

Lab 3

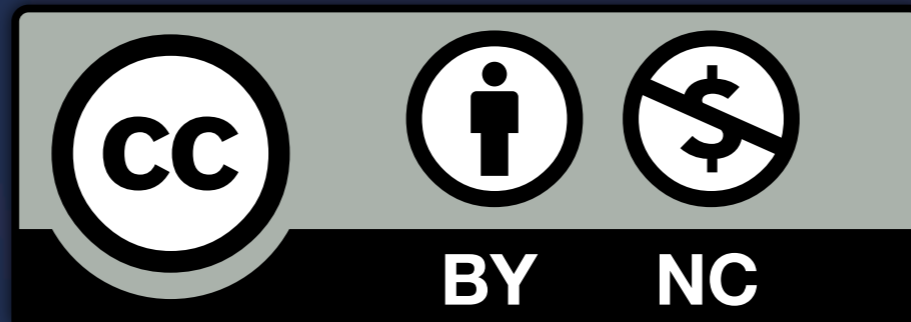
R/ggplot2

Feb 7, 2013 – Michael Porath (@poezn)

Original Slides

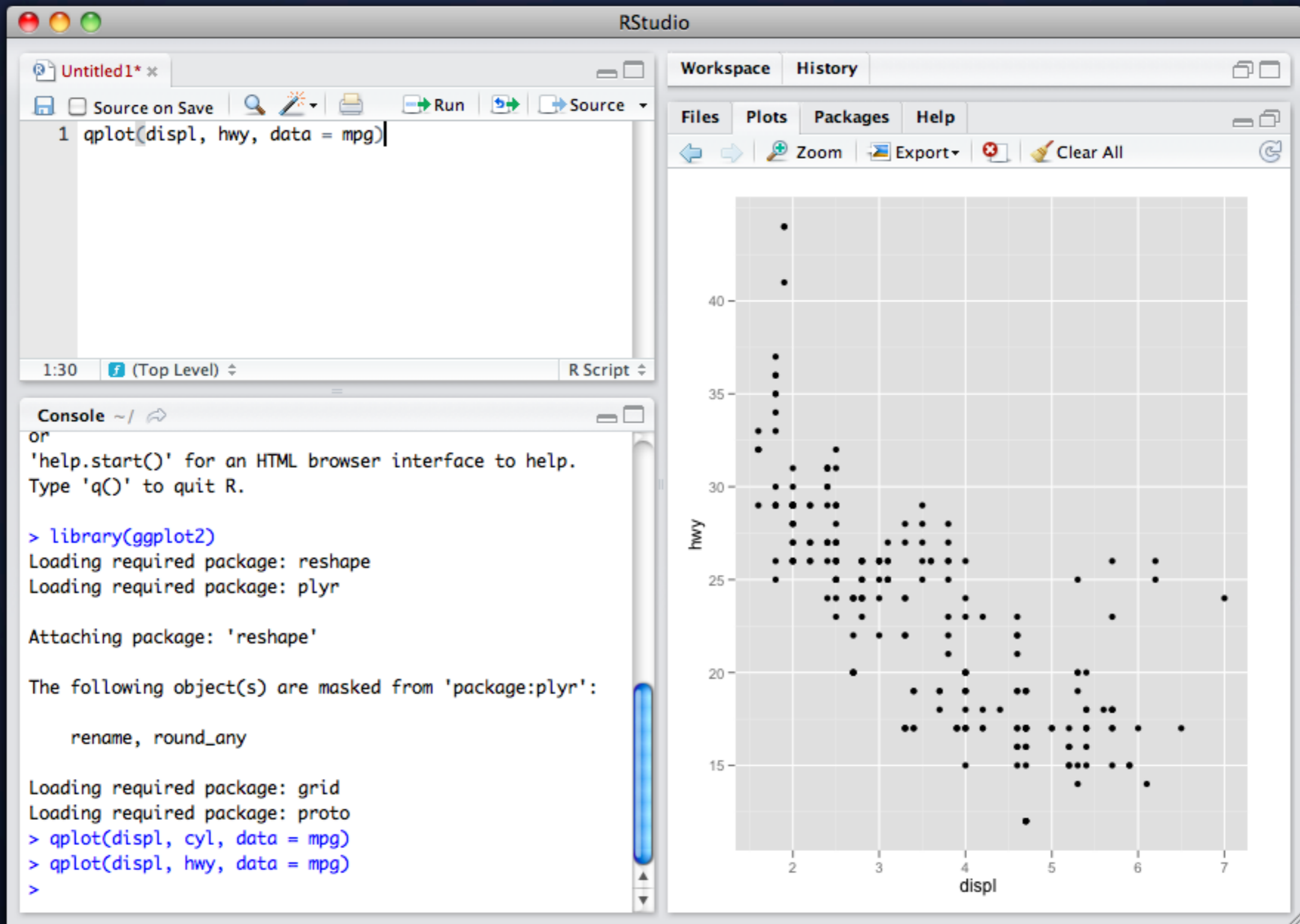
“Intro to R and ggplot2”

by Hadley Wickam, creator of ggplot2



Rstudio

Rstudio



The screenshot displays the RStudio environment with the following components:

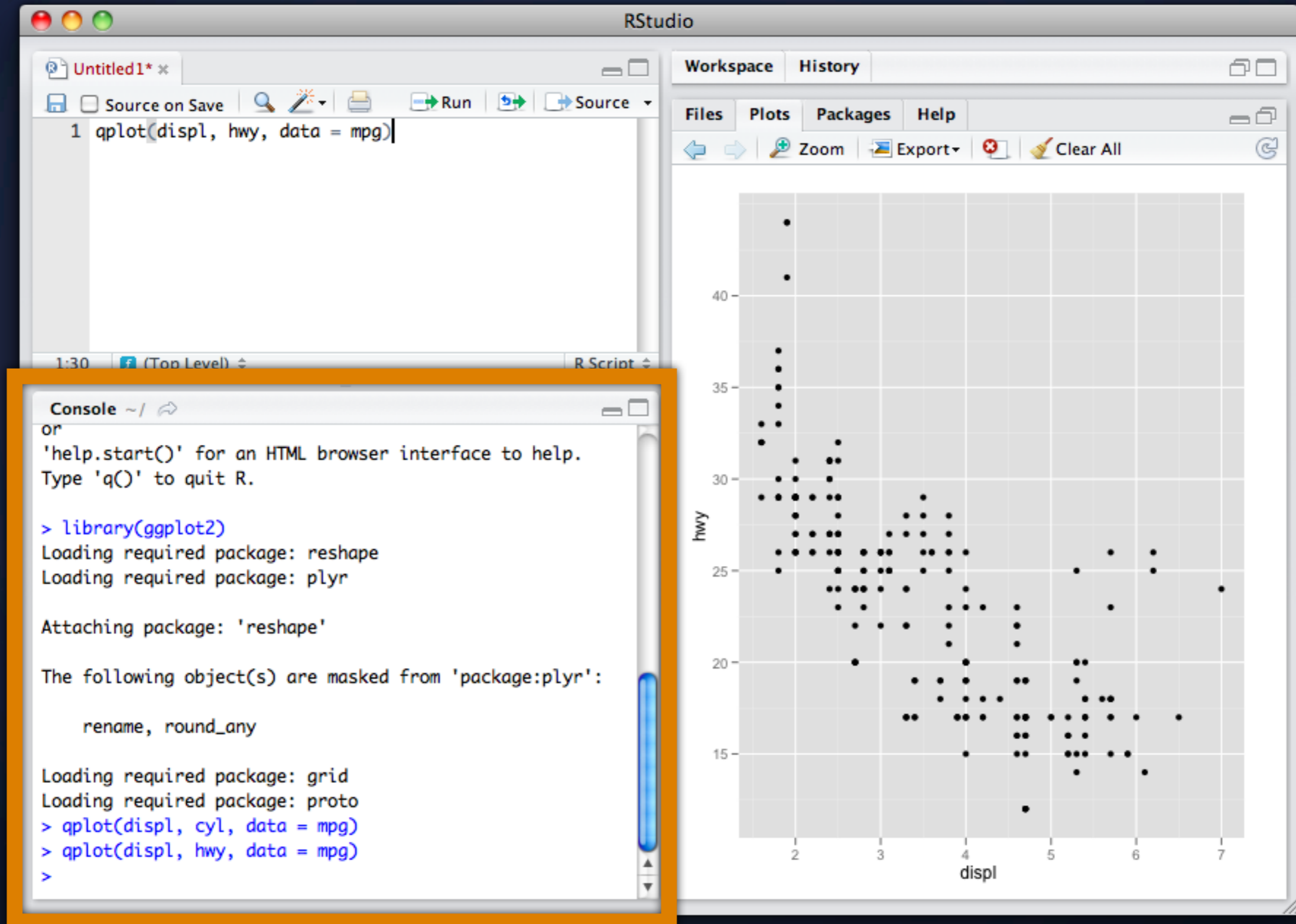
- Source Editor:** Contains the R code `1 qplot(displ, hwy, data = mpg)`.
- Console:** Shows the execution of `library(ggplot2)` and the loading of required packages: `reshape` and `plyr`. It also shows the execution of `qplot(displ, cyl, data = mpg)` and `qplot(displ, hwy, data = mpg)`.
- Plots Panel:** Displays a scatter plot of highway mileage (hwy) versus engine displacement (displ) for the mpg dataset. The plot features a light gray grid and black data points.

```
1 qplot(displ, hwy, data = mpg)
```

```
or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> library(ggplot2)  
Loading required package: reshape  
Loading required package: plyr  
  
Attaching package: 'reshape'  
  
The following object(s) are masked from 'package:plyr':  
  
  rename, round_any  
  
Loading required package: grid  
Loading required package: proto  
> qplot(displ, cyl, data = mpg)  
> qplot(displ, hwy, data = mpg)  
>
```

Rstudio

Console - run code here



The image shows the RStudio interface. The top-left pane is the Source Editor, containing the following R code:

```
1 qplot(displ, hwy, data = mpg)
```

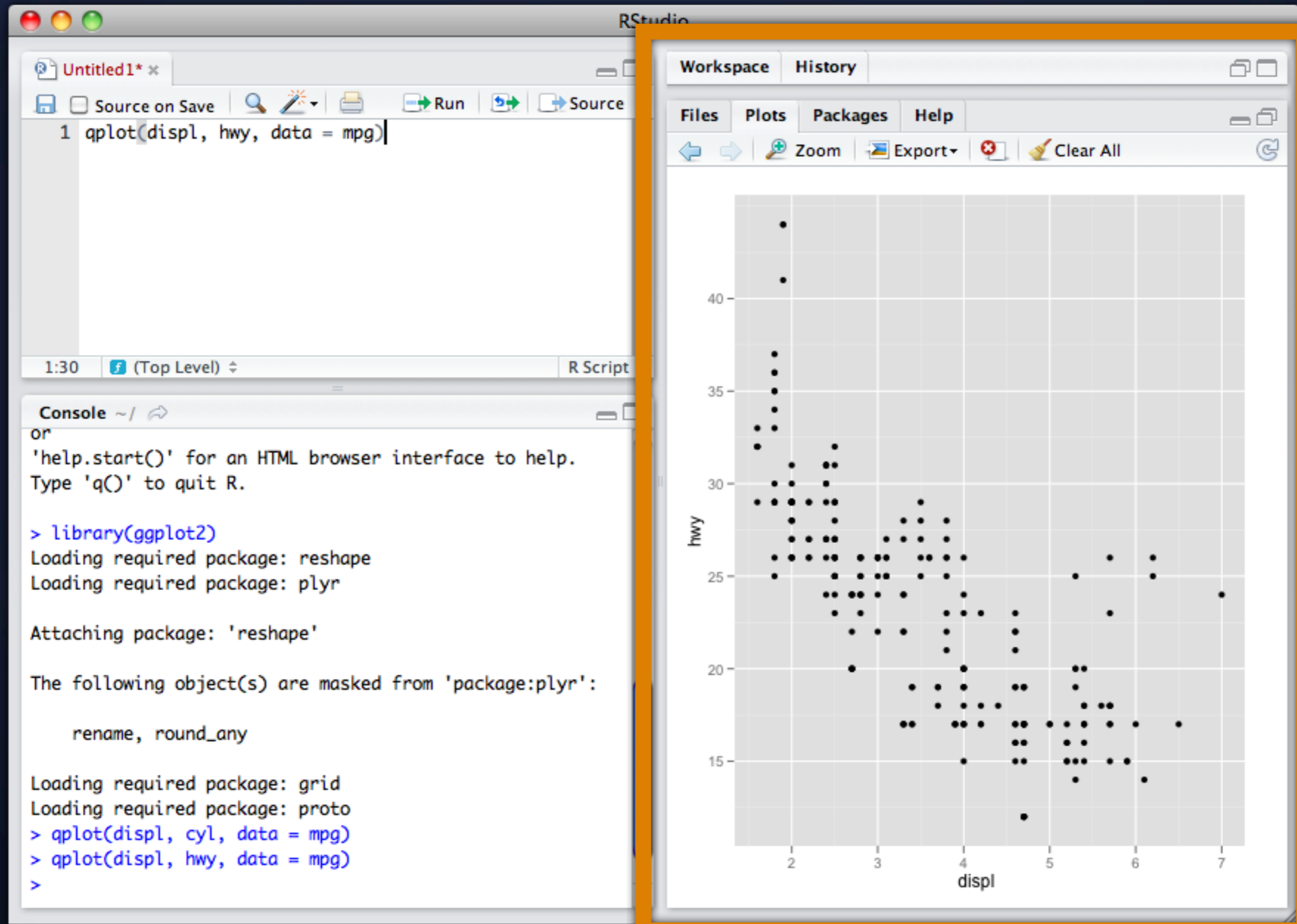
The bottom-left pane is the Console, which shows the following output:

```
or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> library(ggplot2)  
Loading required package: reshape  
Loading required package: plyr  
  
Attaching package: 'reshape'  
  
The following object(s) are masked from 'package:plyr':  
  
  rename, round_any  
  
Loading required package: grid  
Loading required package: proto  
> qplot(displ, cyl, data = mpg)  
> qplot(displ, hwy, data = mpg)  
>
```

