Data Visualization

Dominguez Center for Data Science Workshop

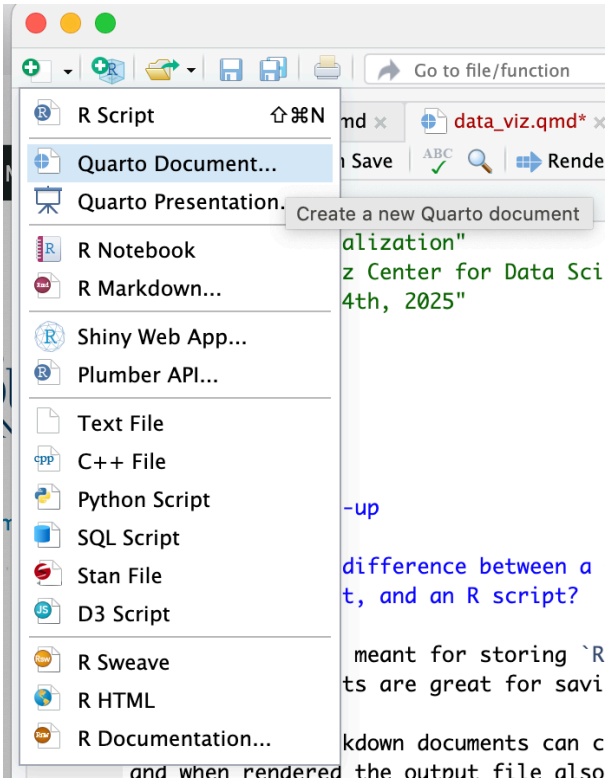
Invalid Date

Questions Round-up

What is the difference between a Quarto document, an RMarkdown document, and an R script?

- `R` scripts are meant for storing `R` code (and maybe comments). Scripts are great for saving code.
- Quarto and RMarkdown documents can contain `R` code and text and when rendered the output file also include output from the code (e.g., tables, graphs). Quarto and RMarkdown are great for archiving all your work.
- Quarto is the successor to RMarkdown.

How do I create a new Quarto document?



Do spaces and indentation matter for R code?

Not for how the code is run. But spaces and indentation will be very important for readability.

```
x<-5
x+2
```

```
[1] 7
```

```
x <- 5
x + 2
```

```
[1] 7
```

Should I analyze my data on Posit Cloud?

If it is a large or sensitive dataset, then I'd recommend downloading R and RStudio and running everything locally. You can find directions here. Let me know if you need help with this!

My code has a bug!

We are going to do some live-coding today but `R` is finicky so be prepared for s. Check out "data_viz_key.qmd" for the correct code or raise your hand for help.

Load Packages

The packages we need for our explorations today (`readr` for reading in data and `ggplot2` for graphing data) are part of a popular suite of packages called the `tidyverse`.

```
library(tidyverse)
library(ggrepel)
```

Data Background

In 2013, the government decided to make data about colleges more accessible so that students and parents could more easily compare schools. These data are called the "College Scorecard" data and the 2024 dataset contains 3,305 variables on 6,484 universities in the US!

I have filtered that 2024 dataset to only include schools which confer majority baccalaureate degrees and where the majority of those degrees are in the arts and sciences based on the Carnegie Classification system. In other words, I filtered the data down to the schools which are "similar" to Bucknell (including Bucknell itself) and picked out some variables for us to explore.

Let's use these data to start exploring and visualizing with `R`!

Data Dictionary

Below are the code names and descriptions of the variables in our dataset.

- `UNITID`: Unique identifier
- `INSTNM`: Name of institution
- `CITY`: City
- `STABBR`: State
- `HIGHDEG`: Highest degree awarded (0 = Non-degree grants, 1 = Certificate degree, 2 = Associate degree, 3 = Bachelor's degree, 4 = Graduate degree)

- `PREDDEG`: Predominant undergraduate degree awarded (0 = Not classified, 1 = Predominantly certificate-degree granting, 2 = Predominantly associate's-degree granting, 3 = Predominantly bachelor's-degree granting, 4 = Entirely graduate-degree granting)
- `CONTROL`: Ownership (1 = Public, 2 = Private non-profit, 3 = Private for-profit)
- `HBCU`: Flag for Historically Black College and University
- `TUITFTE`: Net tuition revenue per full-time equivalent student
- `AVGFACSAL`: Average faculty salary
- `ADM_RATE`: Admission rate
- `SATVR75`: 75th percentile of SAT scores at the institution (critical reading)
- `SATMT75`: 75th percentile of SAT scores at the institution (math)
- `ACTCM75`: 75th percentile of the ACT cumulative score
- `COSTT4_A`: The average annual total cost of attendance, including tuition and fees, books and supplies, and living expenses for all full-time, first-time, degree/certificate-seeking undergraduates who receive Title IV aid.
- `NPT4_PRIV`: The average annual total cost of attendance (CostT4_A, CostT4_P), including tuition and fees, books and supplies, and living expenses, minus the average grant/scholarship aid
- `UGDS`: Enrollment of undergraduate certificate/degree-seeking students
- `UG25ABV`: Percentage of undergraduates aged 25 and above
- `PCTFLOAN_DCS`: Percentage of degree/certificate-seeking undergraduate students awarded a federal loan
- `PCTPELL_DCS`: Percentage of degree/certificate-seeking undergraduate students awarded a Pell Grant
- `DEBT_MDN`: The median original amount of the loan principal upon entering repayment
- `C100_4`: Completion rate for first-time, full-time students at four-year institutions (100% of expected time to completion)
- `RET_FT4`: First-time, full-time student retention rate at four-year institutions
- `MD_EARN_WNE_5YR`: Median earnings of graduates working and not enrolled 5 years after completing
- `Location`: Whether or not the institution is located in Pennslyvania or not

