adams County")
```

```
# A tibble: 4 x 5
  state county_name  metro some_college female_pct
  <chr> <chr>        <dbl>        <dbl>      <dbl>
1 IL    Adams County     0         66.5       50.7
2 IN    Adams County     0         51.9       50.0
3 OH    Adams County     0         43.1       50.4
4 WI    Adams County     0         48.2       46.8
```

3. How many unique county names are seen in the 437 counties in the `midwest_data` data?

```
n_distinct(midwest_data %>% select(county_name))
```

```
[1] 320
```

7.2 Grading Rubric

- If a student successfully identifies Cuyahoga County's `some_college` and `metro` status, using the filter and select tools, they should receive all 10 points.
- If a student reports the number but provides no code or output they should lose all 10 points.

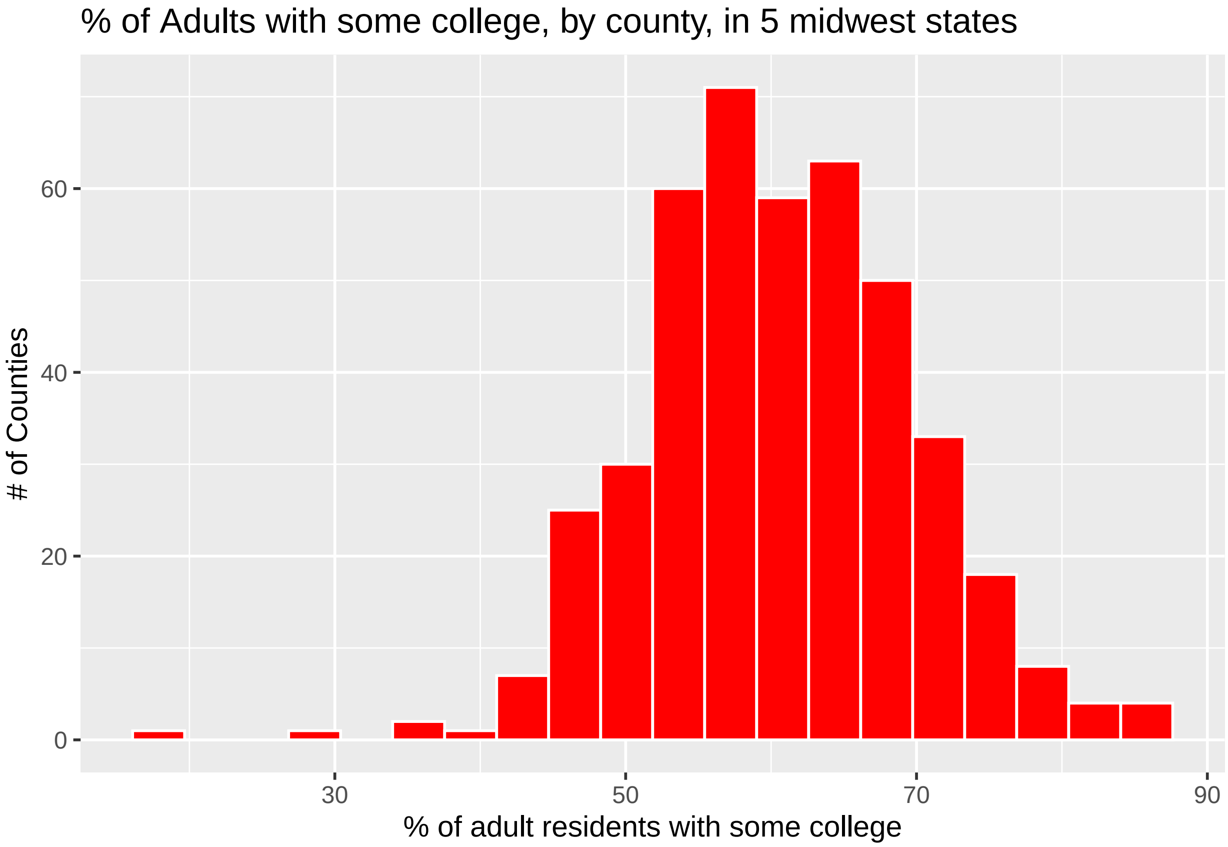
- Other variations of using filter and select should result in 2-point deduction.

8 Question 4 (15 points)

> Use the tools we've been learning in the ggplot2 package to build a histogram of the some_college results across all of the midwest counties represented in the data subset you created in Question 1. Create appropriate (that is to say, meaningful) titles for each axis and for the graph as a whole (don't simply use the default choices.) We encourage you to use something you find more attractive than the default gray fill in the histogram.

Here is a simple and reasonable histogram for the some_college data, mostly using the template but filling in appropriate axis labels and a title, and using a red fill for the bars. 20 bins seems to work pretty well here.

