

Putting it together

Abhijit Dasgupta, PhD

Spring 2019

Customization

Colors

ggplot2 has several ways to customize colors

1. If colors are based on categorical data

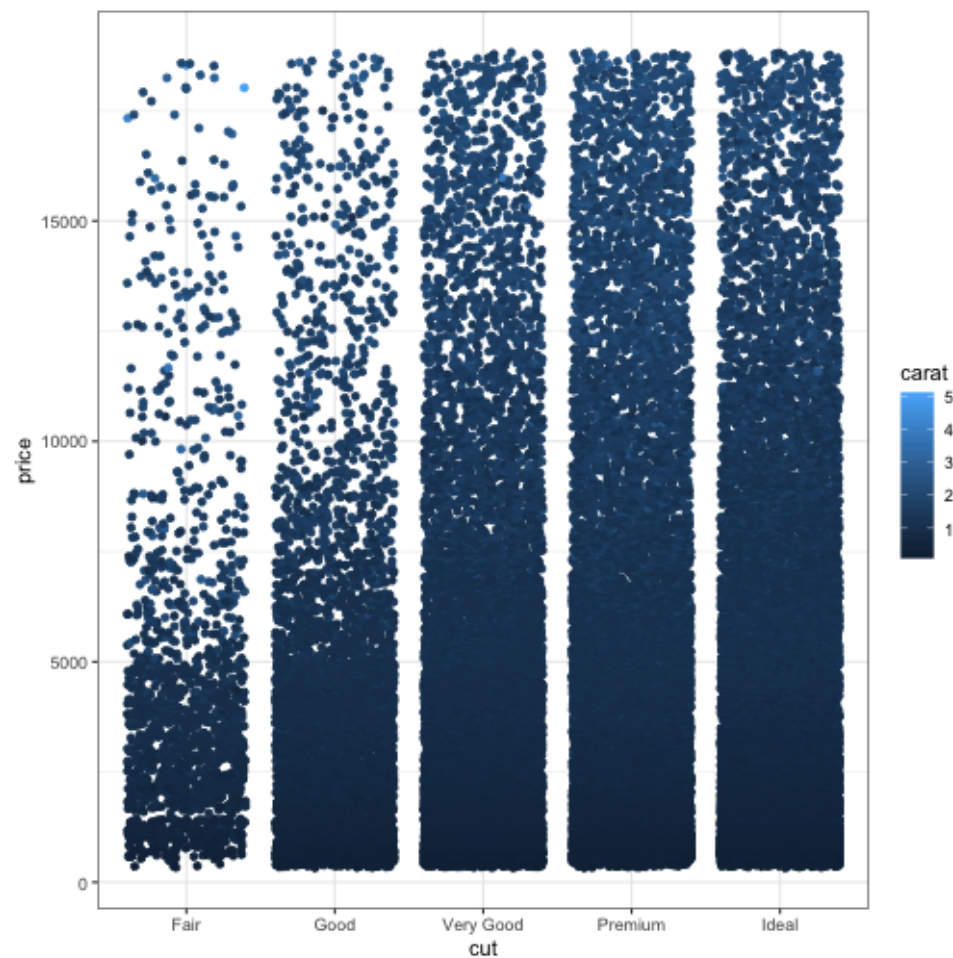
- `scale_color_manual`
- `scale_fill_manual`

2. If colors are based on continuous data

- `scale_{color,fill}_gradient` makes sequential gradients (specify low and high colors)
- `scale_{color,fill}_gradient2` makes divergent gradients (specify low, middle and high colors)

