

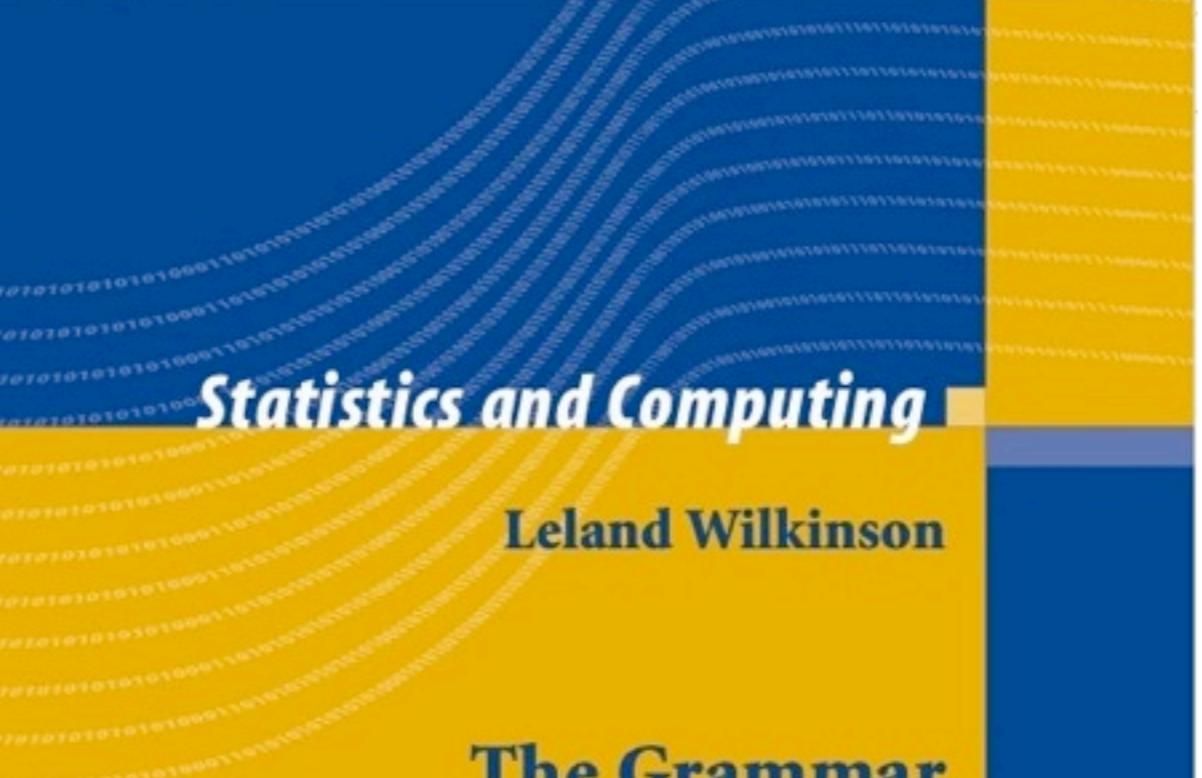
ggplot2

Sean Anderson (sean_anderson@sfu.ca)

