

# Strata+ Hadoop WORLD

PRES EN TED BY

O'REILLY®

cloudera®



250 Northern Ave, Boston, MA 02210

Phone: 844-448-1212

Email: [info@rstudio.com](mailto:info@rstudio.com)

Web: <http://www.rstudio.com>

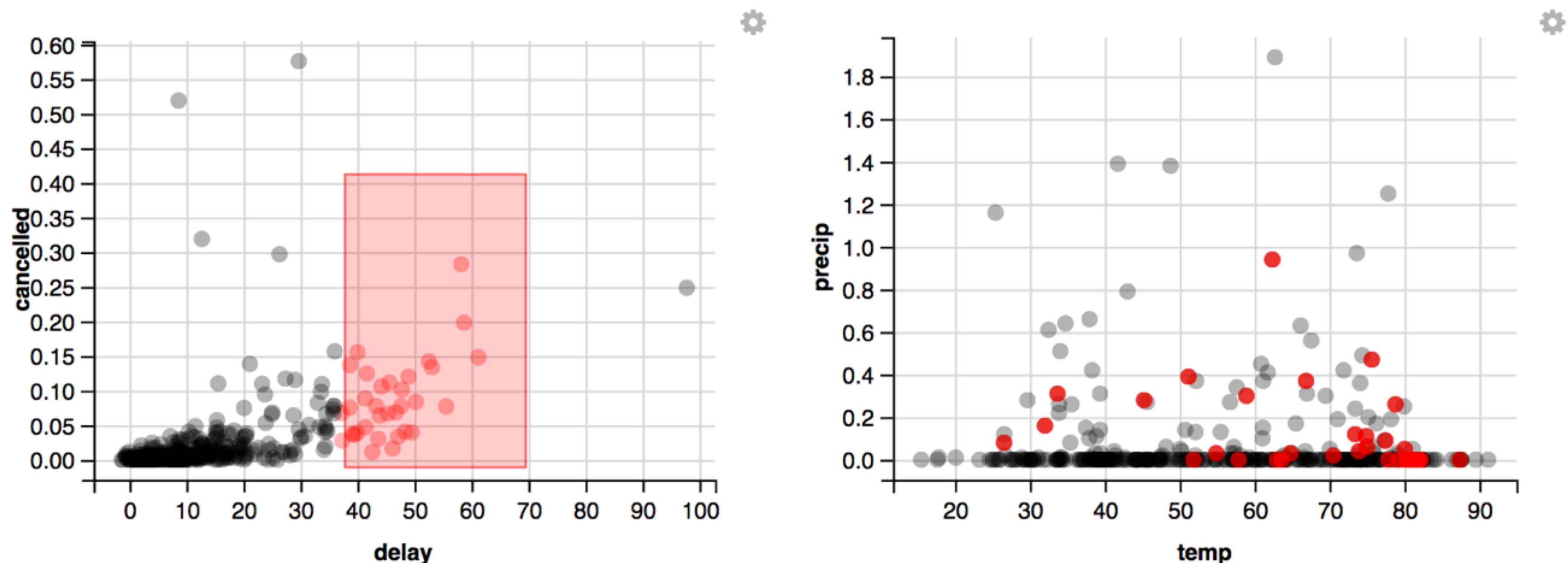
Copyright 2014 RStudio | All Rights Reserved

[strataconf.com](http://strataconf.com)

#StrataHadoop

# ggvis: Interactive, intuitive graphics with R

## Introduction and possibilities



## Garrett Grolemund

Data Scientist and Master Instructor

Email: [garrett@rstudio.com](mailto:garrett@rstudio.com)

Follow [@StatGarrett](https://twitter.com/StatGarrett)

**HELLO**

my name is

**Garrett**



garrett@rstudio.com



@StatGarrett

**what is  
ggvis?**

 set up

The following demos require you to install two free pieces of software:

- R, which you can download from [cran.r-project.org](http://cran.r-project.org)
- RStudio, which you can download from [www.rstudio.com/download](http://www.rstudio.com/download)

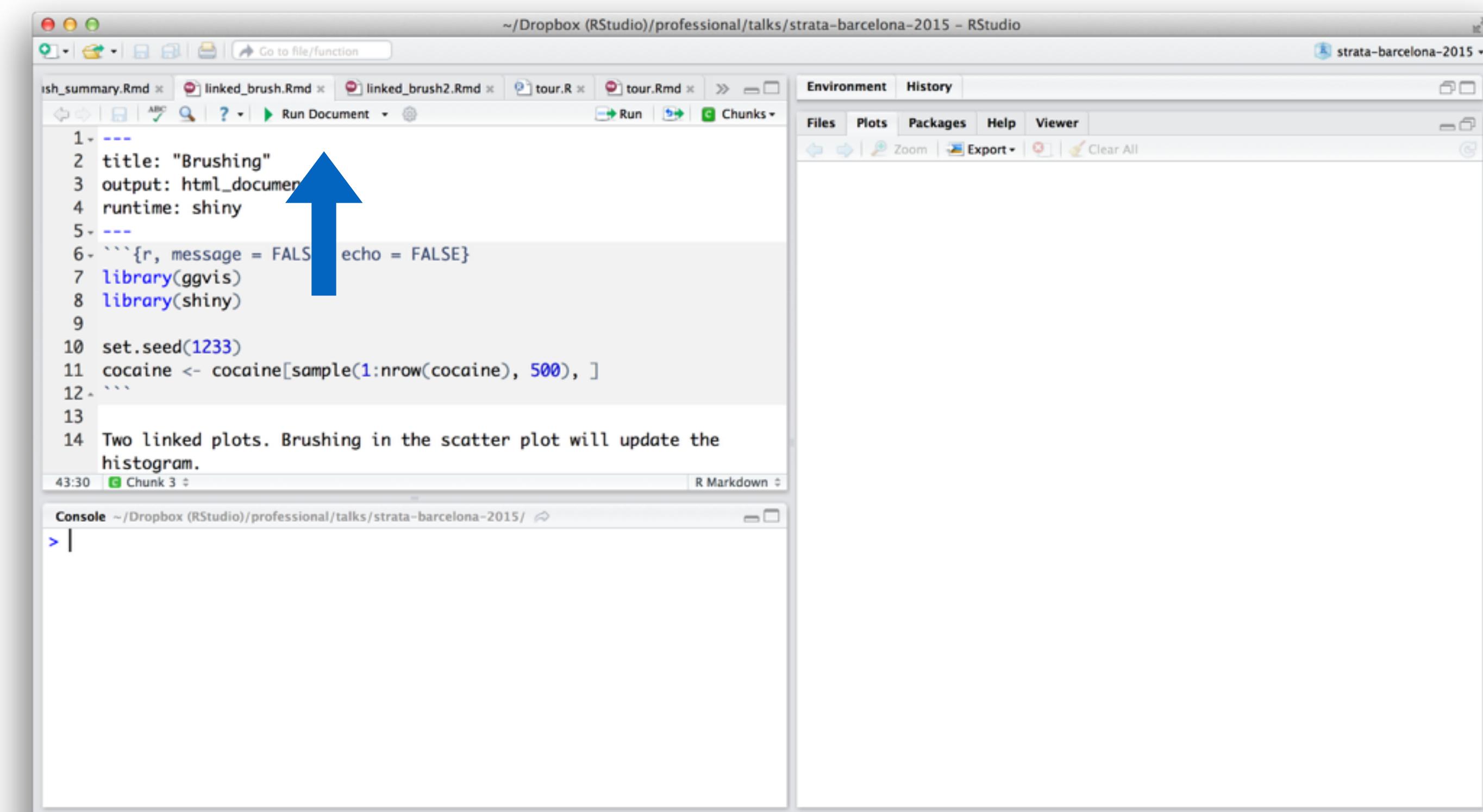
You will also need to open RStudio and run the following command at the command prompt before running the demos:

```
install.packages(c("dplyr", "ggvis", "knitr",
  "lubridate", "nycflights", "shiny", "tourr"))
```

## 1

# Explore points

To run the demo, open `brush_summary.Rmd` in RStudio and click "Run Document" at the top of the file pane.



# 2 Animated rotation

```
# rotation.R
library(tourr)
library(ggvis)
library(shiny)

aps <- 2
fps <- 30

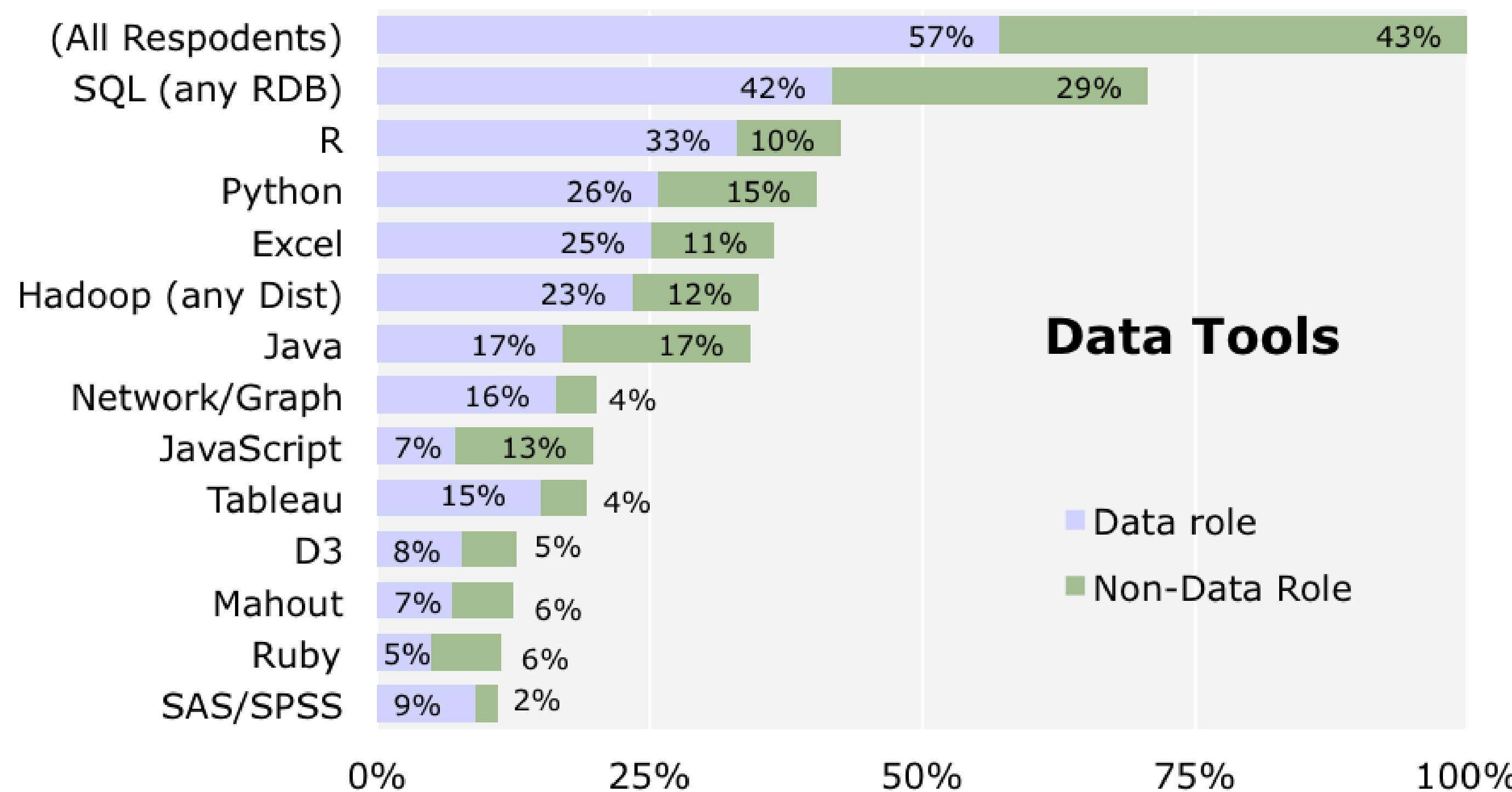
mat <- rescale(as.matrix(flea[c(1,3,4)]))
tour <- new_tour(mat, grand_tour(), NULL)
start <- tour(0)

proj_data <- reactive({
  invalidateLater(1000 / fps, NULL);
  step <- tour(aps / fps)
  data.frame(center(mat %*% step$proj), species = flea$species)
})

proj_data %>% ggvis(~X1, ~X2, fill = ~species) %>%
  layer_points() %>%
  scale_numeric("x", domain = c(-1, 1)) %>%
  scale_numeric("y", domain = c(-1, 1)) %>%
  set_options(duration = 0)
```

**R, ggplot2,  
and vega**

# 2013 Data Science Salary Survey



RStudio

Project: (None)

Console ~/ Go to file/function

> WorldPhones

	N.Amer	Europe	Asia	S.Amer	Oceania	Africa	Mid.Amer
1951	45939	21574	2876	1815	1646	89	555
1956	60423	29990	4708	2568	2366	1411	733
1957	64721	32510	5230	2695	2526	1546	773
1958	68484	35218	6662	2845	2691	1663	836
1959	71799	37598	6856	3000	2868	1769	911
1960	76036	40341	8220	3145	3054	1905	1008
1961	79831	43173	9053	3338	3224	2005	1076

>

>

> summary(iris\$Sepal.Width)

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
2.000	2.800	3.000	3.057	3.300	4.400

>

>

> lm(mpg ~ disp + hp, data = mtcars)

Call:

lm(formula = mpg ~ disp + hp, data = mtcars)

Coefficients:

(Intercept)	disp	hp
30.73590	-0.03035	-0.02484

>

> hist(iris\$Sepal.Width, border = "white", col = "darkgrey")

> |

Environment History

Import Dataset Clear Grid

Global Environment

Name Type Length Size Value

Environment is empty

Files Plots Packages Help Viewer

Zoom Export Clear All

Histogram of iris\$Sepal.Width

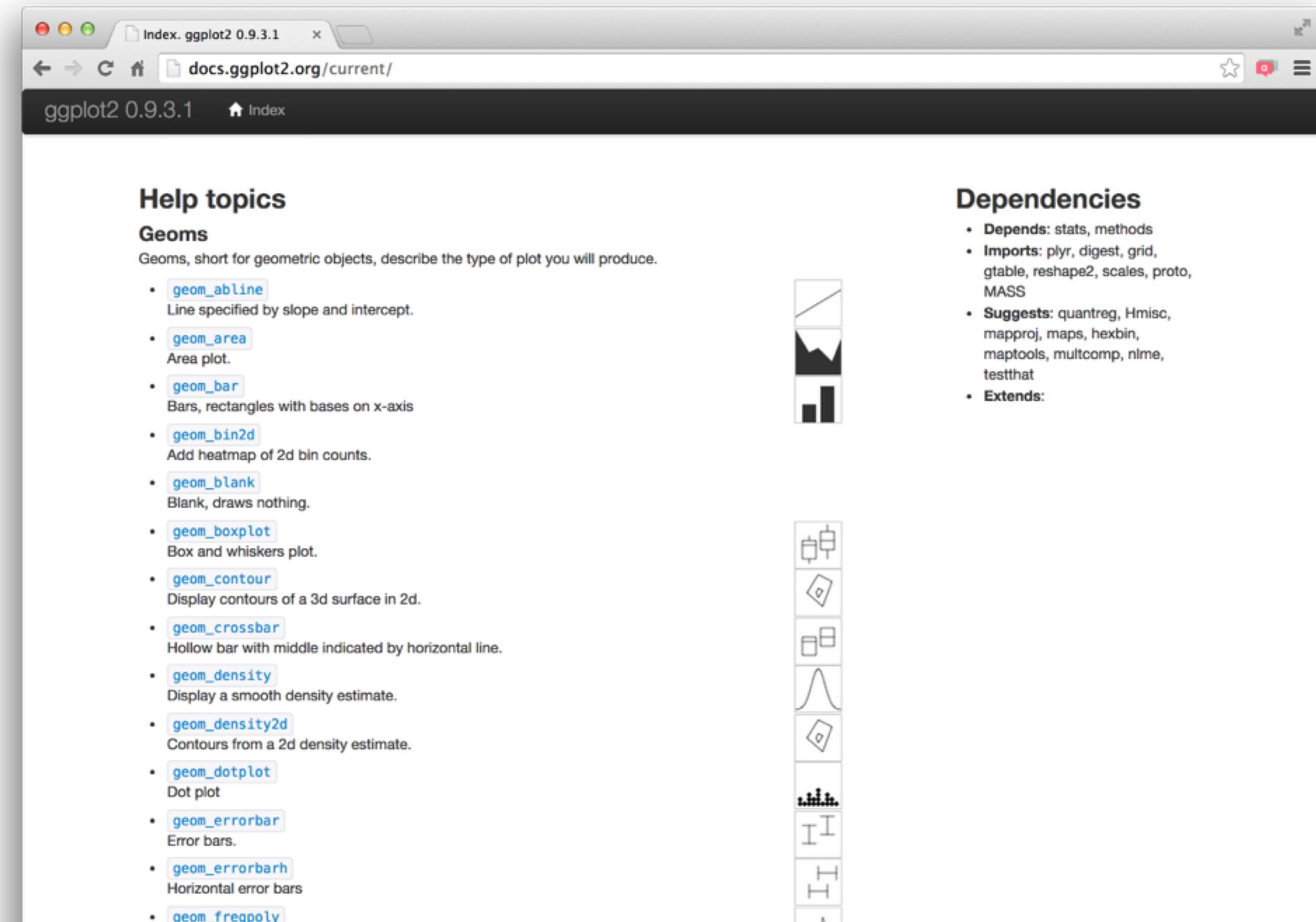
Frequency

iris\$Sepal.Width

2.0 2.5 3.0 3.5 4.0

# ggplot2

A software package for making graphics with R.  
<http://docs.ggplot2.org/current/>



The screenshot shows a web browser displaying the ggplot2 documentation. The title bar says "Index. ggplot2 0.9.3.1" and the address bar says "docs.ggplot2.org/current/". The main content area has a dark header bar with "ggplot2 0.9.3.1" and "Index".

**Help topics**

**Geoms**

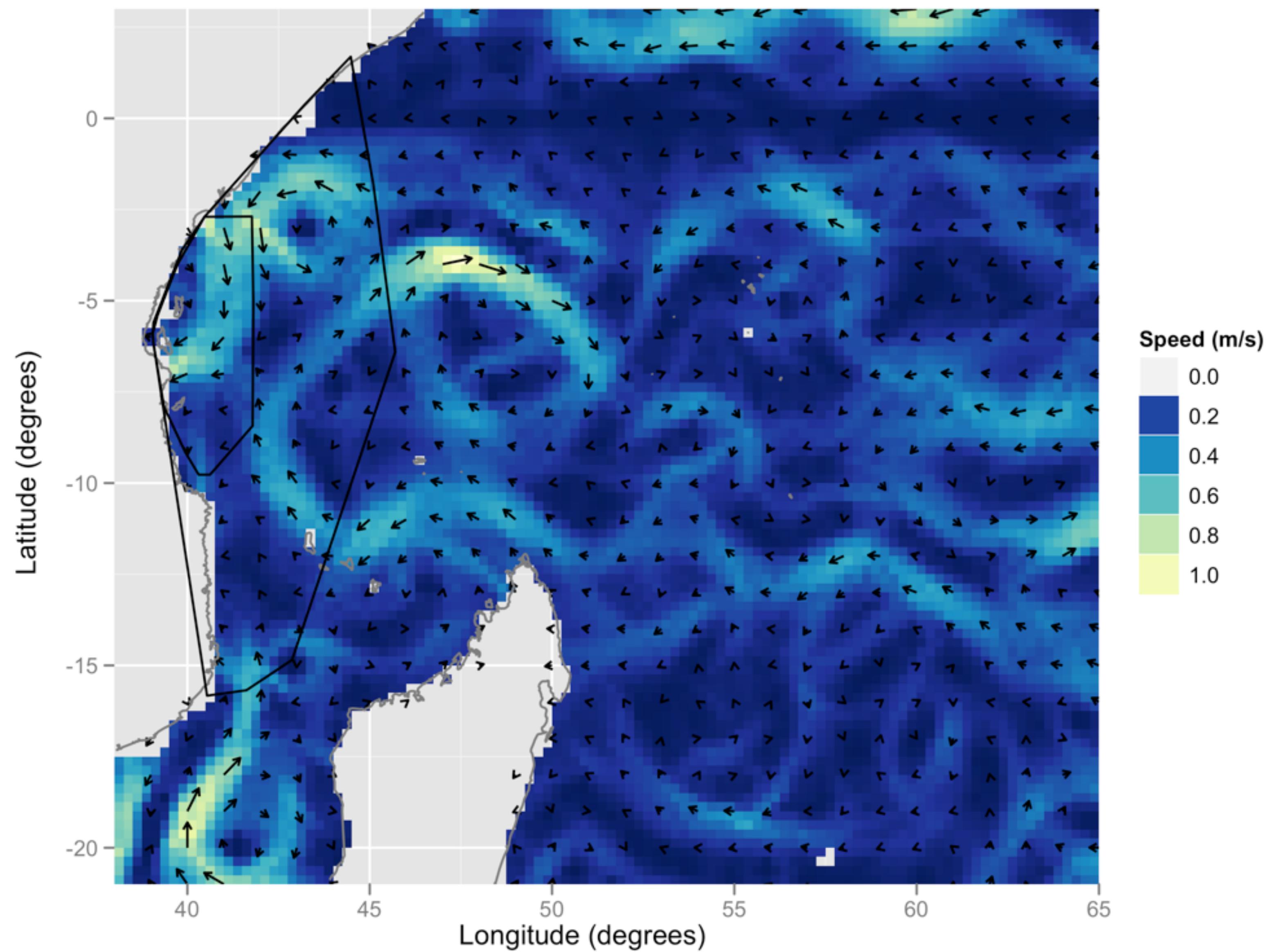
Geoms, short for geometric objects, describe the type of plot you will produce.

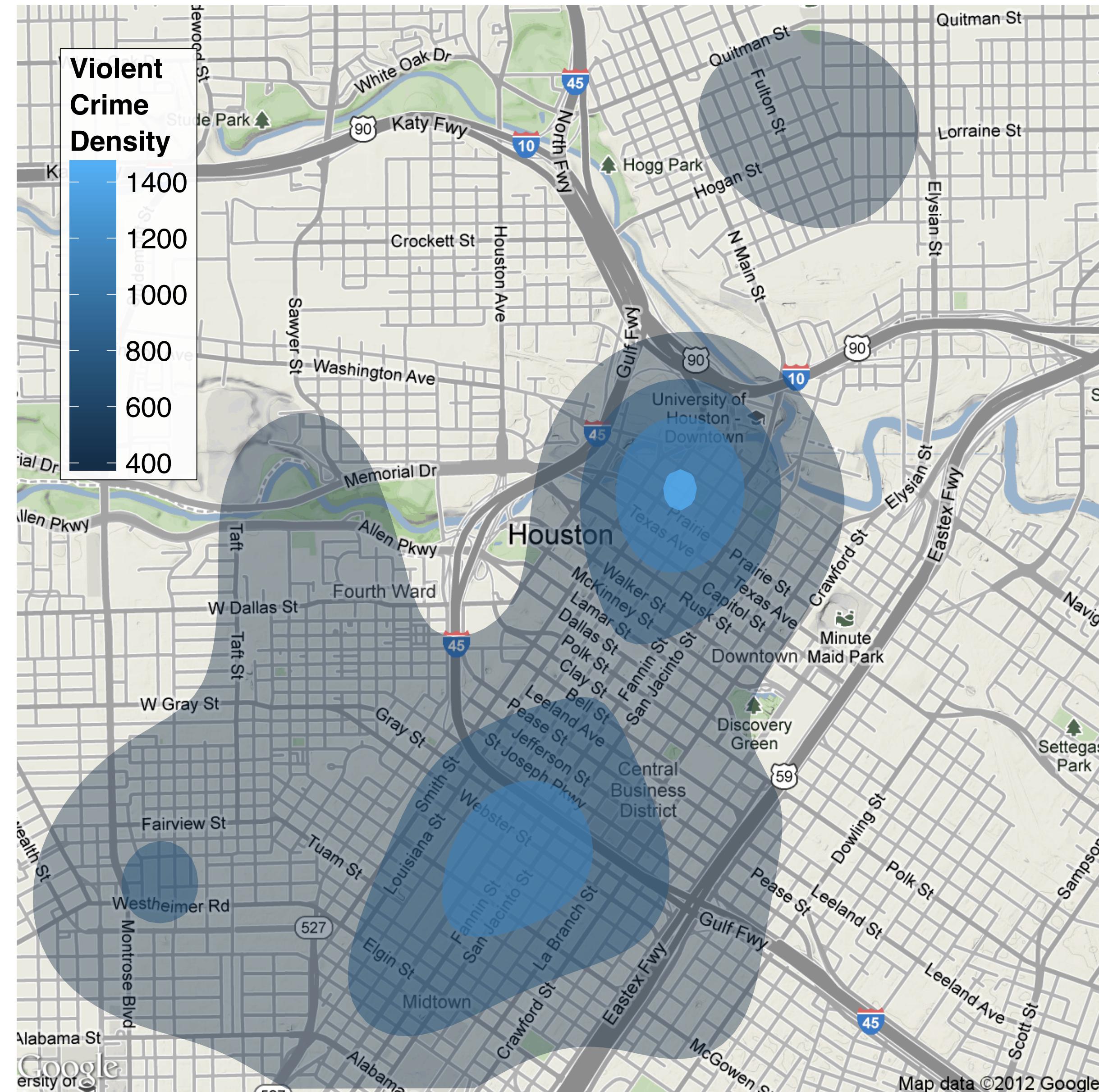
- [geom\\_abline](#) Line specified by slope and intercept.
- [geom\\_area](#) Area plot.
- [geom\\_bar](#) Bars, rectangles with bases on x-axis
- [geom\\_bin2d](#) Add heatmap of 2d bin counts.
- [geom\\_blank](#) Blank, draws nothing.
- [geom\\_boxplot](#) Box and whiskers plot.
- [geom\\_contour](#) Display contours of a 3d surface in 2d.
- [geom\\_crossbar](#) Hollow bar with middle indicated by horizontal line.
- [geom\\_density](#) Display a smooth density estimate.
- [geom\\_density2d](#) Contours from a 2d density estimate.
- [geom\\_dotplot](#) Dot plot
- [geom\\_errorbar](#) Error bars.
- [geom\\_errorbarh](#) Horizontal error bars
- [geom\\_freqpoly](#)

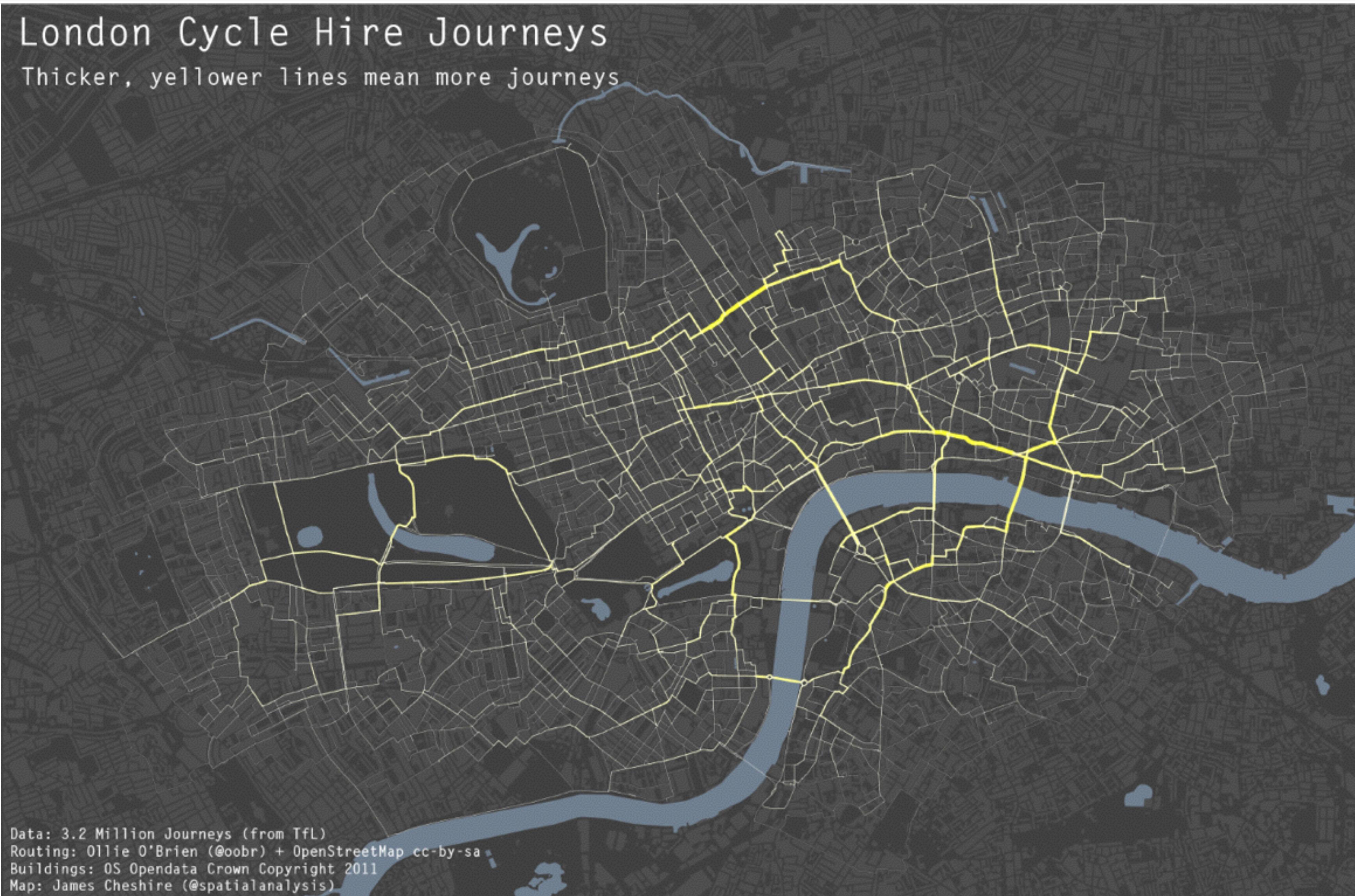
**Dependencies**

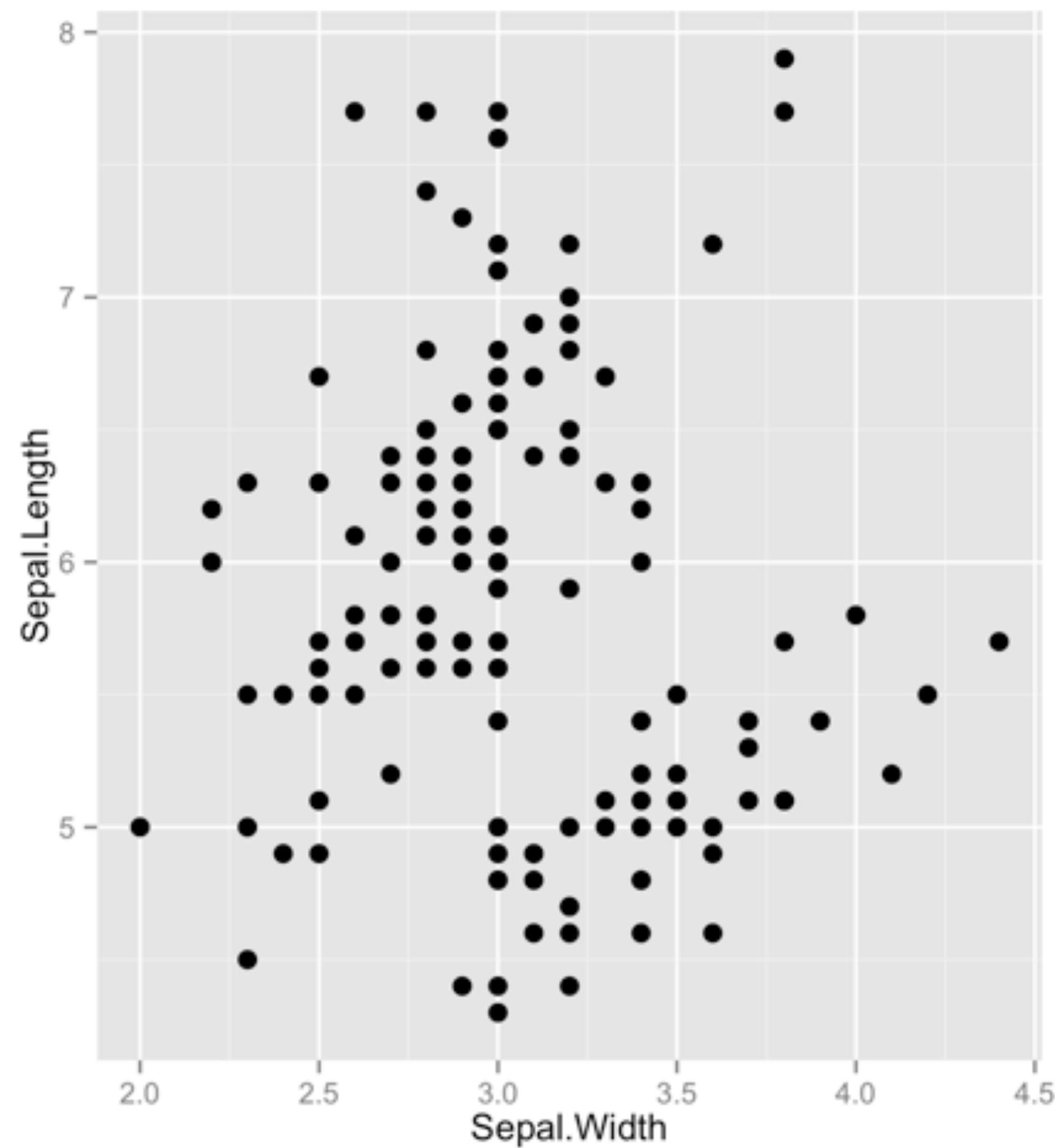
- Depends: stats, methods
- Imports: plyr, digest, grid, gtable, reshape2, scales, proto, MASS
- Suggests: quantreg, Hmisc, mapproj, maps, hexbin, maptools, multcomp, nlme, testthat
- Extends:

