Visual perception in data graphics

Author: Nicholas G Reich

Visual cues

Graphical elements that draw attention

The choice of which visual cues you use inform the story that you are able to convey and the points you can highlight.

We will go through a set of visual cues. These are loosely adapted from Nathan Yau via the Modern Data Science with R textbook.

Visual cues: position (numerical)

Where are the data in relation to each other?

e.g. points and axis alignment.

Visual cues: Length (numerical)

How big (in one dimension)?

e.g. bars, lines (aligned), lines(unaligned) ...

Visual cues: Angle (numerical)

How wide? Parallel to something else?

e.g. lines, pie wedges, ...

Visual cues: Direction/slope (numerical)

Up or down? At what slope?

e.g. lines, time-series, ...

Visual cues: Shape (categorical)

Belonging to which group?

e.g. points

Visual cues: Area/volume (numerical)

How big (in 2/3 dimensions)?

e.g. circles, squares

Visual cues: Shade/intensity (categorical or numerical)

To what extent? How severely?

e.g. points, lines, ...

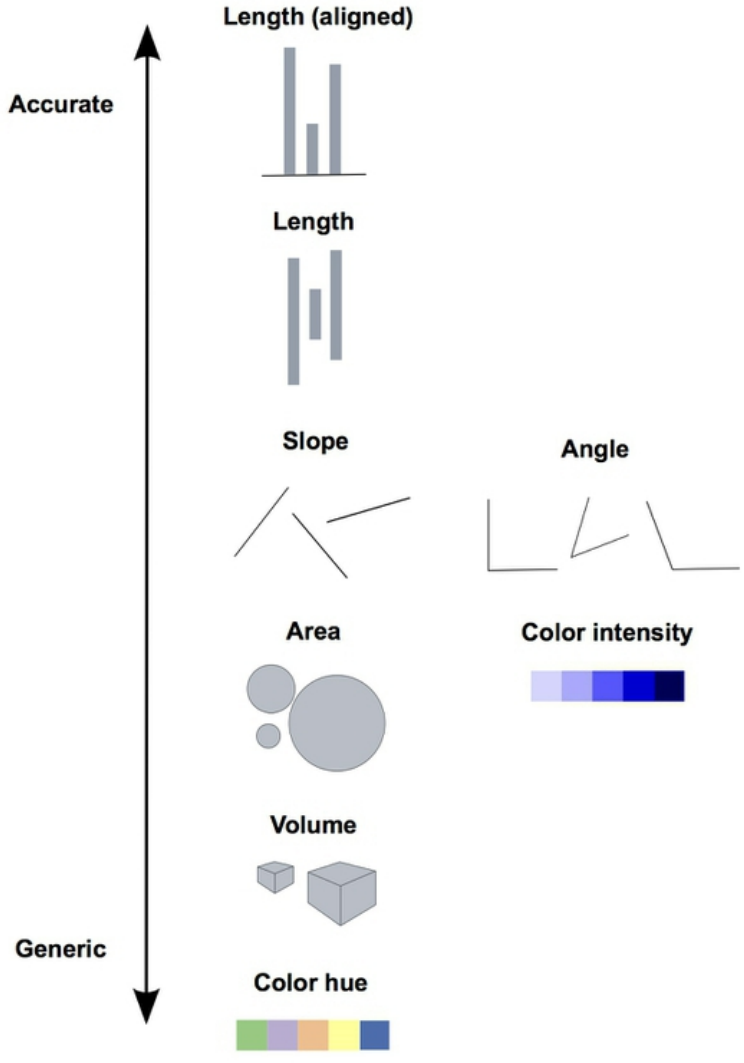
Visual cues: Color (categorical or numerical)

Belonging to which group? To what extent? How severely?

e.g. points, lines, tiles ...

Research on perception of cues

In 1980s, Cleveland and McGill ran experiments to measure accuracy of human perception based on different visual cues.