```
ggplot(midwest_data, aes(x = some_college)) +
    geom_histogram(bins = 20, fill = "red", col = "white") +
    labs(title = "% of Adults with some college, by county, in 5 midwest states",
         y = "# of Counties",
         x = "% of adult residents with some college")
```



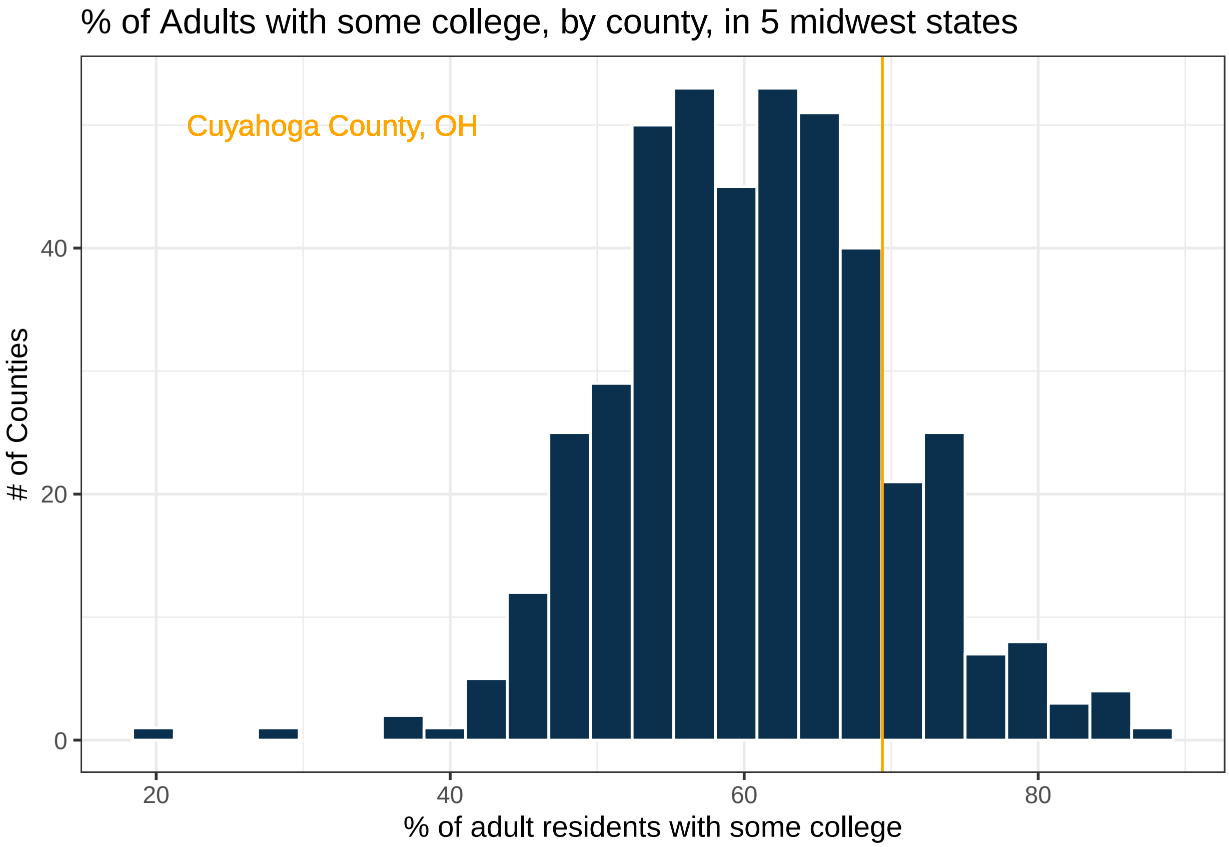
8.1 Comments

The setup below uses theme_bw() to specify a revised theme, reduces the number of bins a bit, and creates the fill with the official blue color of CWRU[1]. Anticipating question 4, I also added a red vertical line to the plot showing the value of Cuyahoga County, Ohio, and a text annotation to indicate what the line means.

```
ggplot(midwest_data, aes(x = some_college)) +
    geom_histogram(bins = 25, fill = "#0a304e", col = "white") +
    geom_vline(xintercept = 69.4, col = "orange") +
    geom_text(x = 32, y = 50, col = "orange", label = "Cuyahoga County, OH") +
```

[1] CWRU's color guide is available at https://case.edu/umc/our-brand/visual-guidelines/color.