Load the Data

Run the following code to load and inspect the data.

```
# Load the data
colleges <- read_csv("data/ccbasic21.csv") %>%
  mutate(Location = if_else(STABBR == "PA", "PA", "Not PA"))

# Inspect the data
glimpse(colleges)
```

```
Rows: 220
Columns: 25
$ UNITID          <dbl> 100937, 101912, 106342, 107080, 107512, 112260, 115409~
$ INSTNM          <chr> "Birmingham-Southern College", "Oakwood University", "~
$ CITY            <chr> "Birmingham", "Huntsville", "Batesville", "Conway", "A~
$ STABBR          <chr> "AL", "AL", "AR", "AR", "AR", "CA", "CA", "CA", "CA", ~
$ HIGHDEG         <dbl> 3, 4, 3, 4, 4, 4, 3, 4, 3, 3, 3, 4, 4, 3, 3, 4, 4, 4, ~
$ PREDDEG         <dbl> 3, 3, 3, 3, 3, 3, 3, 1, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, ~
$ CONTROL         <dbl> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, ~
$ HBCU            <dbl> 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
$ TUITFTE         <dbl> 10340, 12279, 10188, 9685, 10749, 35957, 36122, 26331,~
$ AVGFACSAL       <dbl> 7029, 4842, 5817, 7889, 6735, 15333, 14478, 11309, 117~
$ ADM_RATE        <dbl> 0.5717, 0.6805, 0.5984, 0.6028, 0.7232, 0.1035, 0.1336~
$ SATVR75         <dbl> 670, NA, NA, 680, 610, 760, 770, NA, 750, NA, 770, 760~
$ SATMT75         <dbl> 610, NA, NA, 648, 590, 790, 790, NA, 760, NA, 790, 750~
$ ACTCM75         <dbl> 29, NA, NA, 30, 28, 35, 36, NA, 34, NA, 35, 34, NA, 32~
$ COSTT4_A        <dbl> 35495, 38377, 44749, 49928, 43878, 78723, 82236, NA, 7~
$ NPT4_PRIV       <dbl> 19723, 19686, 25183, 22780, 23086, 19489, 39671, NA, 3~
$ UGDS            <dbl> 968, 1378, 489, 1127, 1587, 1383, 906, 15, 1935, 1212,~
$ UG25ABV         <dbl> 0.0170, 0.1284, 0.0276, 0.0054, 0.0140, 0.0021, 0.0011~
$ PCTFLOAN_DCS    <dbl> 0.6452, 0.6477, 0.5934, 0.4483, 0.6109, 0.1627, 0.3646~
$ PCTPELL_DCS     <dbl> 0.2277, 0.4906, 0.3702, 0.2543, 0.2486, 0.2008, 0.1293~
$ DEBT_MDN        <chr> "16000", "21500", "10699", "19500", "15000", "11948", ~
$ C100_4          <dbl> 0.5854, 0.3351, 0.3085, 0.6743, 0.6174, 0.8318, 0.8826~
$ RET_FT4         <dbl> 0.7746, 0.7706, 0.5072, 0.7905, 0.7897, 0.9579, 0.9733~
$ MD_EARN_WNE_5YR <dbl> 56625, 51429, 45744, 49579, 48168, 108186, 154095, 418~
$ Location        <chr> "Not PA", "Not PA", "Not PA", "Not PA", "Not PA", "Not~
```

- How many institutions are in the dataset?
- How many variables are in in the dataset?
- Verify that Bucknell is in our dataset. (Hint: Click on the dataset name in the Environment tab to pull up a snapshot of the data.)

First `ggplot`!

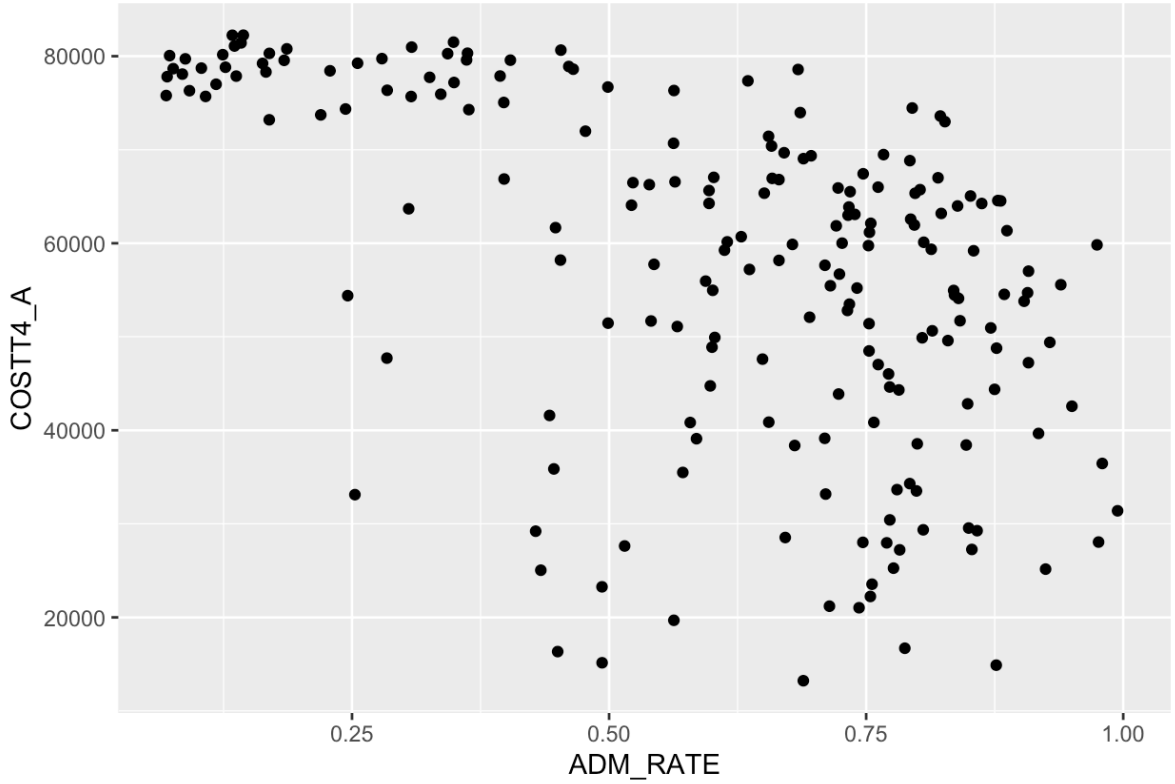
Let's now learn to create graphs with the package `ggplot2`.

Guiding Principle: We will map variables from the **data** to the **aes**thetic attributes (e.g. location, size, shape, color) of **geom**etric objects (e.g. points, lines, bars).

```
ggplot(data = ---, mapping = aes(---)) +
  geom_---(---)
```

Scatterplot

Let's walk through how to re-create the following graph that compares the admission rate (`ADM_RATE`) and total yearly cost (`COSTT4_A`).