FISH507H, Fall 2012

School of Aquatic and Fishery Sciences

University of Washington



Statistics and Computing

Leland Wilkinson

The Grammar of Graphics

Second Edition

 Springer

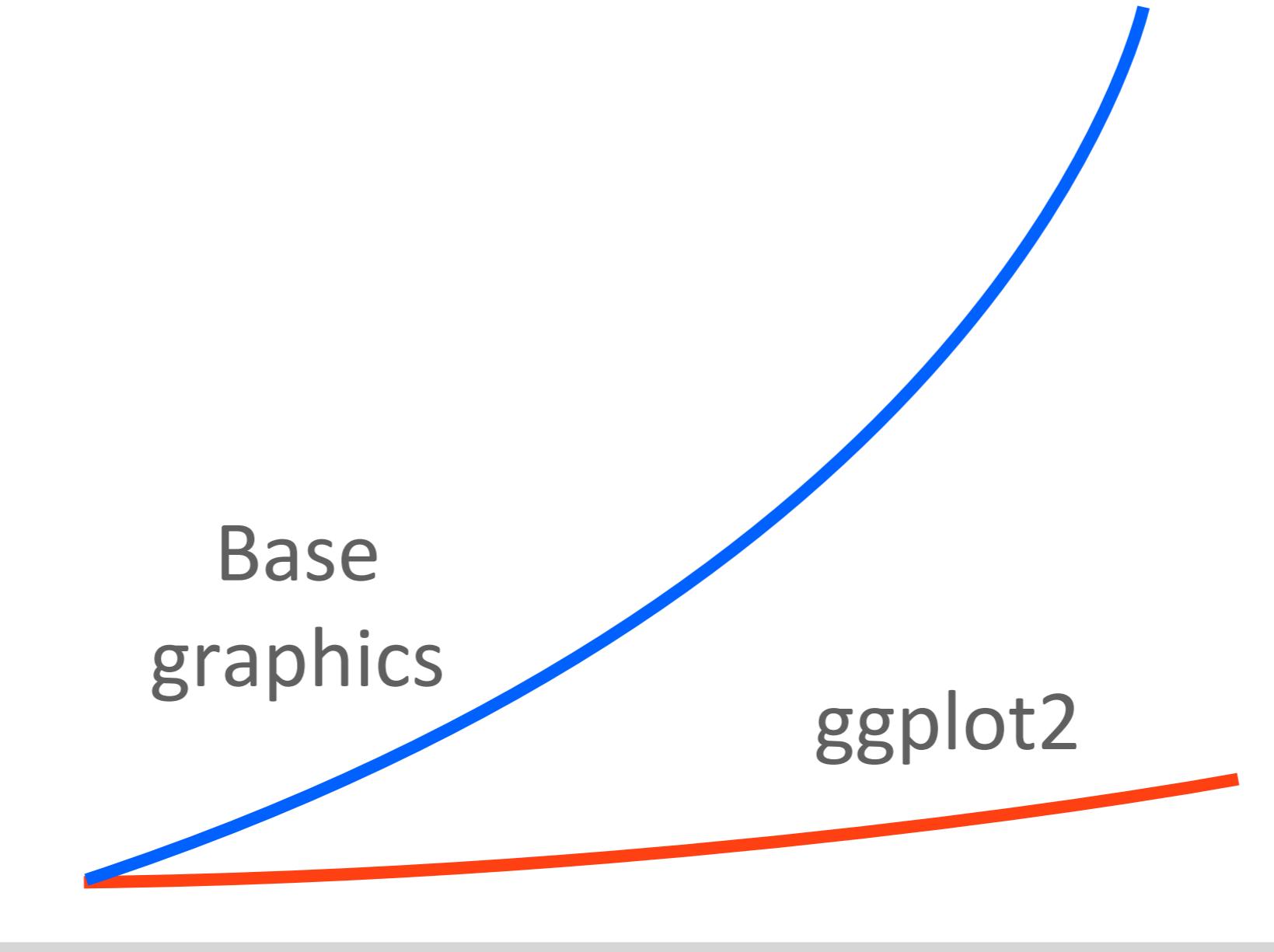


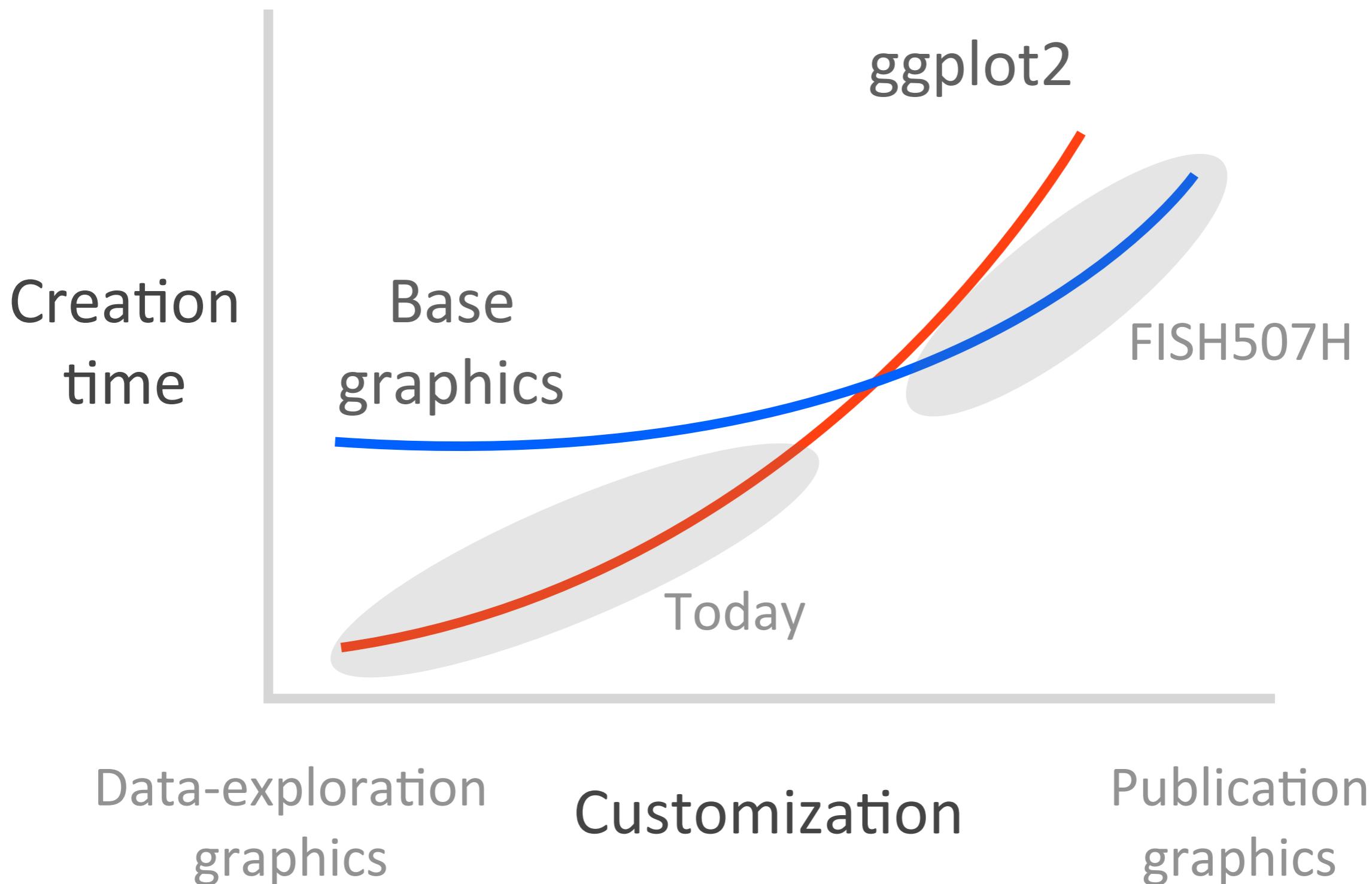
Creation
time

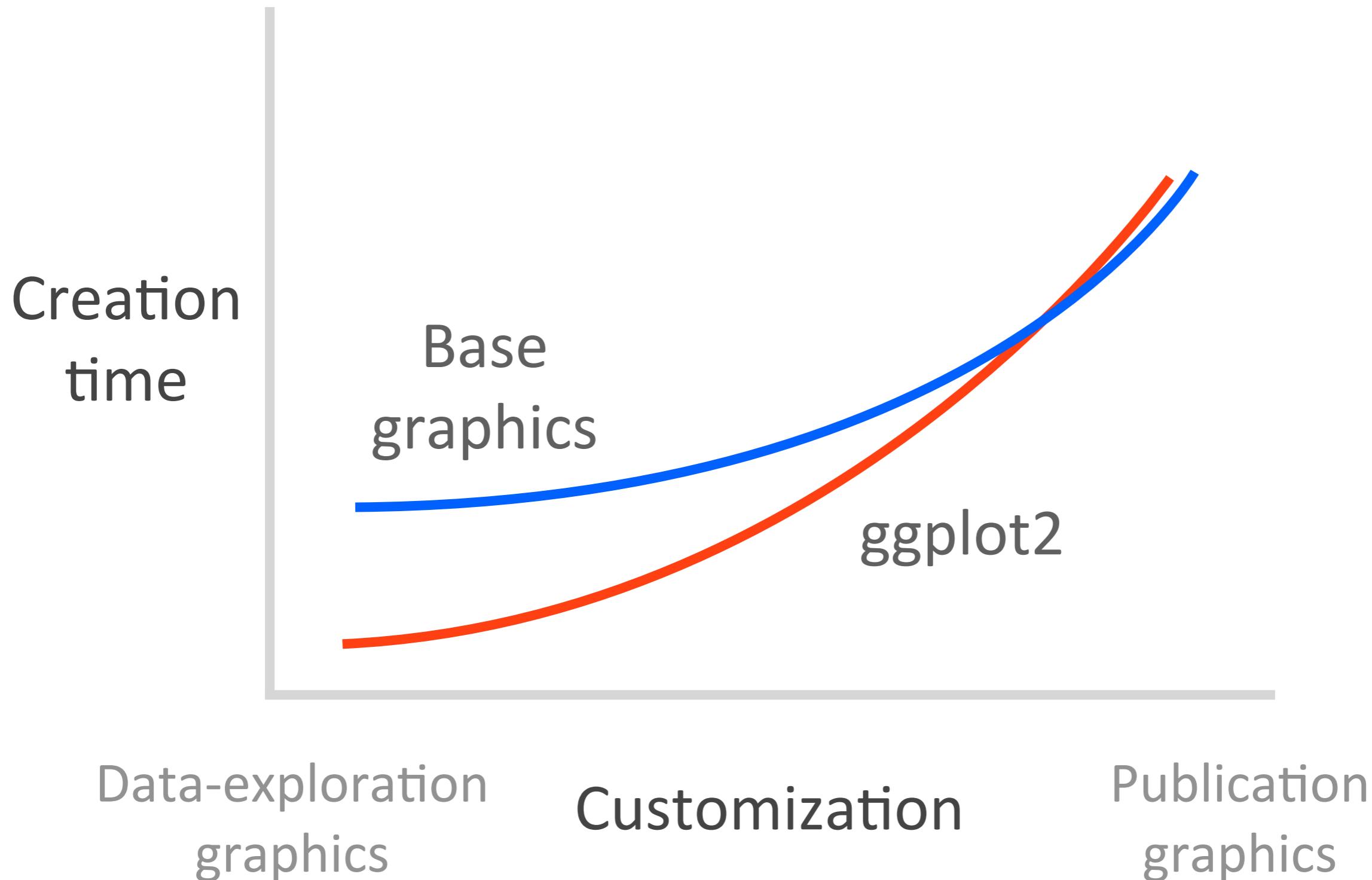
Base
graphics

ggplot2

Number of dimensions







London Cycle Hire Journeys

Thicker, yellower lines mean more journeys



James Cheshire
goo.gl/wK7EP

Data: 3.2 Million Journeys (from TfL)
Routing: Ollie O'Brien (@oobr) + OpenStreetMap cc-by-sa
Buildings: OS Opendata Crown Copyright 2011
Map: James Cheshire (@spatialanalysis)

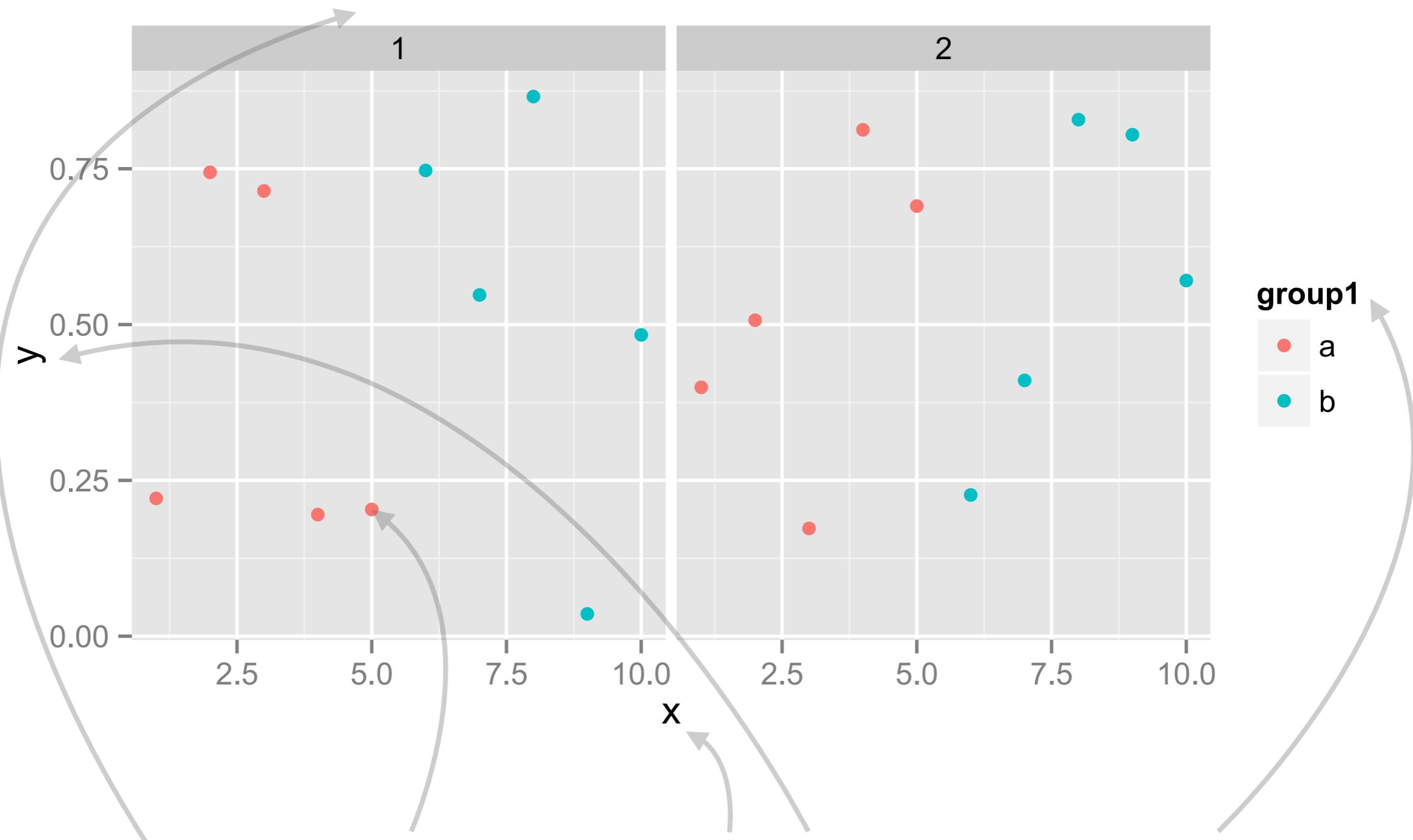
data + geometric representation +
aesthetics + layout ...

```
d <- data.frame(x = c(1:10, 1:10),
                 y = runif(20),
                 group1 = rep(gl(2, 5, labels = c("a", "b"))), 2),
                 group2 = gl(2, 10))
```



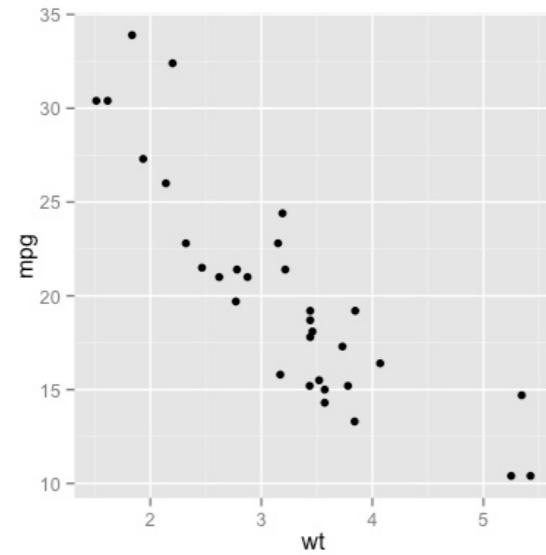
```
> head(d)
```

	x	y	group1	group2
1	0.5776995		a	1
2	0.5658393		a	1
3	0.3671755		a	1
4	0.5569217		a	1
5	0.8970902		a	1
6	0.2722306		b	1

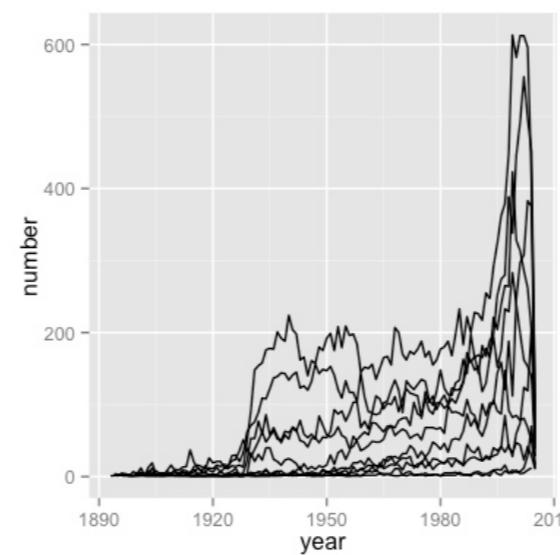


```
ggplot(d) + geom_point(aes(x, y, colour = group1)) +  
  facet_grid(~group2)
```

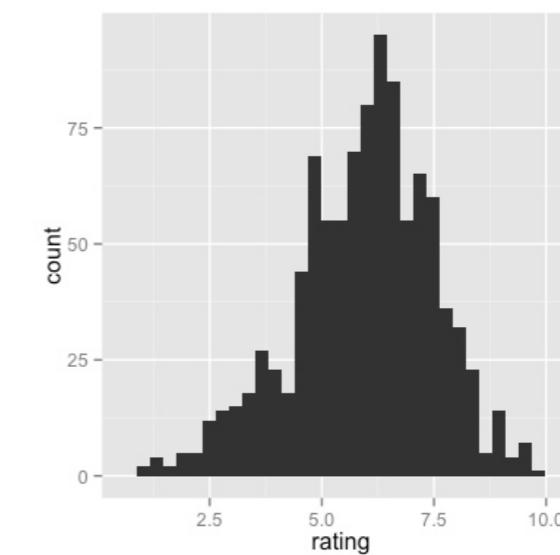
Geoms



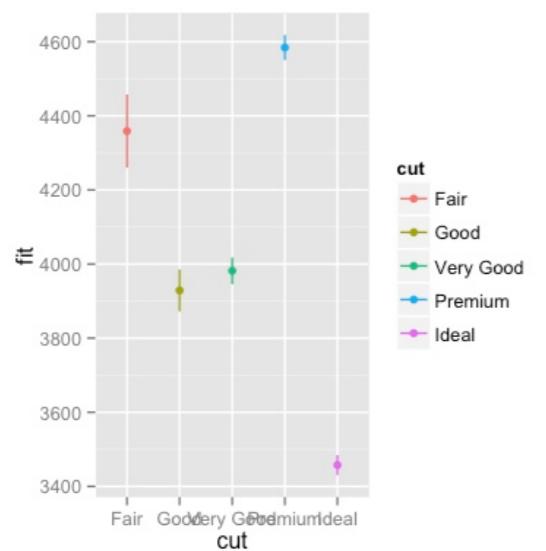
geom_point



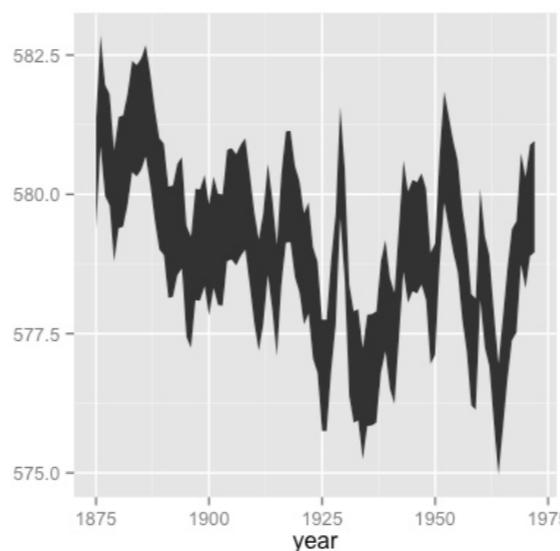
geom_line



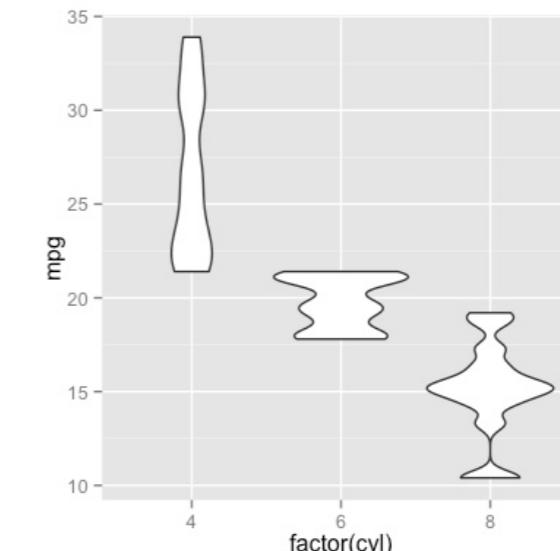
geom_histogram



geom_pointrange



geom_ribbon



geom_violin

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](#)

