Data visualization with R

J. Alexander Branham

Fall 2016

Data visualization

Data visualization

- Plots are (usually) easier to interpret than tables and numbers

Data visualization

- Plots are (usually) easier to interpret than tables and numbers
- We'll use `ggplot2` to visualize data in R. Make sure you have it installed (only the first time) and loaded (every time):

Data visualization

- Plots are (usually) easier to interpret than tables and numbers
- We'll use `ggplot2` to visualize data in R. Make sure you have it installed (only the first time) and loaded (every time):

Data visualization

- Plots are (usually) easier to interpret than tables and numbers
- We'll use `ggplot2` to visualize data in R. Make sure you have it installed (only the first time) and loaded (every time):

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

We'll be working with the `diamonds` dataset, provided by `ggplot2`. Check out its documentation with `?diamonds`:

1. How many diamonds are in the dataset?

We'll be working with the `diamonds` dataset, provided by `ggplot2`. Check out its documentation with `?diamonds`:

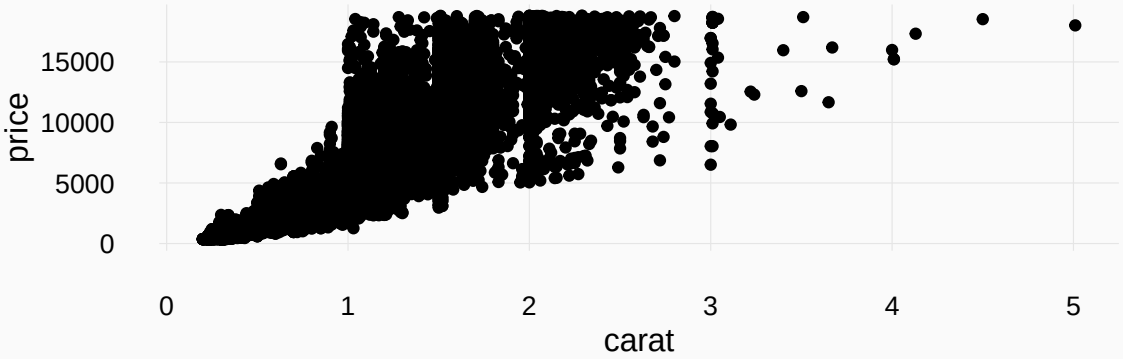
1. How many diamonds are in the dataset?
2. What is the name of the variable for the weight of the diamond?

We'll be working with the `diamonds` dataset, provided by `ggplot2`. Check out its documentation with `?diamonds`:

1. How many diamonds are in the dataset?
2. What is the name of the variable for the weight of the diamond?
3. What is the code for the worst color of diamond?

Scatterplots

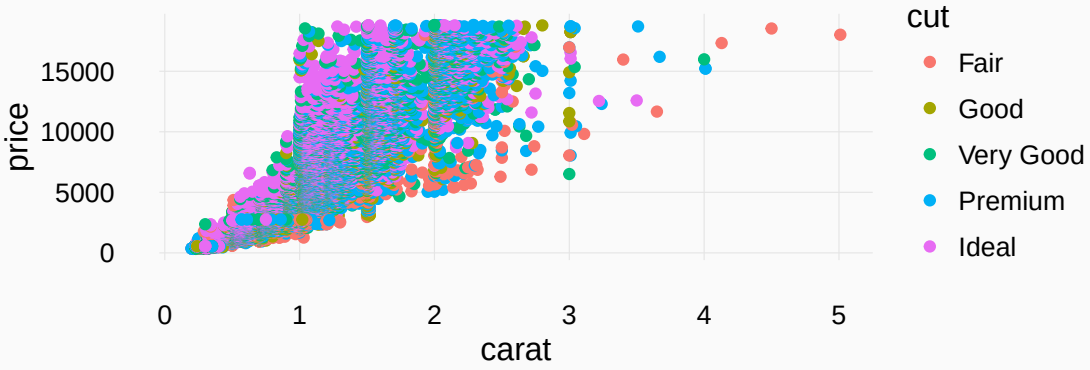
```
ggplot(data = diamonds) +
  geom_point(mapping = aes(x = carat, y = price))
```



