

ggvis vs. ggplot

Kyle Cho

12/03/2017

Introduction

While working on the project one thing I noticed was the thing called ggvis. I had originally thought all the plots were to be generated using ggplot which has been the bread and butter of this course, but later realized I should be using ggvis instead. Though I had never seen a single line of ggvis code before its syntax looked really familiar. The thought of “why have two nearly identical libraries” has been lingering for quite a while, and I’ve decided to use this opportunity to find the answer to the question. In this post I will compare and contrast various aspects of each library.

Overview

ggplot2’s official website says “ggplot2 is a plotting system for R, based on the grammar of graphics, which tries to take the good parts of base and lattice graphics and none of the bad parts.” In short, it does plotting. In contrast, ggvis lets you “declaratively describe data graphics with a syntax similar in spirit to ggplot2” and “create rich interactive graphics that you can play with locally in Rstudio or in your browser.”. In short, it’s like ggplot2 plus interactivity.

Setup

Before getting into details let’s start by importing libraries and a data file.

```
# Import libraries
library(shiny)
library(ggplot2)
library(ggvis)
```

```
##
## Attaching package: 'ggvis'
```

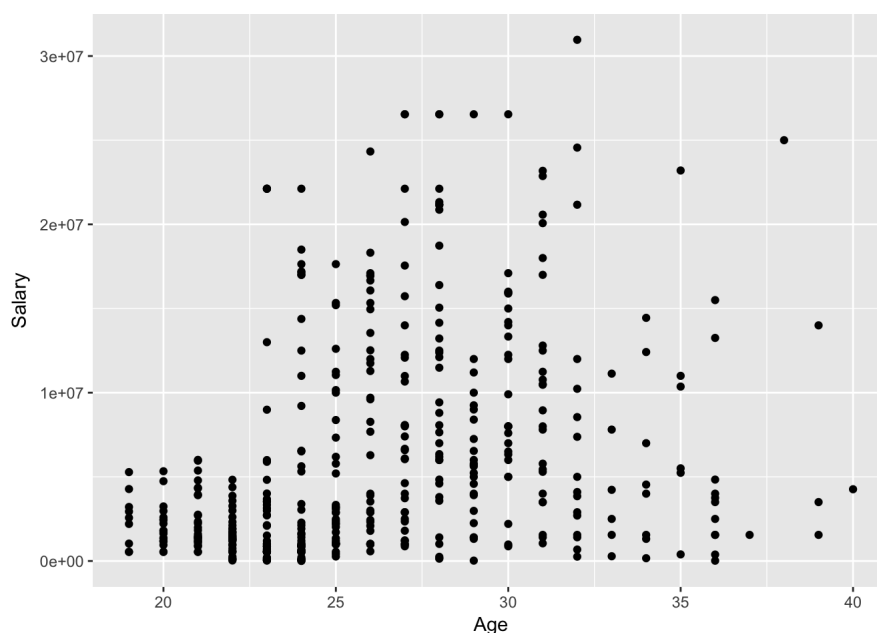
```
## The following object is masked from 'package:ggplot2':
##
## resolution
```

```
# Read CSV file
data = read.csv('nba2017-player-statistics.csv')
```

Syntax

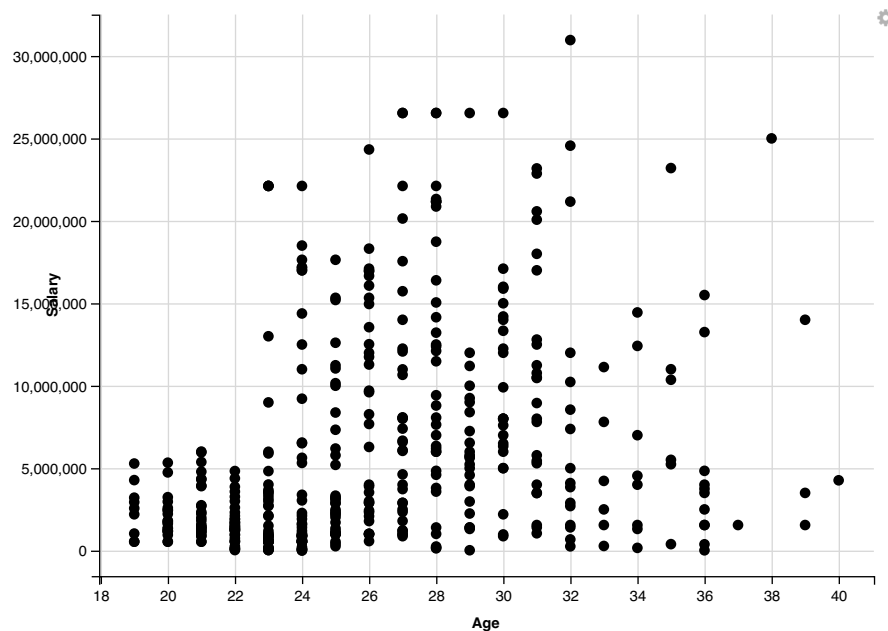
In ggplot2 you create a plot by first calling ggplot() then choose a geom function of your choice to determine its representation.

```
# Age vs Salary
ggplot(data, aes(x=Age, y=Salary)) + geom_point()
```



