Eating Your Own Dogfood

Visualization in the Feedback Loop

Josh Laurito

Today's To-Dos

- About you
- Review last module's homework
- Exploratory data analysis
- ggplot2
- BigVis
- devtools
- This module's homework

About you

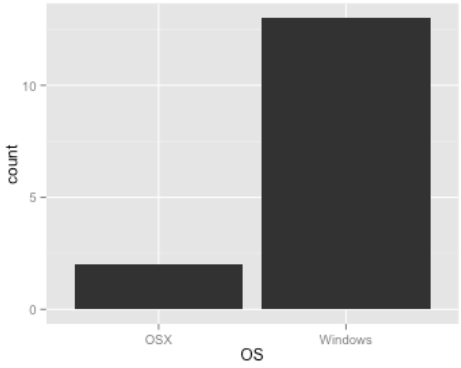
Thanks for filling out the initial survey for the class

(i learned a lot)

About you

- You are very windows heavy

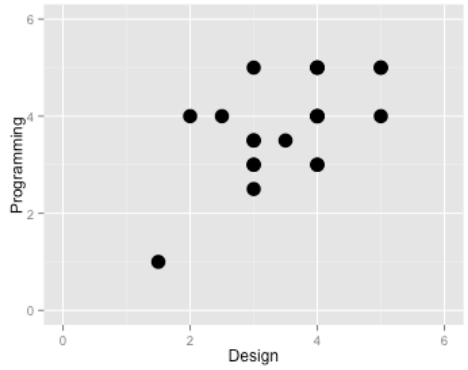
```
suppressPackageStartupMessages(library(ggplot2))
setwd('/Users/josh.laurito/personal/cuny/2016-fall/lecture2')
class <- read.csv('survey.csv')
class <- class[complete.cases(class),]
ggplot(aes(x=OS), data=class) + geom_histogram()
```



About you

- Your confidence varies

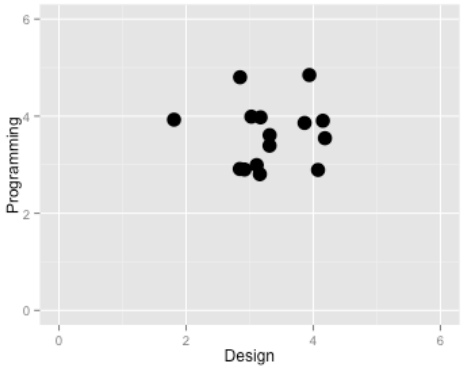
```
p <- ggplot(class, aes(x=Design, y=Programming))
p + geom_point(size=5) + ylim(0,6) + xlim(0,6)
```



About you

- Maybe we should use jitter

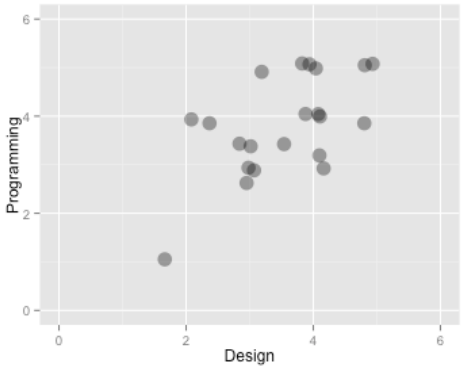
```
p <- ggplot(class, aes(x=Design, y=Programming))
p + geom_point(size=5, position='jitter') + ylim(0,6) + xlim(0,6)
```



About you

- Maybe we should use jitter and alpha

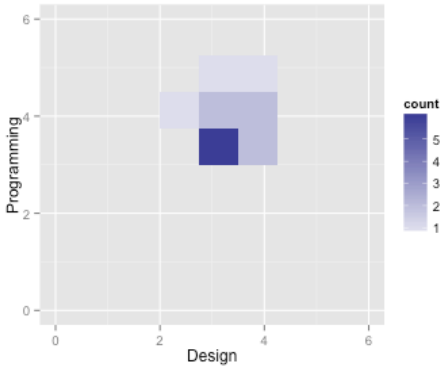
```
p1 <- ggplot(class, aes(x=Design, y=Programming))
p1 + geom_point(size=5, position='jitter', alpha=.4) + ylim(0,6) + xlim(0,6)
```



About you

- or in bins?

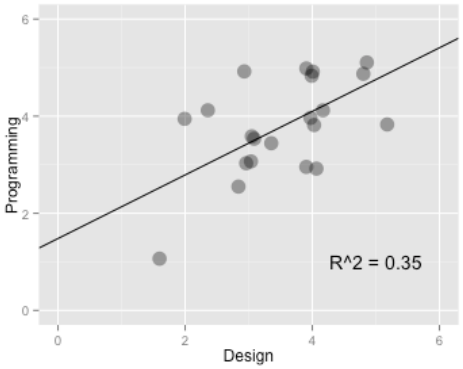
```
p2 <- ggplot(class, aes(x=Design, y=Programming))
p2 + stat_bin2d(bins=8) + scale_fill_gradient2(breaks=c(0,1,2,3,4,5)) +
  ylim(0,6) + xlim(0,6)
```



About you

- For sure, there's a shocking amount of correlation between design & programming

