Teaching R to beginners: a false dichotomy

Nicholas Horton (nhorton@amherst.edu)

July 24, 2017

Teaching beginners: a false dichotomy

I've been reading David Robinson's excellent blog entry "Teach the tidyverse to beginners" (http://varianceexplained.org/r/teach-tidyverse), which argues that a tidyverse approach is the best way to teach beginners. He summarizes two *competing* curricula:

1) "Base R first": teach syntax such as $ and [[]], built in functions like ave() and tapply(), and use base graphics
2) "Tidyverse first": start from scratch with %>% and leverage dplyr and use ggplot2 for graphics

If I had to choose an approach, I'd also go with 2) ("Tidyverse first"), since it helps to move us closer to helping our students "think with data" using more powerful tools (see http://amstat.tandfonline.com/doi/full/10.1080/00031305.2015.1094283 for my sermon on this topic).

A third way

Of course, there's a third option that addresses David's imperative to "get students doing powerful things quickly". The mosaic package (https://cran.r-project.org/web/packages/mosaic), was written to make R easier to use in introductory statistics courses. The package is part of Project MOSAIC (http://mosaic-web.org), an NSF-funded initiative to integrate statistics, modeling, and computing. A paper outlining the mosaic package's "Less Volume, More Creativity" approach was recently published in the R Journal (https://journal.r-project.org/archive/2017/RJ-2017-024). To his credit, David mentions the mosaic package in a response to one of the comments on his blog.

Less Volume, More Creativity

One of the big ideas in the mosaic package is that students build on the existing formula interface in R as a mechanism to calculate summary statistics, generate graphical displays, and fit regression models. Randy Pruim has dubbed this approach "Less Volume, More Creativity".

While teaching this formula interface involves adding a new learning outcome (what is "Y ~ X"?), the mosaic approach simplifies calculation of summary statistics by groups and the generation of two or three dimensional displays on day one of an introductory statistics course (see for example Wang et al., "Data Viz on Day One: bringing big ideas into intro stats early and often" (2017), TISE, http://escholarship.org/uc/item/84v3774z).

The formula interface also prepares students for more complicated models in R (e.g., logistic regression, classification).

Here's a simple example using the `diamonds` data from the `ggplot2` package. We model the relationships between two colors (D and J), number of carats, and price.

I'll begin with a bit of data wrangling to generate an analytic dataset with just those two colors. (Early in a course I would either hide the next code chunk or make the `recoded` dataframe accessible to the students to avoid cognitive overload.)

```
recoded <- diamonds %>%
  filter(color=="D" | color=="J") %>%
  mutate(col = as.character(color))
```

We first calculate the mean price (in US$) for each of the two colors.

```
mean(price ~ col, data = recoded)
```

```
##    D    J
## 3170 5324
```

This call is an example of how the formula interface facilitates calculation of a variable's mean for each of the levels of another variable. We see that D color diamonds tend to cost less than J color diamonds.

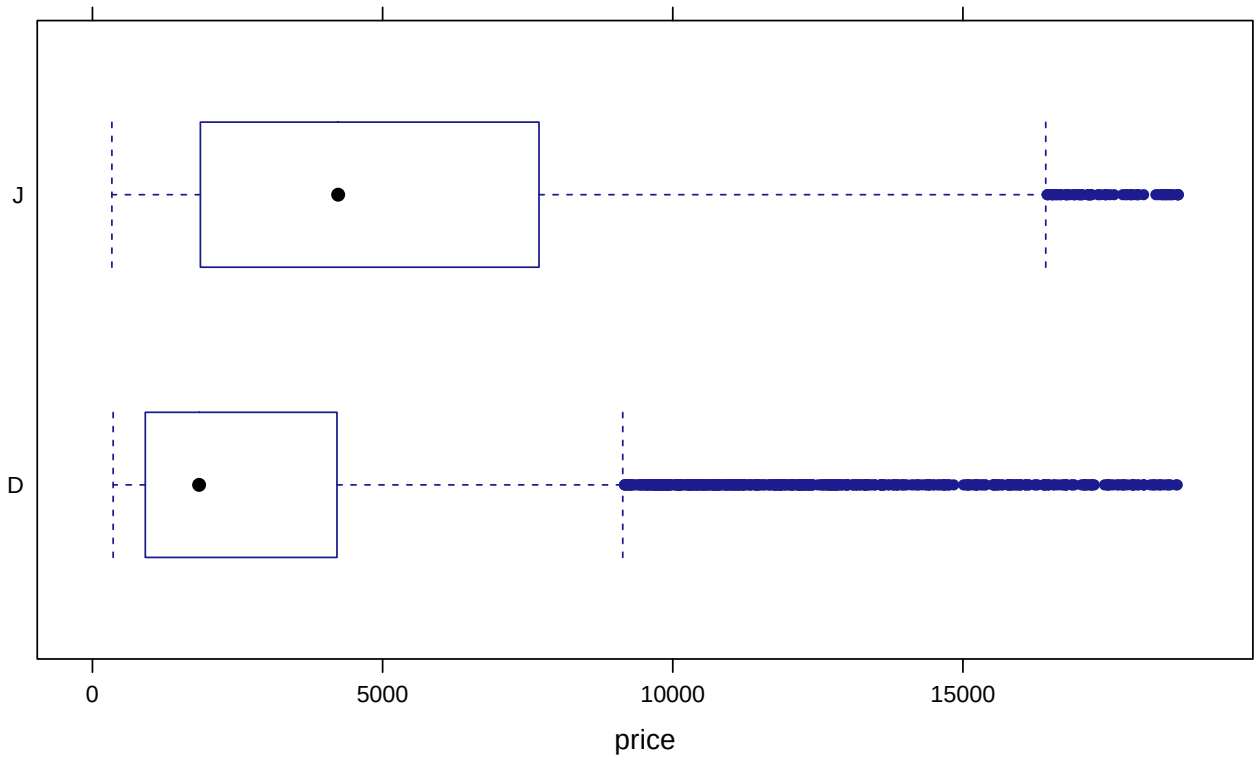
A useful function in `mosaic` is `favstats()` which provides a larger set of statistics and information for sample size and missing values.

```
favstats(price ~ col, data = recoded)
```

```
##   col min   Q1 median   Q3   max mean   sd    n missing
## 1   D 357  911   1838 4214 18693 3170 3357 6775       0
## 2   J 335 1860   4234 7695 18710 5324 4438 2808       0
```

