

# Plotting datasets with ggplot2 in R

Kaitlin Flynn, Ph.D.

Ann Arbor R User Group

2/9/17

Am I out of touch?

No, it's the children who  
are wrong.



# Base graphics

## Pros

- Built in to R
- Single, specialized functions for each type of plot
- Quick (if you learned this first)
- Fast

## Cons

- “Ugly”
- Not as flexible, need to use different commands for different plots
- Need to add legend, color points, etc if you want it
- Can be tedious for multiple plots, or need to add loops

# ggplot2: “Grammar of Graphics”

## Pros

- Makes visually-pleasing graphs from the get-go
- Relies on geom layers to define and tweak each layer of chart
- Automatic coloring
- Automatic legend

## Cons

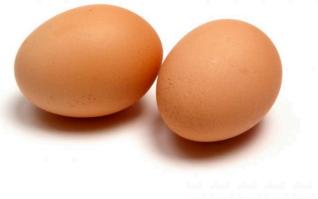
- Layers may be cumbersome to code
- May create a “Prism problem” where graph looks publishable but is not complete
- computationally slower

# Making a base plot: mix ingredients, out comes a cake!

```
plot(x = diamonds$carat, y= diamonds$price,
```

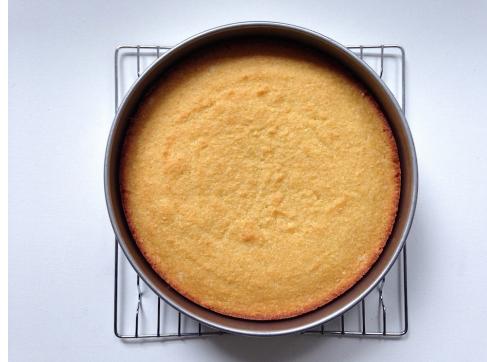


```
type = 'p', main = "Diamond cost by size")
```

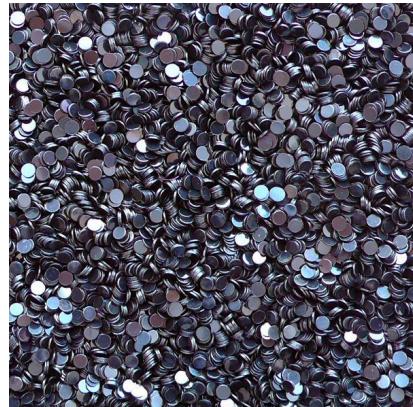


# The Grammar of Graphics: Start with a cake, layer on the good stuff

```
ggplot(data = diamonds, aes(x = carat, y = price))
```

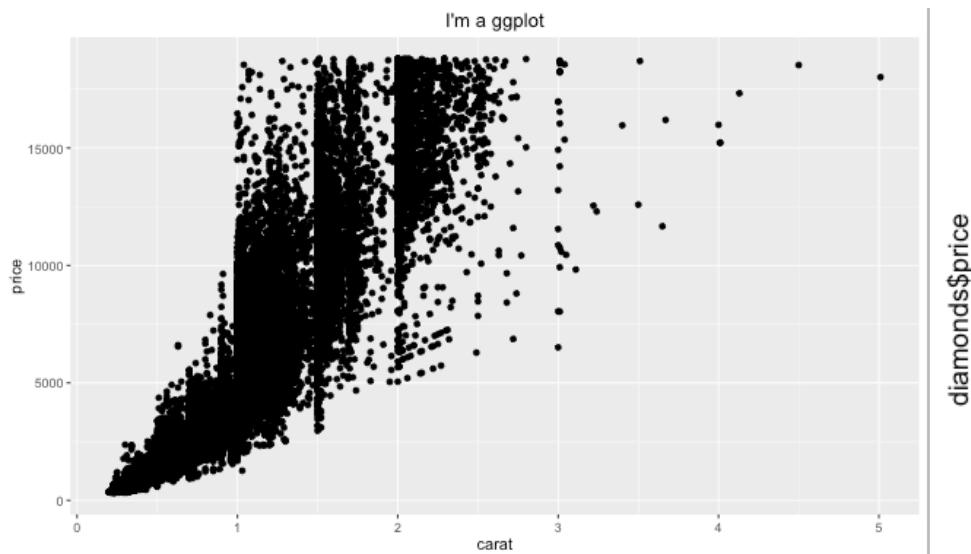


+ geom\_point() + ggtitle("Diamond cost by size")