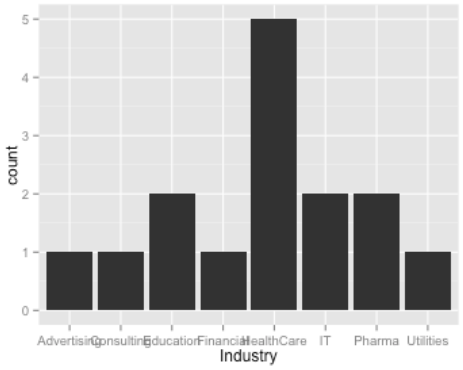
```
model <- lm(Programming ~ Design, data = class)
# summary(model)
pa <- ggplot(class, aes(x=Design, y=Programming))
pa <- pa + geom_point(size=5, position='jitter', alpha=.4) + ylim(0,6) + xlim(0,6)
pa + geom_abline(intercept = 1.48, slope = .655) +
     annotate('text', label= c('R^2 = 0.35'), x = c(5), y=c(1), size=c(5))
```



About you

- And a lot of people in healthcare!

```
p_industry <- ggplot(aes(x=Industry), data=class) + geom_histogram()
p_industry
```



Last Week's Homework

- Let's walk through it
- Gather your Data

```
suppressPackageStartupMessages(library(plyr))
```

```
## Warning: package 'plyr' was built under R version 3.1.3
```

```
inc <- read.csv("inc5000_data.csv", header= TRUE)
head(inc,2)
```

```
##   Rank                 Name Growth_Rate   Revenue
## 1    1                 Fuhu      421.48 117900000
## 2    2 FederalConference.com      248.31  49600000
##                       Industry Employees       City State
## 1 Consumer Products & Services       104 El Segundo    CA
## 2          Government Services        51   Dumfries    VA
```

Last Week's Homework

- Investigate

```
summary(inc[,c(3:6,8)])
```

```
##   Growth_Rate         Revenue                                  Industry
##  Min.   :  0.340   Min.   :2.000e+06   IT Services                   : 733
##  1st Qu.:  0.770   1st Qu.:5.100e+06   Business Products & Services: 482
##  Median :  1.420   Median :1.090e+07   Advertising & Marketing     : 471
##  Mean   :  4.612   Mean   :4.822e+07   Health                       : 355
##  3rd Qu.:  3.290   3rd Qu.:2.860e+07   Software                     : 342
##  Max.   :421.480   Max.   :1.010e+10   Financial Services           : 260
##                                        (Other)                      :2358
##    Employees           State
##  Min.   :    1.0   CA     : 701
##  1st Qu.:   25.0   TX     : 387
##  Median :   53.0   NY     : 311
##  Mean   :  232.7   VA     : 283
##  3rd Qu.:  132.0   FL     : 282
##  Max.   :66803.0   IL     : 273
##  NA's   :12        (Other):2764
```

Last Week's Homework

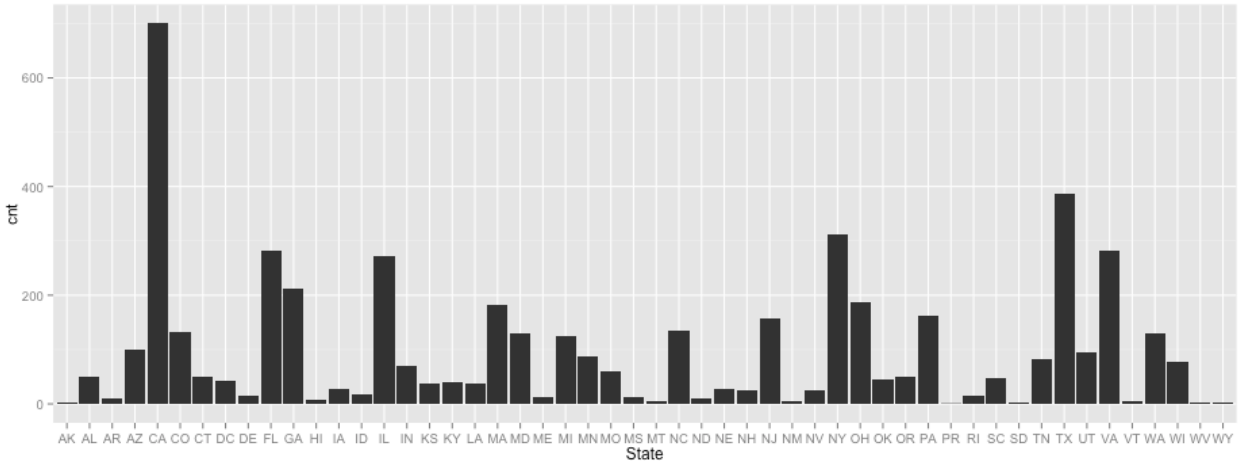
- For this analysis, remove NULL values

```
all_inc <- inc[complete.cases(inc)==TRUE,]
```