Notice how you ‘chain’ things using + operation. Now let’s take a look at the ggvis equivalent.

```
# Age vs Salary
data %>% ggvis(~Age, ~Salary) %>% layer_points()
```



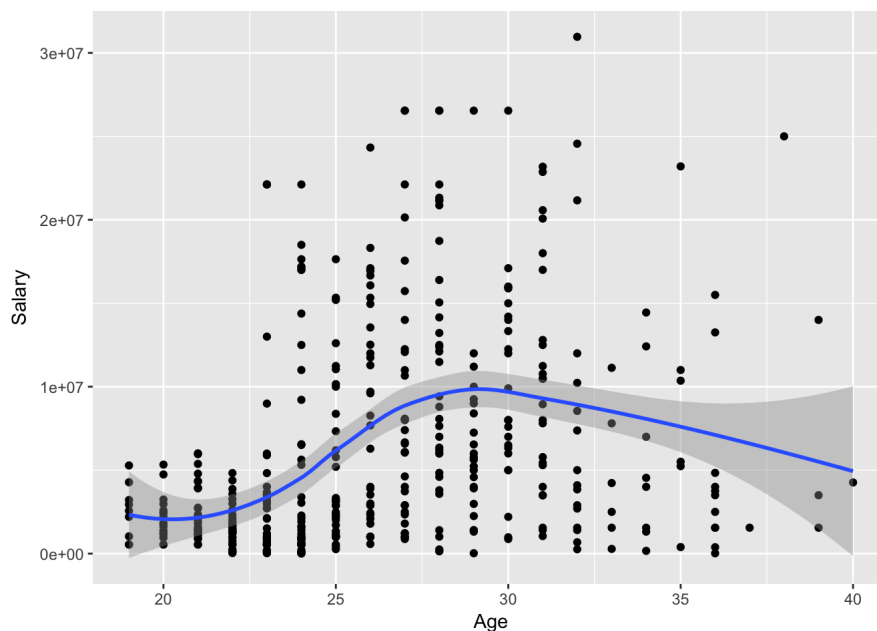
ggvis uses pipes, similar to dplyr, to create a plot. I think this notation makes it easier for beginners to understand the flow of the functions. Also I have always hated how I have to use this mysterious aes - no idea what it stands for - thing to do stuff in ggplot2.

Performance

Since comparing things based on their appearance and syntax is not very objective, let's see if there's any performance difference between these two.

```
# Time before plotting
ggplot2_time_s = as.numeric(Sys.time())
# Age vs Salary w/ loess line
ggplot(data, aes(x=Age, y=Salary)) + geom_point() + geom_smooth()
```

