Data visualization with ggplot2

Stefan Hartmann
HHU Düsseldorf

Plan for today

Heinrich Heine
Universität Düsseldorf

- Basics of data visualization
- Basics of ggplot2
- Example graphs for basic plot types
  - Scatterplot
  - Line plot
  - Boxplot, violin plot
  - Beeswarm plot
- Hands-on exercises
- Some (fairly random but perhaps useful) tips and tricks

Prerequisites

Heinrich Heine
Universität Düsseldorf

Software and packages

- Please make sure that you have R and RStudio installed.
- You also need the following packages / families of packages:

```
tidyverse
patchwork
scales
ggbeeswarm
```

- If they are not installed yet, you can install them in R using the `install.packages()` command, e.g.

```
install.packages("tidyverse")
```

Principles of data visualization

Data visualization

Heinrich Heine
Universität Düsseldorf

Why visualize?

- **For yourself**
  - Exploring your data
  - detecting outliers
  - checking assumptions of statistical tests or models (e.g. are the data normally distributed?)
  - etc.
- **For others**
  - Showing your findings in a clear and efficient way
  - Graphs tend to be more reader-friendly than tables...
  - and *much* more reader-friendly than long inline lists!

Data visualization

Choosing the "right" plot

- What kind of data are you dealing with?
- What is your research question?
- What kind of audience are you expecting?

Best practice for reporting & displaying data

Heinrich Heine
Universität Düsseldorf

- Most importantly: **Know your data!**
- When reporting percentages, also report the denominator (i.e. the size of your sample)
- Example: "50% of academics are alcoholics" - it makes a difference whether your sample size is 2 or 2,000.
- When reporting comparisons of absolute frequencies, double-check if your samples are comparable.
- Example: "255 women agree that cats are adorable, but only 5 men." - it makes a difference whether your sample consists of 300 women and 300 men or of 300 women and 10 men.
- When reporting means, also report dispersion measures (e.g. standard deviations or standard errors)

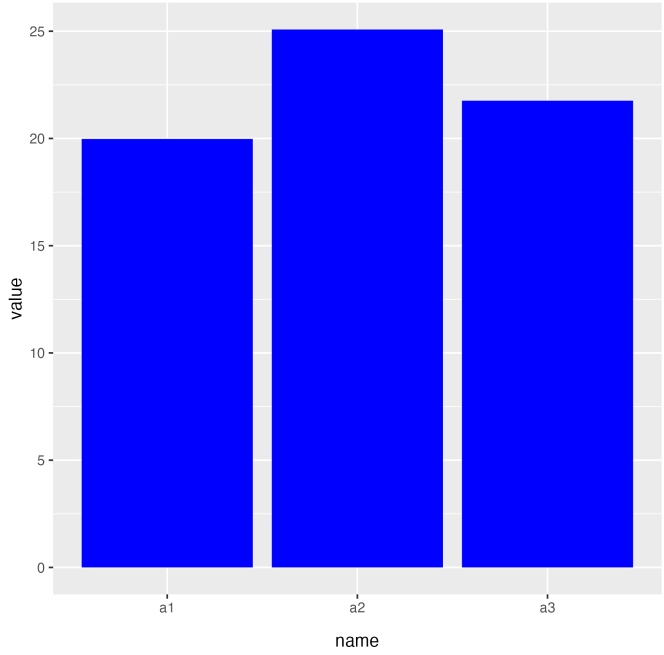
Best practices

Heinrich Heine
Universität Düsseldorf

- Show the data
- Avoid distorting the data
- Aim for a good "ink-to-data ratio": Display as much information as possible with as little ink as possible
- Avoid overplotting (e.g. 3-dimensional plots when only 2 dimensions are displayed)
- Use meaningful x and y labels

