STAT 215A Fall 2023 Week 3

Chengzhong Ye

Lab 1: Clarifications

- Your .Rmd should generate the plot directly (i.e., put the plotting code in the .Rmd file)
  - Try not to save plots separately and load them in
- “Recall the three realms of data science: data / reality, algorithms / models, and **future data / reality**. Where do the different parts of this lab fit into those three realms?”
  - OK if you want to argue not all three realms are covered, but explain why.
- Homework 1 3.3 “Open feedback loop”

Lab 1: if you're stuck

Some thoughts if you're stuck:

- Use your domain knowledge and curiosity to come up with questions you may want to answer
- Check out other papers on CDI development for ideas of what features to focus on
- Look at smaller parts of the data
    - Zoom in on a specific set of features, maybe just demographics or injury presentation features

Will release a sample lab report from past year (for a different dataset)

Lab 1: Using .gitignore

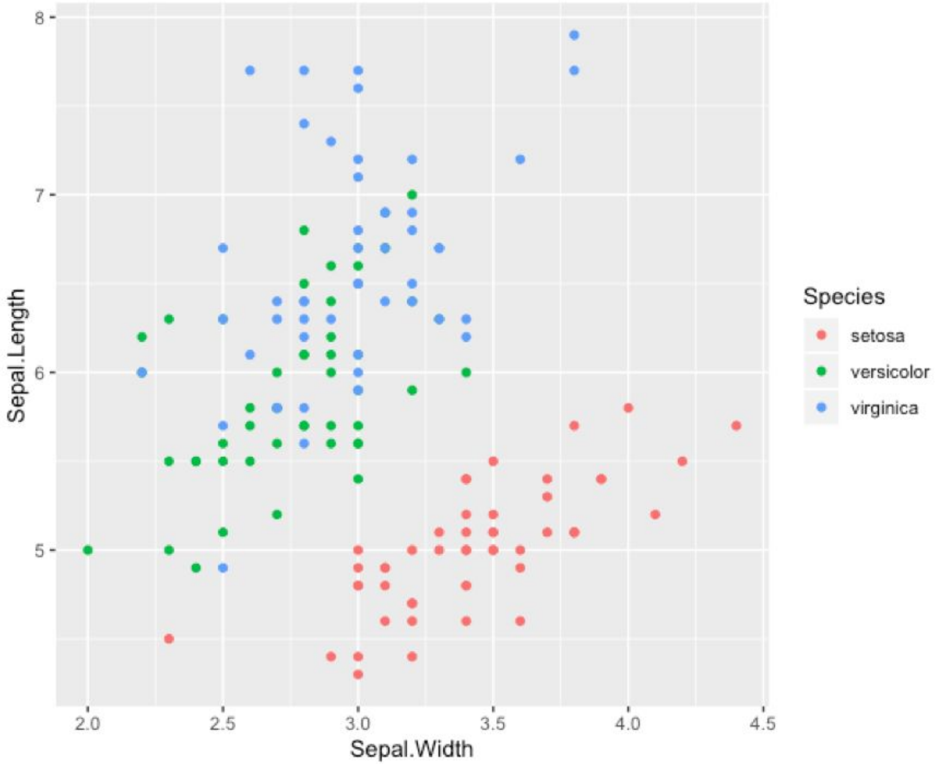
Please be sure to add a .gitignore file to the top directory of your `stat-215-a` repository:

- Useful examples here: https://github.com/github/gitignore
- Add what you don't want to be put in version control:
  - data/ (matches
  - documents/
  - *.csv
  - Exception: !dont_ignore_me.csv
  - .gitignore uses globbing patterns. See https://git-scm.com/docs/gitignore
- Citations: include in bibliography, but don't push pdfs

Lab 1: Check-in

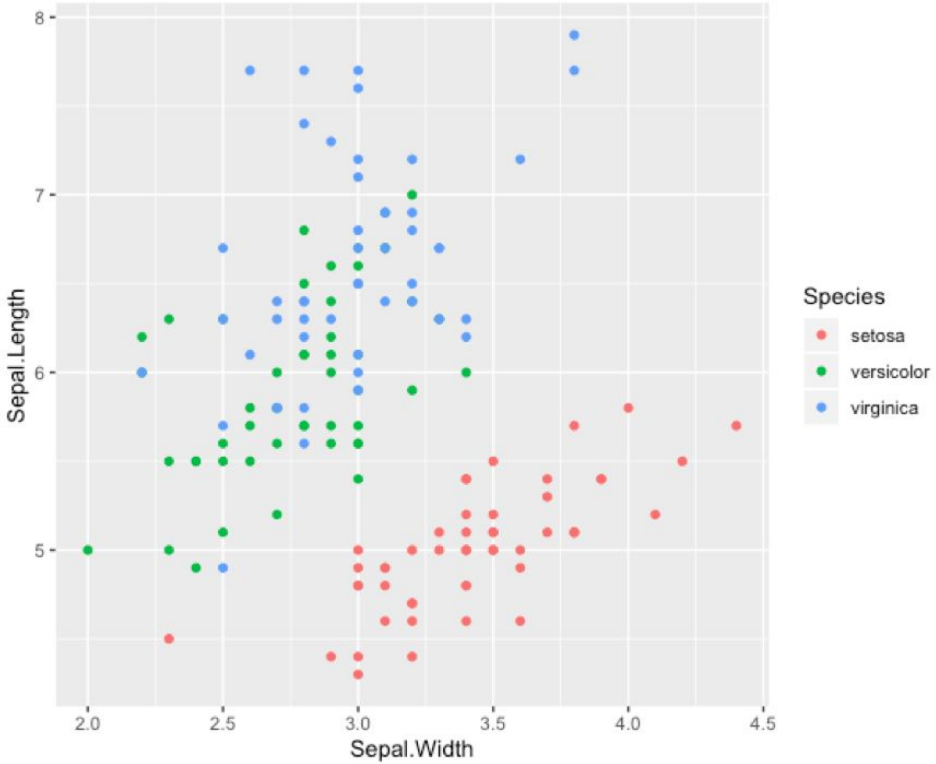
- How is it going?
- Having fun?
- Challenges?
- Questions?
- Remember the due **Friday, Sept 22 at 11:59pm**
- Berkeley SCF Resources: https://github.com/berkeley-scf

Motivation for today



(Selfish) motivation for today

As your GSI, it can become monotonous to look at 100+ plots with the same gridded gray ggplot background and the same default ggplot color scheme... please don't make me go through that

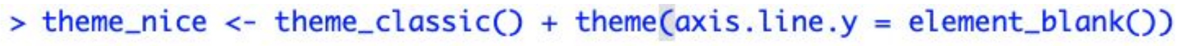


Let’s fix this

- Built-in and custom `ggplot` themes
- Color schemes
- Heatmaps with `superheat`
- `GGally` pair plots
- Ridge Density Plots
- Interactive plots