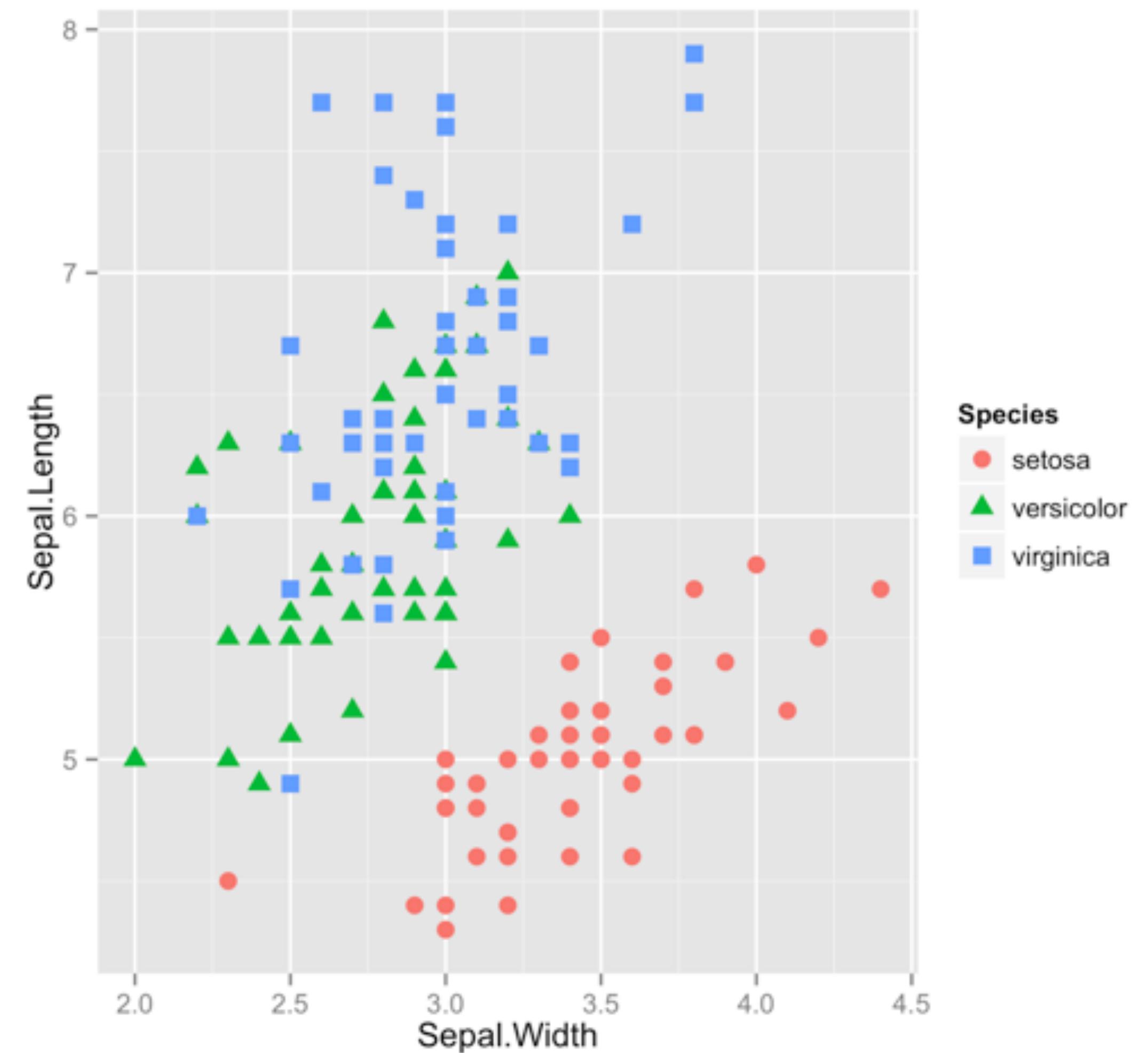
On the right side of the page, there are two vertical columns of small icons representing different types of plots: boxplots, density plots, histograms, and error bars.

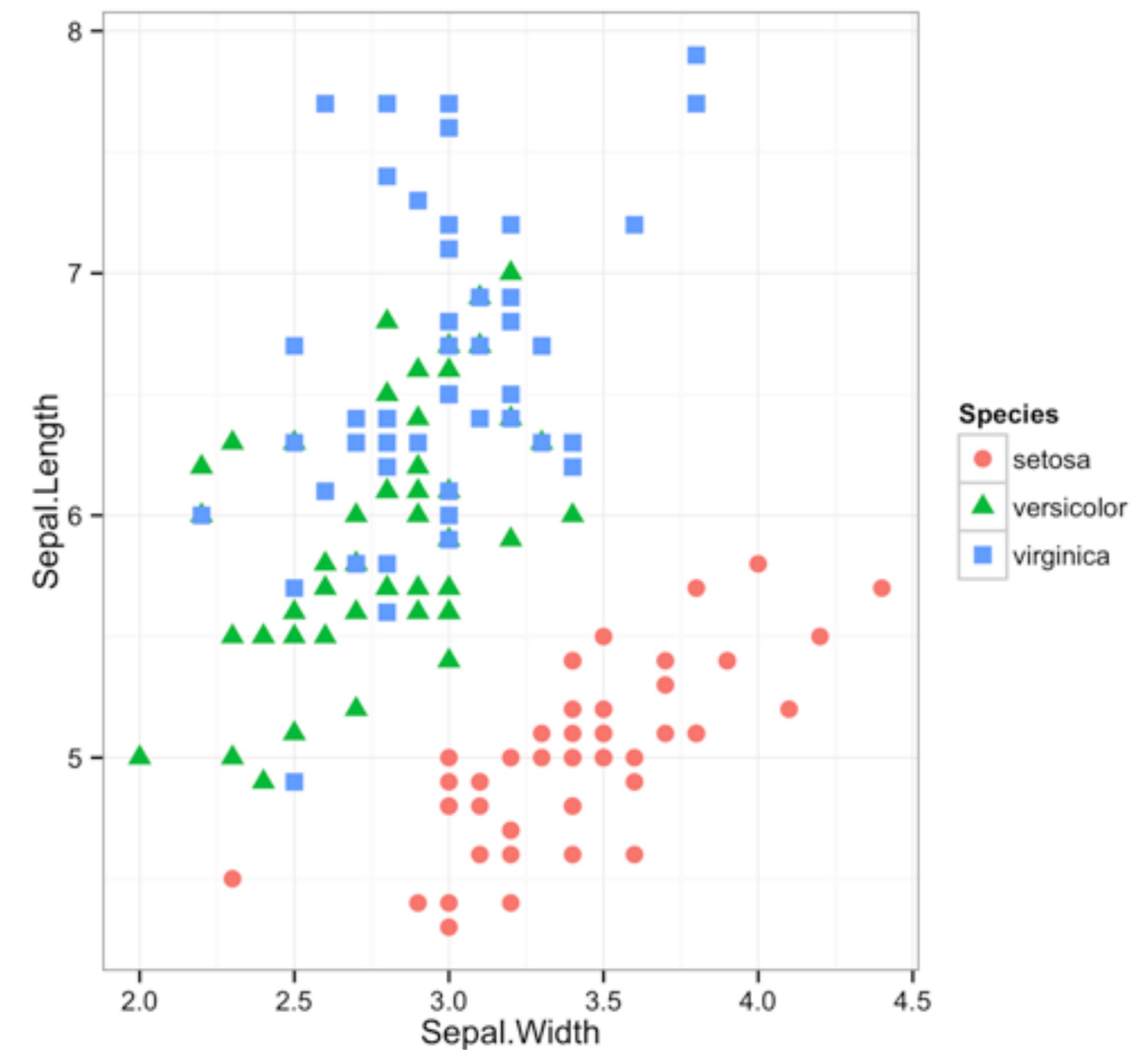


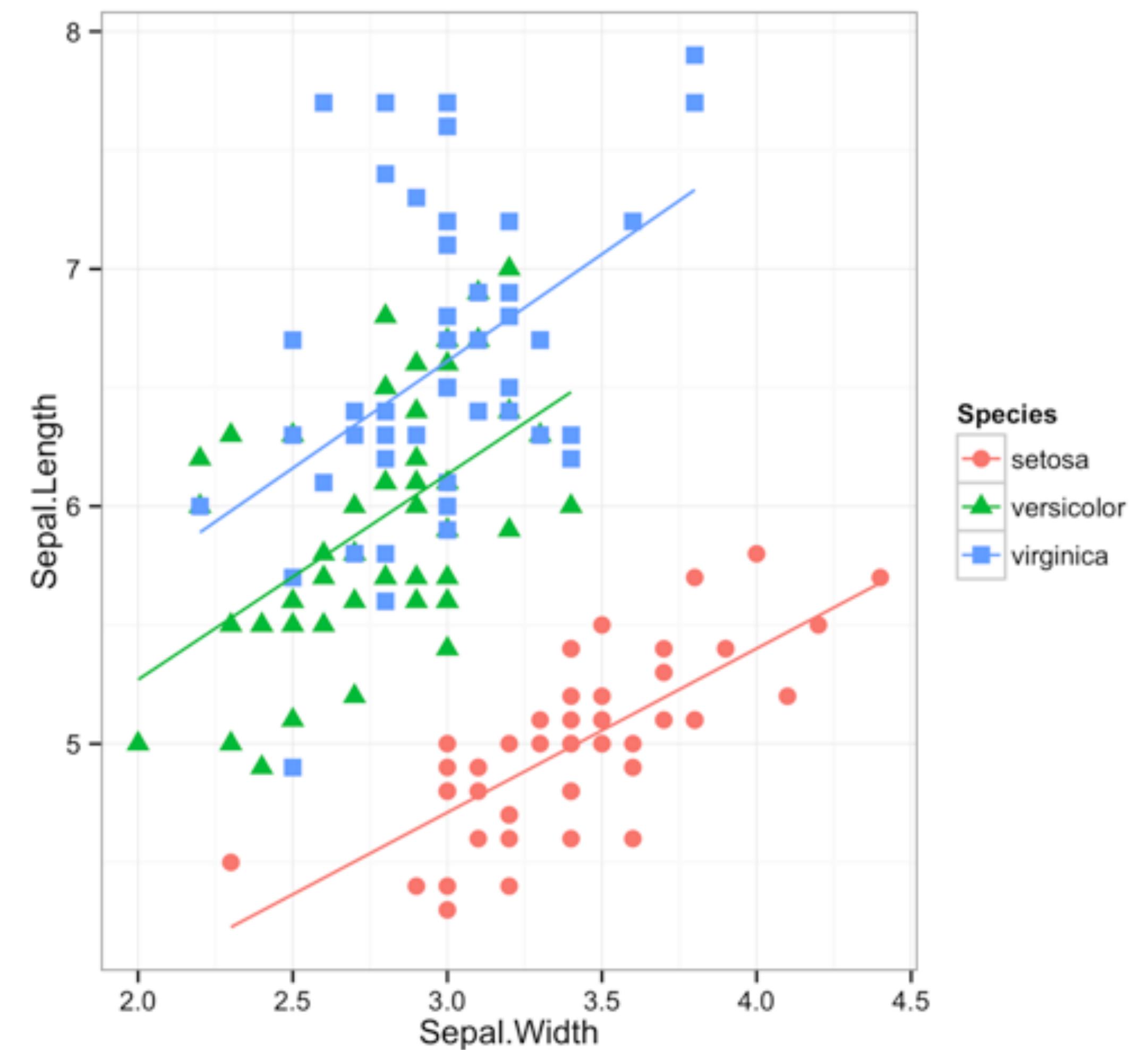


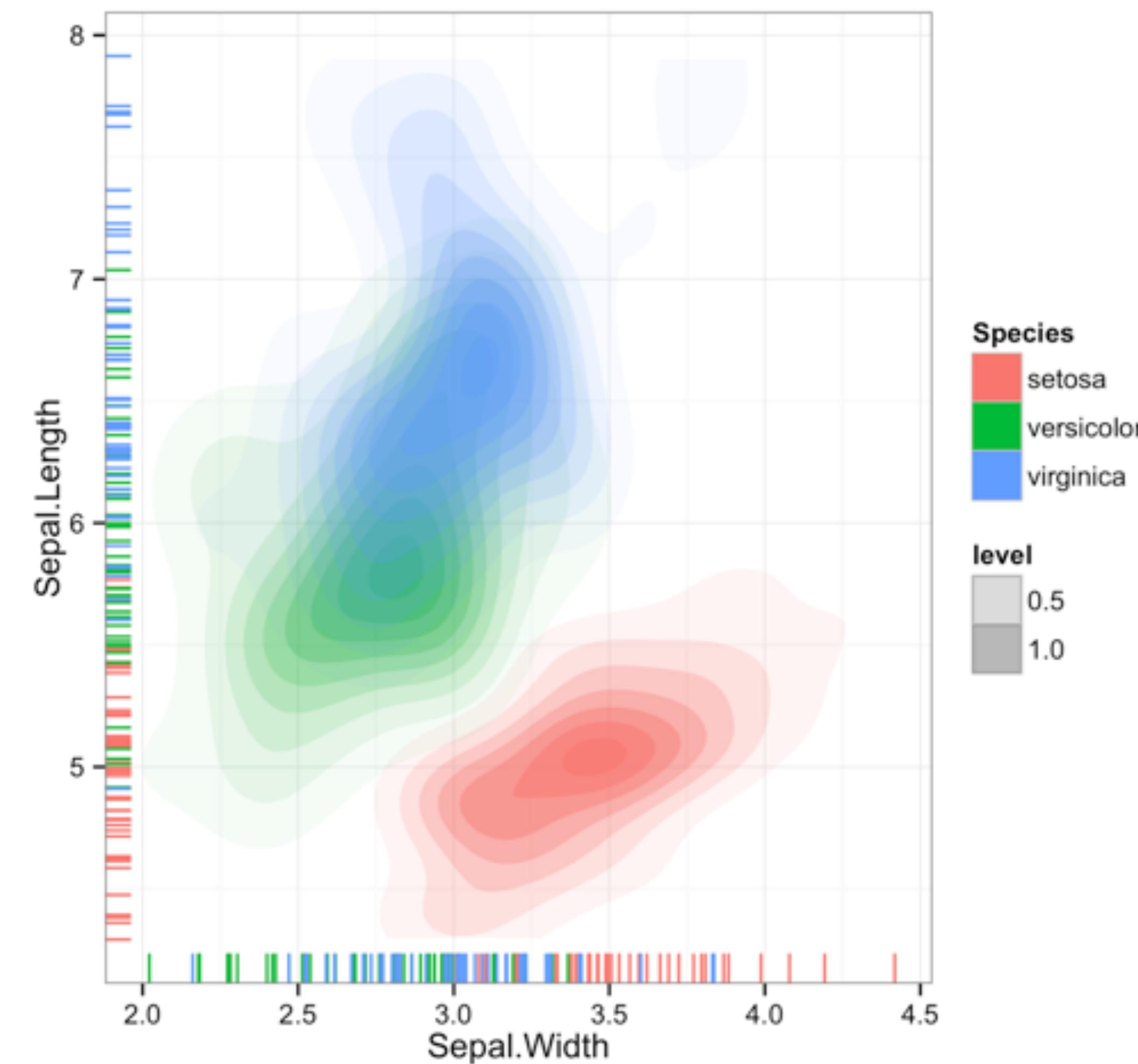


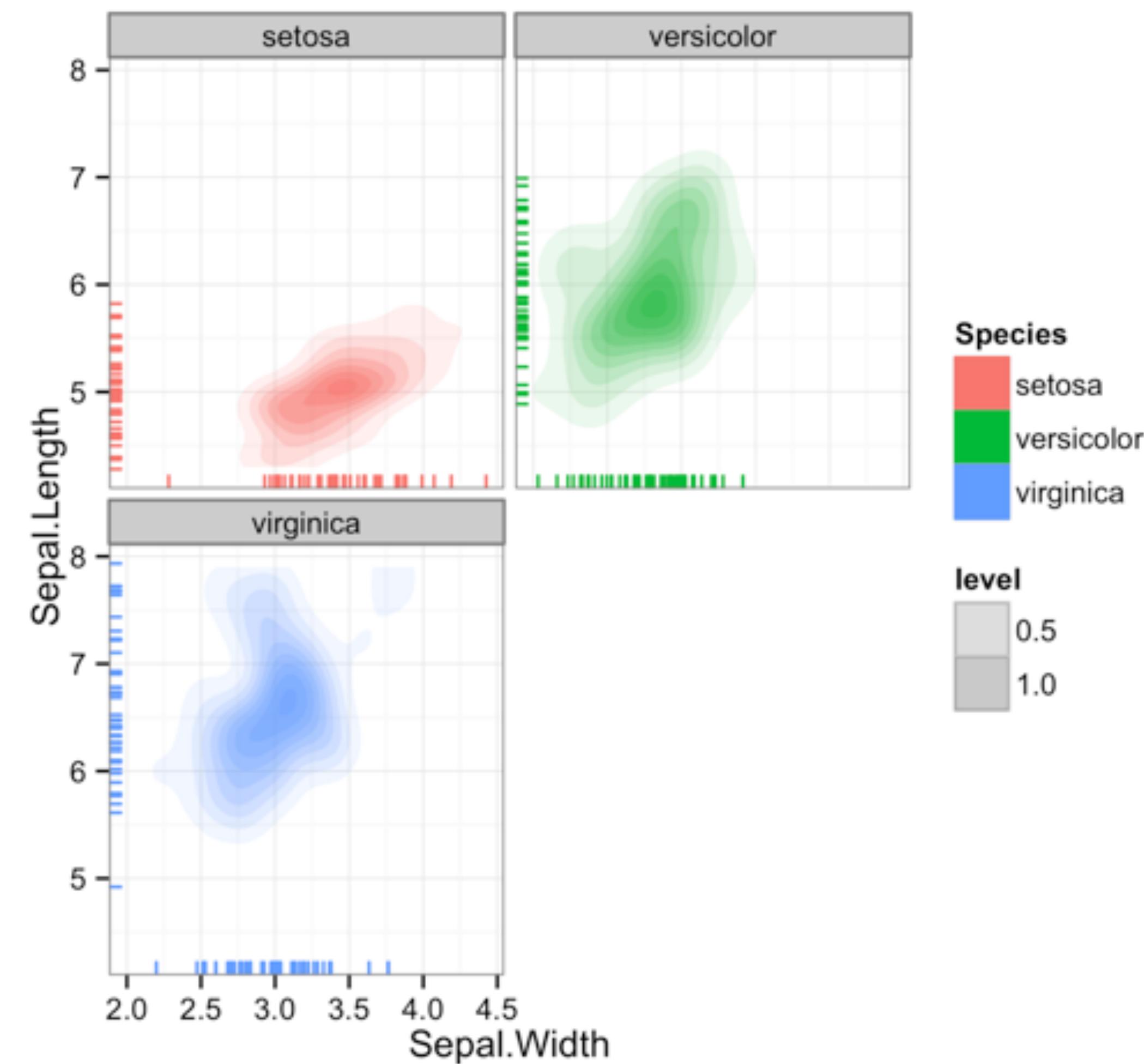












RStudio

Project: (None)

Console ~/ Go to file/function

> WorldPhones

	N.Amer	Europe	Asia	S.Amer	Oceania	Africa	Mid.Amer
1951	45939	21574	2876	1815	1646	89	555
1956	60423	29990	4708	2568	2366	1411	733
1957	64721	32510	5230	2695	2526	1546	773
1958	68484	35218	6662	2845	2691	1663	836
1959	71799	37598	6856	3000	2868	1769	911
1960	76036	40341	8220	3145	3054	1905	1008
1961	79831	43173	9053	3338	3224	2005	1076

>

>

> summary(iris\$Sepal.Width)

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
2.000	2.800	3.000	3.057	3.300	4.400

>

>

> lm(mpg ~ disp + hp, data = mtcars)

Call:

lm(formula = mpg ~ disp + hp, data = mtcars)

Coefficients:

(Intercept)	disp	hp
30.73590	-0.03035	-0.02484

>

> hist(iris\$Sepal.Width, border = "white", col = "darkgrey")

> |

Environment History

Import Dataset Clear Grid

Global Environment

Name Type Length Size Value

Environment is empty

Files Plots Packages Help Viewer

Zoom Export Clear All

Histogram of iris\$Sepal.Width

Frequency

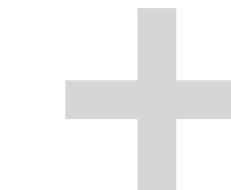
2.0 2.5

Limited display tools

# ggvis



Analytic  
Power



**HTML**



**CSS**



Display abilities  
and interactivity

# ggvis



Analytic  
Power

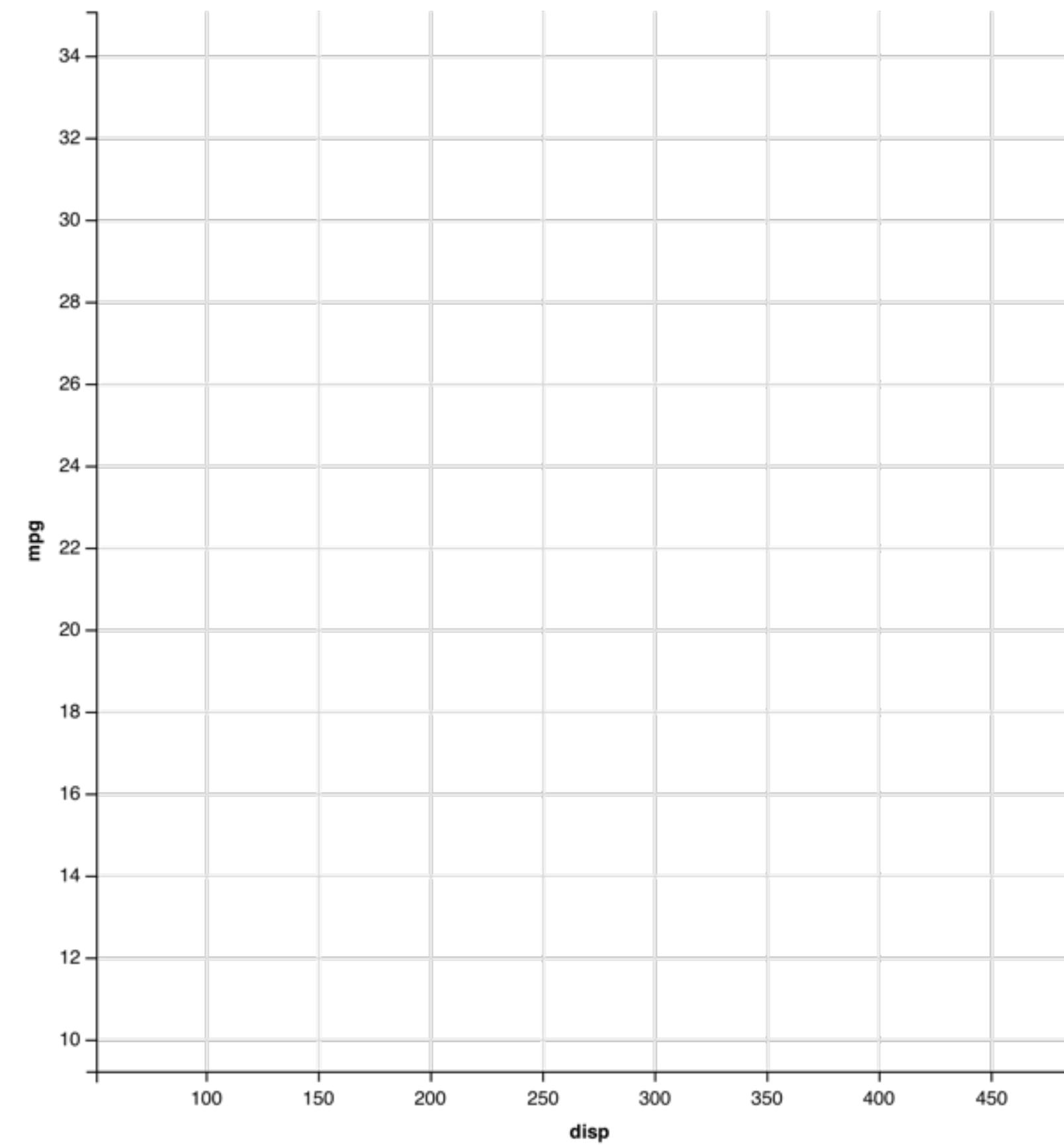
+ vega.js

Display abilities  
and interactivity

# A domain language for graphs

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

**data**



**coordinate  
system**

# Initialize plot

```
mtcars %>% ggvis()
```

data frame  
to plot

%>%

ggvis()