Last Week's Homework

- Get counts by State

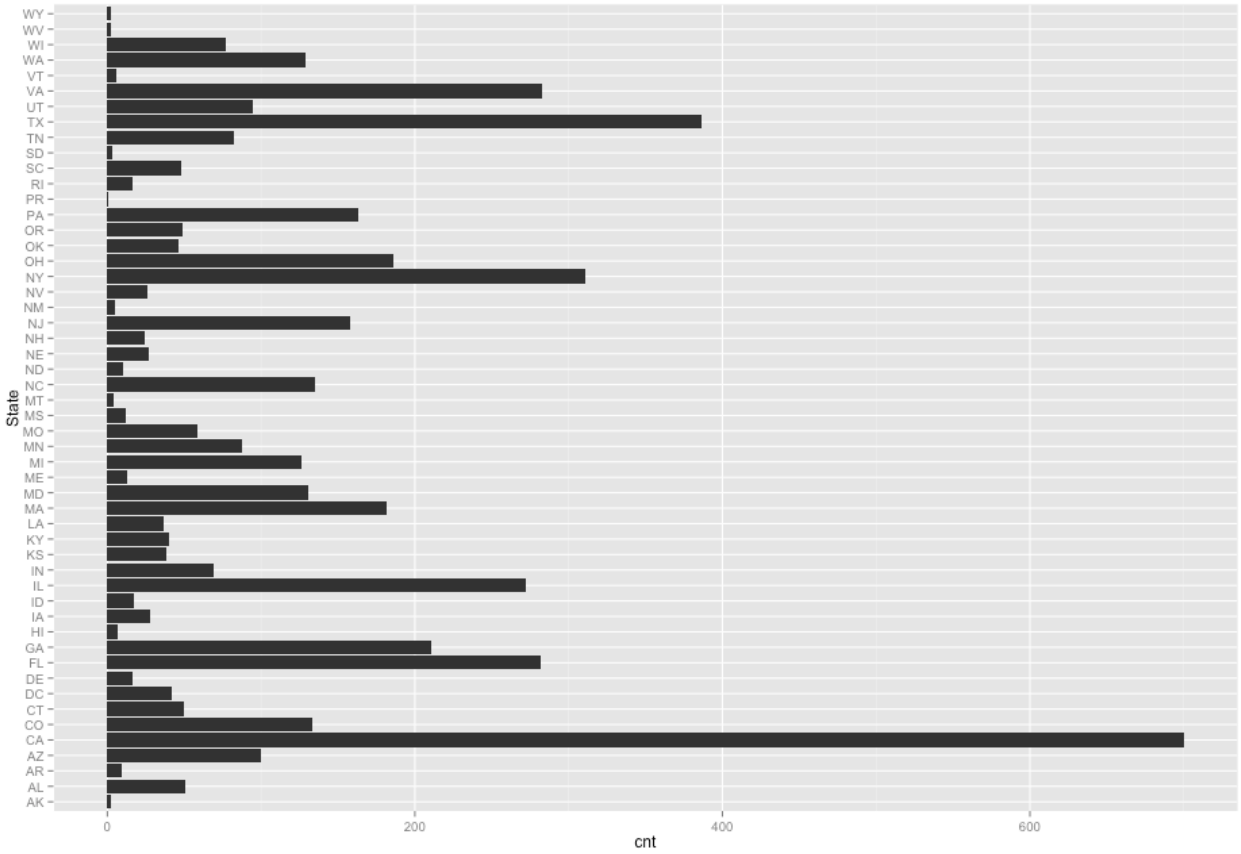
```
cnt <- ddply(all_inc, .(State), summarize, cnt = length(State))
p3 <- ggplot(cnt, aes(x=State, y=cnt)) + geom_bar(stat='identity')
p3
```



Last Week's Homework

- To switch to horizontal bars, use `coord_flip()`
- To show tabular, quantitative data, line or scatter plots are good

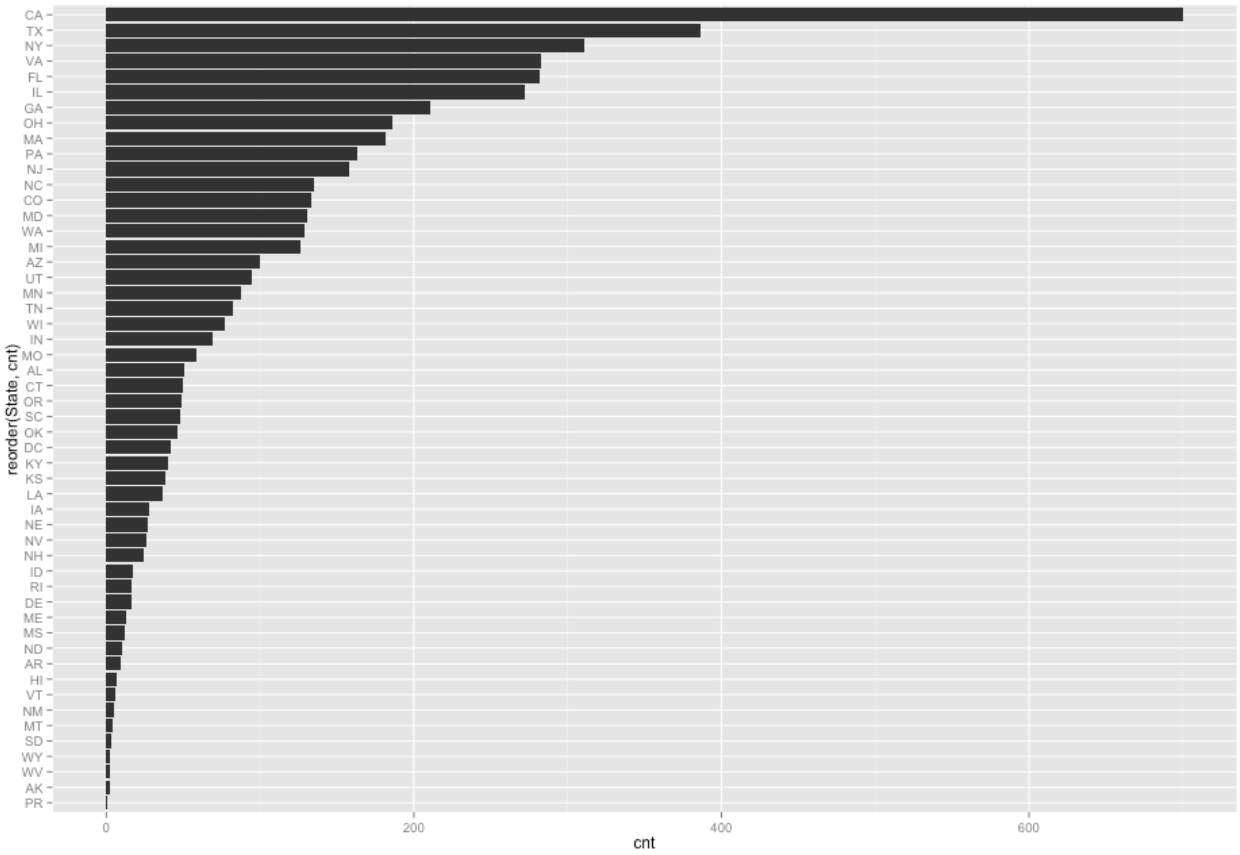
```
p4 <- ggplot(cnt, aes(x=State, y=cnt)) + geom_bar(stat='identity')
p4 + coord_flip()
```