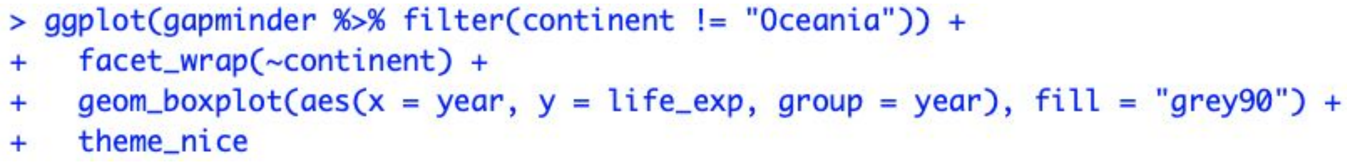
Quick improvements to the classic `ggplot` theme

- Recall in the gapminder lab last week, we had defined this `theme_nice` in utils.R



```
> theme_nice <- theme_classic() + theme(axis.line.y = element_blank())
```

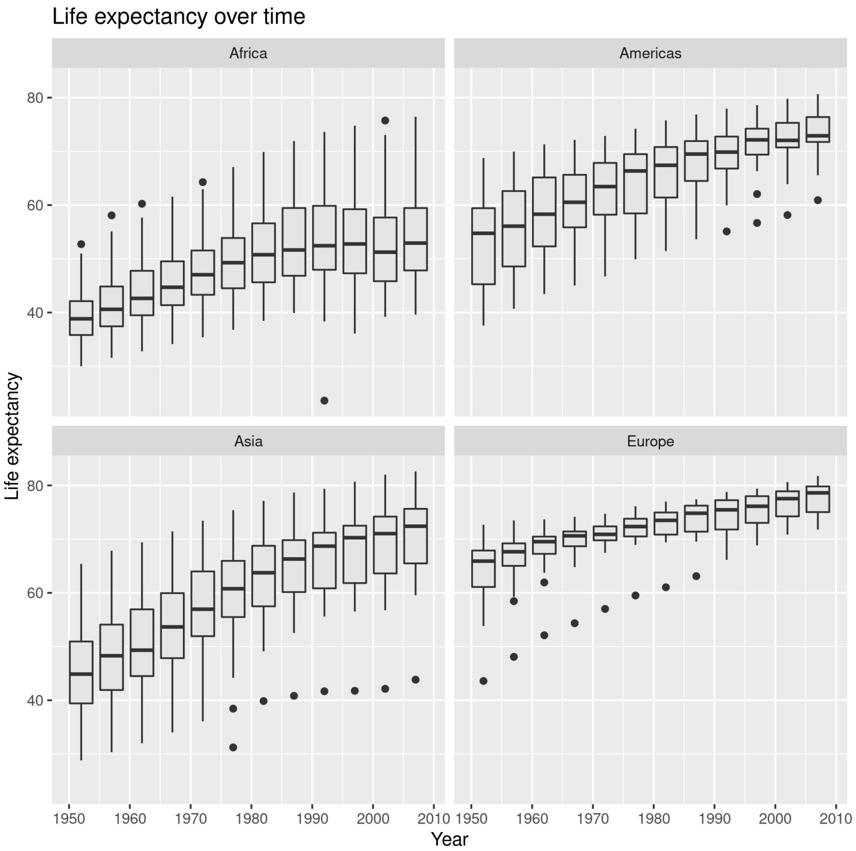
- Then to use this modified theme, we simply ran something like

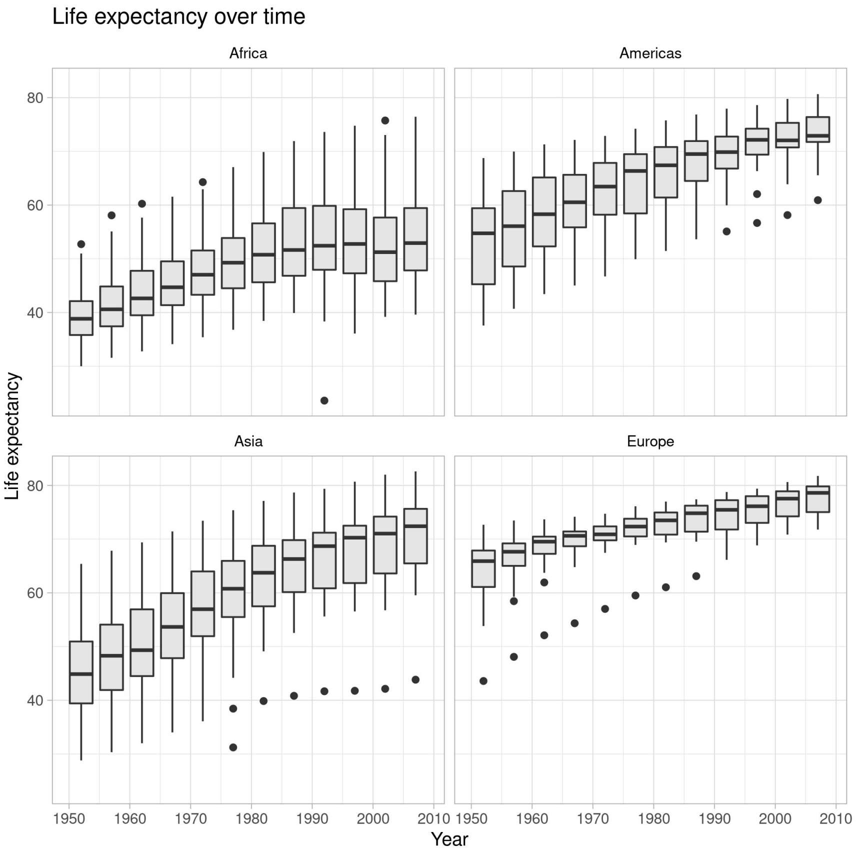


```
> ggplot(gapminder %>% filter(continent != "Oceania")) +
+   facet_wrap(~continent) +
+   geom_boxplot(aes(x = year, y = life_exp, group = year), fill = "grey90") +
+   theme_nice
```