(Figure credit: Peter Aldhous)

Three main types of color palettes

- sequential: a gradient in one direction
- divergent: a gradient away from a center
- qualitative: categorical groupings

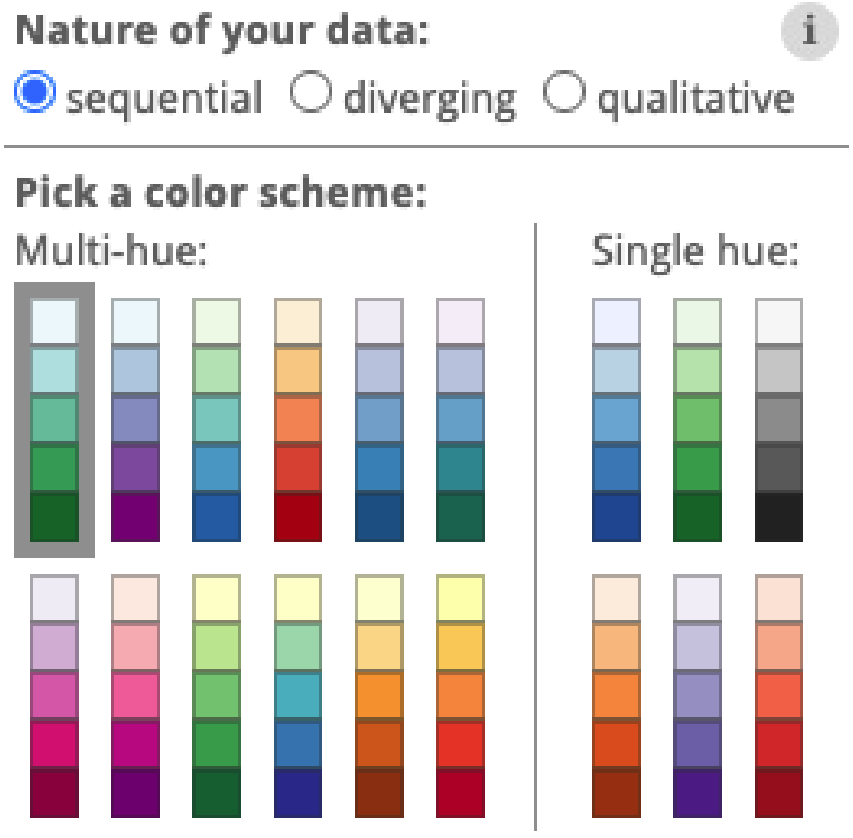
Setting up a small running example

```
library(tidyverse)
gapminder <- read_csv("../../data/gapminder.csv") %>%
  filter(continent == "Asia")
```

Using sequential palettes, with `RColorBrewer`

Use a divergent palette to show a gradient.

