

# R Bootcamp

## Day 1 Exercises — ggplot

September 14, 2015

Create a new RMarkdown document in RStudio (File -> New -> RMarkdown -> Document -> HTML)

Create a code chunk to load the `ggplot2` package (via `library(ggplot2)`). Use the `echo` argument to the code chunk to avoid printing the output of this unexciting command. (Hint: this is quite similar to the `eval` argument that appears in the template document.)

For all of the following exercises, write a brief description of what you intend to do, show the code used to do it and the output generated, and write a brief description of your interpretation.

There is a dataset on the price and attributes of diamonds that comes with the `ggplot2` package. Examine the `diamonds` data.frame using `str`, `summary`, `head`, and/or `View`. What relationships might be worth exploring graphically?

1. Examine the distribution of diamond size ( `carat` ) via
  - a. a histogram
  - b. density curve

Be sure to adjust the bin (via `binwidth` ) or kernel (via `adjust` ) to capture any interesting patterns.

2.
  - a. It appears there are several qualities of diamond `cut` in this dataset (Can you tell how many from `str(diamonds)` ). Create a histogram and density curve to explore the distribution of sizes across the different cut qualities. You could separate cuts by color or fill or facet (or `linetype` , or ???).
  - b. Which of the above plots would be more appropriate if we were interested in absolute quantity of diamonds of each cut, and which would be more appropriate if we were interested in the relative distribution of sizes within each cut? Why?
3. Create a boxplot that compares the distribution of sizes within each `cut` .
4. Change the above plot to be a violin plot. Add a single line to facet the plots by both color and clarity. Don't worry about the legibility of the resulting plot.
5. Let's try to get all that information in a more comprehensible plot. Leave `carat` on the y-axis, and experiment with mapping `cut` , `clarity` , and `color` to x, color/fill, and facets to produce a reasonable plot. Use whichever of boxplots or violin plots is more legible.
6.
  - a. Create a scatterplot of diamond price as a function of size.
  - b. There is a lot of overplotting there, so information is being lost. Adjust the alpha level of the points to fix this.
  - c. Add marginal rug plots to provide information about the distribution of both variables. Consider adjusting the alpha level of the rug plots too.

- d. Add a smoothing curve to plot. What do you think about the right side of the curve? Consider that and the marginal distributions of each variable. Should you transform either or both of the axes? Experiment.
  - e. An expert tells you that diamonds larger than 3 carats or smaller than 0.25 carats are outliers. Set the x-axis limits to exclude those outliers. Also, change the smoothing algorithm to a linear fit.
  - f. The same expert tells you the relationship between price and size depends on the quality of the `cut`. Test her statement by coloring according to `cut`. Should the rugs be colored by cut? How about the fitted lines? Does her statement appear to be true?
  - g. Challenge question: Time to get the plot ready for publication! Replace the ugly gray background by using the `theme_bw`. Remove the rug plots and replace them with log-tick marks if either variable is log-transformed. Change the axis labels to variable (unit). Reverse the order of the legend entries so that they are in the same order as the lines in the plot, change the legend title to “Cut quality”, and move the legend to the empty space in the upper-left part of the graph. The “R Cookbook” <http://www.cookbook-r.com/Graphs/> (<http://www.cookbook-r.com/Graphs/>) may be helpful here; ggplot Google queries often lead to it.
7. There are several other variables in the `diamonds` dataset. Explore their distributions, marginal distributions, relationships with each other, and effects on the relationship between price and size. More information on the dataset is available via `?diamonds`. Consider mapping variables to color/fill, shape/linetype, axes, facets, etc.