DATA608: Homework 1

Jai Jeffryes

9/6/2020

Contents

Question 1 . . . 6

Pet peeve (note to self) . . . 6

Lessons learned . . . 6

Answer . . . 7

Question 2 . . . 9

Pet peeve . . . 9

Lessons learned . . . 9

Answer . . . 9

Question 3 . . . 12

Lessons learned . . . 12

Answer . . . 12

Afterword . . . 14

Preferred solution . . . 14

Alternate approaches . . . 19

Principles of Data Visualization and Introduction to ggplot2

I have provided you with data about the 5,000 fastest growing companies in the US, as compiled by Inc. magazine. lets read this in:

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

```r
inc <- read.csv("https://raw.githubusercontent.com/charleyferrari/CUNY_DATA_608/master/module1/Data/inc
```

And lets preview this data:

```r
head(inc)
```

```
##   Rank                         Name Growth_Rate   Revenue
## 1    1                         Fuhu      421.48 1.179e+08
## 2    2        FederalConference.com      248.31 4.960e+07
## 3    3               The HCI Group      245.45 2.550e+07
## 4    4                      Bridger      233.08 1.900e+09
## 5    5                       DataXu      213.37 8.700e+07
## 6    6 MileStone Community Builders      179.38 4.570e+07
##                       Industry Employees         City State
## 1 Consumer Products & Services       104   El Segundo    CA
## 2          Government Services        51     Dumfries    VA
## 3                       Health       132 Jacksonville    FL
## 4                       Energy        50      Addison    TX
## 5      Advertising & Marketing       220       Boston    MA
## 6                  Real Estate        63       Austin    TX
```

```r
summary(inc)
```

```
##       Rank                          Name       Growth_Rate
##  Min.   :   1   (Add)ventures         :   1   Min.   :  0.340
##  1st Qu.:1252   @Properties           :   1   1st Qu.:  0.770
##  Median :2502   1-Stop Translation USA:   1   Median :  1.420
##  Mean   :2502   110 Consulting        :   1   Mean   :  4.612
##  3rd Qu.:3751   11thStreetCoffee.com  :   1   3rd Qu.:  3.290
##  Max.   :5000   123 Exteriors         :   1   Max.   :421.480
##                 (Other)               :4995
##     Revenue                                     Industry      Employees
##  Min.   :2.000e+06   IT Services                 : 733   Min.   :    1.0
##  1st Qu.:5.100e+06   Business Products & Services: 482   1st Qu.:   25.0
##  Median :1.090e+07   Advertising & Marketing     : 471   Median :   53.0
##  Mean   :4.822e+07   Health                      : 355   Mean   :  232.7
##  3rd Qu.:2.860e+07   Software                    : 342   3rd Qu.:  132.0
##  Max.   :1.010e+10   Financial Services          : 260   Max.   :66803.0
##                      (Other)                     :2358   NA's   :12
##             City          State
##  New York     : 160   CA     : 701
##  Chicago      :  90   TX     : 387
##  Austin       :  88   NY     : 311
##  Houston      :  76   VA     : 283
##  San Francisco:  75   FL     : 282
##  Atlanta      :  74   IL     : 273
##  (Other)      :4438   (Other):2764
```

Think a bit on what these summaries mean. Use the space below to add some more relevant non-visual exploratory information you think helps you understand this data:

```r
# Insert your code here, create more chunks as necessary
# Number of industries
length(unique(inc$Industry))
```

```
## [1] 25
```

```r
# Frequency table of industries
table(inc$Industry)
```

```
## 
##      Advertising & Marketing Business Products & Services 
##                          471                          482 
##            Computer Hardware                 Construction 
##                           44                          187 
## Consumer Products & Services                    Education 
##                          203                           83 
##                       Energy                  Engineering 
##                          109                           74 
##       Environmental Services           Financial Services 
##                           51                          260 
##              Food & Beverage          Government Services 
##                          131                          202 
##                       Health              Human Resources 
##                          355                          196 
##                    Insurance                  IT Services 
##                           50                          733 
##   Logistics & Transportation                Manufacturing 
##                          155                          256 
##                        Media                  Real Estate 
##                           54                           96 
##                       Retail                     Security 
##                          203                           73 
##                     Software           Telecommunications 
##                          342                          129 
##         Travel & Hospitality 
##                           62
```

```r
# Top revenue by industry
inc %>%
    group_by(Industry) %>%
    summarise(Tot_Revenue = sum(Revenue)) %>%
    arrange(desc(Tot_Revenue))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 25 x 2
##    Industry                     Tot_Revenue
##    <fct>                              <dbl>
##  1 Business Products & Services 26367900000
##  2 IT Services                  20681300000
##  3 Health                       17863400000
##  4 Consumer Products & Services 14956400000
##  5 Logistics & Transportation   14840500000
##  6 Energy                       13771600000
##  7 Construction                 13174300000
##  8 Financial Services           13150900000
##  9 Food & Beverage              12911300000
## 10 Manufacturing                12684000000
## # ... with 15 more rows
```

```r
# Top growth rate by industry
inc %>%
    group_by(Industry) %>%
    summarise(mean_Growth_Rate = mean(Growth_Rate)) %>%
    arrange(desc(mean_Growth_Rate))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 25 x 2
##    Industry                     mean_Growth_Rate
##    <fct>                                   <dbl>
##  1 Energy                                   9.60
##  2 Consumer Products & Services             8.78
##  3 Real Estate                              7.75
##  4 Government Services                      7.24
##  5 Advertising & Marketing                  6.23
##  6 Retail                                   6.18
##  7 Financial Services                       5.44
##  8 Software                                 5.02
##  9 Health                                   4.86
## 10 Media                                    4.37
## # ... with 15 more rows
```

```r
# Top growth rate by industry
inc %>%
    group_by(Industry) %>%
    summarise(mean_Growth_Rate = mean(Growth_Rate)) %>%
    arrange(desc(mean_Growth_Rate))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 25 x 2
##    Industry                     mean_Growth_Rate
##    <fct>                                   <dbl>
##  1 Energy                                   9.60
##  2 Consumer Products & Services             8.78
##  3 Real Estate                              7.75
##  4 Government Services                      7.24
##  5 Advertising & Marketing                  6.23
##  6 Retail                                   6.18
##  7 Financial Services                       5.44
##  8 Software                                 5.02
##  9 Health                                   4.86
## 10 Media                                    4.37
## # ... with 15 more rows
```

```r
# Top revenue by city
inc %>%
    group_by(City, State) %>%
    summarise(Tot_Revenue = sum(Revenue)) %>%
    arrange(desc(Tot_Revenue))
```

```
## `summarise()` regrouping output by 'City' (override with `.groups` argument)
```

```
## # A tibble: 1,654 x 3
## # Groups:   City [1,519]
##    City         State Tot_Revenue
##    <fct>        <fct>       <dbl>
##  1 New York     NY    10500800000
##  2 Vernon Hills IL    10106100000
##  3 Chicago      IL     8737100000
##  4 Houston      TX     6553100000
##  5 Beloit       WI     4700000000
##  6 Mt. Sterling IL     4500000000
##  7 Cincinnati   OH     4459900000
##  8 Tarrytown    NY     3804600000
##  9 Huntersville NC     3516800000
## 10 Washington   DC     3268800000
## # ... with 1,644 more rows
```

```r
# Top growth rate by city
inc %>%
    group_by(City, State) %>%
    summarise(mean_Growth_Rate = mean(Growth_Rate)) %>%
    arrange(desc(mean_Growth_Rate))
```

```
## `summarise()` regrouping output by 'City' (override with `.groups` argument)
```

```
## # A tibble: 1,654 x 3
## # Groups:   City [1,519]
##    City          State mean_Growth_Rate
##    <fct>         <fct>            <dbl>
##  1 Dumfries      VA               248.
##  2 Chino         CA               111.
##  3 columbus      OH               100.
##  4 Cupertino     CA                92.4
##  5 Bluffdale     UT                59.9
##  6 El Segundo    CA                56.2
##  7 Rock Hill     NY                53.6
##  8 Rochelle Park NJ                53.4
##  9 Saugus        MA                46.5
## 10 Warwick       RI                44.5
## # ... with 1,644 more rows
```

```r
# Top productivity
inc_revenue_per_employee <- inc %>%
    select(Name, Revenue, Employees) %>%
    mutate(Revenue_per_Employee = Revenue / Employees) %>%
    arrange(desc(Revenue_per_Employee))

inc_revenue_per_employee[1:10, c(1,4)]
```

```
##                             Name Revenue_per_Employee
## 1          Cedar Petrochemcials             40740000
## 2                        Bridger             38000000
## 3  Hightowers Petroleum Company             15947368
## 4                    Fast Fusion             12800000
## 5                         NeoGov             12600000
## 6              Intelligent Audit              9666667
## 7             Advanced BioEnergy              7797333
## 8                        P-Fleet              7778571
## 9    Pivot Employment Platforms              7707692
## 10                Apex Resources              7600000
```

```
# Note to self: You don't have to subset the columns,
# just move the select below the mutation and soring.
inc_revenue_per_employee <- inc %>%
    mutate(Revenue_per_Employee = Revenue / Employees) %>%
    arrange(desc(Revenue_per_Employee)) %>%
    select(Name, Revenue_per_Employee)
```