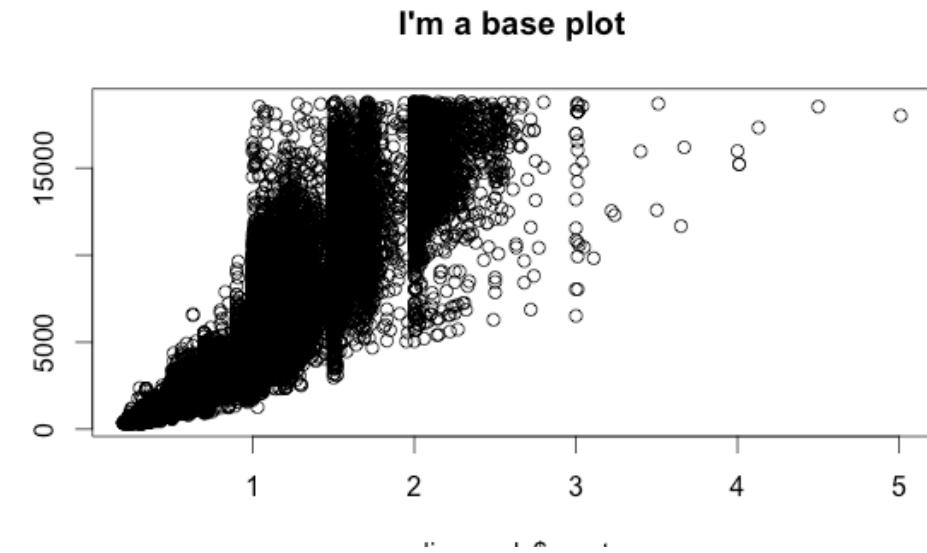
# Two ways to make a cake (plot)

**ggplot**



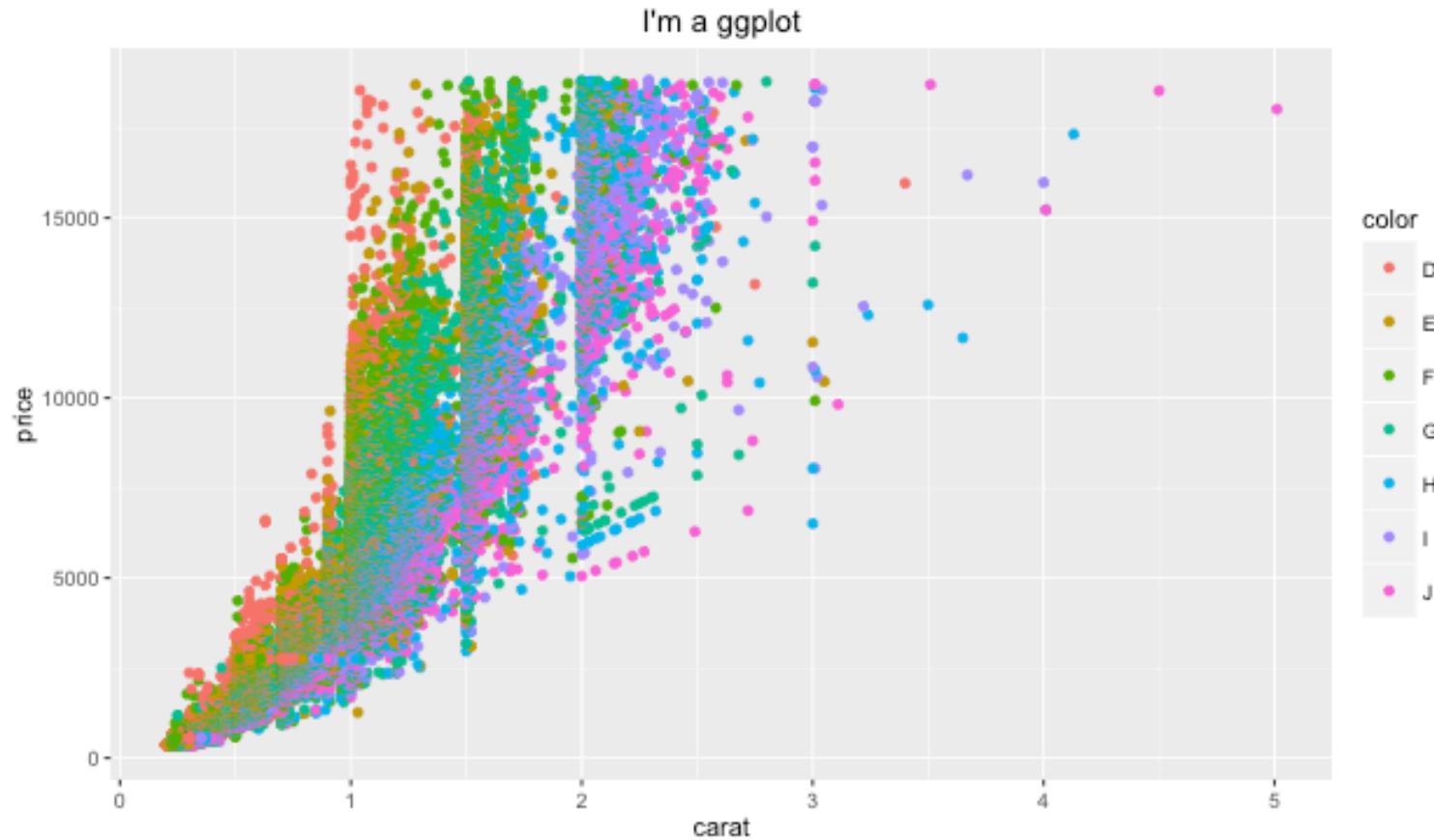
```
ggplot(data=diamonds, aes(x=carat, y=price)) + geom_point() +  
  ggtitle("I'm a ggplot")
```