The top-right pane is the Plots window, displaying a scatter plot of highway mileage (hwy) versus engine displacement (displ) for the mpg dataset. The plot shows a negative correlation between the two variables. The x-axis (displ) ranges from approximately 1.6 to 7.0, and the y-axis (hwy) ranges from approximately 12 to 44. The plot is styled with a light gray background and a white grid.

Rstudio

Output - plots and help



The image shows the RStudio interface. The top-left pane is the Source Editor, containing the following R code:

```
1 qplot(displ, hwy, data = mpg)
```

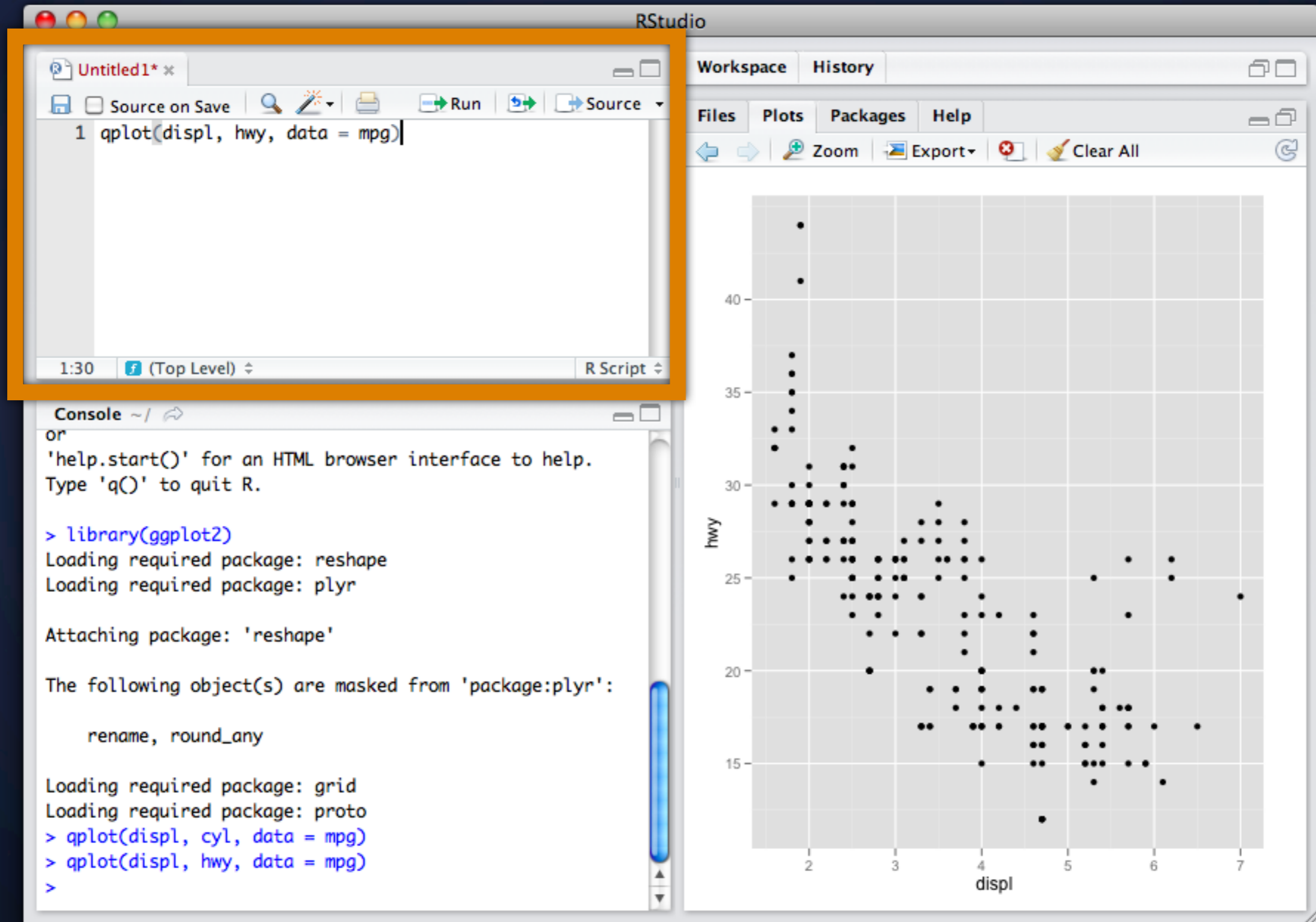
The bottom-left pane is the Console, showing the following output:

```
or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> library(ggplot2)  
Loading required package: reshape  
Loading required package: plyr  
  
Attaching package: 'reshape'  
  
The following object(s) are masked from 'package:plyr':  
  
  rename, round_any  
  
Loading required package: grid  
Loading required package: proto  
> qplot(displ, cyl, data = mpg)  
> qplot(displ, hwy, data = mpg)  
>
```

The top-right pane is the Plots window, which displays a scatter plot of highway mileage (hwy) versus engine displacement (displ) for the mpg dataset. The plot features a light gray grid and black circular markers. The x-axis (displ) ranges from approximately 1.6 to 7.0, and the y-axis (hwy) ranges from approximately 12 to 44. The plot shows a clear negative correlation between engine displacement and highway mileage.

Rstudio

Editor - Save code here



The screenshot displays the RStudio environment. The top-left pane is the Source Editor, containing the following R code:

```
1 qplot(displ, hwy, data = mpg)
```

The bottom-left pane is the Console, showing the following output:

```
or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> library(ggplot2)  
Loading required package: reshape  
Loading required package: plyr  
  
Attaching package: 'reshape'  
  
The following object(s) are masked from 'package:plyr':  
  
  rename, round_any  
  
Loading required package: grid  
Loading required package: proto  
> qplot(displ, cyl, data = mpg)  
> qplot(displ, hwy, data = mpg)  
>
```

The top-right pane shows the Workspace and History tabs. The bottom-right pane displays a scatter plot of highway mileage (hwy) versus engine displacement (displ) for the mpg dataset. The plot shows a negative correlation between the two variables. The x-axis (displ) ranges from approximately 1.6 to 7.0, and the y-axis (hwy) ranges from approximately 12 to 44. The plot is styled with a light gray background and a white grid.

Shortcuts

Learn them!

In editor

Cmd/ctrl + enter – send code to console

ctrl + 2 – move cursor to console

In console

Up arrow – retrieve previous command

ctrl + up arrow – search commands

ctrl + 1 – move cursor to editor

Scatter Plot Basics

```
install.packages("ggplot2")  
library(ggplot2)  
  
?mpg  
head(mpg)  
str(mpg)  
summary(mpg)  
  
qplot(displ, hwy, data = mpg)
```

Scatter Plot Basics

```
install.packages("ggplot2")  
library(ggplot2)
```

```
?mpg
```

```
head(mpg)
```

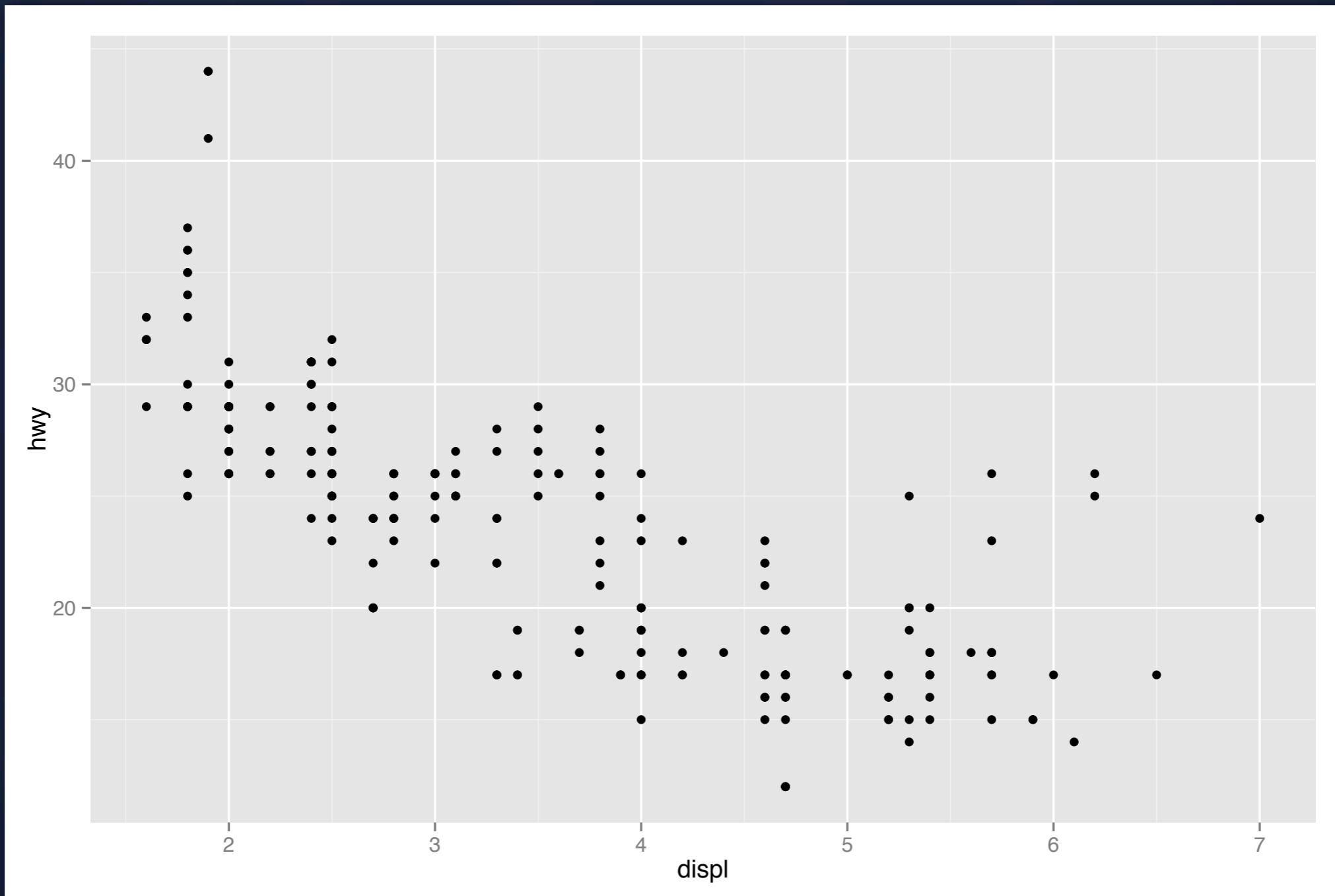
```
str(mpg)
```

```
summary(mpg)
```

```
ggplot(displ, hwy, data = mpg)
```