Last Week's Homework

- Can sort using `reorder`

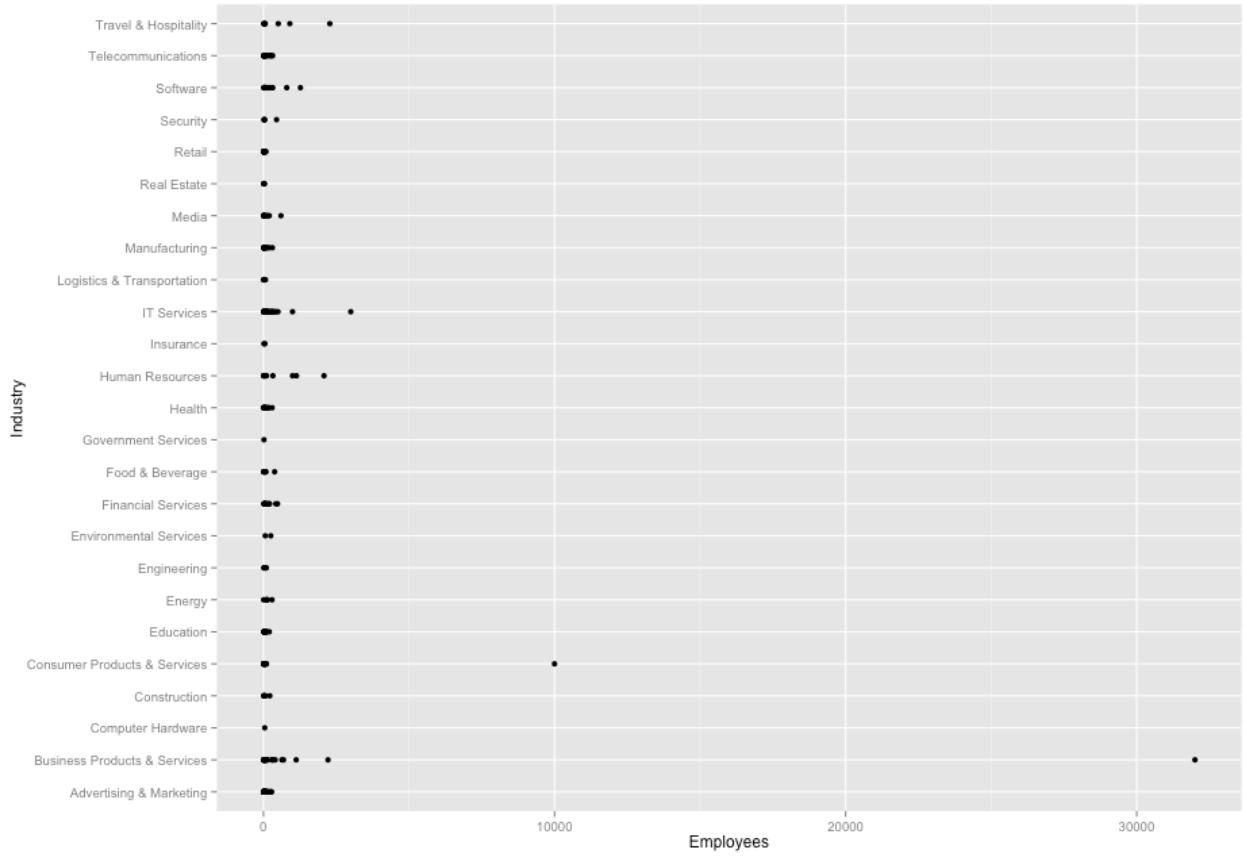
```
p_states <- ggplot(cnt, aes(x=reorder(State,cnt), y=cnt)) + geom_bar(stat='identity')
p_states + coord_flip()
```



Last Week's Homework

- New York is the #3 State, so let's dig in

```
ny <- subset(all_inc, State == 'NY')
p5 <- ggplot(ny, aes(x=Industry, y=Employees)) + geom_point()
p5 + coord_flip()
```