# The pipe %>% operator

```
library(ggvis)  
ggvis(mtcars)  
mtcars %>% ggvis()
```

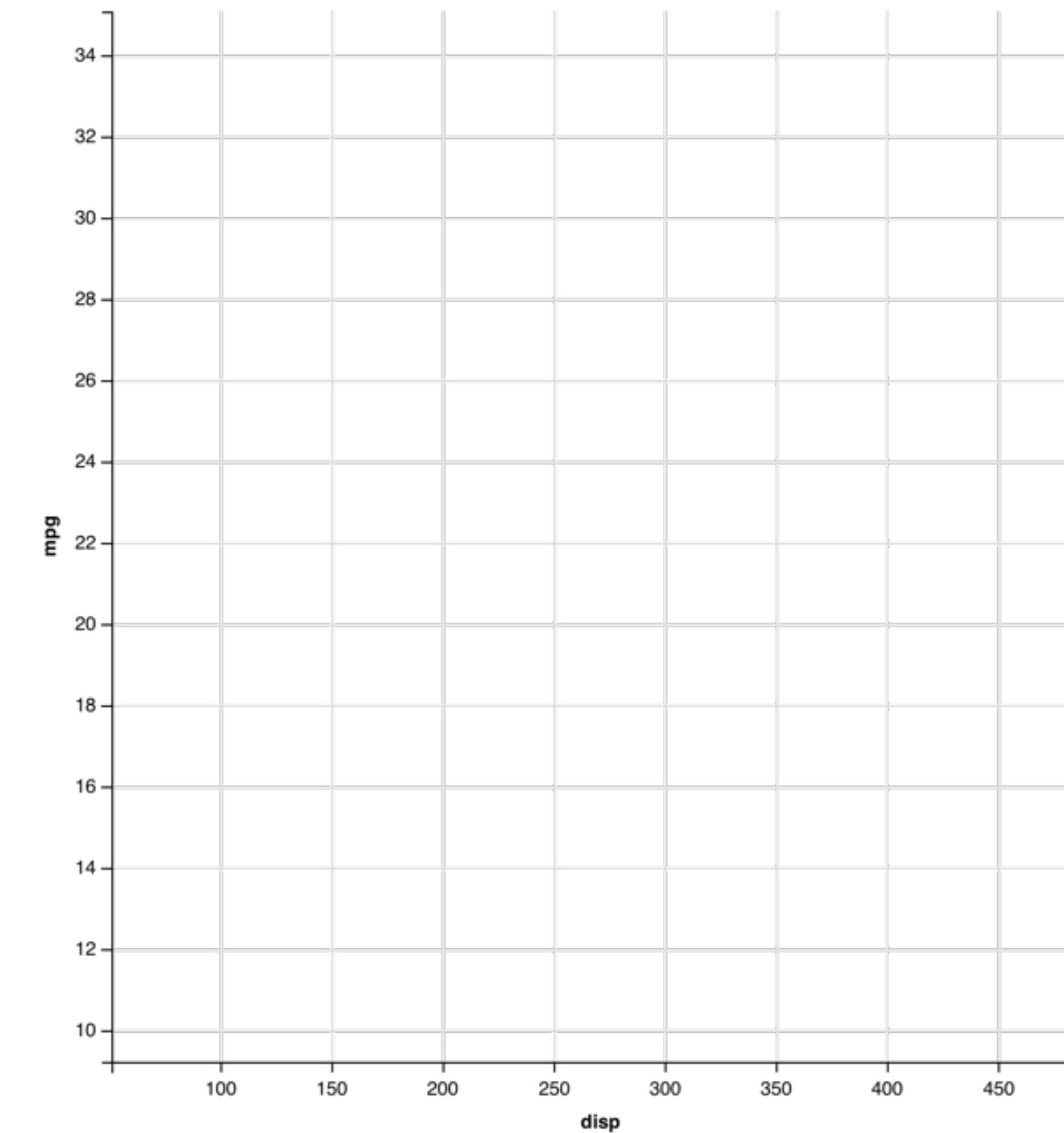
These do the same thing



mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

**data**

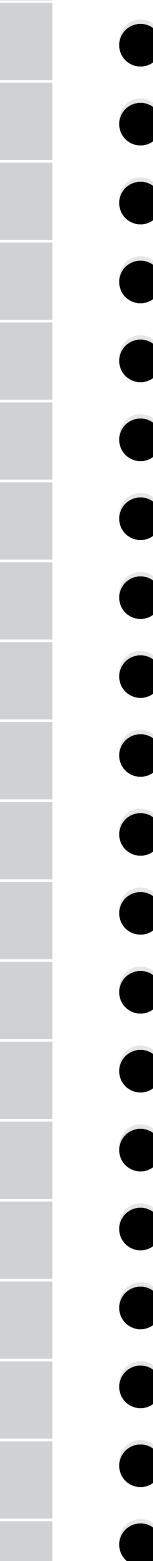
**mark**



**coordinate  
system**

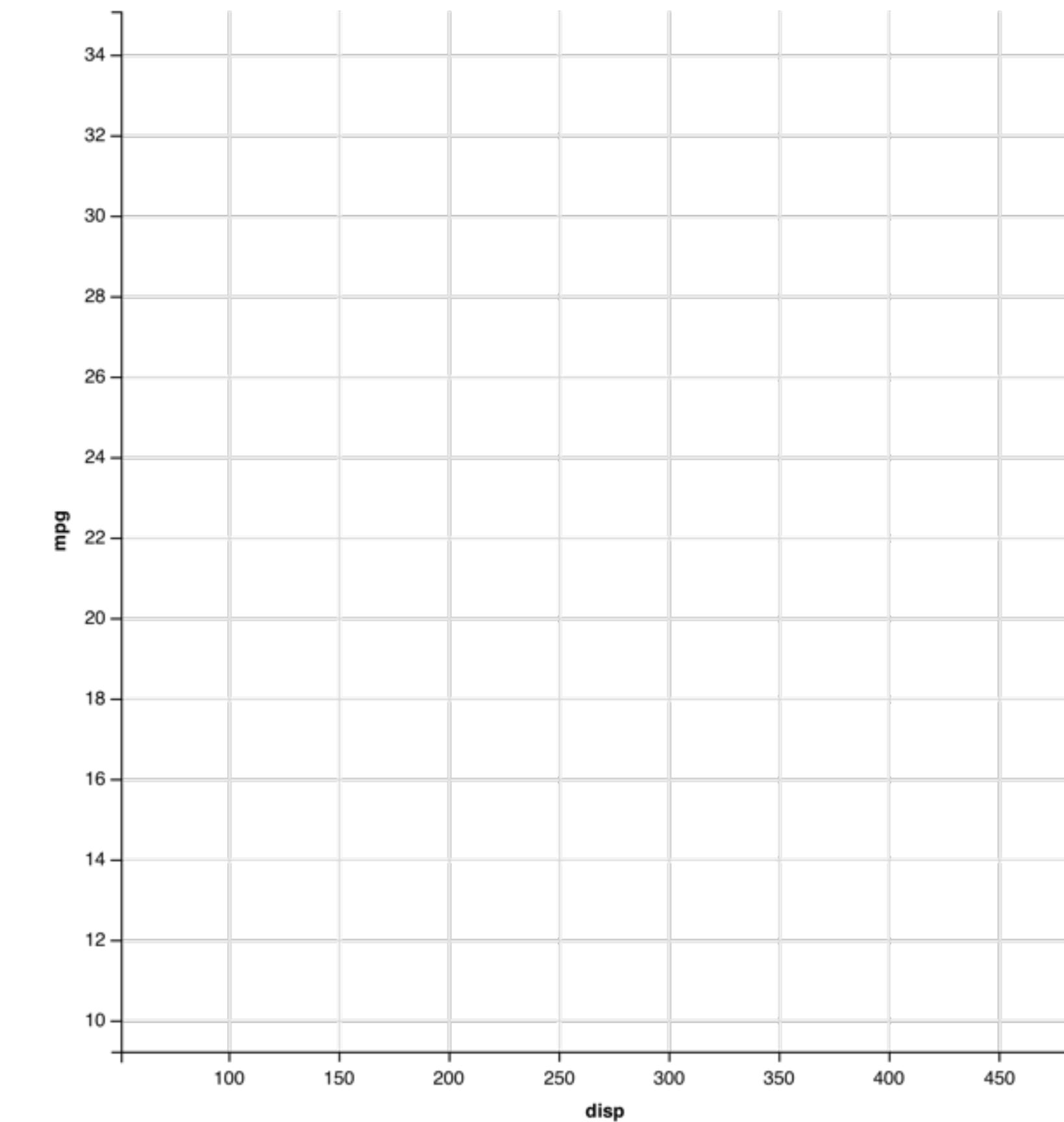
## properties

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1



**data**

**mark**

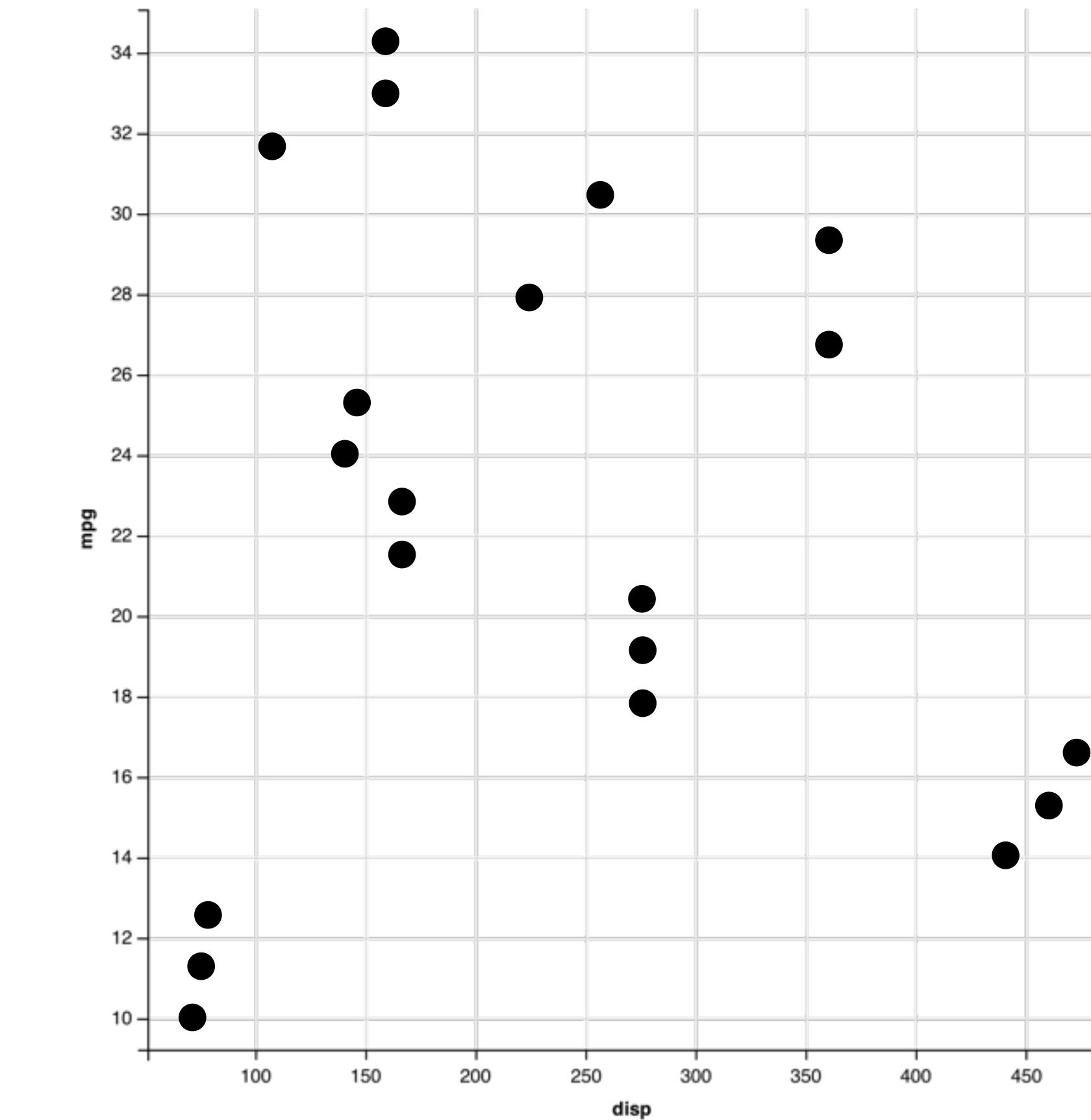


**coordinate  
system**

## properties

y  
↑  
**mpg**   cyl   disp   hp

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1



**data**

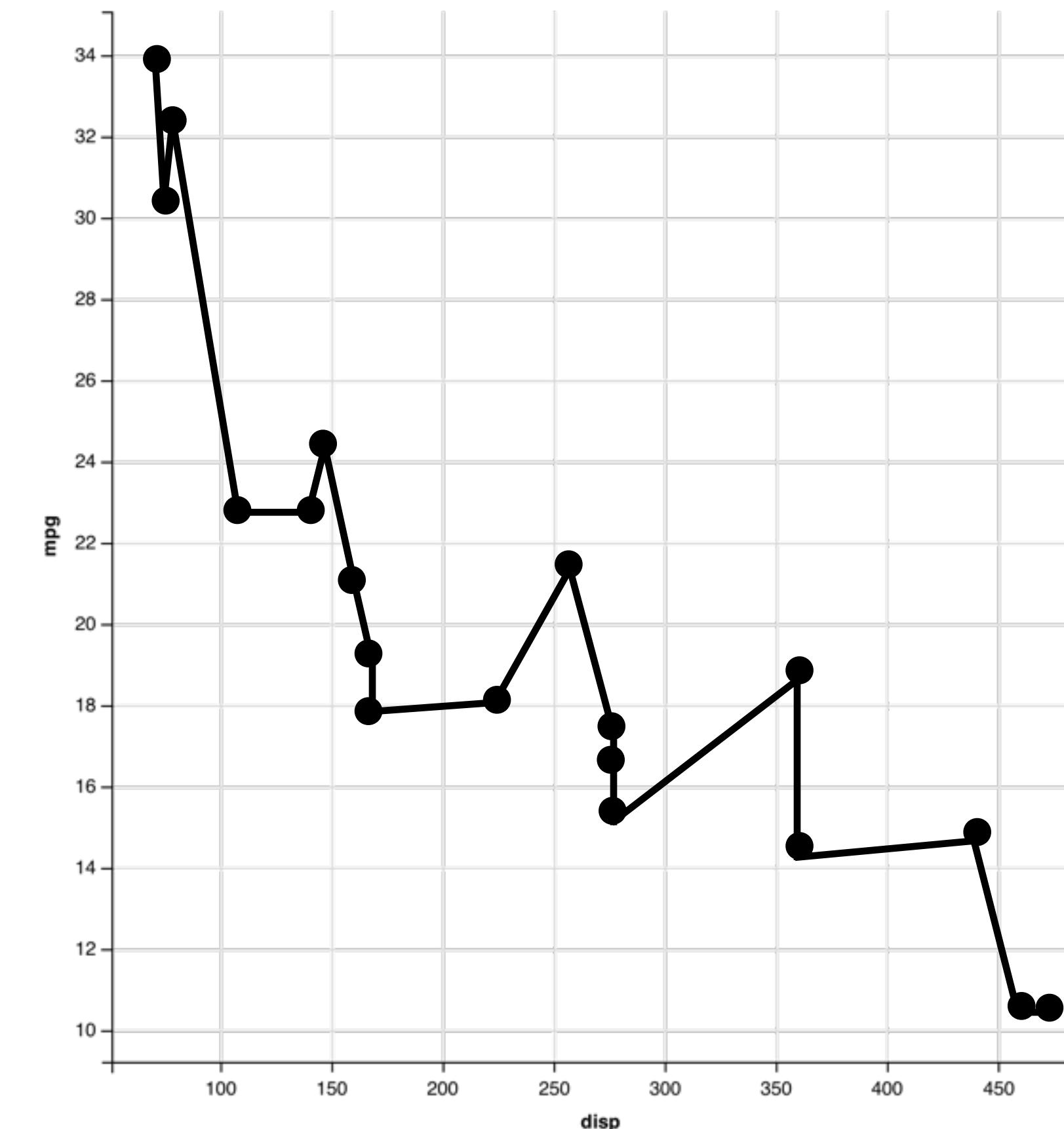
**mark**

**coordinate system**

## properties

Y  
↑  
X  
↓

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1



**data**

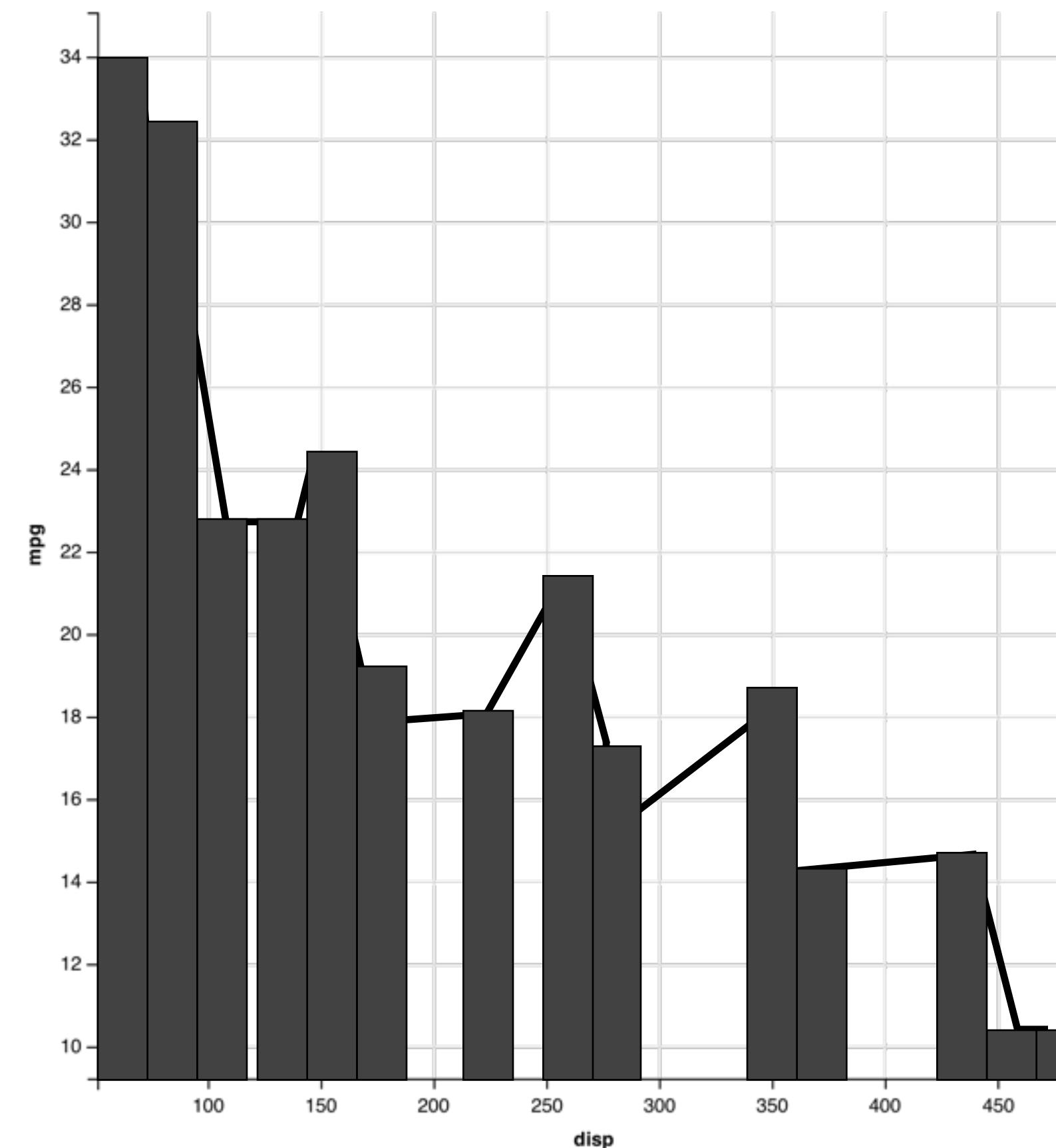
**mark**  
points  
lines

**coordinate  
system**

## properties

Y  
↑ ↓  
X  
↑ ↓

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1



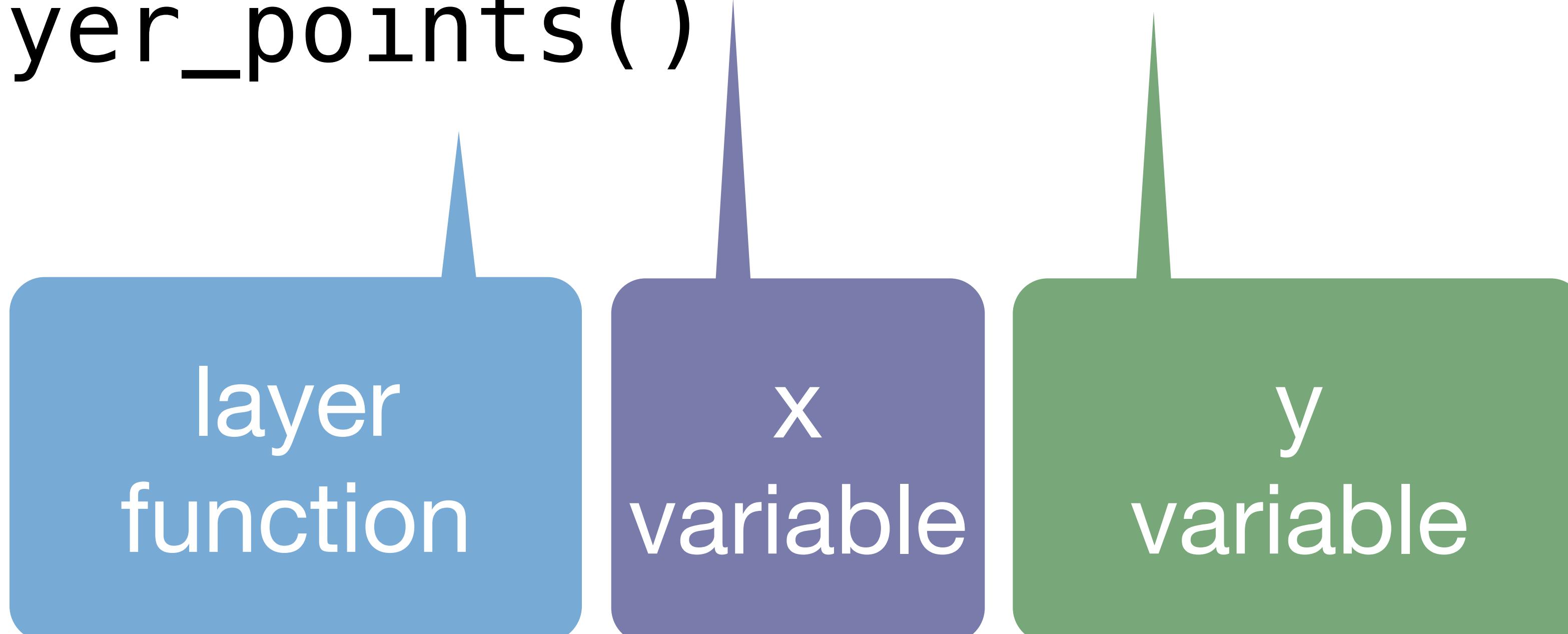
**data**

**mark**  
points  
lines  
rects

**coordinate**  
**system**

# Add marks

```
mtcars %>%  
  ggvis(x = ~disp, y = ~mpg) %>%  
  layer_points()
```



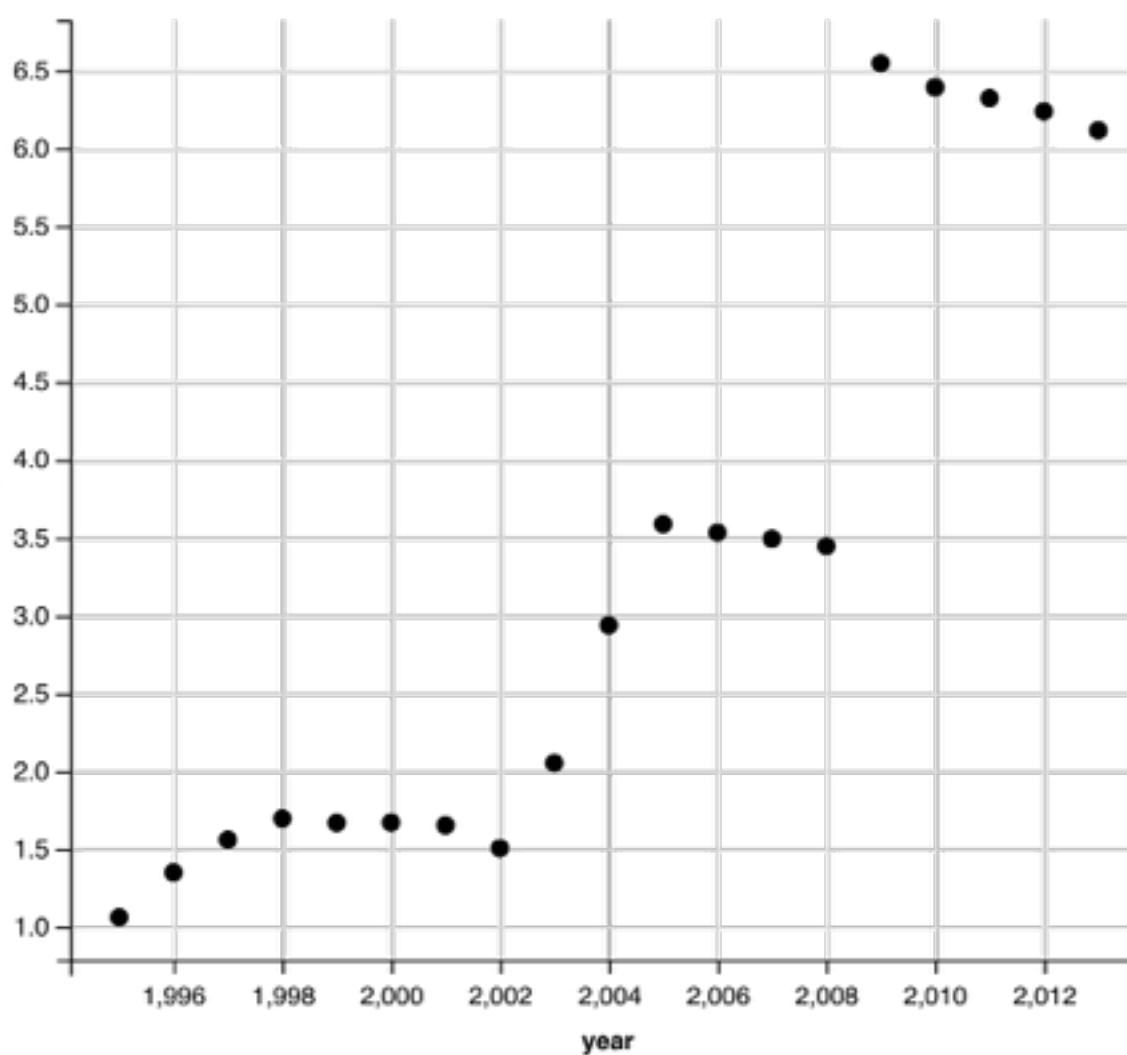
layer  
function

x  
variable

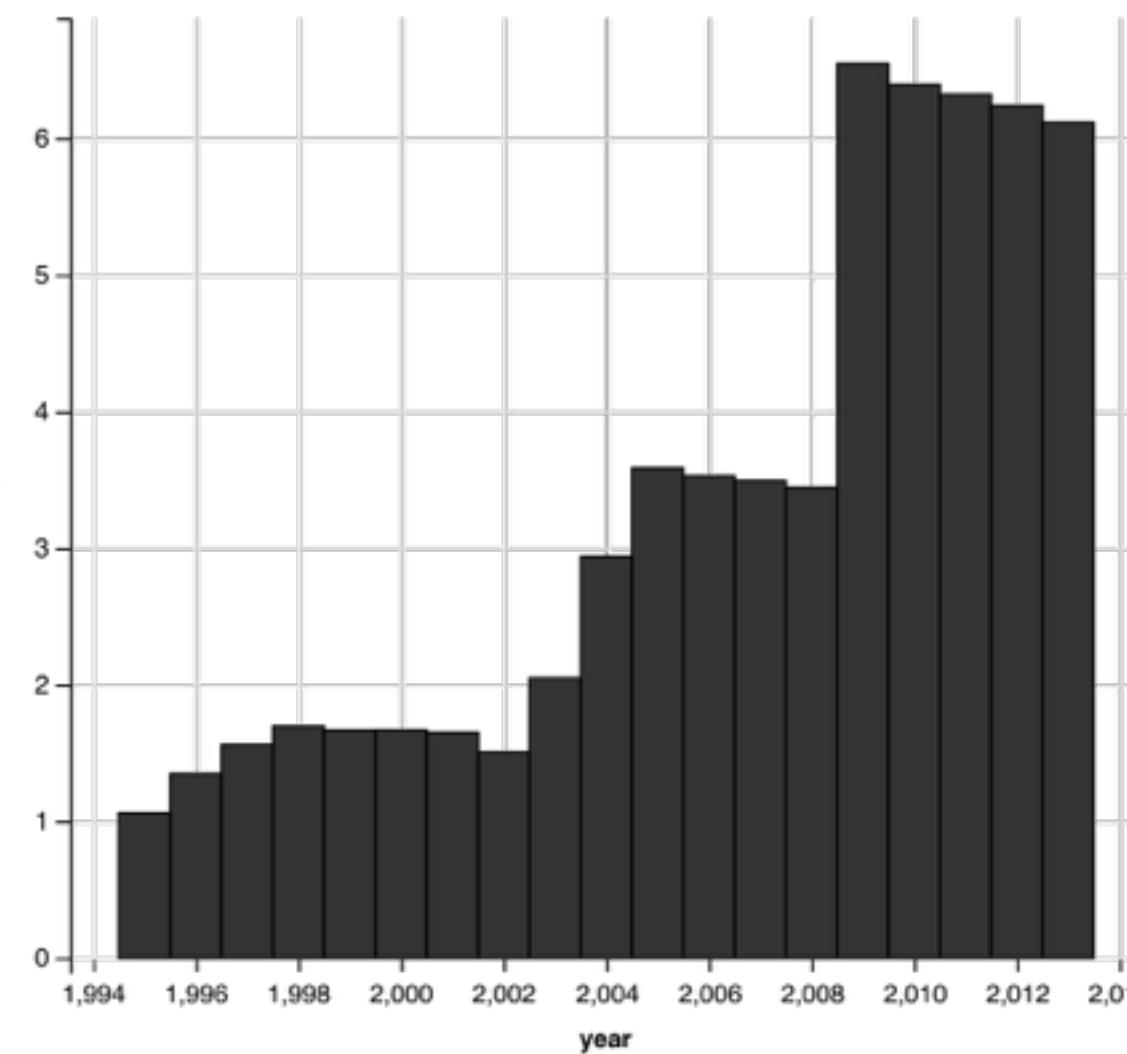
y  
variable

```
mtcars %>%  
  ggvis(x = ~disp, y = ~mpg) %>%  
  layer_points()  
  
layer_points(ggvis(mtcars, x = ~disp, y = ~mpg))
```