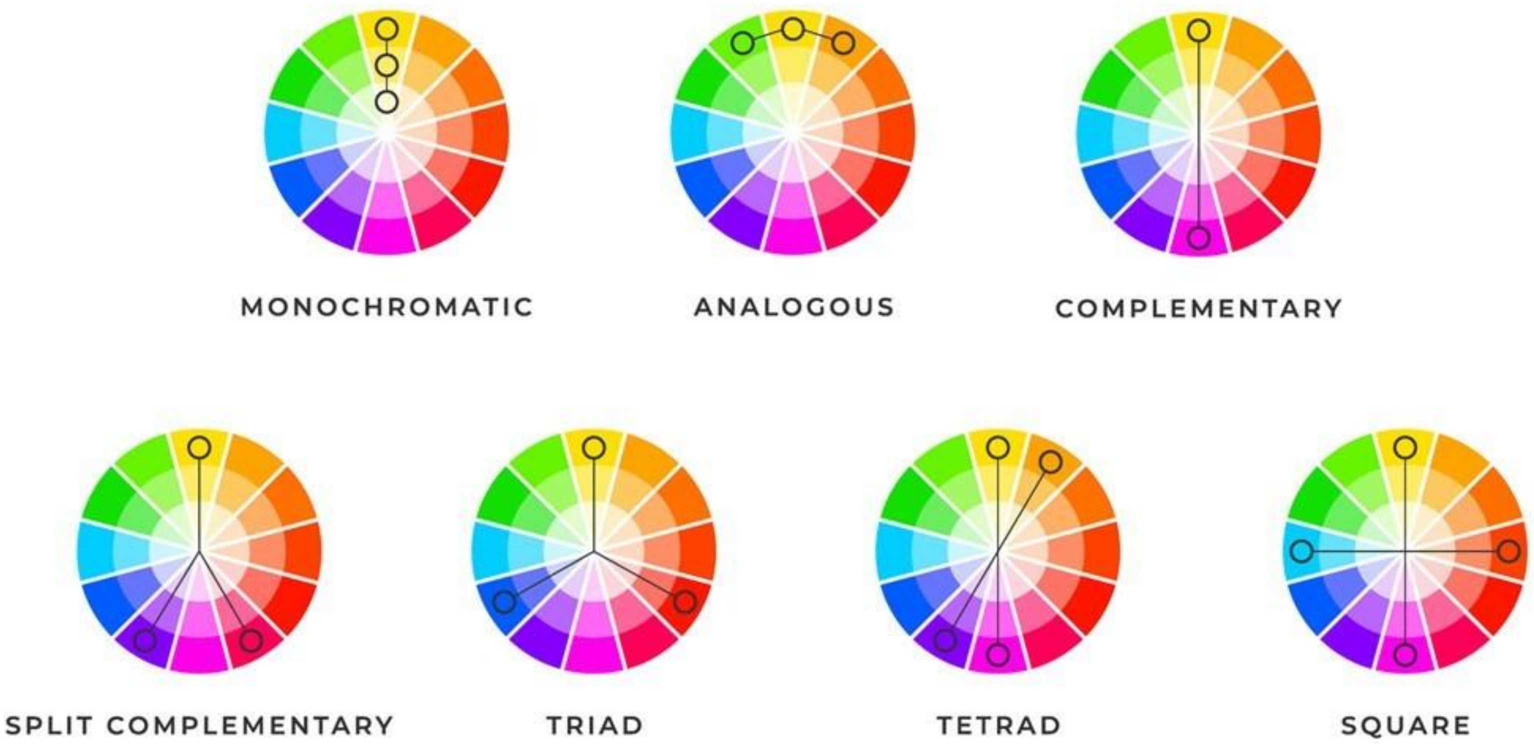
- Built-in ggplot themes: https://ggplot2.tidyverse.org/reference/ggtheme.html
- Or simply google “custom ggplot themes”