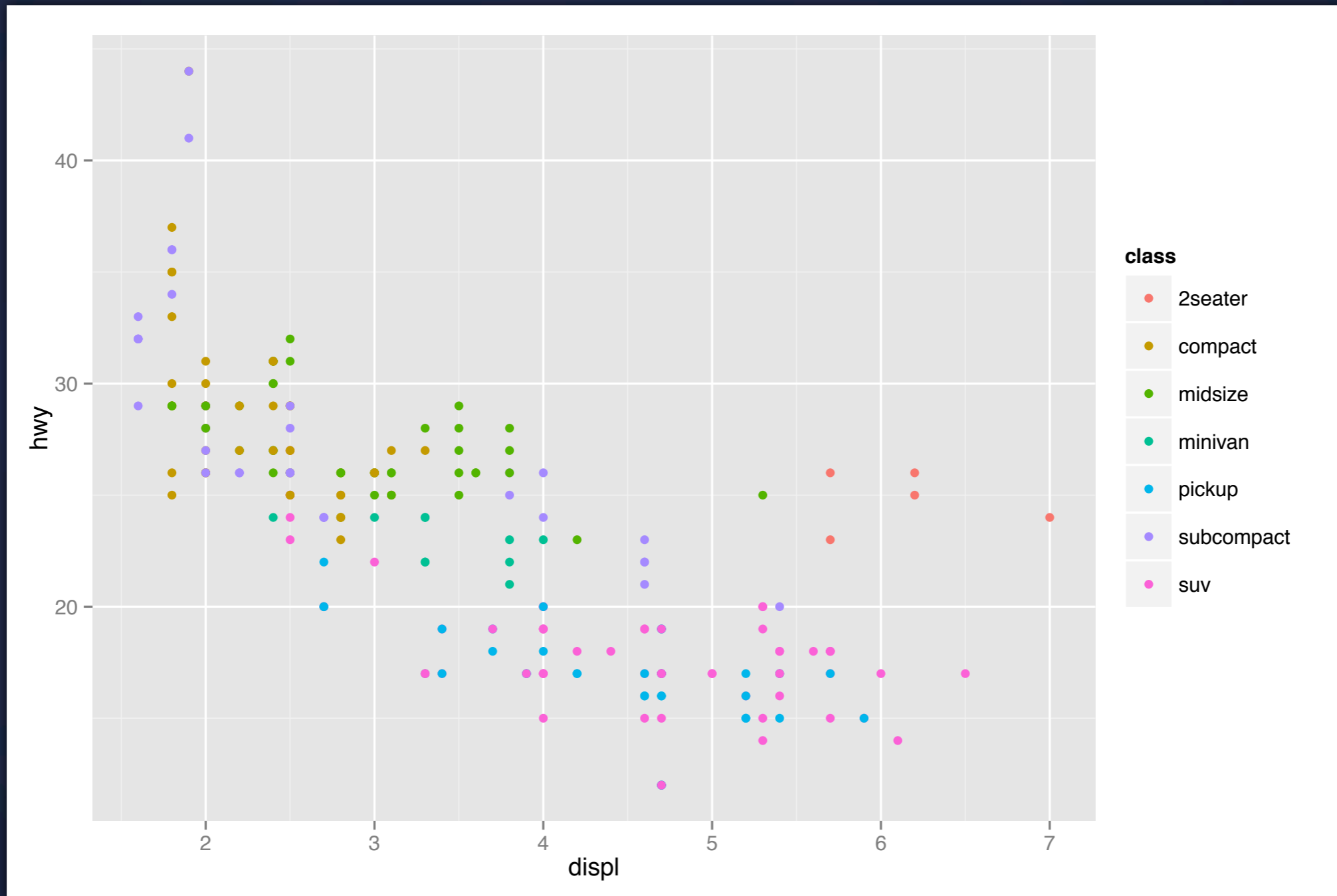
*always explicitly
specify the data*

Scatter Plot Basics



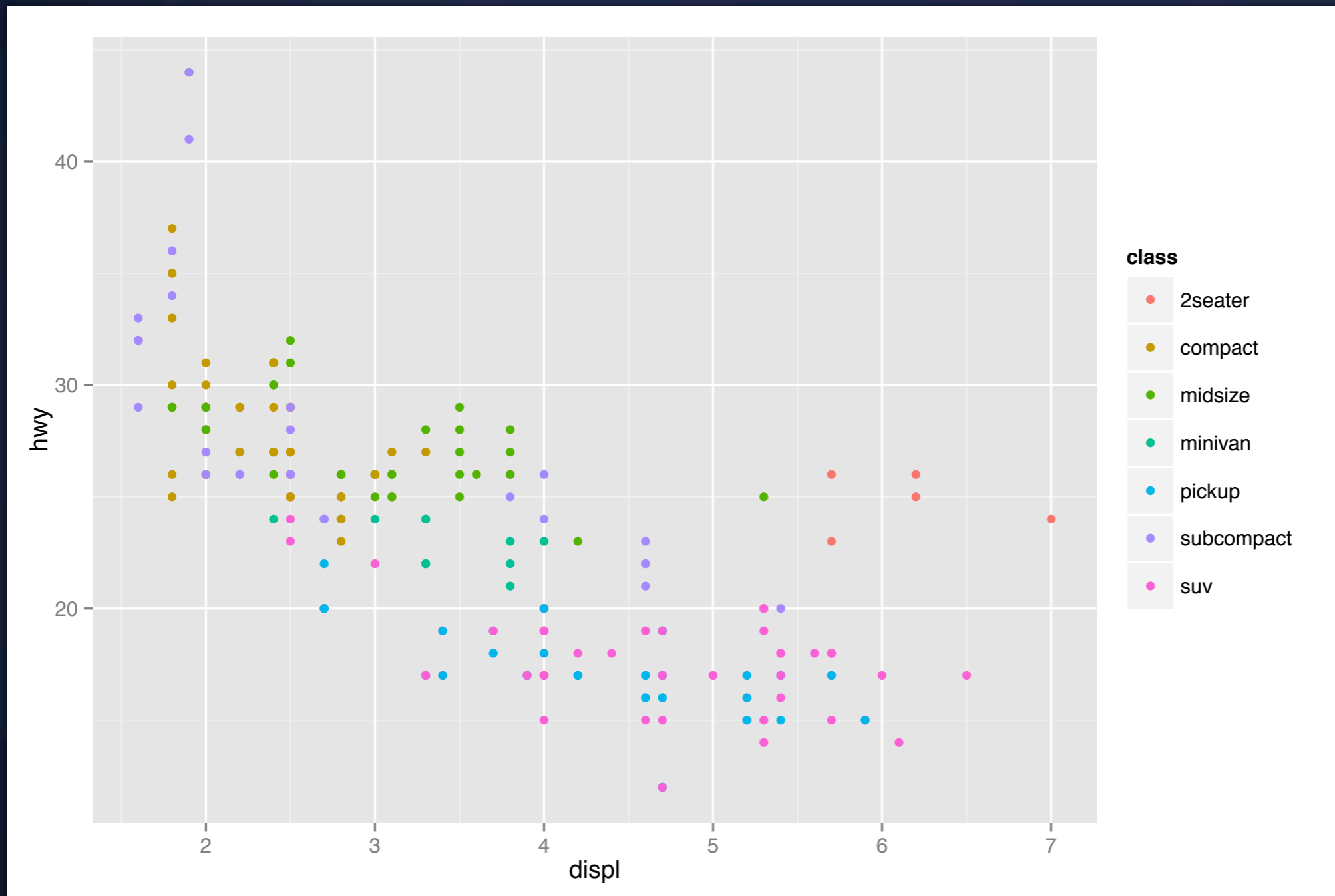
```
qplot(displ, hwy, data = mpg)
```

Additional Dimensions?



```
qplot(displ, hwy, colour=class, data=mpg)
```

Additional Dimensions?



*legend chosen
and displayed
automatically*

```
qplot(displ, hwy, colour=class, data=mpg)
```

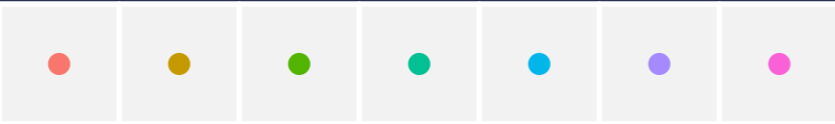


Your Turn

Experiment with color, size, and shape aesthetics.

What's the difference between discrete or continuous variables?

What happens when you combine multiple aesthetics?

Discrete vs Continuous variables

	Discrete	Continuous
Color		
Size	discrete size steps	Linear mapping between radius and value
Shape		?

Faceting

= Small Multiples

Your turn!

```
qplot(displ, hwy, data=mpg) + facet_grid(. ~ cyl)
```

```
qplot(displ, hwy, data=mpg) + facet_grid(drv ~ .)
```

```
qplot(displ, hwy, data=mpg) + facet_grid(drv ~ cyl)
```

```
qplot(displ, hwy, data=mpg) + facet_wrap(~ class)
```


Faceting

= Small Multiples

Your turn!

```
qplot(displ, hwy, data=mpg) + facet_grid(. ~ cyl)
```

```
qplot(displ, hwy, data=mpg) + facet_grid(drv ~ .)
```

```
qplot(displ, hwy, data=mpg) + facet_grid(drv ~ cyl)
```

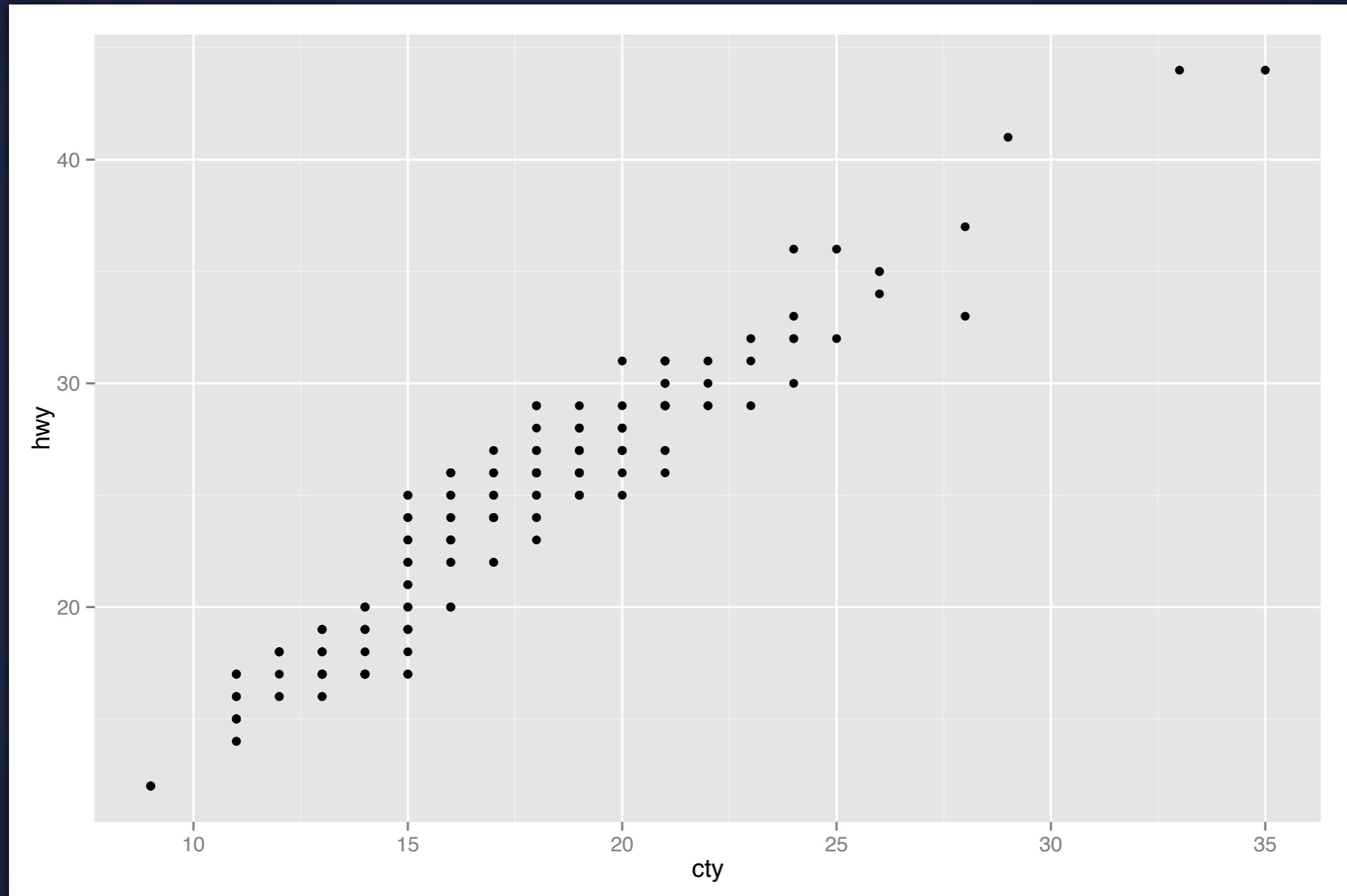
```
qplot(displ, hwy, data=mpg) + facet_wrap(~ class)
```