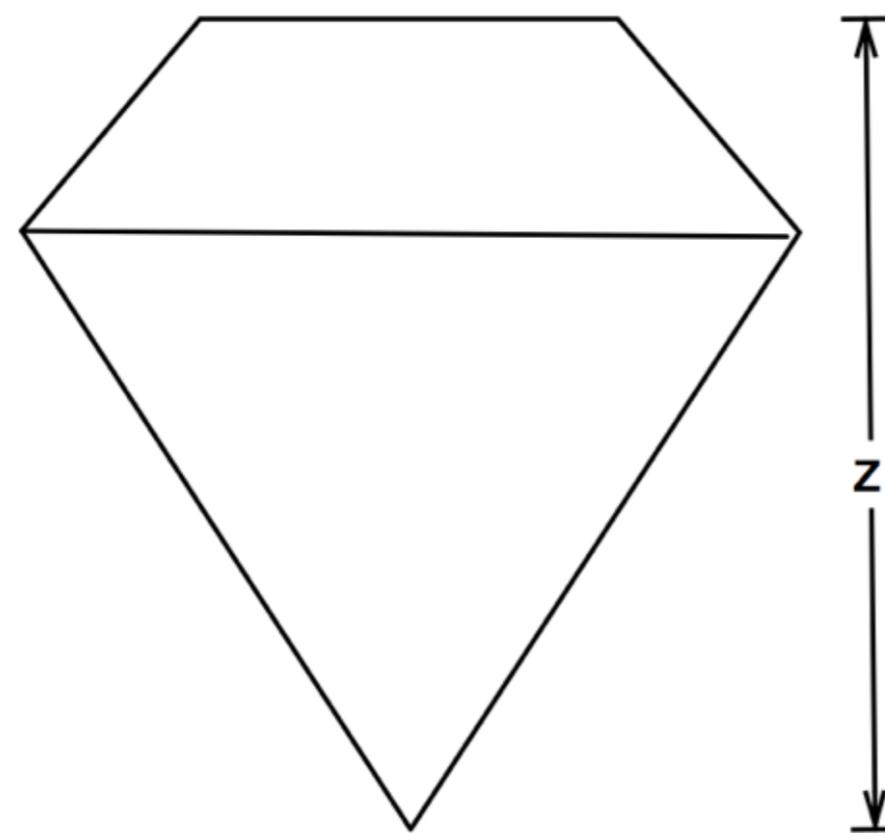


```
> head(diamonds)
```

	carat	cut	color	clarity	depth	table	price	x	y	z
1	0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
2	0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
3	0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
4	0.29	Premium	I	VS2	62.4	58	334	4.20	4.23	2.63
5	0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75
6	0.24	Very Good	J	VVS2	62.8	57	336	3.94	3.96	2.48

\leftarrow x \rightarrow

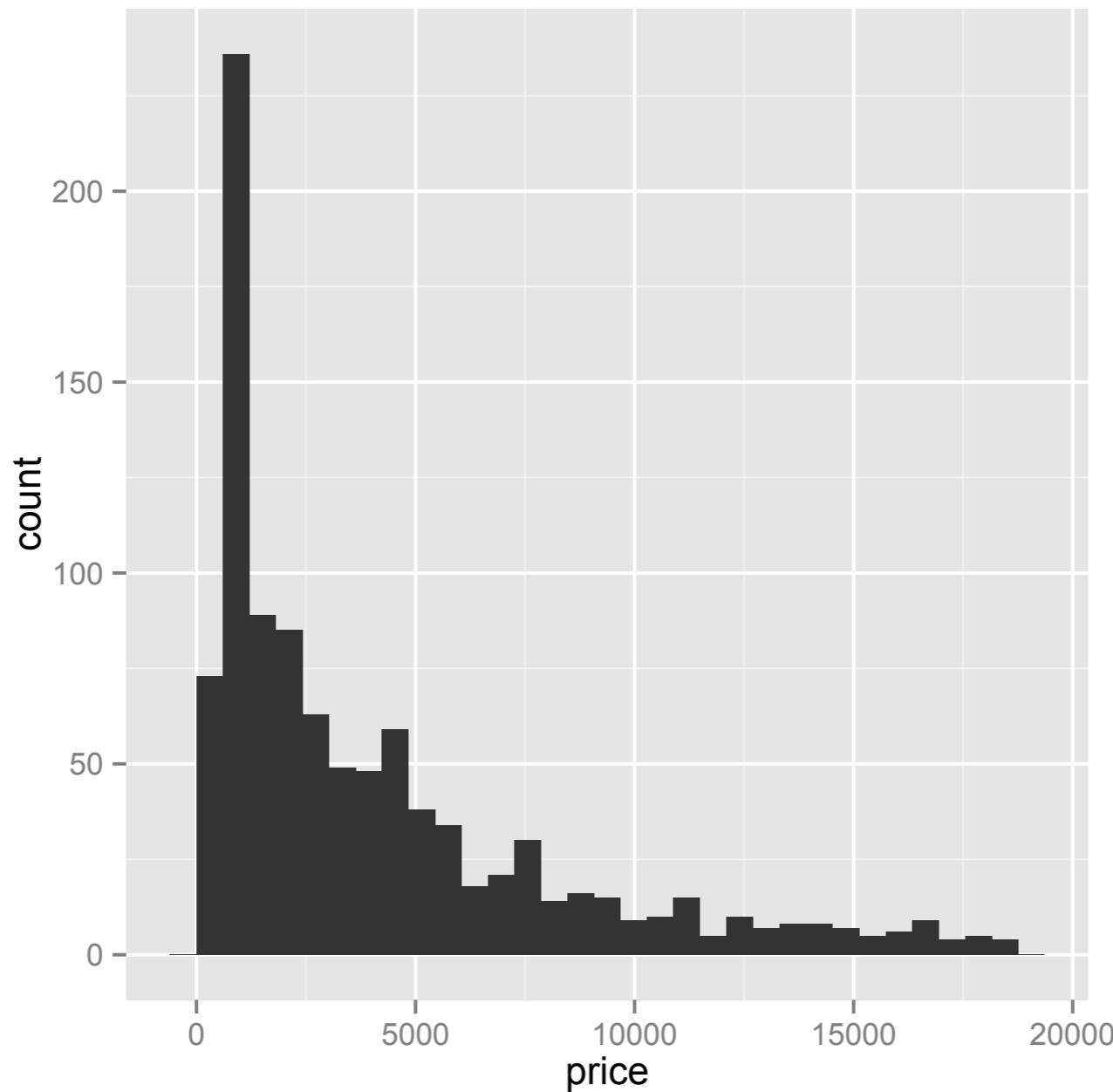
\leftarrow table width \rightarrow



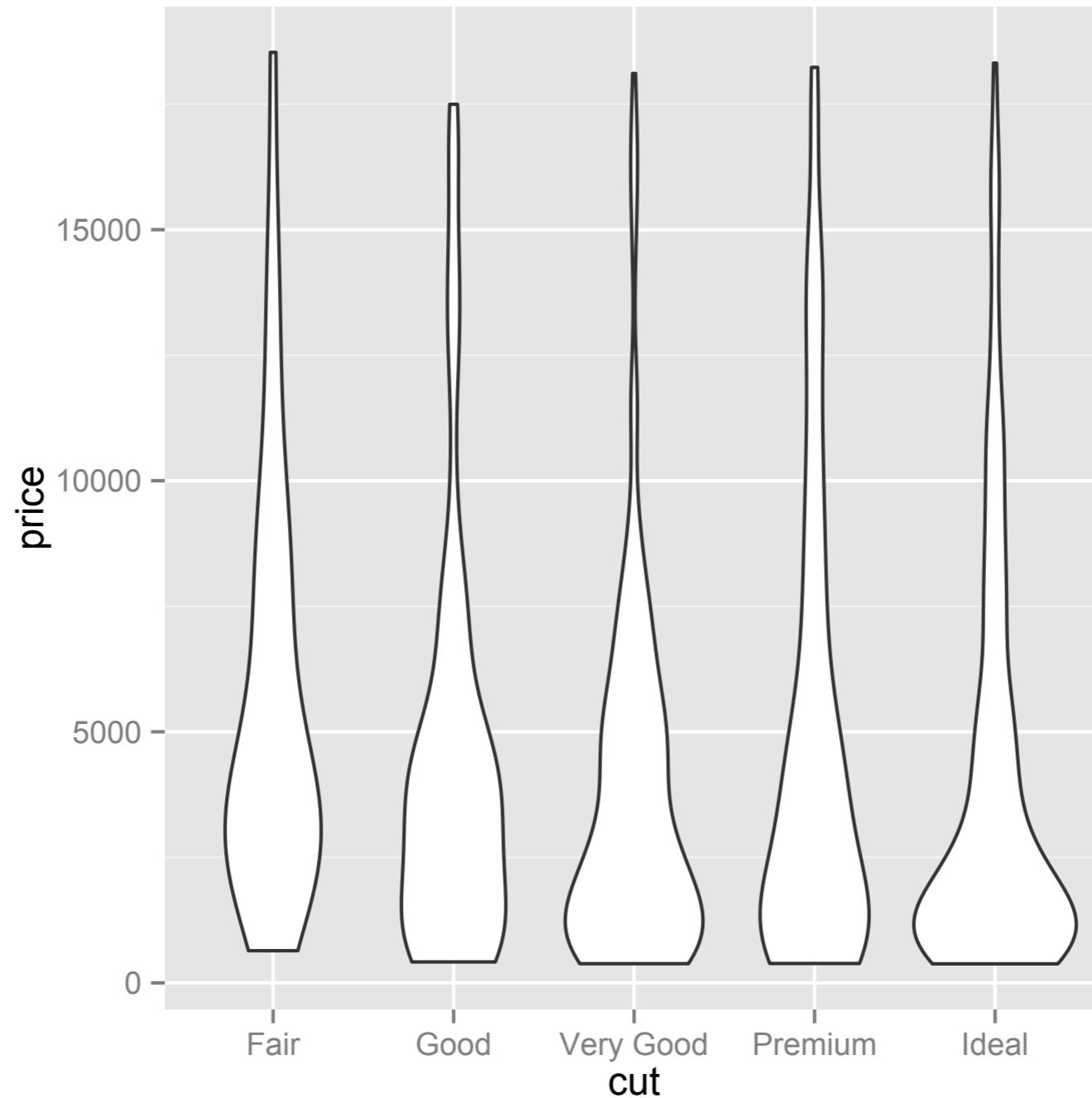
depth = $z / \text{diameter}$
table = table width / $x * 100$