Question 1

Create a graph that shows the distribution of companies in the dataset by State (ie how many are in each state). There are a lot of States, so consider which axis you should use. This visualization is ultimately going to be consumed on a ‘portrait’ oriented screen (ie taller than wide), which should further guide your layout choices.

Pet peeve (note to self)

These are the things about R that yank my chain. It shouldn’t be this hard to change the order of a plot. You can’t just tell the aesthetic function to reorder the column State. Instead, you have to dive into arcane discussions on StackOverflow about peculiarities of flipping coordinates. Then you have to stand on your head with commands that are not at all intuitive in order to hack the output you want. But it’s in my code now, and `grep` is my friend. When I need this trick again six years from now, I’ll remember I said this yanked my chain and I can find it.

References

- ggplot2 - sorting a plot. See MatteoS’s answer about `xlim(rev(levels()))`.
- Also, this page: Order of legend entries in ggplot2 barplts with coord flip.

Lessons learned

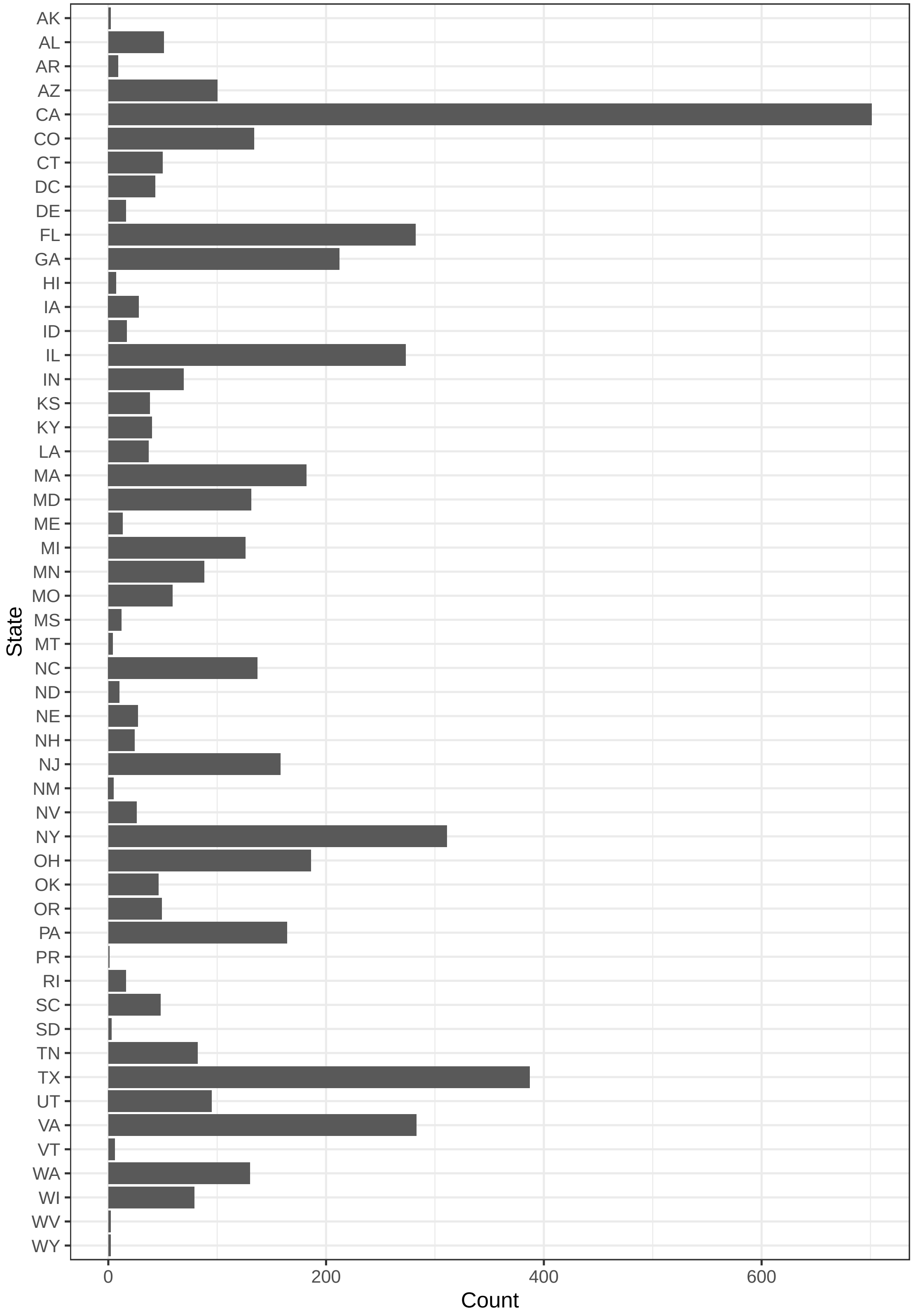
- `coord_flip()` orders bottom to top, that’s all. It would be great if `coord_flip()` had a parameter for controlling that. Since it doesn’t, you have to hack the x axis with `xlim` and then `rev()` (base `R`) or `sort(desc = T)`.
- Control the dimension of the plot, not in the plot, but in the target presentation; here, `knitr`. The bars are too narrow by default. Don’t think “thicken the bars,” think “resize the output figure.” Do that in the chunk options.

