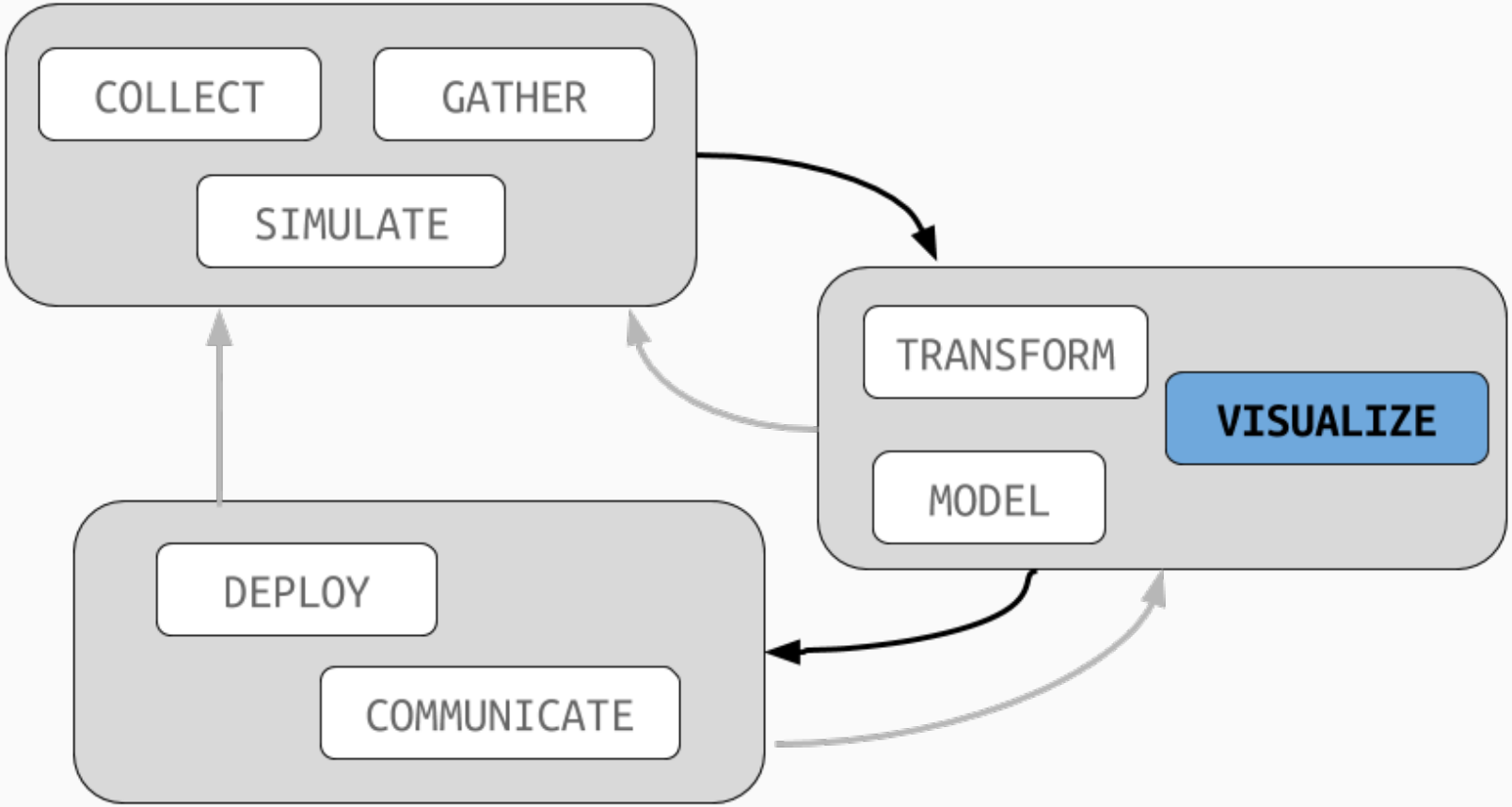
Lecture 26: ggplot2

Taylor Arnold


COLLECT
GATHER
SIMULATE
TRANSFORM
VISUALIZE
MODEL
DEPLOY
COMMUNICATE

qplot

When we first saw plots in this course, I introduced the `qplot` function. It takes an x-coordinate, option y-coordinate, and dataset. From these, it determines the best type of plot to create. Later I showed how to add additional layers onto the plot and even how to turn off the original default.

qqplot

The Q in `qplot` stands for "quick" because (as you saw), it can used with a very minimal amount of understanding about the **ggplot2** package.

Today I'll show you how to build plots without `qplot`. Given everything we know already, this will be quite easy.

ggplot

The `ggplot` function serves to create a new, empty graphic. The following would create a completely empty plot:

```
ggplot()
```

ggplot

```
ggplot()
```

ggplot: data

Next, the first argument to `ggplot` is to dataset name that we want to work with.

```
ggplot(msleep)
```

On its own this will still construct just an empty plot, however it now has a dataset (invisibly) attached to it.

ggplot: aesthetics

The next step is to assigned the aesthetics to the plot. These should be familiar to us, and consist of things like `x`, `y`, `color`, and `label`. We cannot give these directly however; they need to be wrapped in the `aes` function:

```
ggplot(msleep, aes(x = awake, y = sleep_rem, color = vore))
```

Again, nothing will happen yet, but we have made invisible changes to the graphic.