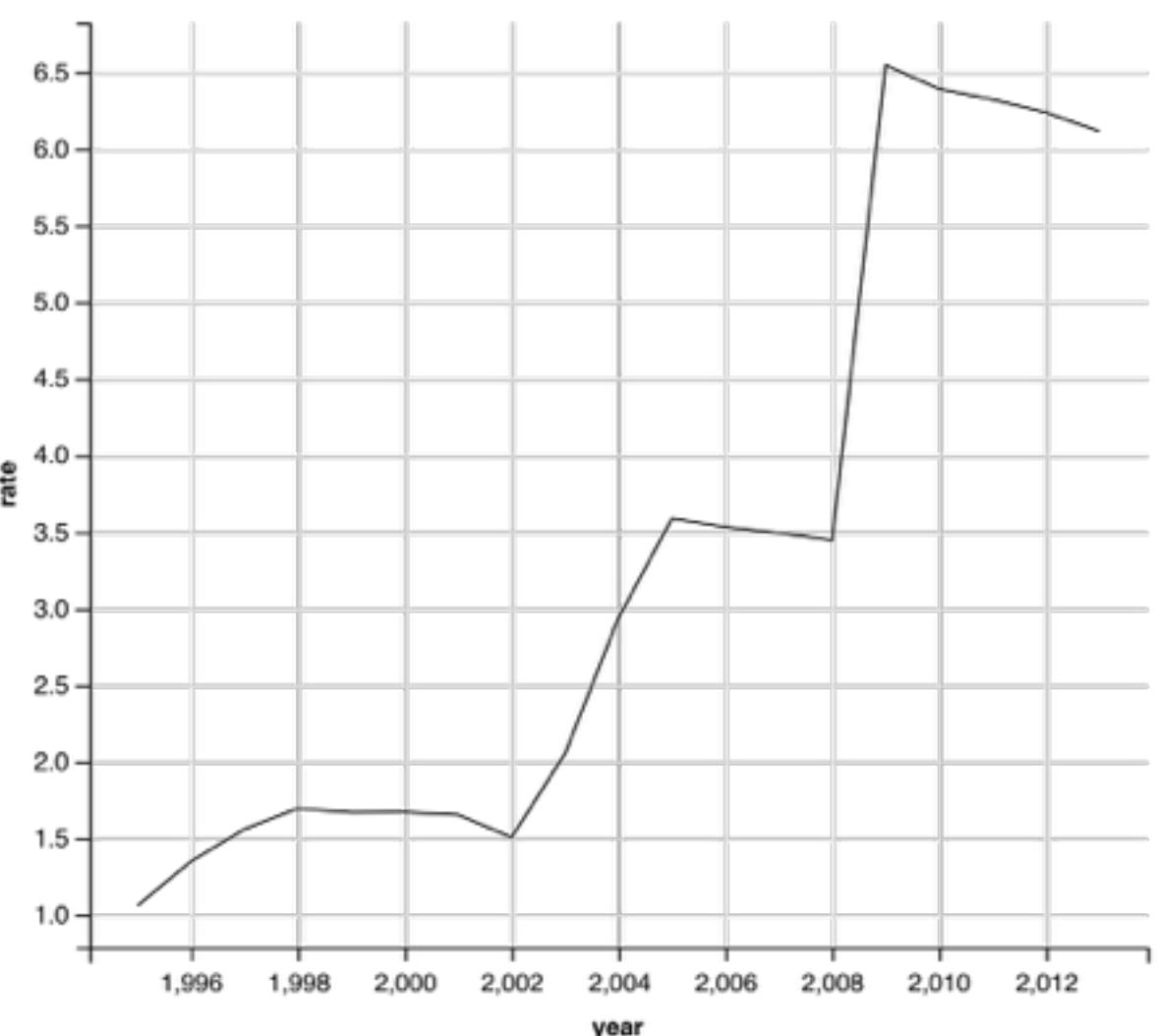
```
china %>%  
  ggvis(x = ~year,  
        y = ~rate) %>%  
layer_points()
```



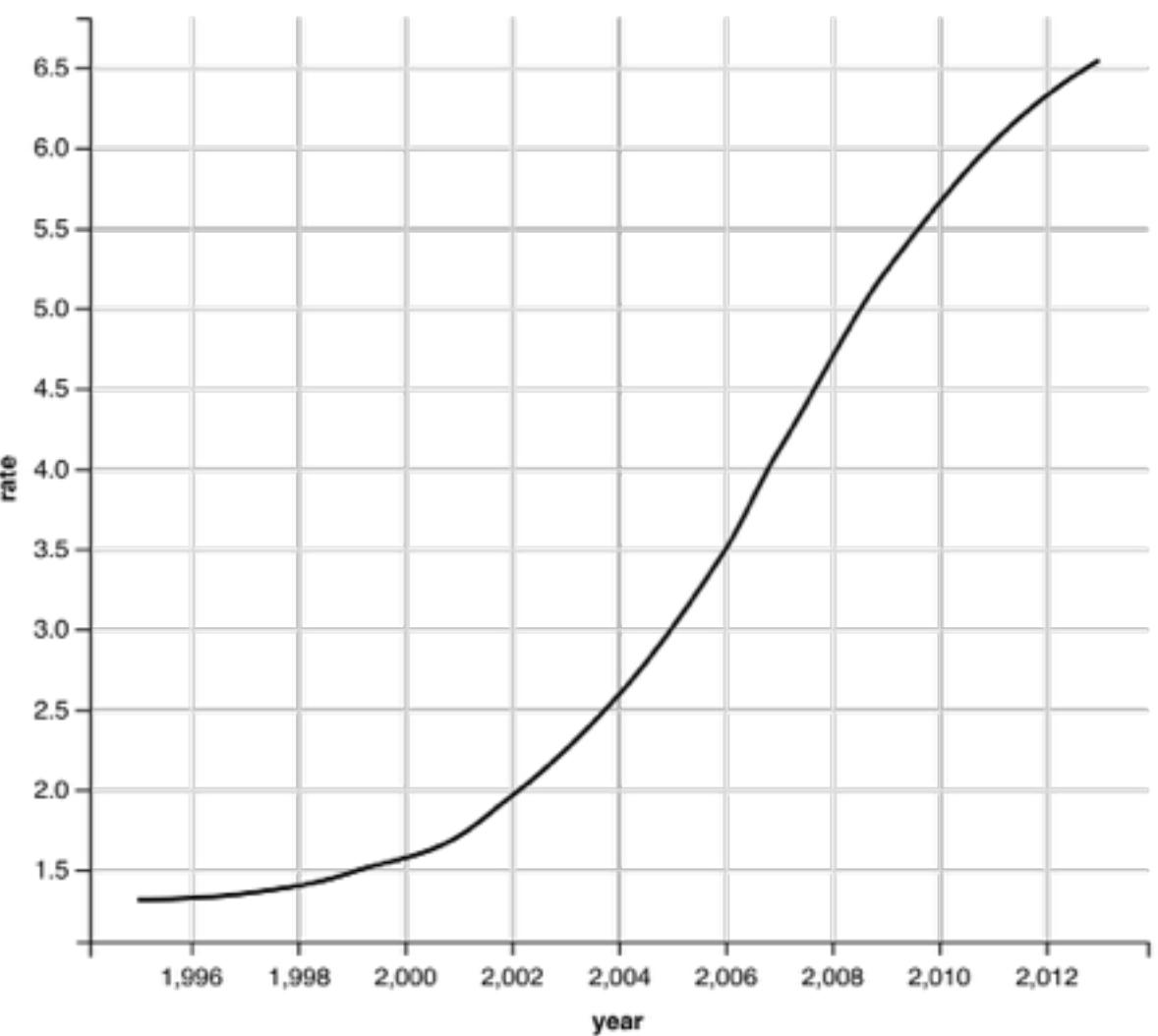
```
china %>%  
  ggvis(x = ~year,  
        y = ~rate) %>%  
layer_bars()
```



```
china %>%  
  ggvis(x = ~year,  
        y = ~rate) %>%  
layer_lines()
```

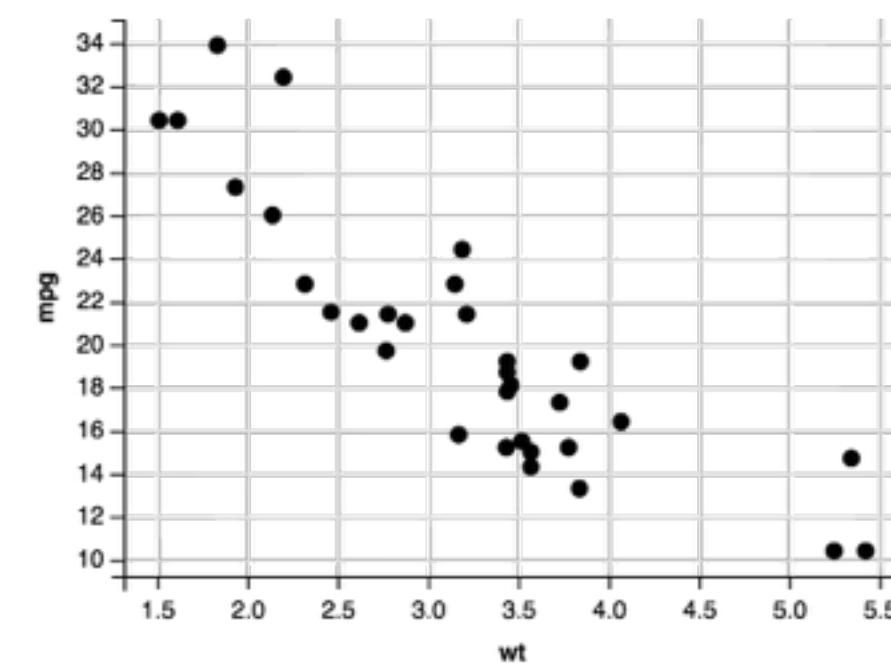


```
china %>%  
  ggvis(x = ~year,  
        y = ~rate) %>%  
layer_smooths()
```

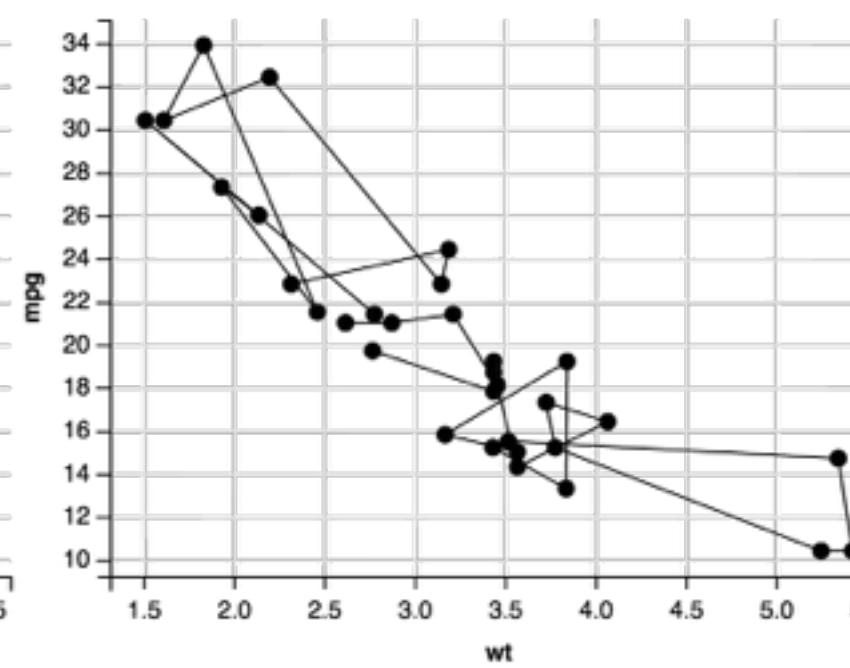


# Layers

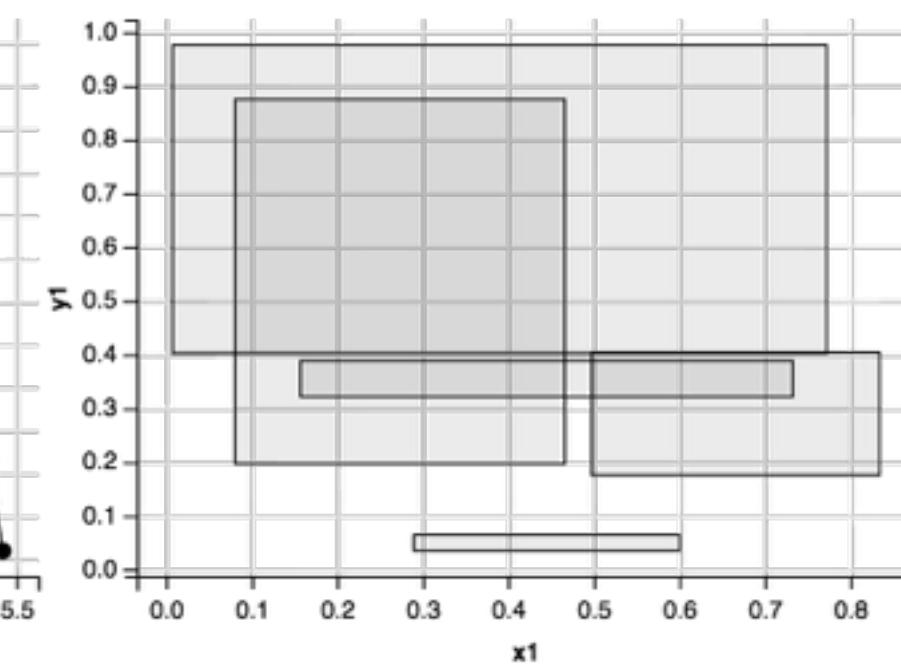
ggvis includes five layers to match the five basic marks of vega.



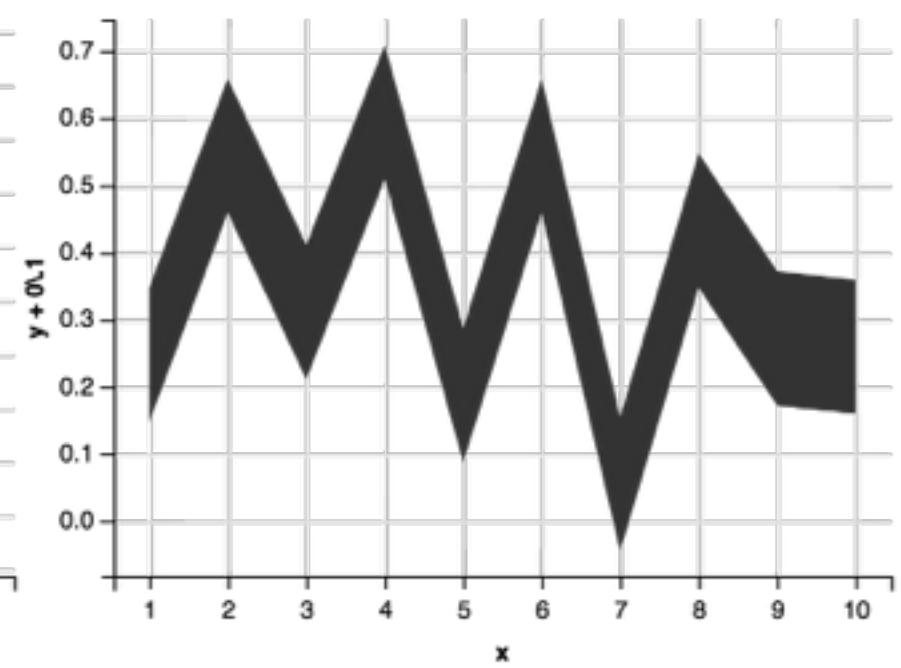
`layer_points`



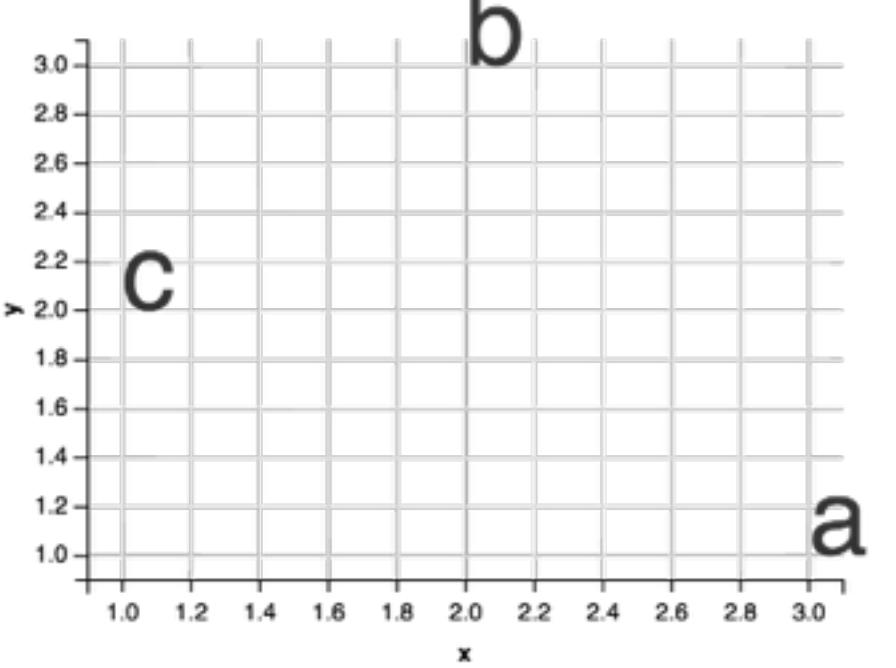
`layer_paths`



`layer_rects`



`layer_ribbons`



`a`

`b`

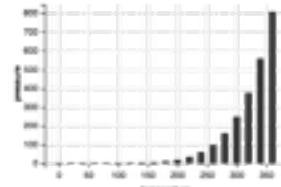
`c`

# Layers

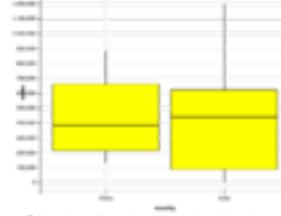
ggvis includes five layers to match the five basic marks of vega.  
And nine more layers to match how data scientists plot data.



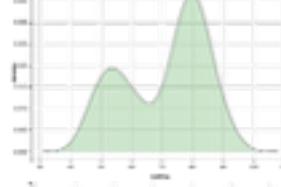
[layer\\_arcs](#)



[layer\\_bars](#)



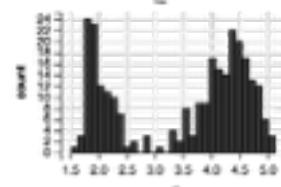
[layer\\_boxplots](#)



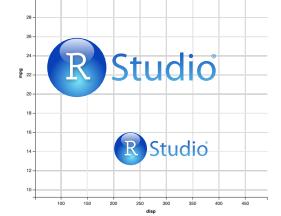
[layer\\_densities](#)



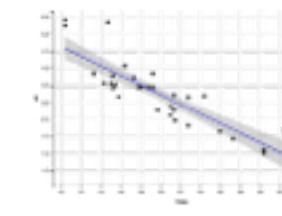
[layer\\_freqpolys](#)



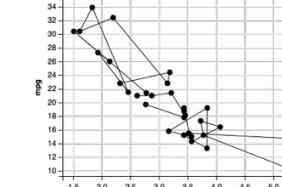
[layer\\_histograms](#)



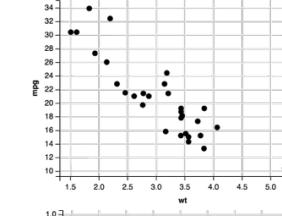
[layer\\_images](#)



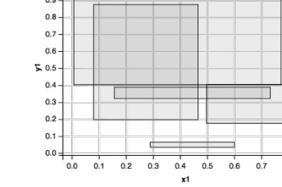
[layer\\_model\\_predictions](#)



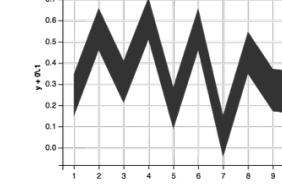
[layer\\_paths](#)



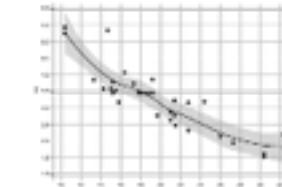
[layer\\_points](#)



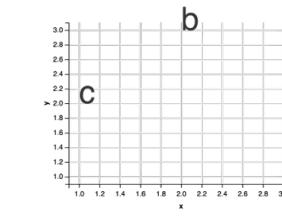
[layer\\_rects](#)



[layer\\_ribbons](#)



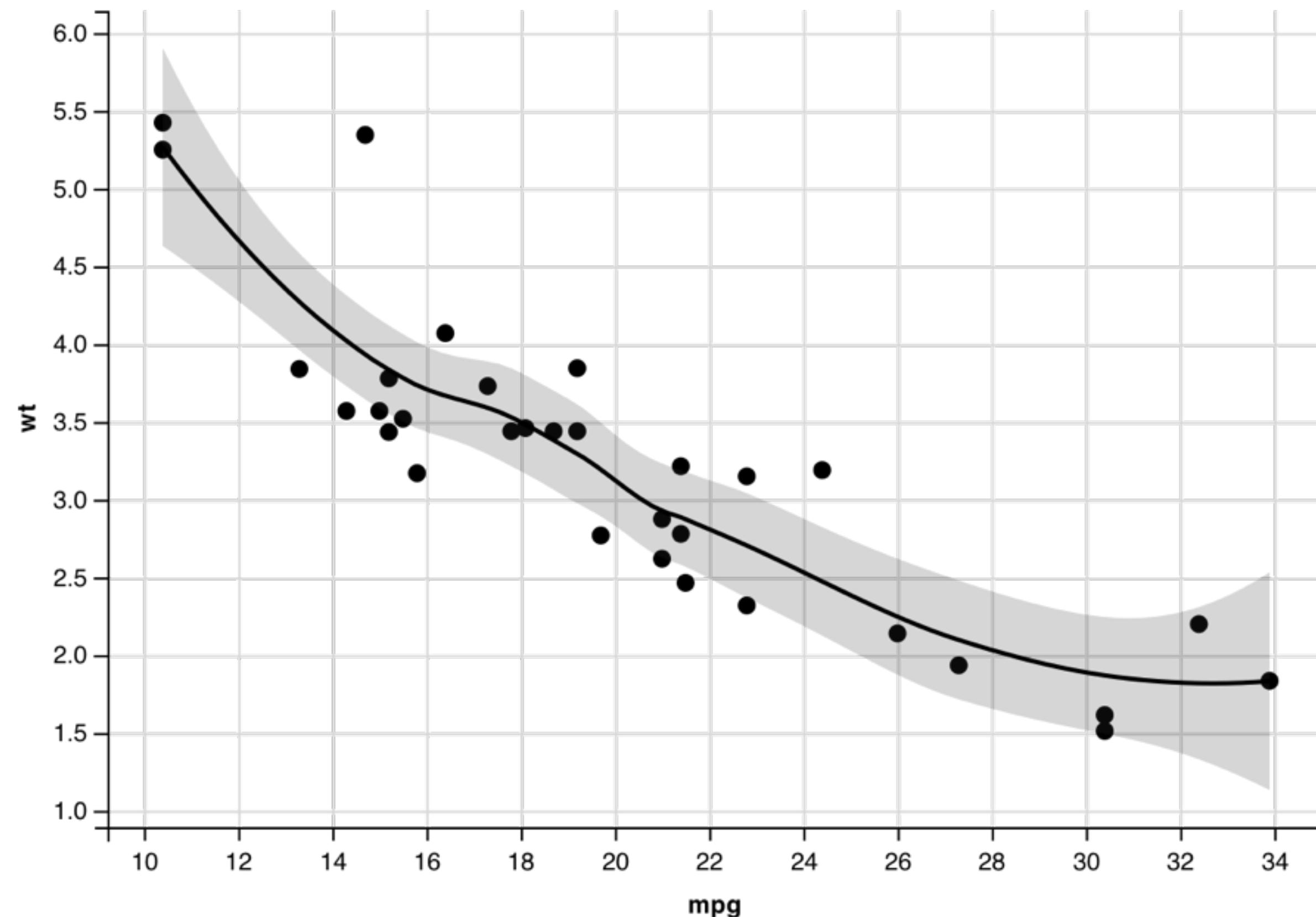
[layer\\_smooths](#)



[layer\\_text](#)

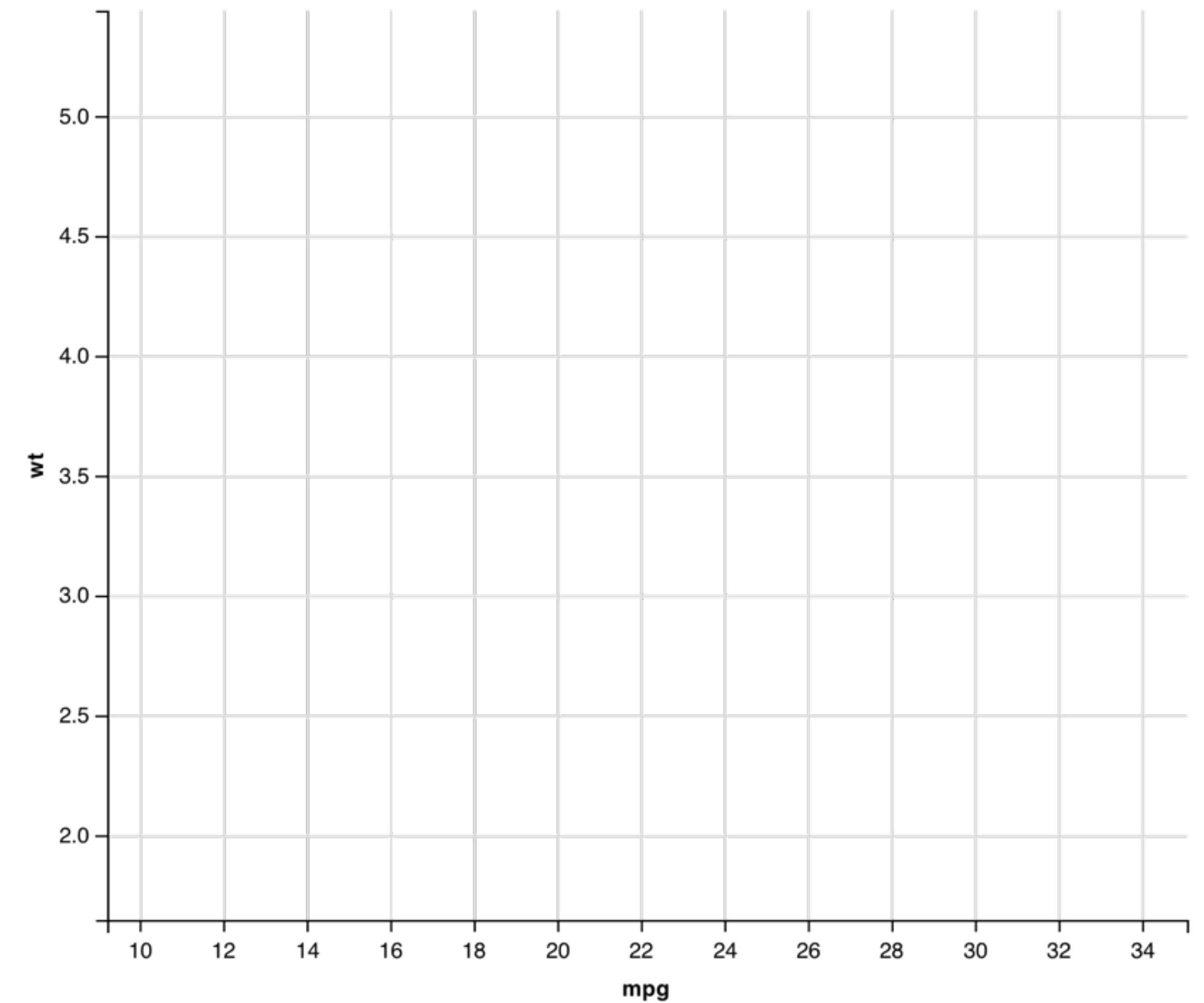
# Complex marks

**layer\_smooths** fits a smoothed line to a set of data. Here the smooth line is shown onto of the raw data, and with a standard error band.