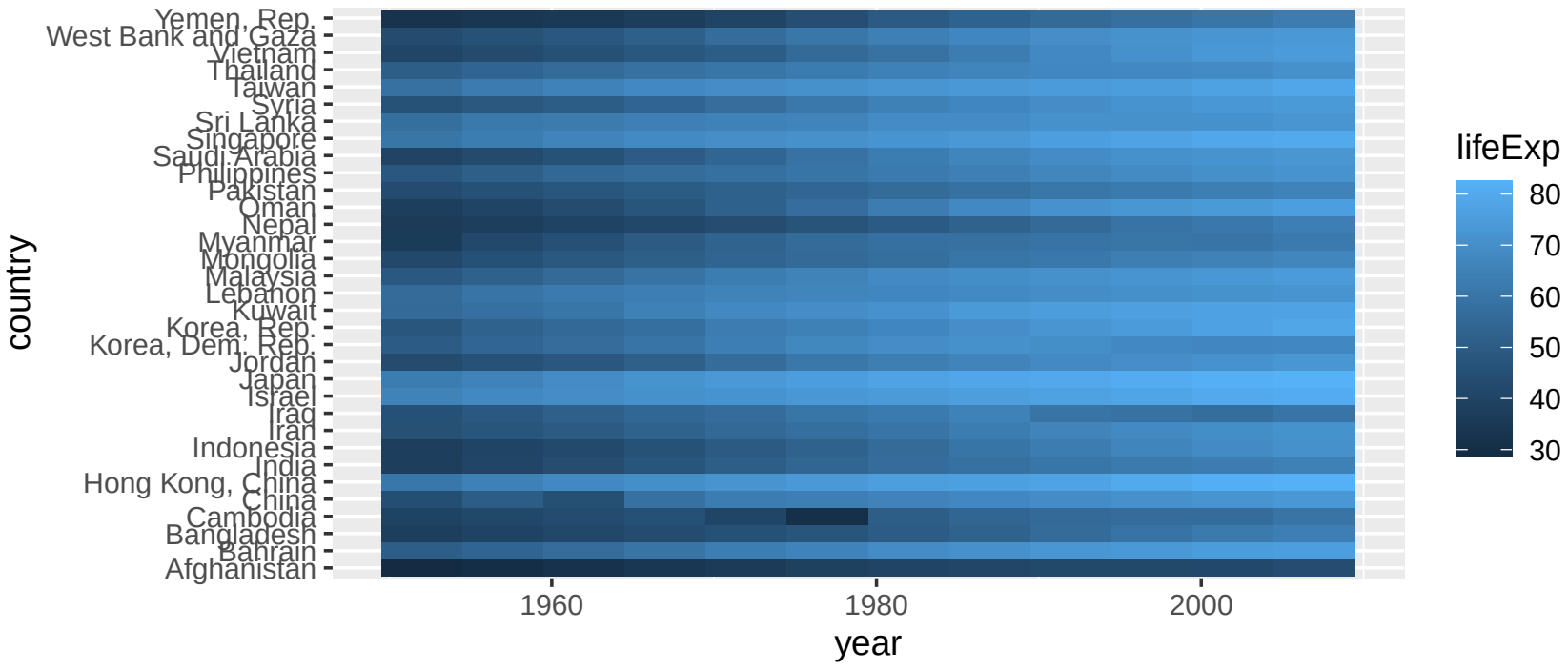
How severe? How high? Possibly an ordered grouping.



Using sequential palettes

Default gradient

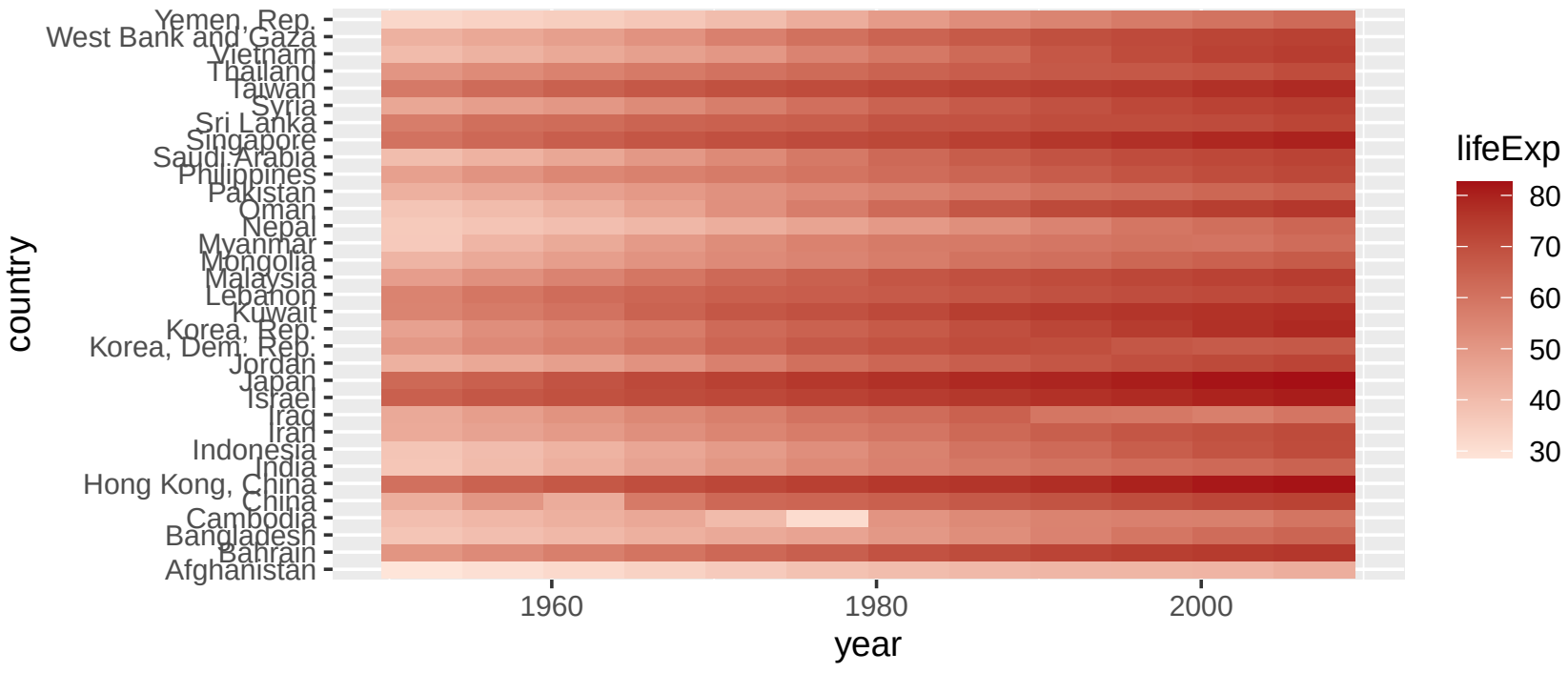
```
ggplot(gapminder, aes(x=year, y=country, fill=lifeExp)) +
  geom_tile()
```