Answer

```r
# Answer Question 1 here
ggplot(inc) +
    aes(x = State) +
    geom_bar(stat = "count") +
    coord_flip() +
    xlim(sort(levels(inc$State), decreasing = T)) +
    theme_bw() +
    labs(title = "5,000 Fastest Growing Companies in the U.S.",
         subtitle = "Number of Companies by State",
         x = "State",
         y = "Count",
         caption = "Source: Inc. Magazine")
```

5,000 Fastest Growing Companies in the U.S.

Number of Companies by State



Source: Inc. Magazine

Question 2

Lets dig in on the state with the 3rd most companies in the data set. Imagine you work for the state and are interested in how many people are employed by companies in different industries. Create a plot that shows the average and/or median employment by industry for companies in this state (only use cases with full data, use R's `complete.cases()` function.) In addition to this, your graph should show how variable the ranges are, and you should deal with outliers.

Pet peeve

- I dislike mixing query approaches. Base R or `dplyr`, not both. Filtering should happen in one place. If I'm going to use `dplyr`, I don't want `complete_cases()` outside of the query. I wonder how I would do that.

Lessons learned

- My SQL head tends to forget I can code `select()` after other grouping and filtering, which allows me to return only the variables I want while not making the code above complain that it is missing dependant columns.
- Two ways to pick off specific rows.
  - `filter()` with `row_number()`. Knew that, but had to look it up.
  - `slice()`. New one on me.
- Two ways to count within groups.
  - `group_by()` with `summarise(n = n())`. Knew that.
  - `count()`. New one. Counts by group according to variables in args. Parameter `sort = t` orders descending.
- `GROUP BY/HAVING`: dplyr and SQL
- Outliers. You can omit them from boxplots with a parameter. I found the reference to that when I was working on the third question. I moved all of my earlier solutions to Question 2 down to the bottom in order to keep the code.

Answer

The question pertains to central tendancy. Therefore, inclusion of outliers is not germane and are omitted from the boxplots.

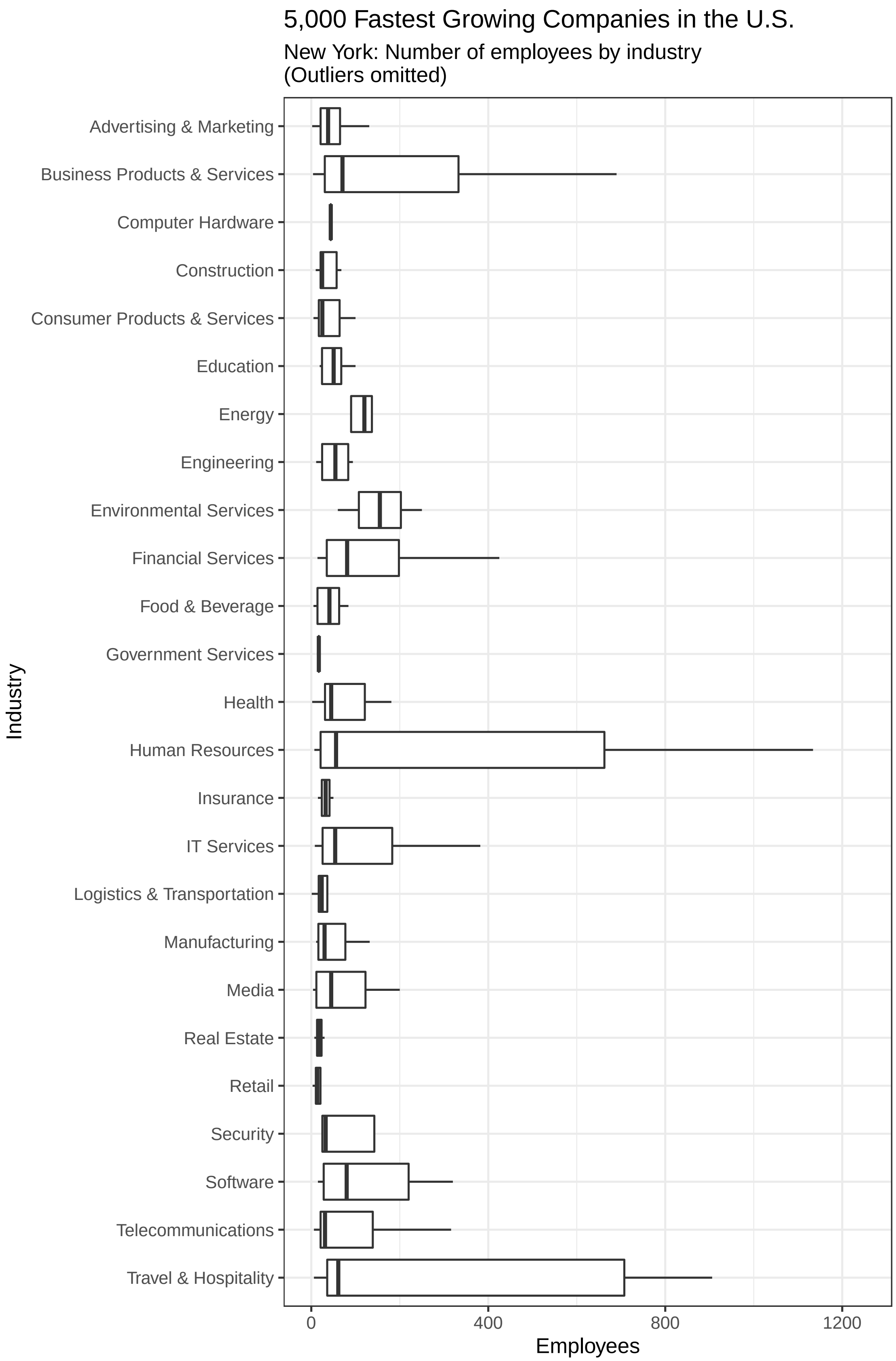
```r
# Answer Question 2 here
# Identify the 3rd largest state.
State_3 <- inc %>%
  group_by(State) %>%
  summarise(n = n()) %>%
  arrange(desc(n)) %>%
  filter(row_number() == 3) %>%
  select(State) # It returns as a tibble, so convert the column.
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
State_3 <- State_3$State

# Pick off complete data for 3rd largest state.
inc_state3 <- inc[complete.cases(inc), ] %>%
  filter(State == State_3)

# Boxplot each industry, omitting outliers.
inc_state3 %>%
  ggplot(aes(x = Industry, y = Employees)) +
  geom_boxplot(outlier.shape = NA) +
  coord_flip(ylim = c(0, 1250)) +
  xlim(sort(levels(inc_state3$Industry), decreasing = T)) +
    theme_bw() +
    labs(title = "5,000 Fastest Growing Companies in the U.S.",
         subtitle = "New York: Number of employees by industry\n(Outliers omitted)",
         caption = "Source: Inc. Magazine")
```