Summary

`facet_grid()` 2d grid, rows ~ cols, . for no split

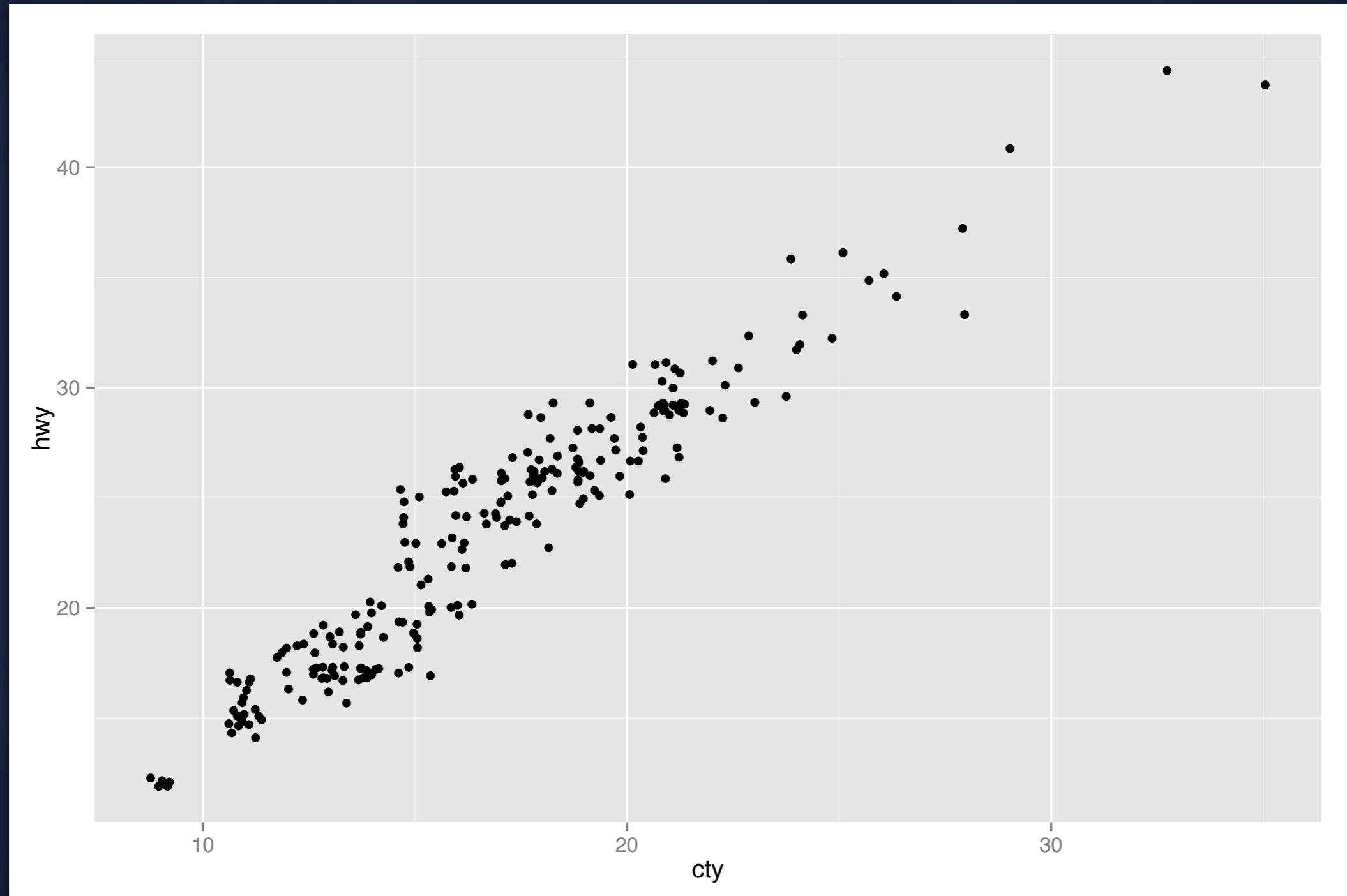
`facet_wrap()` 1d ribbon wrapped into 2d

What's the problem here?



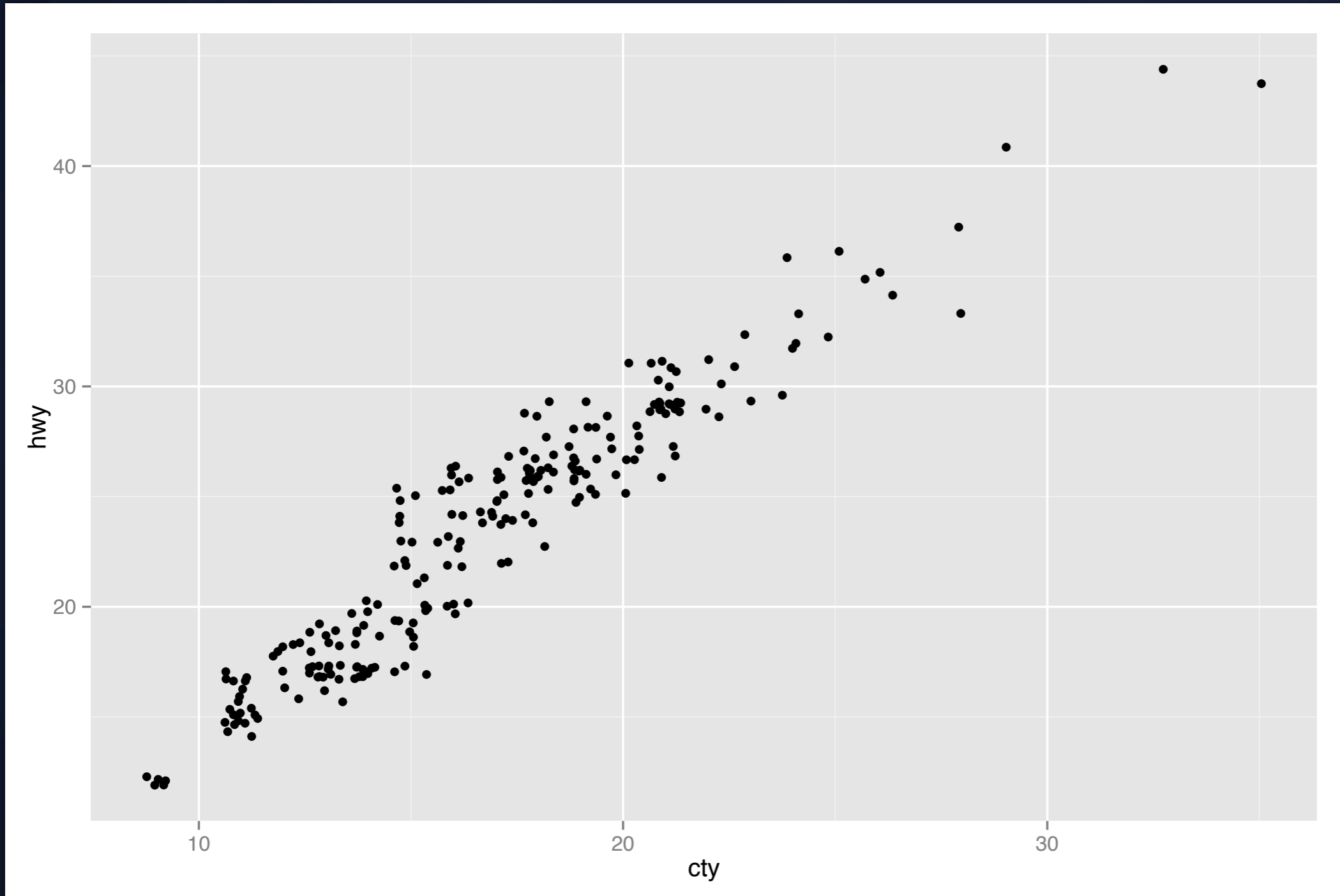
```
qplot(cty, hwy, data = mpg)
```

What's the problem here?



```
ggplot(cty, hwy, data = mpg, geom = "jitter")
```

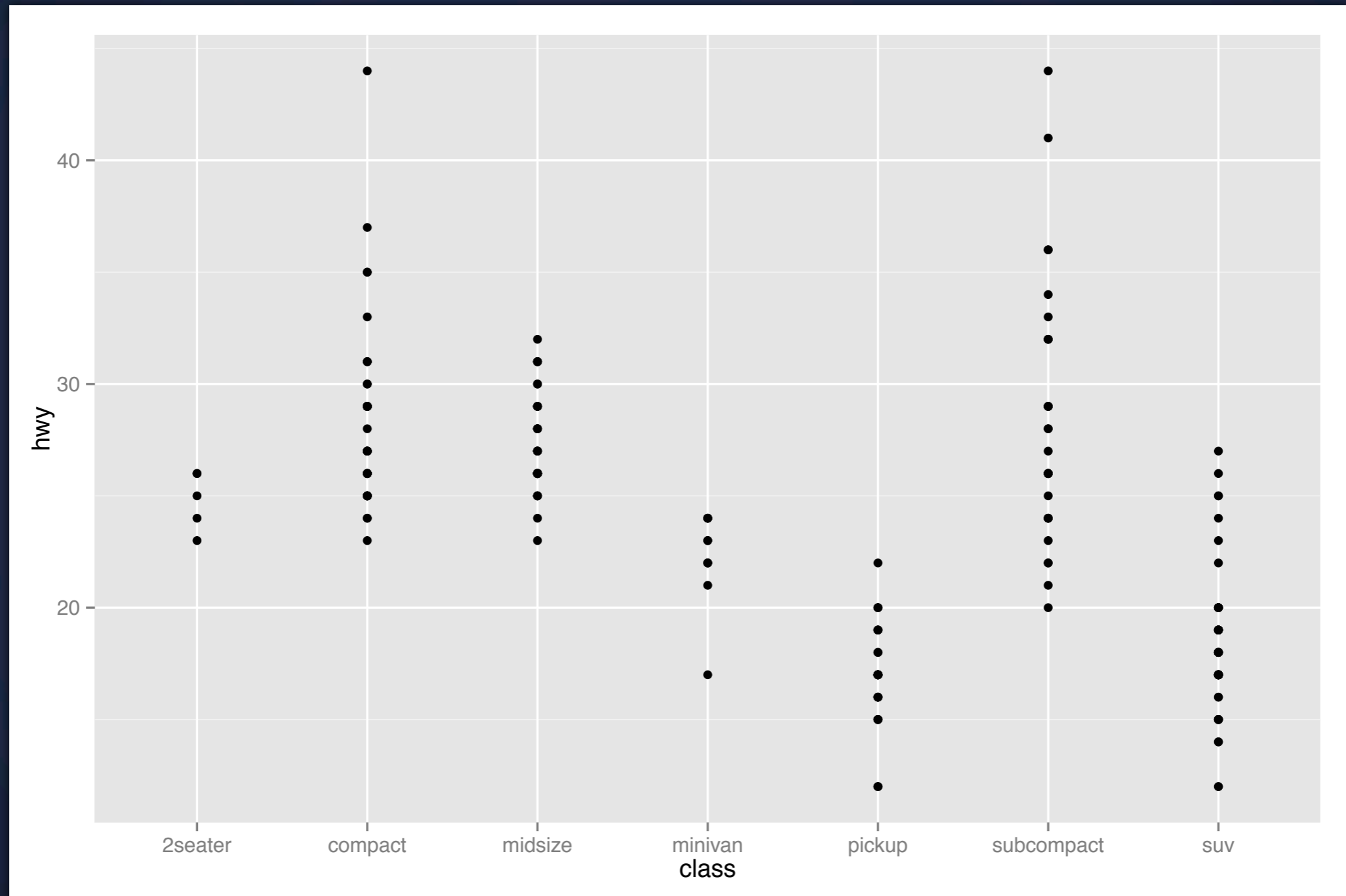
What's the problem here?