Using sequential palettes, with assist from colorbrewer2.org

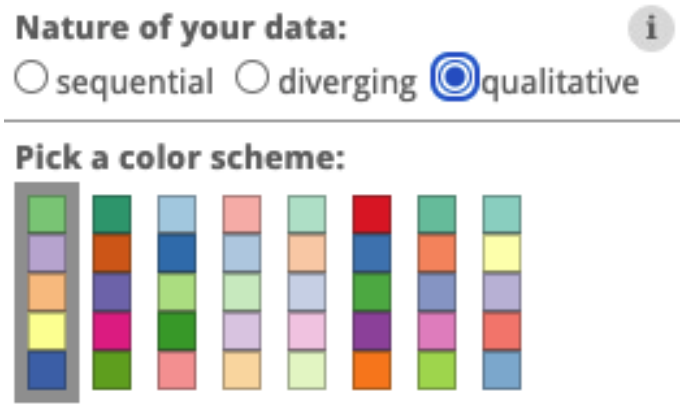
Picking ColorBrewer colors.

```
(pal <- RColorBrewer::brewer.pal(n=5, name="Reds"))


## [1] "#FEE5D9" "#FCAE91" "#FB6A4A" "#DE2D26" "#A50F15"


ggplot(gapminder, aes(x=year, y=country, fill=lifeExp)) +
    geom_tile() +
    scale_fill_gradient(low=pal[1], high=pal[5])
```

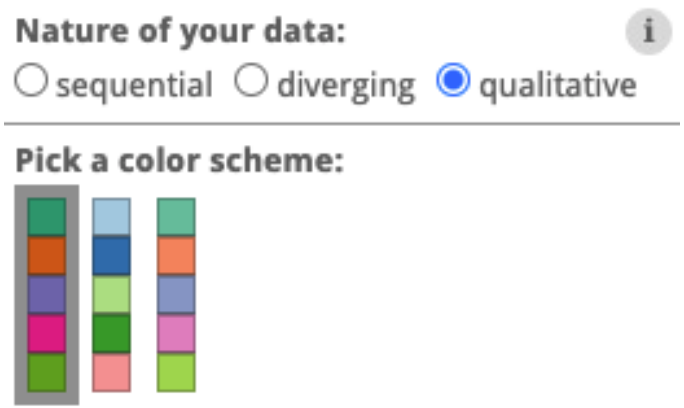


Using qualitative palettes to show groupings[1]