Source: Inc. Magazine

Question 3

Now imagine you work for an investor and want to see which industries generate the most revenue per employee. Create a chart that makes this information clear. Once again, the distribution per industry should be shown.

Lessons learned

- Ignore outliers in ggplot2 boxplot in R.

Answer

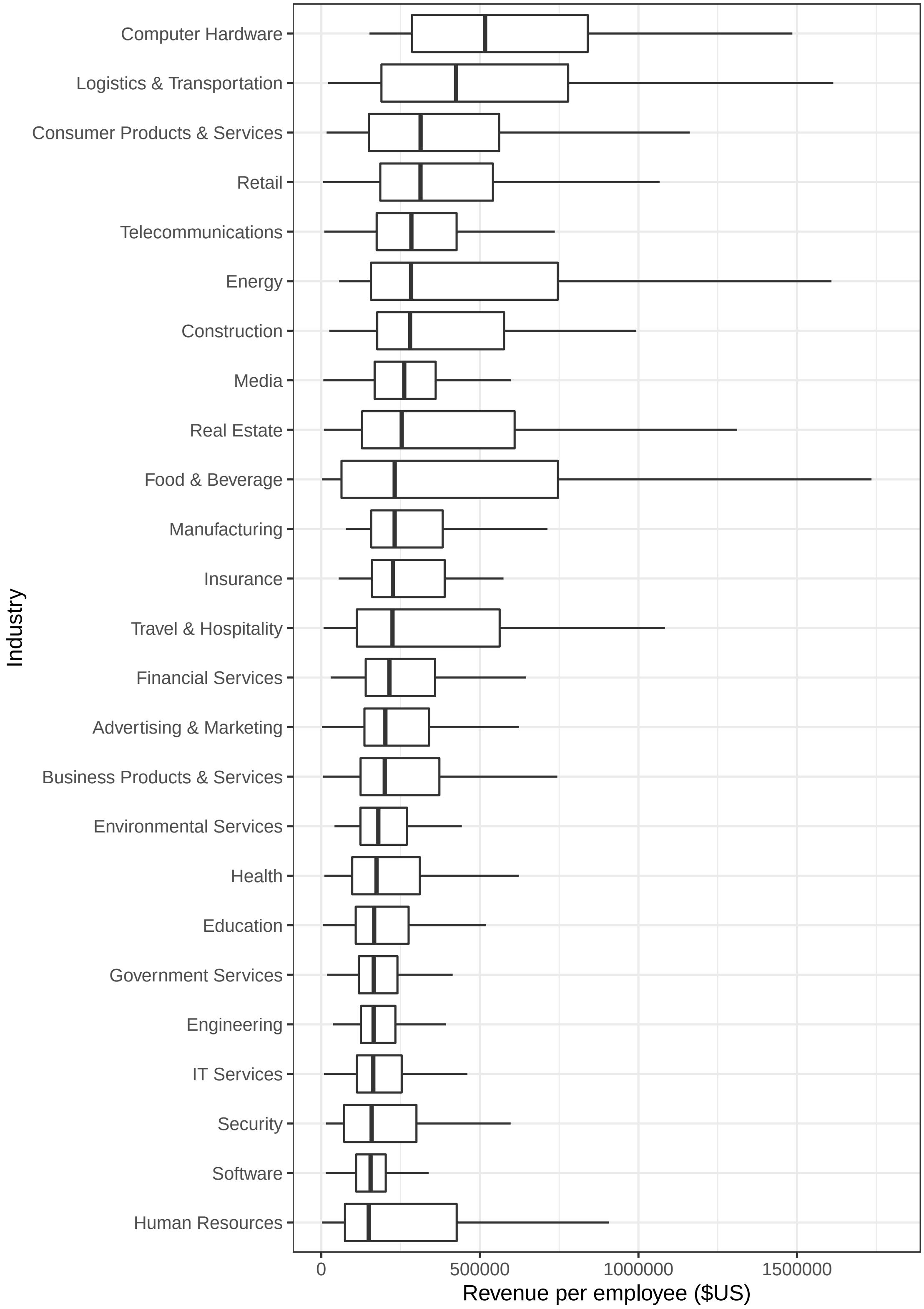
```r
# Answer Question 3 here
# This was one of the questions from my own EDA at the top.

inc_productivity <- inc[complete.cases(inc), ] %>%
    mutate(Revenue_per_Employee = Revenue / Employees)

ggplot(inc_productivity) +
  aes(x = reorder(Industry, Revenue_per_Employee, FUN = median), y = Revenue_per_Employee) +
  geom_boxplot(outlier.shape = NA) +
  coord_flip(ylim = c(0, 1800000)) +
  theme_bw() +
  labs(title = "5,000 Fastest Growing Companies in the U.S.",
       subtitle = "Revenue per employee by Industry\n(Outliers omitted)",
       caption = "Source: Inc. Magazine",
       x = "Industry",
       y = "Revenue per employee ($US)")
```

5,000 Fastest Growing Companies in the U.S.

Revenue per employee by Industry
(Outliers omitted)



Source: Inc. Magazine

Afterword

These are approaches I discarded, but I'm keeping the code around as work papers for my learning.