```
    theme_bw() +
    labs(title = "% of Adults with some college, by county, in 5 midwest states",
         y = "# of Counties",
         x = "% of adult residents with some college")
```



8.2 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, they should receive all 15 points.
- If any of the following occur the student should lose 5 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 10 points and the TA should provide *very* detailed comments.

9 Question 5 (10 points)

> Based on your results in Questions 3 and 4, write a short description (2-3 sentences) of Cuyahoga County's position relative to the full distribution of counties in terms of `some_college`.

Cuyahoga County's `some_college` rate was 69.4%, which looks to be above the median level for the counties included in the data. There are certainly more counties with rates below Cuyahoga's than above it.

9.1 Comments

We could, if we like, be more precise, and perhaps identify the **ranking** of Cuyahoga County within the data set. We know there are 437 counties in all. How many have a *higher* value of `some_college` than Cuyahoga County?

```
midwest_data %>% count(some_college > 69.4)
```

```
# A tibble: 2 x 2
  `some_college > 69.4`     n
  <lgl>                 <int>
1 FALSE                   369
2 TRUE                     68
```

68 of the 437 counties rate above Cuyahoga on this measure, so that's 15.6% of those midwest counties.

9.2 Grading Rubric

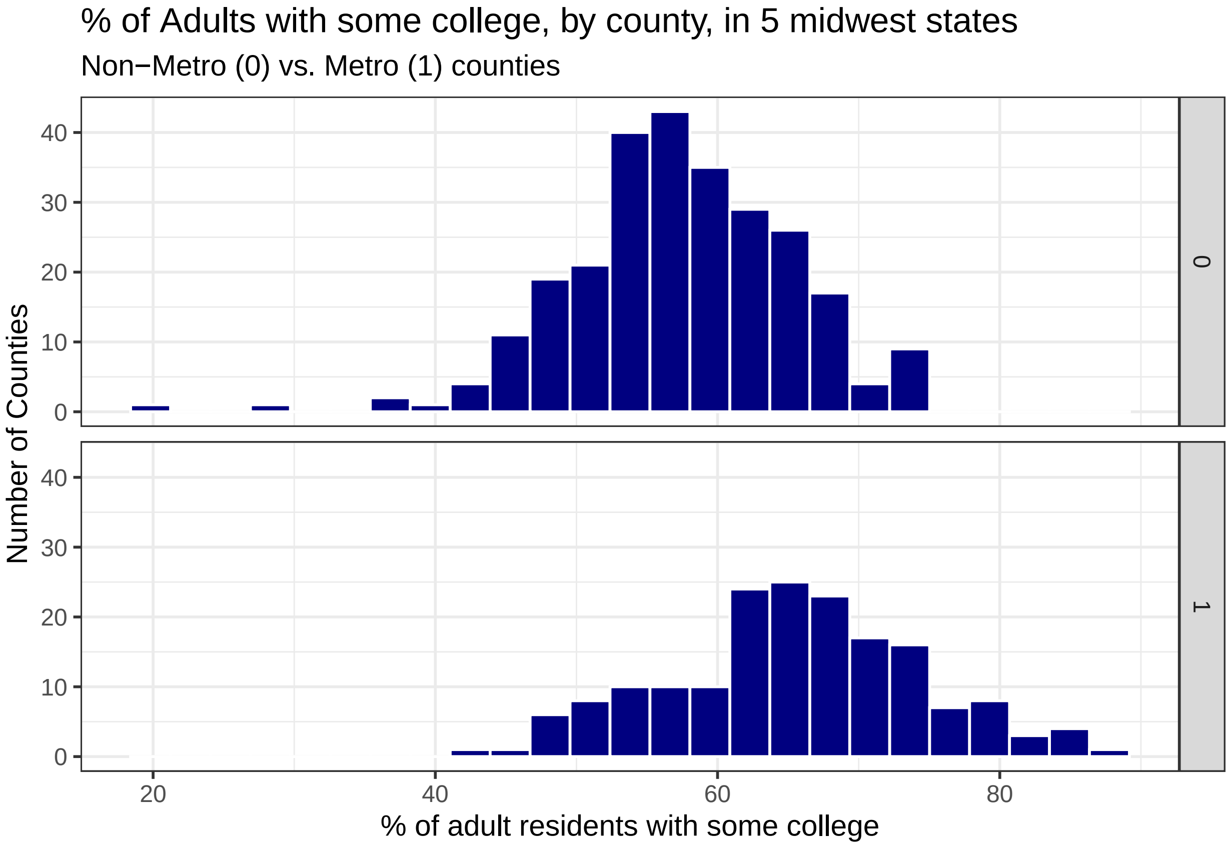
- If a student successfully states that Cuyahoga county is on the upper-end of the distribution, they should receive all 10 points. The numeric summaries presented here are not necessary for full credit.
- If a student does not successfully conclude this, but is on the right track they should lose 5 points.
- If a student is not on the right track and does not properly interpret the location of Cuyahoga county, they should lose all 10 points.

10 Question 6 (15 points)

Use ggplot2 to build a single plot (a pair of histograms after faceting would be one approach, or perhaps a comparison boxplot) which nicely compares the some_college distribution for counties within metropolitan areas to counties outside of metropolitan areas. Again, make an effort to build and incorporate useful titles and labels so that the resulting plot stands on its own, rather than just accepting all of the defaults that appear.

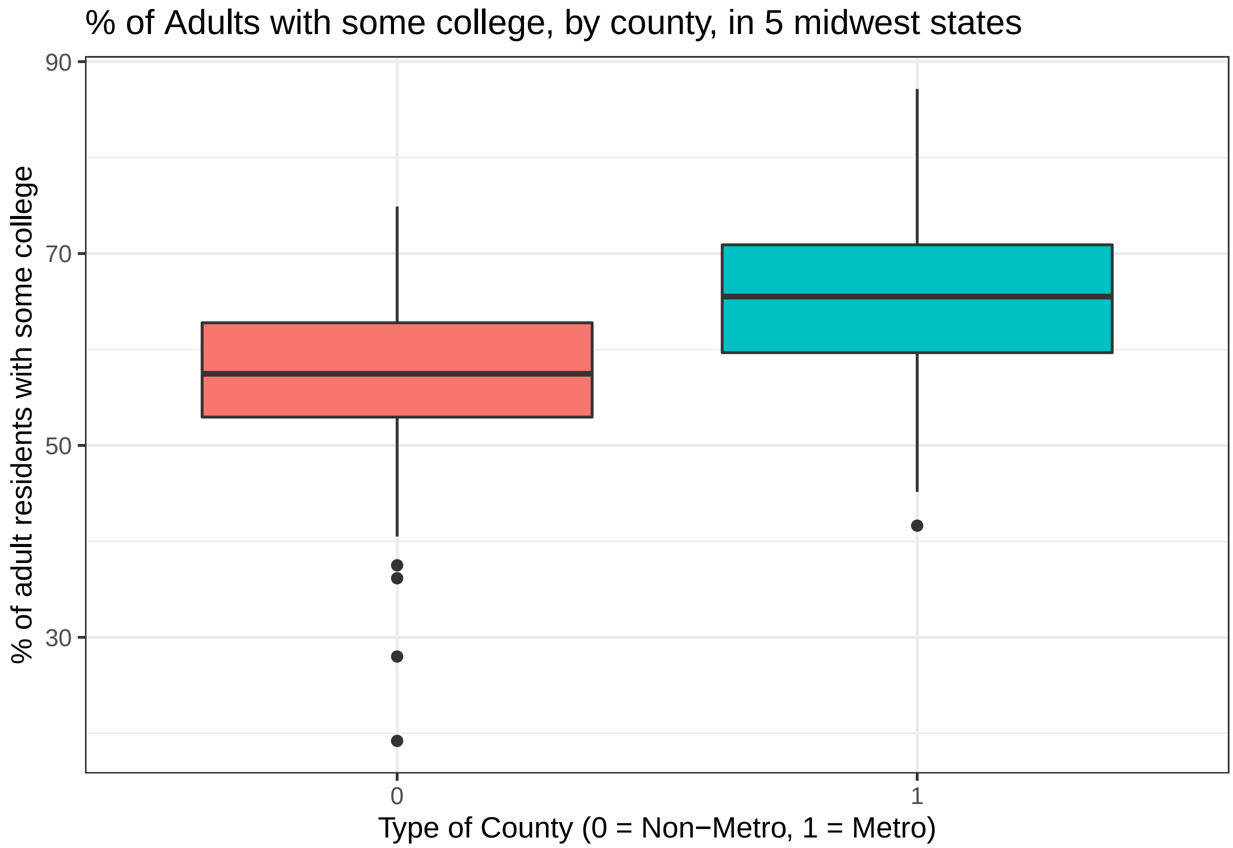
Here is a reasonable result.

