

Statistics II

Week 2: **Causes & Effects**

Content for Today

1. Review of core concepts from lecture
2. Calculating NATE and ATE
3. Creating plots with `ggplot2`
4. Creating and plotting simulated data
5. Plotting DAGs with `ggdag`

Lecture Review

Potential Outcomes Framework

Key concept: Every individual in a sample could *theoretically* be exposed to two states with different potential outcomes. However, we can only ever observe one state. The other will always be a counterfactual, meaning we cannot observe causal effects at the individual level.

Therefore, we look at averages:

ATE: Expected mean value for those in the treatment group minus the expected mean value for those same individuals, had they not been treated.

NATE: The difference in the sample means on the observed outcome variable Y for the observed treatment and control groups.

Estimating Bias

We can define the ATE as a weighted average of the ATT and ATC:

ATT: The expected mean for those in the treatment group *with* treatment, minus the expected mean of those in the treatment group *without* treatment.

ATC: The expected mean for those in the control group *with* treatment, minus the expected mean of those in the control group *without* treatment.

$$ATE = \pi ATT + (1 - \pi)ATC$$

Where π is the proportion of subjects under treatment

Estimating Bias

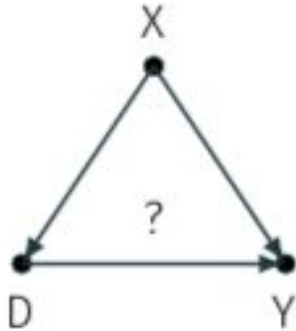
Total bias of NATE as an estimate of ATE is simply $\text{NATE} - \text{ATE}$

It is made up of the **baseline bias** (difference in average outcome without treatment for the treatment and control groups) and **differential treatment effect bias** (the difference in the average treatment effect between the treatment and control groups, weighted by the proportion of the population in the control group).

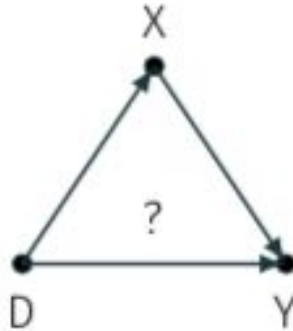
Causal Graphs/DAGs

Basic causal patterns

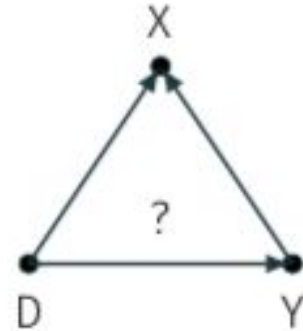
X is **confounder**.



X is **mediator**.



X is **collider**.



Causal Graphs/DAGs

Back-door path: Any path between the causal variable and the outcome variable that begins with an arrow that points into the causal variable.

Satisfying the **backdoor criterion** just means you closed all the backdoor paths.

The key goal of a conditioning strategy is not to adjust for any particular confounder, but rather to remove the portion of the total association between D and Y that is non-causal.

Randomized Experiments

- In randomized experiments, treatment is randomly assigned.
- This is important because:
 - Causal effects are identified if potential outcomes are independent of the treatment assignment process (independence/ignorability assumption).
 - A randomized experiment is the only research design that justifies the independence/ignorability assumption a priori on procedural grounds.
- In this case, the expected difference in Y between treatment and control is just the average of the above differences: $NATE = ATE$.

Independence/ignorability assumption:

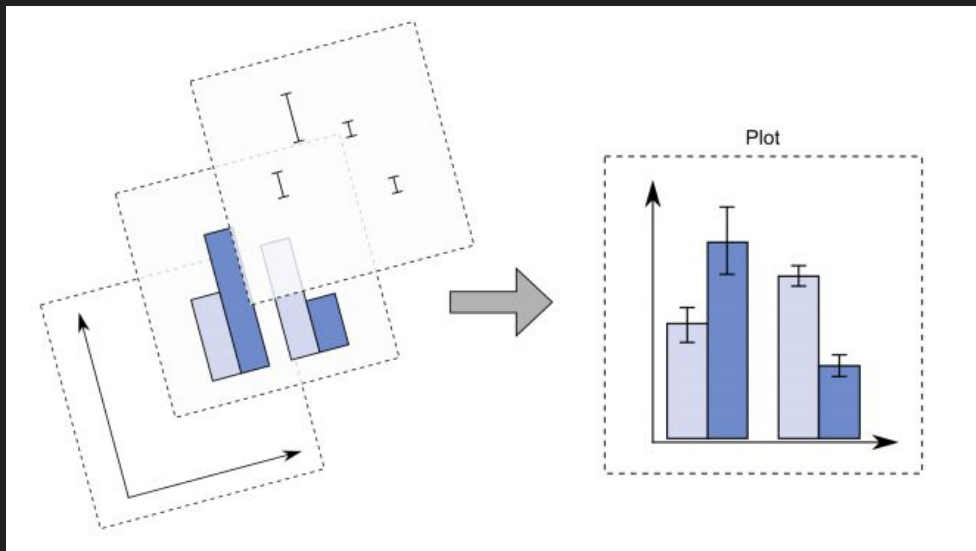
The expected outcome of treatment for those treated would be the same for those in treatment or control group, and vice versa. Basically, the fact that you get treatment or not does not affect your potential outcomes.

Questions?

Plotting with `ggplot2`

ggplot2

In `ggplot2`, a graph is made up of a series of layers



Download the cheat sheet: <https://tinyurl.com/h5o9tfq>

ggplot2

Describes all the non-data ink

Plotting space for the data

Statistical models & summaries

Rows and columns of sub-plots

Shapes used to represent the data

Scales onto which data is mapped

The actual variables to be plotted

Theme

Coordinates

Statistics

Facets

Geometries

Aesthetics

Data



ggplot2

Geometric objects are the visual elements such as bars and points: `geom()`

- There are many kinds of geoms, such as scatterplots (`geom_point`) or barplots (`geom_bar`)

The appearance and location of these geoms (such as size and color) are controlled by the **aesthetics properties**: `aes()`

- The variables you want to plot are referred to here.

```
myGraph <- ggplot(data,  
                  aes(x = variable_for_x_axis,  
                     y = variable_for_y_axis)) +  
  geom()
```

ggplot2

Types of Geoms:

Scatterplot: `geom_point()`

Histogram: `geom_histogram()`

Barplot: `geom_bar()`

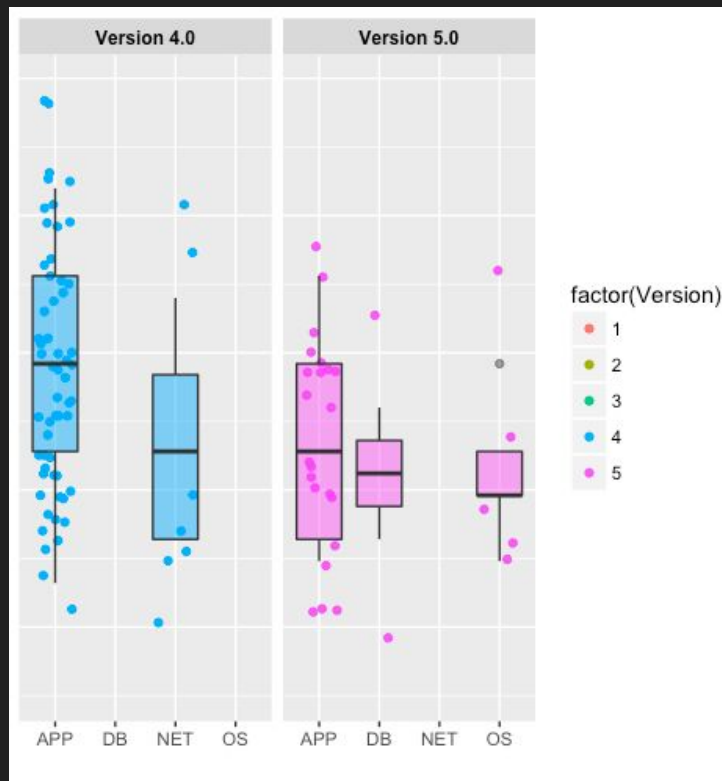
Boxplot: `geom_boxplot()`

Density: `geom_density()`

Adding a “linear regression” line:

`geom_smooth(model = lm)`

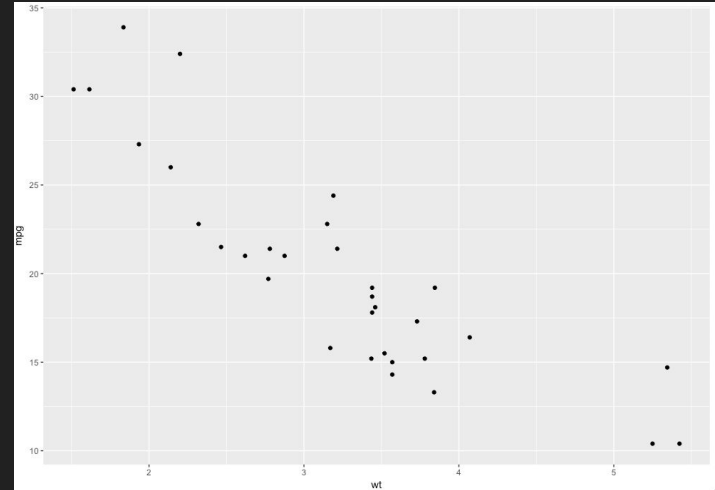
Or a combination of multiple.



ggplot2

Using the build-in `mtcars` data frame,
we can plot a basic scatterplot:

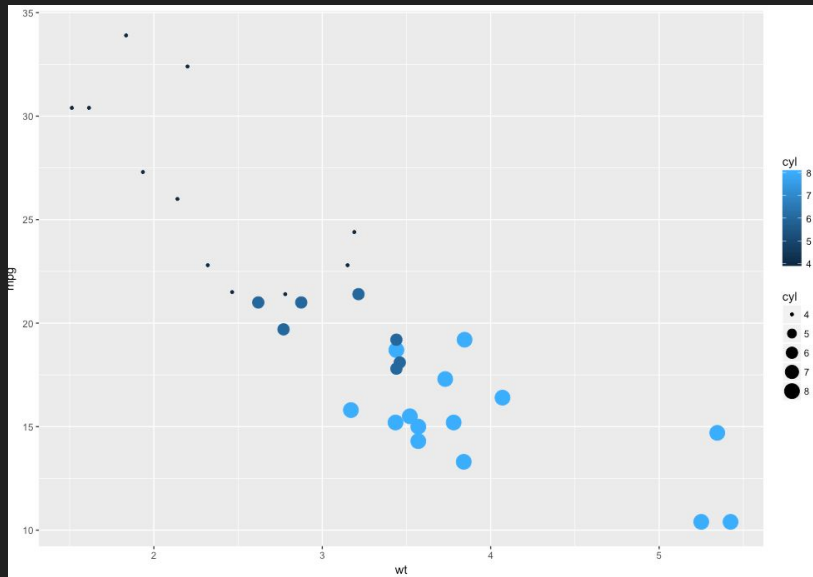
```
scatterplot <- ggplot(  
  data = mtcars,  
  aes(x = wt, y = mpg) ) +  
  geom_point()  
  
print(scatterplot)
```



ggplot2

From here we can add extra design elements, like changing the color and size of the points based on cylinder size:

```
scatterplot <- ggplot(  
  data = mtcars,  
  aes(x = wt,  
      y = mpg,  
      col = cyl,  
      size = cyl)) +  
  geom_point()
```



ggplot2

There are lots of other things we can do, too!

Change the transparency using `alpha` and a number from 0-1: ex. `alpha = 0.6`

Add a theme: ex. `geom_bar() + theme_bw()`

Add labels and titles: ex. `+ xlab('X label') + ylab('Y label') + ggtitle(' Title')`

Change the size and shape of lines, points, etc.

Zoom in on a certain part of a graph

And more :)