Preferred solution

Extreme outliers squash the interquartile ranges too much to be discernable. In this plot, the company with the highest number of employees within each industry is omitted. We can still interpret the variability from the IQR and whiskers, and we can see the medians.

Maximum outlier omitted

```r
# Answer Question 2 here
# Identify the 3rd largest state.
State_3 <- inc %>%
  group_by(State) %>%
  summarise(n = n()) %>%
  arrange(desc(n)) %>%
  filter(row_number() == 3) %>%
  select(State) # It returns as a tibble, so convert the column.
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
State_3 <- State_3$State

# Alternate queries (unused)
dummy <- inc %>%
  group_by(State) %>%
  summarise(n = n()) %>%
  arrange(desc(n)) %>%
  slice(3) %>%
  select(State)
```

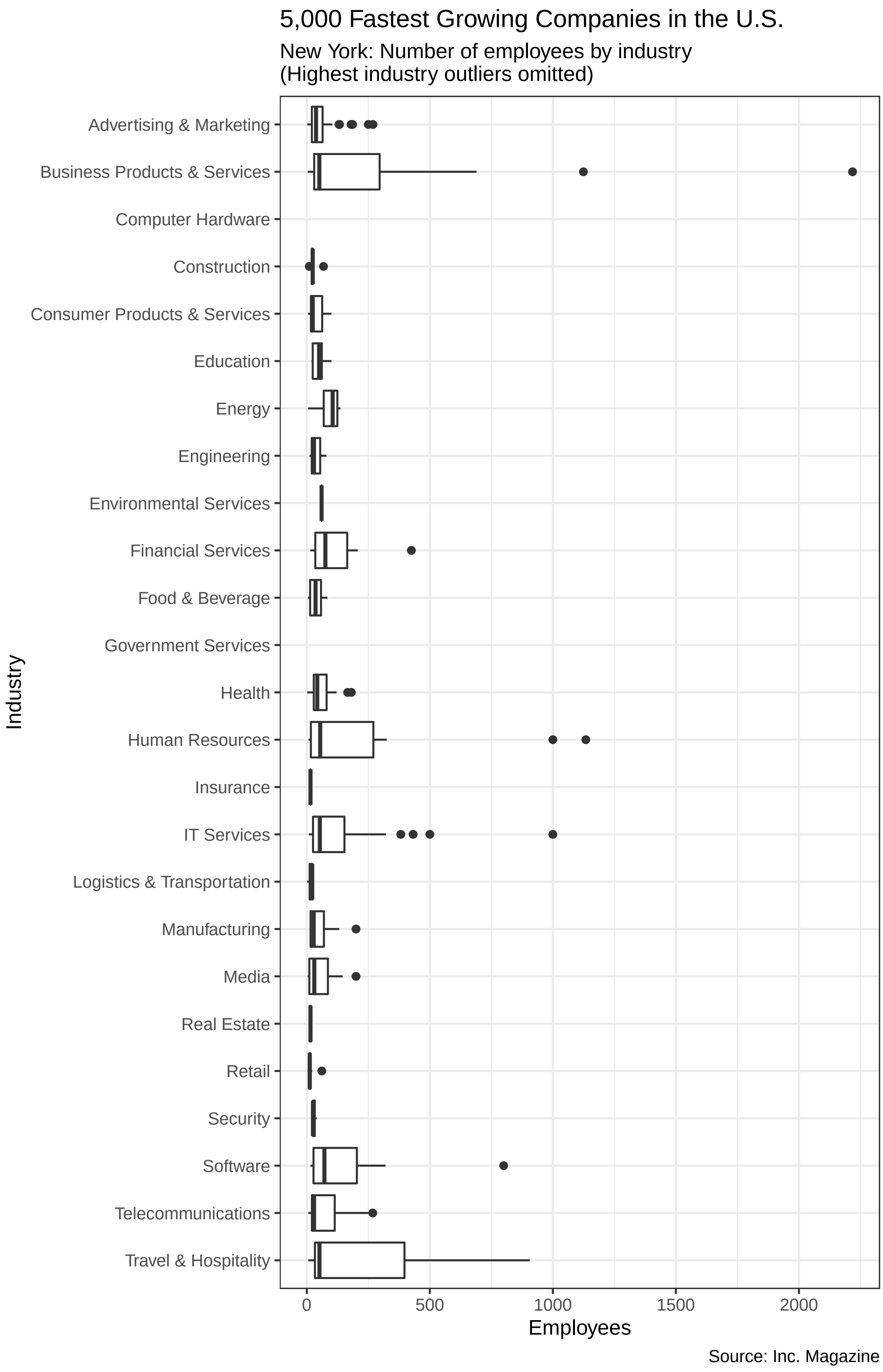
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
dummy <- inc %>%
  count(State, sort = T)

# Pick off complete data for 3rd largest state.
inc_state3 <- inc[complete.cases(inc), ] %>%
  filter(State == State_3)

# Identify highest outlier in each industry
inc_state3_max_emp <- inc_state3 %>%
  group_by(Industry) %>%
  filter(Employees == max(Employees))