A similar command can be used to generate side by side boxplots. Here we illustrate the use of lattice graphics. (An alternative formula based graphics system (ggformula) will be the focus of a future post.)

```
bwplot(col ~ price, data = recoded)
```



The distributions are skewed to the right (not surprisingly since they are prices). If we wanted to formally compare these sample means we could do so with a two-sample t-test (or in a similar fashion, by fitting a linear model).

```
t.test(price ~ col, data = recoded)
```

```
##
##  Welch Two Sample t-test
##
## data:  price by col
## t = -20, df = 4000, p-value <2e-16
```

```
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2336 -1971
## sample estimates:
## mean in group D mean in group J
##            3170            5324
```

```
msummary(lm(price ~ col, data = recoded))
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3170.0       45.0    70.4   <2e-16 ***
## colJ          2153.9       83.2    25.9   <2e-16 ***
##
## Residual standard error: 3710 on 9581 degrees of freedom
## Multiple R-squared:  0.0654, Adjusted R-squared:  0.0653
## F-statistic:  670 on 1 and 9581 DF,  p-value: <2e-16
```

The results from the two approaches are consistent: the group differences are highly statistically significant. We could conclude that J diamonds tend to cost more than D diamonds, back in the population of all diamonds.

Let's do a quick review of the syntax to date:

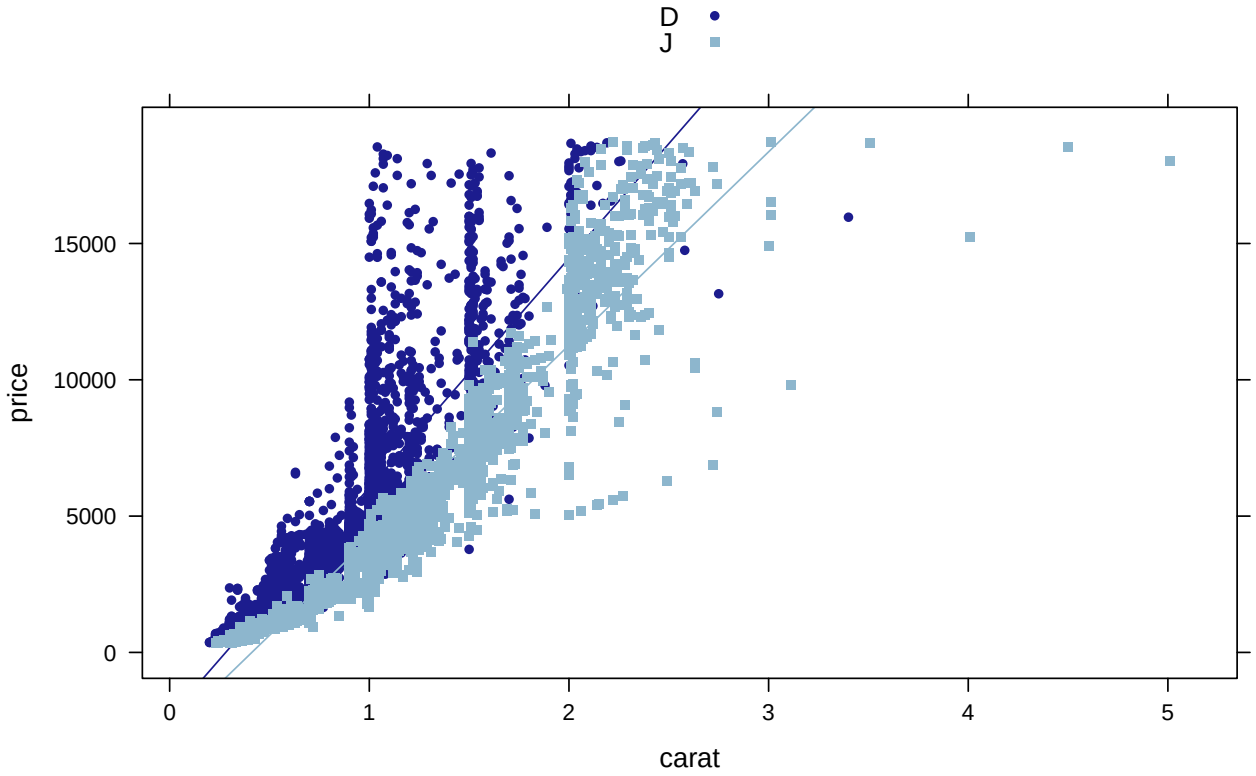
```
mean(price ~ col)
favstats(price ~ col)
bwplot(col ~ price)
t.test(price ~ col)
lm(price ~ col)
```

