Topics in Data Visualization

ggplot2

Jun 30 2014

Charlotte Wickham

vis.cwick.co.nz

The ggplot2 grammar

The components of a plot are:

- a default dataset and default set of mappings for variables to aesthetics

- one or more layers each with

  - a geometric object

  - a statistical transformation

  - a position adjustment

  - a dataset

  - a set of aesthetic mappings

- one scale for each aesthetic mapping

- a coordinate system

- a facet specification
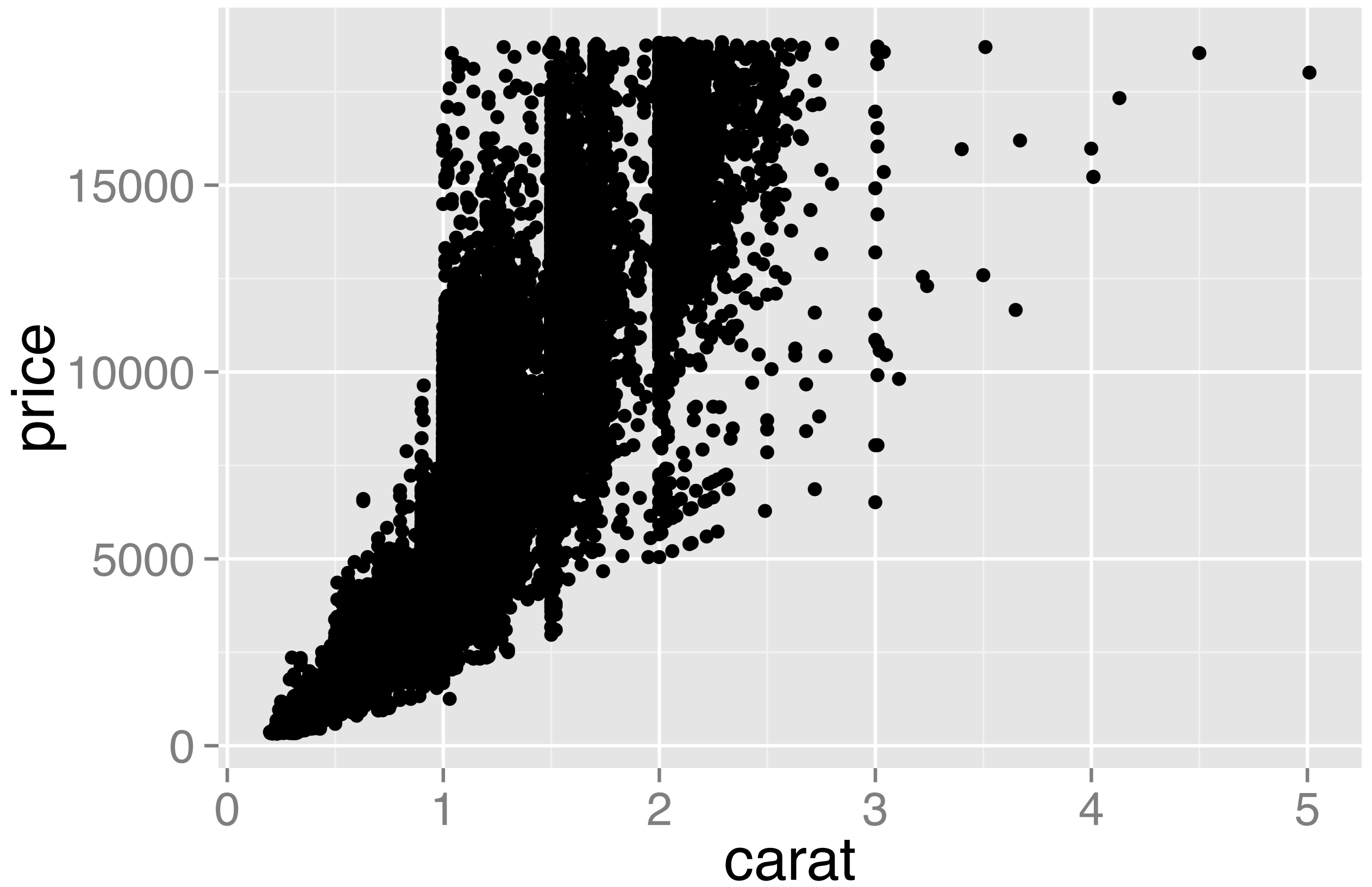
ggplot2 is a domain specific language

A language specifically built for specifying graphics.

ggplot2 is an R package - it's written in R
but it doesn't necessarily behave like R.

Expects data in a data.frame (if not looks for a fortify method for it)

Evaluates a lot of arguments relative to the data (**non-standard evaluation**)

+ has a special meaning

```
ggplot() +
  layer(
    data = diamonds,
    mapping = aes(x = carat, y = price),
    geom = "point",
    stat = "identity",
    position = "identity" ) +
  scale_y_continuous() +
  scale_x_continuous() +
  coord_cartesian()
```

Intelligent defaults save typing...

```
ggplot() +
  layer(
    data = diamonds,
    mapping = aes(x = carat, y = price),
    geom = "point")
```

Each geom has a default statistic and position, scales are automatically generated for each aesthetic, cartesian coordinates are default.