# Boxplot each industry, omitting its highest outlier.
inc_state3 %>%
  filter(!(Name %in% inc_state3_max_emp$Name)) %>%
  ggplot(aes(x = Industry, y = Employees)) +
  geom_boxplot() +
  coord_flip() +
  xlim(sort(levels(inc_state3$Industry), decreasing = T)) +
    theme_bw() +
    labs(title = "5,000 Fastest Growing Companies in the U.S.",
         subtitle = "New York: Number of employees by industry\n(Highest industry outliers omitted)",
         caption = "Source: Inc. Magazine")
```


5,000 Fastest Growing Companies in the U.S.
New York: Number of employees by industry
(Highest industry outliers omitted)
Industry
Advertising & Marketing
Business Products & Services
Computer Hardware
Construction
Consumer Products & Services
Education
Energy
Engineering
Environmental Services
Financial Services
Food & Beverage
Government Services
Health
Human Resources
Insurance
IT Services
Logistics & Transportation
Manufacturing
Media
Real Estate
Retail
Security
Software
Telecommunications
Travel & Hospitality
0
500
1000
1500
2000
Employees
Source: Inc. Magazine

When I look at that chart, I would like to examine further the differences between the industries with fewer than 500 employees.

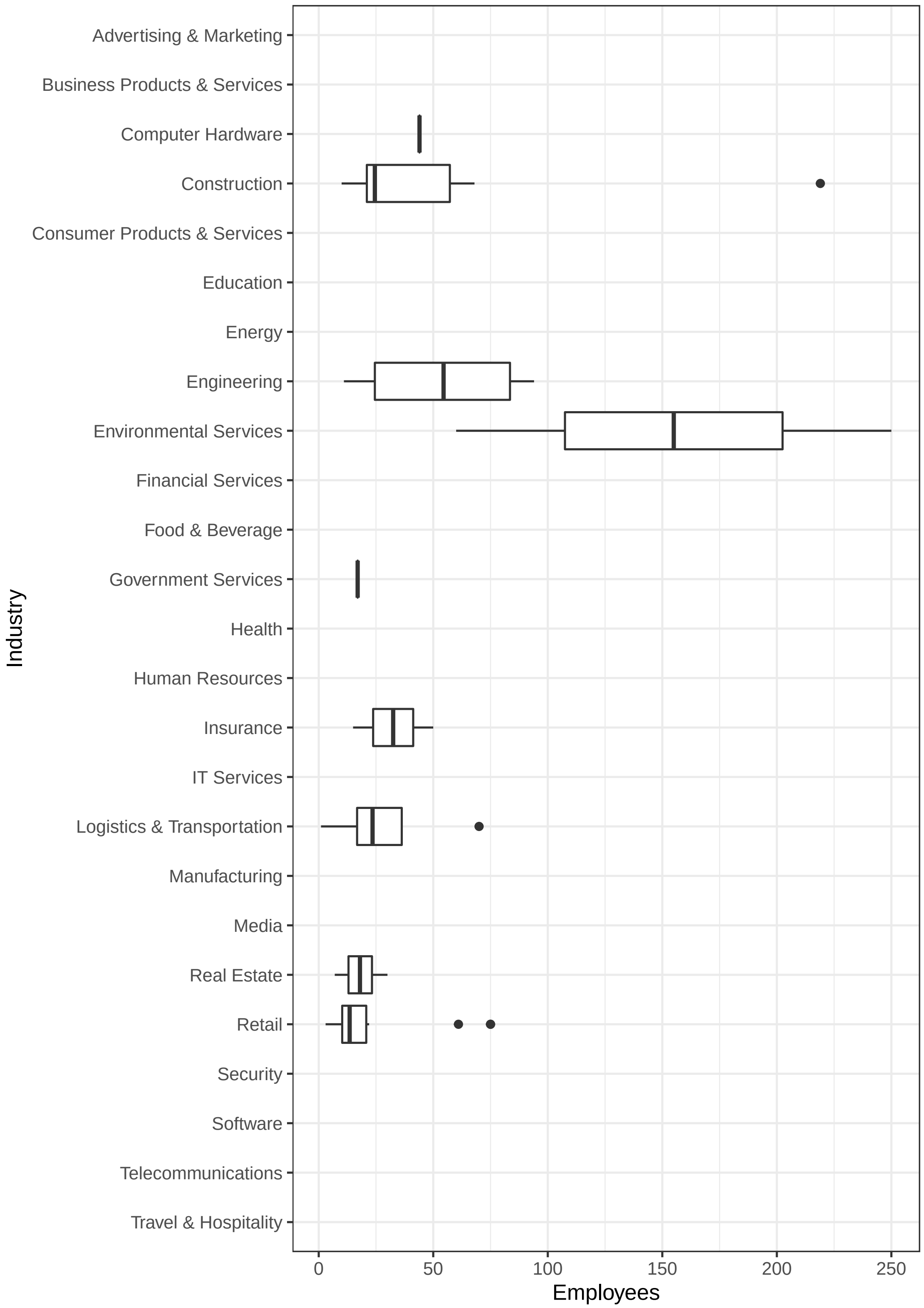
```r
# Lowest outlier companies.
industry_low_outliers <- inc_state3 %>%
  group_by(Industry) %>%
  summarize(emp_count = sum(Employees)) %>%
  filter(emp_count < 500) %>%
  select(Industry)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
# Boxplot each industry whose employees total no more than 500.
inc_state3 %>%
  filter(Industry %in% industry_low_outliers$Industry) %>%
  ggplot(aes(x = Industry, y = Employees)) +
  geom_boxplot() +
  coord_flip() +
  xlim(sort(levels(inc_state3$Industry), decreasing = T)) +
    theme_bw() +
    labs(title = "5,000 Fastest Growing Companies in the U.S.",
         subtitle = "New York: Number of employees by industry\n(Industries with 500 employees or less)
         caption = "Source: Inc. Magazine")
```

5,000 Fastest Growing Companies in the U.S.

New York: Number of employees by industry
(Industries with 500 employees or less)



Source: Inc. Magazine

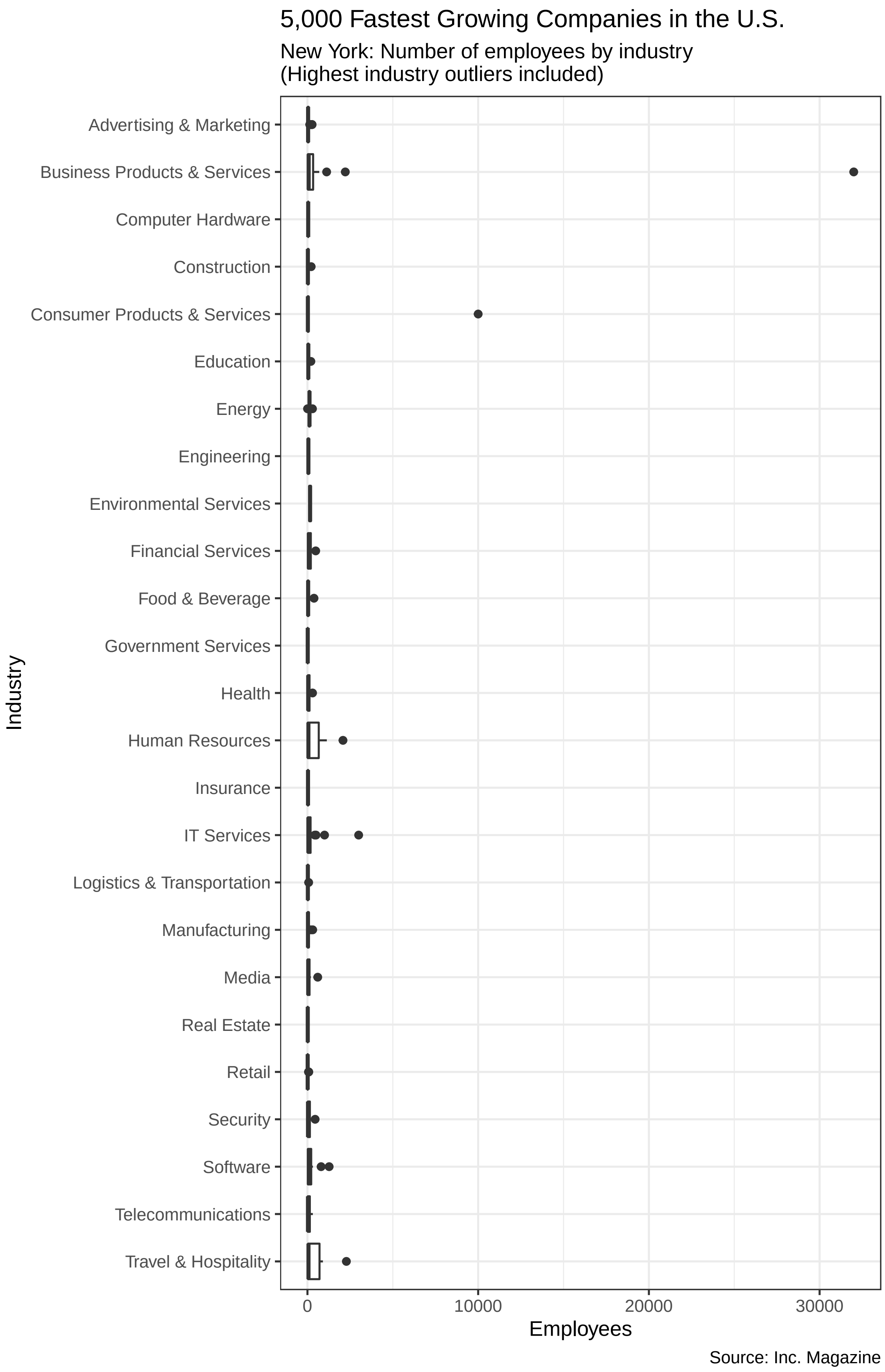
Alternate approaches

The question pertains to central tendancy. Therefore, inclusion of outliers is not germane and I would omit any of these other approaches from an actual presentation. However, examination of outliers is possible.

Here is the same approach as above, with outliers included. It squashes the IQRs, but you can see the relation of the outliers to the central tendency.

Maximum outlier included