**plot**



```
plot(x=diamonds$carat, y=diamonds$price, type='p',  
      main = "I'm a base plot")
```

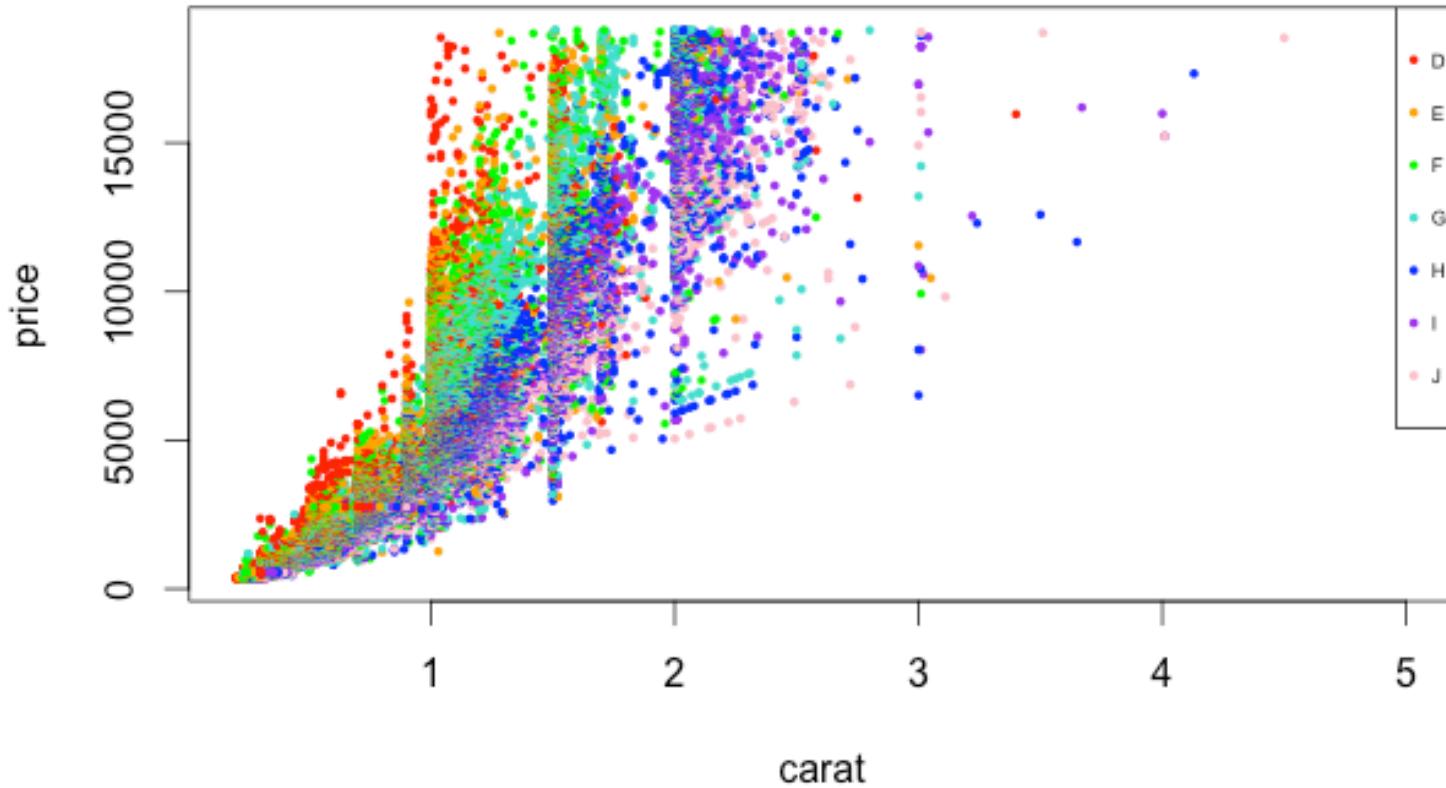
# Making more complex plots:



```
ggplot(data = diamonds, aes(x = carat, y = price, colour = color))  
+ geom_point() + ggtitle("I'm a ggplot")
```