To fill in the code below, we should ask ourselves the following questions:

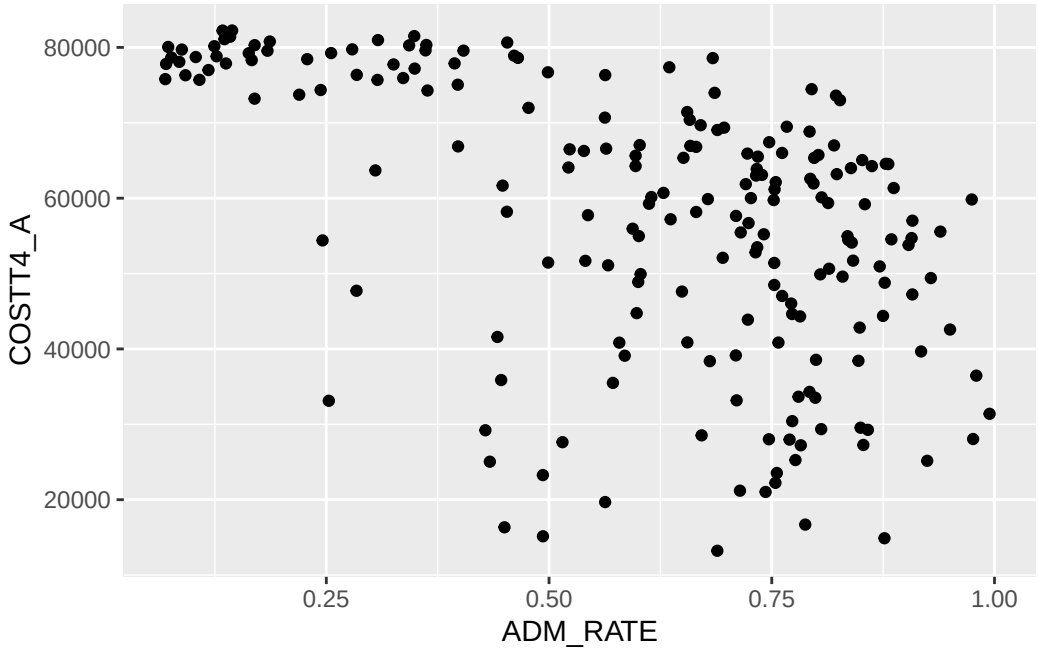
- What is the name of the dataset?
- What is the geom (shape)?
- What are the variables?

- How should we map the variables to the geom?

```
ggplot(data = ---, mapping = aes(---)) +
  geom_---(---)
```

Make sure to remove the `#| eval: false` before rendering the document so that the graph appears in the output.

```
ggplot(data = colleges,
       mapping = aes(x = ADM_RATE,
                     y = COSTT4_A)) +
  geom_point()
```



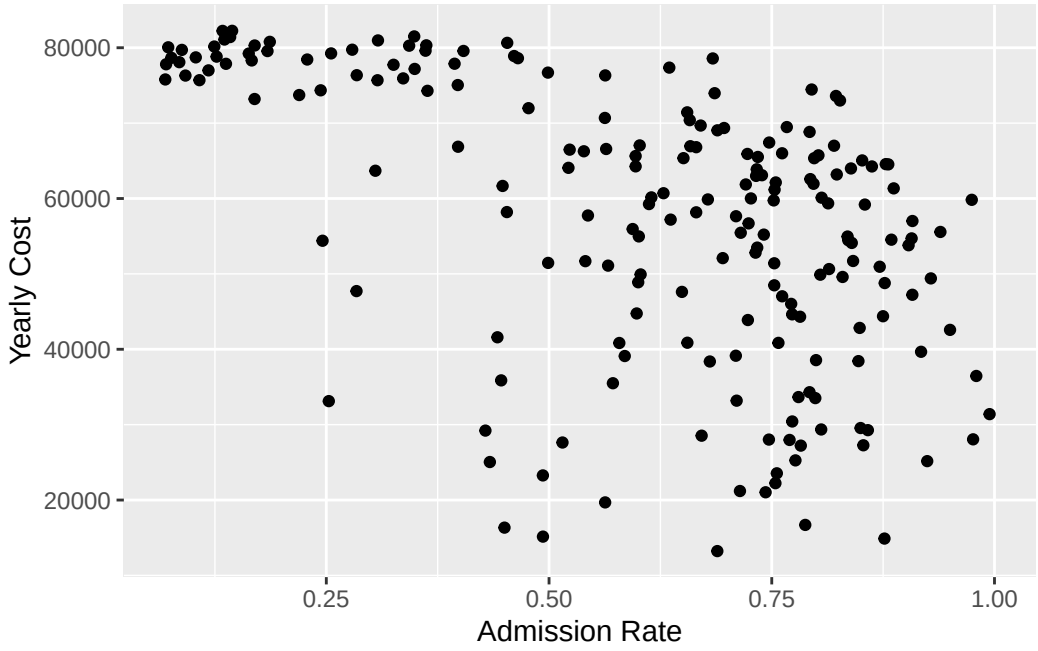
Describe the relationship between these two variables.

Labels!

We can use the `labs()` layer to add context and fix our axis labels.

```
ggplot(data = ---, mapping = aes(---)) +
  geom_---(---) +
  labs(---)
```

```
ggplot(data = colleges,
       mapping = aes(x = ADM_RATE,
                     y = COSTT4_A)) +
  geom_point() +
  labs(x = "Admission Rate", y = "Yearly Cost")
```