See the pattern?

On a statistical note, it's important to remember that the diamonds were *not* randomized into colors: this is a found (observational dataset) so there may be other factors at play. The revised GAISE College report reiterates the importance of multivariate thinking in intro stats: http://www.amstat.org/asa/education/Guidelines-for-Assessment-and-Instruction-in-Statistics-Education-Reports.aspx)

Let's continue with the "Less Volume, More Creativity" approach to bring in a third variable: the number of carats in each diamond.

```
xyplot(price ~ carat, groups=col, auto.key=TRUE, type=c("p", "r"), data = recoded)
```

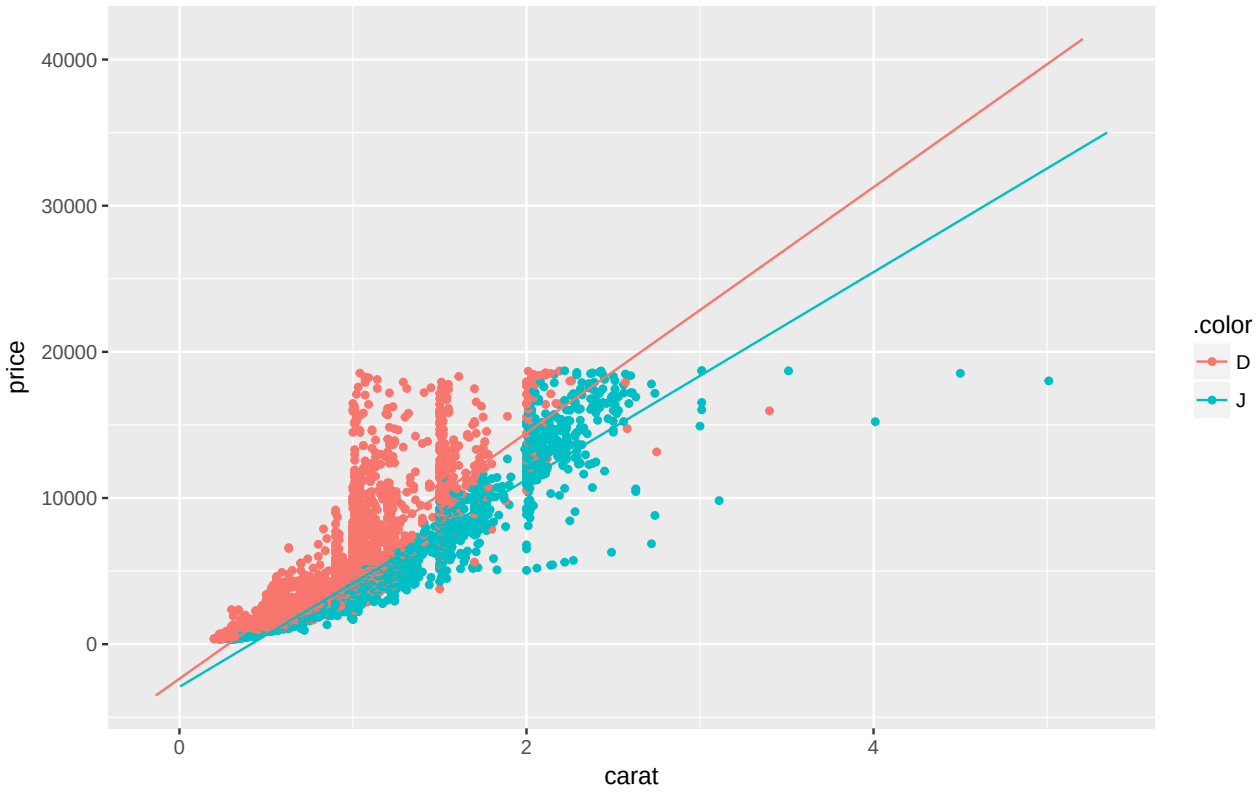


We see that controlling for the number of carats, the D color diamonds tend to sell for more than the J color diamonds. We can confirm this with a regression model (and display the resulting predicted values using `plotModel()`).

```
mod <- lm(price ~ carat + col + carat*col, data = recoded)
msummary(mod)
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2361.0       36.2  -65.26  < 2e-16 ***
## carat         8408.4       48.3  174.22  < 2e-16 ***
## colJ          -559.6       69.3   -8.08  7.5e-16 ***
## carat:colJ   -1314.2       66.2  -19.86  < 2e-16 ***
##
## Residual standard error: 1430 on 9579 degrees of freedom
## Multiple R-squared:  0.861,  Adjusted R-squared:  0.861
## F-statistic: 1.98e+04 on 3 and 9579 DF,  p-value: <2e-16
```

```
plotModel(mod, system="ggplot2")
```



We observe a highly statistically significant interaction. The association between price and carat is much smaller for J diamonds than for D diamonds. This is a great example of Simpson's paradox (https://en.wikipedia.org/wiki/Simpson%27s_paradox): accounting for the number of carats has yielded opposite results from a model that didn't include carats.

Moving beyond mosaic

The revised GAISE College report enunciated the importance of technology when teaching statistics. Many courses still use calculators or web-based applets to incorporate technology into their classes.

R is an excellent environment for teaching statistics, but many instructors feel uncomfortable using it (particularly if they succumb to the $ and [[]] syntax, which many find offputting). The mosaic approach helps make the use of R feasible for many audiences by keeping things simple.

It's unfortunately true that many introductory statistics courses don't move beyond bivariate relationships (so students may feel paralyzed about what to do about other factors). The mosaic approach has the advantage that it can bring multivariate thinking, modeling, and exploratory data tools together with a single interface (and modest degree of difficulty in terms of syntax). I've been teaching multiple regression as a descriptive method early in an intro stat course for the past ten years (and it helps to get students excited about material that they haven't seen before).