Custom `ggplot` themes with `theme()`



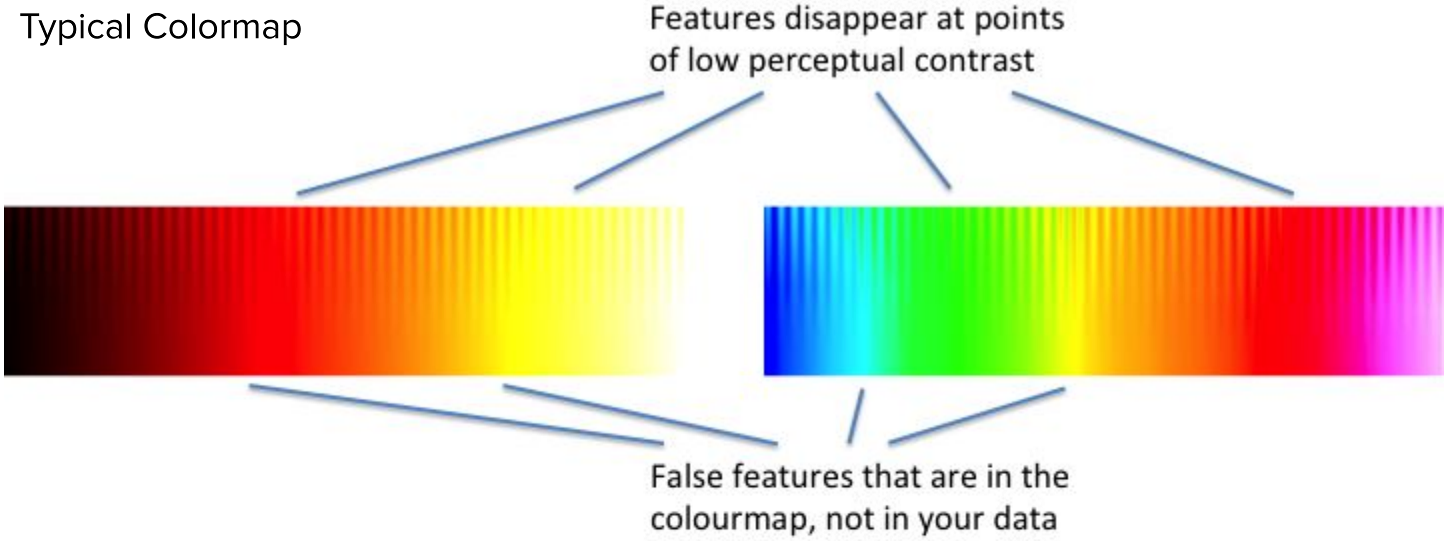


Color schemes



Color choice can lead to misleading visualizations

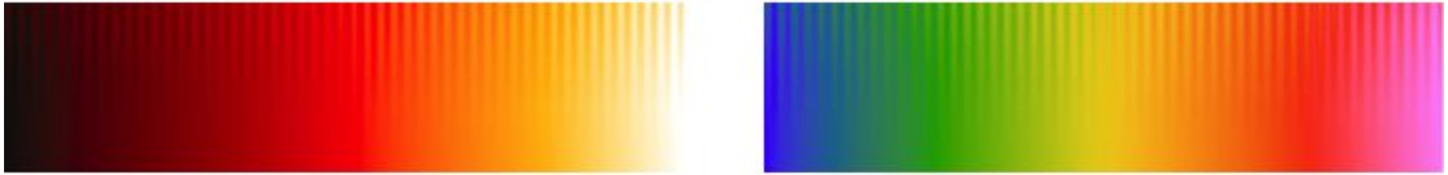
Typical Colormap



Features disappear at points of low perceptual contrast

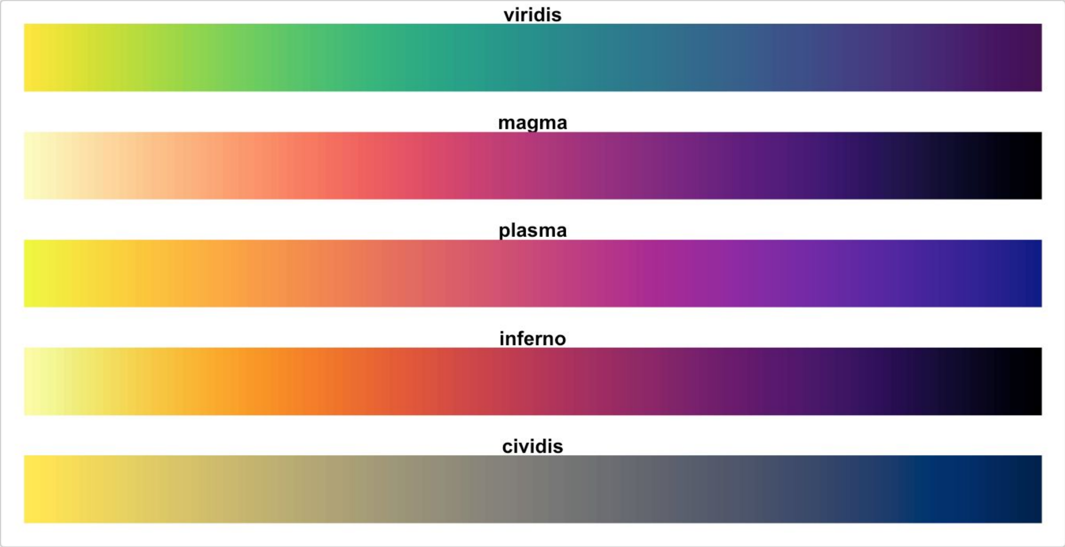
False features that are in the colourmap, not in your data

Perceptually Uniform Colormap

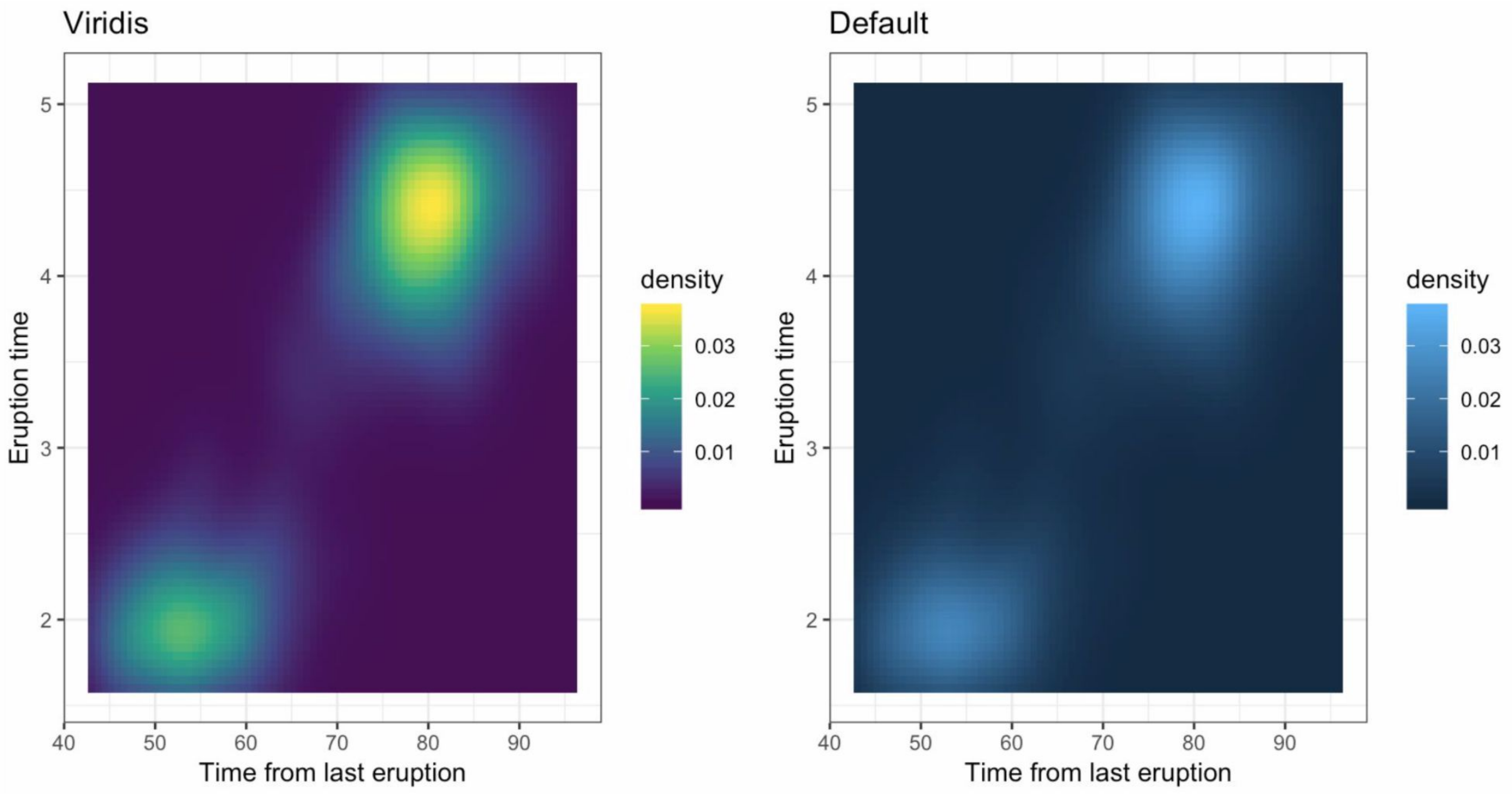


Viridis color scheme

- Makes pretty plots!
- Perceptually uniform colors (meaning changes in the data should be accurately decoded by our brains)
    - Another colormap with this quality is **RColorBrewer**
- Perceived by most common forms of color blindness



Viridis color scheme



Color schemes

- Default color scheme in base R or ggplot is not always the best choice
- Think about what you are trying to convey in the plot
- Color choices can affect the way we perceive the plot
- Some helpful websites
  - https://coolors.co/app
  - http://colorbrewer2.org/
  - https://color.hailpixel.com/