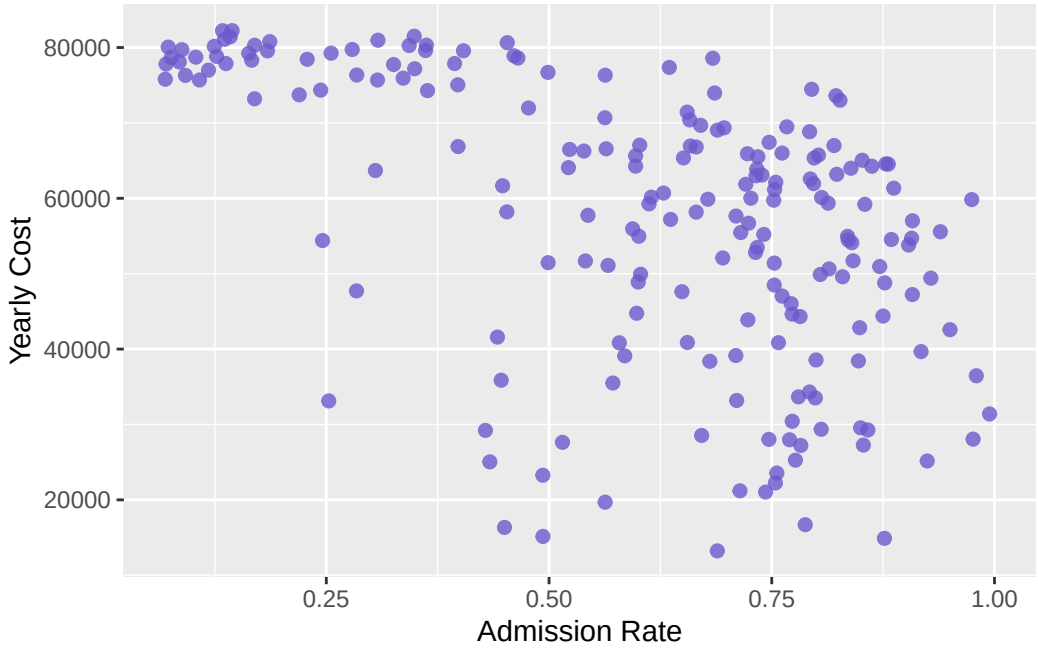


Setting versus Mapping

We can change the look of our graph by **setting** the value of some of the aesthetics. Let's learn how to do the following:

- Make the points somewhat transparent.
- Make the points bigger.
- Color the points by your favorite color.

```
ggplot(data = colleges,
       mapping = aes(x = ADM_RATE,
                     y = COSTT4_A)) +
  geom_point(alpha = 0.8, size = 2, color = "slateblue") +
  labs(x = "Admission Rate", y = "Yearly Cost")
```

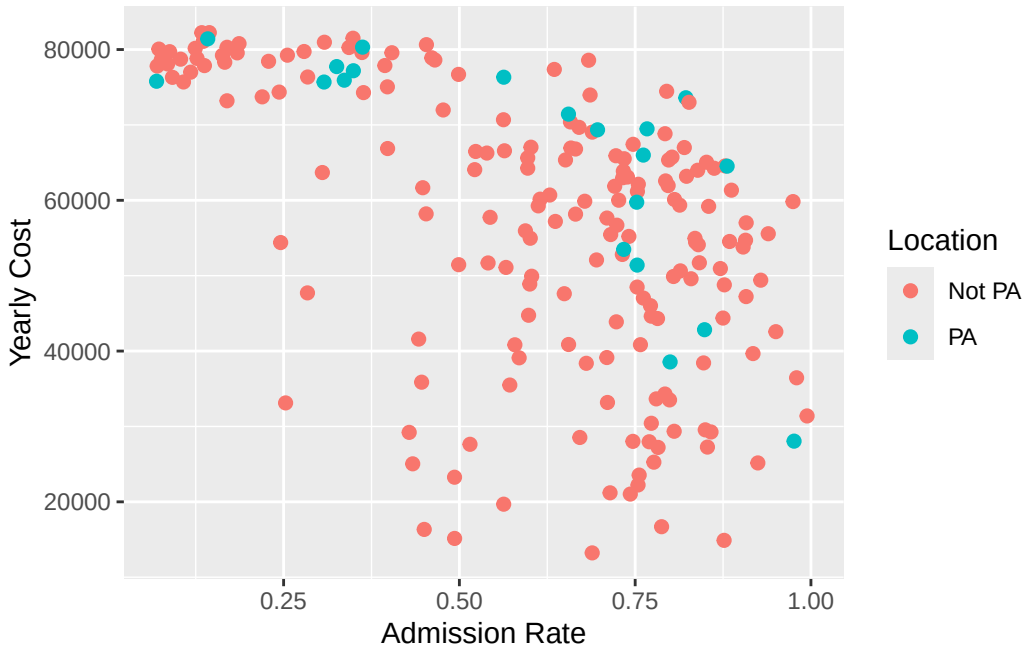


This is called **setting** because we applied the same treatment to all points (e.g., now all of the points are your favorite color).

We would say that we **mapped** `ADM_RATE` to the x-axis and `COSTT4_A` to the y-axis because we used a variable in the dataset to determine the value of these aesthetics.

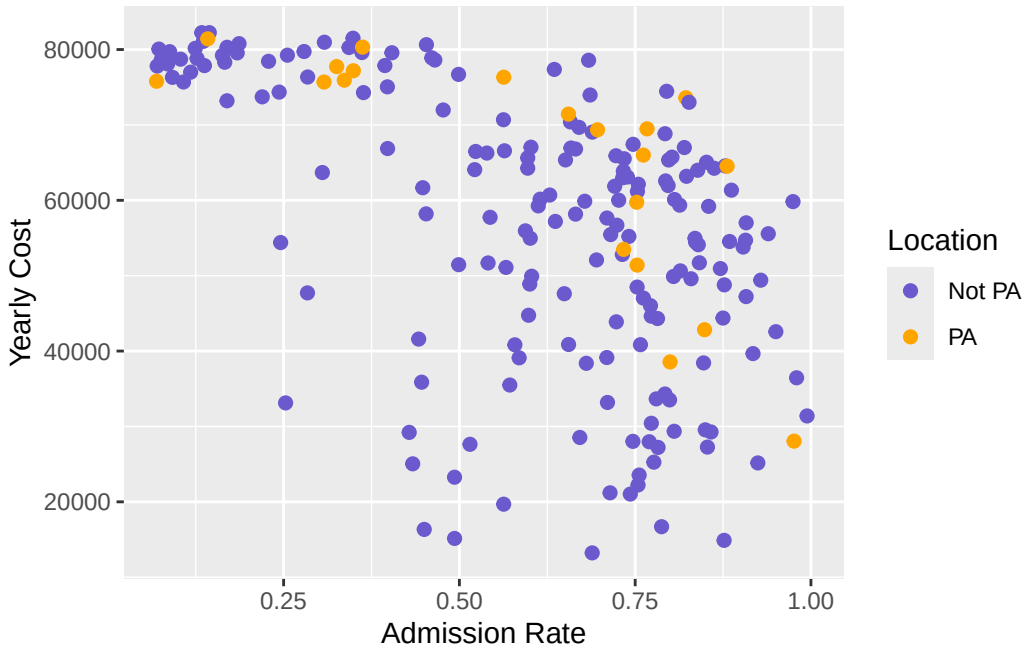
Let's change our graph so that instead of **setting** the color we **map** `Location` to color.

```
ggplot(data = colleges,
       mapping = aes(x = ADM_RATE,
                     y = COSTT4_A,
                     color = Location)) +
  geom_point(size = 2) +
  labs(x = "Admission Rate", y = "Yearly Cost")
```