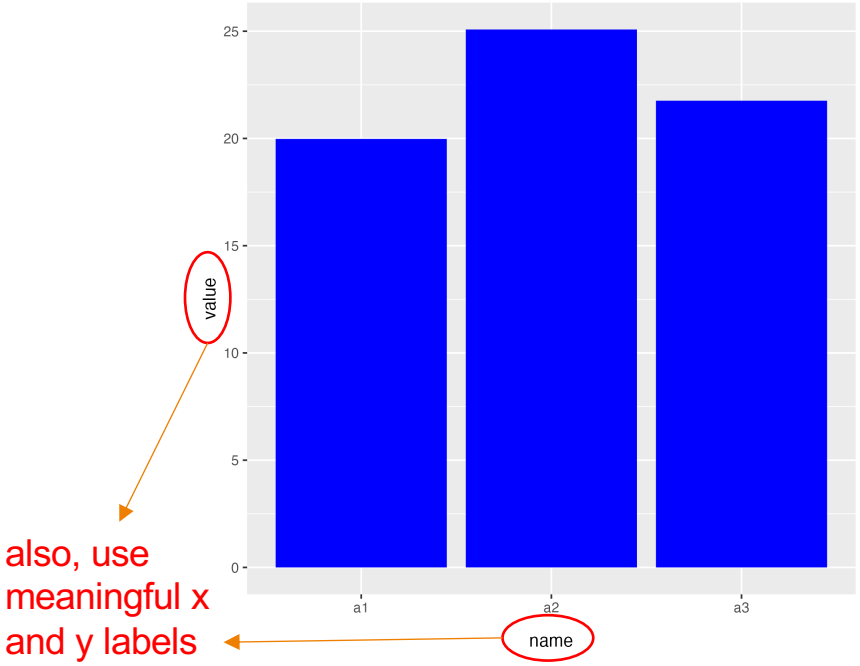
Best practices

Show the data

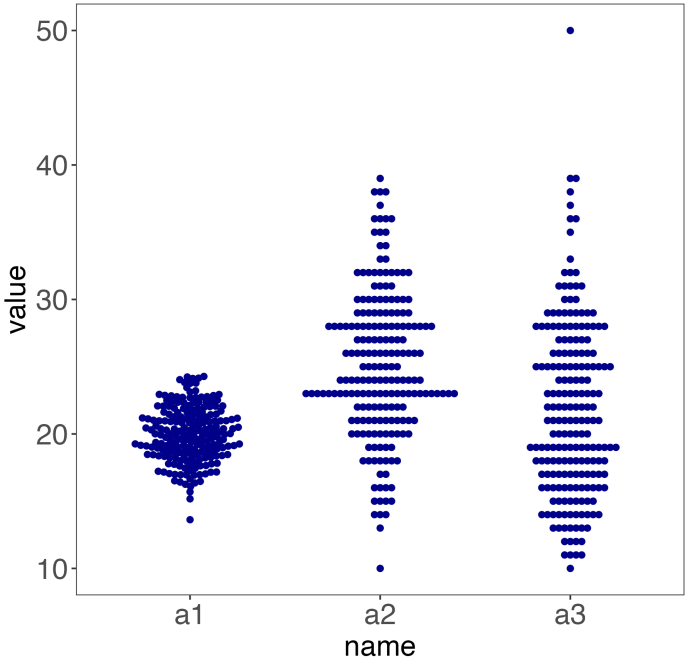


Best practices

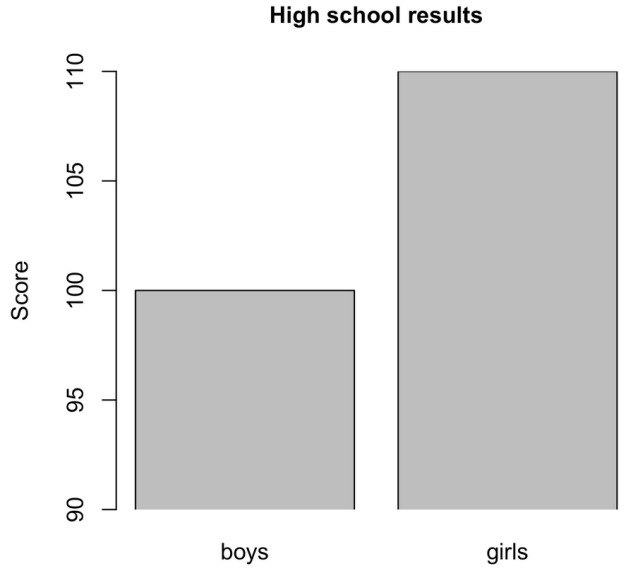
Show the data



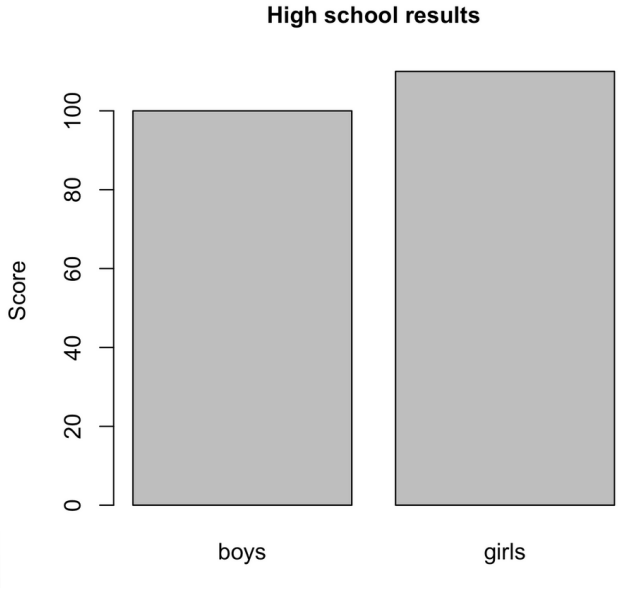
also, use meaningful x and y labels