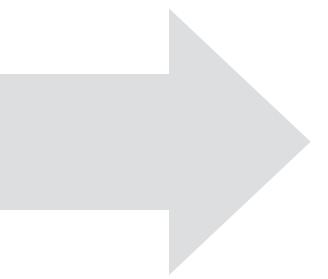
mpg	cyl	disp	wt
21.0	6	160.0	2.6
21.0	6	160.0	2.9
22.8	4	108.0	2.3
21.4	6	258.0	3.2
18.7	8	360.0	3.4
18.1	6	225.0	3.5
14.3	8	360.0	3.6
24.4	4	146.7	3.2
22.8	4	140.8	3.1
19.2	6	167.6	3.4
17.8	6	167.6	3.4
16.4	8	275.8	4.1
17.3	8	275.8	3.7
15.2	8	275.8	3.8
10.4	8	472.0	5.2
10.4	8	460.0	5.4
14.7	8	440.0	5.3
32.4	4	78.7	2.2
30.4	4	75.7	1.6
33.9	4	71.1	1.8

**data**



**coordinate system**

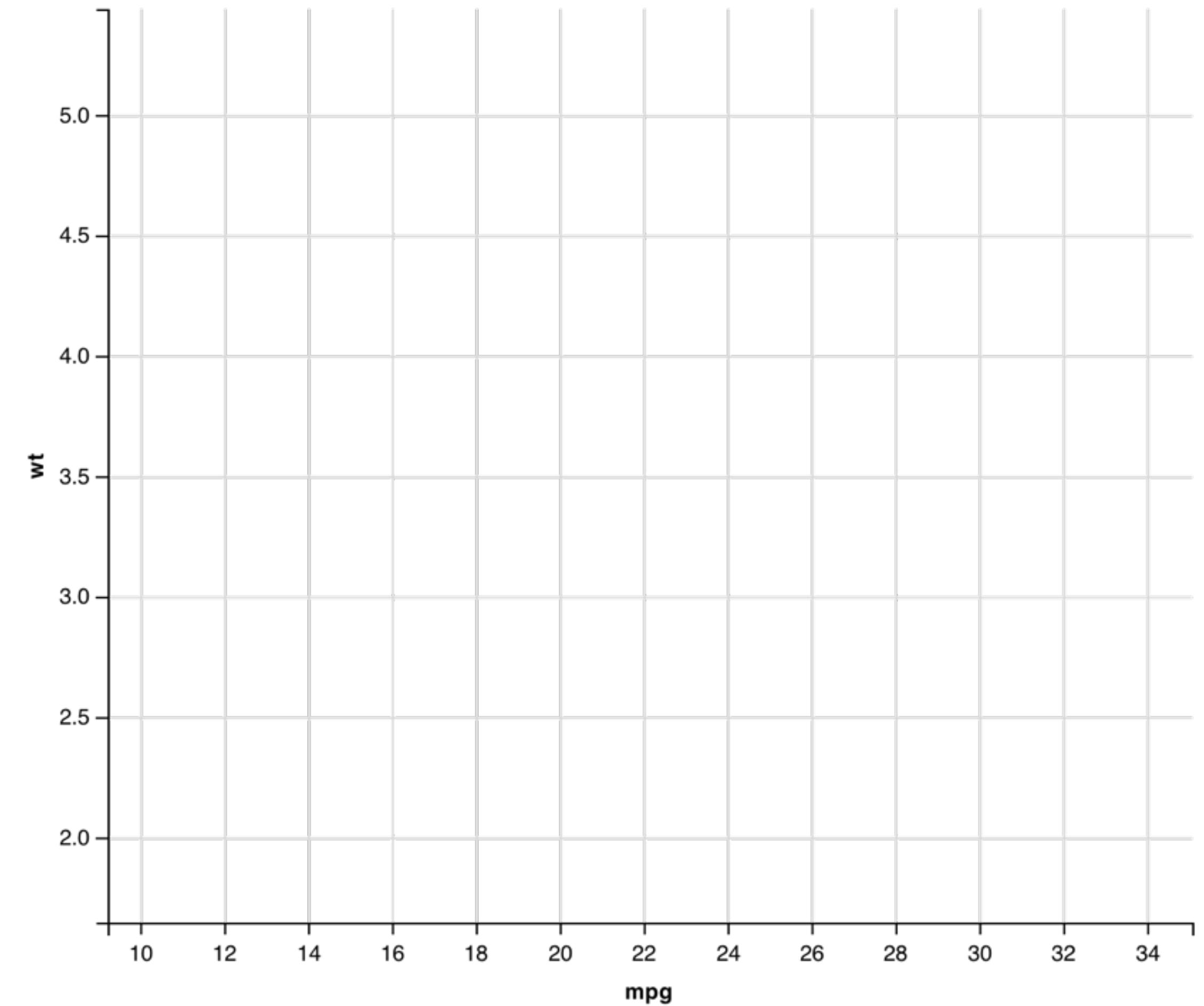
mpg	cyl	disp	wt
21.0	6	160.0	2.6
21.0	6	160.0	2.9
22.8	4	108.0	2.3
21.4	6	258.0	3.2
18.7	8	360.0	3.4
18.1	6	225.0	3.5
14.3	8	360.0	3.6
24.4	4	146.7	3.2
22.8	4	140.8	3.1
19.2	6	167.6	3.4
17.8	6	167.6	3.4
16.4	8	275.8	4.1
17.3	8	275.8	3.7
15.2	8	275.8	3.8
10.4	8	472.0	5.2
10.4	8	460.0	5.4
14.7	8	440.0	5.3
32.4	4	78.7	2.2
30.4	4	75.7	1.6
33.9	4	71.1	1.8



pred_	resp_
1.51	32.09
1.56	31.69
1.61	31.28
1.66	30.87
1.71	30.45
1.76	30.03
1.81	29.61
1.86	29.18
1.91	28.74
1.96	28.30
2.01	27.83
2.06	27.35
2.11	26.84
2.16	26.33
2.21	25.82
2.26	25.30
2.31	24.79
2.35	24.30
2.40	23.82
2.45	23.37

**data**

**compute**  
smoothed  
predictions



**coordinate system**

mpg	cyl	disp	wt
21.0	6	160.0	2.6
21.0	6	160.0	2.9
22.8	4	108.0	2.3
21.4	6	258.0	3.2
18.7	8	360.0	3.4
18.1	6	225.0	3.5
14.3	8	360.0	3.6
24.4	4	146.7	3.2
22.8	4	140.8	3.1
19.2	6	167.6	3.4
17.8	6	167.6	3.4
16.4	8	275.8	4.1
17.3	8	275.8	3.7
15.2	8	275.8	3.8
10.4	8	472.0	5.2
10.4	8	460.0	5.4
14.7	8	440.0	5.3
32.4	4	78.7	2.2
30.4	4	75.7	1.6
33.9	4	71.1	1.8

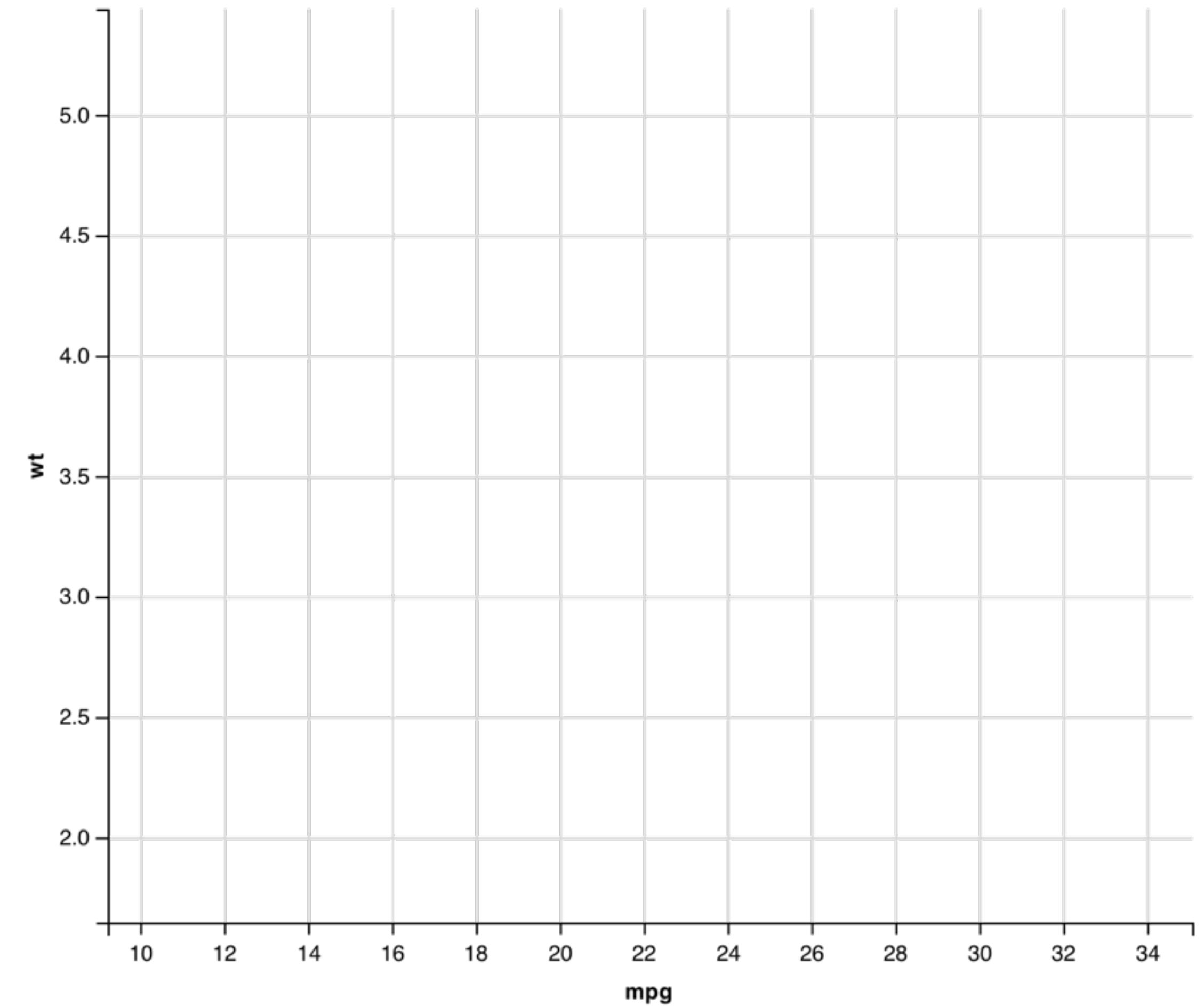
**compute**  
smoothed  
predictions

pred_	resp_
1.51	32.09
1.56	31.69
1.61	31.28
1.66	30.87
1.71	30.45
1.76	30.03
1.81	29.61
1.86	29.18
1.91	28.74
1.96	28.30
2.01	27.83
2.06	27.35
2.11	26.84
2.16	26.33
2.21	25.82
2.26	25.30
2.31	24.79
2.35	24.30
2.40	23.82
2.45	23.37

**data**

**mark**

**coordinate system**



## properties

mpg	cyl	disp	wt
21.0	6	160.0	2.6
21.0	6	160.0	2.9
22.8	4	108.0	2.3
21.4	6	258.0	3.2
18.7	8	360.0	3.4
18.1	6	225.0	3.5
14.3	8	360.0	3.6
24.4	4	146.7	3.2
22.8	4	140.8	3.1
19.2	6	167.6	3.4
17.8	6	167.6	3.4
16.4	8	275.8	4.1
17.3	8	275.8	3.7
15.2	8	275.8	3.8
10.4	8	472.0	5.2
10.4	8	460.0	5.4
14.7	8	440.0	5.3
32.4	4	78.7	2.2
30.4	4	75.7	1.6
33.9	4	71.1	1.8

x ↑  
y ↑

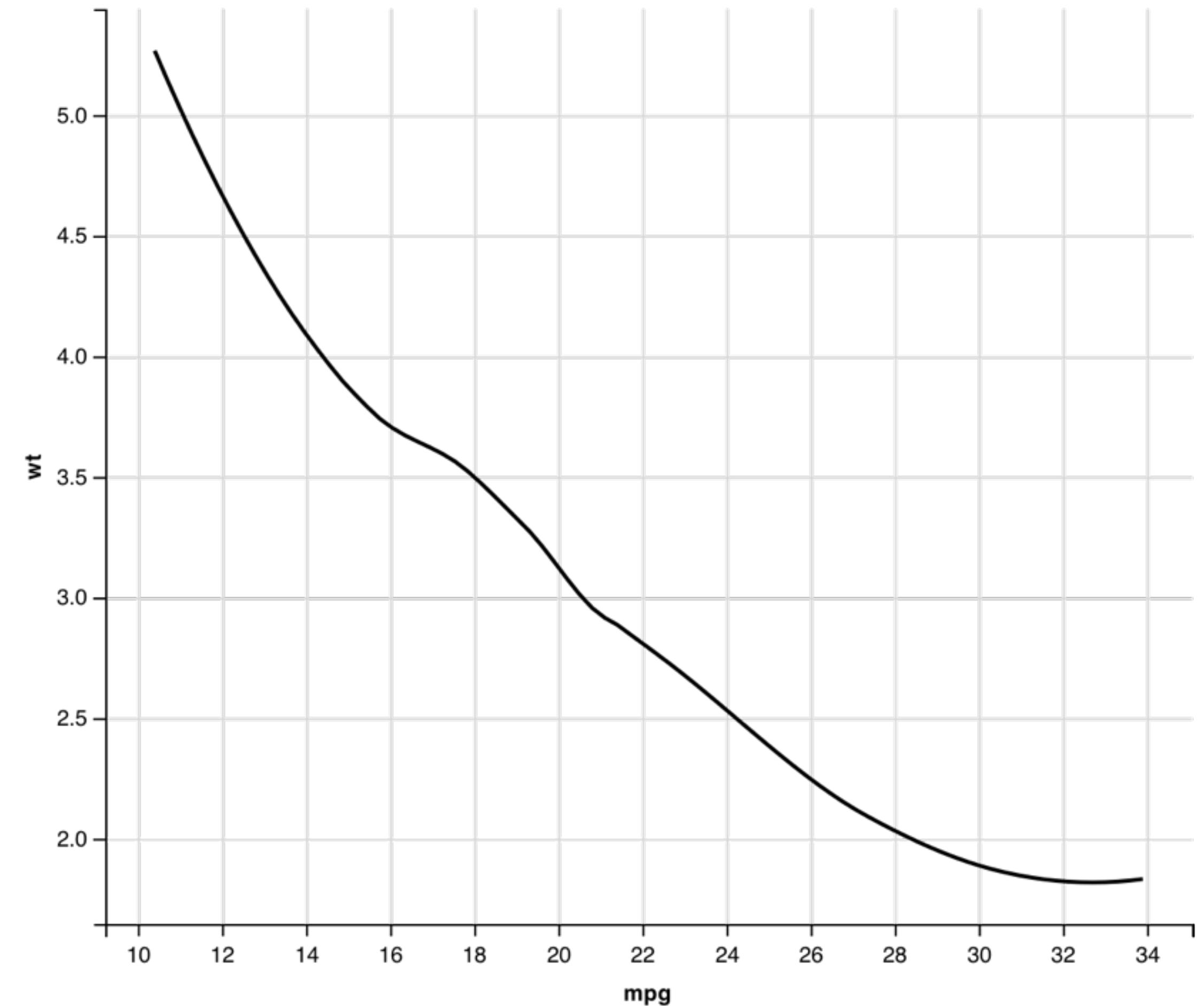
pred_	resp_
1.51	32.09
1.56	31.69
1.61	31.28
1.66	30.87
1.71	30.45
1.76	30.03
1.81	29.61
1.86	29.18
1.91	28.74
1.96	28.30
2.01	27.83
2.06	27.35
2.11	26.84
2.16	26.33
2.21	25.82
2.26	25.30
2.31	24.79
2.35	24.30
2.40	23.82
2.45	23.37

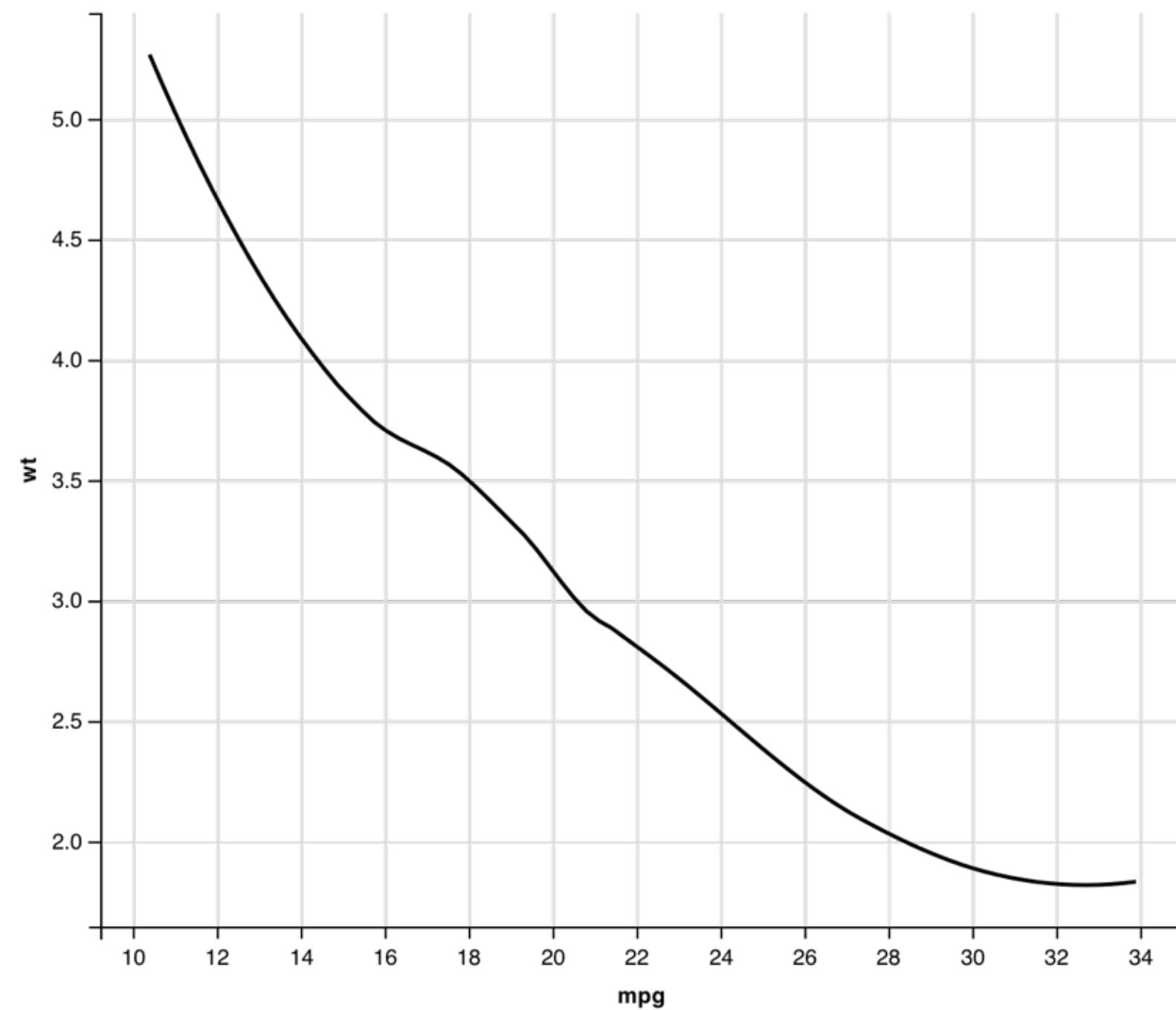
**data**

**compute**  
smoothed  
predictions

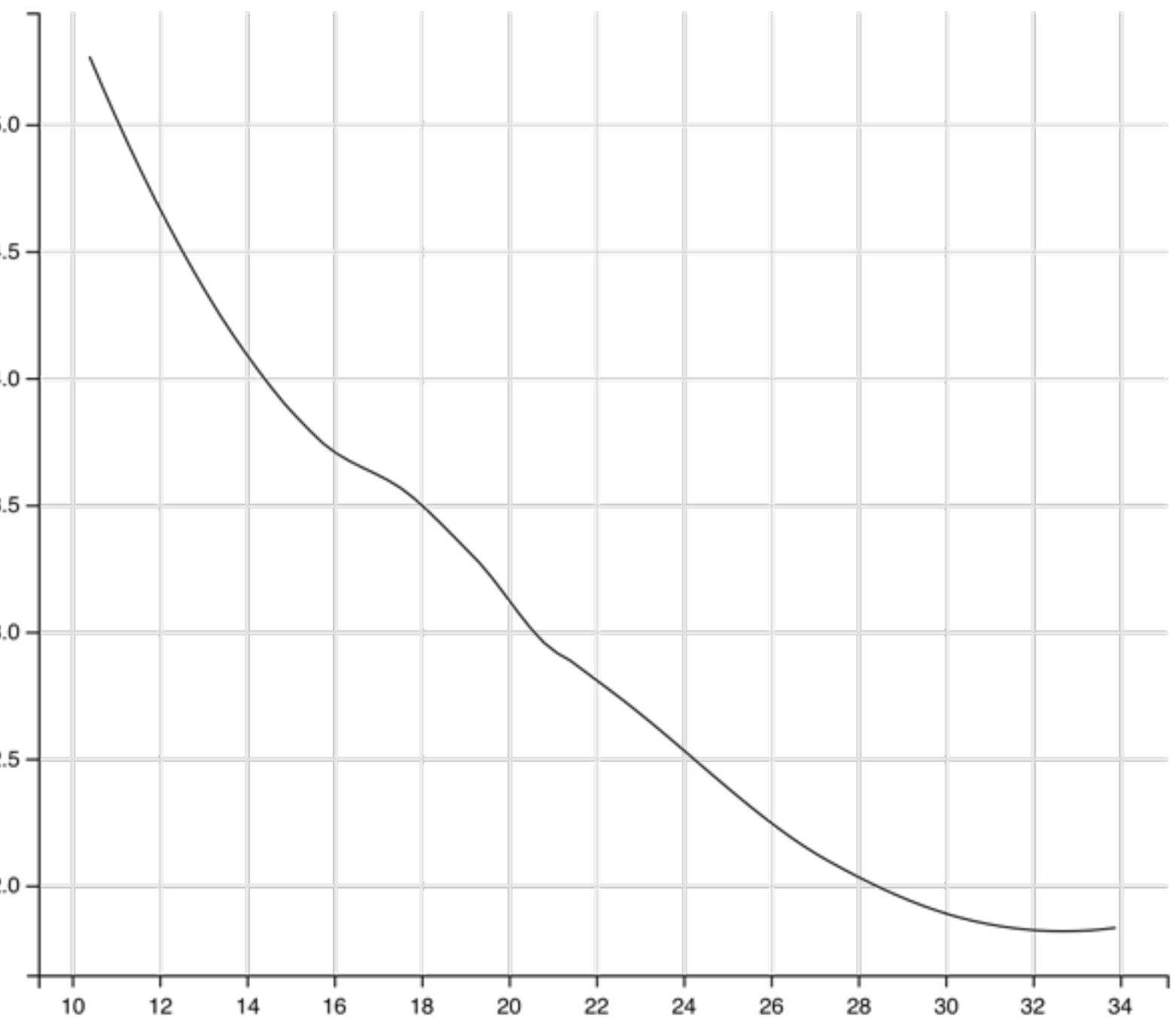
**mark**

**coordinate system**





```
mtcars %>%
  ggvis(~mpg, ~wt) %>%
  layer_smooths()
```



```
mtcars %>%
  compute_smooth(wt ~ mpg) %>%
  ggvis(~pred_, ~resp_) %>%
  layer_paths()
```

```
mtcars %>%
  compute_smooth(wt ~ mpg)
```

```
##   pred_   resp_
## 1 10.40000 5.267099
## 2 10.69747 5.147040
## 3 10.99494 5.031189
## 4 11.29241 4.919509
## 5 11.58987 4.811967
## 6 11.88734 4.708527
## ...
```

# Compute functions

`compute_align`

`compute_bin`

`compute_boxplot`

`compute_count`

`compute_density`

`compute_model_predictions`

`compute_smooth`

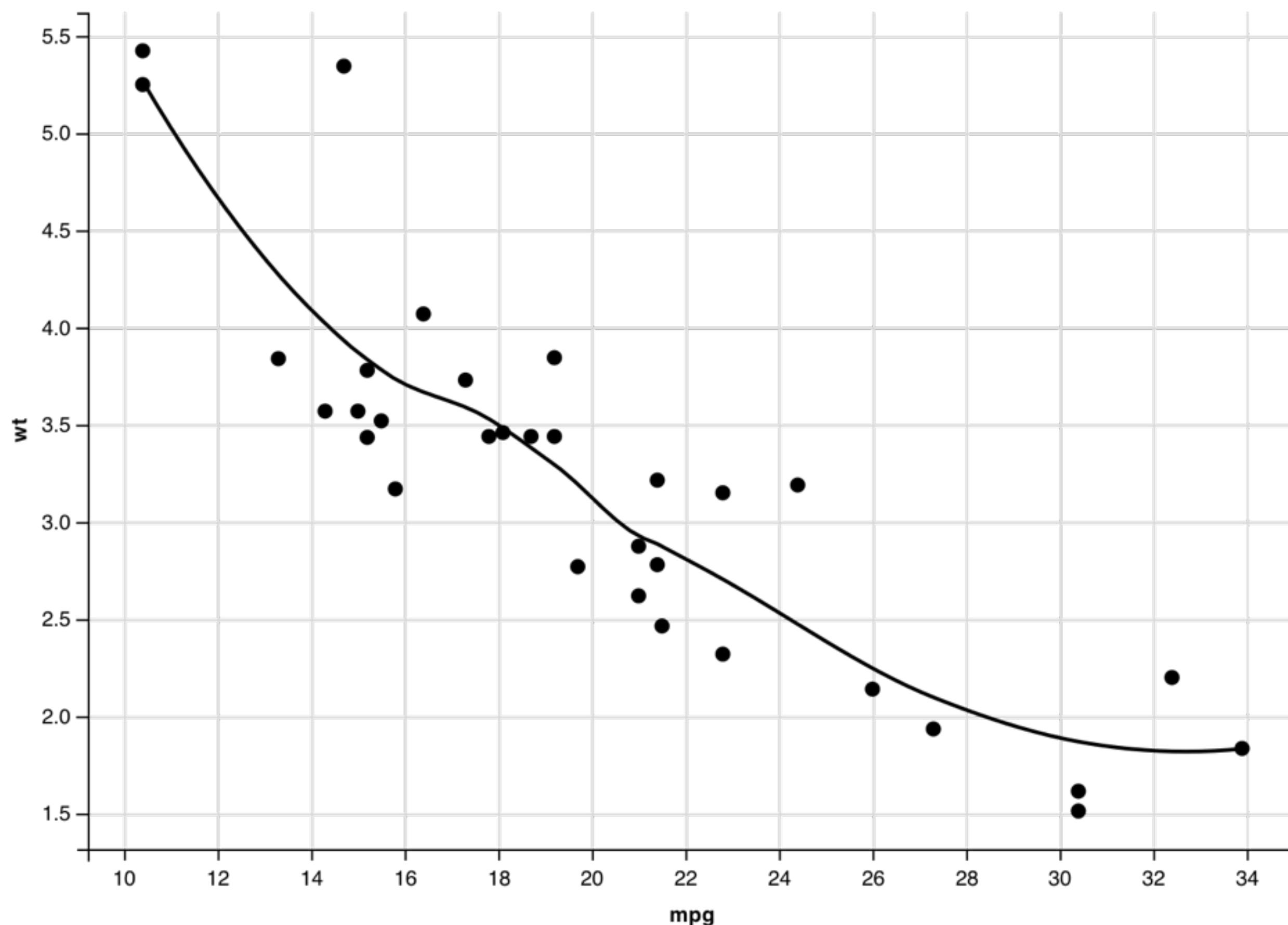
`compute_stack`

`compute_tabulate`

# Add layers

Add multiple layers to your plot by calling multiple layer functions.

```
mtcars %>%  
  ggviz(~mpg, ~wt) %>%  
  layer_smooths() %>%  
  layer_points()
```

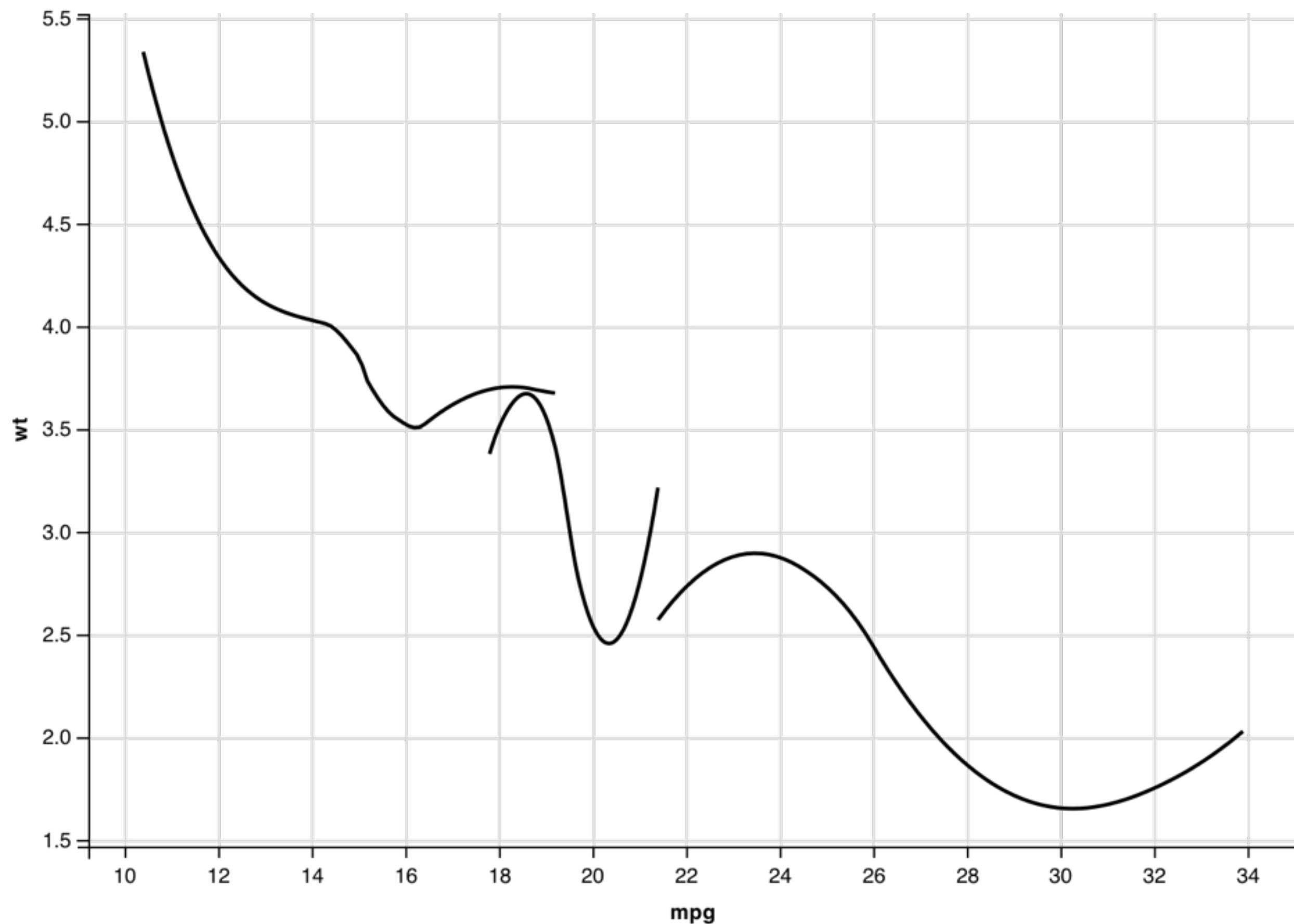


# Group observations

Draw a separate mark for each group with `dplyr::group_by()`

```
library(dplyr)
```