```
ggplot(midwest_data, aes(x = some_college)) +
    geom_histogram(bins = 25, col = "white", fill = "navy") +
    facet_grid(metro ~ .) +
    theme_bw() +
    labs(x = "% of adult residents with some college",
         y = "Number of Counties",
         title = "% of Adults with some college, by county, in 5 midwest states",
         subtitle = "Non-Metro (0) vs. Metro (1) counties")
```



Another common approach, that shows a bit less of the data, would be a comparison boxplot. Note that it's important to get R to treat the 0-1 information in metro as categorical here, and we do this by telling R to see it as a **factor**.

```
ggplot(midwest_data, aes(x = factor(metro), y = some_college,
                         fill = factor(metro))) +
    geom_boxplot() +
    guides(fill = "none") +
    theme_bw() +
    labs(y = "% of adult residents with some college",
         x = "Type of County (0 = Non-Metro, 1 = Metro)",
         title = "% of Adults with some college, by county, in 5 midwest states")
```

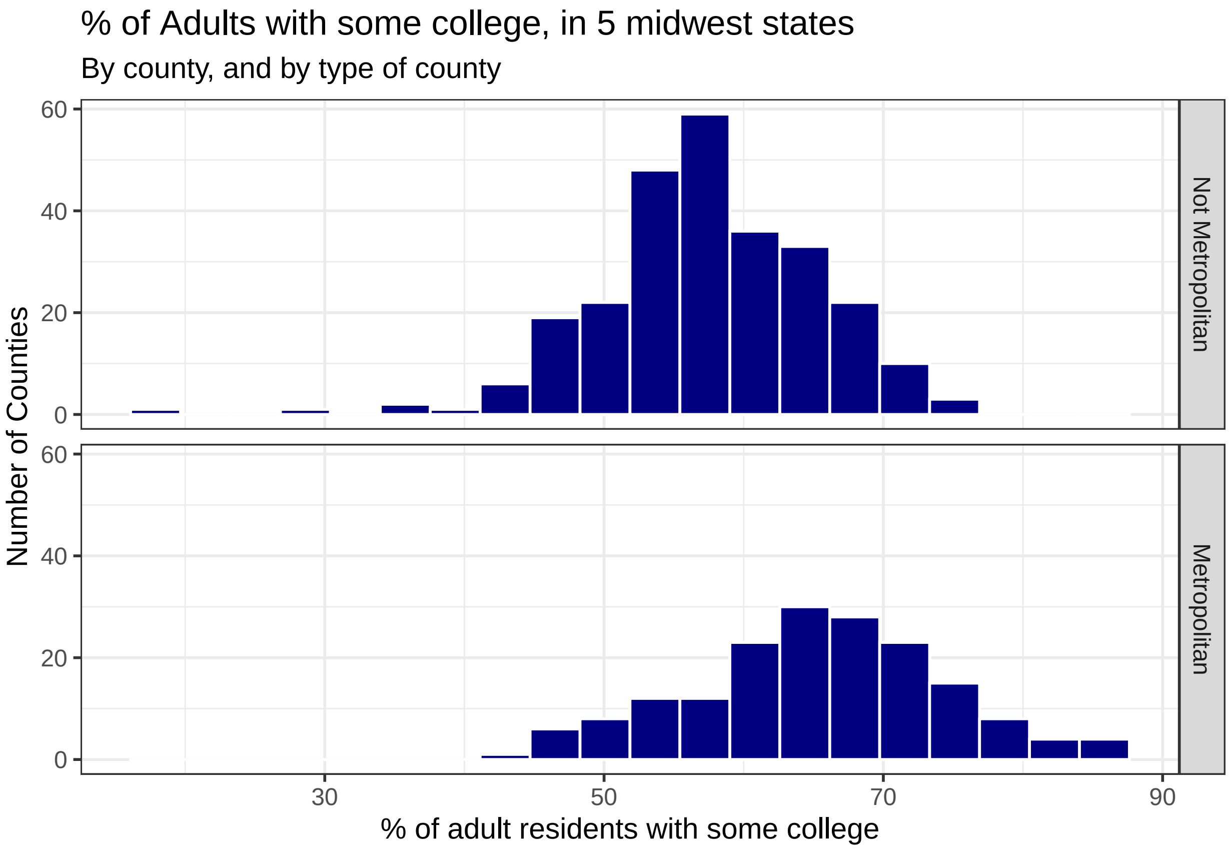


10.1 Comments

I think this version of the faceted set of histograms plot is slightly better. What has changed?