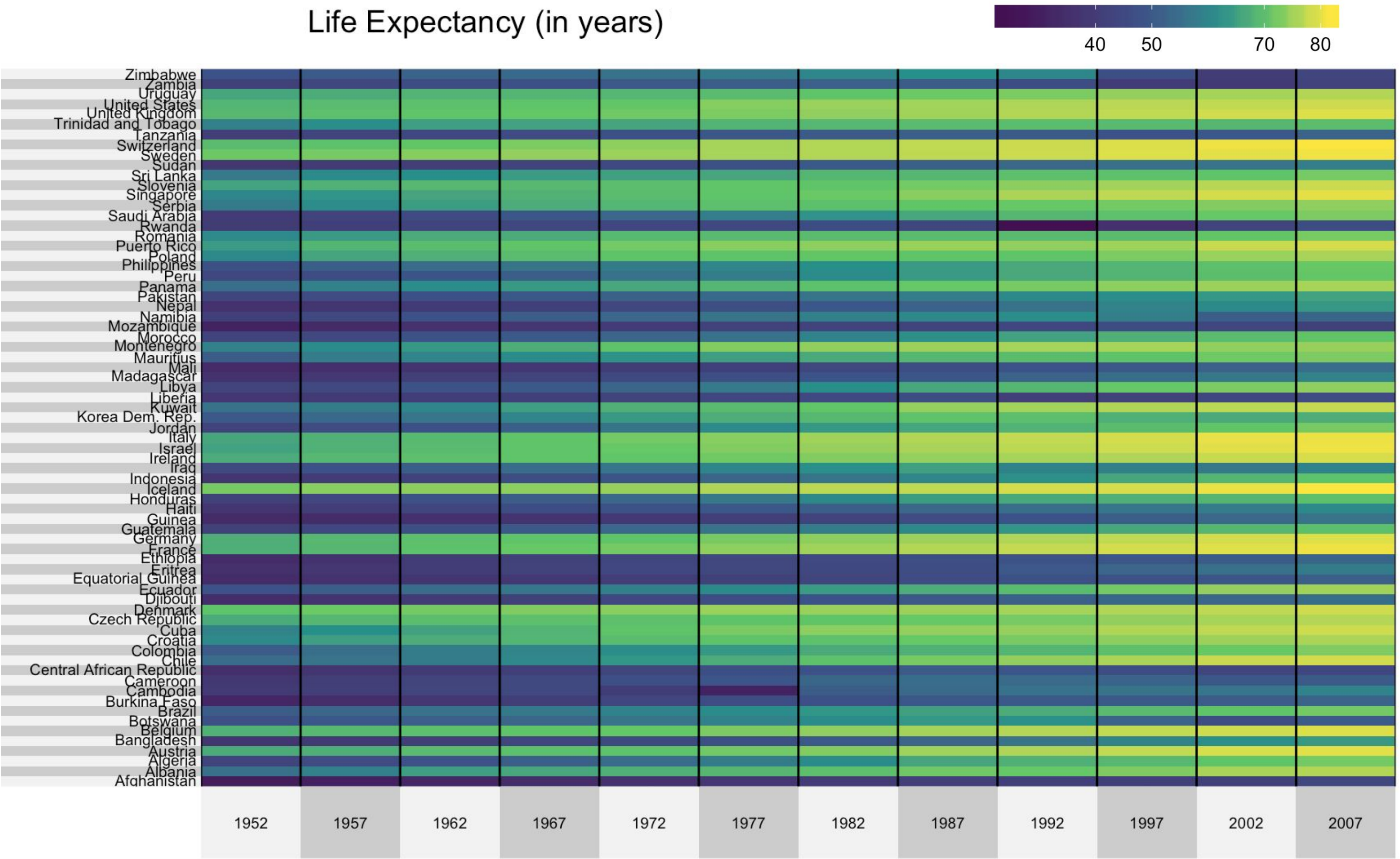
Beyond the world of ggplot...

Heatmaps with `superheat`

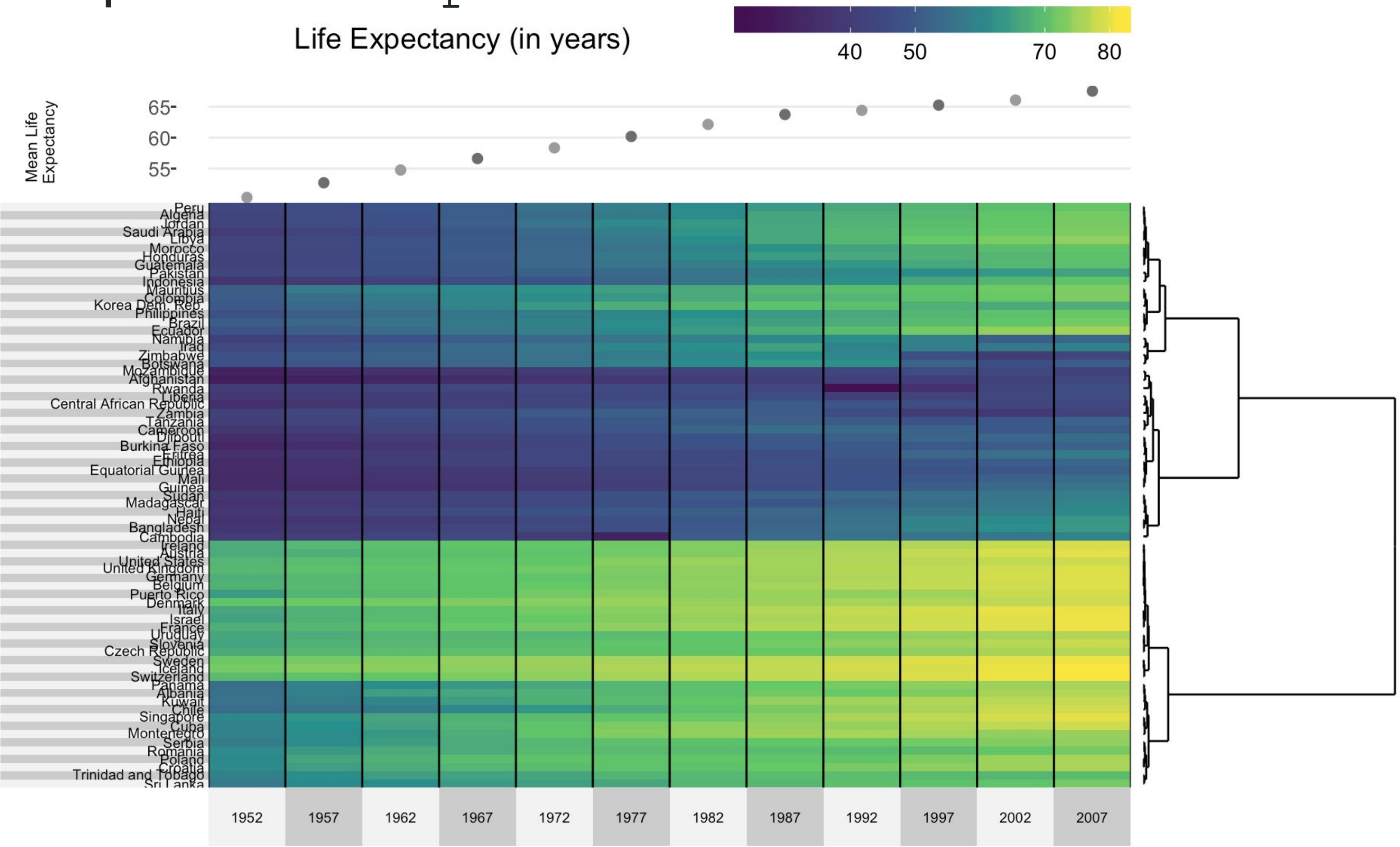
```
For latest development version
install.packages("devtools")

devtools::install_github("rlbarter/superheat")

library(superheat)
```

```
Or for latest stable version
install.packages("superheat")
```