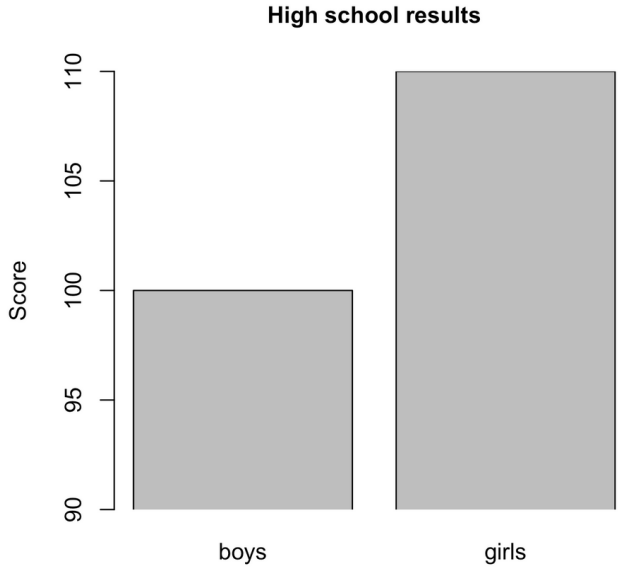


Avoid distorting the data



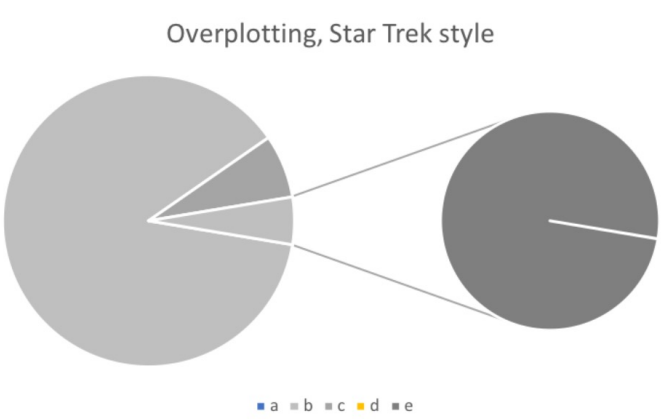
Avoid distorting the data

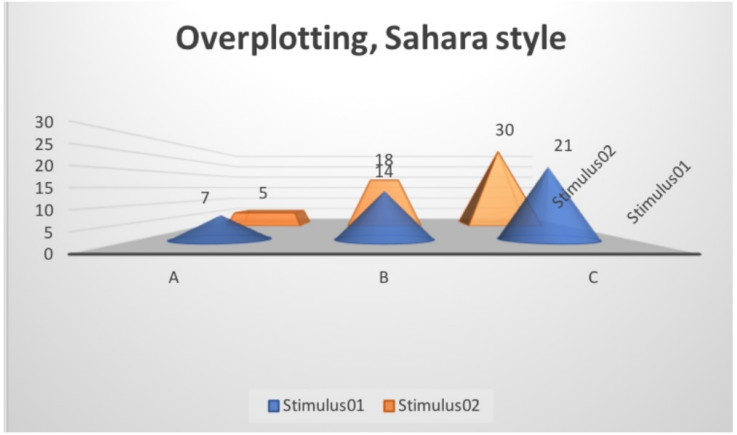


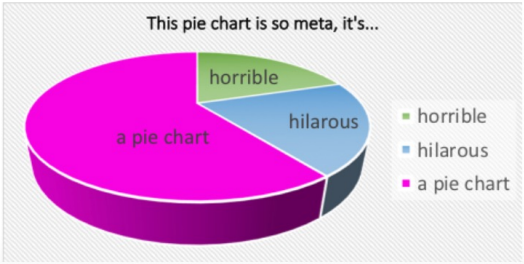


"Ink-to-data ratio" / Overplotting

Heinrich Heine