```
midwest_data2 <- midwest_data %>%
  mutate(inmetro_f = fct_recode(factor(midwest_data$metro),
                                "Metropolitan" = "1", "Not Metropolitan" = "0"))

ggplot(midwest_data2, aes(x = some_college)) +
    geom_histogram(bins = 20, col = "white", fill = "navy") +
    facet_grid(inmetro_f ~ .) +
    theme_bw() +
    labs(x = "% of adult residents with some college",
         y = "Number of Counties",
         title = "% of Adults with some college, in 5 midwest states",
         subtitle = "By county, and by type of county")
```

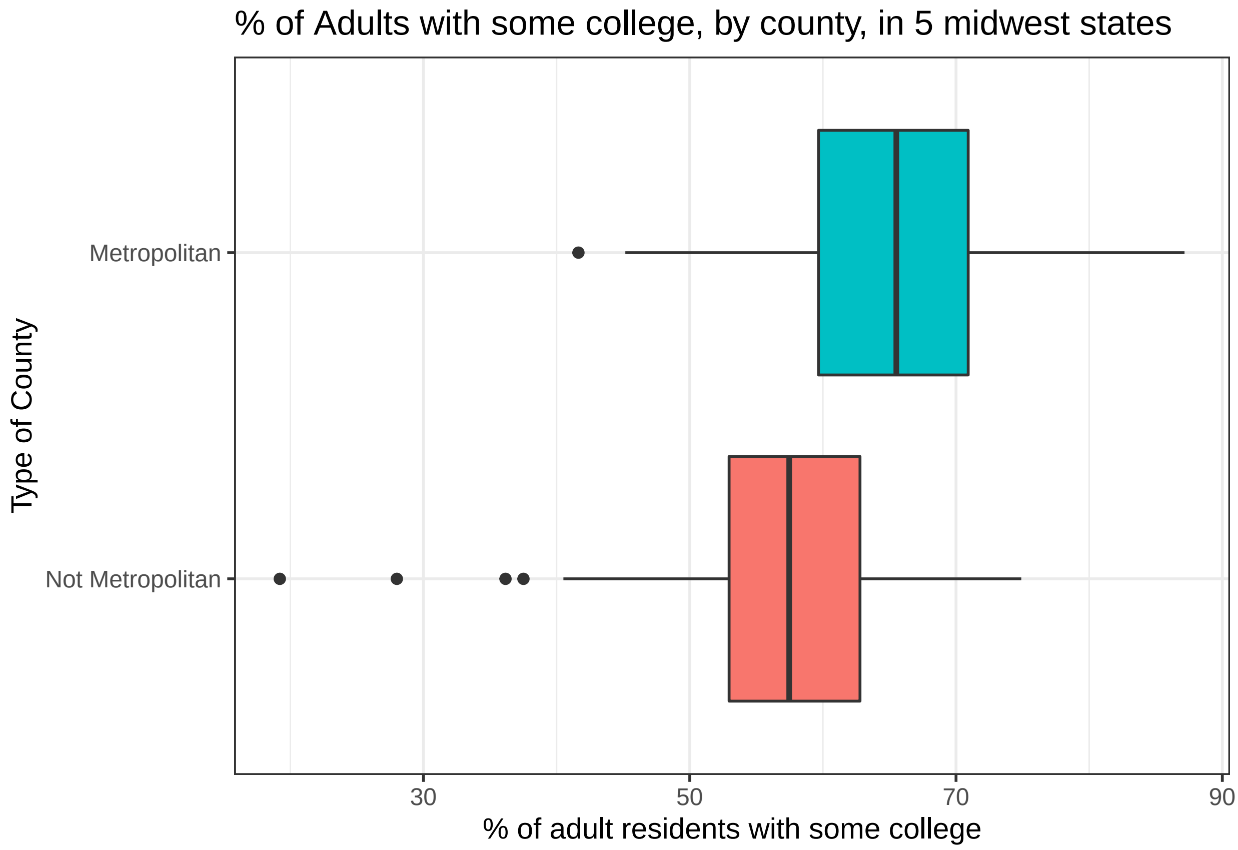


By pre-specifying the `metro` variable as a factor, and giving it meaningful names for its two levels (Metropolitan and Non-Metropolitan, instead of 1 and 0) we improve the plot.

Could we do the same sort of thing to improve the comparison boxplot? And what if we wanted a horizontal boxplot instead of a vertical one?

Sure. Note the use of `coord_flip()` to flip the Y and X coordinates and axes.