By Hadley Wickham

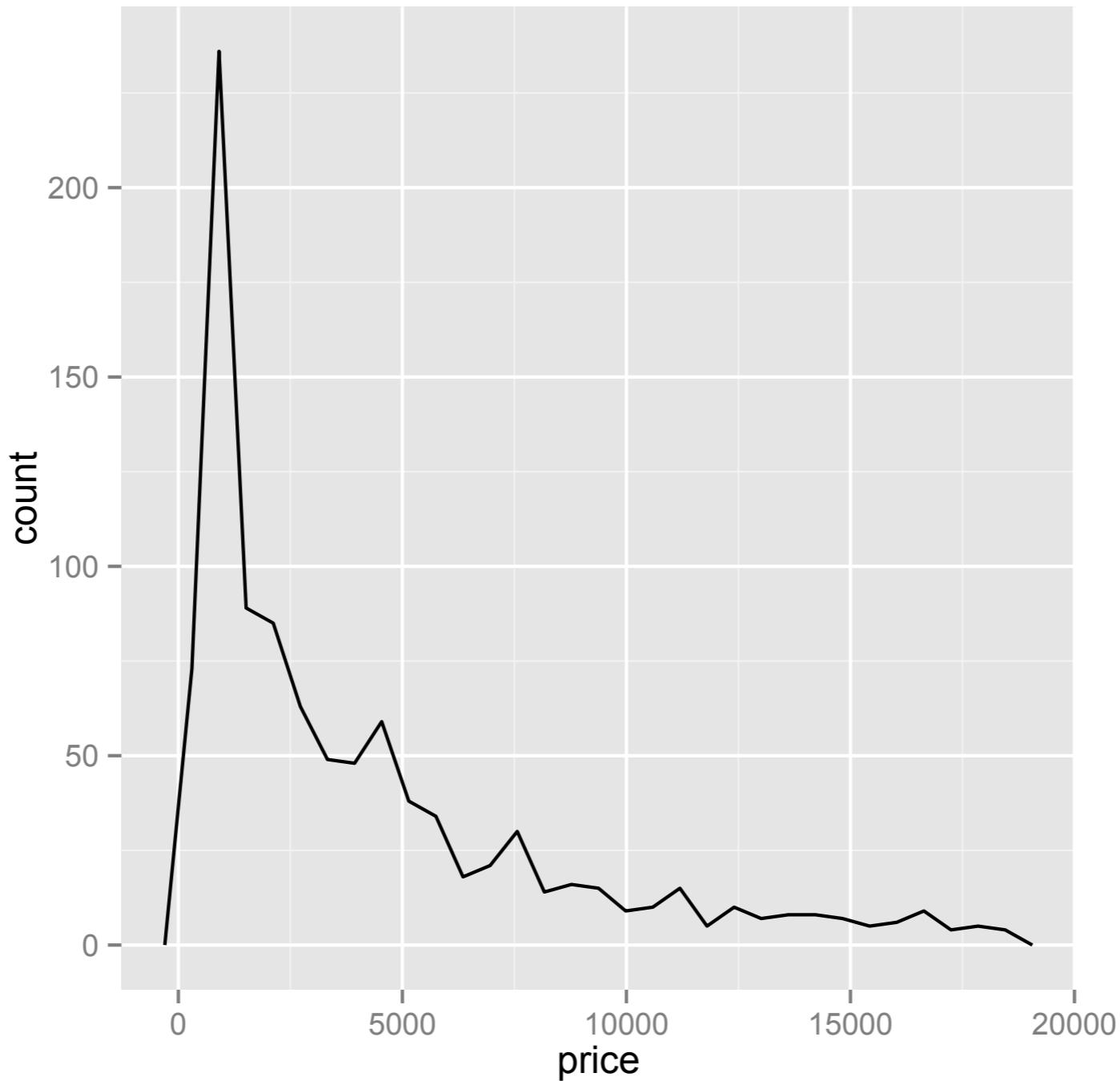
Explore the diamonds
dataset with at least 3
different geoms.



```
ggplot(diamonds2, aes(x = price)) +  
  geom_histogram()
```

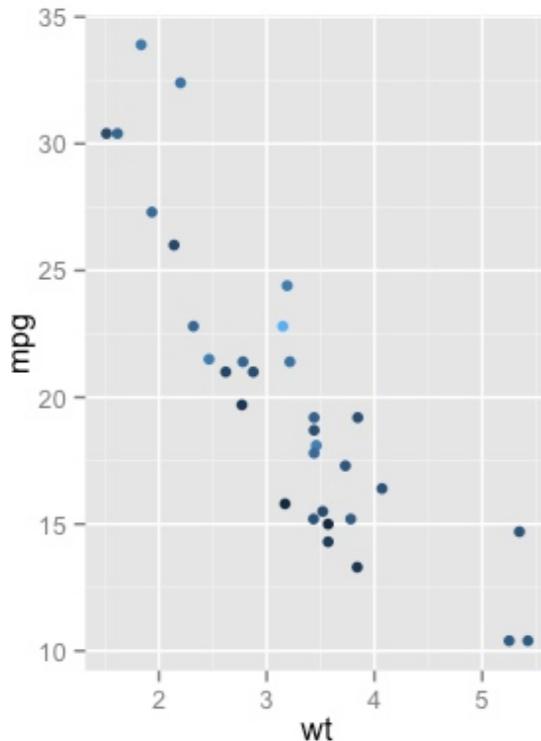


```
ggplot(diamonds2, aes(x = cut, y = price))  
+ geom_violin()
```

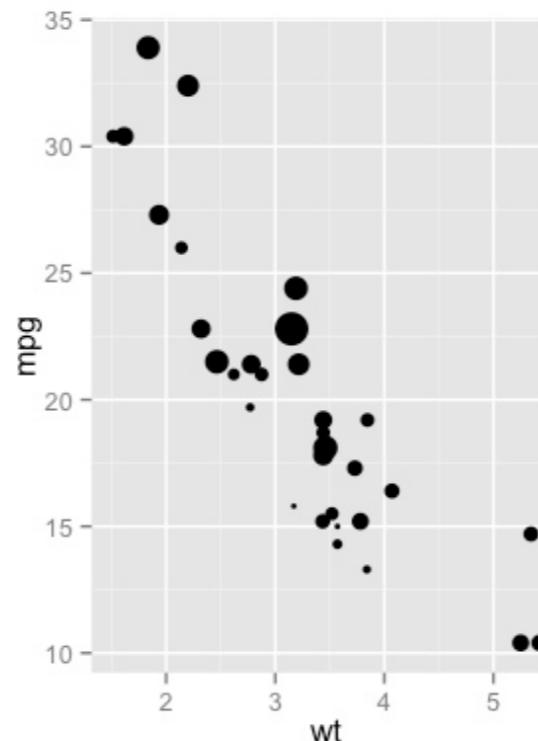


```
ggplot(diamonds2, aes(x = price)) +  
  geom_freqpoly()
```