Aesthetics

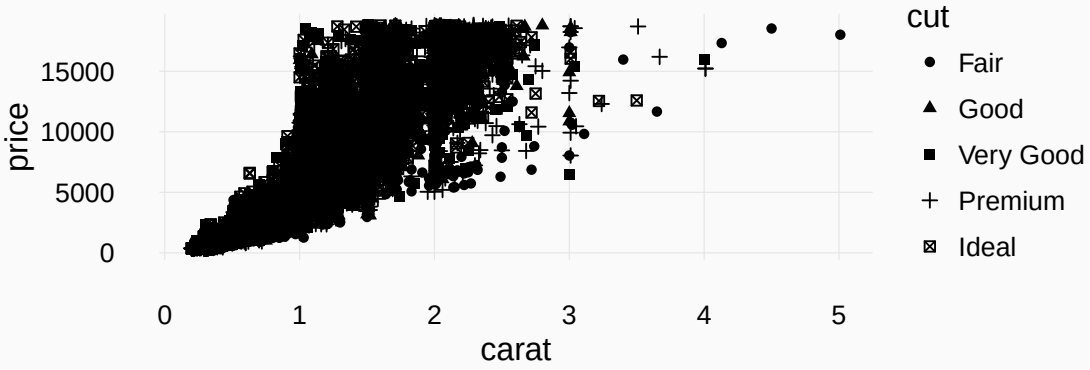
Color

```
ggplot(data = diamonds) +
  geom_point(mapping = aes(x = carat, y = price, color = cut))
```



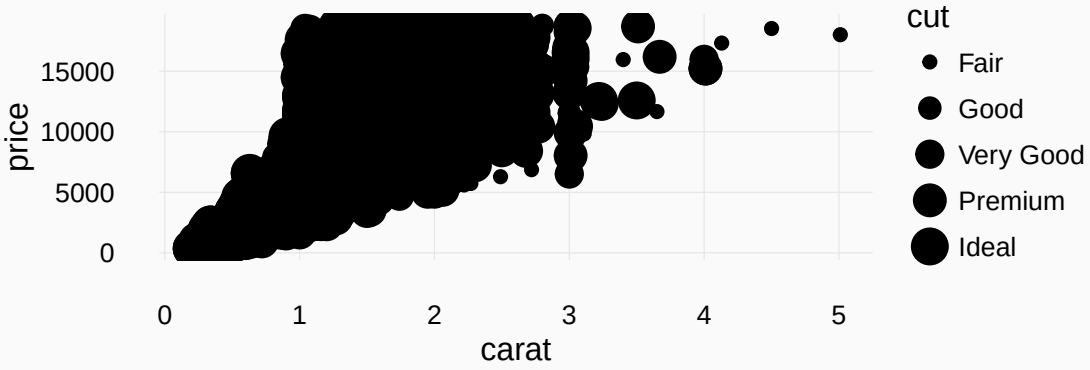
Shape

```
ggplot(data = diamonds) +
  geom_point(mapping = aes(x = carat, y = price, shape = cut))
```



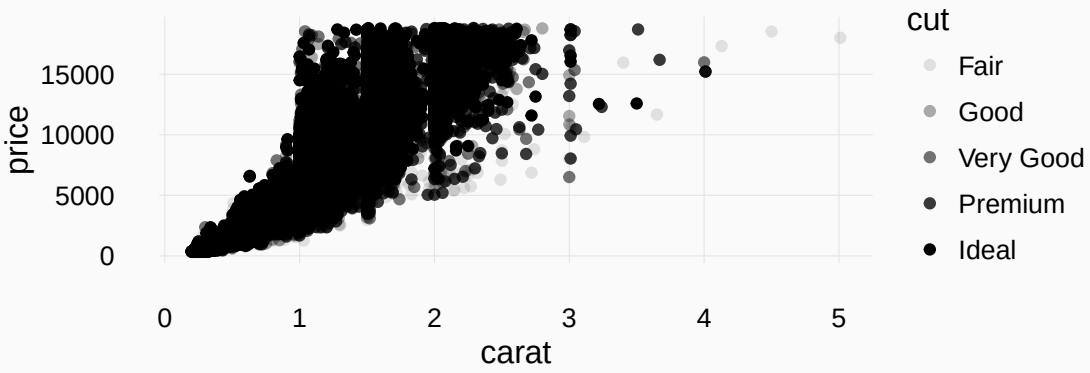
Size

```
ggplot(data = diamonds) +
  geom_point(mapping = aes(x = carat, y = price, size = cut))
```



Alpha

```
ggplot(data = diamonds) +
  geom_point(mapping = aes(x = carat, y = price, alpha= cut))
```



You try!