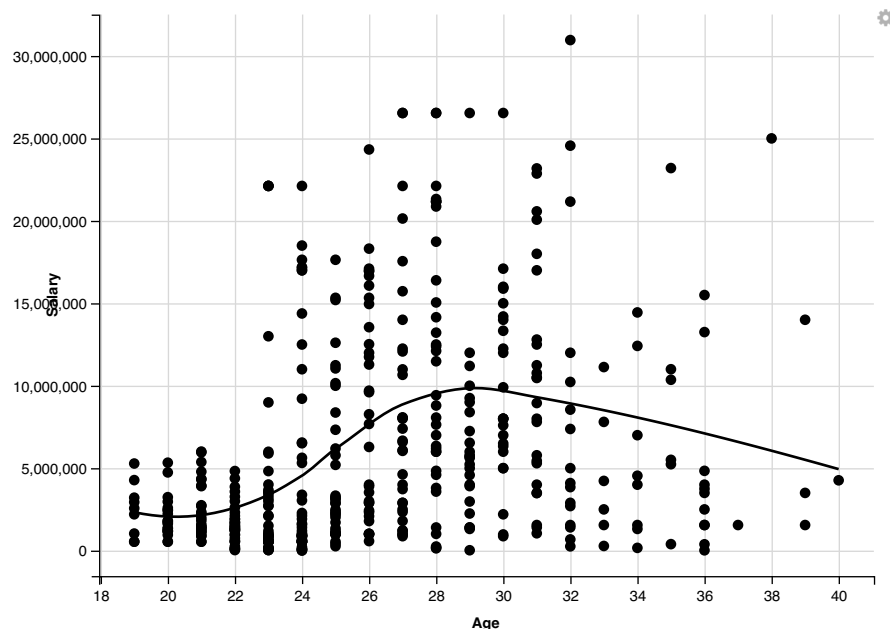
```
## `geom_smooth()` using method = 'loess'
```



```
# Time after plotting
ggplot2_time_e = as.numeric(Sys.time())
# Difference
ggplot2_time = ggplot2_time_e - ggplot2_time_s
```

As you see it took 0.712117 seconds to produce the plot above. What about ggvis?

```
# Time before plotting
ggvis_time_s = as.numeric(Sys.time())
# Age vs Salary w/ loess line
data %>% ggvis(~Age, ~Salary) %>% layer_points() %>% layer_smooths()
```



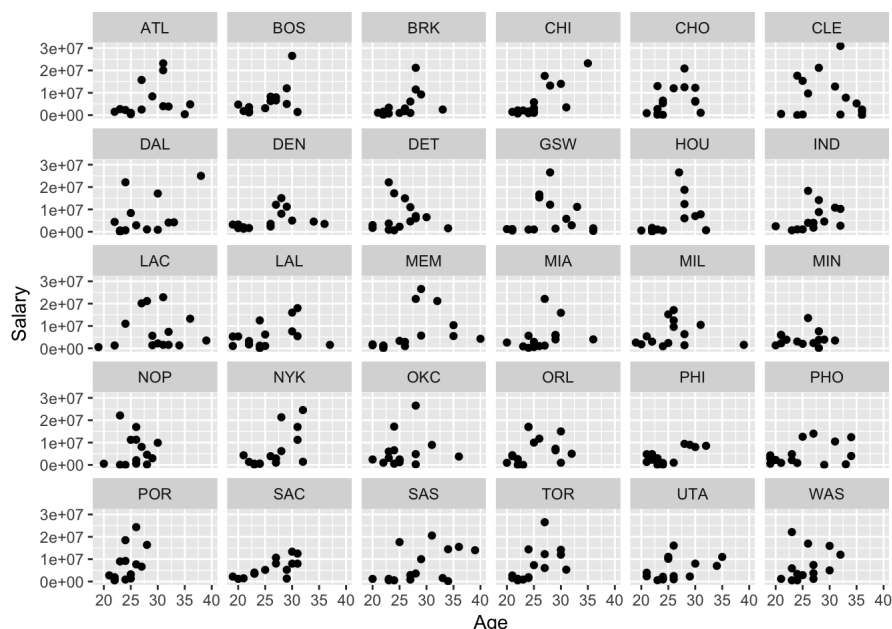
```
# Time after plotting
ggvis_time_e = as.numeric(Sys.time())
# Difference
ggvis_time = ggvis_time_e - ggvis_time_s
```

ggvis plots the same data in 0.2185569 seconds, 0.4935601 faster than ggplot2. While my test isn't guaranteed to output the same result all the time (there is no way of reproducing this exact same result on a different machine - as long as ggvis does the job faster, consider it reproducible), I think it is safe to conclude that ggvis is superior performance-wise.