The mosaic approach also scales well: it's straightforward to teach students dplyr/tidyverse data wrangling by adding in the pipe operator and some key data idioms. (So perhaps the third option should be labeled "mosaic and tidyverse".) See the following for an example of how favstats() can be replaced by dplyr idioms.

```
recoded %>%
  group_by(col) %>%
  summarize(meanval = mean(price, na.rm=TRUE))
```

```
## # A tibble: 2 x 2
```

```
##     col meanval
##   <chr>   <dbl>
## 1     D    3170
## 2     J    5324
```

That being said, I suspect that many students (and instructors) will still use `favstats()` for simple tasks (e.g., to check sample sizes, check for missing data, etc). I know that I do. But the important thing is that unlike training wheels, mosaic doesn't hold them back when they want to learn new things.

I'm a big fan of `ggplot2`, but even Hadley agrees that the existing syntax is not what he wants it to be (http://coleoguy.blogspot.com/2015/08/hadley-and-winston-talk-about-future-of.html). While it's not hard to learn to use `+` to glue together multiple graphics commands and to get your head around aesthetics, teaching `ggplot2` adds several additional learning outcomes to a course that's already overly pregnant with them.

Side note

A lot of what is in mosaic *should* have been in base R (e.g., formula interface to `mean()`, data= option for `mean()`). Other parts are more focused on teaching (e.g., `plotModel()`, `xpnorm()`, and resampling with the `do()` function).

Closing thoughts

In summary, I argue that the mosaic approach is consistent with the tidyverse. It dovetails nicely with David's "Teach tidyverse" as an intermediate step that may be more accessible for undergraduate audiences without a strong computing background. I'd encourage people to check it out (and let us know if there are ways to improve the package).

Want to learn more about mosaic? In addition to the R Journal paper referenced above, you can see how we get students using R quickly in the package's "Less Volume, More Creativity" and "Minimal R" vignettes. We also provide curated examples from commonly used textbooks in the "mosaic resources" vignette and a series of freely downloadable and remixable monographs including "The Student's Guide to R" and "Start Teaching with R" at https://github.com/ProjectMOSAIC/LittleBooks.