# Making more complex plots:

I'm a base plot

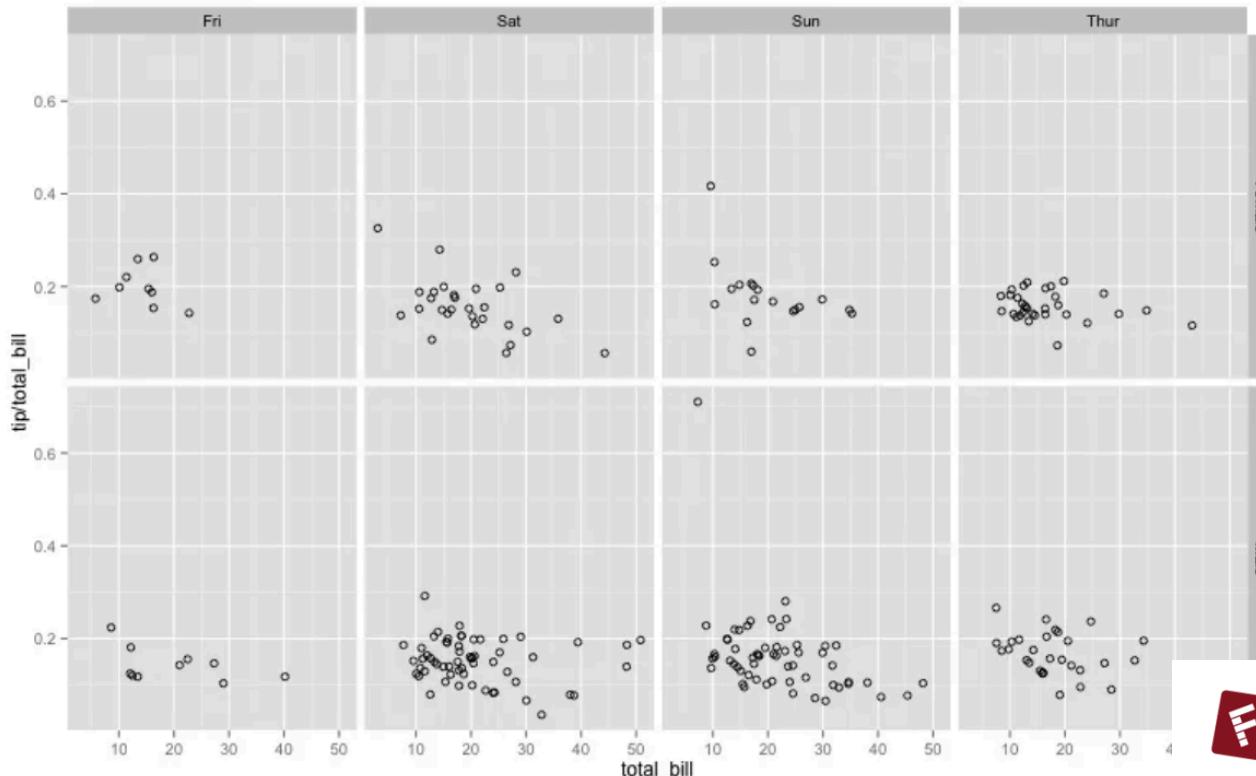


```
plot(x=diamonds$carat, y=diamonds$price, type='p', xlab= "carat", ylab="price",
      main = "I'm a base plot", pch=16, cex=0.5,
      col=c('red', 'orange', 'green', 'turquoise', 'blue', 'purple', 'pink')[diamonds$color])
legend <- c("D", "E", "F", "G", "H", "I", "J")
legend(x = "topright", legend, col=c('red', 'orange', 'green', 'turquoise', 'blue', 'purple', 'pink'), pch=16, cex=0.5)
```

# ggplot rocks at: Facet plots (cupcakes!)

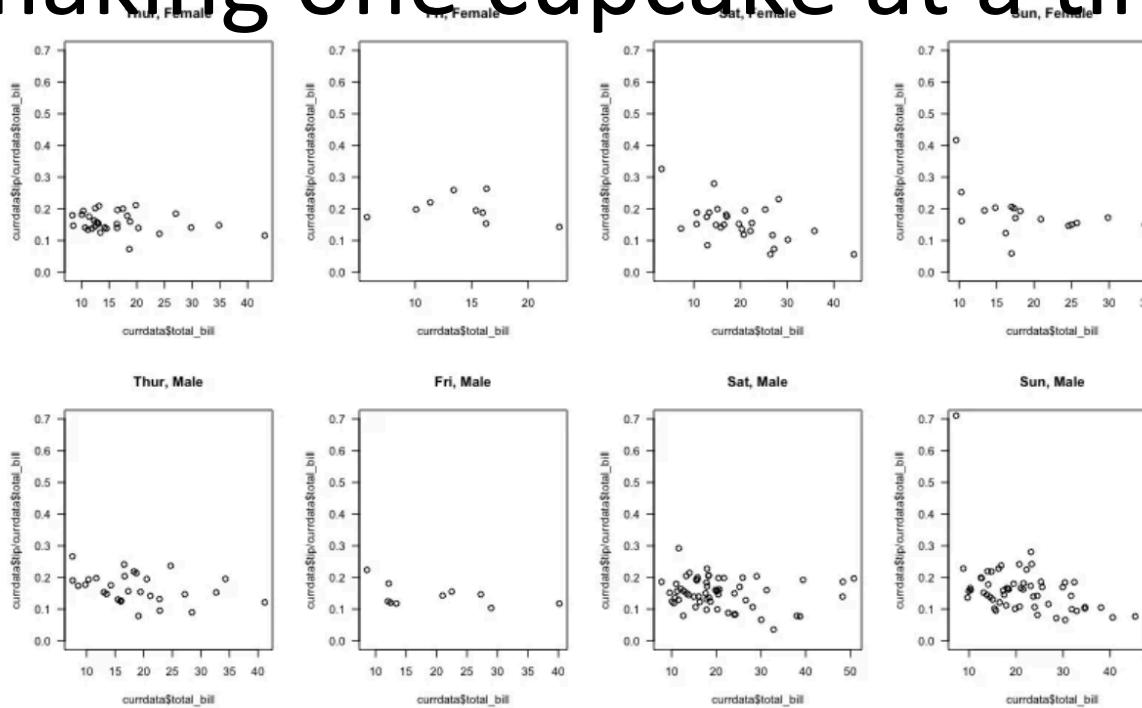
```
sp <- ggplot(tips, aes(x=total_bill, y=tip/total_bill))  
  + geom_point(shape=1)  
sp + facet_grid(sex ~ day)
```

You get a breakdown for age and sex.