```
ggplot(midwest_data2, aes(x = inmetro_f, y = some_college,
                    fill = inmetro_f)) +
    geom_boxplot() +
    guides(fill = "none") +
    theme_bw() +
    coord_flip() +
    labs(y = "% of adult residents with some college",
         x = "Type of County",
         title = "% of Adults with some college, by county, in 5 midwest states")
```



10.2 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, they should receive all 15 points.
- If any of the following occur the student should lose 5 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 10 points and the TA should provide *very* detailed comments.

11 Question 7 (10 points)

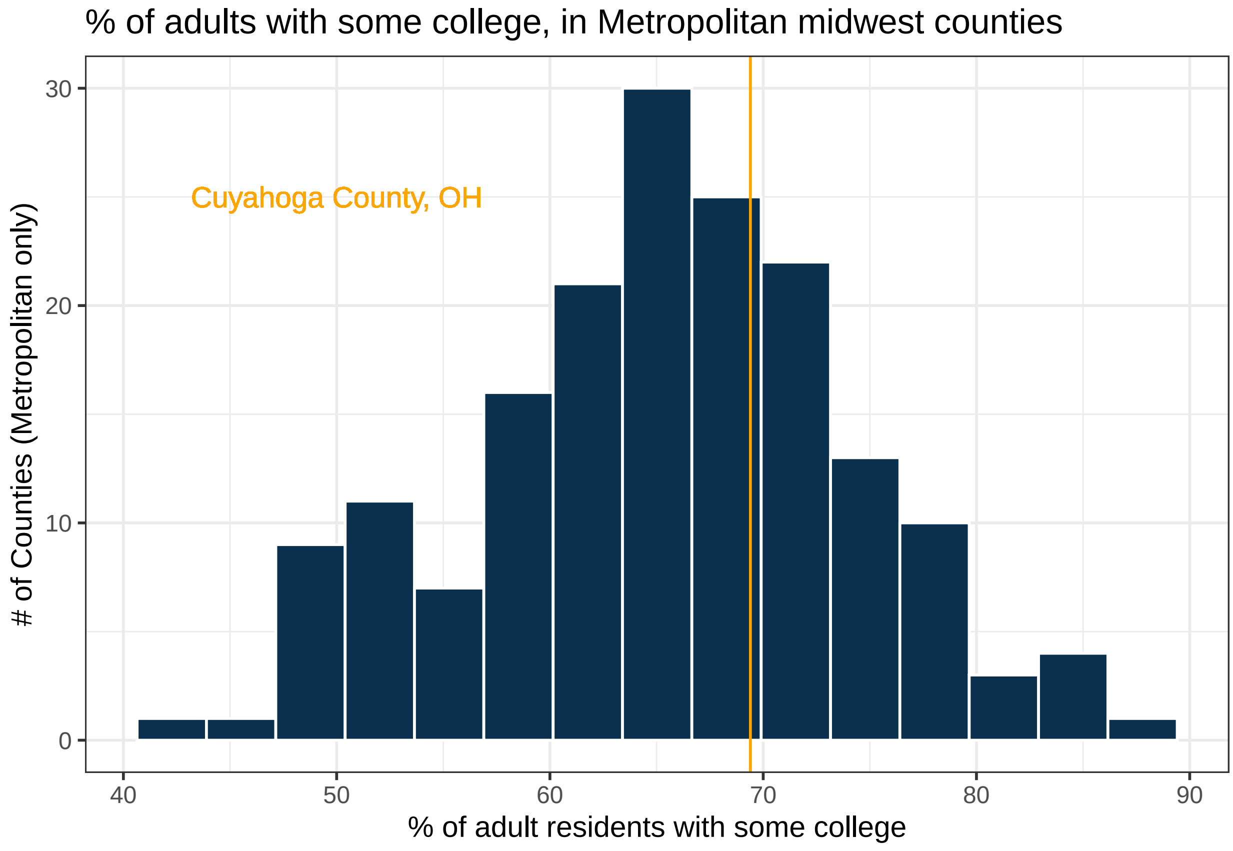
> Write a short description of where Cuyahoga County falls within the plot you built in Question 6. Specifically, comment on the position of Cuyahoga County in terms of `some_college` relative to the other counties within its `metro` category. Two sentences should be sufficient here.

Within the metropolitan counties, Cuyahoga's position is still above the median county, but a bit closer to the center of the group on `some_college` than it was when we looked at the whole data set.

11.1 Comments

The plot below is restricted to those counties in the "Metropolitan" group.

```r
ggplot(filter(midwest_data, metro == 1), aes(x = some_college)) +
    geom_histogram(bins = 15, fill = "#0a304e", col = "white") +
    geom_vline(xintercept = 69.4, col = "orange") +
    geom_text(x = 50, y = 25, col = "orange", label = "Cuyahoga County, OH") +
    theme_bw() +
    labs(title = "% of adults with some college, in Metropolitan midwest counties",
         y = "# of Counties (Metropolitan only)",
         x = "% of adult residents with some college")
```



It turns out that Cuyahoga County ranks 56th among the 174 metropolitan counties.