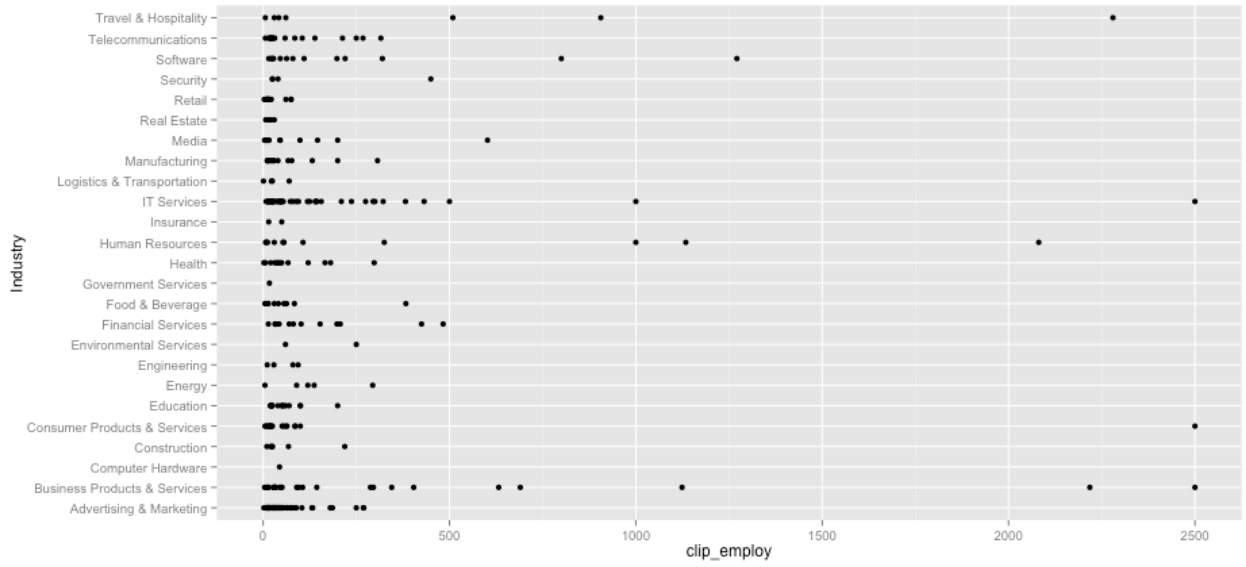
Last Week's Homework

- Serious outlier issue: how do we handle?
- Do we include, make a note (annotate) or ignore?
- Do we care more about the mean or median?
- If we care more about the median, outliers are distractions
- 'Winsorize' Data

```
winsor <- function(x, bot, top)  { return(min(top, max(x, bot))) }
ny$clip_employ <- sapply(ny$Employees, winsor, bot=0, top =2500)
p5 <- ggplot(ny, aes(x=Industry, y=clip_employ))
```

Last Week's Homework

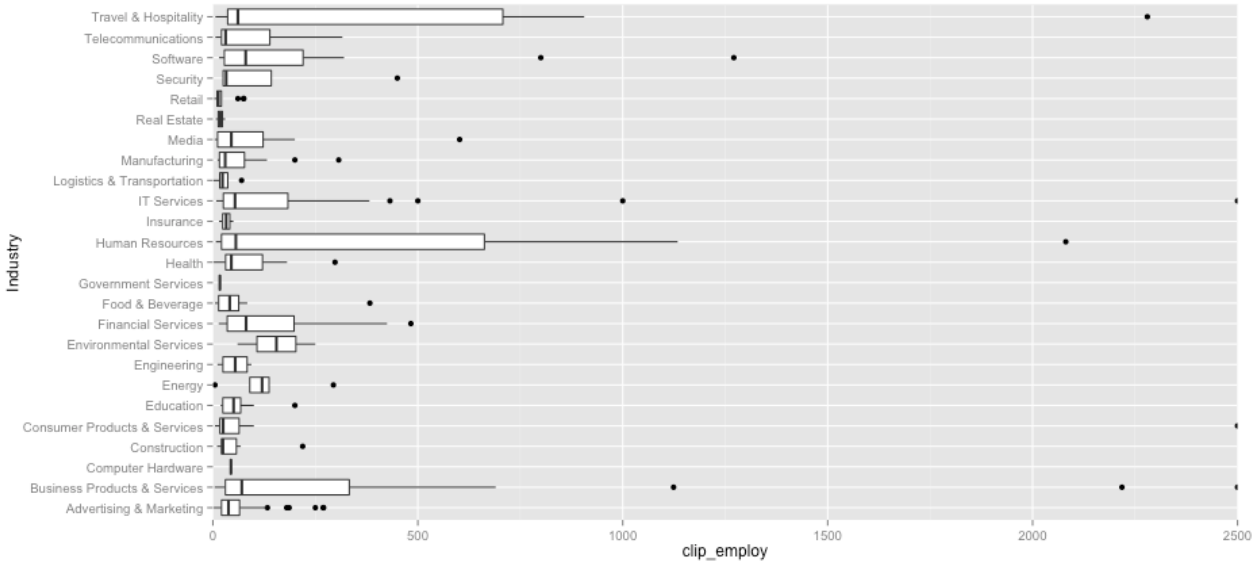
```
p5 + geom_point() + coord_flip()
```



Last Week's Homework

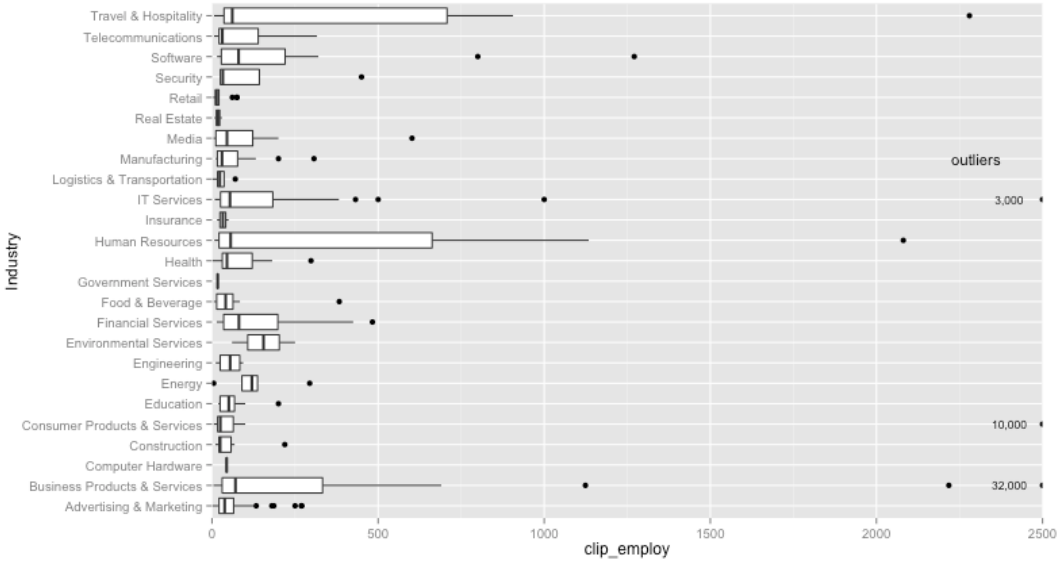
- A relative of the scatter plot is the box plot

```
p5 + geom_boxplot() + coord_flip(ylim=c(0,2500))
```