FLOWINGDATA

# Facet plots in R require loops- like making one cupcake at a time



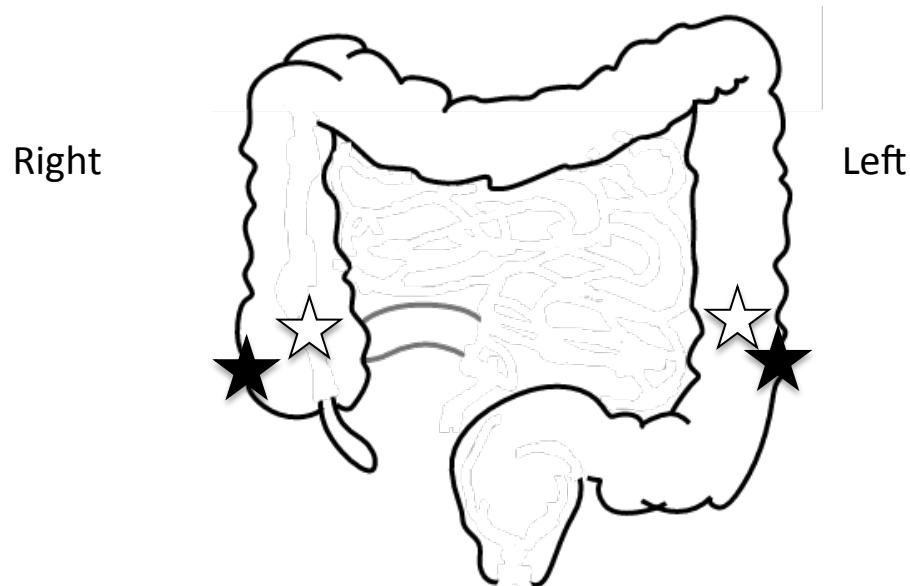
```
par(mfrow=c(2,4))
days <- c("Thur", "Fri", "Sat", "Sun")
sexes <- unique(tips$sex)
for (i in 1:length(sexes)) {
  for (j in 1:length(days)) {
    currdata <- tips[tips$day == days[j] & tips$sex == sexes[i],]
    plot(currdata$total_bill, currdata$tip/currdata$total_bill,
         main=paste(days[j], sexes[i], sep=", "), ylim=c(0,0.7), las=1)
  }
}
```



FLOWINGDATA

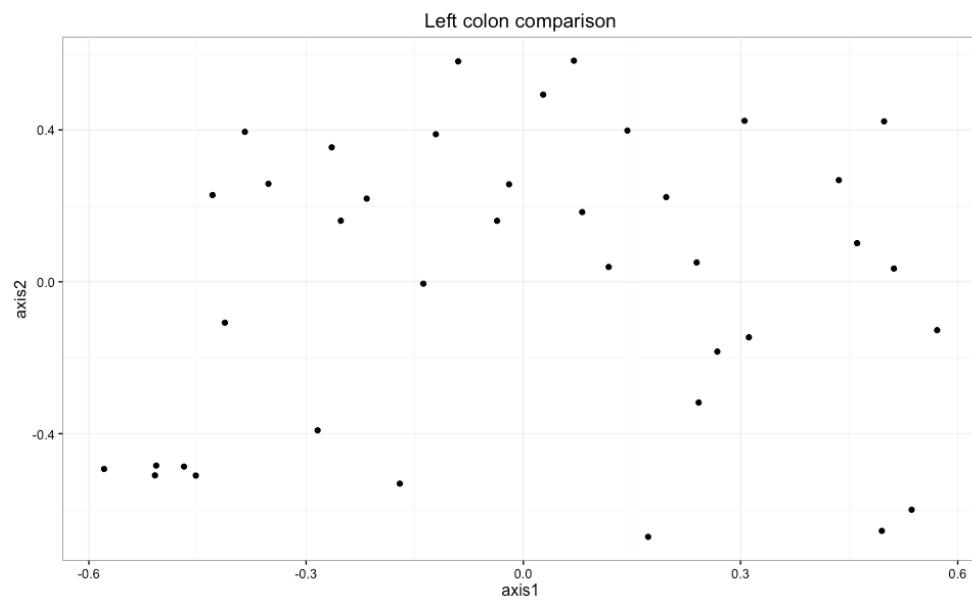
# How do I use ggplot in my workflow?