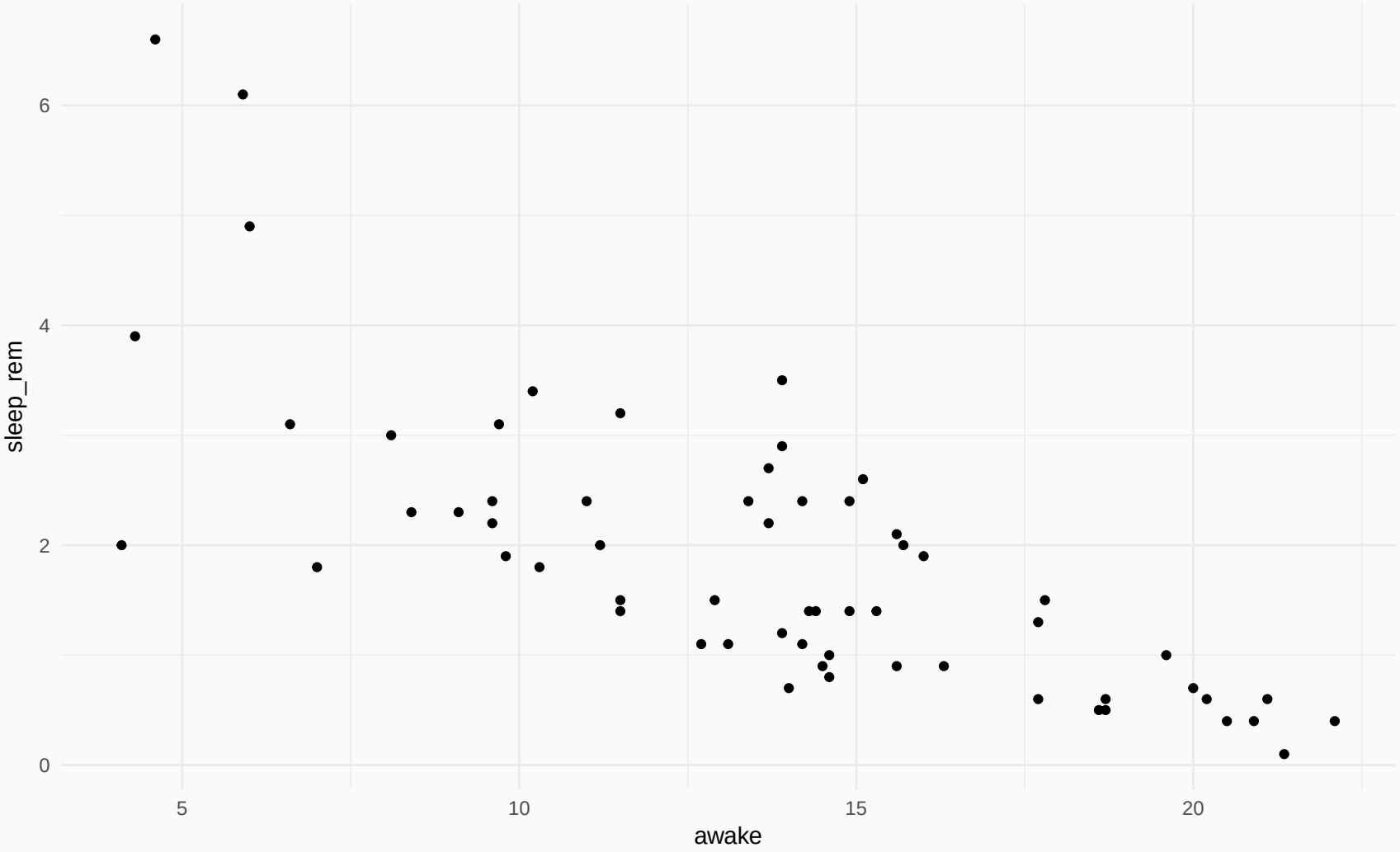
ggplot: layers

The rest we actually already know; we need to add layers such as `geom_point`, `geom_line`, and `geom_text` to this base plot:

```
ggplot(msleep, aes(x = awake, y = sleep_rem, color = vore)) +
    geom_point()
```

ggplot: layers

```
ggplot(msleep, aes(x = awake, y = sleep_rem), color = "red") +
    geom_point()
```



ggplot: why?

One reason for learning this new calling structure is that you'll find many code snippets online that use it. Searching for help with **ggplot2** will more likely than not lead to StackOverflow or other resources that assume you are familiar with it.

The second reason is that the `aes()` calling mechanism is actually accepted by all of the `geom_` layers. This allows us to plot multiple facets at once.

ggplot: example

Here, we color the points by the vore variable but fit just a single line to the plot:

```
ggplot(msleep, aes(x = awake, y = sleep_rem)) +
    geom_point(aes(color = vore)) +
    geom_smooth()
```

ggplot: example

```
ggplot(msleep, aes(x = awake, y = sleep_rem)) +
    geom_point(aes(color = vore)) +
    geom_smooth()
```