```r
# Boxplot each industry, including the highest outlier.
inc_state3 %>%
  ggplot(aes(x = Industry, y = Employees)) +
  geom_boxplot() +
  coord_flip() +
  xlim(sort(levels(inc_state3$Industry), decreasing = T)) +
    theme_bw() +
    labs(title = "5,000 Fastest Growing Companies in the U.S.",
         subtitle = "New York: Number of employees by industry\n(Highest industry outliers included)",
         caption = "Source: Inc. Magazine")
```

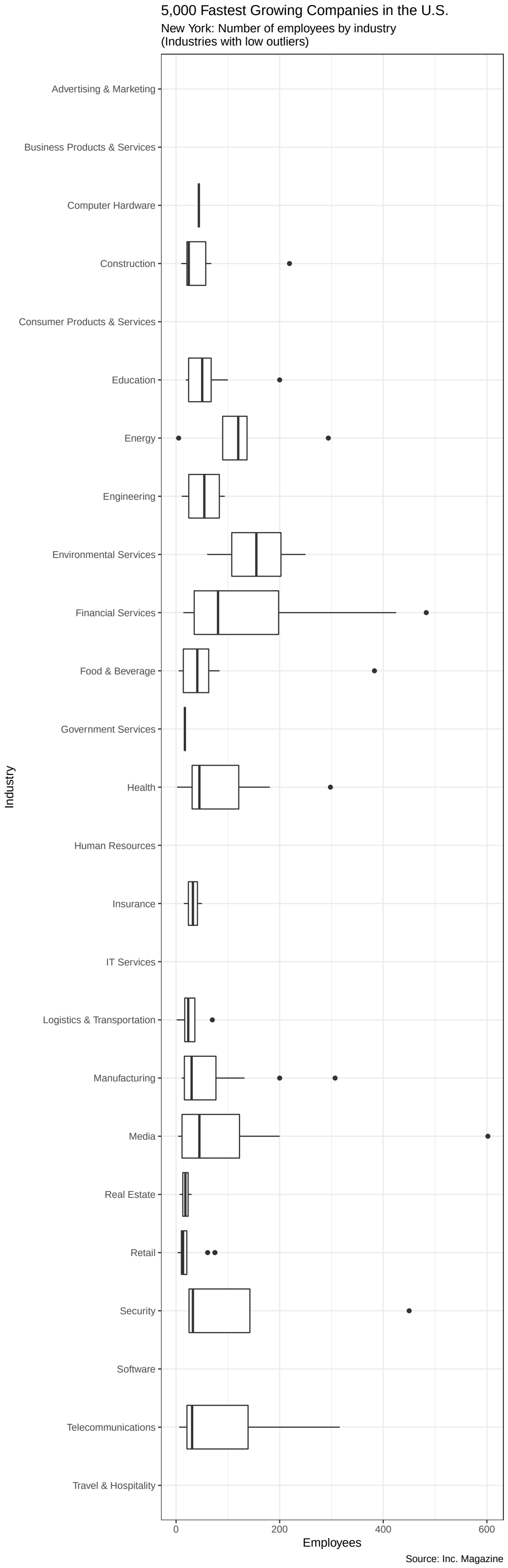

5,000 Fastest Growing Companies in the U.S.
New York: Number of employees by industry
(Highest industry outliers included)
Advertising & Marketing
Business Products & Services
Computer Hardware
Construction
Consumer Products & Services
Education
Energy
Engineering
Environmental Services
Financial Services
Food & Beverage
Government Services
Health
Human Resources
Insurance
IT Services
Logistics & Transportation
Manufacturing
Media
Real Estate
Retail
Security
Software
Telecommunications
Travel & Hospitality
Industry
0
10000
20000
30000
Employees
Source: Inc. Magazine

Another approach is to divide the industries into ranges of outliers. Most companies can be viewed according to the preferred approach, while the remaining can be viewed with vertical boxplots. The advantage is being able to see the maximum outliers in relation to the IQR, if that is desired. However, the disadvantage is being unable to compare directly by visual inspection the distributions of the other companies.