*geom
controls type
of plot*

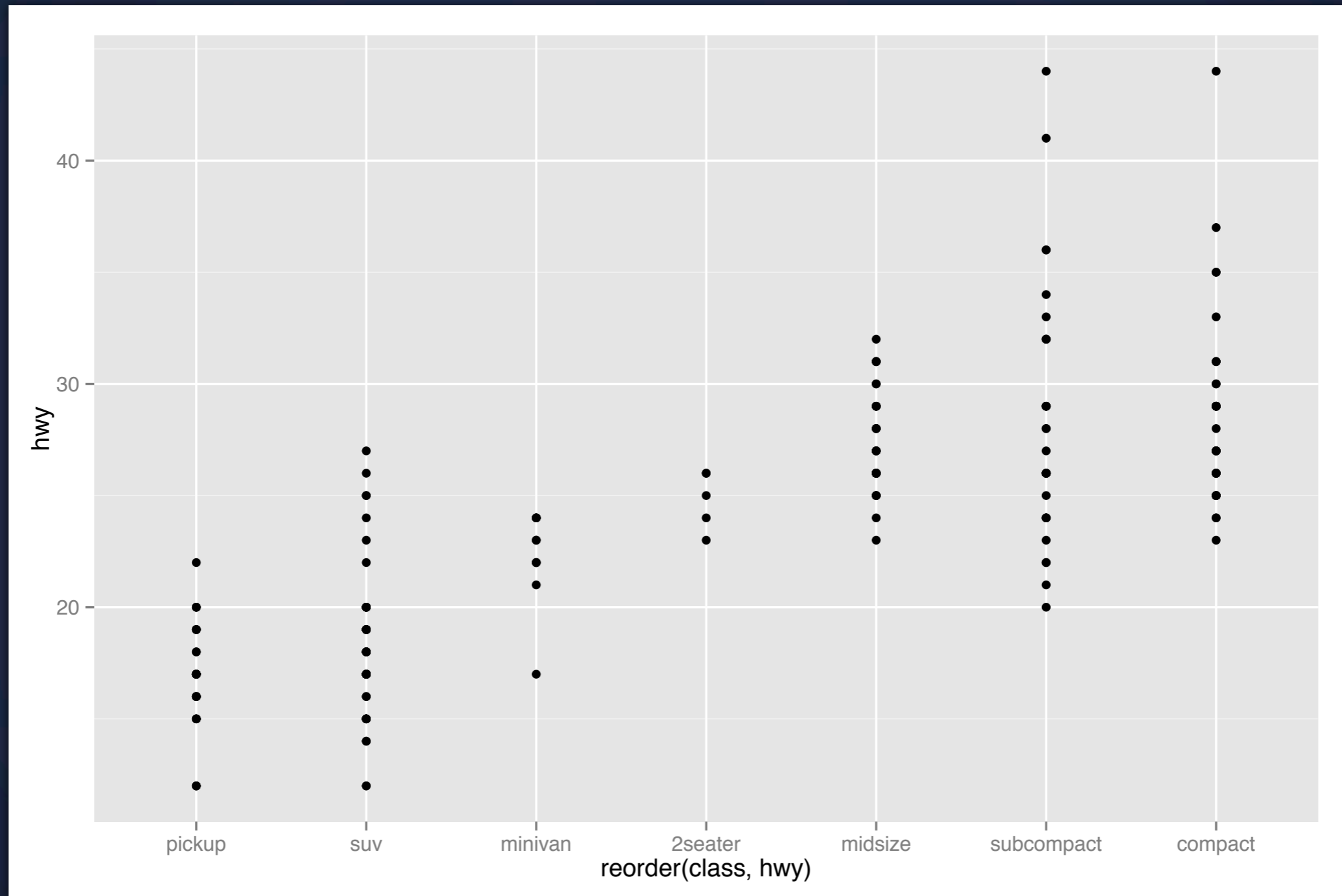
```
qplot(cty, hwy, data = mpg, geom = "jitter")
```

How can we improve this plot?



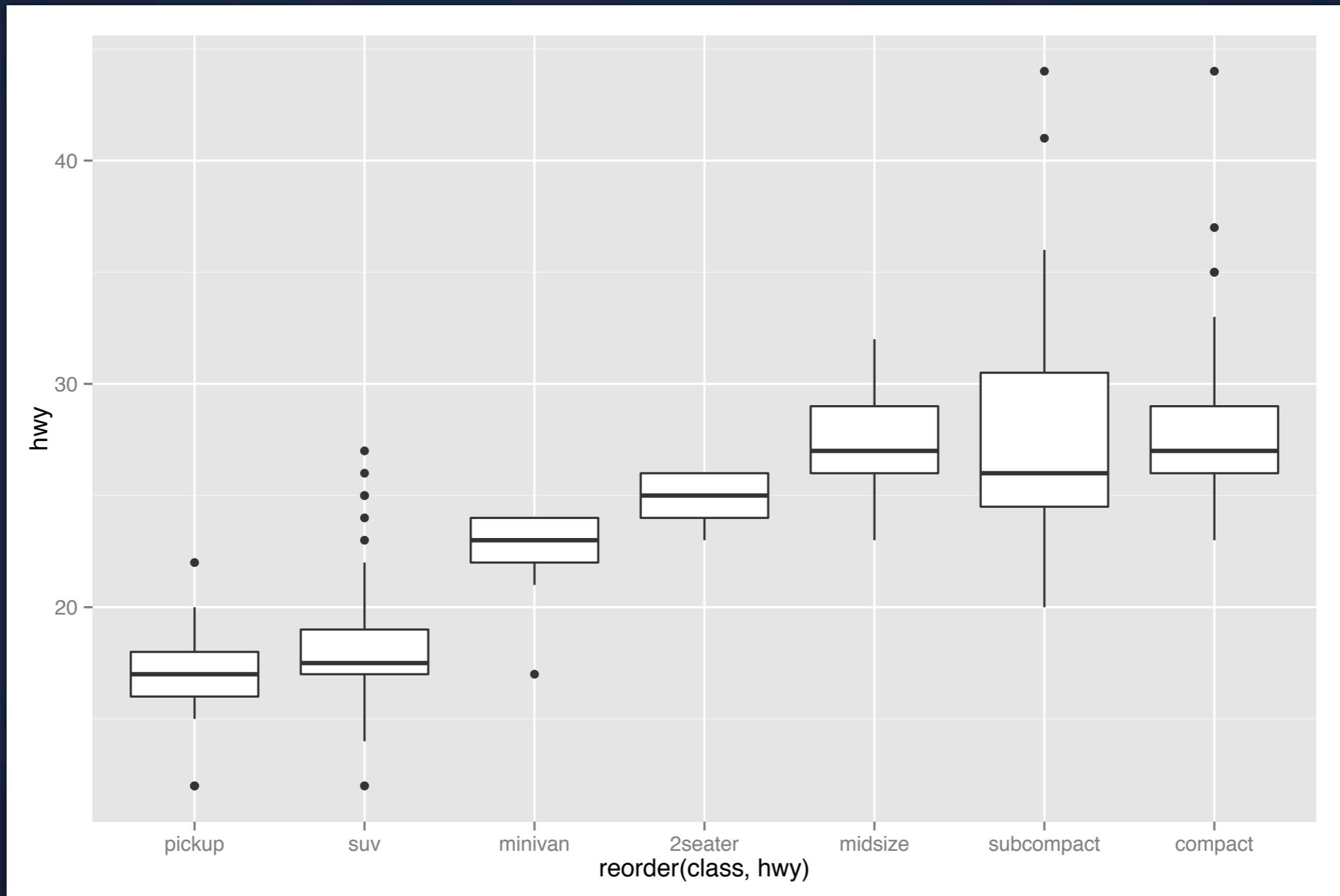
```
qplot(class, hwy, data = mpg)
```

How can we improve this plot?



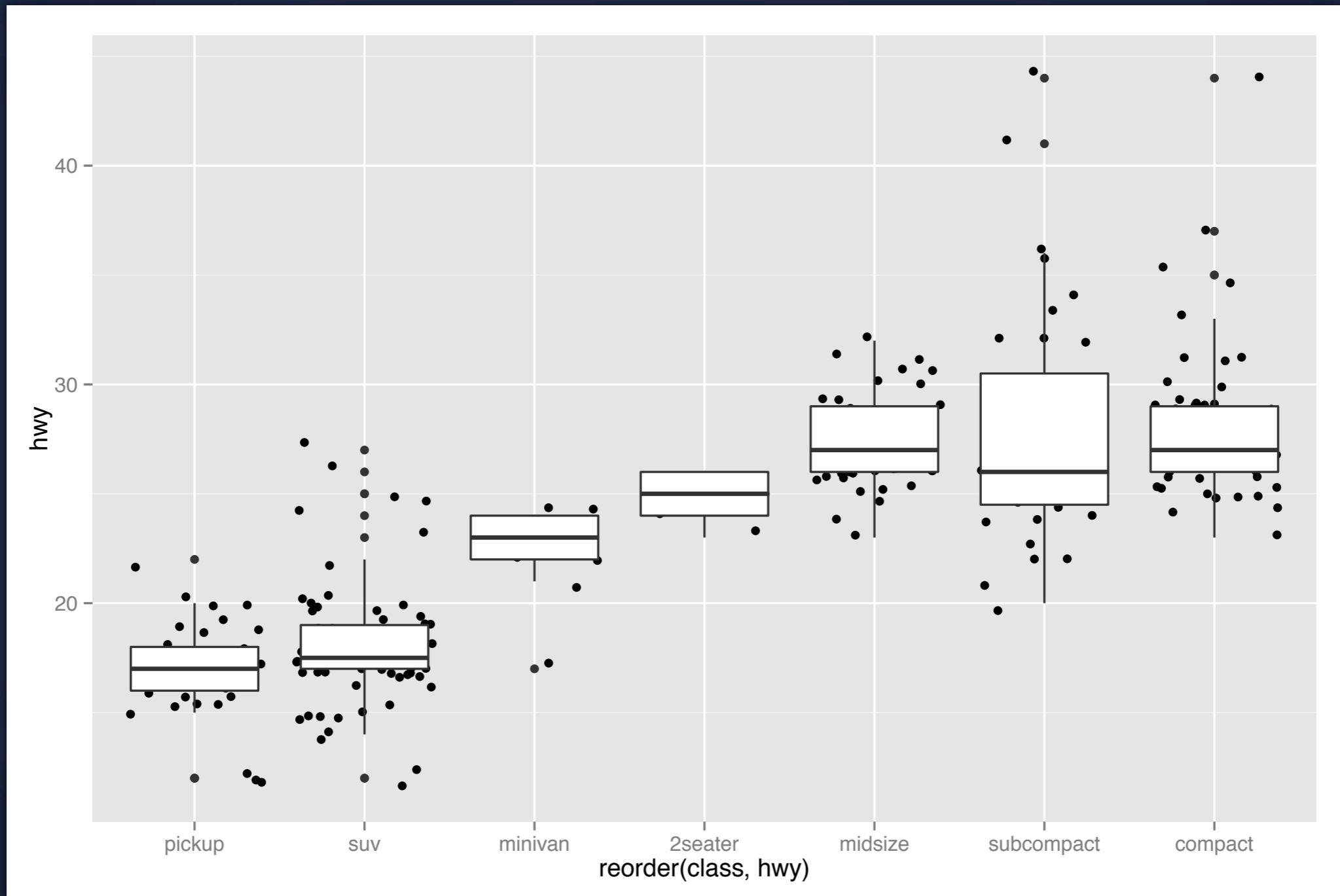
```
qplot(reorder(class, hwy), hwy, data = mpg)
```

How can we improve this plot?



```
ggplot(reorder(class, hwy), hwy, data = mpg, geom = "boxplot")
```

How can we improve this plot?



```
ggplot(reorder(class, hwy), hwy, data=mpg, geom=c("jitter", "boxplot"))
```


Your Turn

Read the help for `reorder`. Redraw the previous plots with class ordered by `median hwy`.