Last Week's Homework - Marking Outliers

```
p5 + geom_boxplot() + coord_flip(ylim=c(0,2500)) +
annotate('text', label= c('outliers','3,000','10,000','32,000'),
x = c(18,16,5,2), y=c(2300,2400,2400,2400), size=c(4,3,3,3))
```



Last Week's Homework

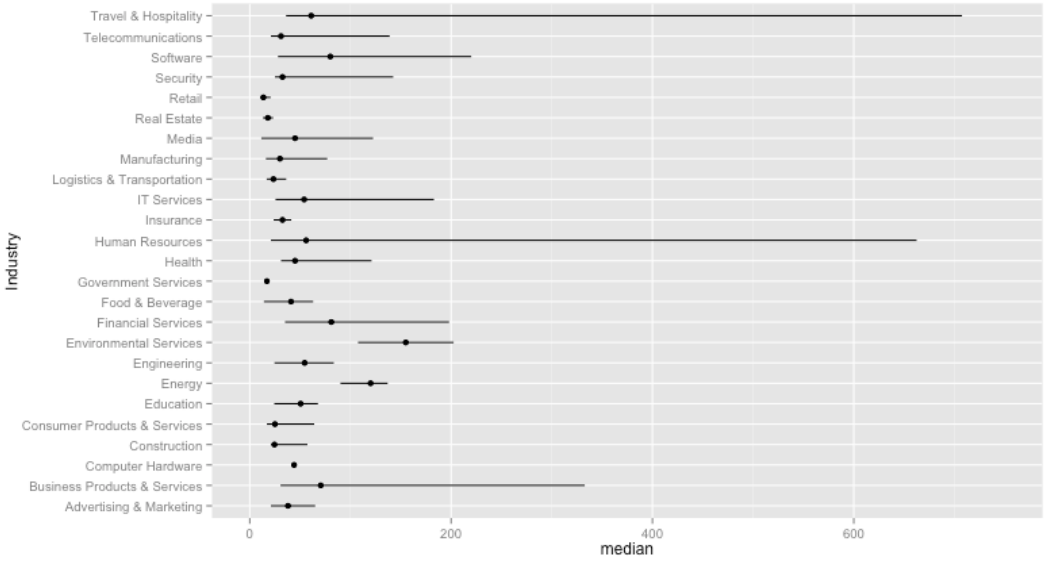
- There are other ways to show variance
- But we need to create averages

```
ny_ave <- ddply(ny, .(Industry), summarize,

                mean <- mean(Employees),
                sd <- sd(Employees),
                median <- median(clip_employ),
                lower <- quantile(clip_employ)[2],
                upper <- quantile(clip_employ)[4]

               )
names(ny_ave) <- c('Industry', 'mean', 'sd', 'median', 'lower', 'upper')

head(ny_ave,2)
```

```
##                       Industry      mean         sd median lower  upper
## 1      Advertising & Marketing   58.4386   62.22971   38.0  21.0  65.00
## 2 Business Products & Services 1492.4615 6240.70574   70.5  30.5 332.75
```

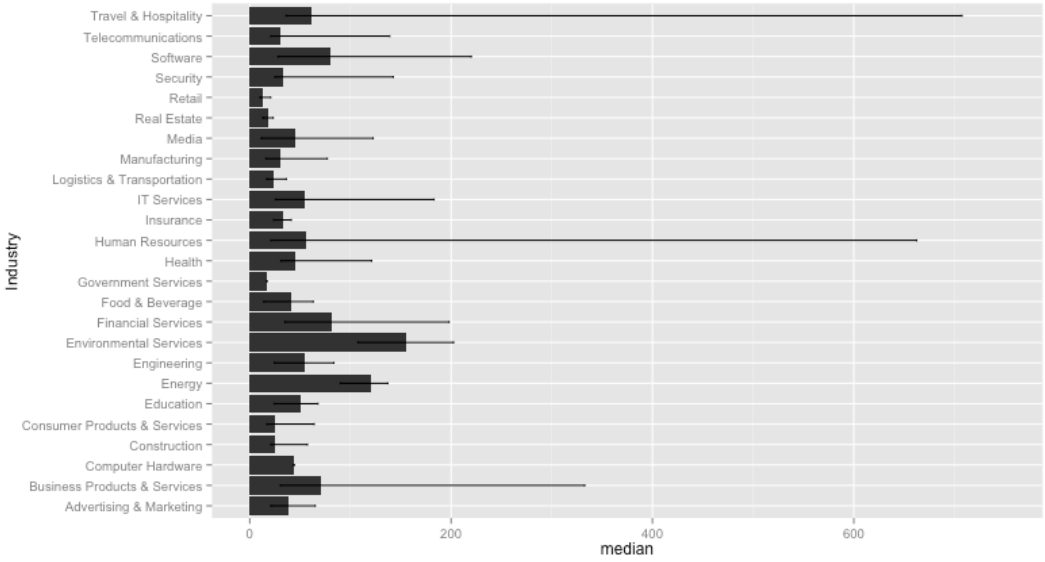
Last Week's Homework - Point ranges

```
p6 <- ggplot(ny_ave, aes(x=Industry, y=median)) + geom_point()
p6 <- p6 + geom_pointrange(ymin=ny_ave$lower, ymax=ny_ave$upper)
p6 + ylim(c(0,750)) + coord_flip()
```