What ggplot2 can do that ggvis can't: Facetting

Based on our analysis so far ggvis seems to dominate ggplot2 on every aspect. So why use ggplot2 at all? Consider the following plot.

```
# Facetting
ggplot(data, aes(x=Age, y=Salary)) + geom_point() + facet_wrap(~Team)
```



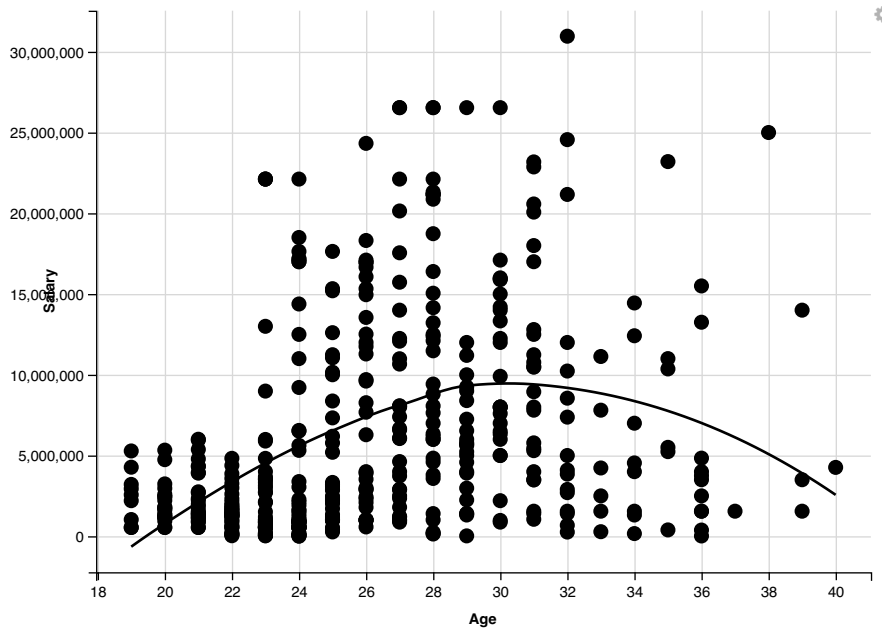
See how it groups players based on the team column and then makes 30 different plots? This feature is called facetting and is not supported by ggvis. We technically could achieve a similar result using a for loop with ggvis, but it still doesn't give this nice grid layout. We always work with categorical data, so this isn't an issue that should be overlooked.

What ggvis can do that ggplot2 can't: Interactivity

Plots generated by ggplot2 are static - that is, once you see them they never change. In contrast, you can play with ggvis plots in real time.

```
# Interactive plot
data %>% ggvis(~Age, ~Salary) %>%
  layer_points(size := input_slider(100, 1000, value = 100)) %>%
  layer_smooths(span = input_slider(0.5, 1, value = 1))
```

```
## Warning: Can't output dynamic/interactive ggvis plots in a knitr document.  
## Generating a static (non-dynamic, non-interactive) version of the plot.
```



(Note: As the warning says, interactivity is missing on the knitted HTML, so if you want to check if it's reproducible copy and paste the entire thing in RStudio.)

This added interactivity is really great when you're displaying the HTML output to someone else.

Conclusion

While I think ggvis has better syntax, it's a matter of preference after all so I won't say you should think the same. Similarly, I won't say one is aesthetically better than the other. I do however would like to leave a meaningful message to help you choose the right tool for your future projects.

ggplot2 is a very mature library, and it has various functions - facetting for example - that ggvis doesn't. If you want to make a pdf-ish static document with traditional data analysis where performance nor interactivity is required, ggplot2 would be a good choice.

In contrast, if you're looking to create an app that lets users tweak parameters, ggvis would be much better, especially in conjunction with Shiny. First off ggplot2 alone can't handle any interactivity at all, and obviously you want your app to be fast so the users won't complain.

References

- <https://ggvis.rstudio.com/ggplot2.html>
- <http://ggplot2.org/>
- <https://stats.stackexchange.com/questions/117078/for-plotting-with-r-should-i-learn-ggplot2-or-ggvis>
- <https://datascience.stackexchange.com/questions/9885/ggvis-vs-ggplot2shiny-which-one-to-choose-for-interactive-visualization>
- <http://jimhester.github.io/ggplot2ToGgvis/>
- https://www.reddit.com/r/rstats/comments/5zr3cg/ggvis_vs_ggplot_when_to_use_one_package_vs_the/
- <https://stackoverflow.com/questions/36353348/pipe-ggplot2-like-ggvis>