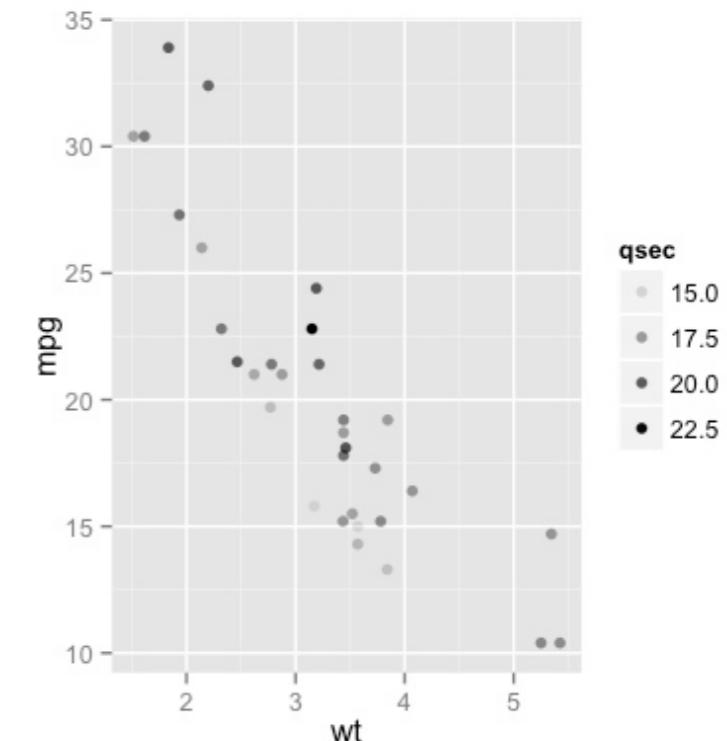
Aesthetics



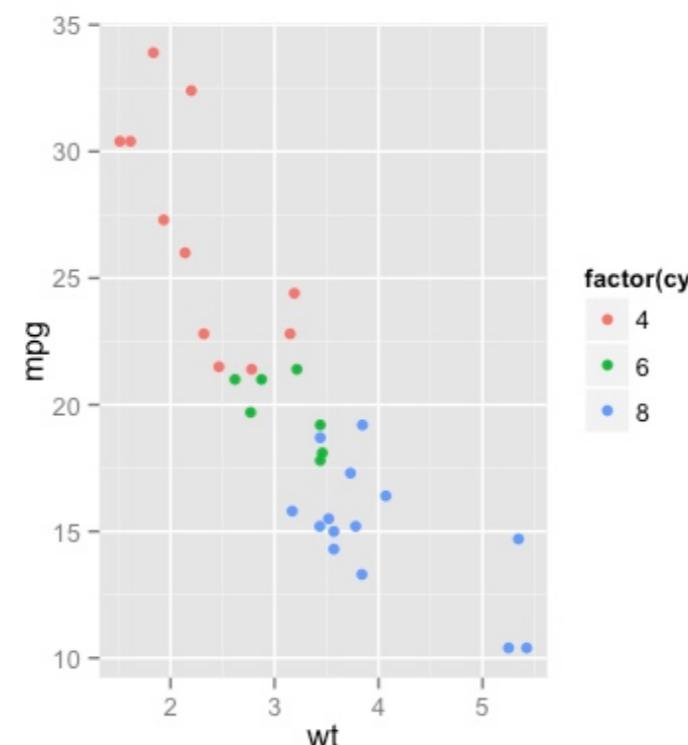
colour



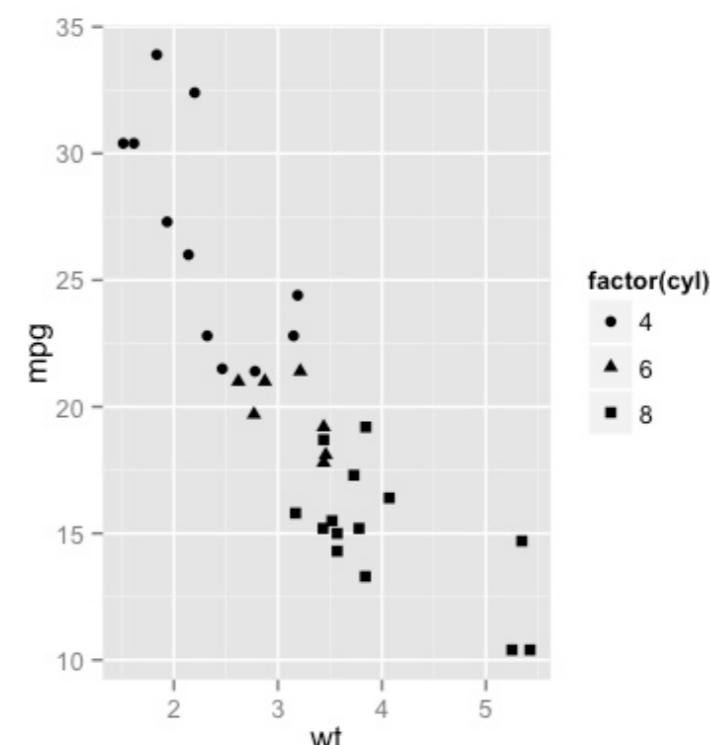
size



alpha

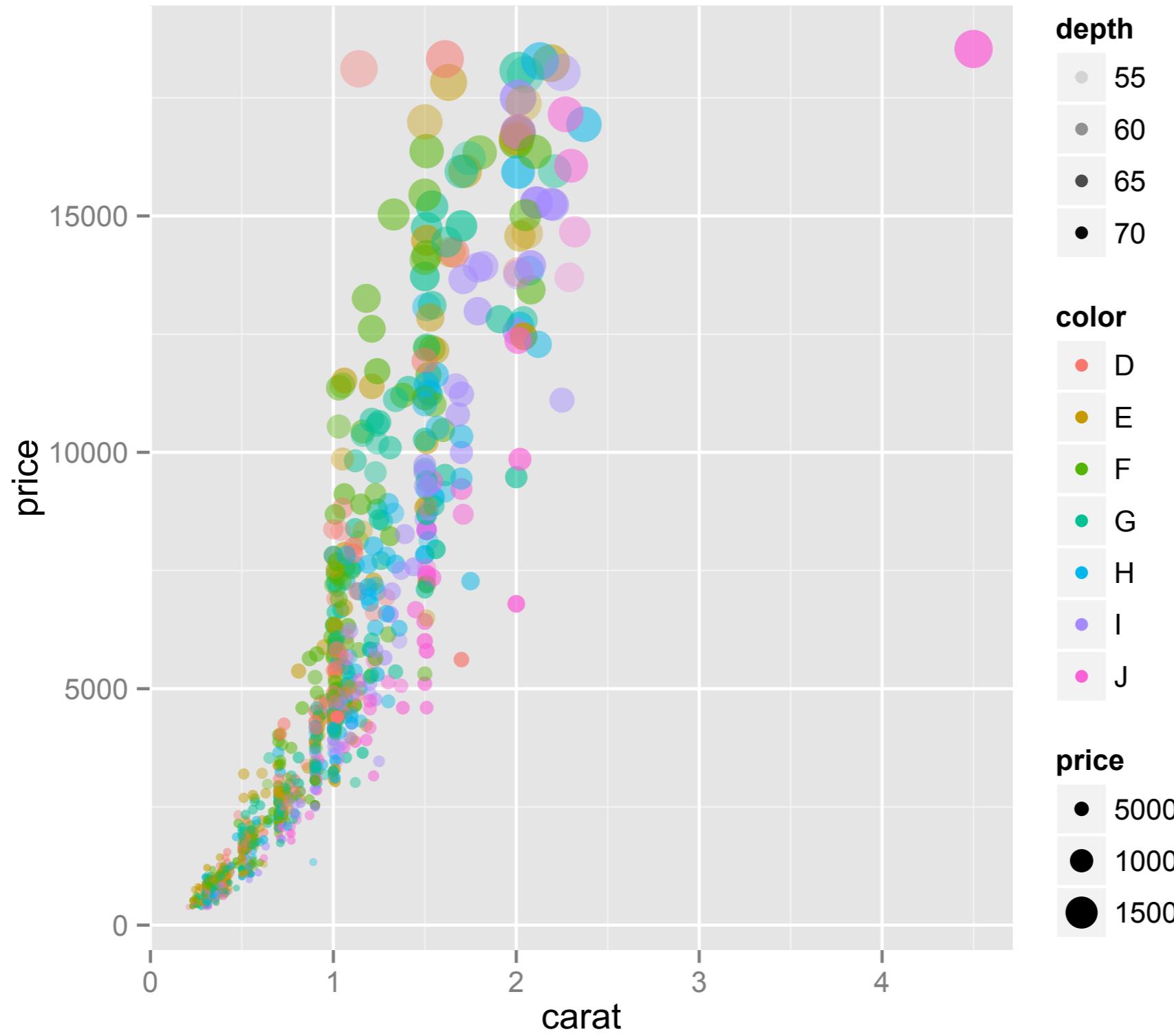


colour

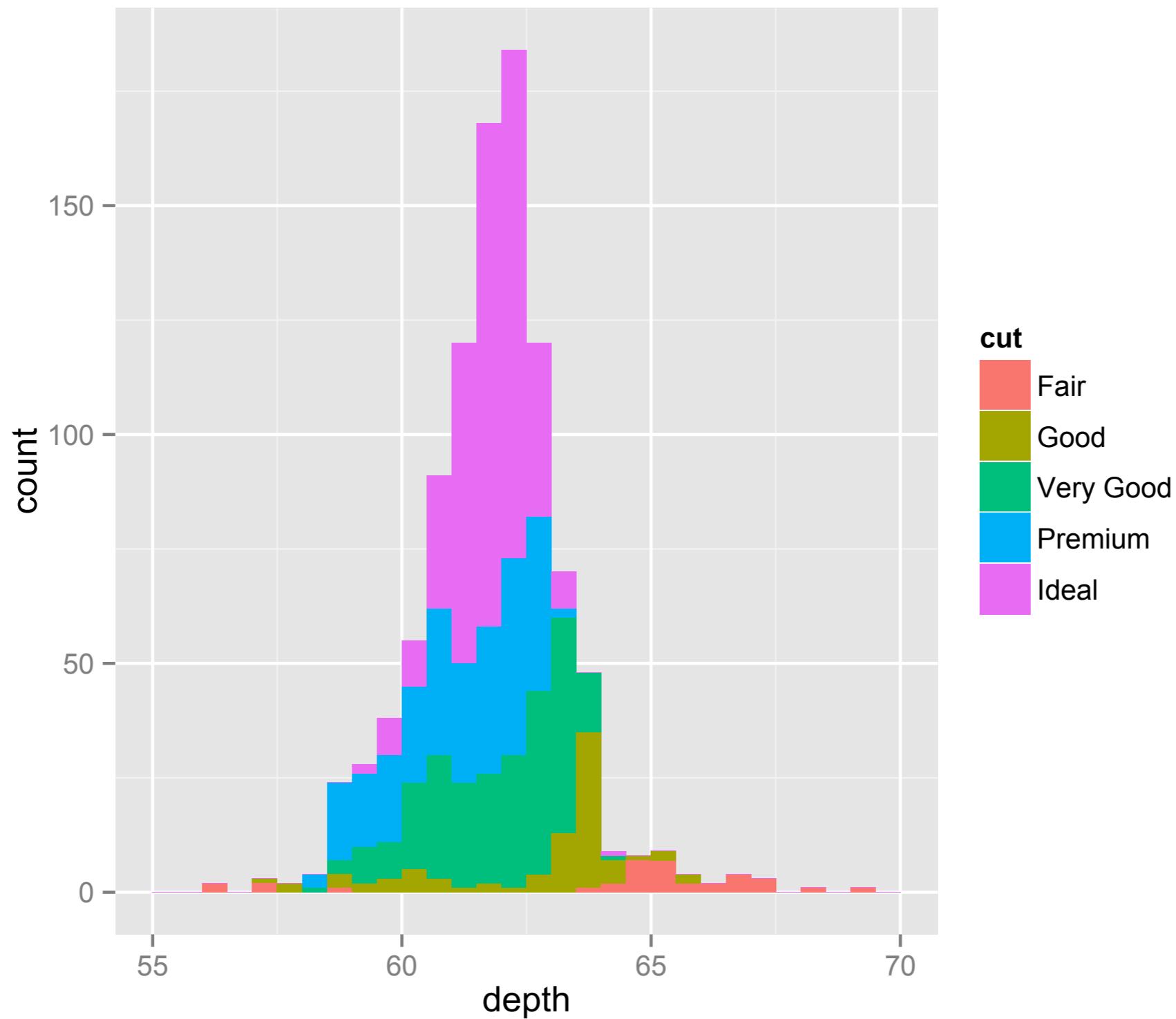


shape

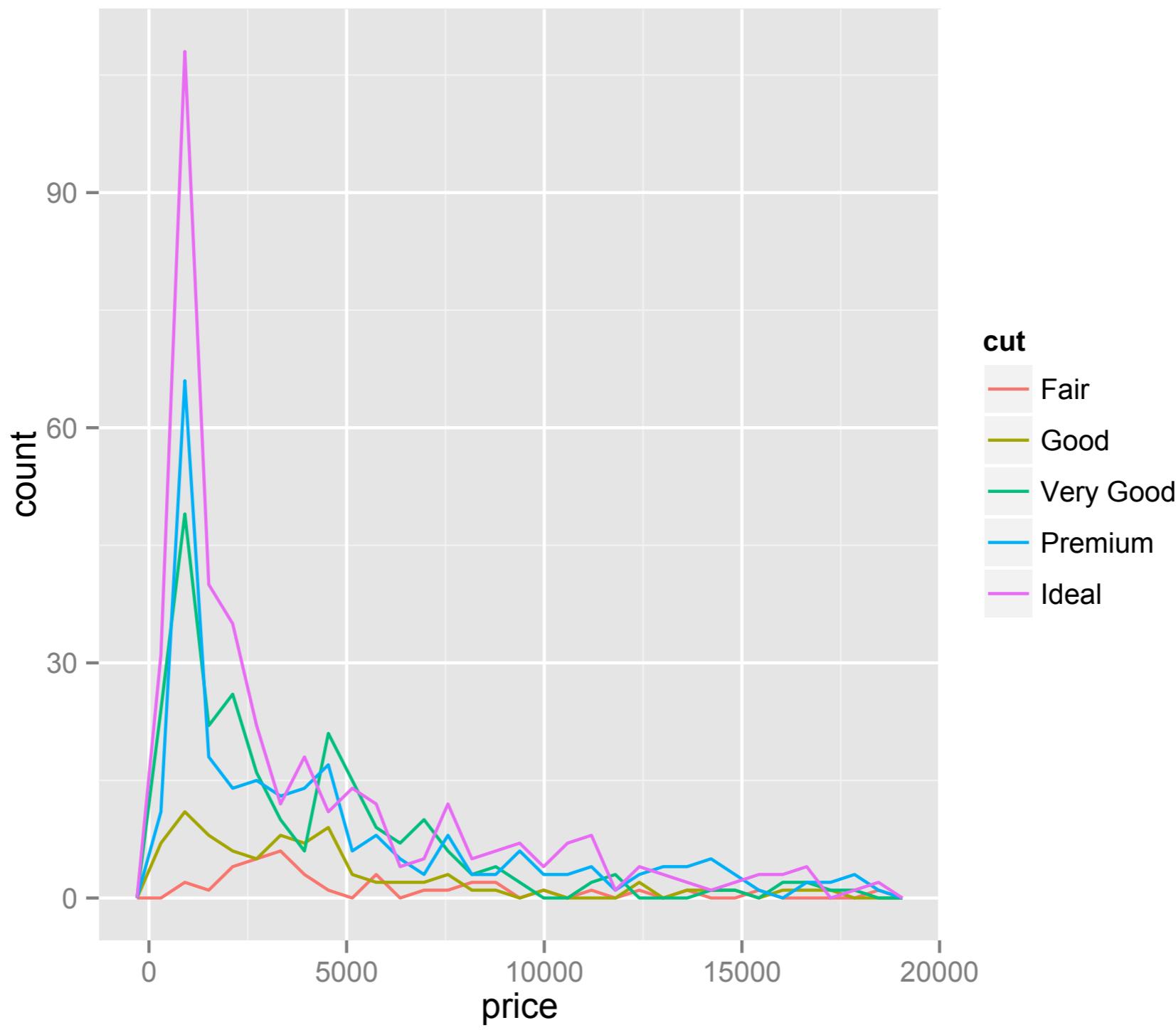
See how many dimensions of
the data you can display
simultaneously with aesthetics.



```
ggplot(diamonds2, aes(carat, price, colour = color, size =  
price, alpha = depth)) + geom_point()
```