If we want to pick better colors than salmon and teal, we need to learn about the `scale_()` layer.

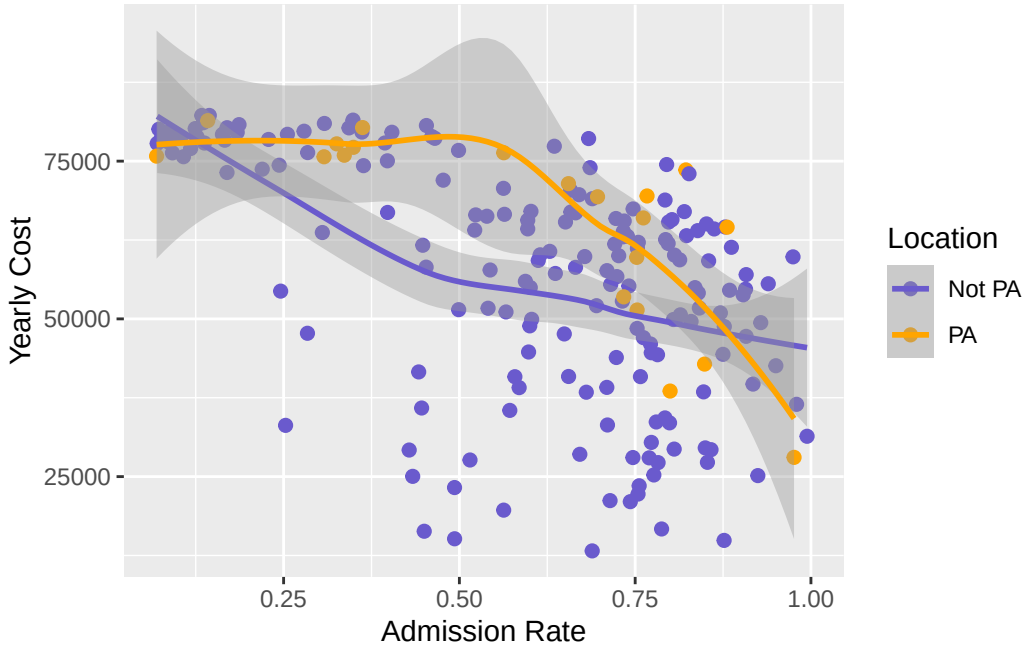
```
ggplot(data = colleges,
       mapping = aes(x = ADM_RATE,
                     y = COSTT4_A,
                     color = Location)) +
  geom_point(size = 2) +
  labs(x = "Admission Rate", y = "Yearly Cost") +
  scale_color_manual(values = c("slateblue", "orange"))
```



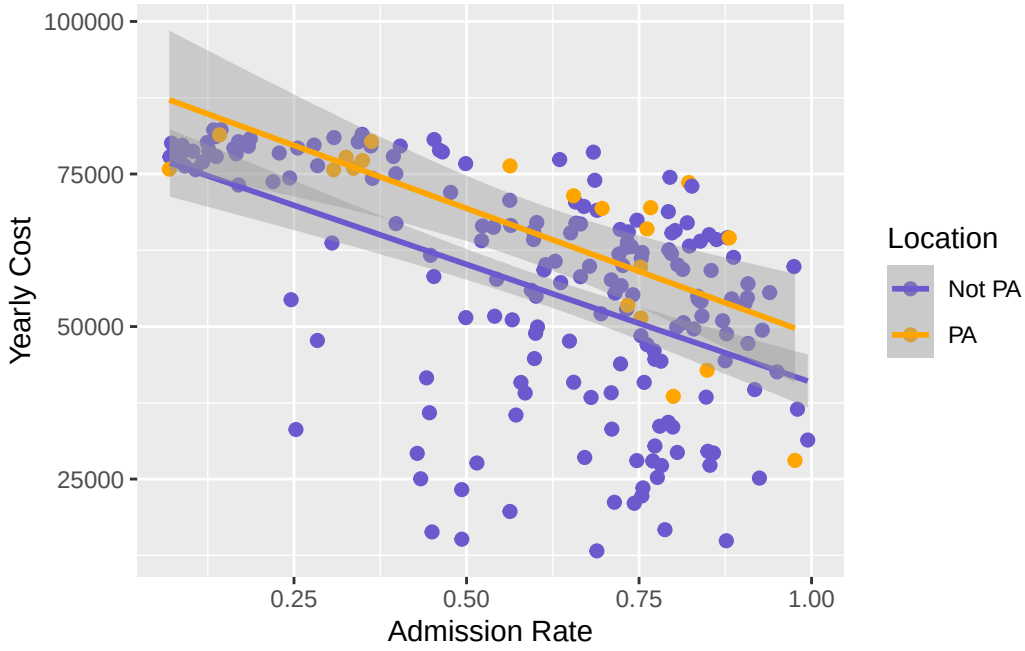
Trends

We can add other `geom`'s to our graph. In particular, I'd like to add a `geom` that captures the general trend between the variables.

```
ggplot(data = colleges,
       mapping = aes(x = ADM_RATE,
                     y = COSTT4_A,
                     color = Location)) +
  geom_point(size = 2) +
  labs(x = "Admission Rate", y = "Yearly Cost") +
  scale_color_manual(values = c("slateblue", "orange")) +
  geom_smooth()
```




```
ggplot(data = colleges,
       mapping = aes(x = ADM_RATE,
                     y = COSTT4_A,
                     color = Location)) +
  geom_point(size = 2) +
  labs(x = "Admission Rate", y = "Yearly Cost") +
  scale_color_manual(values = c("slateblue", "orange")) +
  geom_smooth(method = lm)
```



Do lines seem to capture the general trend in the variables?