Use the `mpg` dataset from `ggplot2` to:

1. Make a scatterplot with displ on the x-axis and hwy on the y-axis

You try!

Use the `mpg` dataset from `ggplot2` to:

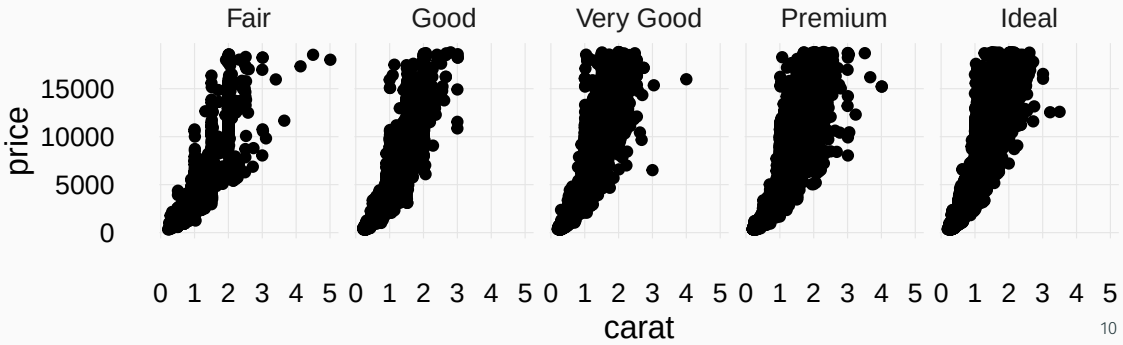
1. Make a scatterplot with displ on the x-axis and hwy on the y-axis
2. Add color, size, and shape aesthetics

Facets

Facets

Can make facets by adding `facet_grid`:

```
ggplot(data = diamonds) +
  geom_point(mapping = aes(x = carat, y = price)) + facet_grid(. ~ cut)
```



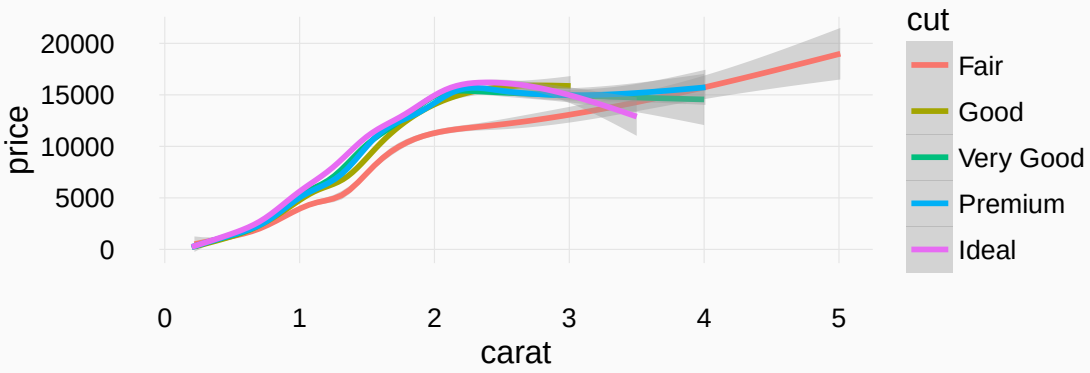
Geoms

Geoms

There are other kinds of graphs than scatterplots. `geom_` takes care of this for ggplot2 (“geometric object”):