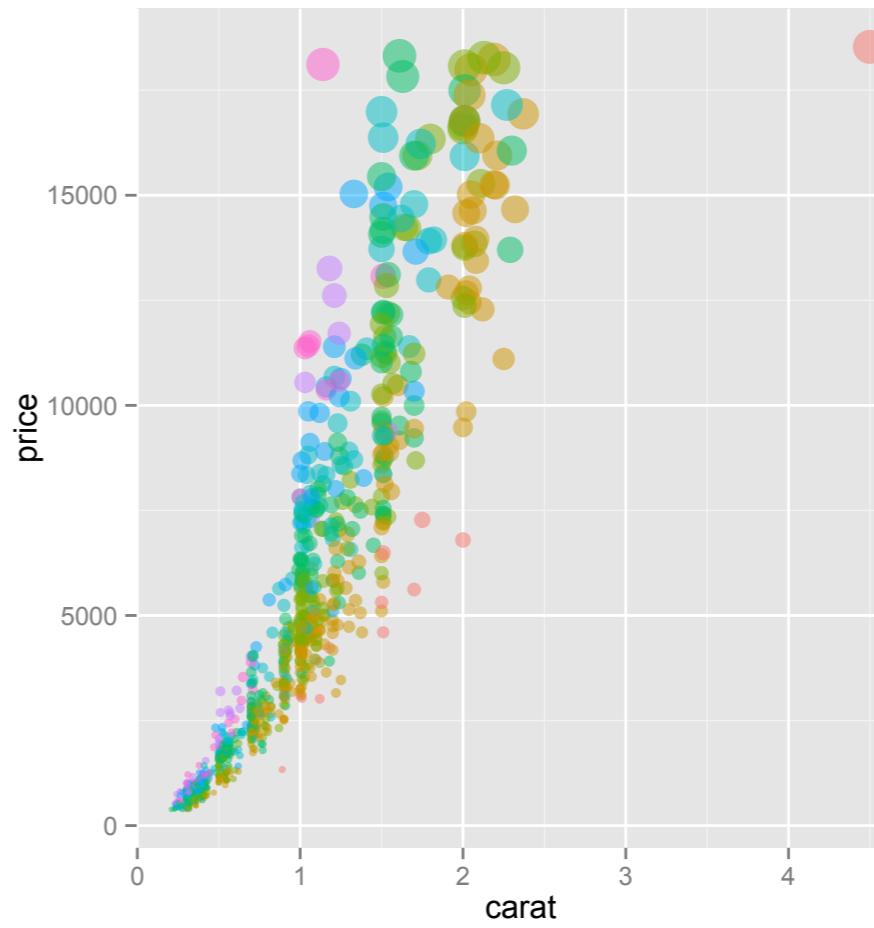
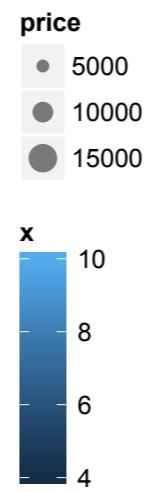
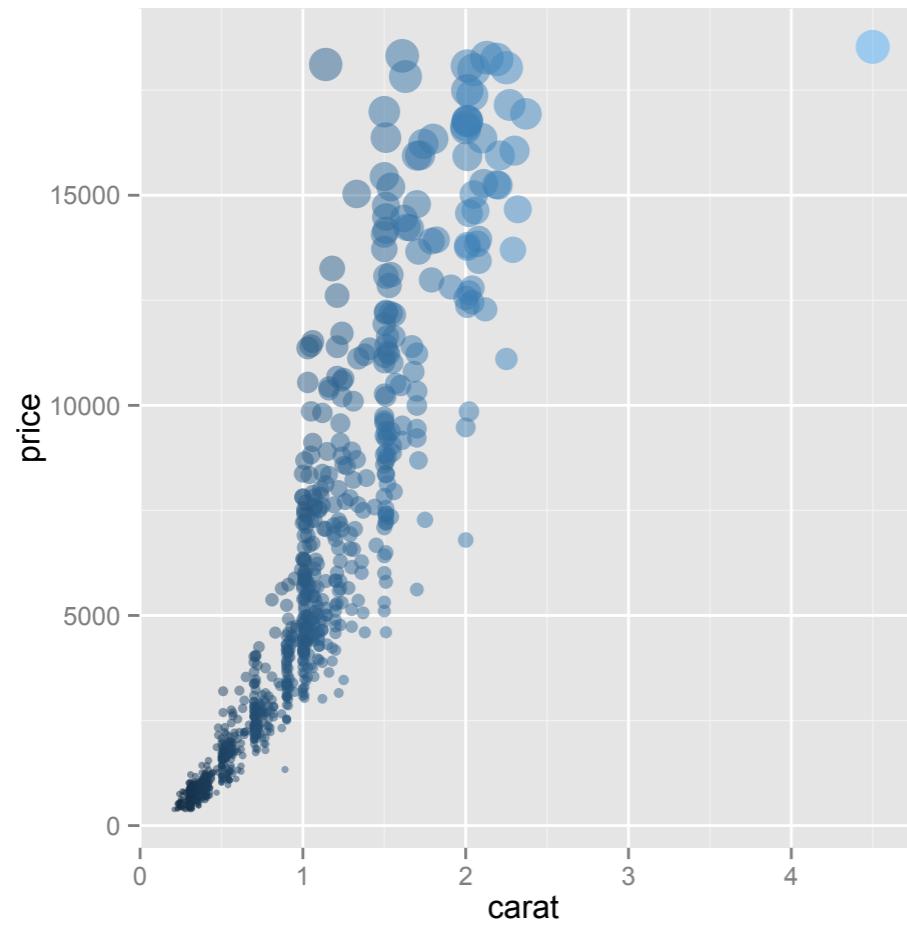


```
ggplot(diamonds2, aes(x = depth)) +  
  geom_histogram(aes(fill = cut)) + xlim(55, 70)
```



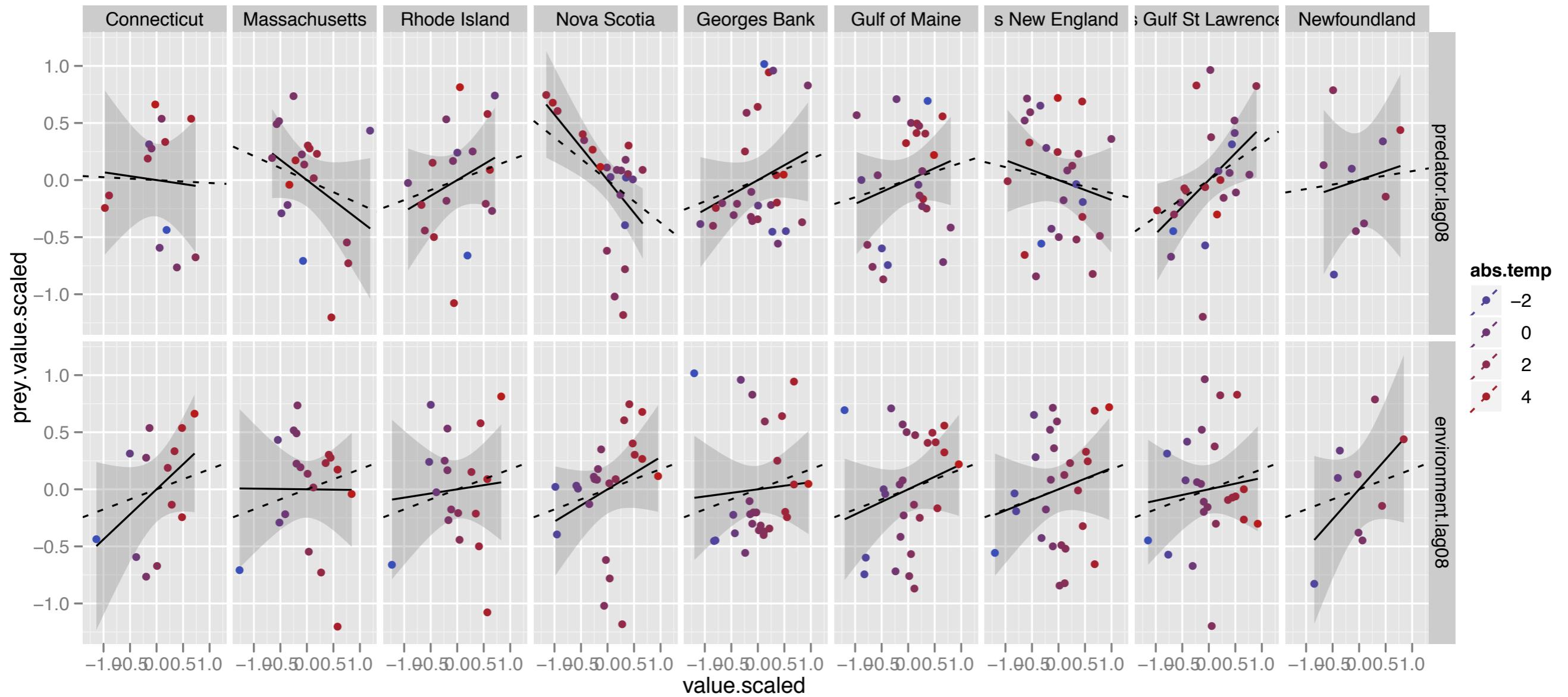
```
ggplot(diamonds2) + geom_freqpoly(aes(price, colour = cut))
```

What happens when you specify
a continuous vs. a factor variable
for an aesthetic like colour?



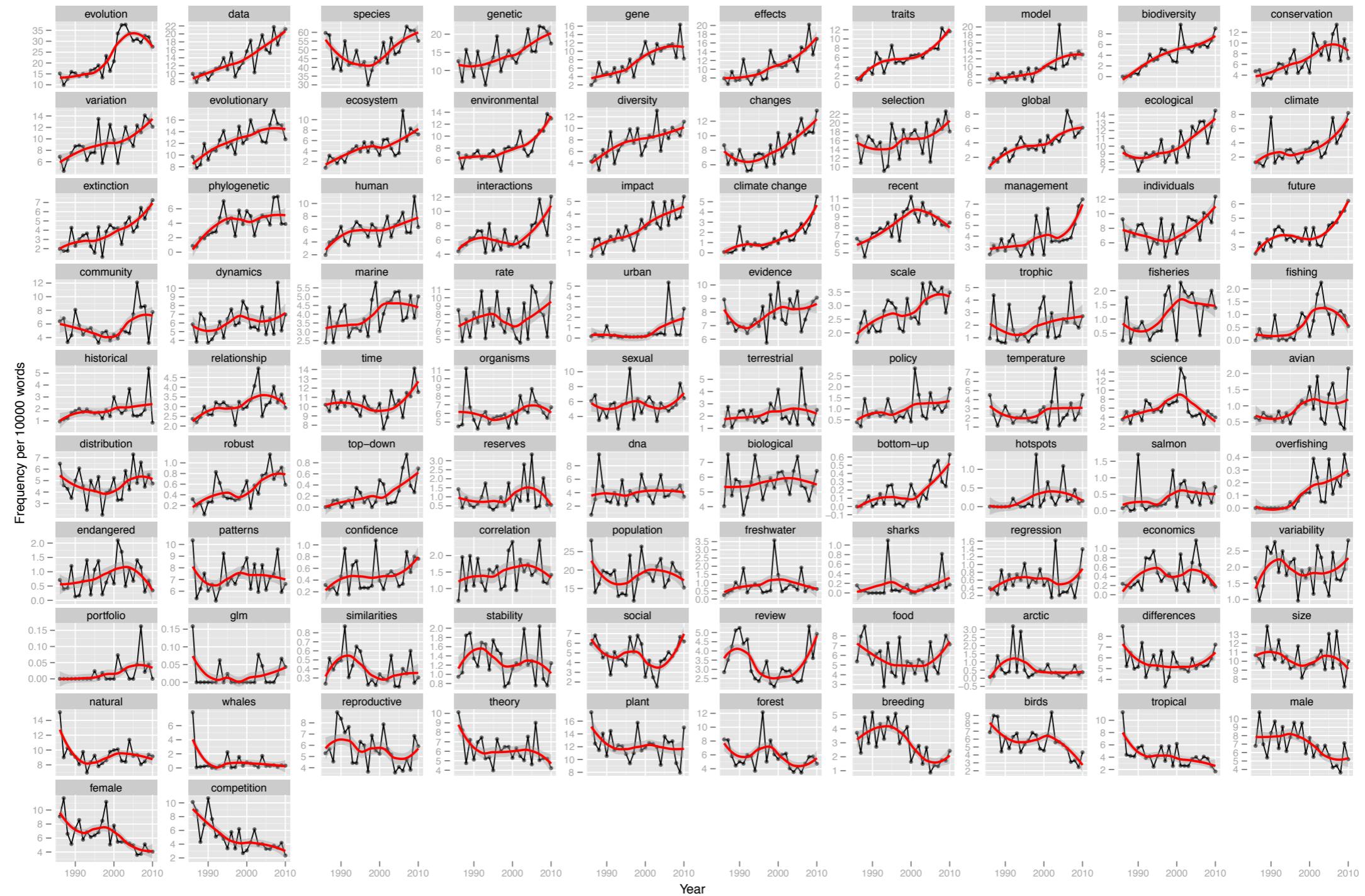
```
ggplot(diamonds2, aes(carat, price, colour = clarity, size = price)) + geom_point(alpha = 0.5)
```

```
ggplot(diamonds2, aes(carat, price, colour = x, size = price)) +  
  geom_point(alpha = 0.5)
```



facet_grid

```
ggplot(lmer.j6, aes(value.scaled, prey.value.scaled, col = abs.temp)) +
  facet_grid(variable ~ region.ordered.by.mean.temp) + stat_smooth(method = "lm") + geom_point()
  + geom_abline(aes(intercept = lmer.inter, slope = lmer.lag08.slope), lty = 2)
```