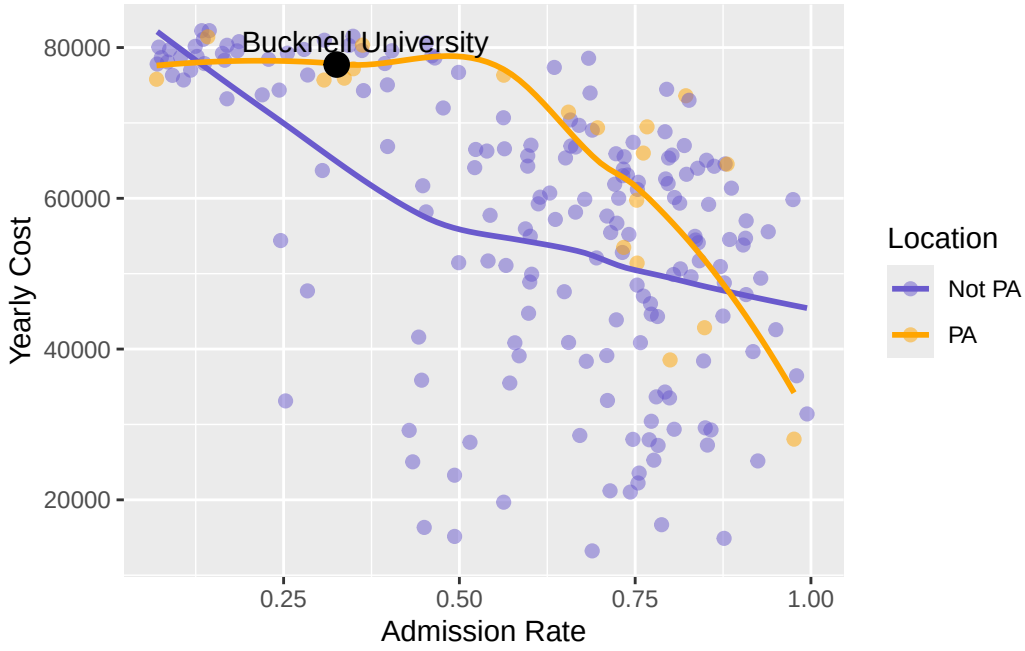
Where is Bucknell?

The following code creates a dataset with only Bucknell's info. (We will learn more about the `filter()` function next time!)

```
Bucknell <- filter(colleges, INSTNM == "Bucknell University")
```

Let's annotate our graph to label Bucknell.

```
ggplot(data = colleges,
       mapping = aes(x = ADM_RATE,
                     y = COSTT4_A,
                     color = Location)) +
  geom_point(size = 2, alpha = .5) +
  labs(x = "Admission Rate", y = "Yearly Cost") +
  scale_color_manual(values = c("slateblue", "orange")) +
  geom_smooth(se = FALSE) +
  geom_point(data = Bucknell, size = 4, color = "black") +
  geom_text_repel(data = Bucknell, mapping = aes(label = INSTNM),
            color = "black")
```



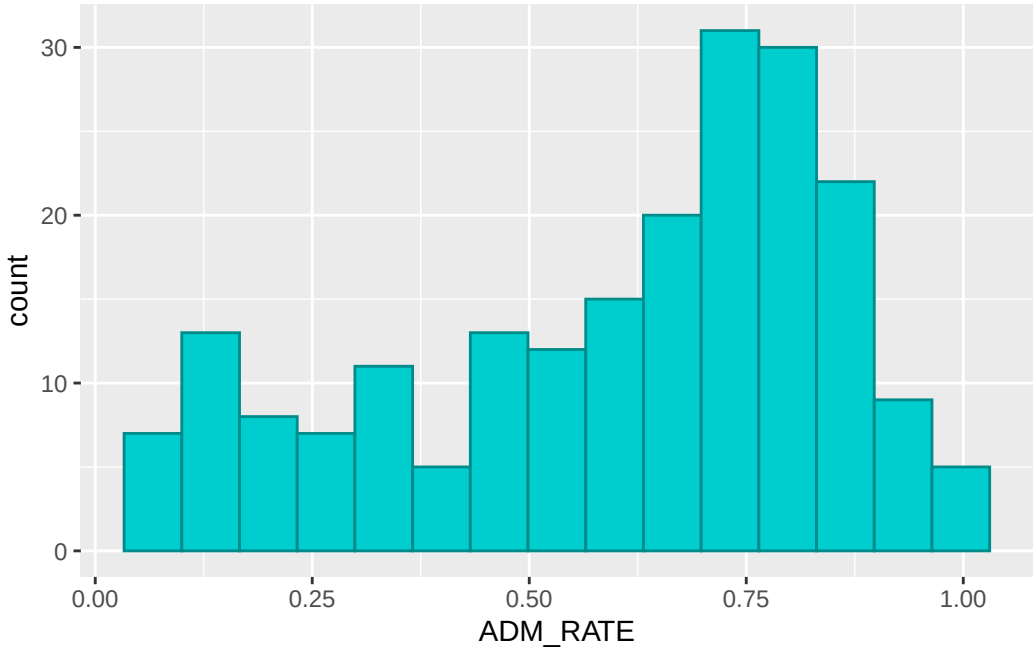
More Graphs

`geom_histogram()`

Let's create a histogram of `ADM_RATE`. Let's also

- Play around with the number of bins.
- Make the bars and outline of the bars fun colors!