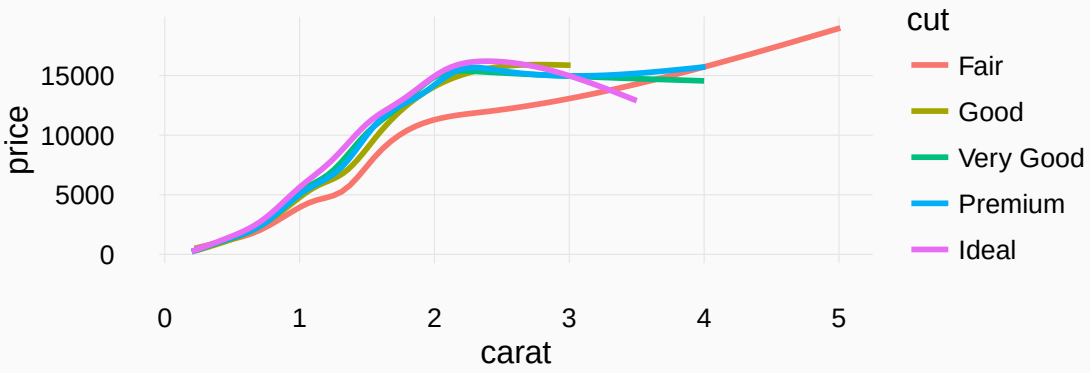
Smooth

```
ggplot(data = diamonds, mapping = aes(x = carat, y = price, color = cut))
  geom_smooth()
```



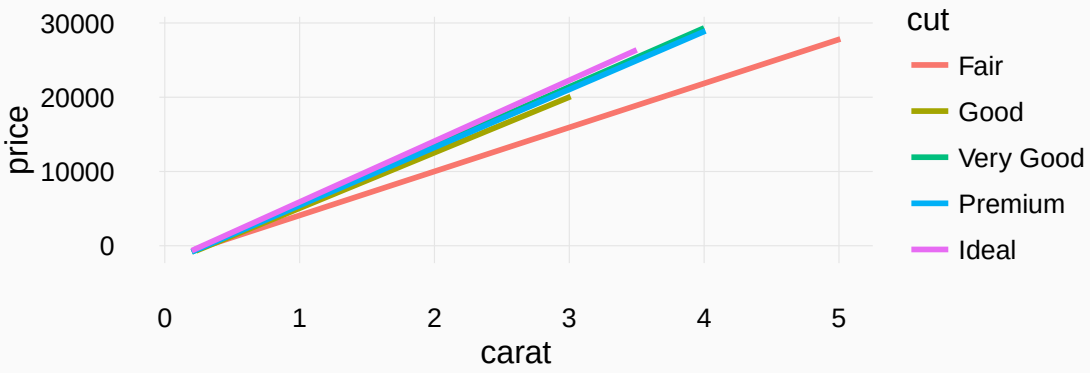
Smooth - SE's

```
ggplot(data = diamonds, mapping = aes(x = carat, y = price, color = cut)) +
  geom_smooth(se = FALSE)
```



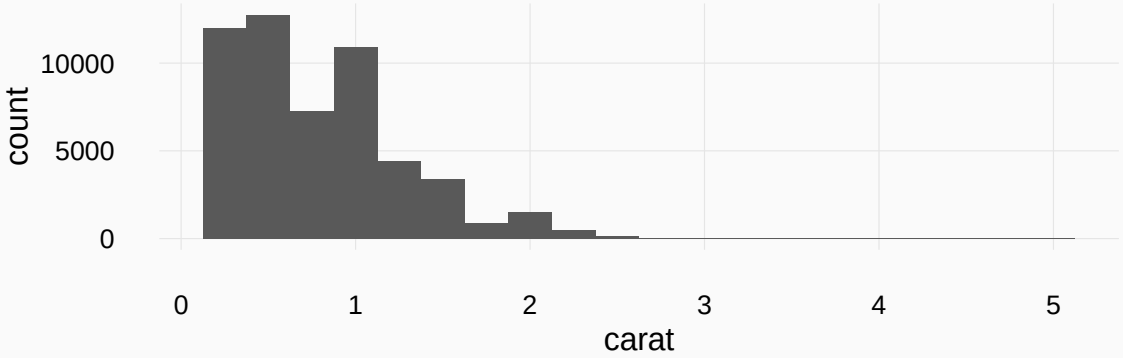
Smooth - OLS

```
ggplot(data = diamonds, mapping = aes(x = carat, y = price, color = cut)) +
  geom_smooth(se = FALSE, method = "lm")
```