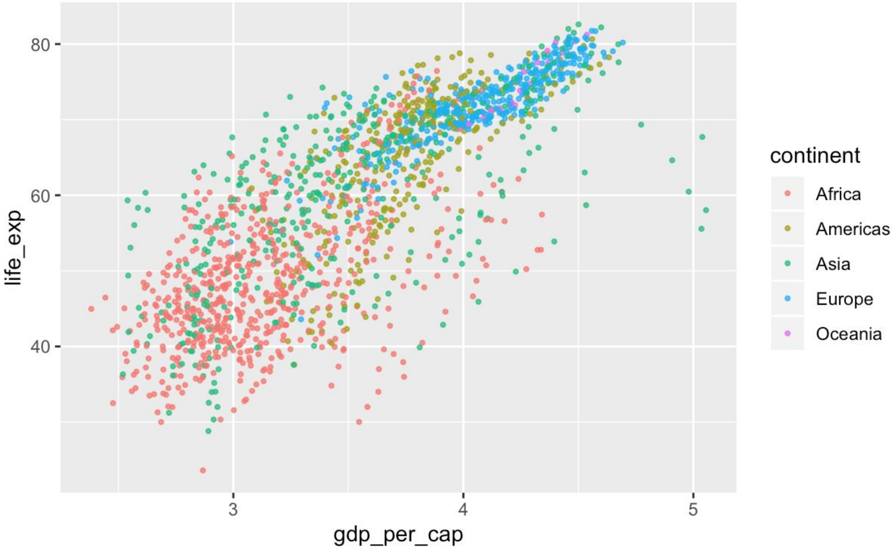
Heatmaps with `superheat`



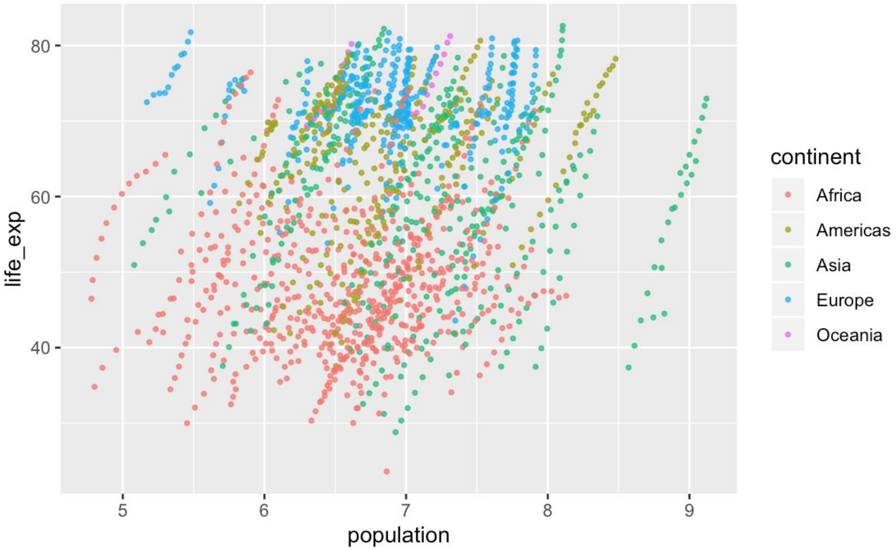
Heatmaps+ with `superheat`



Pair plots

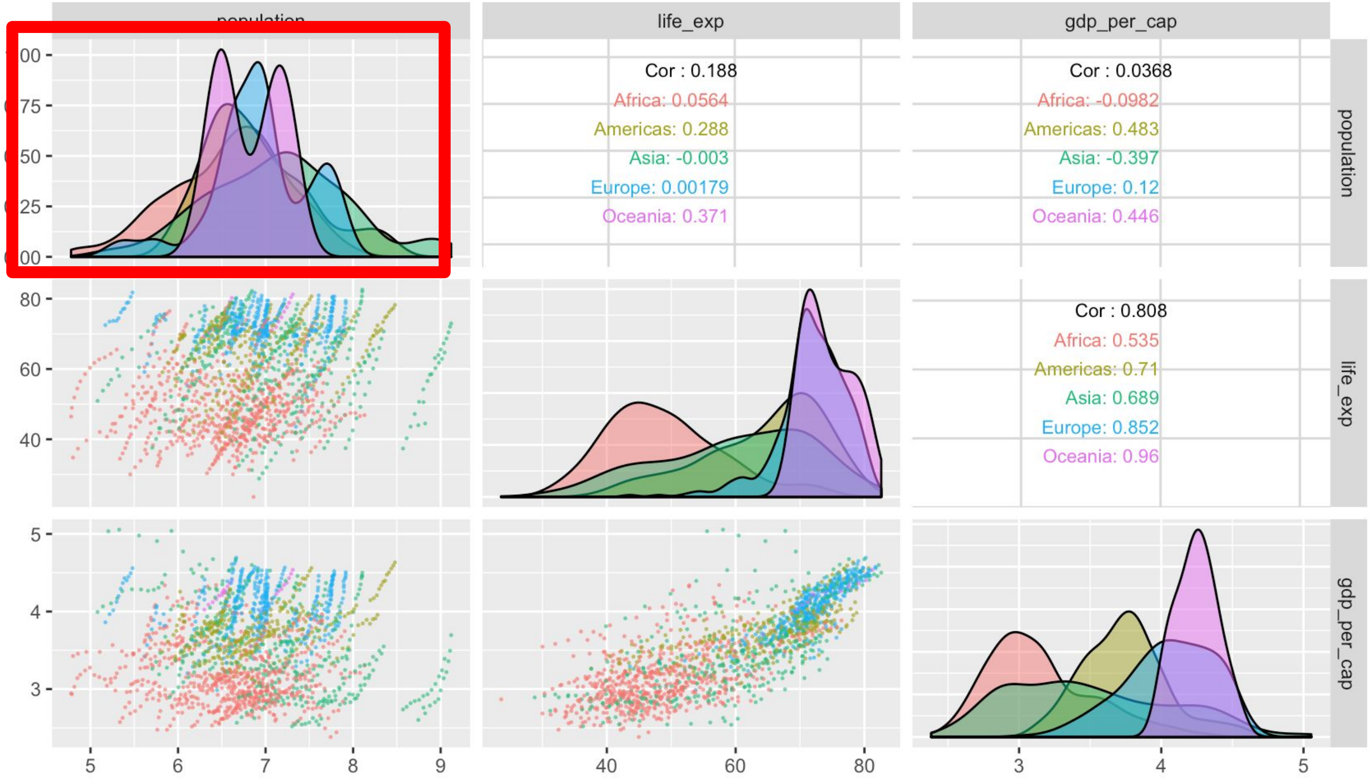


life_exp
80
60
40
3
4
5
gdp_per_cap
continent
● Africa
● Americas
● Asia
● Europe
● Oceania