- I have bacterial data from biopsies of the left and the right colon for 20 people



# How do I use ggplot in my workflow?

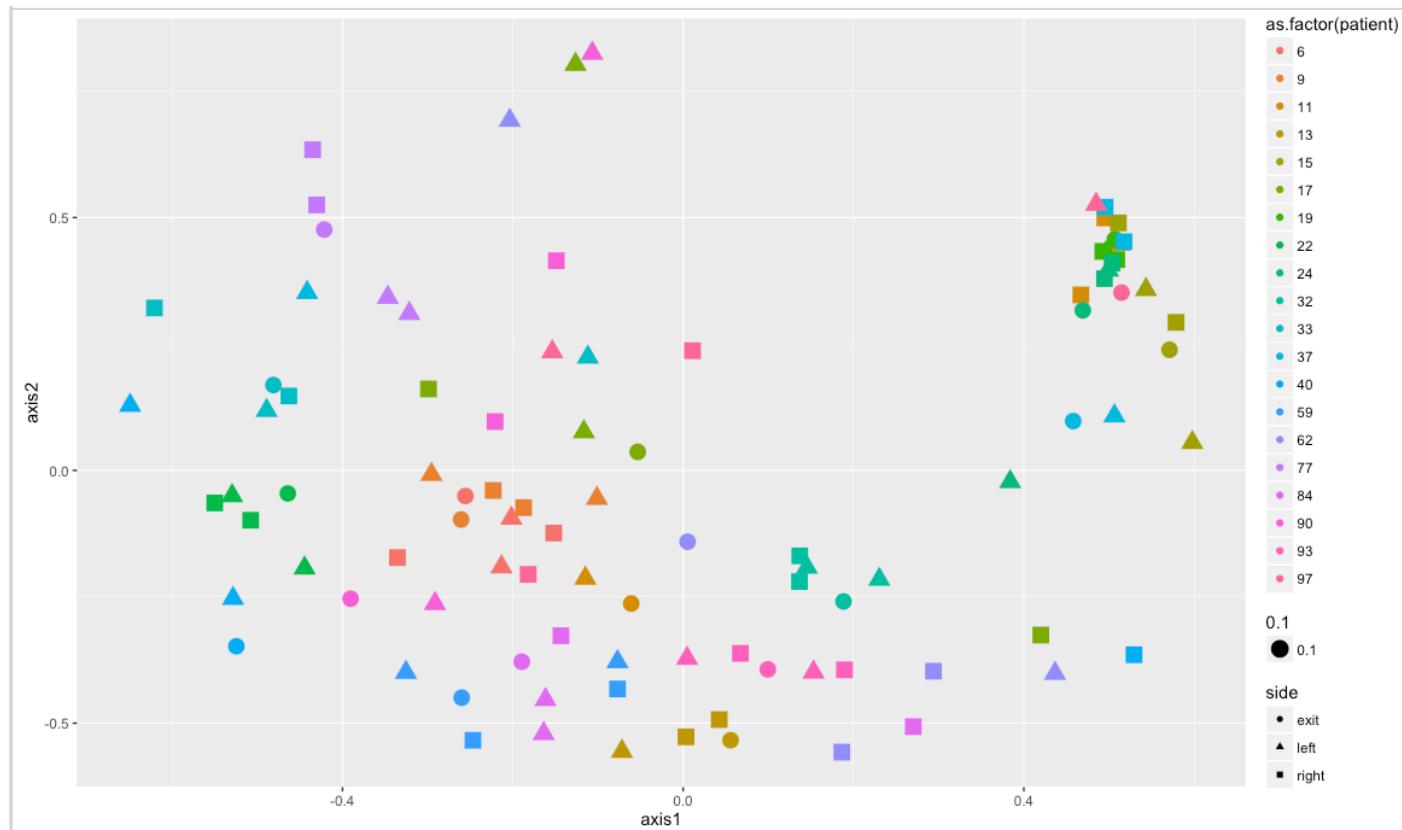
- First, I'll do a ggplot of all of the data on the left, or all of the data on the right just to get an overall idea of the results



```
ggplot(leftnmds, aes(axis1, axis2)) +geom_point() +theme_bw() +ggtitle("Left colon comparison")
```