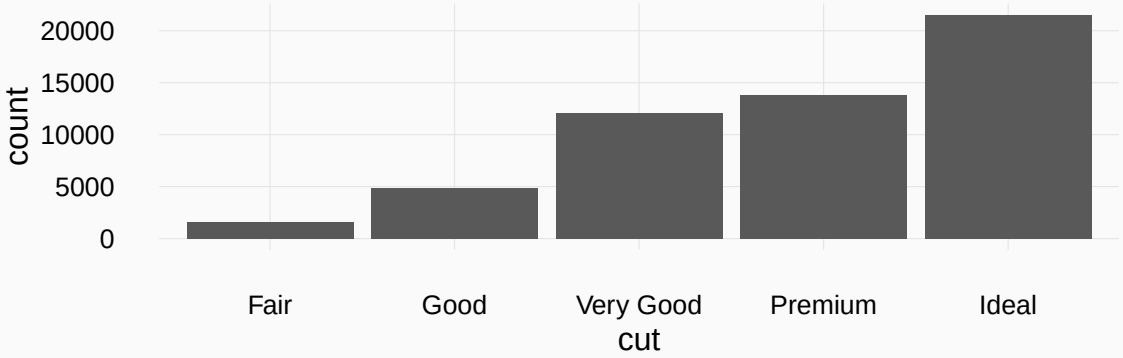
Histogram

```
ggplot(data = diamonds, mapping = aes(x = carat)) +
  geom_histogram(binwidth = .25)
```



Bar charts

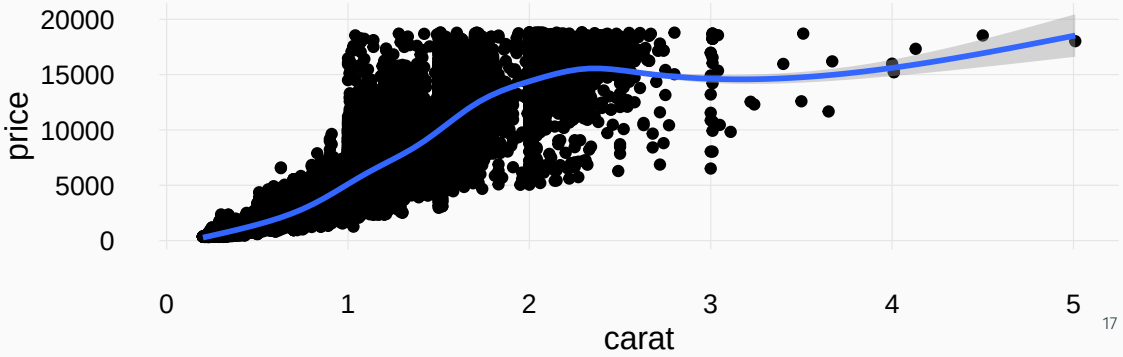
```
ggplot(data = diamonds, mapping = aes(x = cut)) +
  geom_bar()
```



Combining geoms

We can layer multiple geoms on top of each other:

```
ggplot(data = diamonds, mapping = aes(x = carat, y = price)) +
  geom_point() + geom_smooth()
```