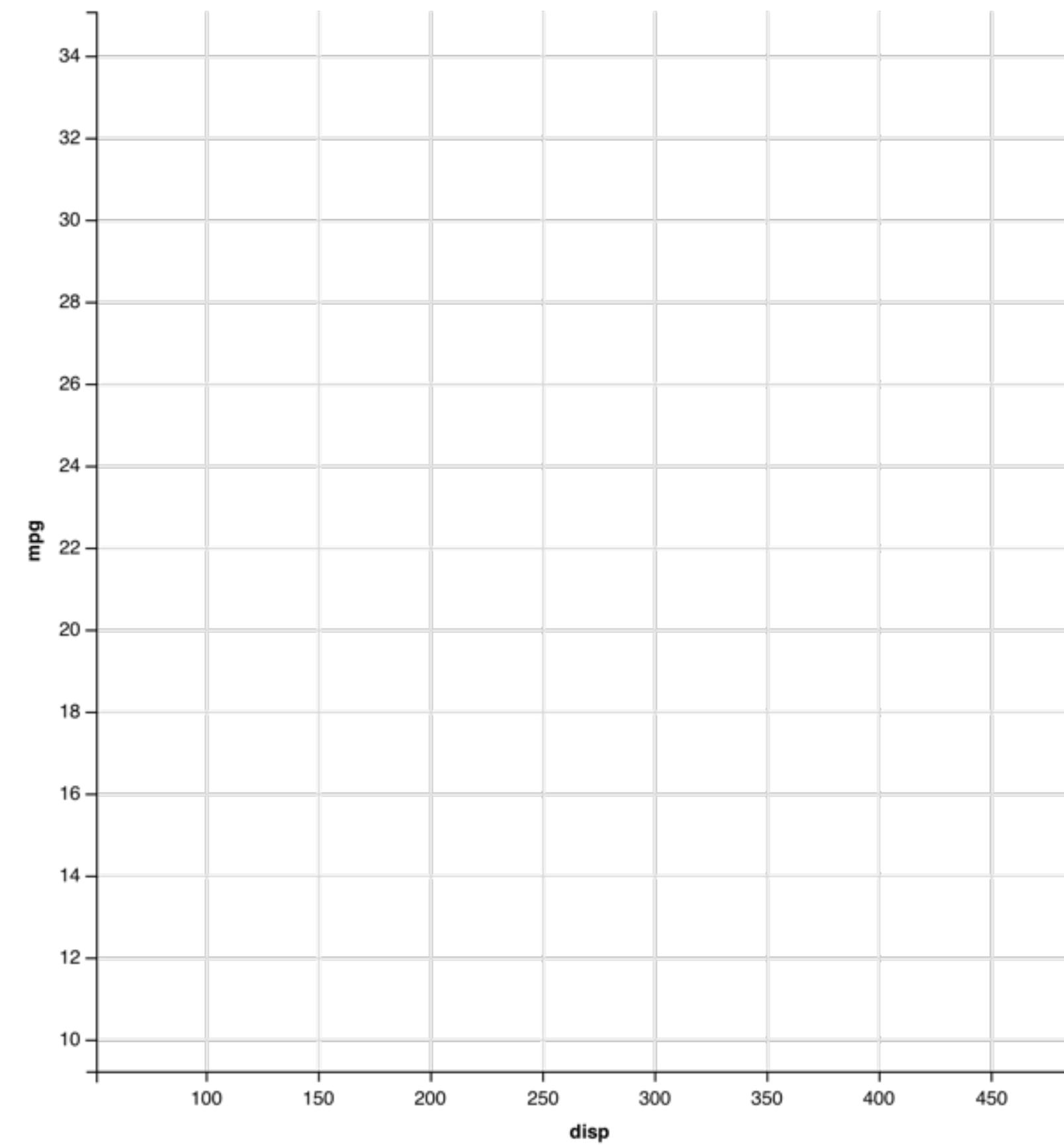
```
mtcars %>%  
  group_by(cyl) %>%  
  ggvis(~mpg, ~wt) %>%  
  layer_smooths()
```



# Adding variables

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

**data**

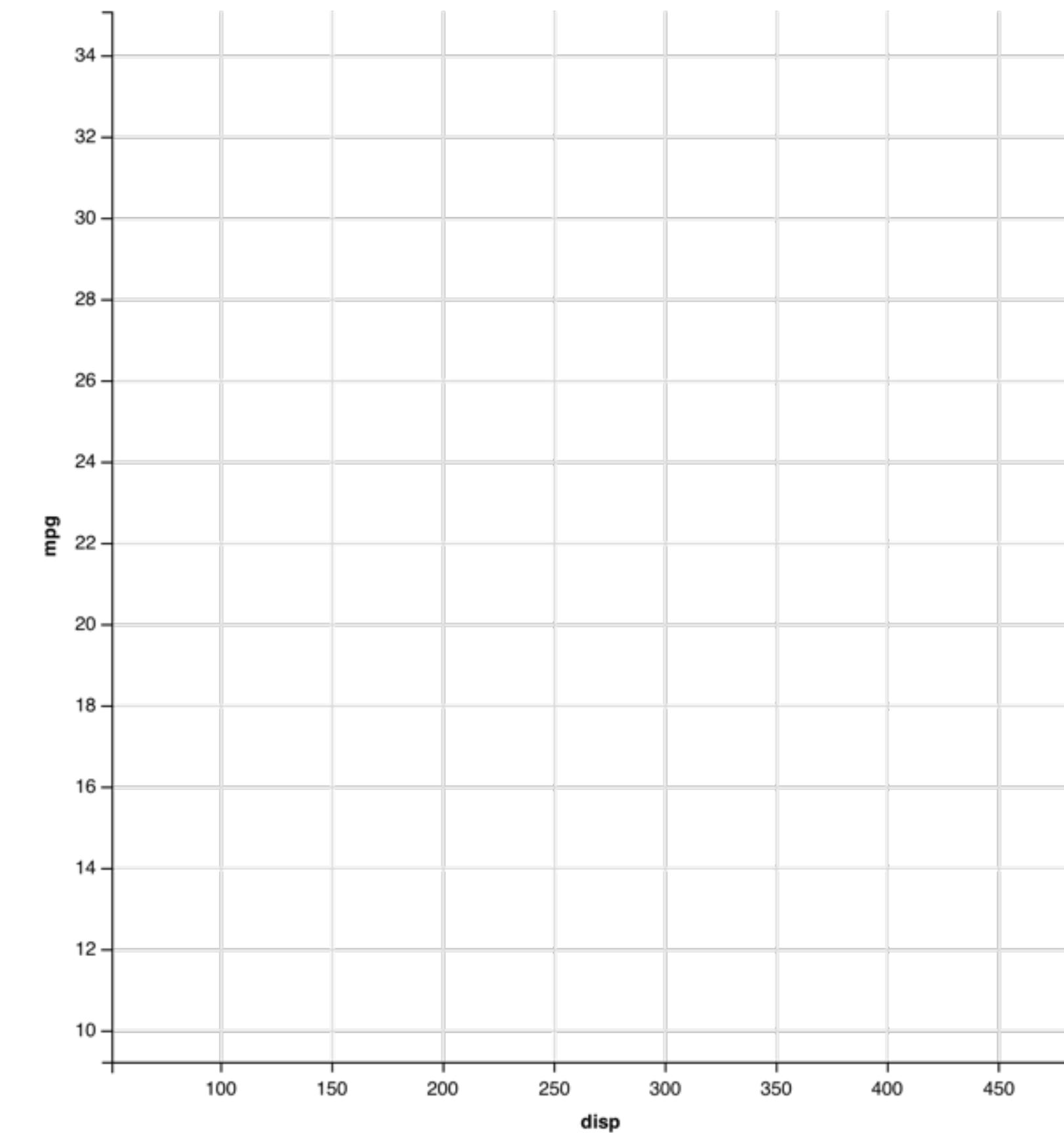


**coordinate  
system**

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

**data**

**mark**



**coordinate  
system**

# properties

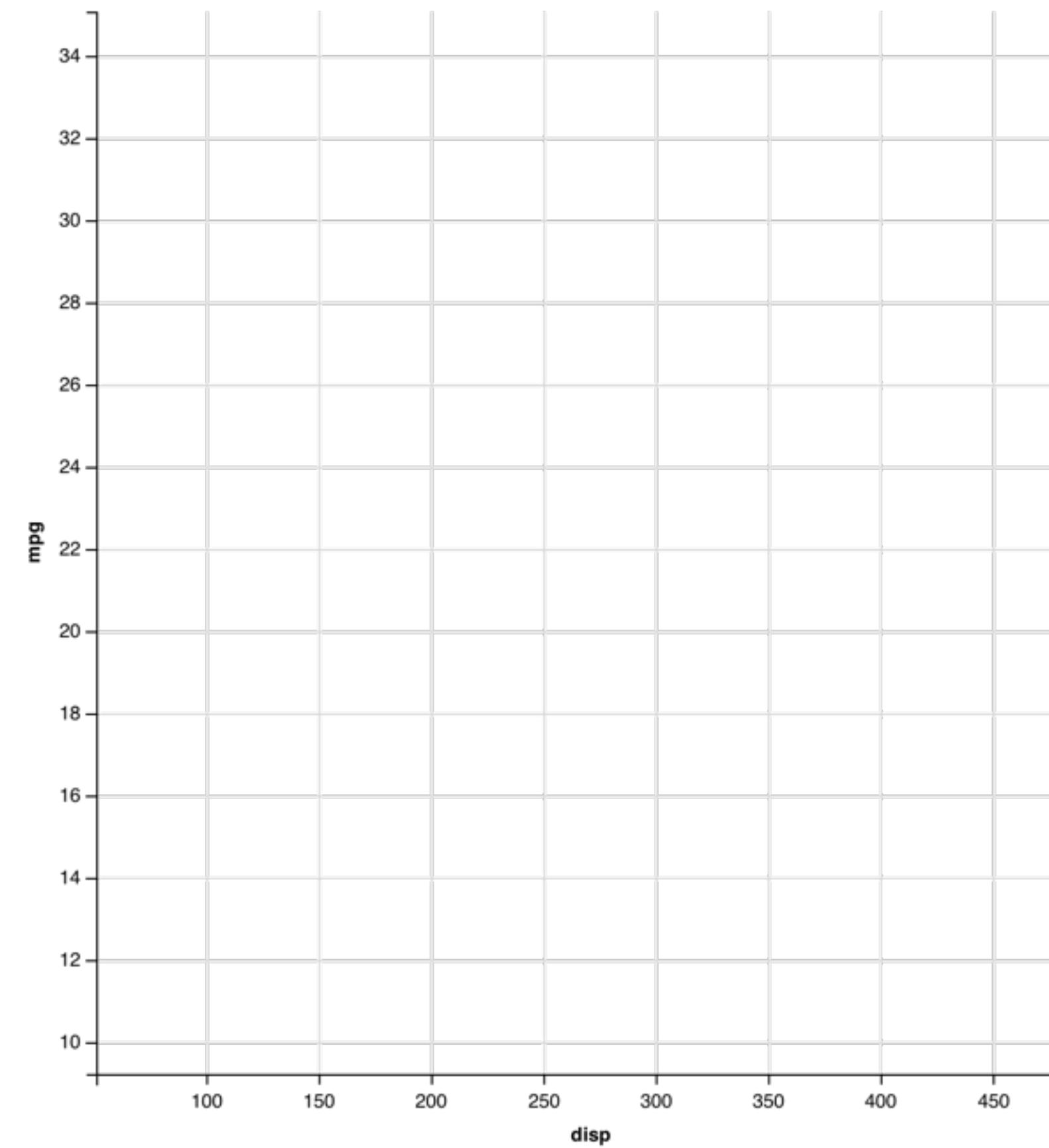


A vertical color scale bar on the right side of the table, showing a gradient from dark blue at the bottom to dark red at the top, with intermediate colors green, yellow, and light blue.

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

# data

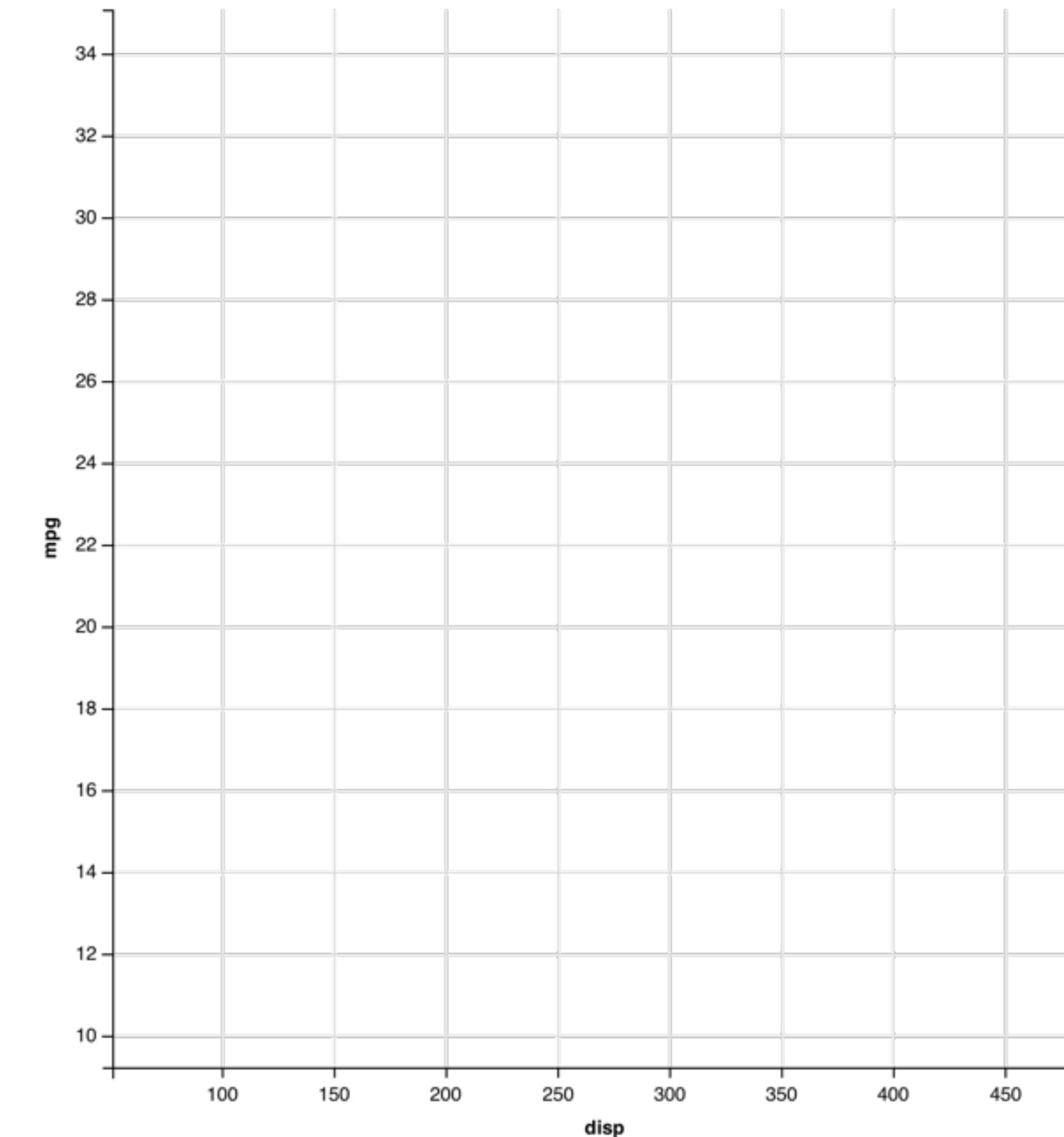
**mark**



# coordinate system

## properties

shape		fill	
mpg	cyl	disp	hp
21.0	6 +	160.0	2
21.0	6 +	160.0	2
22.8	4 ●	108.0	1
21.4	6 +	258.0	2
18.7	8 ♦	360.0	3
18.1	6 +	225.0	2
14.3	8 ♦	360.0	5
24.4	4 ●	146.7	1
22.8	4 ●	140.8	1
19.2	6 +	167.6	2
17.8	6 +	167.6	2
16.4	8 ♦	275.8	3
17.3	8 ♦	275.8	3
15.2	8 ♦	275.8	3
10.4	8 ♦	472.0	4
10.4	8 ♦	460.0	4
14.7	8 ♦	440.0	4
32.4	4 ●	78.7	1
30.4	4 ●	75.7	1
33.9	4 ●	71.1	1



**data**

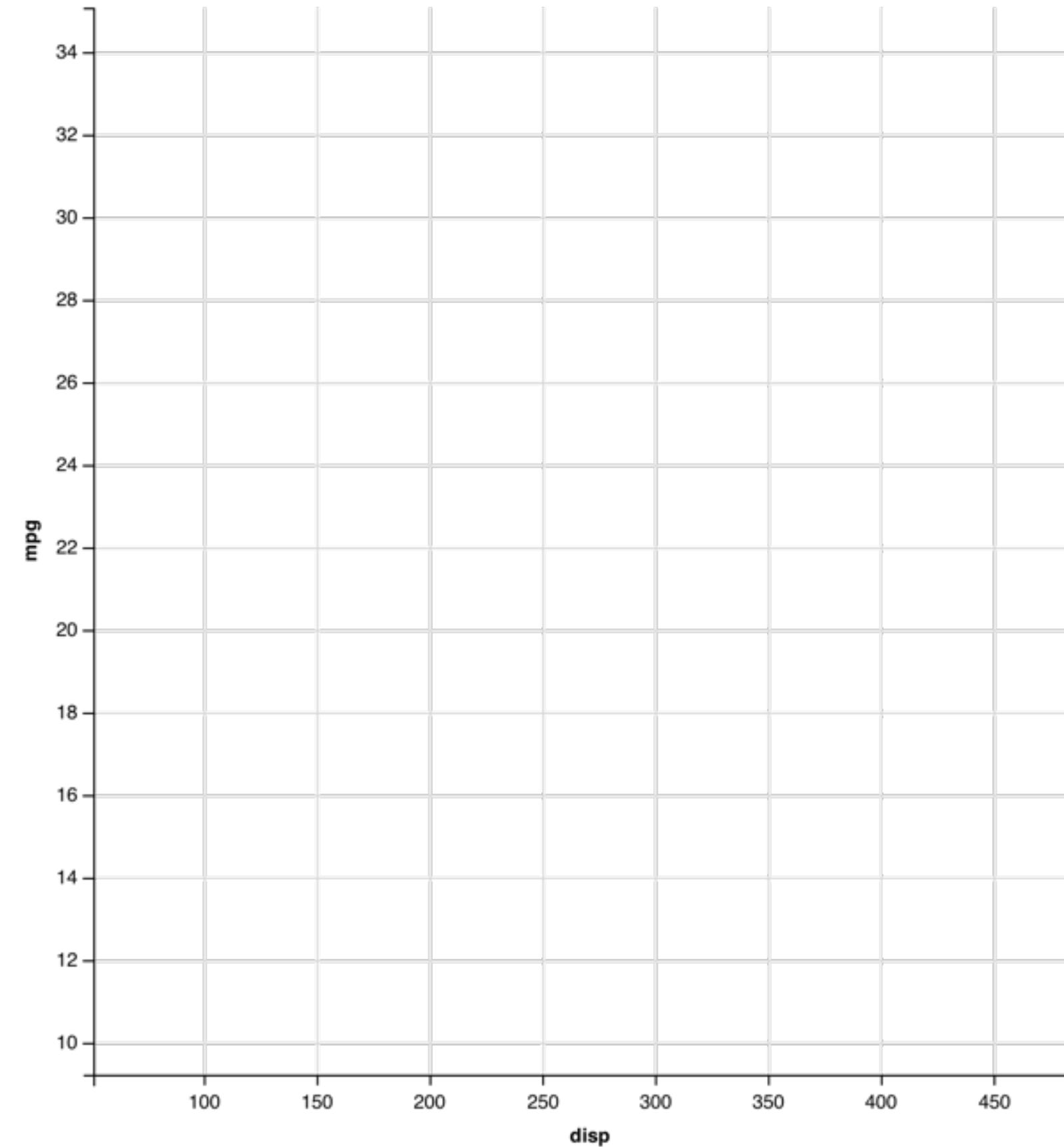
**mark**

**coordinate  
system**

## properties

	shape	x	fill
mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

+ ● △ ◆ ▲ ◇ ◆ ◇



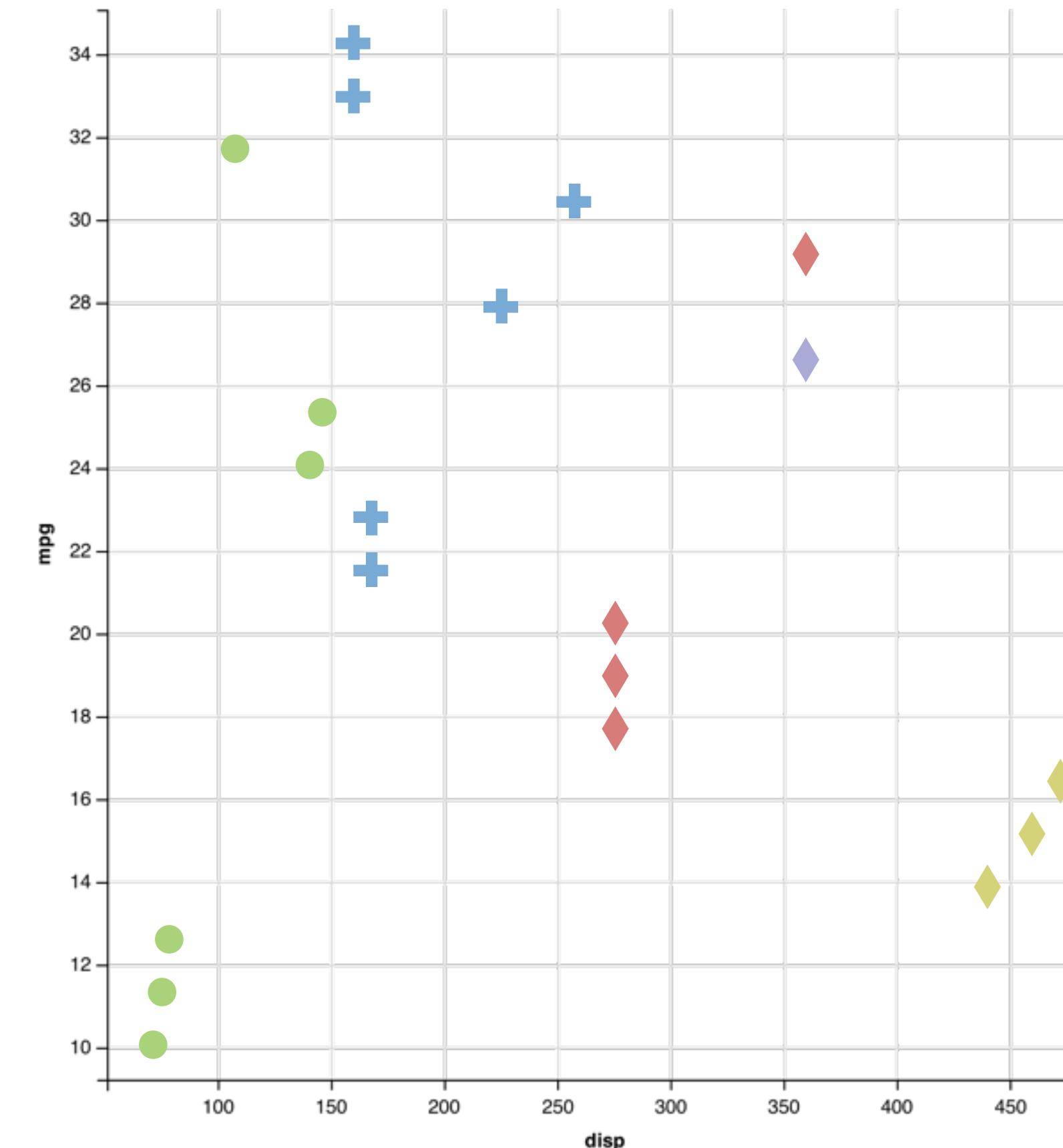
**data**

**mark**

**coordinate  
system**

## properties

y	shape	x	fill
mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1



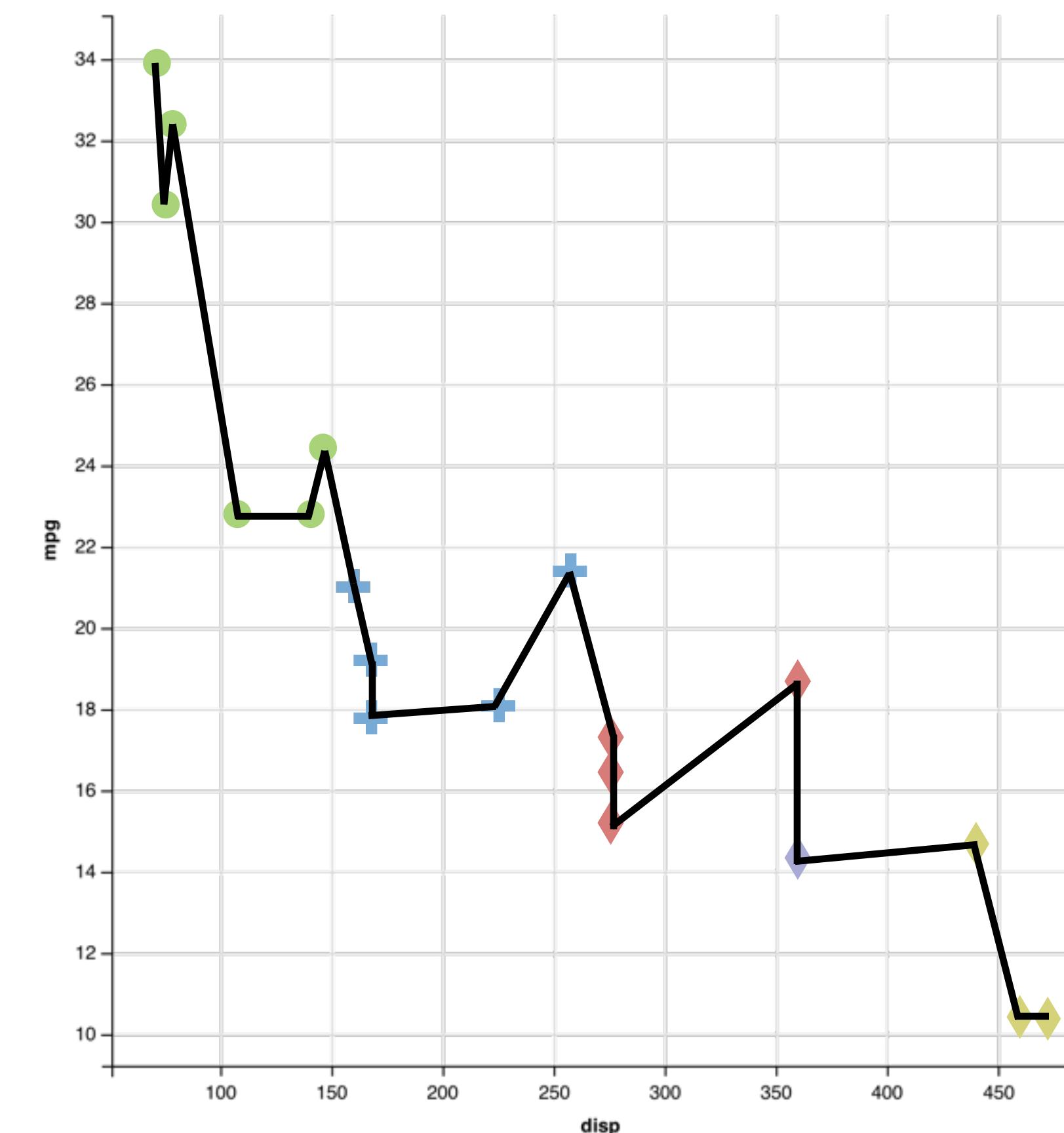
**data**

**mark**

**coordinate  
system**

## properties

	y	shape	x	fill
	mpg	cyl	disp	hp
21.0	6	160.0	2	
21.0	6	160.0	2	
22.8	4	108.0	1	
21.4	6	258.0	2	
18.7	8	360.0	3	
18.1	6	225.0	2	
14.3	8	360.0	5	
24.4	4	146.7	1	
22.8	4	140.8	1	
19.2	6	167.6	2	
17.8	6	167.6	2	
16.4	8	275.8	3	
17.3	8	275.8	3	
15.2	8	275.8	3	
10.4	8	472.0	4	
10.4	8	460.0	4	
14.7	8	440.0	4	
32.4	4	78.7	1	
30.4	4	75.7	1	
33.9	4	71.1	1	