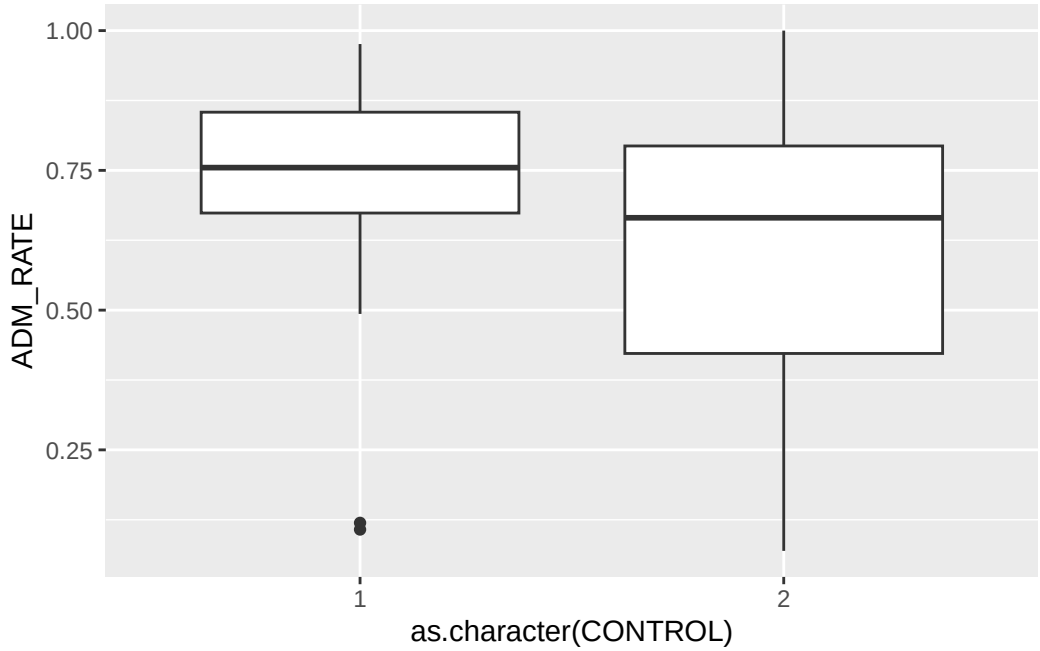
```
ggplot(data = colleges, mapping = aes(x = ADM_RATE)) +
  geom_histogram(bins = 15, color = "cyan4", fill = "cyan3")
```



`geom_boxplot()`

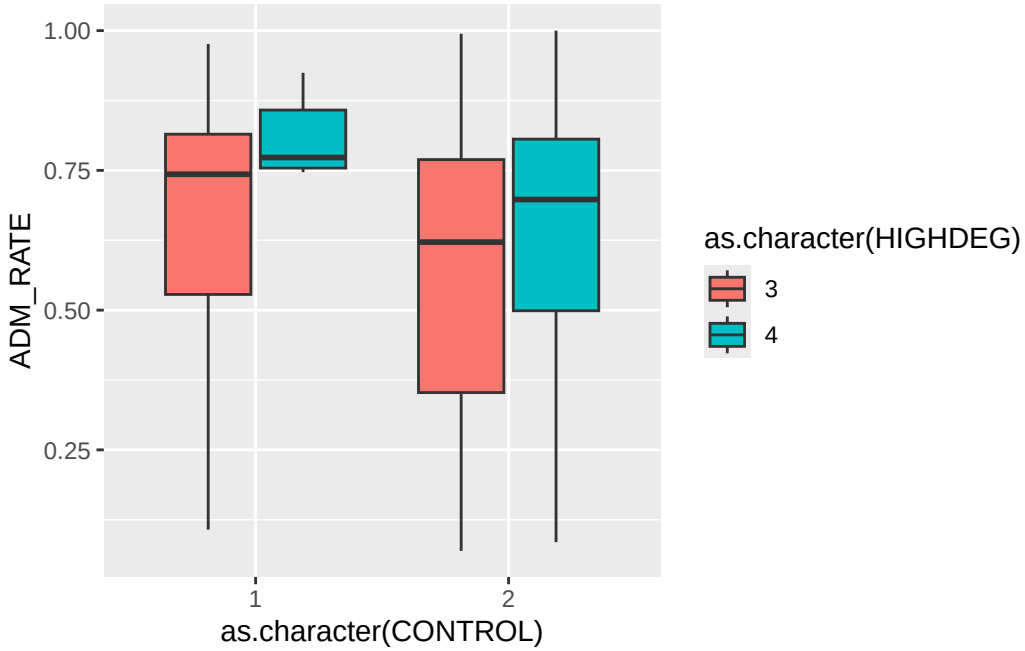
Let's create boxplots of `ADM_RATE` by whether or not a school is private or public (`CONTROL`).

```
ggplot(data = colleges, mapping = aes(y = ADM_RATE,
                                      x = as.character(CONTROL))) +
  geom_boxplot()
```



Let's add `HIGHDEG` to our boxplots by mapping it to `fill`.

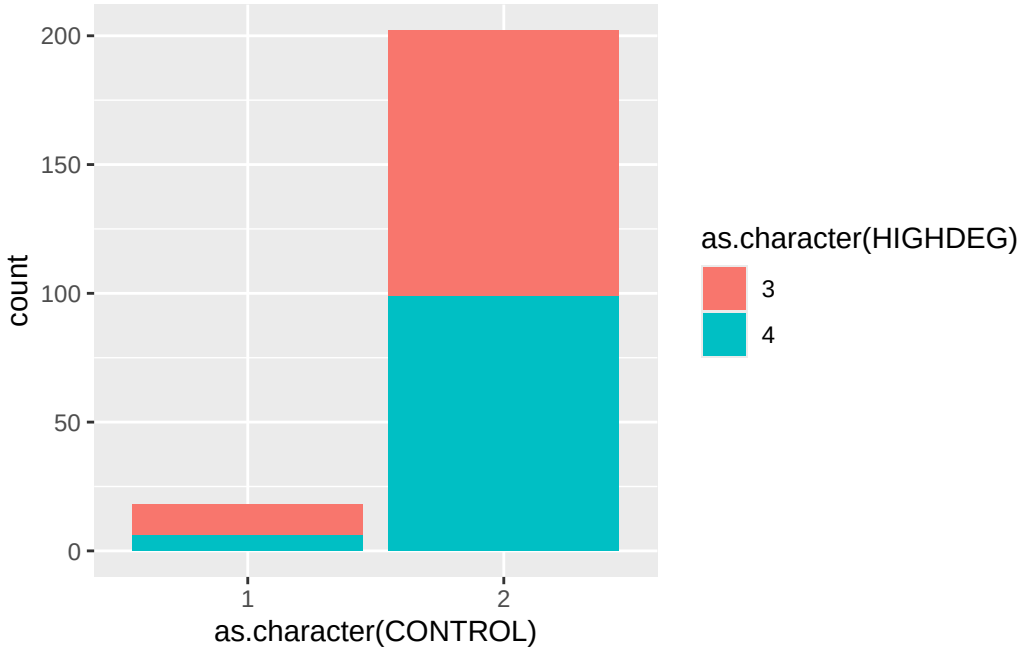
```
ggplot(data = colleges, mapping = aes(y = ADM_RATE,
                                      x = as.character(CONTROL),
                                      fill = as.character(HIGHDEG))) +
  geom_boxplot()
```