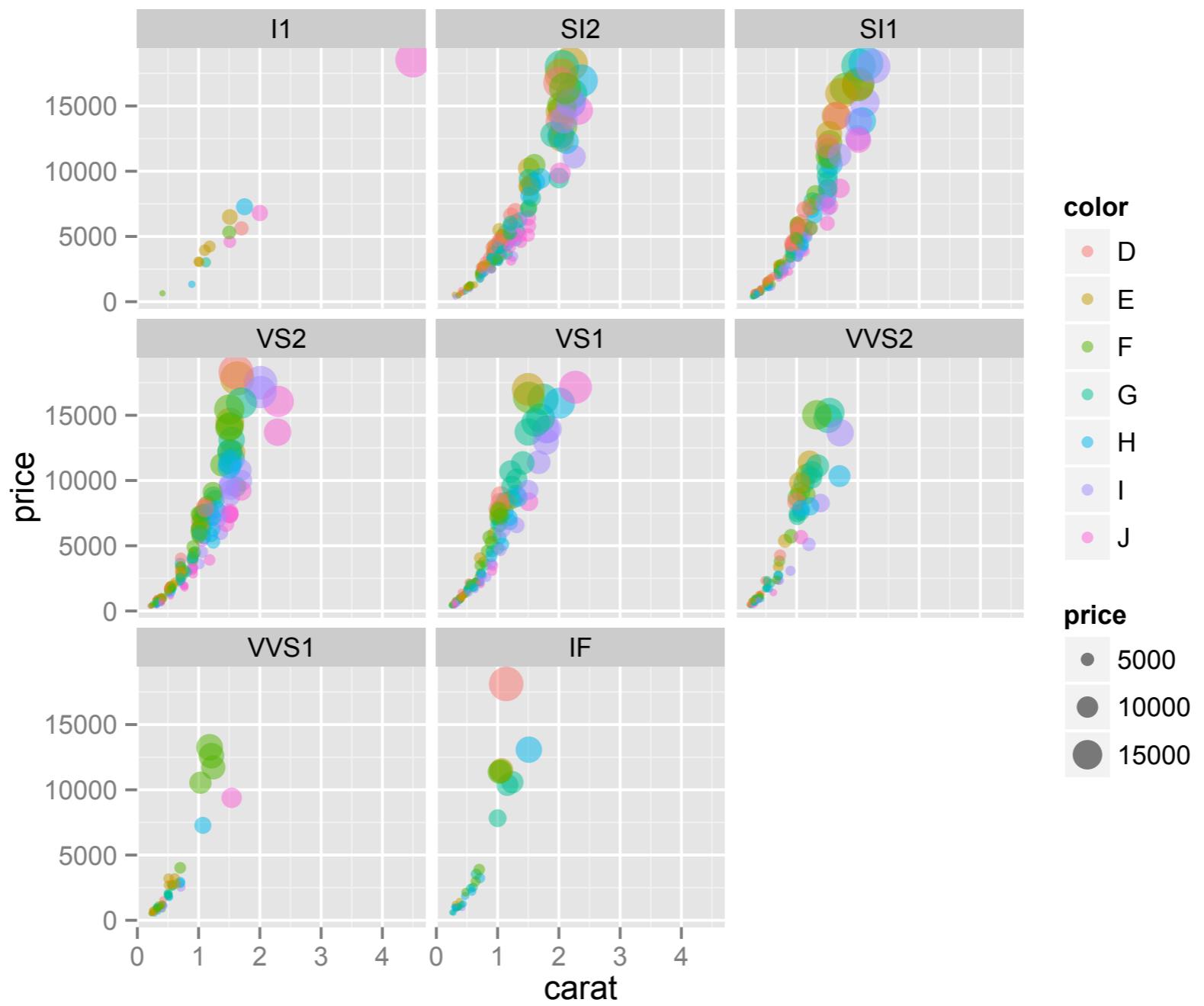


facet_wrap

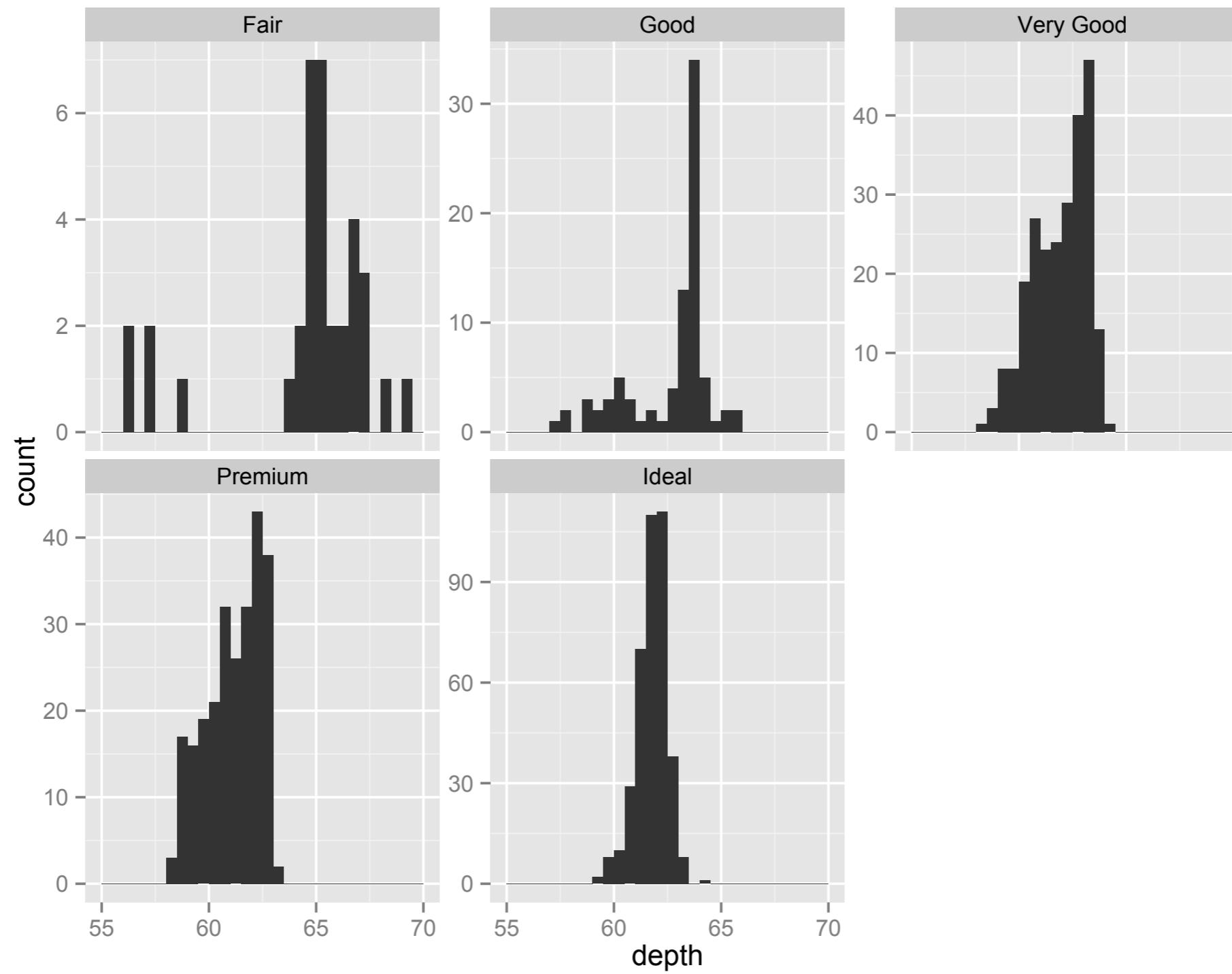
```
p <- ggplot(freq.melt), aes(year, frequency.adj)) + geom_line() + geom_point(col = "#00000090", cex = 1.75) + facet_wrap(~variable, scales = "free_y") + xlab("Year") + ylab("Frequency per 10000 words") + stat_smooth(method="loess", size=1, col = "red")
```

Try replacing one of the aesthetics with a `facet_wrap`.

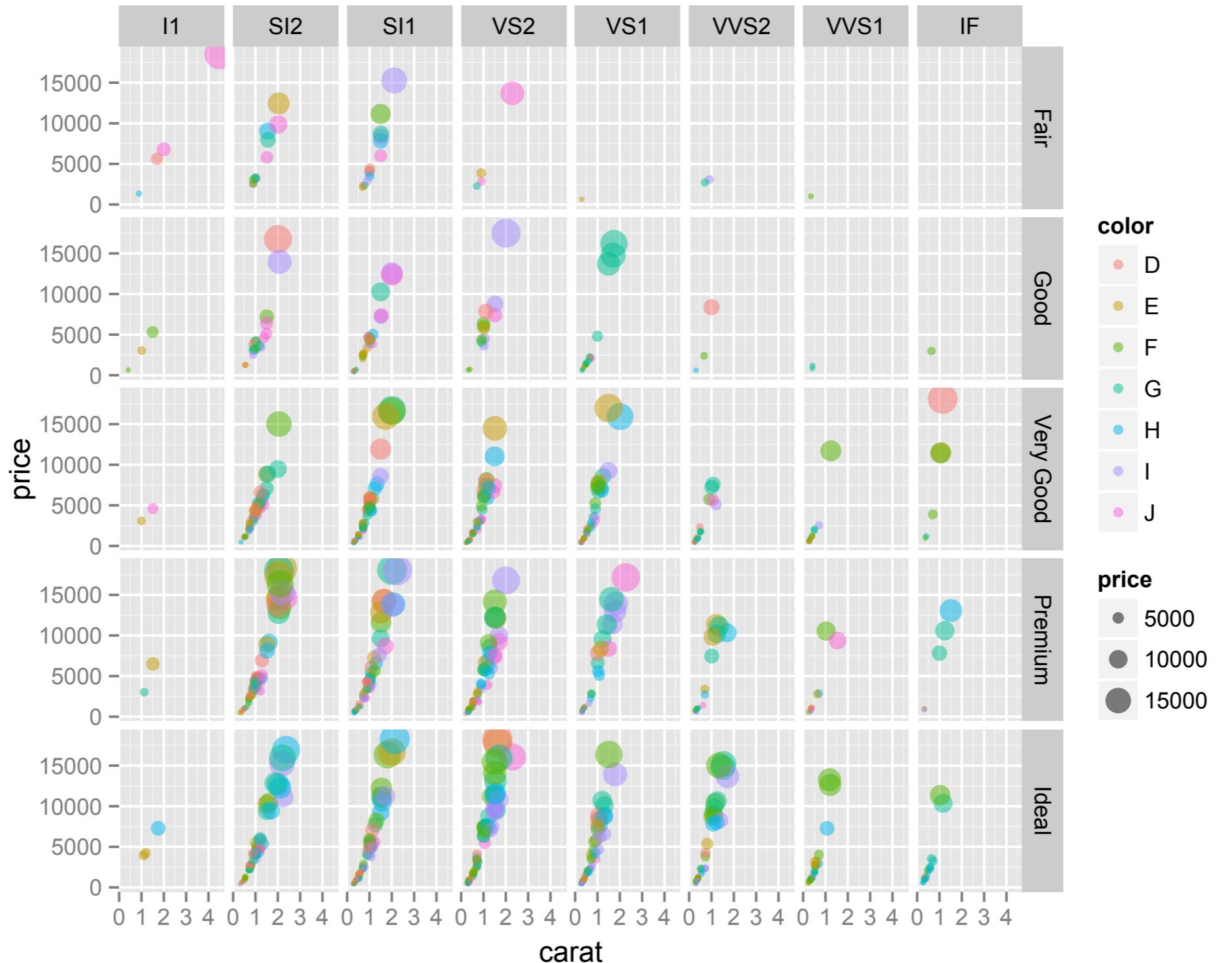
Now use `facet_grid` with the diamonds dataset.



```
ggplot(diamonds2, aes(carat, price, colour = color, size =  
price)) + geom_point(alpha = 0.5) + facet_wrap(~clarity)
```