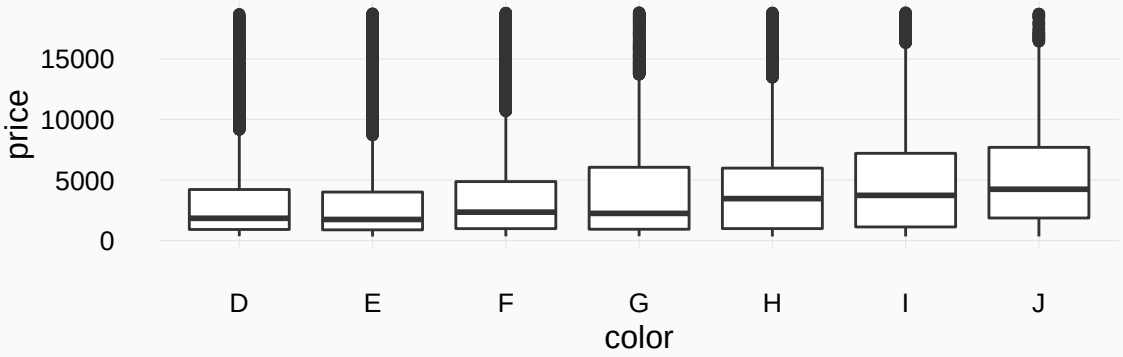


You try!

Make a boxplot with color on the x-axis and price on the y-axis

You try (answers)

```
ggplot(data = diamonds, mapping = aes(x = color, y = price)) +
  geom_boxplot()
```



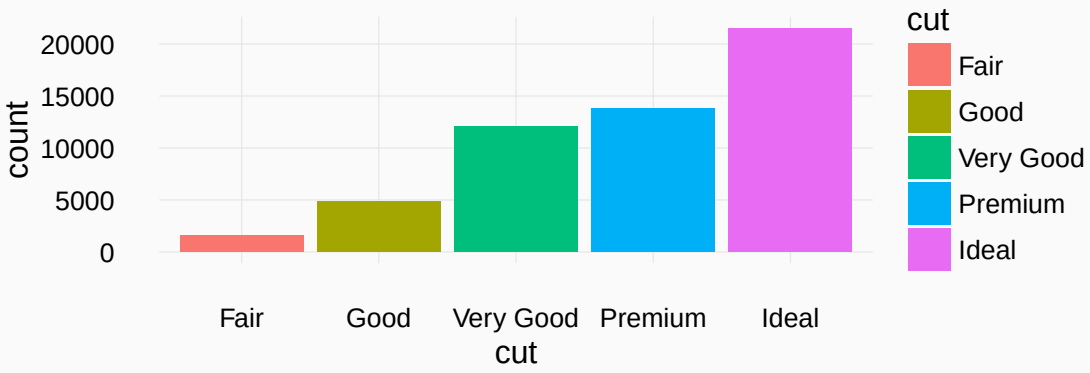
Fill vs color

What do you expect this to do:

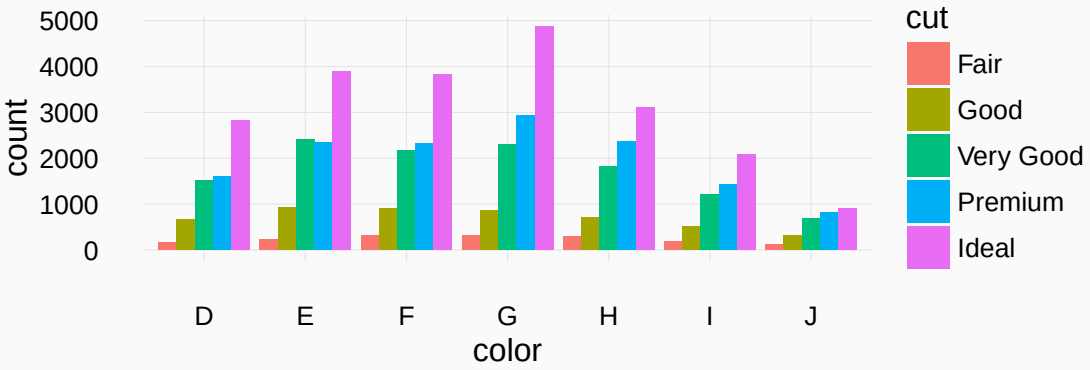
```
ggplot(data = diamonds, mapping = aes(x = cut, color = cut)) +
  geom_bar()
```

Use fill for bar charts

```
ggplot(data = diamonds, mapping = aes(x = cut, fill = cut)) +
  geom_bar()
```



Try to make this plot



Positions

```
ggplot(data = diamonds, mapping = aes(x = color, fill = cut)) +
  geom_bar(position = "dodge")
```

There's "stack", "dodge", "identity", "fill" - try them all out!