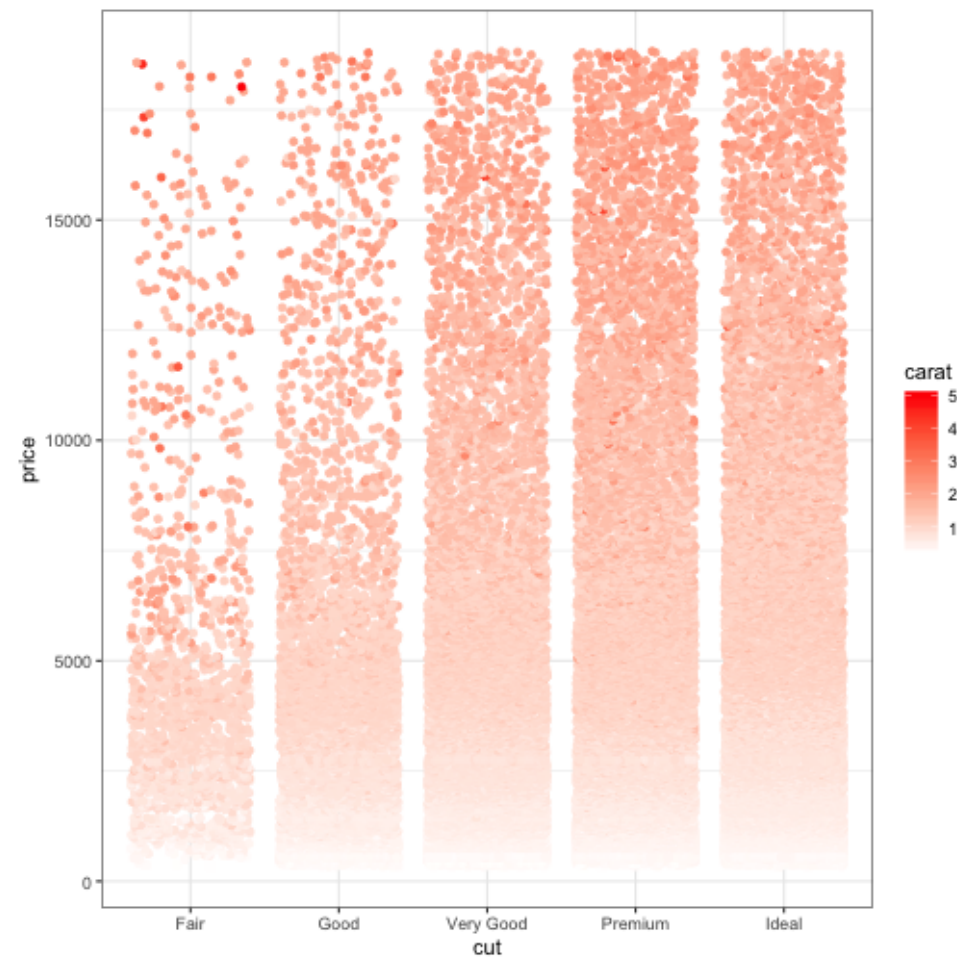
Colors

```
(g1 <- ggplot(diamonds,  
              aes(x = cut, y = price,  
                  color = carat))+  
  geom_jitter()  
)
```