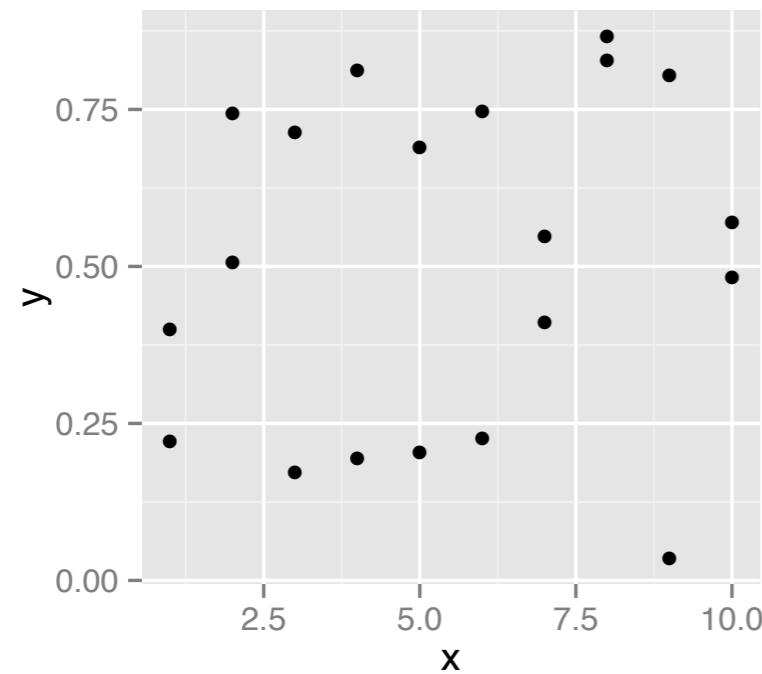


```
ggplot(diamonds2, aes(x = depth)) + geom_histogram() +  
  xlim(55, 70) + facet_wrap(~cut, scales = "free_y")
```

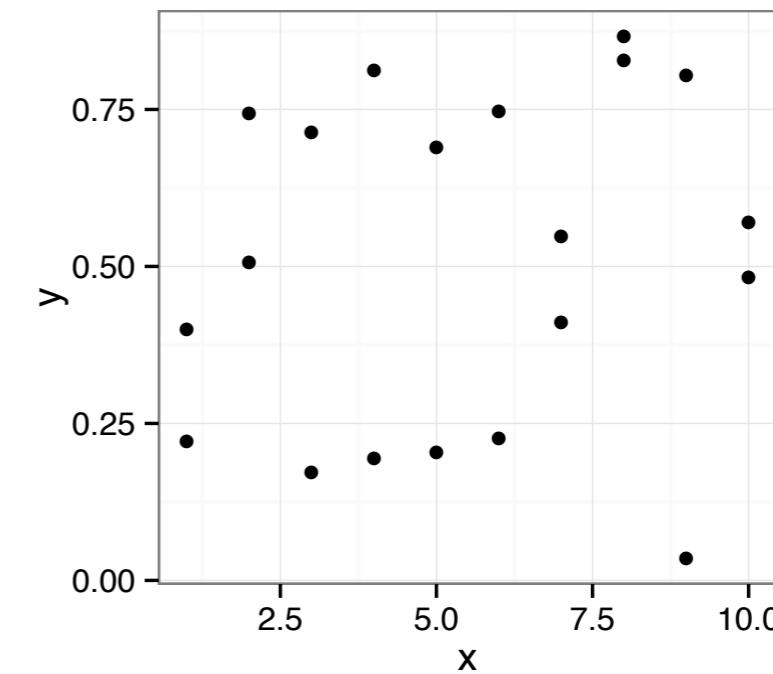


```
ggplot(diamonds2, aes(carat, price, colour = color, size = price)) + geom_point(alpha = 0.5) + facet_grid(cut ~ clarity)
```

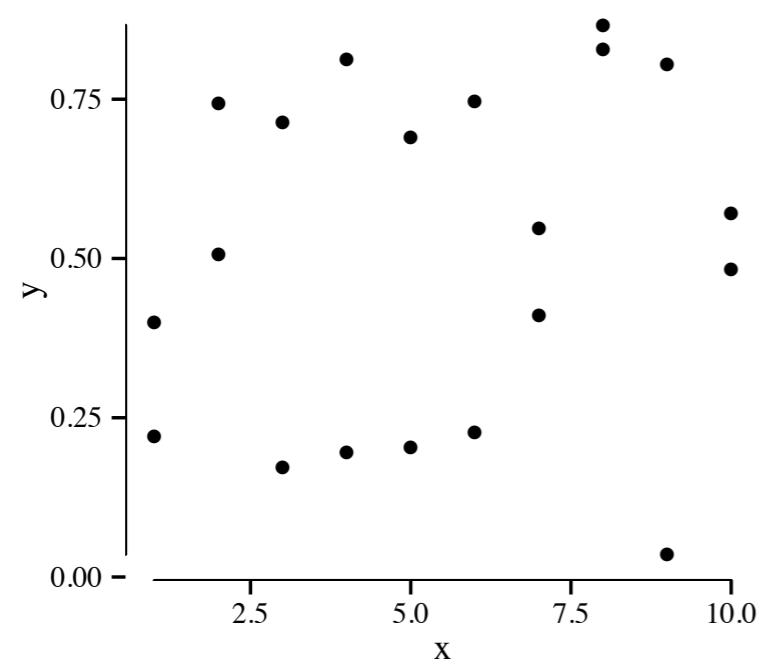
A taste of what
else is out there...



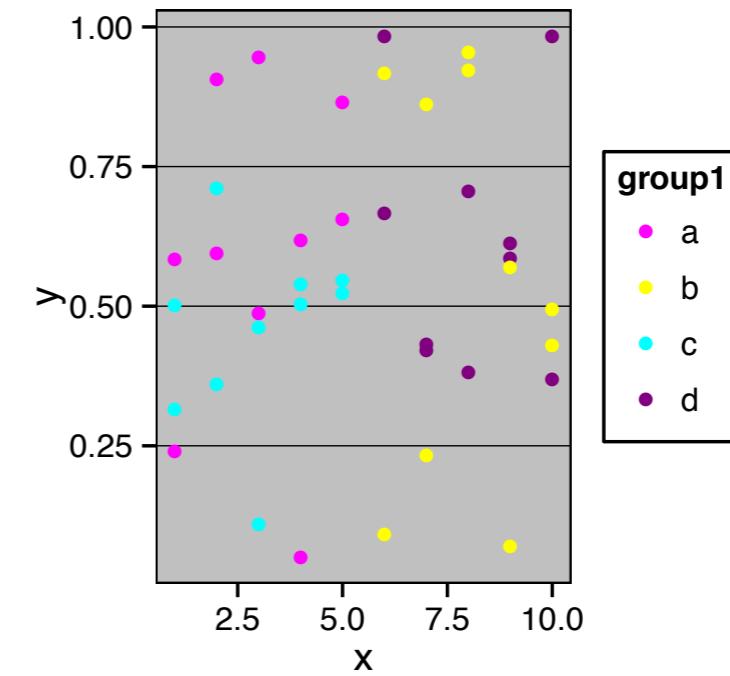
theme_grey



theme_bw

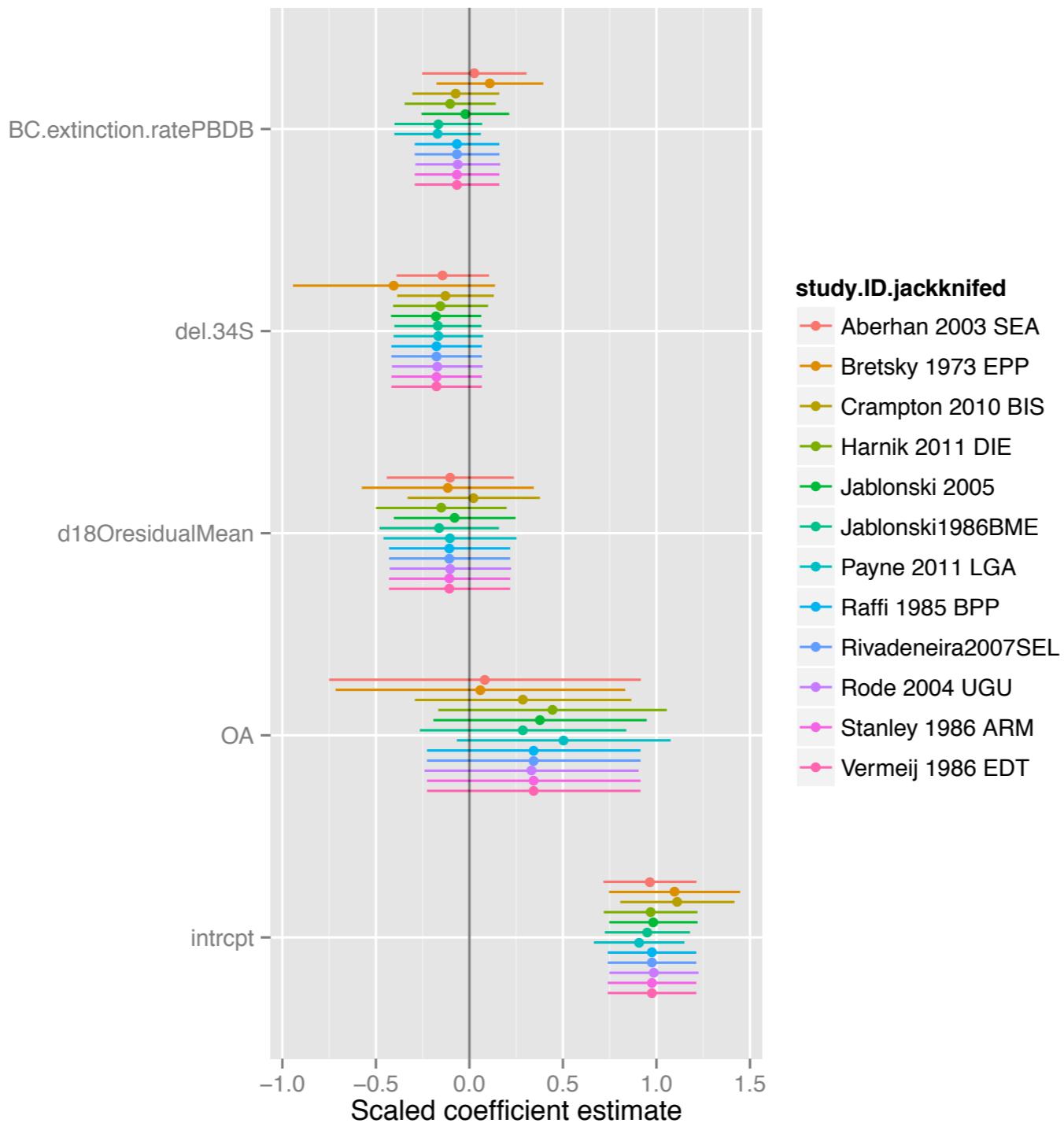


theme_tufte



theme_excel2003

```
p <- ggplot(...  
ggsave("filename.pdf")
```



```
ggplot(jackknifed_coefs) + geom_pointrange(aes(x = variable, y = coef, ymin = ci.lb,
ymax = ci.ub, colour = study.ID.jackknifed), position = position_dodge(height = 0, width
= -0.6)) + coord_flip() + xlab("") + ylab("Scaled coefficient estimate") +
geom_hline(yintercept = 0, lty = 1, col = "#00000080")
```

Getting help

<http://docs.ggplot2.org/>

ggplot2 book

ggplot2 Google Groups

stackoverflow.com

ggplot is easy

Data manipulation is hard

```
library(reshape)
library(plyr)
```

method1	method2	method3
-2.18	0.91	1.31
-1.04	-0.68	1.34
0.22	-1.87	2.24

```
> library(reshape)
> melt(d)
```

variable	value
method1	-2.18
method1	-1.04
method1	0.22
method2	0.91
method2	-0.68

... .

Why use ggplot2 for data exploration?

1. Fast iteration
2. (mostly) Intuitive syntax
3. (mostly) Sane defaults
4. Great online help