# How do I use ggplot in my workflow?

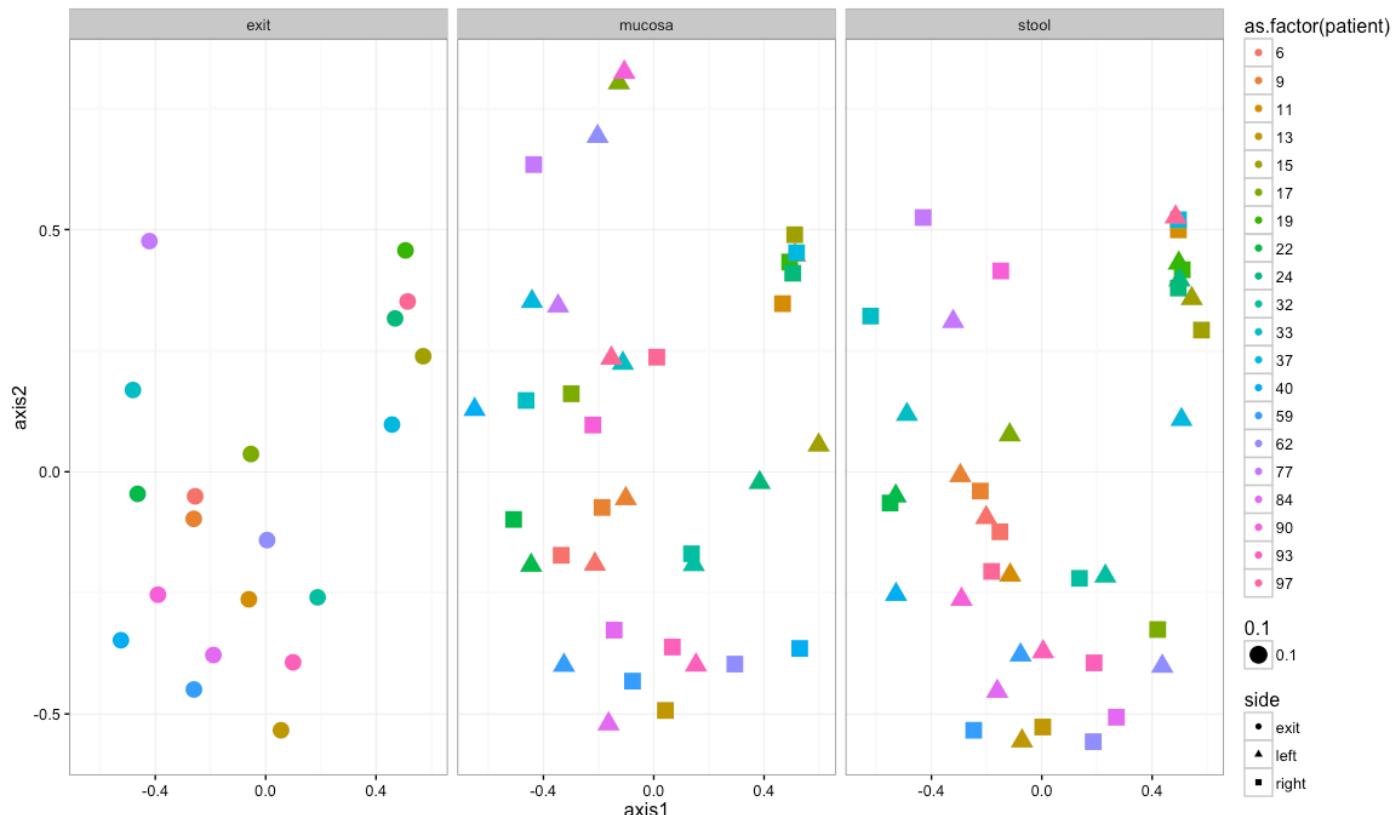
- I will then color or change the shape of the points by the location of the sample I'm measuring



```
ggplot(nmds, aes(axis1, axis2)) +geom_point(aes(color=as.factor(patient), shape=side, size = 0.1))
```

# How do I use ggplot in my workflow?

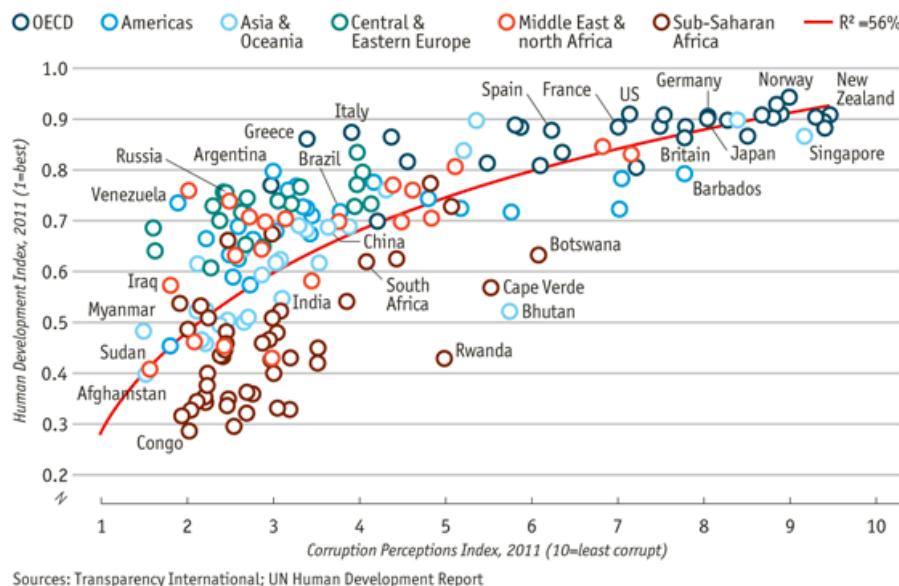
- Then I can subset or facet the data by another factor, like the side each sample comes from