How would you put the jittered points on top of the boxplots?

Diamonds

A bigger data set

Diamonds data

~54,000 round diamonds from
<http://www.diamondse.info>

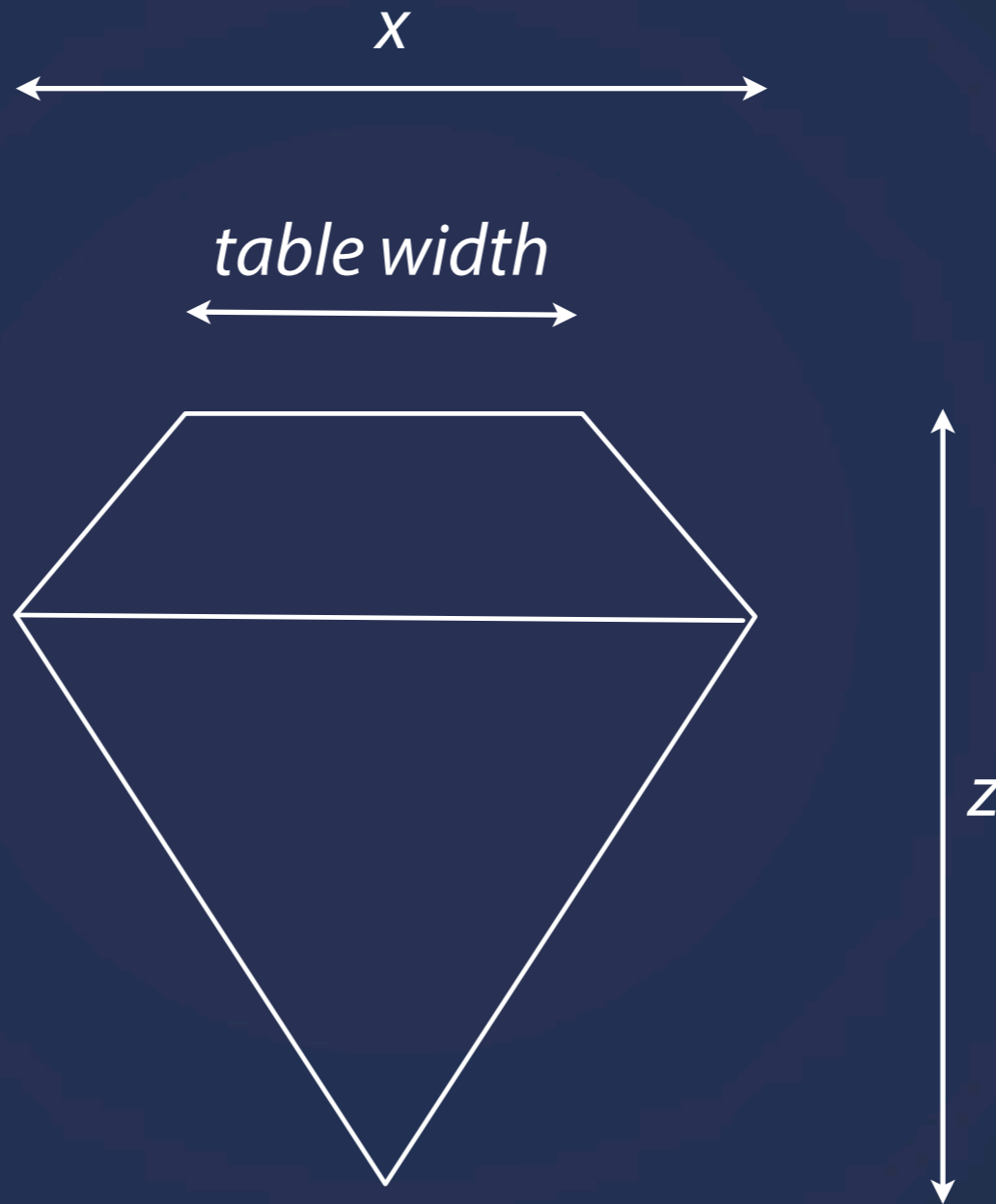
Carat, colour, clarity, cut

Total depth, table, depth, width, height

Price



Metrics of a diamond

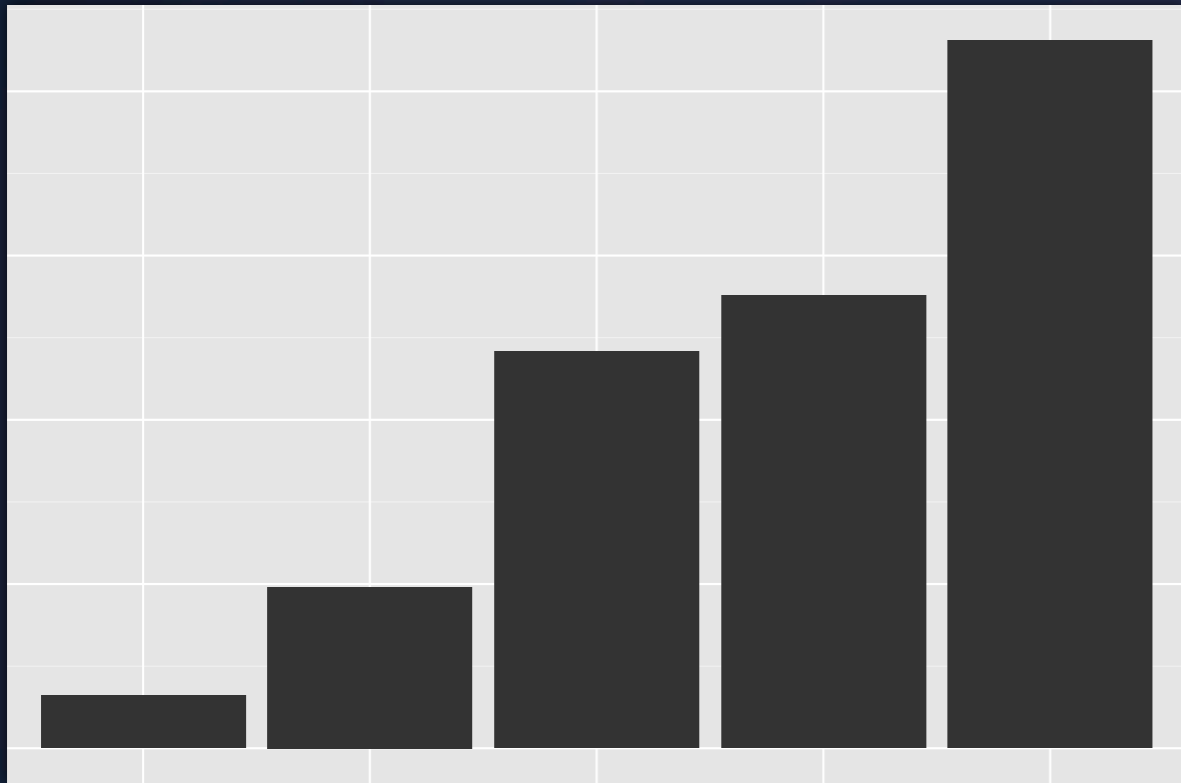


$$\text{depth} = z / \text{diameter}$$
$$\text{table} = \text{table width} / x * 100$$

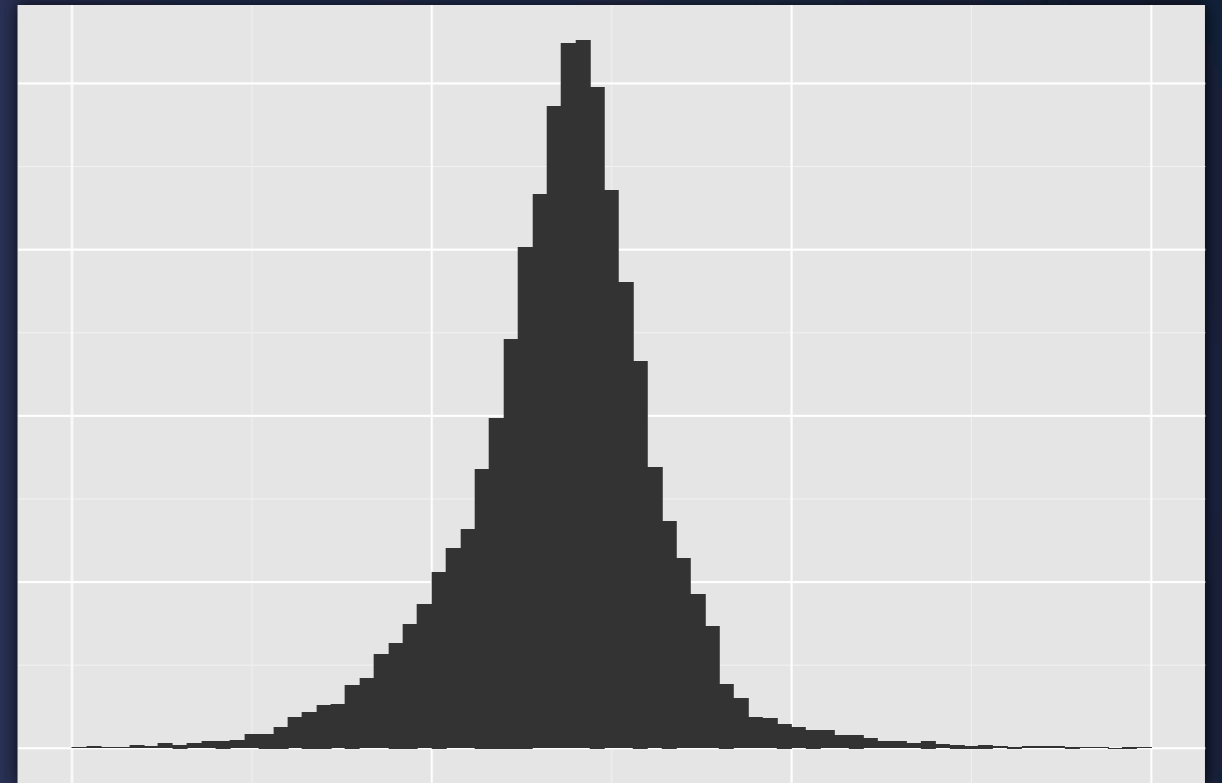
Your Turn

Inspect the data set

Barcharts vs Histograms



Nominal/categorical
variables



Continuous
variables

Let's plot

With only one variable, qplot guesses that you want a bar chart or histogram

```
qplot(cut, data = diamonds)
```

```
qplot(carat, data = diamonds)
```

```
# Change binwidth
```

```
qplot(carat, data = diamonds, binwidth = 1)
```

```
qplot(carat, data = diamonds, binwidth = 0.1)
```