All palettes:



Only color-blind friendly ones:

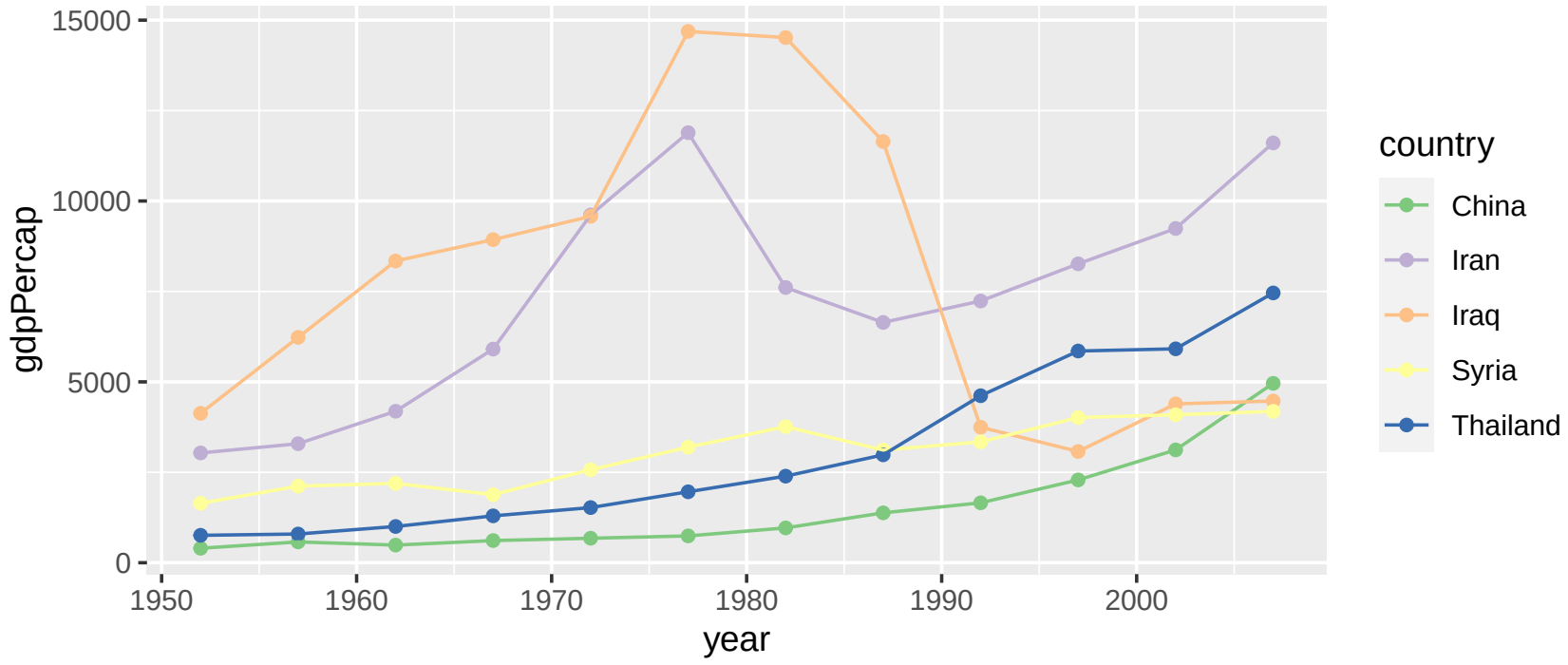


[1] colorbrewer2.org

Using qualitative palettes from `RColorBrewer`

For low-number palettes (usually less than 12 colors) you can access the palettes directly.

```
gapminder %>%
  filter(country %in% c("Syria", "Iraq", "Iran", "China", "Thailand")) %>%
ggplot(aes(x=year, y=gdpPercap, color=country)) +
    geom_point() + geom_line() +
    scale_color_brewer(type = "qual")
```