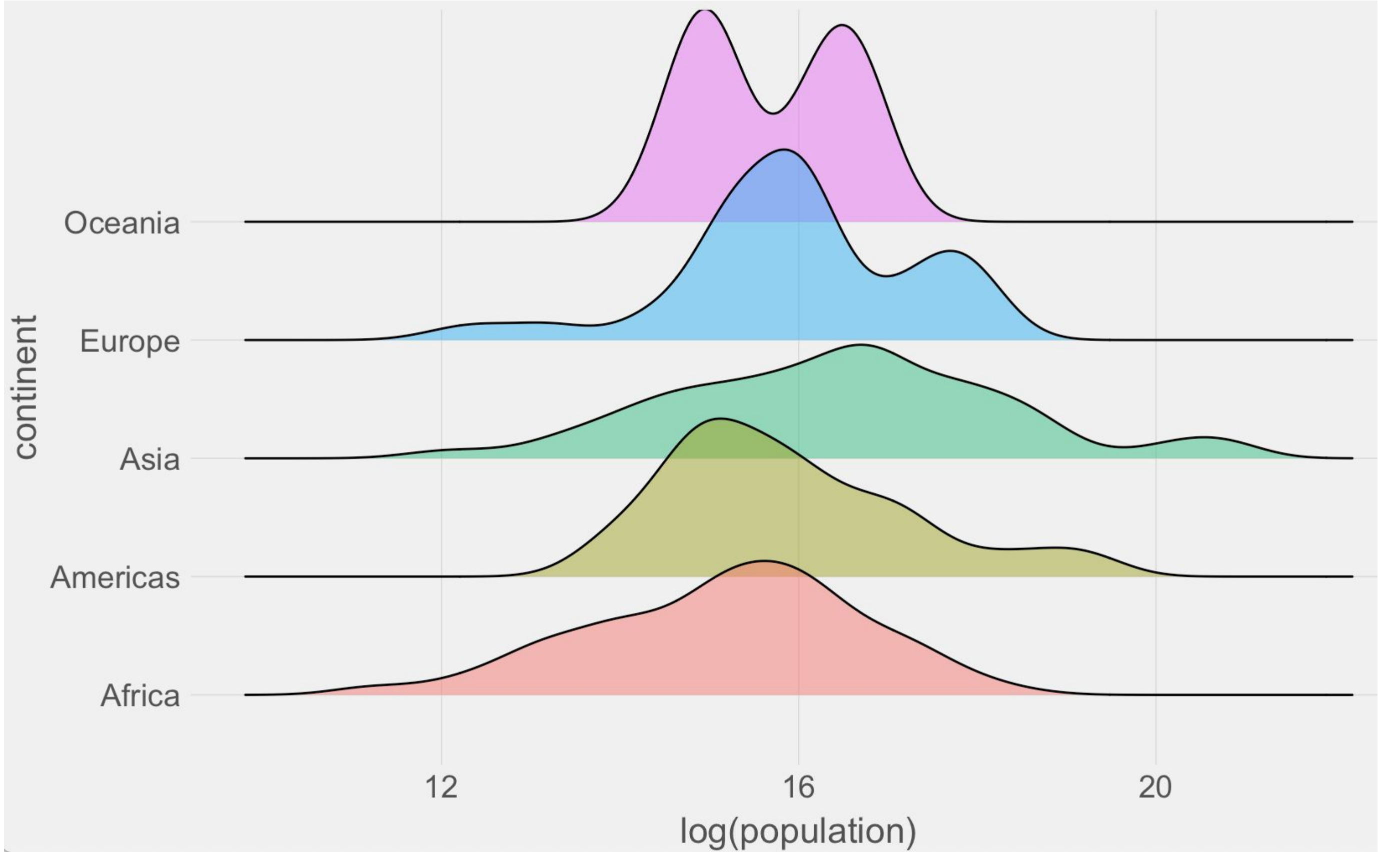
life_exp
80
60
40
5
6
7
8
9
population
continent
● Africa
● Americas
● Asia
● Europe
● Oceania

Pair plots with `GGally::ggpairs`



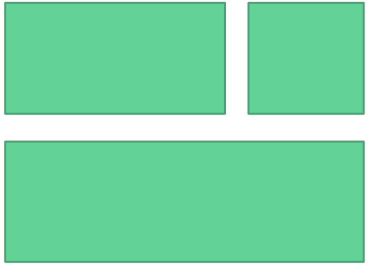
- A word of caution: be wary of over-plotting; consider subsampling points, limiting the number of variables in pair plot, etc.