```r
# Lowest outlier companies.
industry_low_outliers <- inc_state3 %>%
  group_by(Industry) %>%
  summarize(emp_count = sum(Employees)) %>%
  filter(emp_count < 2000) %>%
  select(Industry)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
# Plot lowest outliers.
inc_state3 %>%
  filter(Industry %in% industry_low_outliers$Industry) %>%
  ggplot(aes(x = Industry, y = Employees)) +
  geom_boxplot() +
  coord_flip() +
  xlim(sort(levels(inc_state3$Industry), decreasing = T)) +
    theme_bw() +
    labs(title = "5,000 Fastest Growing Companies in the U.S.",
         subtitle = "New York: Number of employees by industry\n(Industries with low outliers)",
         caption = "Source: Inc. Magazine")
```


5,000 Fastest Growing Companies in the U.S.
New York: Number of employees by industry
(Industries with low outliers)
Advertising & Marketing
Business Products & Services
Computer Hardware
Construction
Consumer Products & Services
Education
Energy
Engineering
Environmental Services
Financial Services
Food & Beverage
Government Services
Health
Human Resources
Insurance
IT Services
Logistics & Transportation
Manufacturing
Media
Real Estate
Retail
Security
Software
Telecommunications
Travel & Hospitality
Industry
0
200
400
600
Employees
Source: Inc. Magazine

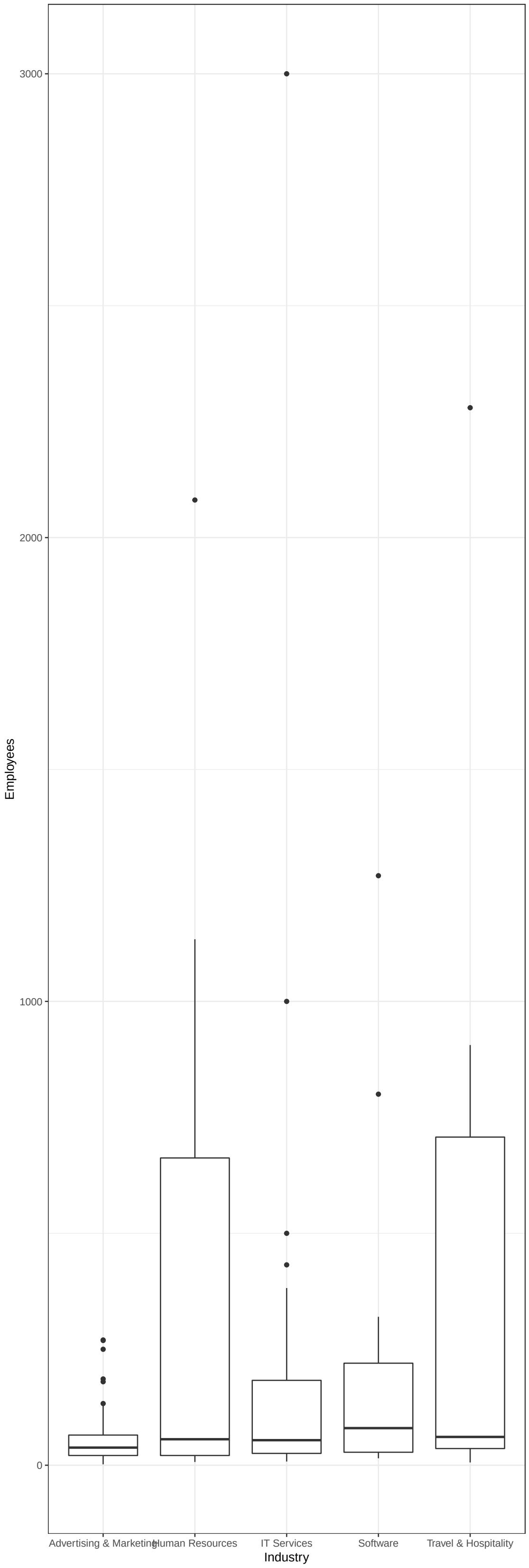
```
# Medium outlier companies.
industry_mid_outliers <- inc_state3 %>%
  group_by(Industry) %>%
  summarize(emp_count = sum(Employees)) %>%
  filter(between(emp_count, 2000, 10000)) %>%
  select(Industry)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
# Plot medium outliers vertically.
inc_state3 %>%
  filter(Industry %in% industry_mid_outliers$Industry) %>%
  ggplot(aes(x = Industry, y = Employees)) +
  geom_boxplot() +
    theme_bw() +
    labs(title = "5,000 Fastest Growing Companies in the U.S.",
         subtitle = "New York: Number of employees by industry\n(Industries with medium outliers)",
         caption = "Source: Inc. Magazine")
```

5,000 Fastest Growing Companies in the U.S.

New York: Number of employees by industry
(Industries with medium outliers)



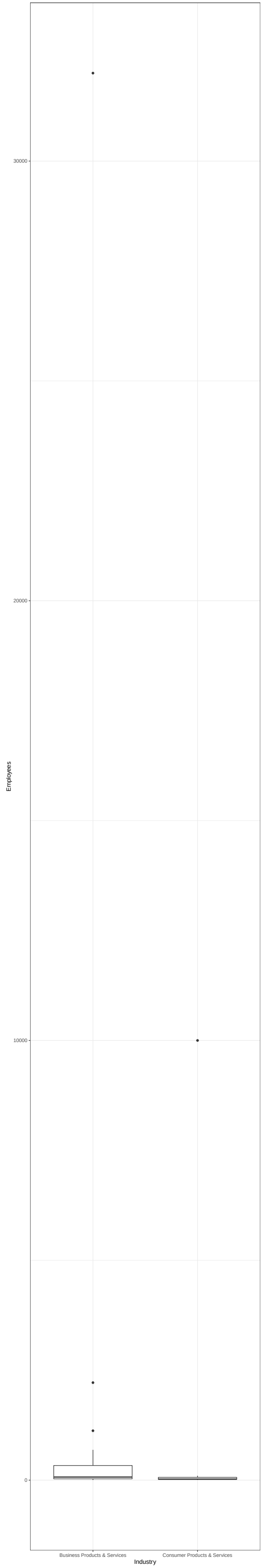
Source: Inc. Magazine

Comment: The plot below is worthless. I wonder if a legal solution would be to truncate the vertical axis, snip out a portion of the range in order to collapse the extreme displacement. You could put in some kind of graphic equivalent of an ellipses to show the rante you're "tearing out" and an annotation about the omission.

```
# Extreme outlier companies.
industry_hi_outliers <- inc_state3 %>%
  group_by(Industry) %>%
  summarize(emp_count = sum(Employees)) %>%
  filter(emp_count > 10000) %>%
  select(Industry)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
inc_state3 %>%
  filter(Industry %in% industry_hi_outliers$Industry) %>%
  ggplot(aes(x = Industry, y = Employees)) +
  geom_boxplot() +
    theme_bw() +
    labs(title = "5,000 Fastest Growing Companies in the U.S.",
         subtitle = "New York: Number of employees by industry\n(Industries with extreme outliers)",
         caption = "Source: Inc. Magazine")
```



5,000 Fastest Growing Companies in the U.S.

New York: Number of employees by industry
(Industries with extreme outliers)

Source: Inc. Magazine