```r
# number of Metropolitan counties
nrow(midwest_data %>% filter(metro == 1))
```

```
[1] 174
```

```
# number with some_college higher than Cuyahoga
midwest_data %>% filter(metro == 1) %>%
    count(some_college > 69.4)
```

```
# A tibble: 2 x 2
  `some_college > 69.4`     n
  <lgl>                 <int>
1 FALSE                   119
2 TRUE                     55
```

55 of the 174 metropolitan counties in these five states rate above Cuyahoga County on this measure, so that's 31.6%. Compare this to Cuyahoga's ranking behind only 15.6% of all counties (metropolitan and non-Metropolitan) in those same five states.

11.2 Grading Rubric

- If a student successfully states that Cuyahoga county is still higher, but a bit closer to the center among metropolitan counties, they should receive all 10 points. The numeric summaries presented here are not necessary for full credit.
- If a student does not successfully conclude this shift once the figures are stratified, but is on the right track they should lose 5 points.
- If a student is not on the right track and does not properly interpret the location of Cuyahoga county with respect to other metropolitan counties, they should lose all 10 points.

12 Question 8 (20 points)

> By now, we'd like you to have read through Chapter 3 of David Spiegelhalter's *The Art of Statistics*. In the above questions we, broadly, examined the relationship between county metropolitan status and the percent of residents who have completed some college. In our first step, we limited to just counties in 5 midwestern states. Reflecting on Chapter 3 of *The Art of Statistics*, please write a brief essay (100-150 words) that dicusses the process of inductive inference and how that influences the conclusions we can draw from our work in this assignment. As always, use complete and clear English sentences in your essay.

We don't write sketches for essay questions.

12.1 Grading Rubric

For essay questions, each student starts with 20 full points, and deductions should be noted in the TA notes of the grading sheet.

A 20-point answer will include, broadly: discussion of the data, how our sample was a non-random sample, thoughts regarding our study population, a comment on external validity, and a comment on generalizability.

- If a student covers most of these topics then a score of 20 points is appropriate.
- If a student discusses a smaller number of these topics, but in greater depth, than a score of 19 or 20 is appropriate.
- If a student fails to discuss these topics but relates to Spiegelhalter's Figure 3.1, they should score 15 to 17.
- If a student does not discuss these issues, but other population or measurement concepts they should score 12 to 15.
- If a student does not relate their response to the previous questions in this homework, and also does not discuss any of these concepts, they should score 10 to 12.
- If no essay is provided a student should receive 0 points and "No Essay" should be noted on the grading sheet.

13 Session Information

```
sessionInfo()
```

```
R version 4.1.0 (2021-05-18)
Platform: x86_64-apple-darwin17.0 (64-bit)
Running under: macOS Big Sur 10.16

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRblas.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRlapack.dylib

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] forcats_0.5.1   stringr_1.4.0   dplyr_1.0.7     purrr_0.3.4
[5] readr_1.4.0     tidyr_1.1.3     tibble_3.1.2    ggplot2_3.3.5
[9] tidyverse_1.3.1

loaded via a namespace (and not attached):
 [1] tidyselect_1.1.1  xfun_0.24         haven_2.4.1       colorspace_2.0-2
 [5] vctrs_0.3.8       generics_0.1.0    htmltools_0.5.1.1 yaml_2.2.1
 [9] utf8_1.2.1        rlang_0.4.11      pillar_1.6.1      glue_1.4.2
[13] withr_2.4.2       DBI_1.1.1         dbplyr_2.1.1      modelr_0.1.8
[17] readxl_1.3.1      lifecycle_1.0.0   munsell_0.5.0     gtable_0.3.0
[21] cellranger_1.1.0  rvest_1.0.0       evaluate_0.14     labeling_0.4.2
[25] knitr_1.33        fansi_0.5.0       highr_0.9         broom_0.7.8
[29] Rcpp_1.0.6        scales_1.1.1      backports_1.2.1   jsonlite_1.7.2
[33] farver_2.1.0      fs_1.5.0          hms_1.1.0         digest_0.6.27
[37] stringi_1.6.2     grid_4.1.0        cli_3.0.0         tools_4.1.0
[41] magrittr_2.0.1    crayon_1.4.1      pkgconfig_2.0.3   ellipsis_0.3.2
[45] xml2_1.3.2        reprex_2.0.0      lubridate_1.7.10  assertthat_0.2.1
[49] rmarkdown_2.9     httr_1.4.2        rstudioapi_0.13   R6_2.5.0
[53] compiler_4.1.0
```