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

use the shortcut geom_xxx instead of layer(geom = "xxx",…)

define default data and mapping in ggplot, so don't have to specify it in layer

```
ggplot(diamonds, aes(carat, price)) +
geom_point()
```

use positional argument matching

learning ggplot2 involves…

becoming familiar with:

    the geometric objects available

        their aesthetics

        their defaults statistics and positioning

    ways to change and adjust scales

    the coordinate systems available

    the theming system

using ggplot2 to make good plots involves…

getting comfortable moving between ideas in your head to ggplot2 specifications

identifying when the data is not in the right form for the plot you want

getting exposure to different ways to display data

critically evaluating the plots you make

becoming familiar with:

the geometric objects available

their aesthetics

their defaults statistics and positioning

http://docs.ggplot2.org/

+ lots of playing around

Where the Heat and the Thunder Hit Their Shots

The shooting patterns for the players on the Miami Heat and the Oklahoma City Thunder reveal where they are most dangerous on the court. Below, compare each player's strengths using court maps and analysis by Kirk Goldsberry, a geography professor at Michigan State. Related Article »

| All Shots	3-Pointers	Midrange	Close Range	Number of attempts Low ○ ○ ○ High	Points per region Low ▬▬ High

Miami Heat

TOTAL SHOTS **5,209** | POINTS PER SHOT **1.01** | F.G. PERCENT **47%**

Oklahoma City Thunder

TOTAL SHOTS **5,228** | POINTS PER SHOT **1.03** | F.G. PERCENT **47.1%**

Can we replicate this?

The Heat rely on player positioning to create isolation plays for LeBron James and Dwyane Wade, often on the left side. The Heat take many fewer 3-point shots than the Thunder.
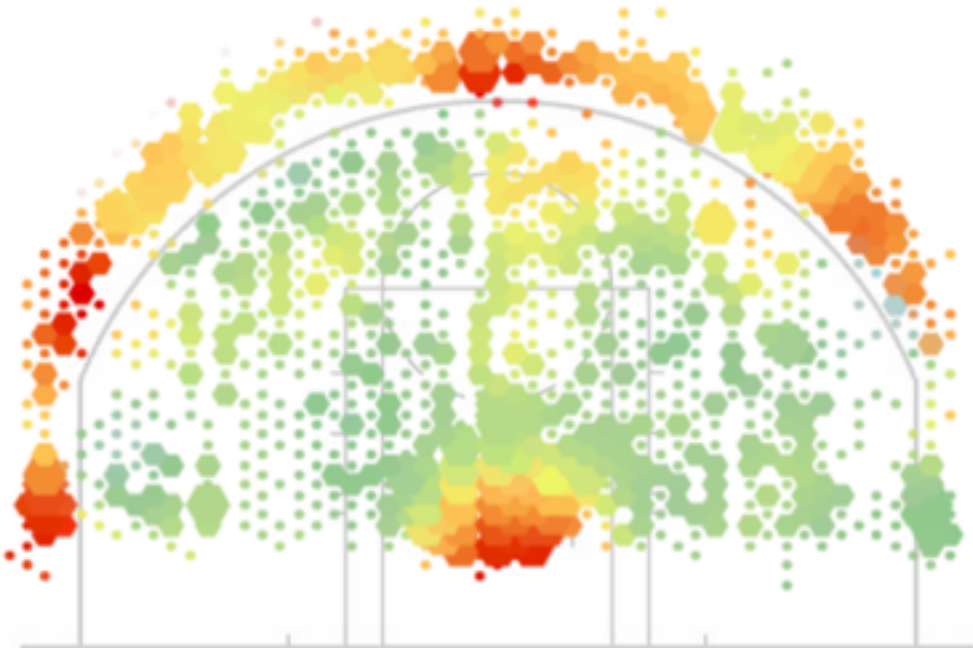
The Thunder are effective from almost any area on the court and shoot many more 3-point shots than the league average. Kevin Durant and James Harden are potent from the top of the arc.

http://www.nytimes.com/interactive/2012/06/11/sports/basketball/nba-shot-analysis.html

Distance from end line	Distance from LHS	Number of shots	Average number of points	Team
0.25	0.25	5	1.2	Miami Heat
0.25	0.50	10	2.1	Miami Heat
0.25	0.75	5	0.8	Miami Heat
0.25	1.00	3	0.1	Miami Heat

shots_sum = All LA Lakers shots 2009/10

```
> head(shots_sum)
  x  y num_shots num_made prop_made avg_points total_points
1 0  4         2        1       0.5        1.5            3
2 0  5         3        0       0.0        0.0            0
3 0  6         1        0       0.0        0.0            0
4 0  7         1        0       0.0        0.0            0
5 0  9         2        0       0.0        0.0            0
6 0 10         1        0       0.0        0.0            0
```