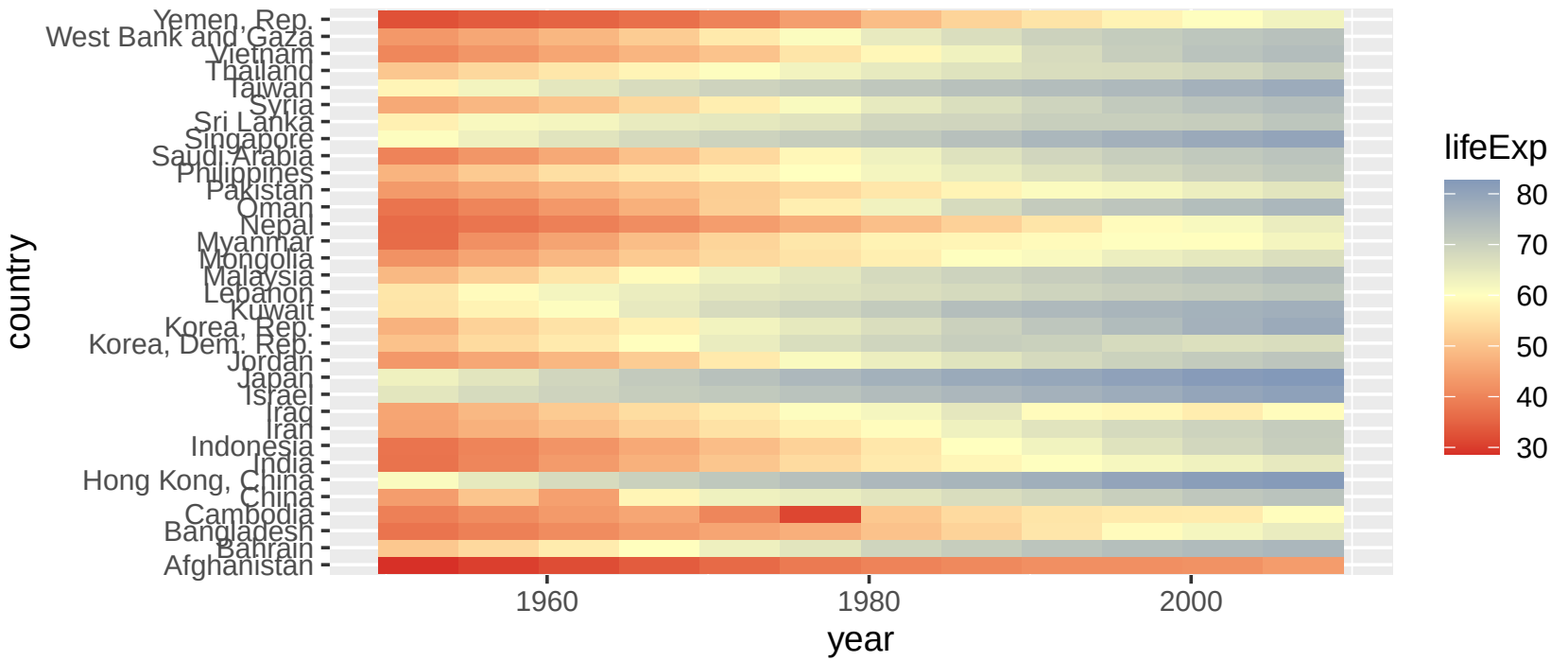
Using divergent palettes to compare values to a reference level

This is not a great example, because the scale is not naturally bi-directional.

```
pal <- RColorBrewer::brewer.pal(n=9, name="RdYlBu")
(mean_lifeExp <- mean(gapminder$lifeExp))
```

```
## [1] 60.0649
```

```
ggplot(gapminder, aes(x=year, y=country, fill=lifeExp)) +
    geom_tile() +
    scale_fill_gradient2(low=pal[1], mid=pal[5], high=pal[9], midpoint = mean_lifeExp)
```



Using divergent scale to compare model accuracy

An example from my research. Each cell represents the relative accuracy of the given model (column) to a baseline model for a location (row). Blue = more accurate than baseline. Red = less.