`ggridges`: another way of viewing multiple densities

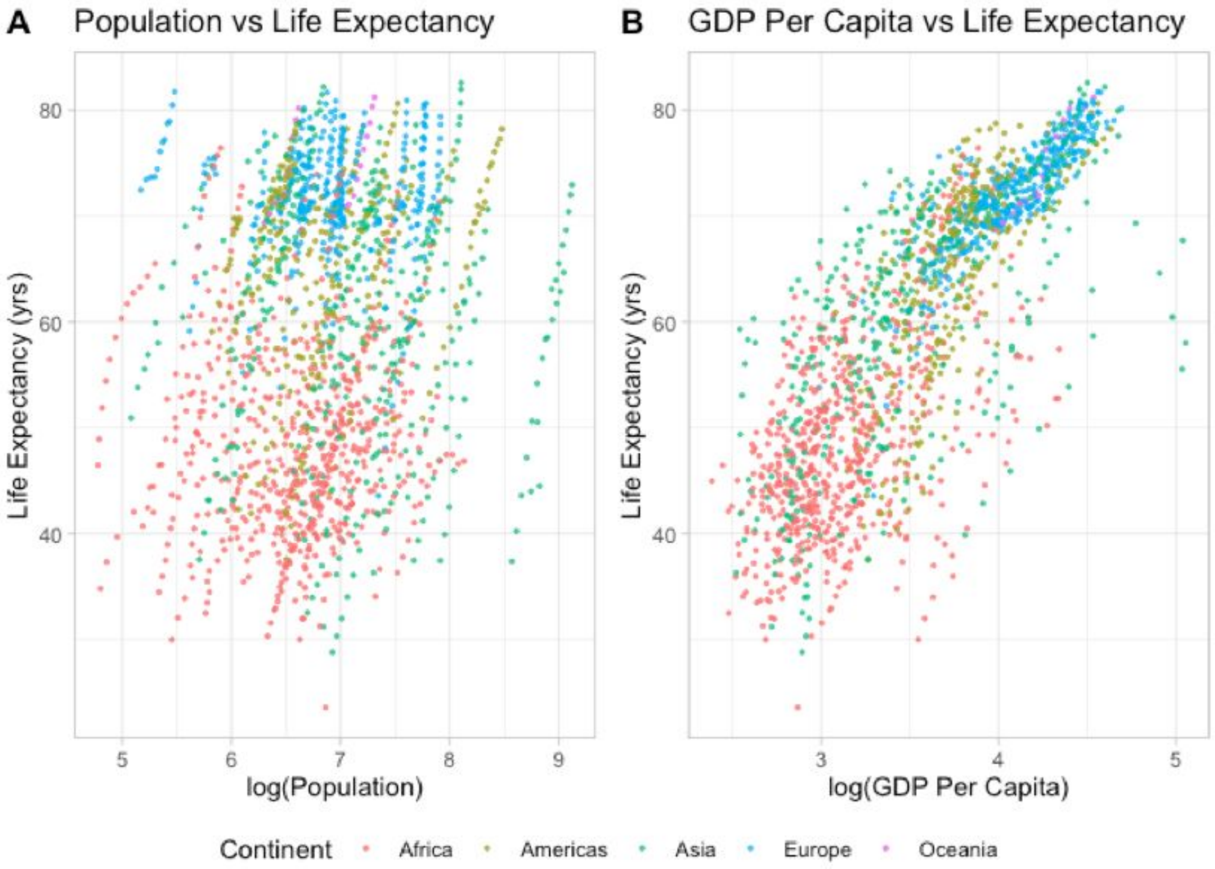


Creating sub-plots

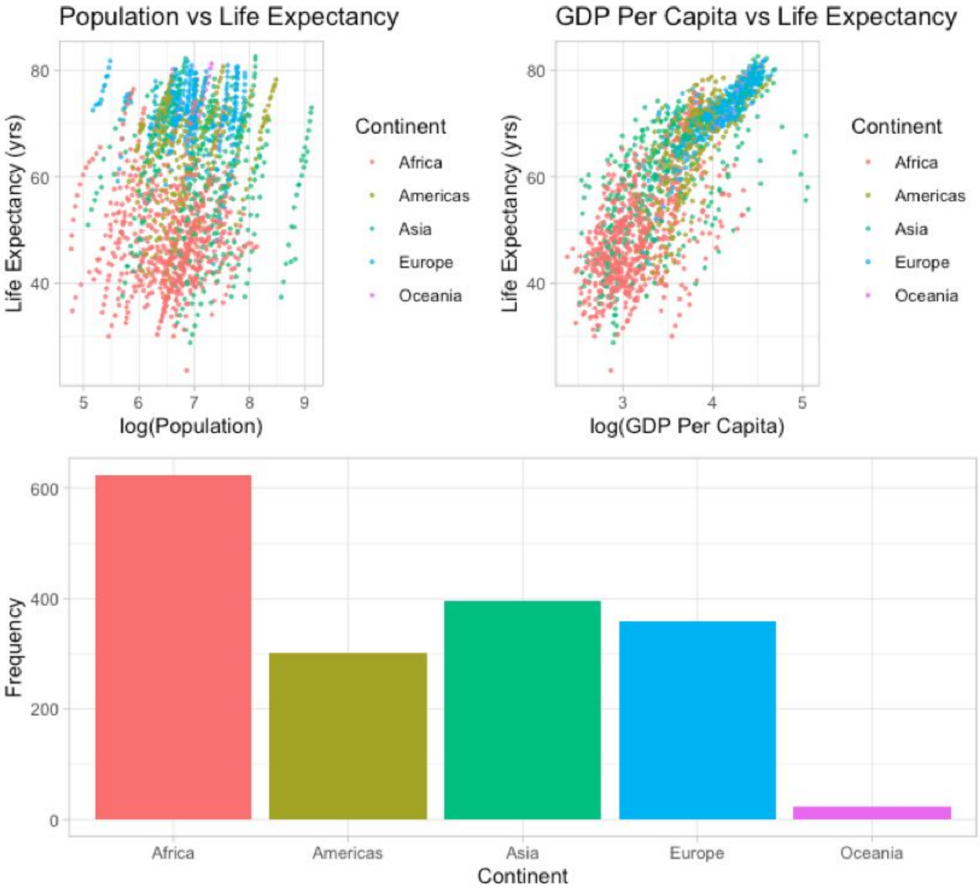
- Two useful functions:
    - `ggpubr::ggarrange()`
    - `gridExtra::grid.arrange()`
- Can easily set a common legend and subplot labels with `ggarrange()`
- `grid.arrange()` is better for fancier "non-matrix" arrangements



ggpubr::ggarrange

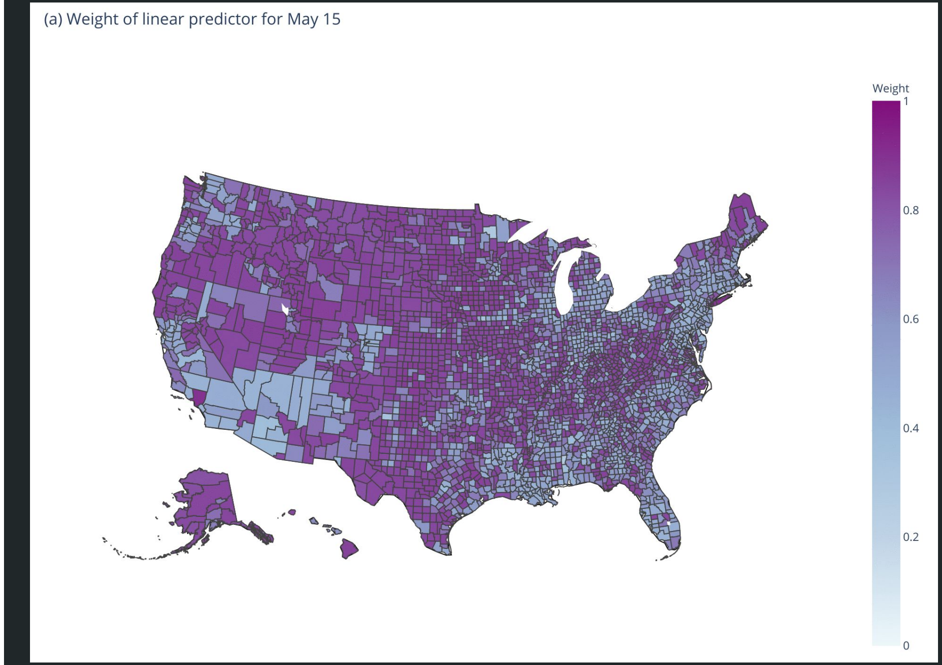


`gridExtra::grid.arrange`



Interactive plots

- Shiny: https://shiny.rstudio.com/gallery
  - Tutorial: https://shiny.rstudio.com/tutorial/
  - Leaftlet
- Plotly: https://plot.ly/r/
- Crosstalk: https://rstudio.github.io/crosstalk/using.html
- Highcharter: https://jkunst.com/highcharter/index.html



(a) Weight of linear predictor for May 15