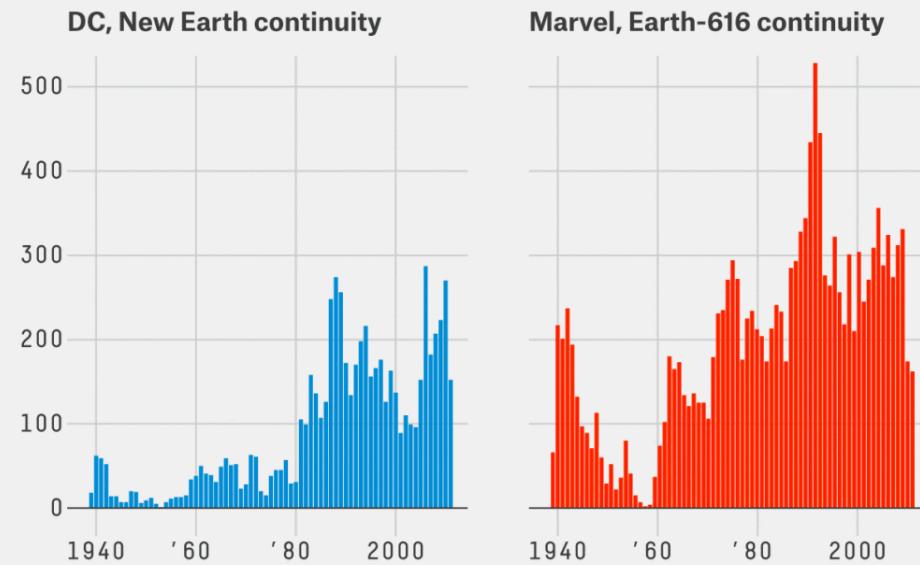
```
ggplot(nmds, aes(axis1, axis2)) +geom_point(aes(color=as.factor(patient), shape=side, size = 0.1)) + facet_wrap(~site) +theme_bw()
```

# ggplot is widely used in data journalism

## Corruption and human development



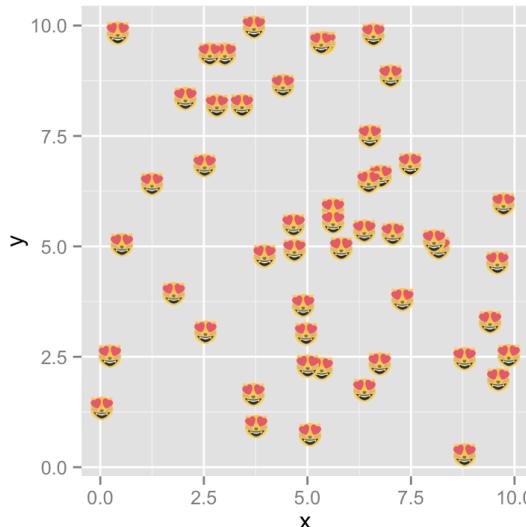
## New Comic Book Characters Introduced Per Year



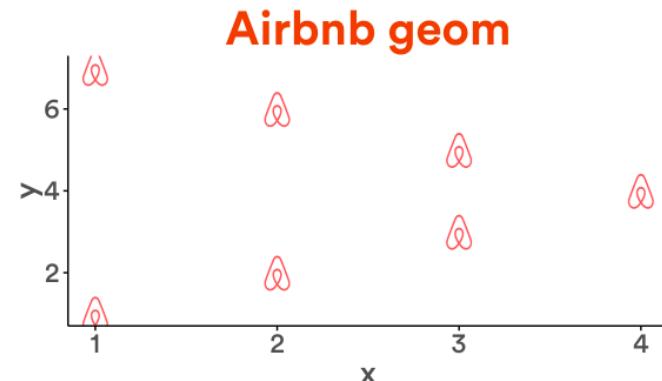
# ggplot has fun extensions

- Plotly package works with ggplot2 plots to make interactive charts ([online example](#))
- ggthemes() to get the FiveThirtyEight, Airbnb theme, emoGG to add emoji to plots
- Other profession-specific add-ons, network graphs, phylogenetic trees, radar charts

```
posx <- runif(50, 0, 10)
posy <- runif(50, 0, 10)
ggplot(data.frame(x = posx, y = posy), aes(x, y)) + geom_emoji(emoji="1f63b")
```



```
ggplot(aes(x,y), data=d2) +
  geom_tech(size=0.25, theme="airbnb") +
  theme_tech("airbnb") +
  ggtitle("Airbnb geom")
```



# In summary...

Now that I looked at `ggplot2` more closely, do I want to switch away from base graphics? No. -Nathan Yau, FlowingData

Simply Statistics A statistics blog by Rafa Irizarry, Roger Peng, and Jeff Leek

## Why I don't use ggplot2

11 Feb 2016



MANDY MEJIA

### 10 REASONS TO SWITCH TO GGPLOT

Making plots is a n  
who works with da  
a pretty straightfor  
programming lang  
using R's base grap  
you actually *want* t  
As with many of lif



David Robinson

Data Scientist at Stack Overflow, works in R and Python.

### Why I use ggplot2

If you've read my blog, taken one of  
you probably know I'm a big fan of H  
compared to base R plotting.

Not everyone agrees. Among the  
who yesterday wrote up his thoughts

*...one place I lose tons of street cred in  
about ggplot2... ggplot2 is an R packa  
was created by Hadley Wickham, who  
important statistician/data scientist on*

There's also no problem with using everything available to you. At the end of the day,  
it's all R. -Nathan Yau, FlowingData