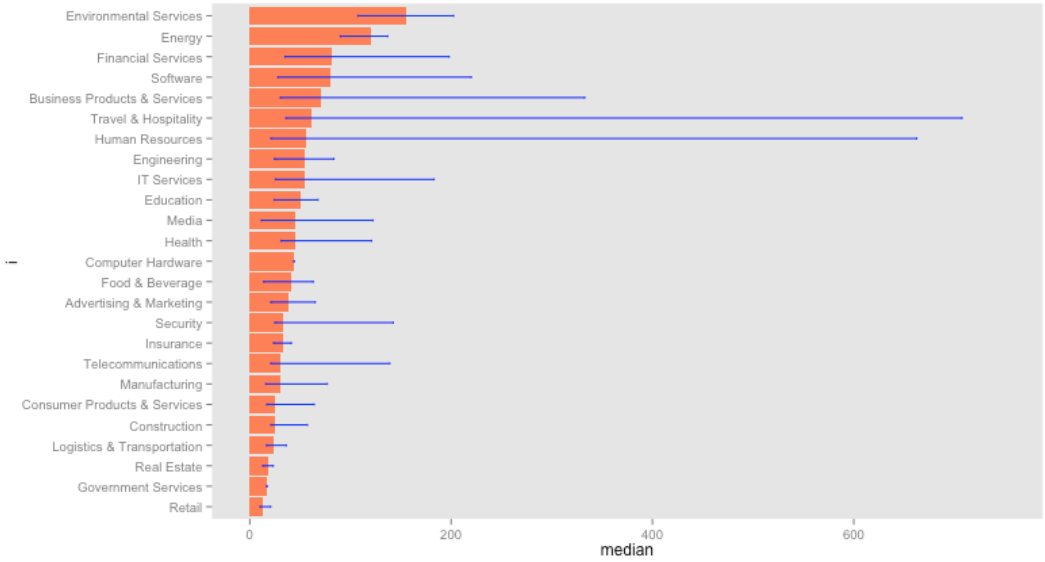
Last Week's Homework - Error bars

```
p7 <- ggplot(ny_ave, aes(x=Industry, y=median)) + geom_bar(stat='identity')
p7 <- p7 + geom_errorbar(ymin=ny_ave$lower, ymax=ny_ave$upper, width=.1)
p7 + ylim(c(0,750)) + coord_flip()
```



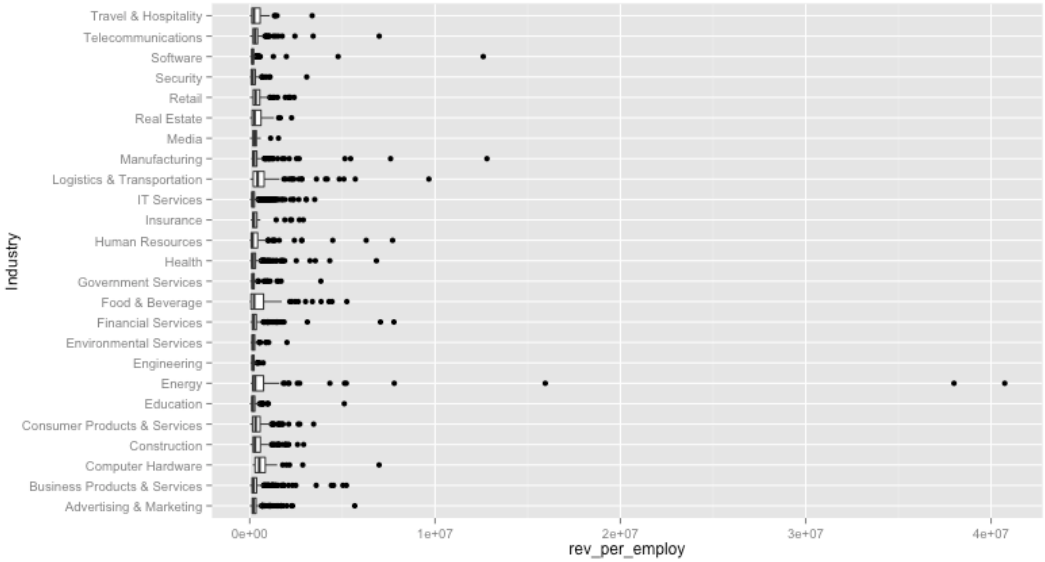
Last Week's Homework - Error bars

```
ny_ave$i = reorder(ny_ave$Industry, ny_ave$median)
p8 <- ggplot(ny_ave, aes(x=i, y=median)) + geom_bar(stat='identity',fill='coral')
p8 <- p8 + geom_errorbar(ymin=ny_ave$lower, ymax=ny_ave$upper, width=.1, color='blue')
p8 + ylim(c(0,750)) + coord_flip() + theme(panel.grid.major = element_blank(), panel.grid.minc
```