**data**

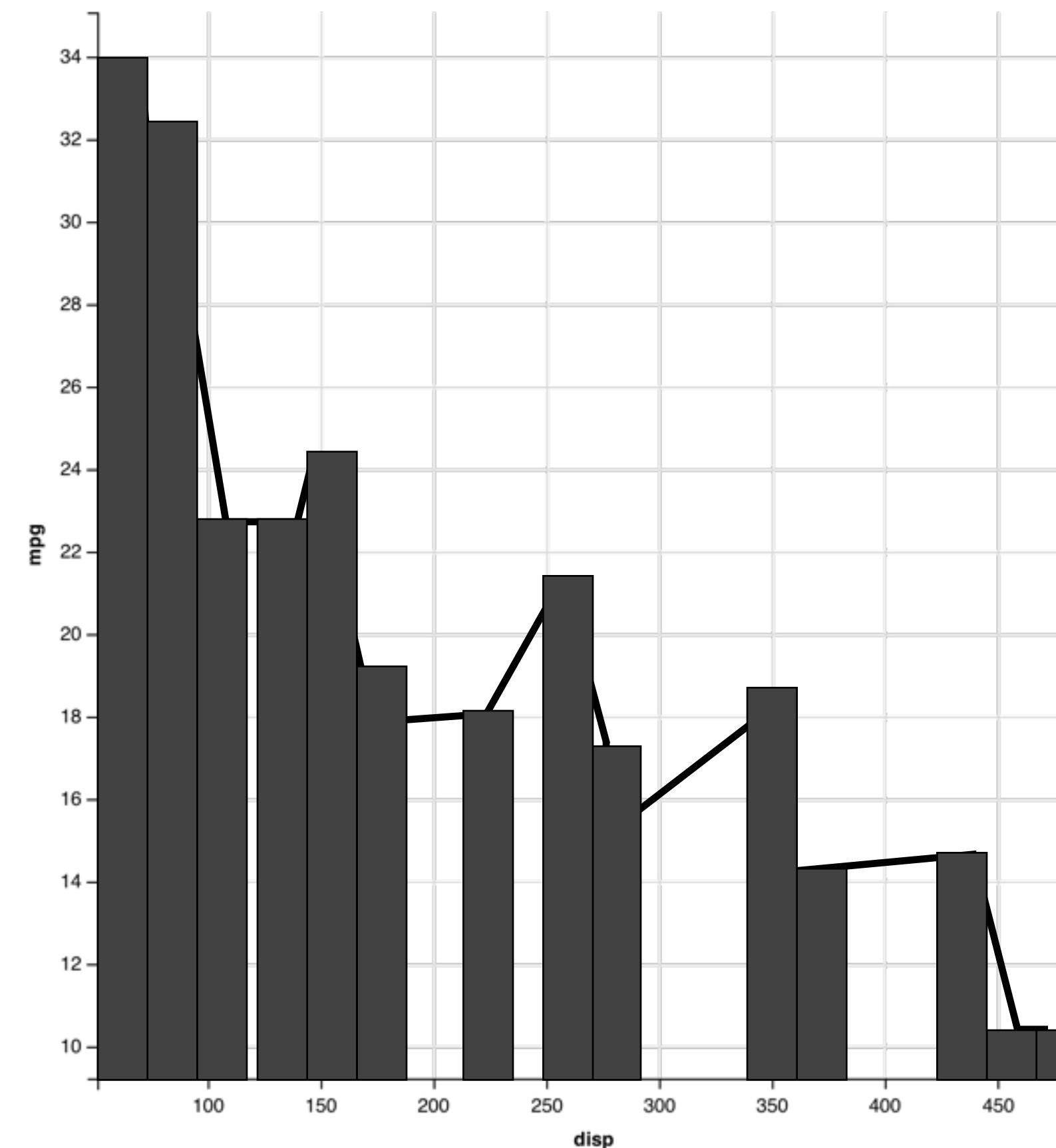
**mark**  
points  
lines

**coordinate  
system**

## properties

Y  
↑ ↓  
X  
↑ ↓

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1



**data**

**mark**  
points  
lines  
bars

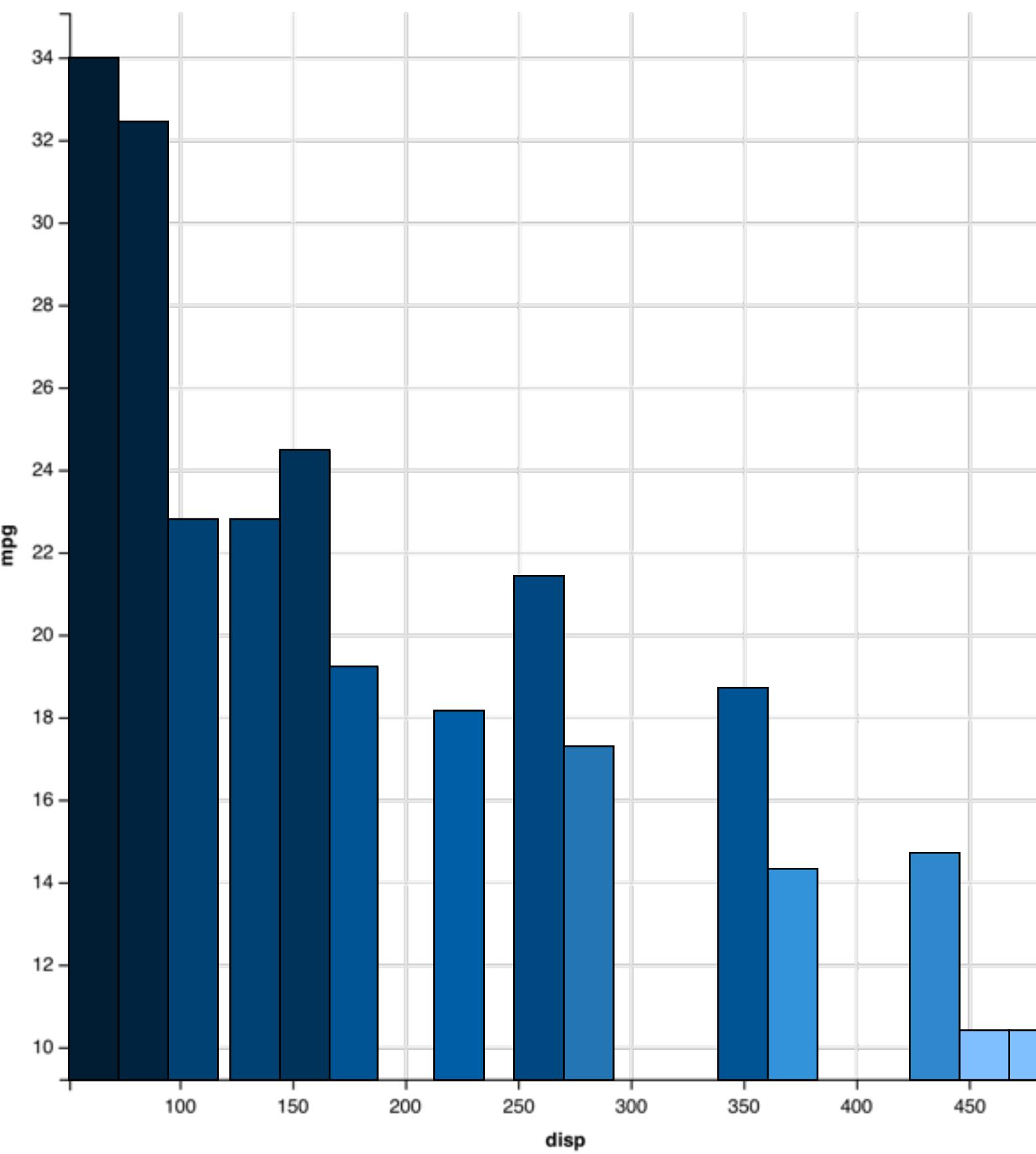
**coordinate  
system**

## properties

y  
↑ ↓

fill  
↑ ↓

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1



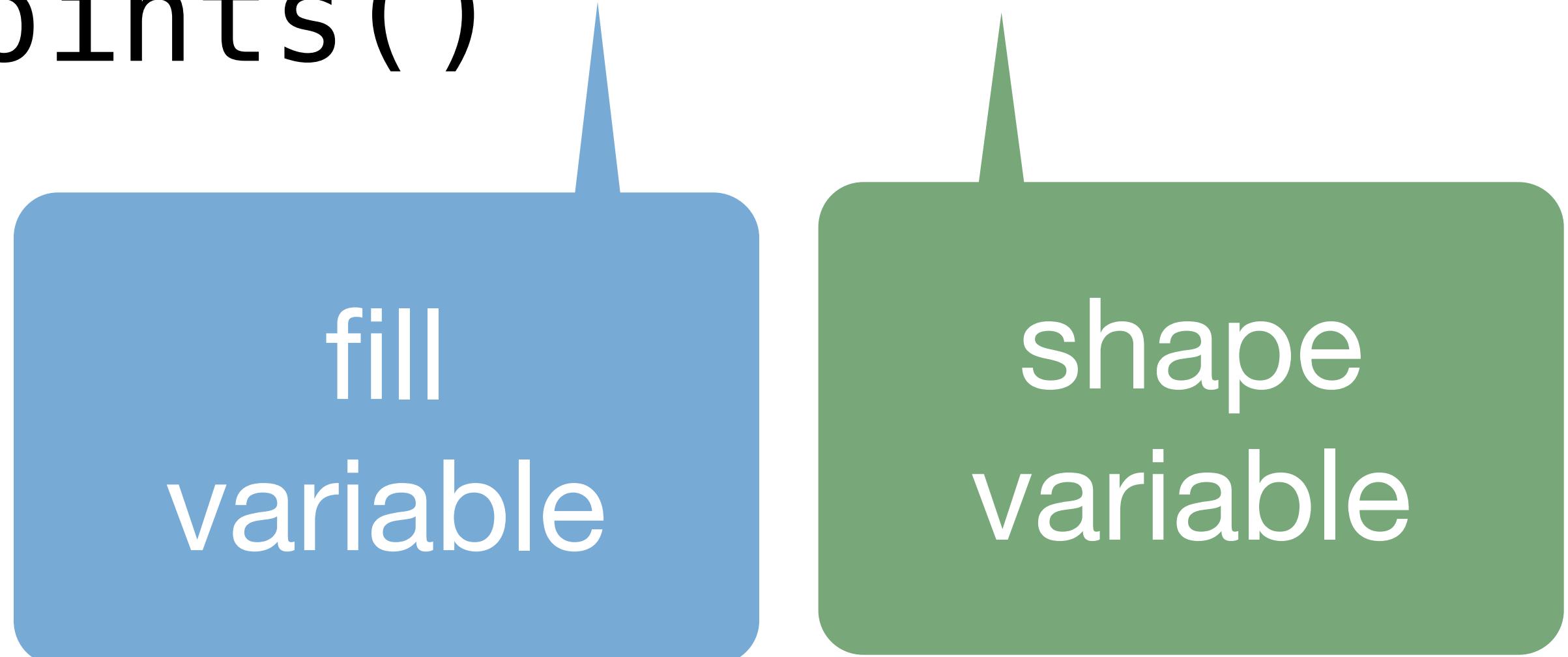
**data**

**mark**  
points  
lines  
bars

**coordinate**  
**system**

# Set properties

```
mtcars %>%  
  ggvis(x = ~disp, y = ~mpg,  
        fill = ~hp, shape = ~cyl) %>%  
  layer_points()
```



fill  
variable

shape  
variable

# Set properties

```
mtcars %>%  
  ggvis(x = ~disp, y = ~mpg,  
        fill = ~hp, shape = ~cyl) %>%  
  layer_points()
```

```
x <- c(1, 2, 3)
```

```
df <- data.frame(x = c("a", "b", "c"), y = 1:3,  
stringsAsFactors = FALSE)
```

x

```
## 1 2 3
```

The object  
named x

"x"

```
## "x"
```

The literal value  
"x" (a string)

df\$x

```
## "a" "b" "c"
```

The column named  
x in the data frame  
named df

~

The ~ syntax provides a shortcut for referring to a column in your data frame. Code after the ~ will be evaluated in the context of your data frame.

fill x = x

The object  
named rate

fill'x' = "x"

The literal value  
"rate" (a string)

fill df\$x = ~x  
fill df\$x = df\$x

The column named rate  
in the data frame that  
you passed to ggvis

**interactive  
graphs**

# Inputs

Make plots interactive by mapping properties to an input control. Create input controls with an **input\_** function.

```
sliderBox <- input_slider(.1, 2, value = 1, step = .1,  
label = "Bandwidth adjustment")  
  
selectBox <- input_select(c("Gaussian" = "gaussian",  
"Epanechnikov" = "epanechnikov", "Rectangular" = "rectangular",  
"Triangular" = "triangular", "Biweight" = "biweight",  
"Cosine" = "cosine", "Optcosine" = "optcosine"), label = "Kernel")  
  
mtcars %>%  
  ggvis(x = ~wt) %>%  
  layer_densities(adjust = sliderBox, kernel = selectBox)
```

# Inputs

Currently available input functions.

`input_checkbox`

`input_checkboxgroup`

`input_numeric`

`input_radiobuttons`

`input_select`

`input_slider`

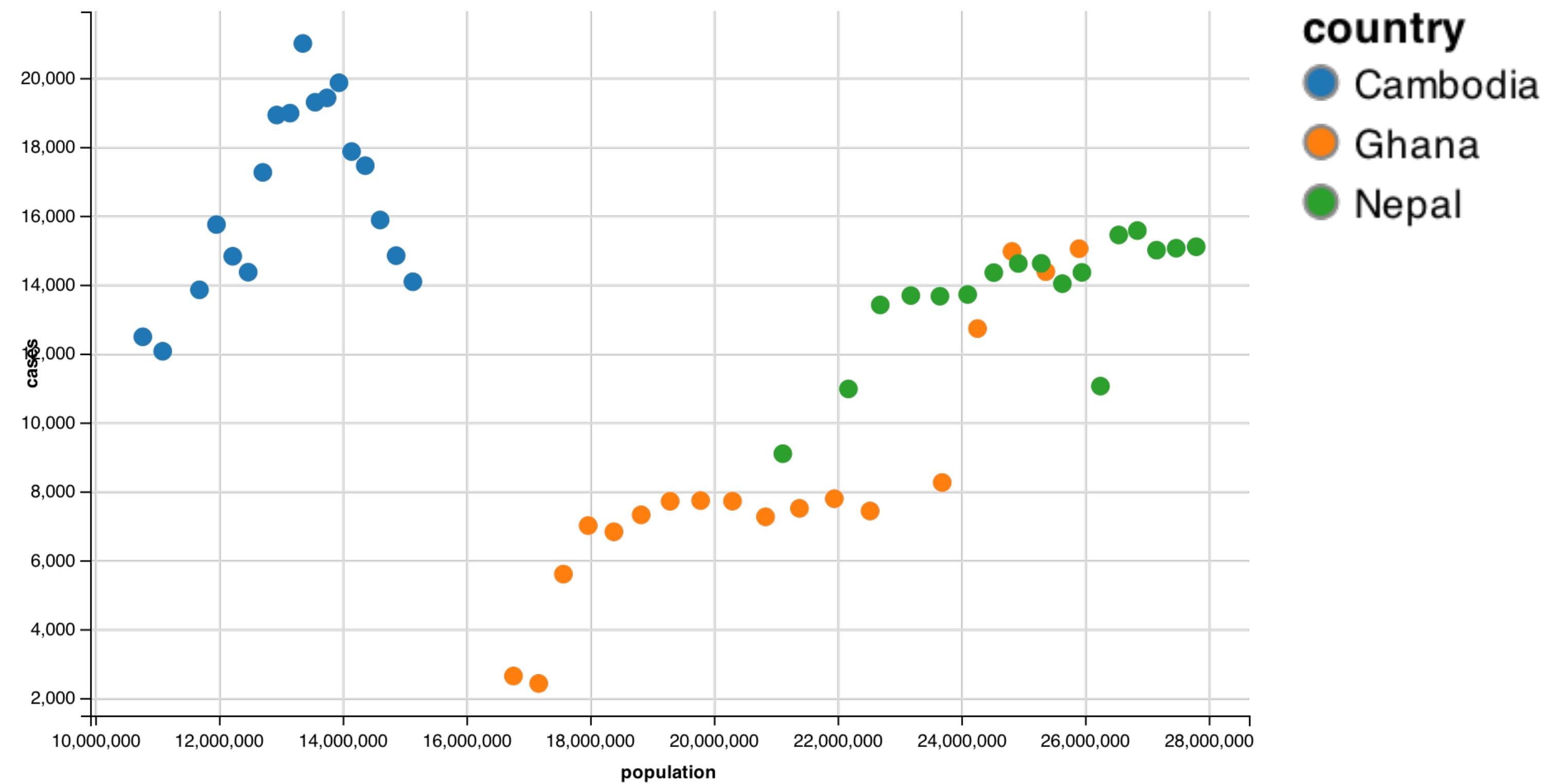
`input_text`

# Hover events

```
# This function receives information about the hovered
# point and returns an HTML string to display
all_values <- function(x) {
  if(is.null(x)) return(NULL)
  paste0(names(x), ": ", format(x), collapse = "<br />")
}

mtcars %>% ggvis(x = ~wt, y = ~mpg) %>%
  layer_points(fill.hover := "red") %>%
  add_tooltip(all_values, "hover")
```

# Customizing graphs



## Data Space

**country**

Cambodia

Ghana

Nepal

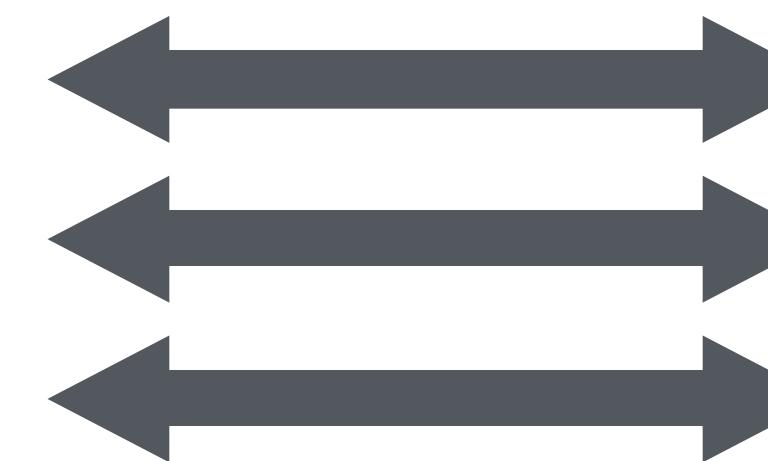
## Visual Space

**fill**

Blue

Orange

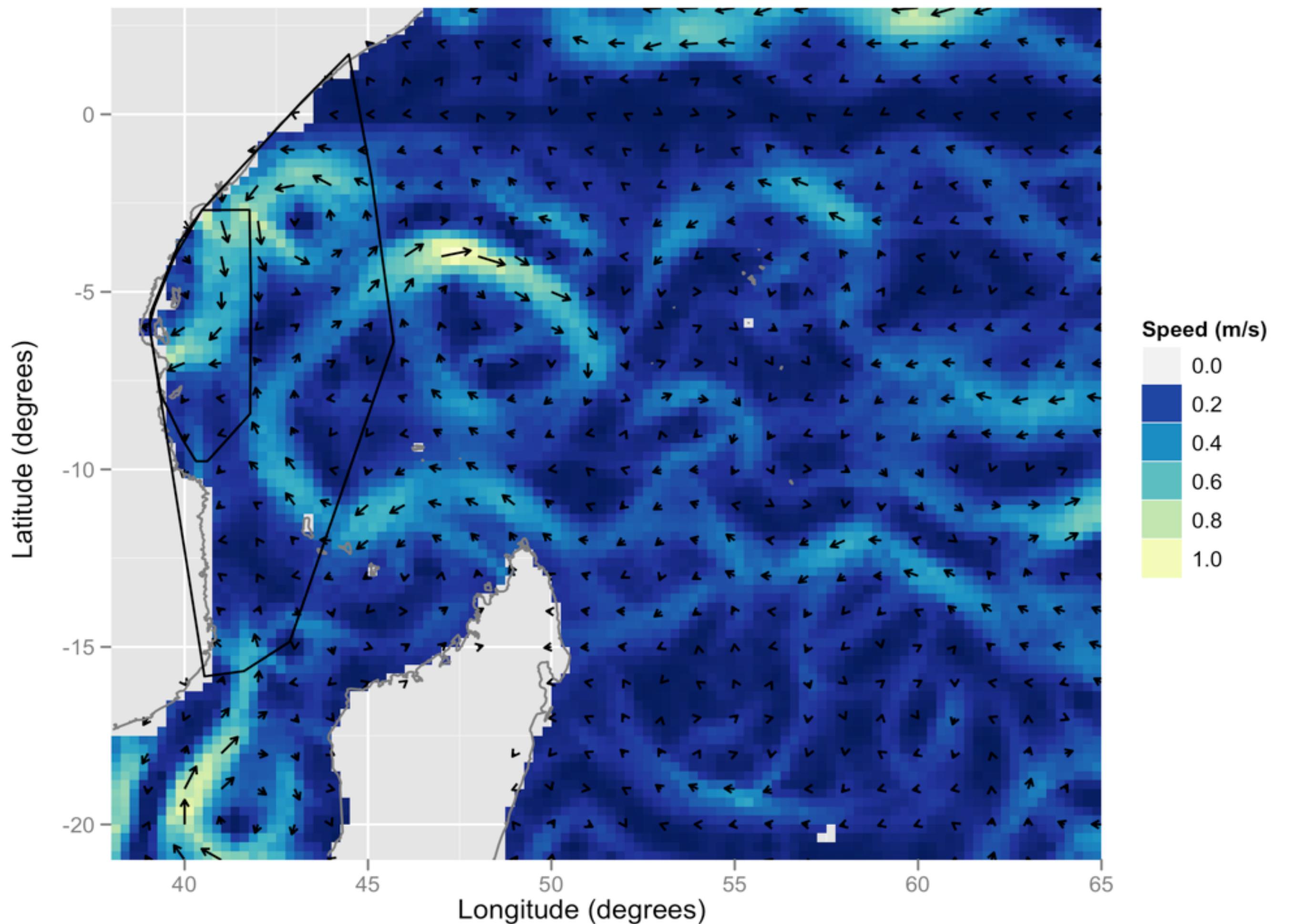
Green



= vs :=

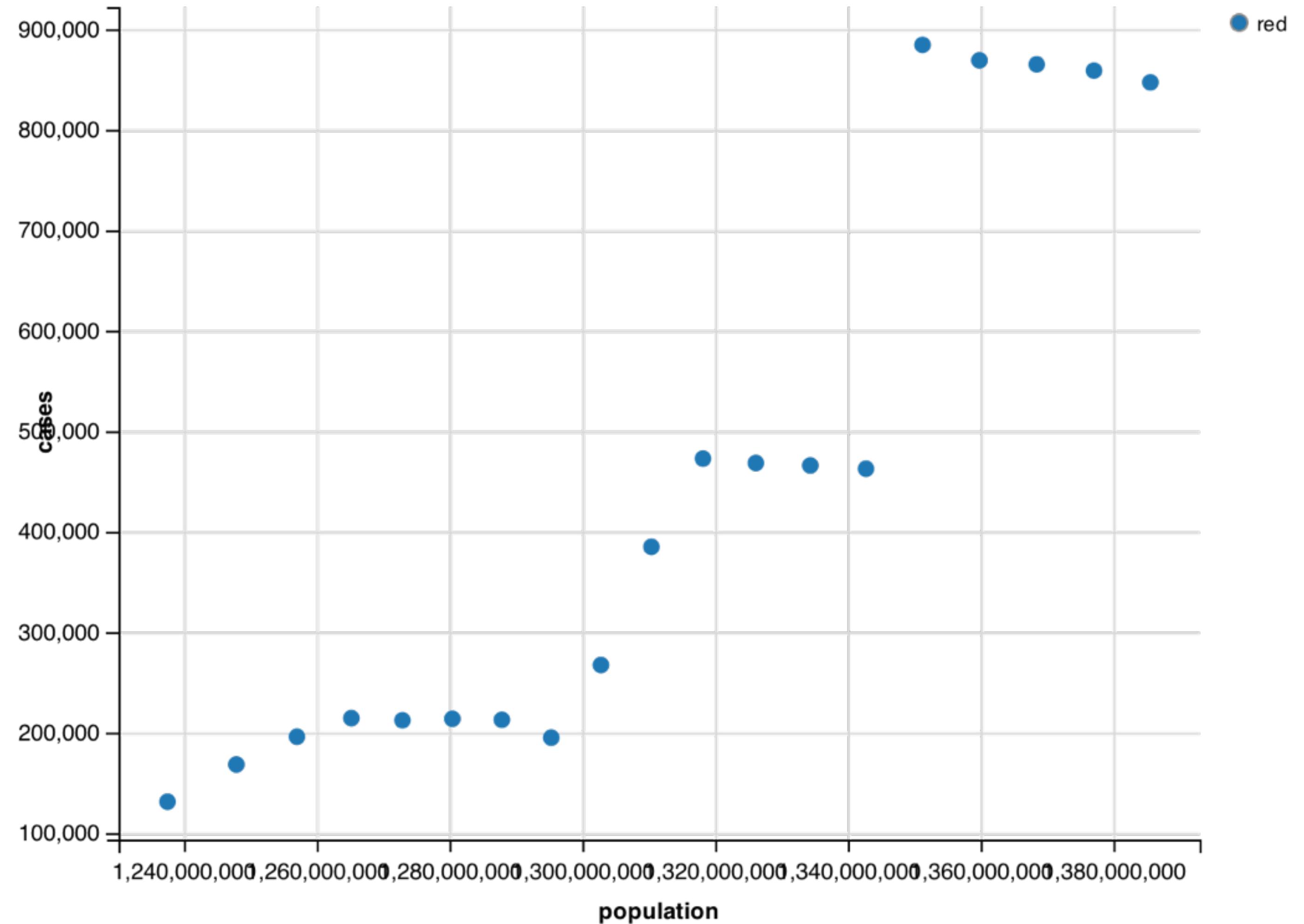
What if you want to manually set a property?

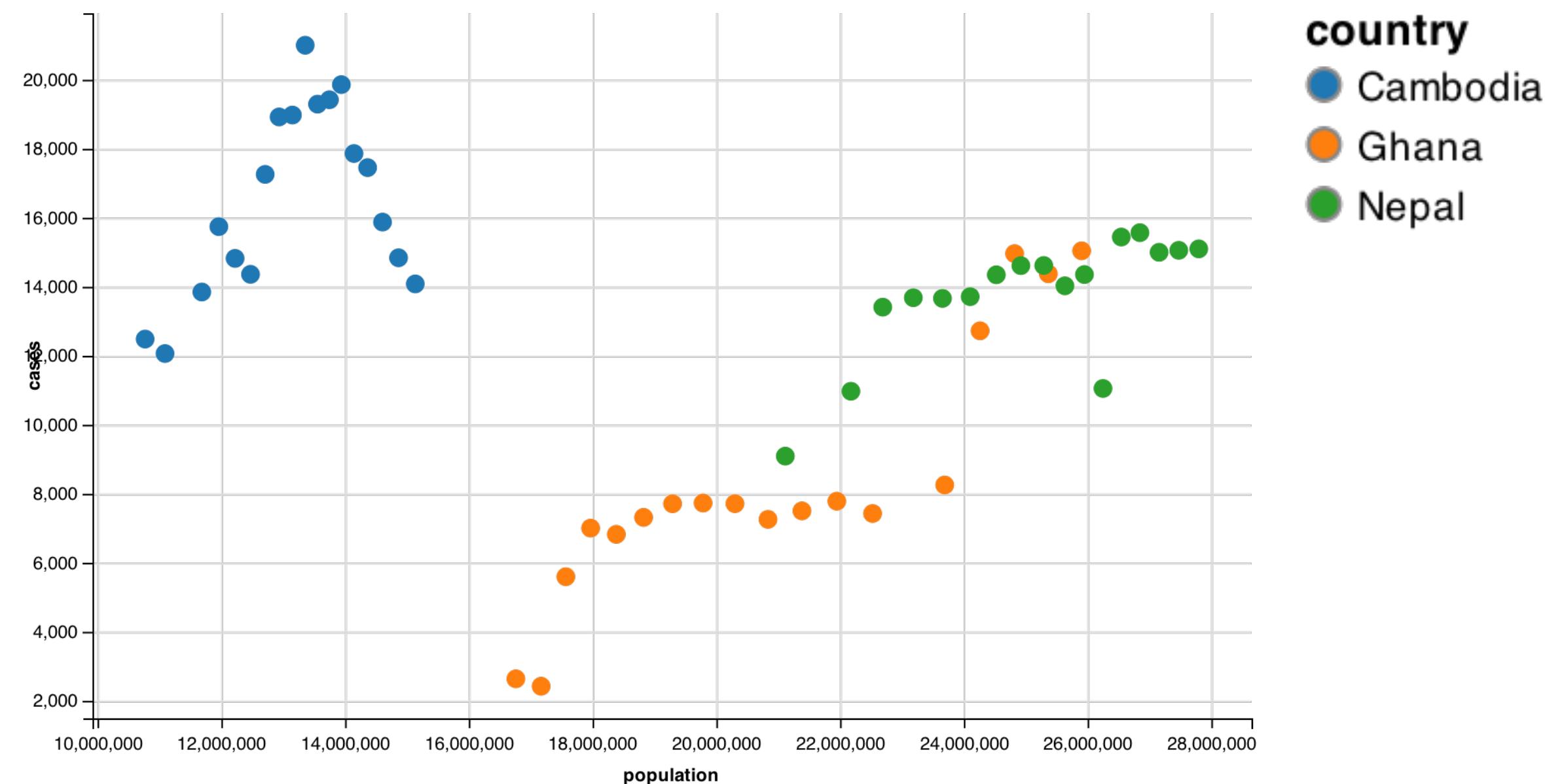
e.g. make all of the points **red**?