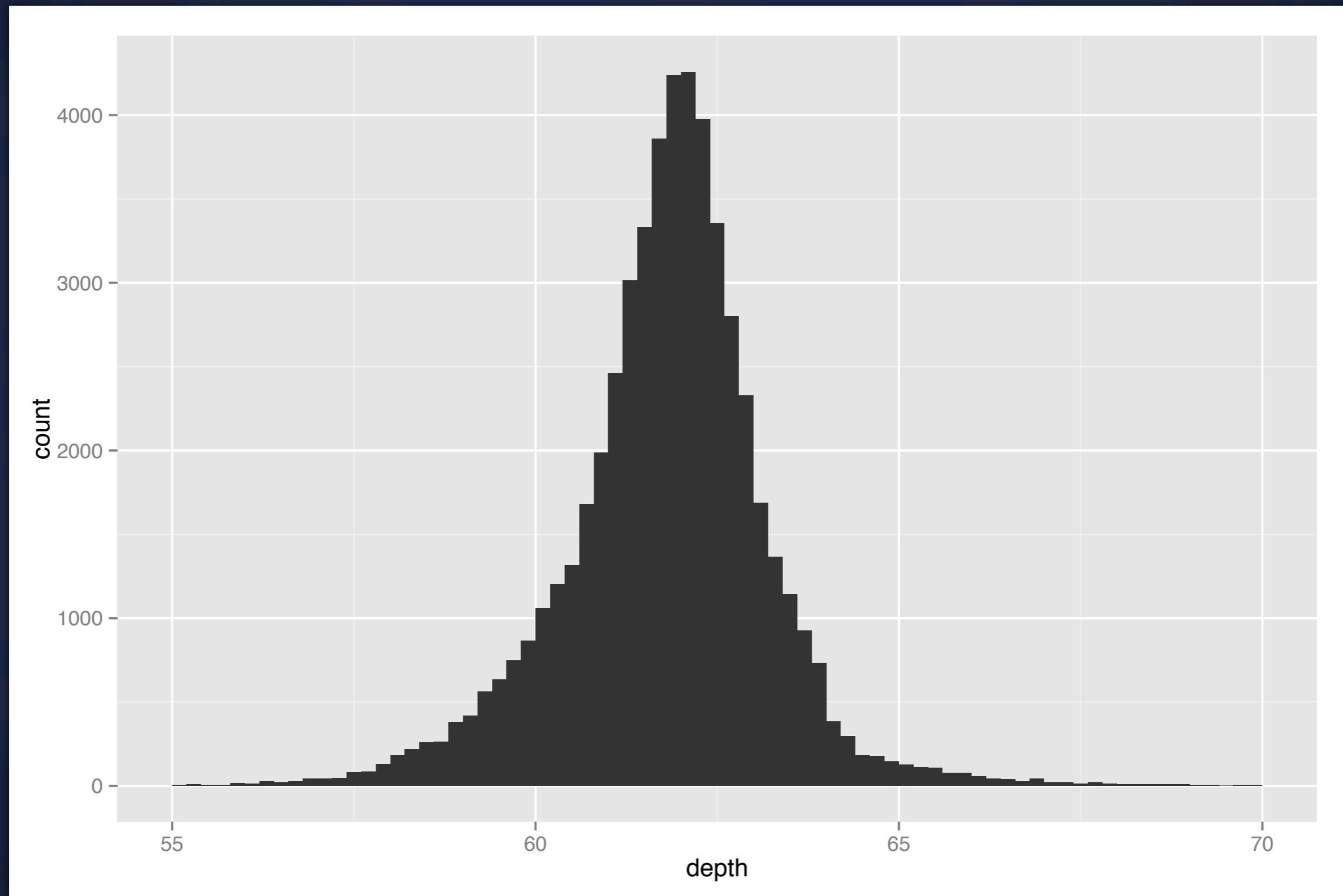
```
qplot(carat, data = diamonds, binwidth = 0.01)
```

```
last_plot() + xlim(0, 3)
```

```
resolution(diamonds$carat)
```

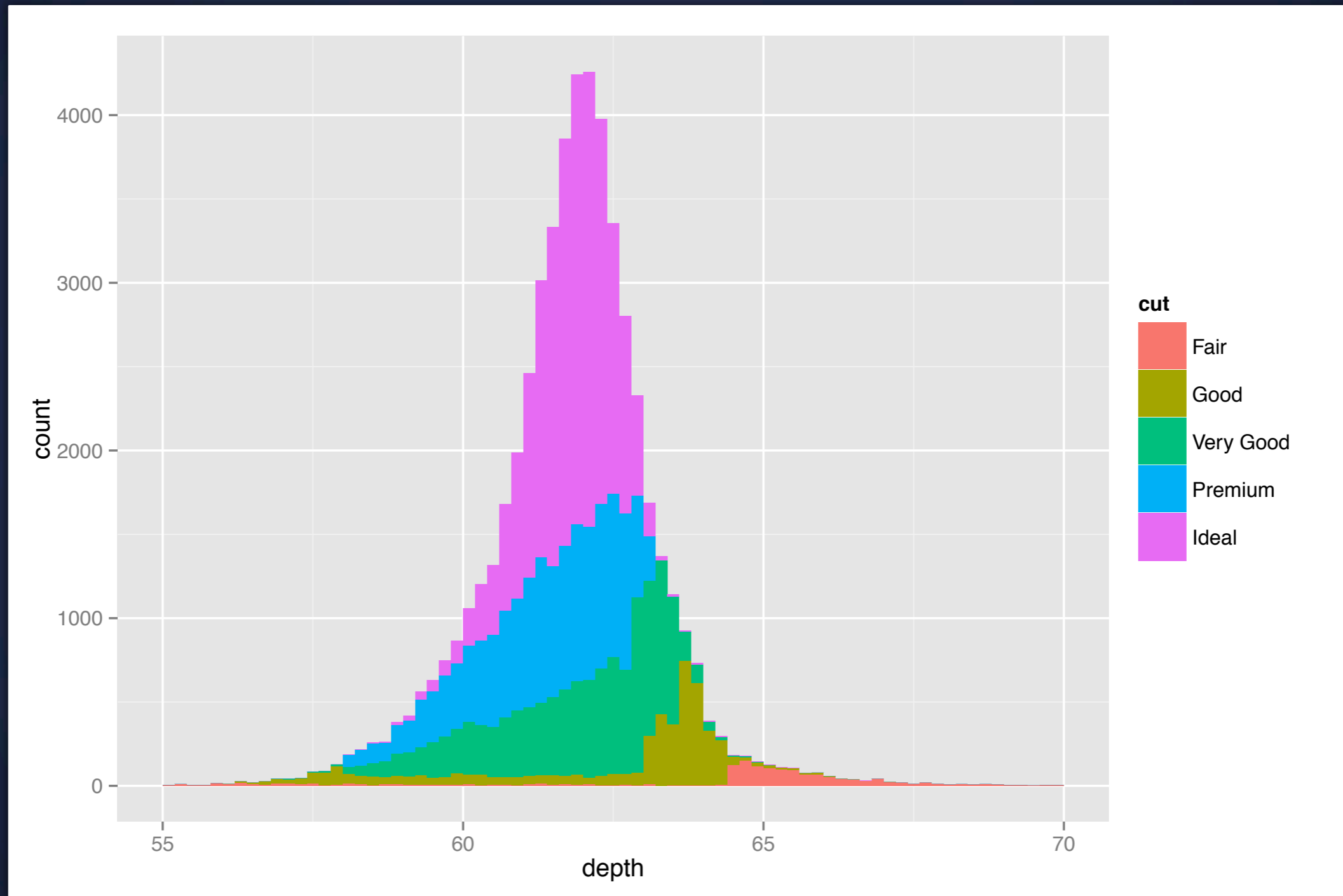
**Always experiment
with the bin width!**

Additional Dimensions



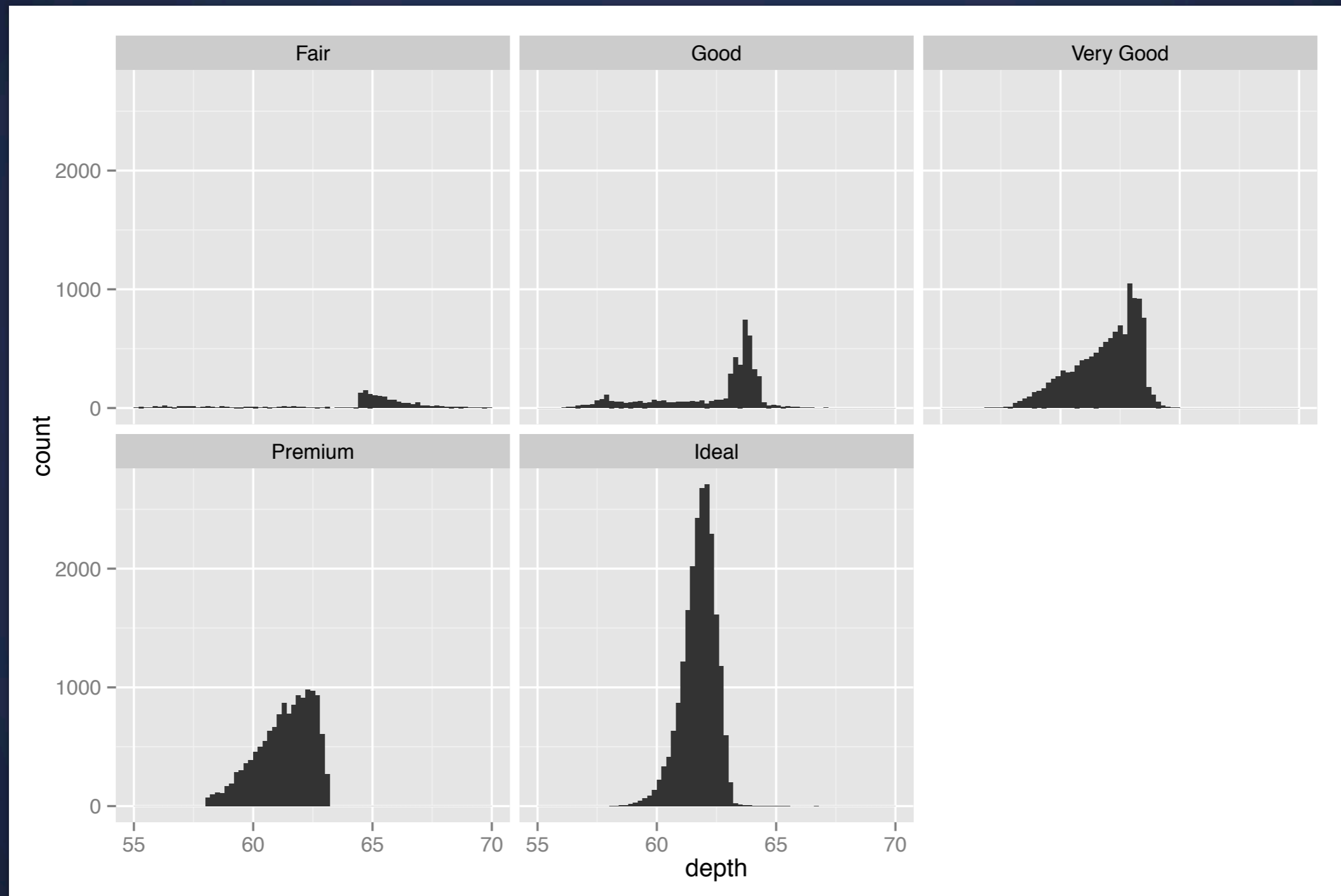
```
qplot(depth, data = diamonds, binwidth = 0.2) + xlim(55, 70)
```

Additional Dimensions



```
qplot(depth, data = diamonds, binwidth = 0.2, fill = cut) + xlim(55, 70)
```

Additional Dimensions



```
ggplot(depth, data = diamonds, binwidth = 0.2, fill = cut) + xlim(55, 70)
  + facet_wrap(~ cut)
```

Your Turn

Explore the distribution of price. What is a good binwidth to use?