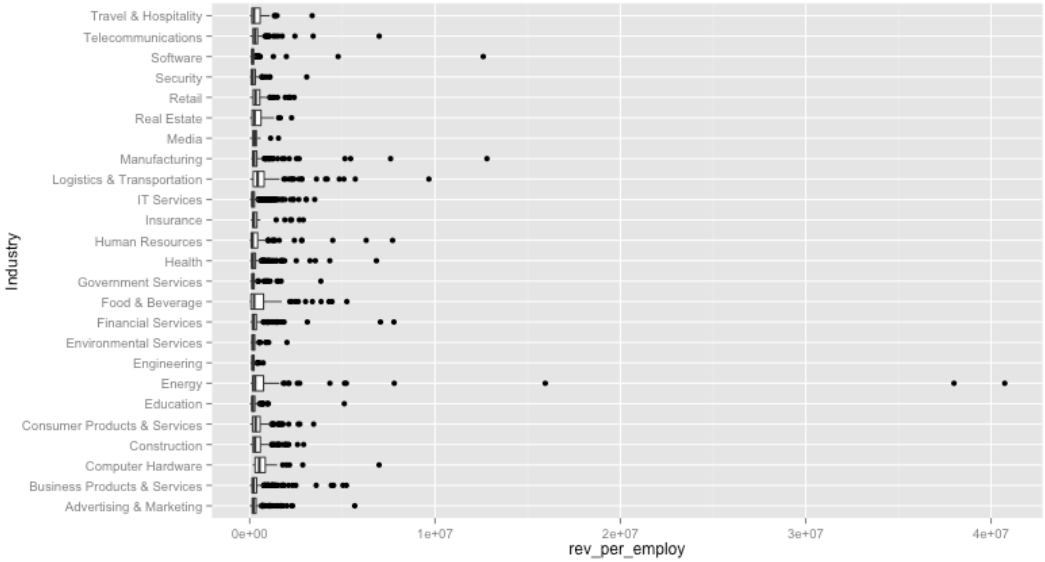
Last Week's Homework - Investors care about the money

```
all_inc$rev_per_employ <- all_inc$Revenue / all_inc$Employees
p9 <- ggplot(all_inc, aes(x=Industry, y=rev_per_employ))
p9 + geom_boxplot() + coord_flip()
```



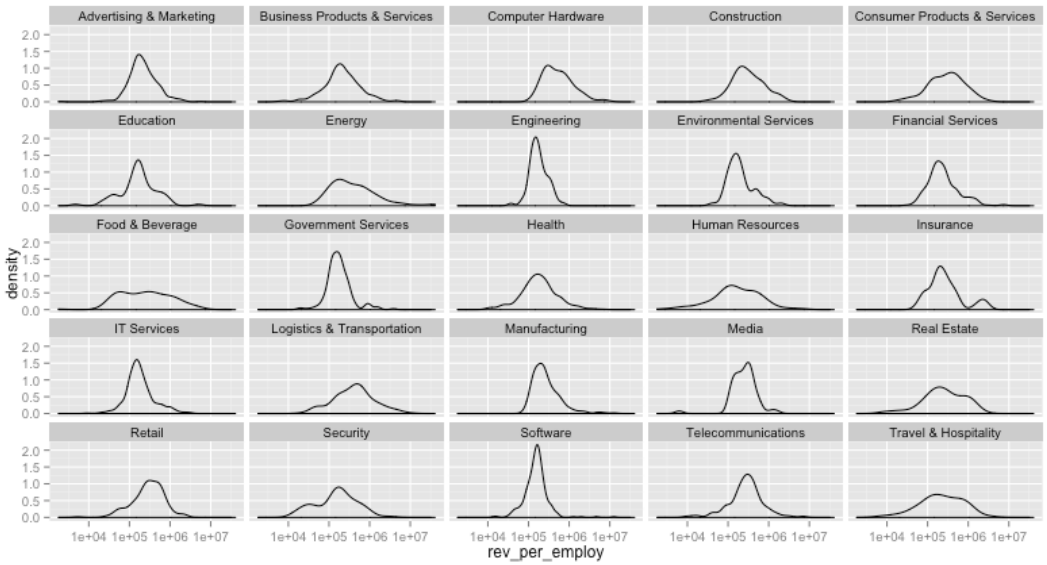
Last Week's Homework - Revenue per Employee

```
all_inc$rev_per_employ <- all_inc$Revenue / all_inc$Employees
p10 <- ggplot(all_inc, aes(x=Industry, y=rev_per_employ))
p10 + geom_boxplot() + coord_flip()
```



Last Week's Homework - Likely Outcomes and Distributions

```
p11 <- ggplot(all_inc,aes(x=rev_per_employ))
p11 <- p11 + geom_density() + facet_wrap(~ Industry)
p11 + scale_x_log10(breaks=c(10000, 100000, 1000000, 10000000))
```



Exploratory Data Analysis

- A great way to test your visualizations - do you find them useful?
- We basically just did it!
- Should always use to understand your data set

ggplot2

- Most popular visualization framework
- Developed by Hadley Wickham
- Easy to learn, supports lots of features
- Being ported to other languages
- We will focus on these design patterns throughout the semester

BigVis

- Also written by Hadley Wickham
- Geared towards larger data sets
- Not on CRAN

devtools

- In order to install BigVis, you need to install devtools
- Go to http://www.rstudio.com/projects/devtools/
- Depending on your operating system, go to the Rtools/Xcode/r-devel page
- Follow the instructions carefully
- Once devtools is installed, follow the directions at https://github.com/hadley/bigvis

This week's homework

- We will be working with the set of all NYC tax lot data
- Go to http://www.nyc.gov/html/dcp/html/bytes/applbyte.shtml#pluto
- Download the PLUTO data set
- The data is in separate files for each boro: you will need to combine

This week's homework - hints

- You don't need every column of data in your combined file
- If you can't combine files, do the homework with Manhattan-only data
- If you can't install devtools/BigVis, try again
- If you can't install devtools/BigVis after an hour, email me
- I will put some sample BigVis code on GitHub to help you get started

That's it

- This presentation will be on the GitHub page for reference
- Good luck! Any questions?