= VS :=

```
china %>%  
  ggvis(x = ~population,  
        y = ~cases,  
        fill = "red") %>%  
  layer_points()
```





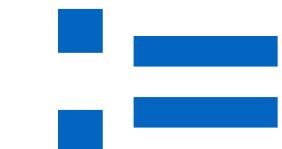
## Data Space

**country**

Cambodia

Ghana

Nepal



## Visual Space

**fill**

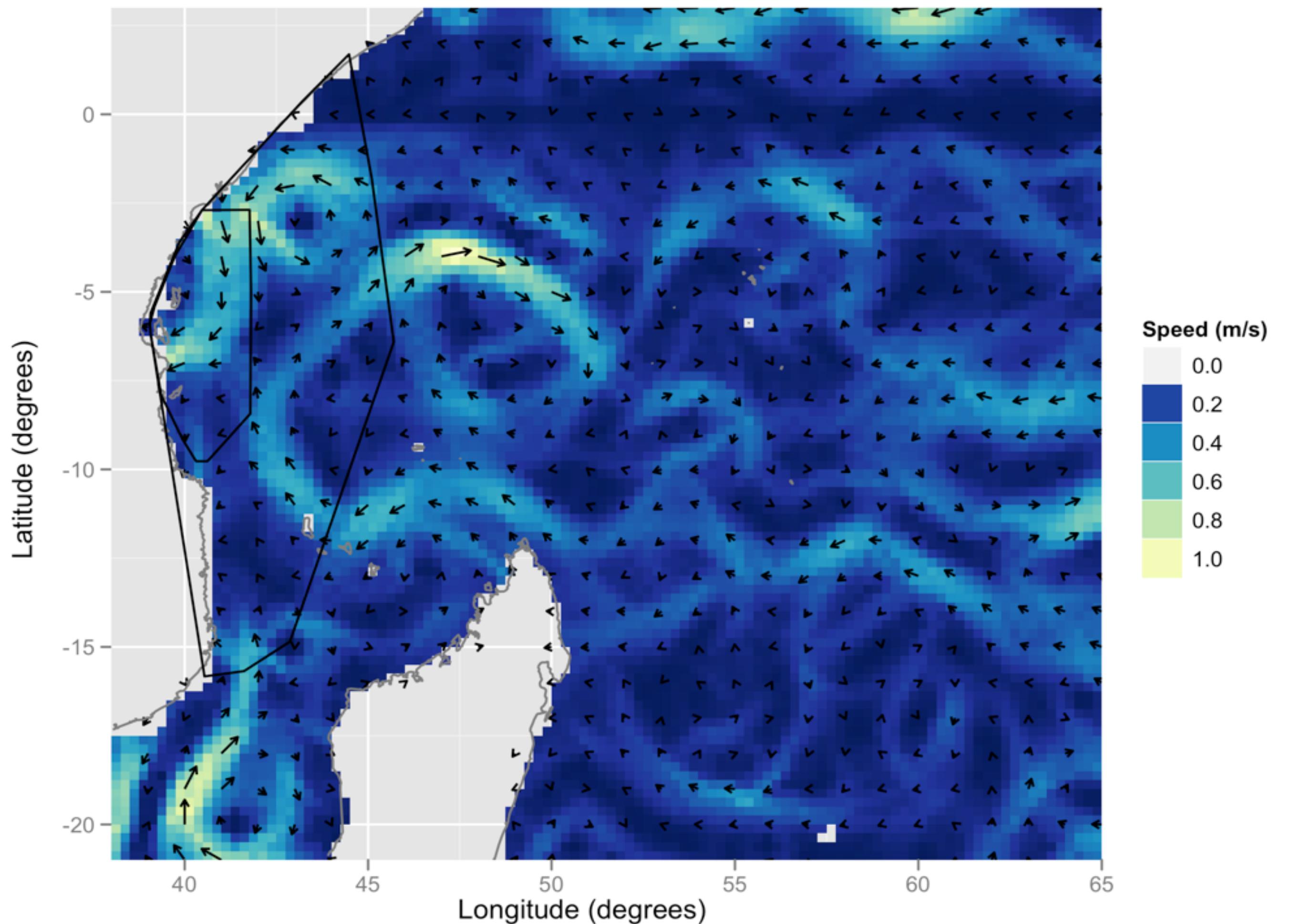
Blue

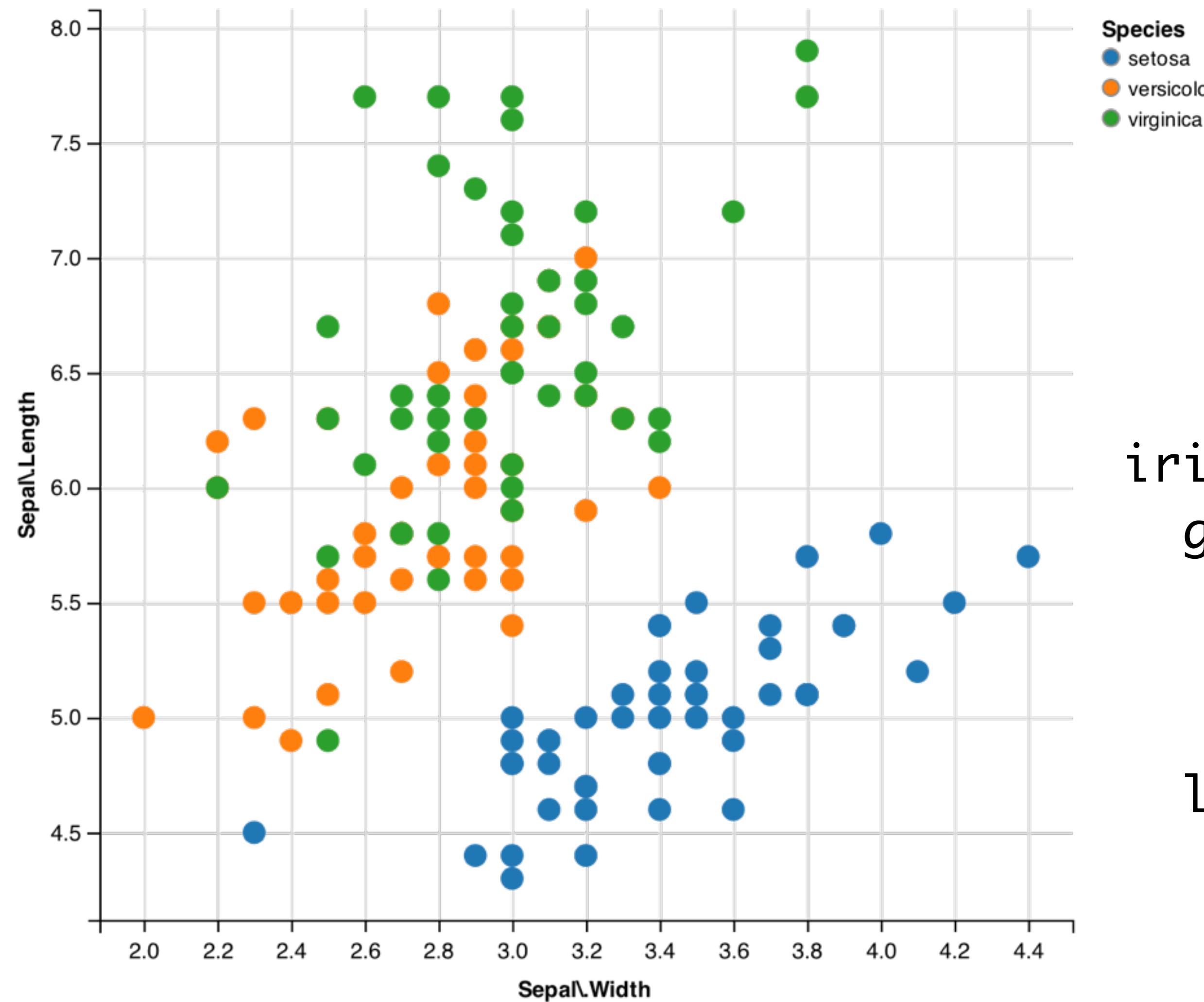
Orange

Green

= vs :=

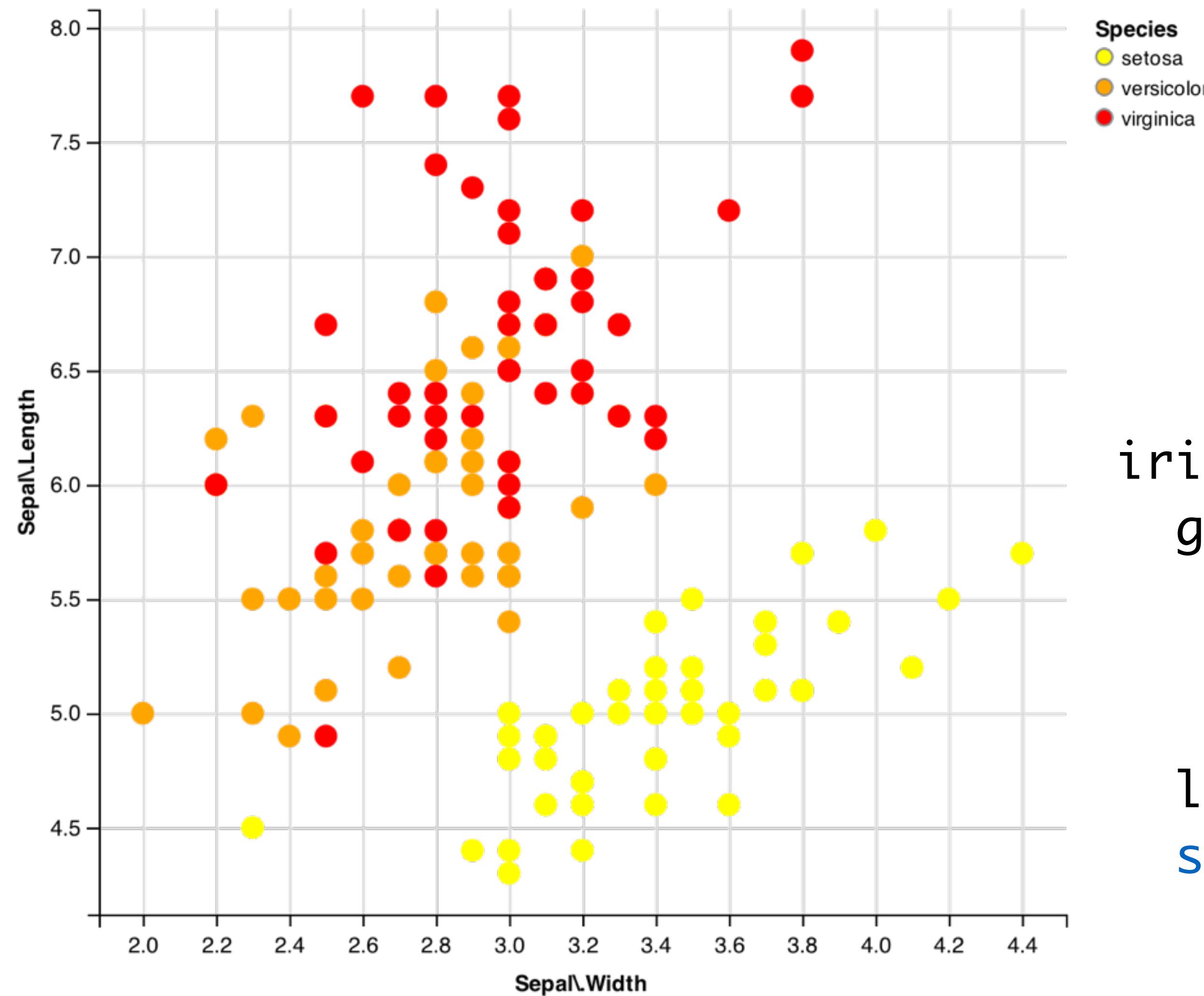
```
china %>%  
  ggvis(x = ~population,  
        y = ~cases,  
        fill := "red") %>%  
  layer_points()
```





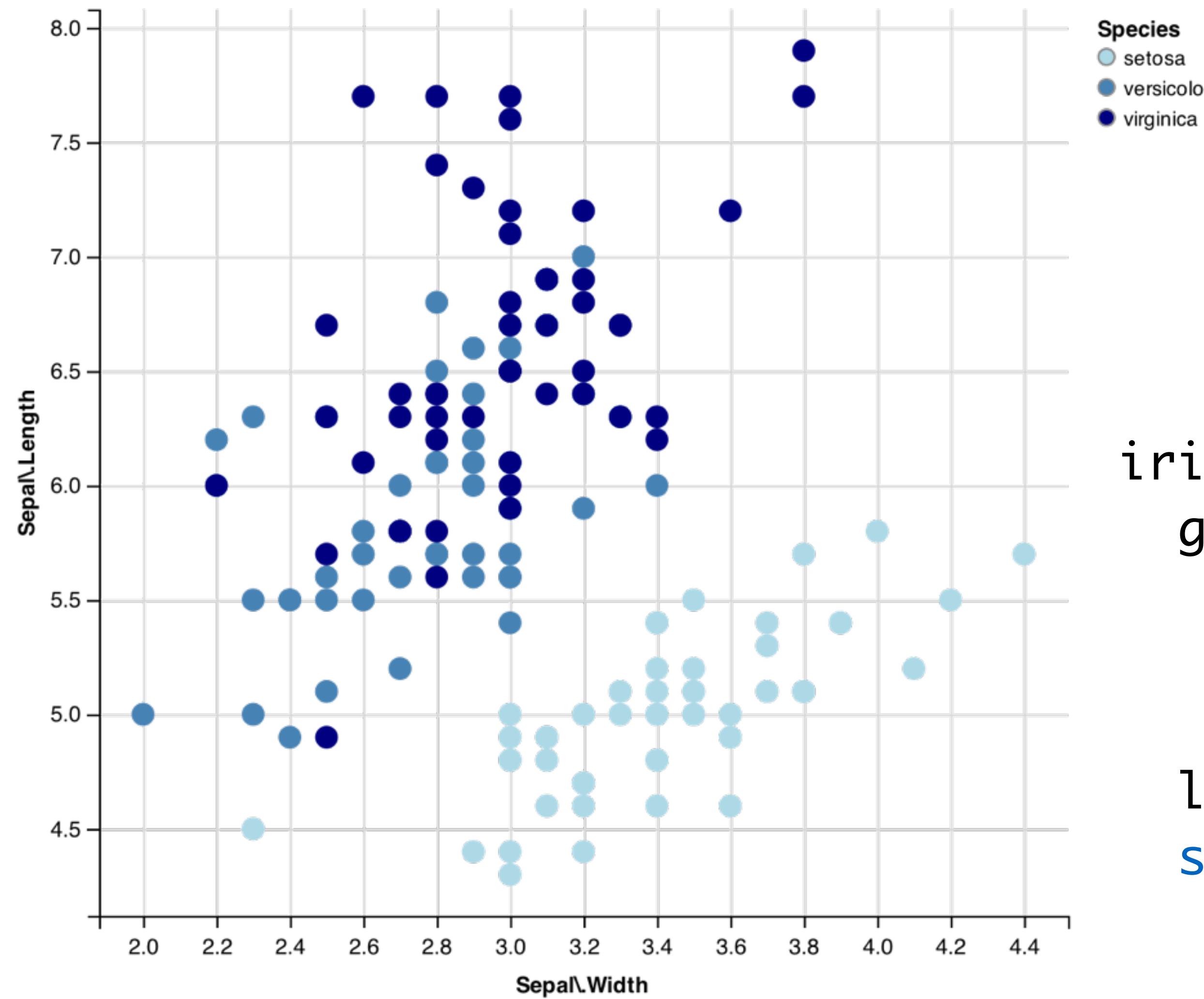
Change how the data space is mapped to the visual space with a **scale** function.

```
iris %>%  
  ggvis(x = ~Sepal.Width,  
        y = ~Sepal.Length,  
        fill = ~Species,  
        size := 100) %>%  
  layer_points()
```



Change how the data space is mapped to the visual space with a **scale** function.

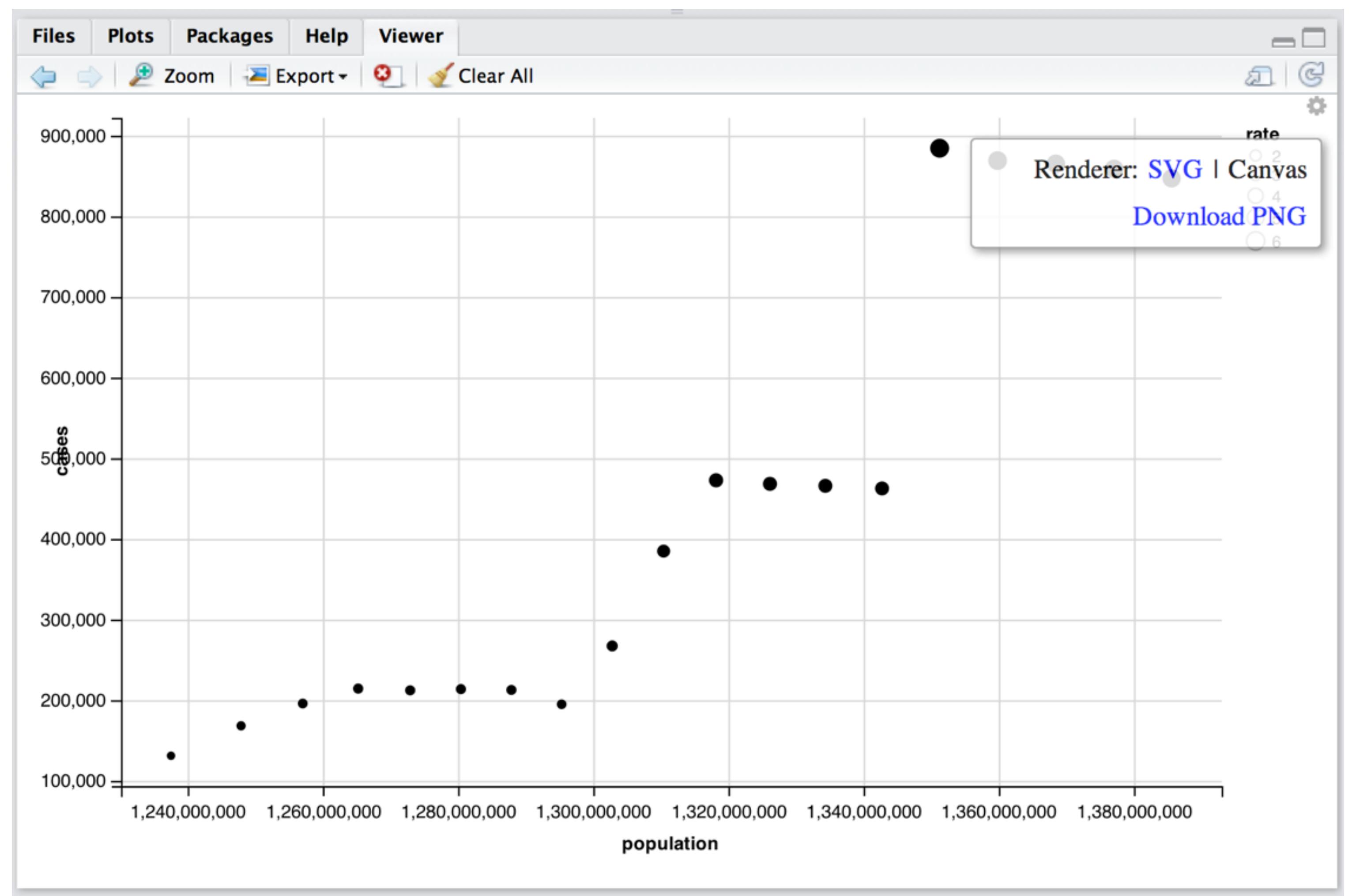
```
iris %>%  
  ggvis(x = ~Sepal.Width,  
        y = ~Sepal.Length,  
        fill = ~Species,  
        size := 100) %>%  
  layer_points() %>%  
  scale_nominal("fill",  
    range = c("yellow", "orange", "red"))
```



Change how the data space is mapped to the visual space with a **scale** function.

```
iris %>%  
  ggvis(x = ~Sepal.Width,  
        y = ~Sepal.Length,  
        fill = ~Species,  
        size := 100) %>%  
  layer_points() %>%  
  scale_nominal("fill",  
    range = c("lightblue", "steelblue",  
             "navy"))
```

# Saving plots

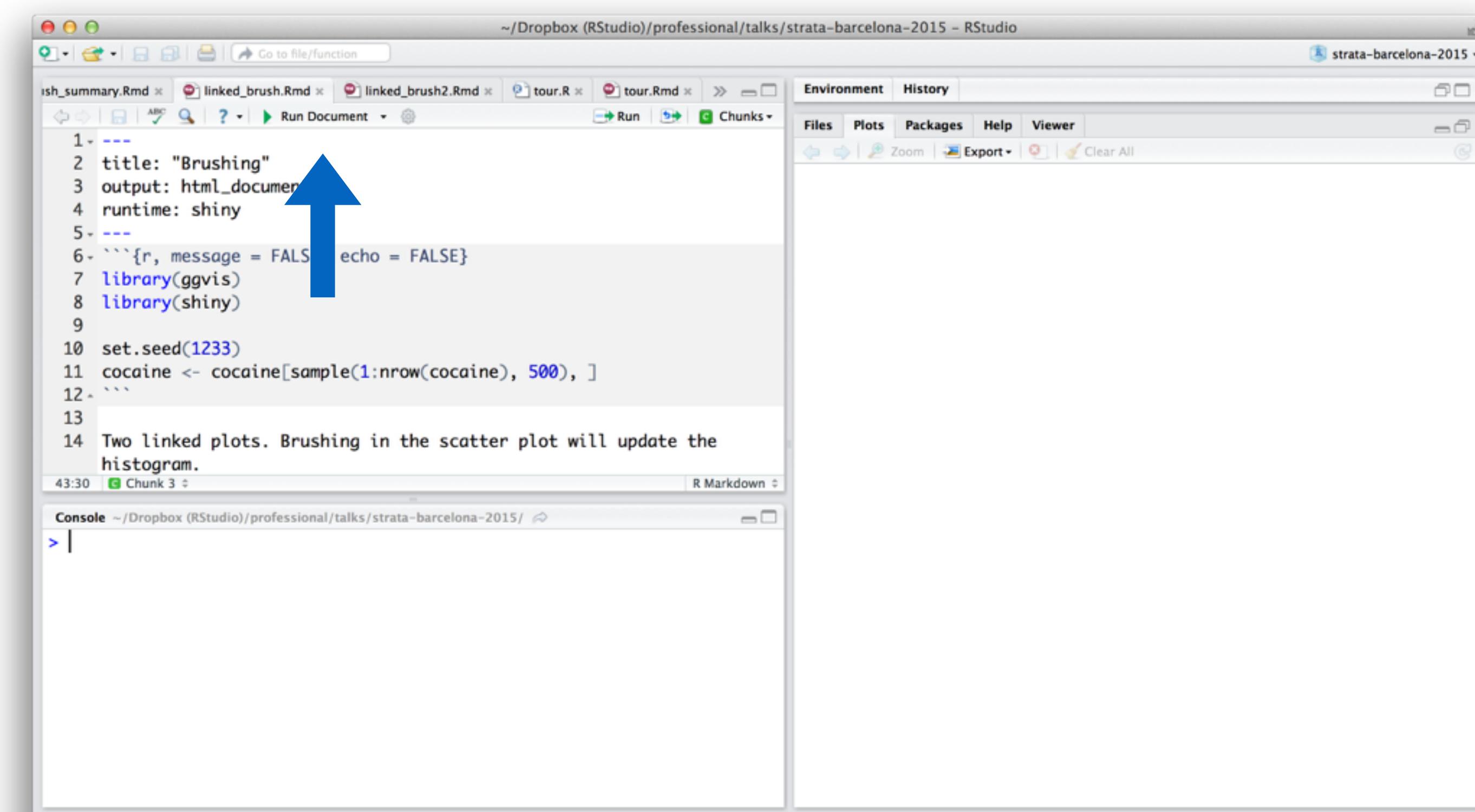


# opportunities

## 1

# Linked brushing

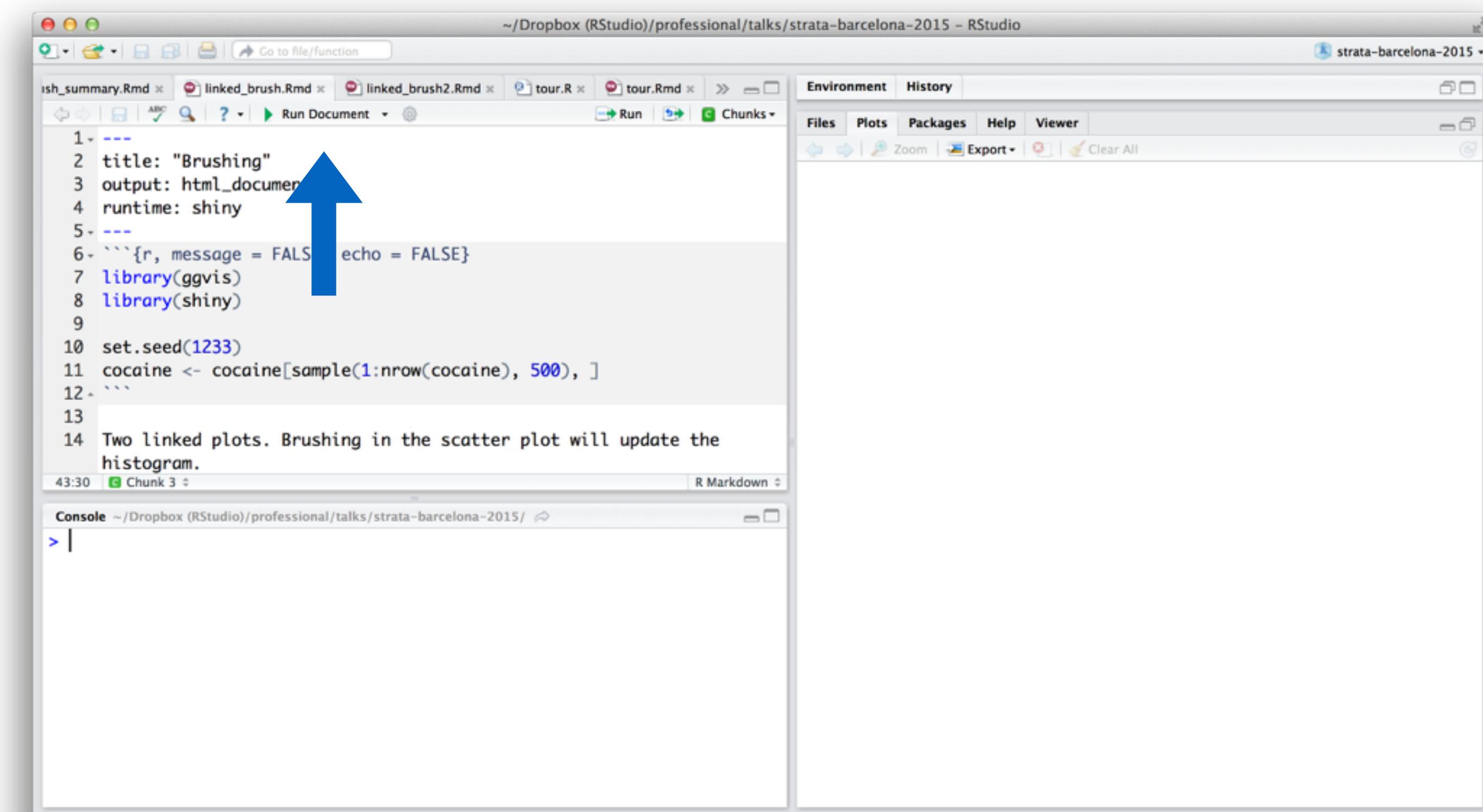
To run the demo, open [linked\\_brush2.Rmd](#) in RStudio and click "Run Document" at the top of the file pane.



## 2

# Grand tour

To run the demo, open [linked\\_brush2.Rmd](#) in RStudio and click "Run Document" at the top of the file pane.



**What's  
missing  
(for now)**

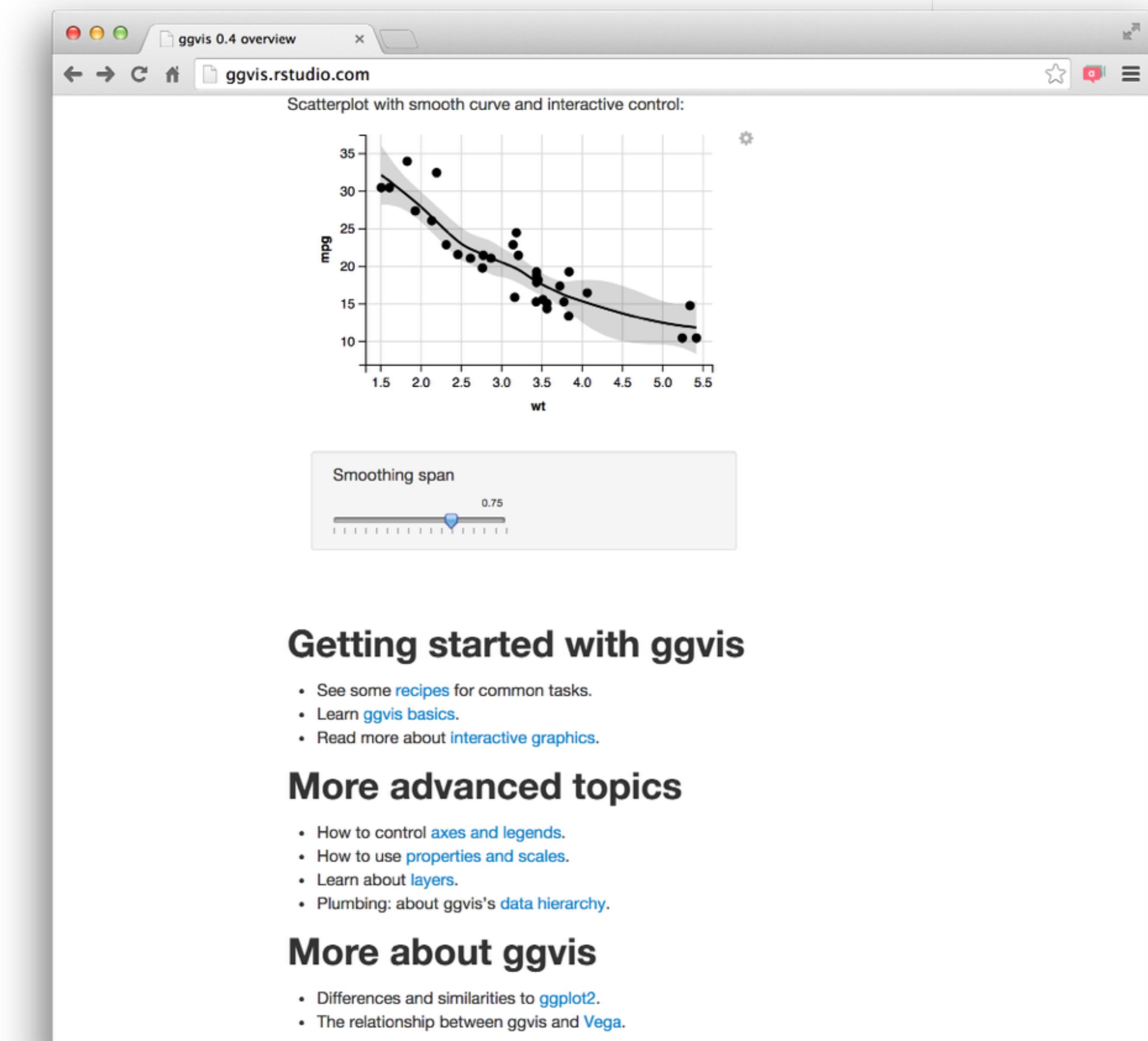
1. more layer types
2. facetting (subplots)
3. customization
4. zooming and panning
5. legends

# **How to learn more**

# Project site

[ggvis.rstudio.com](http://ggvis.rstudio.com)

Tutorials, demonstrations,  
and examples of ggvis in  
action.



# More examples

[github.com/rstudio/  
ggvis/tree/master/demo](https://github.com/rstudio/ggvis/tree/master/demo)

ggvis/demo at master · rst · GitHub, Inc. [US] https://github.com/rstudio/ggvis/tree/master/demo

This repository Search Explore Gist Blog Help garrettgman Unwatch 73 Star 325 Fork 85

branch: master ggvis / demo / +

Rename some uses of 'origin' to 'boundary'  
wch authored on Oct 15 latest commit 8c0c1dbf02

File	Description	Date
apps	Rename some uses of 'origin' to 'boundary'	a month ago
rmarkdown	Rename some uses of 'origin' to 'boundary'	a month ago
00Index	Add boxplot to demo index	3 months ago
bar.r	Update demos	6 months ago
boxplot.r	Rename layer_boxplot to layer_boxplots	3 months ago
brush.r	Update demos	6 months ago
dynamic.r	More demo updates	6 months ago
guides.r	Update demos and tests for new legend syntax	5 months ago
histogram.r	Replace remaining 'binwidth' with 'width'	2 months ago
hover.r	Update hover demos	7 months ago
interactive.R	Update demo/interactive	7 months ago
lines.r	Add support for strokeDash property	5 months ago
scales.r	Rename add_guide_legend and add_guide_axis to add_legend and add_axis	5 months ago
scatterplot.r	Update demos and vignettes with new props interface	5 months ago
size.r	Update demo/size	7 months ago
smooth.r	Fix smooth demo	7 months ago
subvis.R	Edits to subvis examples to satisfy R CMD check	4 months ago
tile.r	Update demo/tile	7 months ago
tourr.r	Update demos	6 months ago

# www.rstudio.com