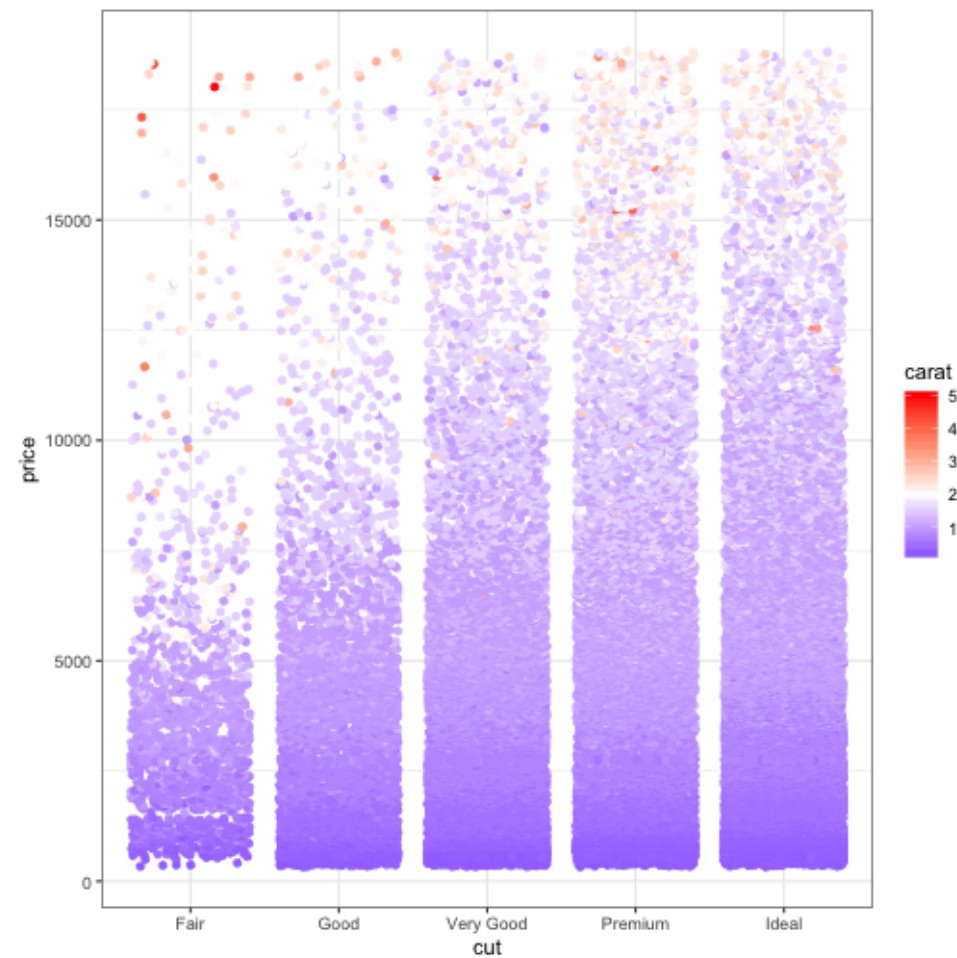
Colors

```
g1 + scale_color_gradient(low='white',high = 'red')
```



Colors

```
g1 + scale_color_gradient2(low = 'blue',  
                           mid='white',  
                           high='red',  
                           midpoint = 2)
```



Text

The extrafont package allows you to use fonts already on your computer in your graphics.

```
library(extrafont)
loadfonts()

g1 + theme(text = element_text(family='Georgia'))
```

Text

The `extrafont` package allows you to use fonts already on your computer in your graphics.

```
g1 + theme(text = element_text(family='Comic Sans MS')
```


Saving your work

File types

Raster

- PNG, TIFF, JPEG
- Have to specify a resolution (dots per inch or dpi)
- Enlarging a file can create pixelation
- Preferred for web

Vector

- PDF, SVG, EMF
- Infinite resolution
- No pixelation
- Preferred for print

Saving your work

Generally, in `ggplot2`, you can use the `ggsave` function with a file name. It figures out the file type from the last letters

```
ggsave('test1.pdf', plot = g1)  
ggsave('test1.tiff', plot = g1)
```

If you're saving the last graph rendered, you don't need the `plot = g1` part.

You can also specify height and width of the graph.

A note about TIFF files

TIFF files are not saved with the proper resolution from R. It defaults to 72 dpi, which is inadequate for publication.

Save your graph as PDF, and then use Acrobat or Illustrator to change it to a TIFF file with appropriate resolution. Also, make sure that you use **LZW** compression, otherwise the TIFF file will be unreasonably large.

I have some scripts that I can make available for conversion

Creating a final product

R Markdown

[R Markdown](#) is a lightweight markup language that can inter-weave text and R code to create different kinds of outputs.

The [formats page](#) gives you the breadth of documents that can be generated using R Markdown.

R Markdown

You can directly create PowerPoint presentations from R Markdown. This may be useful for your final presentations

To create PDF documents, you must have LaTeX installed. This can be achieved by installing the `tinytex` package in R. Or, you can install [MikTeX](#) on Windows or [MacTeX](#) on Mac OS X.

Demos

Dashboard

The `flexdashboard` package can help create interactive dashboards using R Markdown. The example we will use is based on [this](#) flexdashboard example. The data I use is also in that repository.