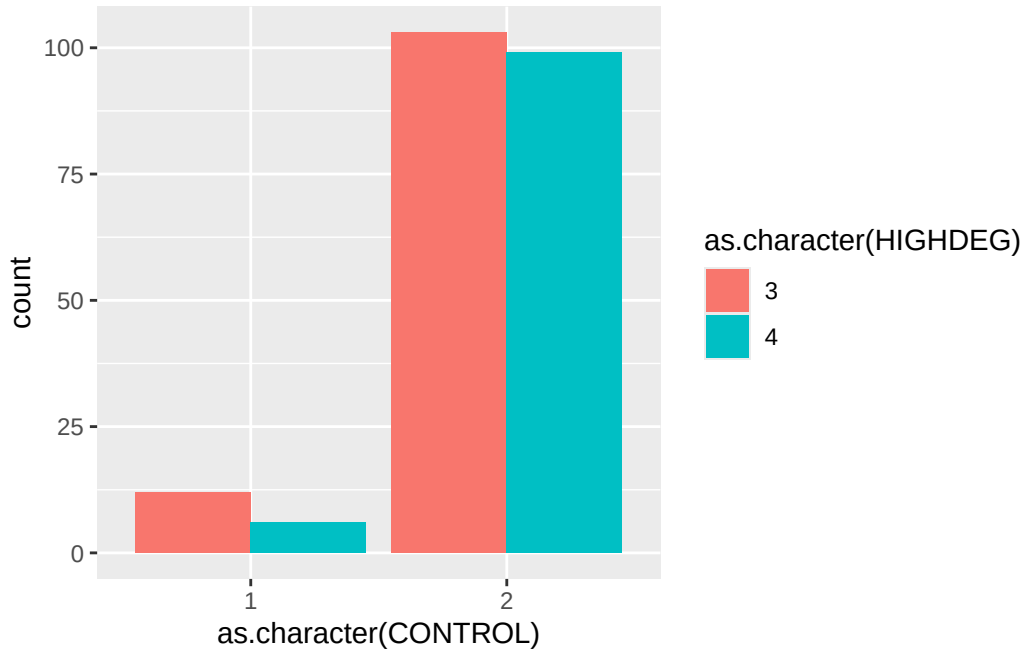
geom_bar()

Let's create bar graphs that compare the number of public and private schools by highest degree awarded.

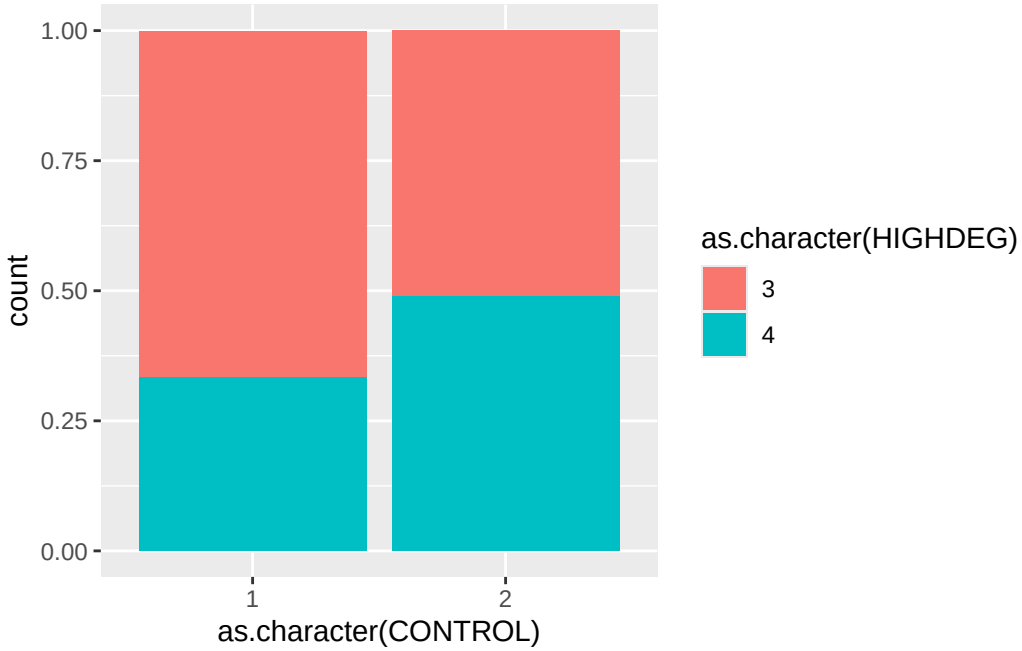
```
# Graph of counts
ggplot(data = colleges, mapping = aes(x = as.character(CONTROL),
                                      fill = as.character(HIGHDEG))) +
  geom_bar()
```




```
# Graph of dodged counts
ggplot(data = colleges, mapping = aes(x = as.character(CONTROL),
                                      fill = as.character(HIGHDEG))) +
  geom_bar(position = "dodge")
```




```
# Graph of proportions
ggplot(data = colleges, mapping = aes(x = as.character(CONTROL),
                                      fill = as.character(HIGHDEG))) +
  geom_bar(position = "fill")
```



Your Optional Homework

Create plots of some of the other variables in the dataset or of some of your own data. For each, think about:

- The variables and what to map them to (`x`, `y`, `color`, `fill`, …)
- Creating informative labels
- Setting and mapping additional aesthetics
- The story your graph tells

Resources for Learning More about Graphing with `ggplot2`

- Modern Dive's chapter on Data Visualization
- R for Data Science's chapter on Data Visualization
- `ggplot2` cheatsheet: https://rstudio.github.io/cheatsheets/data-visualization.pdf