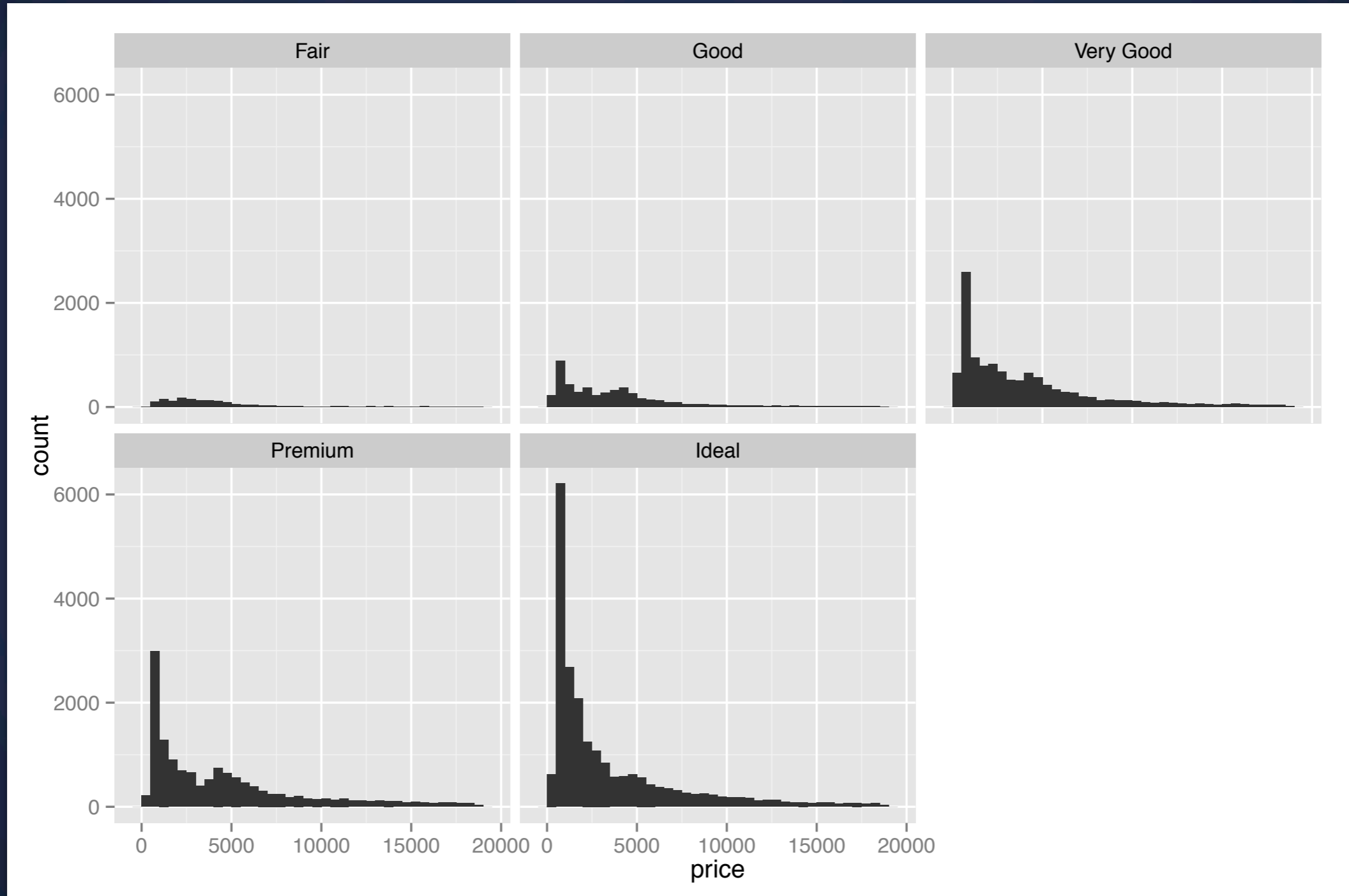
Hint: Diamonds are expensive!

Practice zooming in on regions of interest.

How does price vary with color, cut, or clarity?

Frequency Histogram

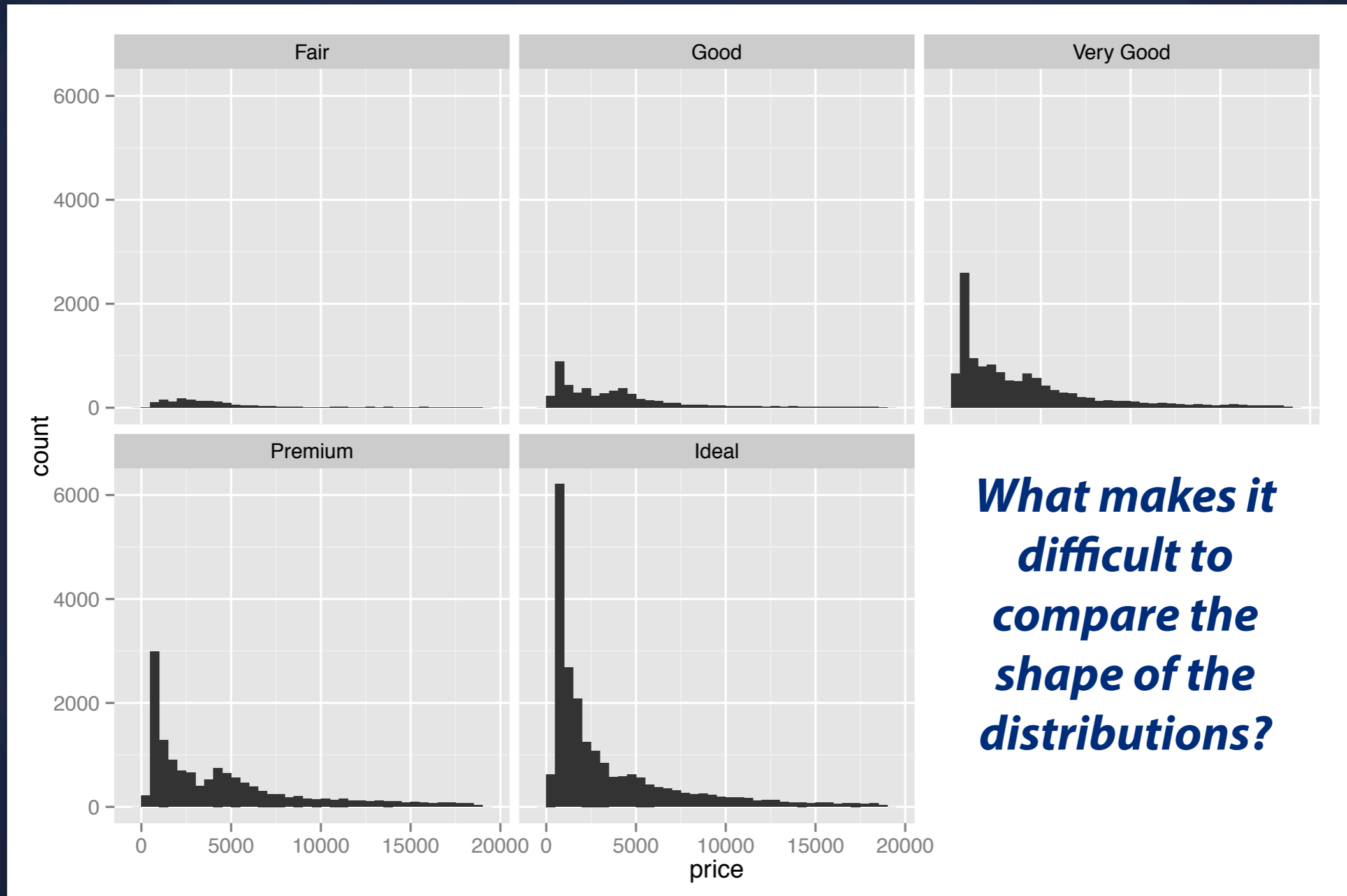
They're helpful, but come with caveats



```
qplot(price, data = diamonds, binwidth = 500) + facet_wrap(~ cut)
```

Frequency Histogram

They're helpful, but come with caveats



```
ggplot(price, data = diamonds, binwidth = 500) + facet_wrap(~ cut)
```

Frequency Histogram

They're helpful, but come with caveats

```
# Large distances make comparisons hard
```

```
qplot(price, data = diamonds, binwidth = 500) +  
facet_wrap(~ cut)
```

```
# Stacked heights hard to compare
```

```
qplot(price, data = diamonds, binwidth = 500, fill = cut)
```

```
# Much better - but still have differing relative  
abundance
```

```
qplot(price, data = diamonds, binwidth = 500,  
      geom = "freqpoly", colour = cut)
```

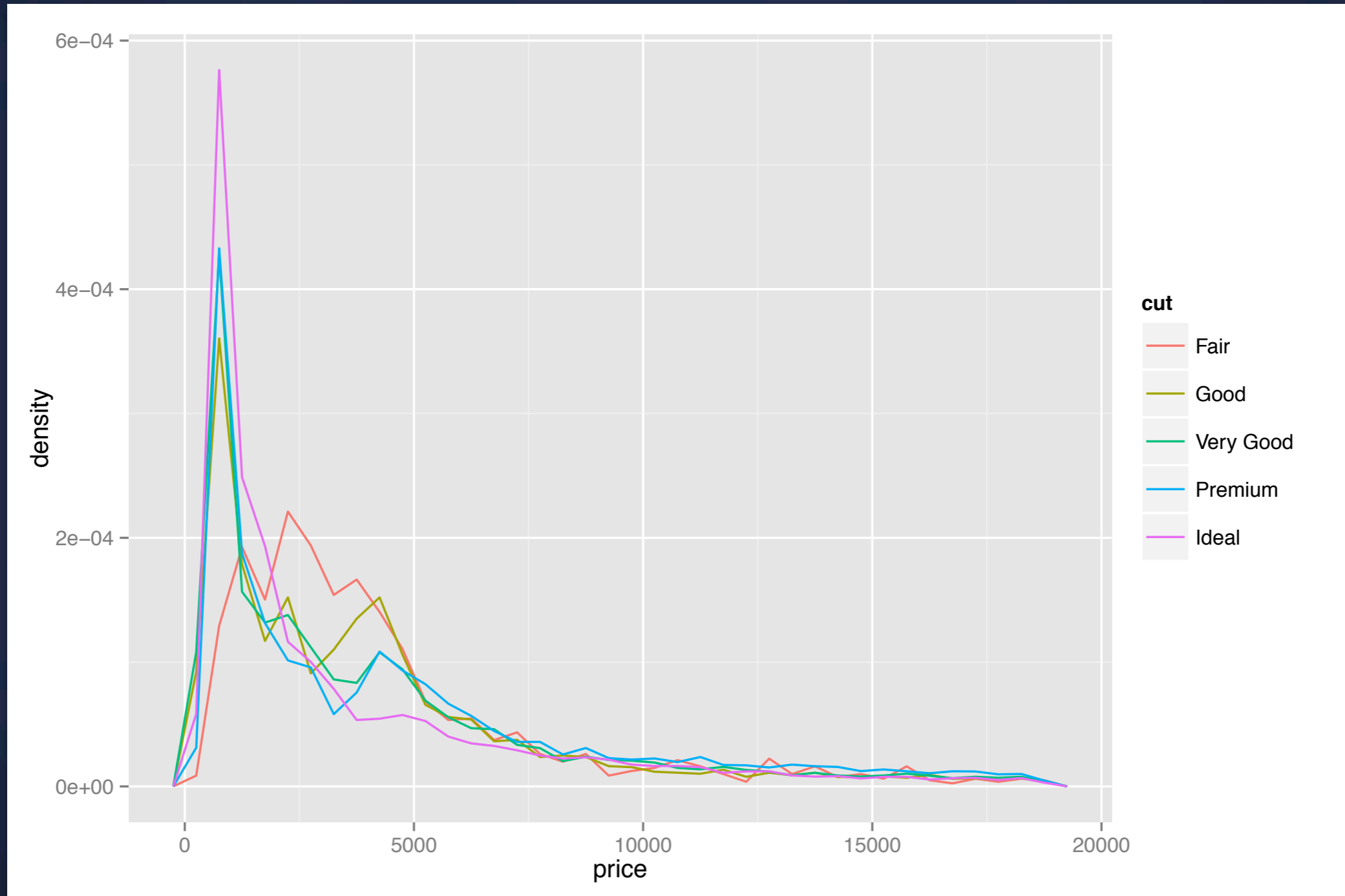
```
# Instead of displaying count on y-axis, display density
```

```
# .. indicates that variable isn't in original data
```

```
qplot(price, ..density.., data = diamonds, binwidth = 500,  
      geom = "freqpoly", colour = cut)
```

Density Histogram

Shows relative distribution better



```
qplot(price, ..density.., data = diamonds, binwidth = 500,  
       geom = "freqpoly", colour = cut)
```


Where Next?

Learn more about

- Aggregating your data: `plyr`
- Working with dates: `lubridate`
- Regular expressions: `stringr`
- A consistent philosophy of data: google "tidy data"
- `ggplot2`: <http://blog.ggplot2.org/> + `ggplot2` mailing list

Other Resources

- The art of R programming <http://amzn.com/1593273843>
- Data manipulation with R <http://amzn.com/0387747303>
- <http://www.r-bloggers.com/>
- <http://stackoverflow.com/questions/tagged/r>

This work is licensed under the Creative Commons Attribution-Noncommercial 3.0 United States License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/3.0/us/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Next Lecture

Dashboards. Guest lecture by Stephen Few