Universität Düsseldorf

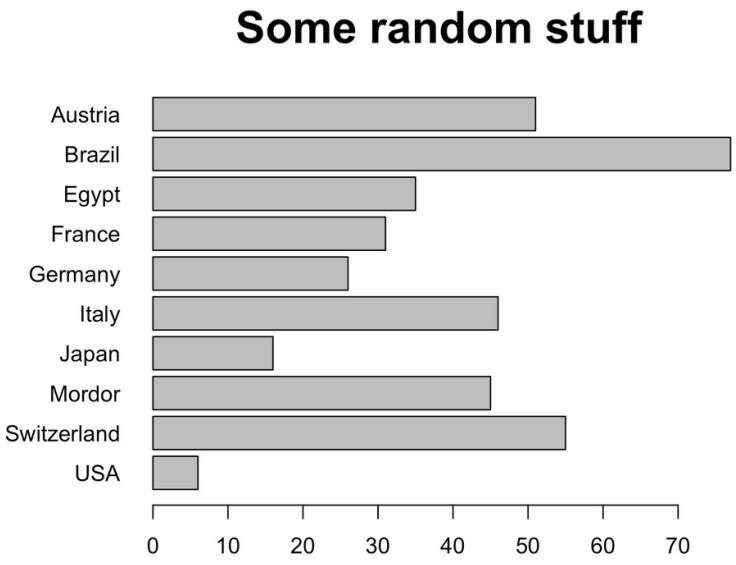


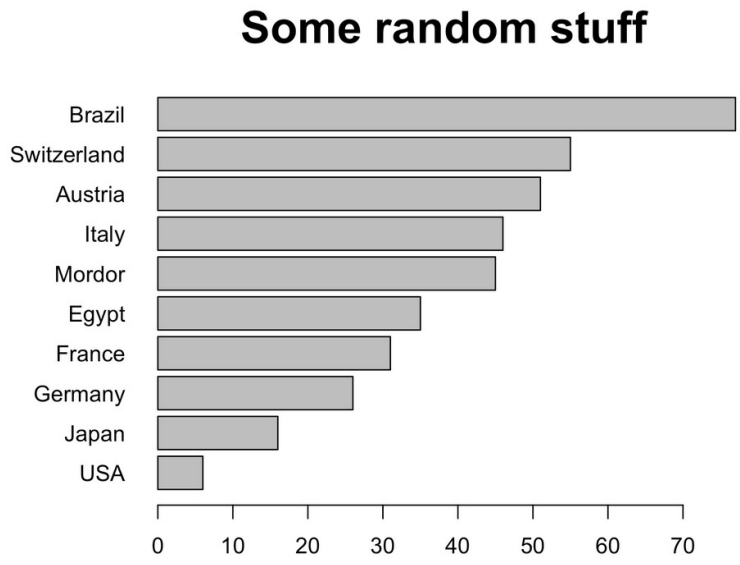




Further tips

- If there is no natural order to your data, order them by value

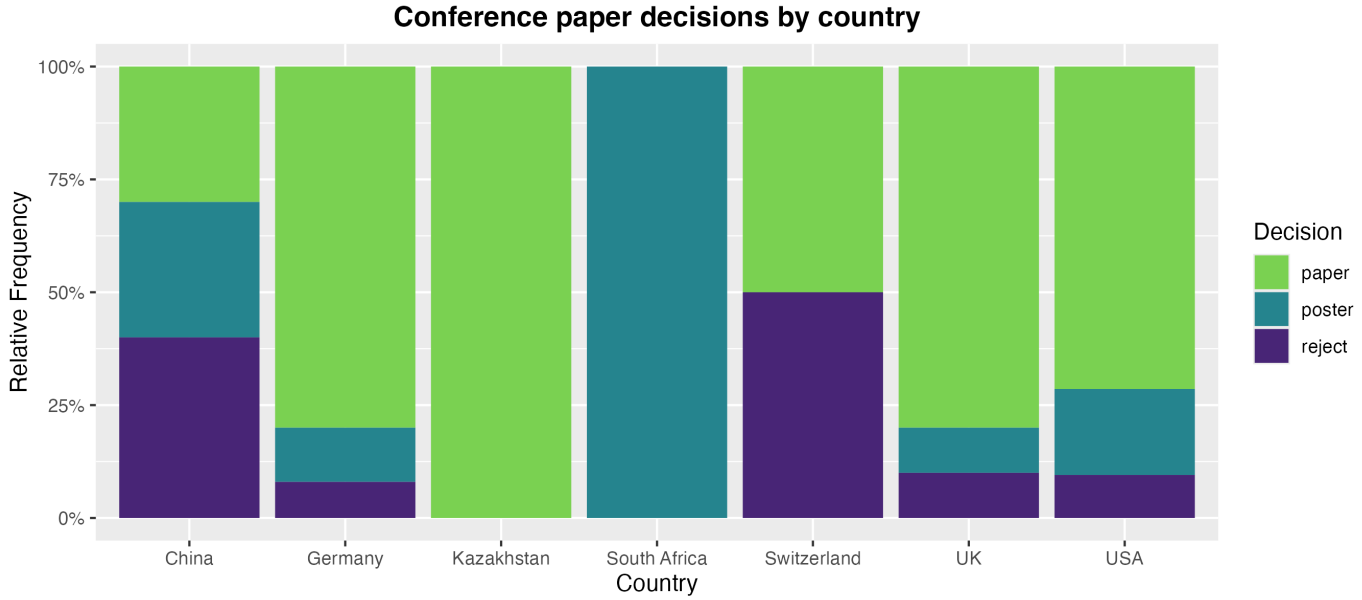




(Freeman et al. 2008: 4)

Further tips

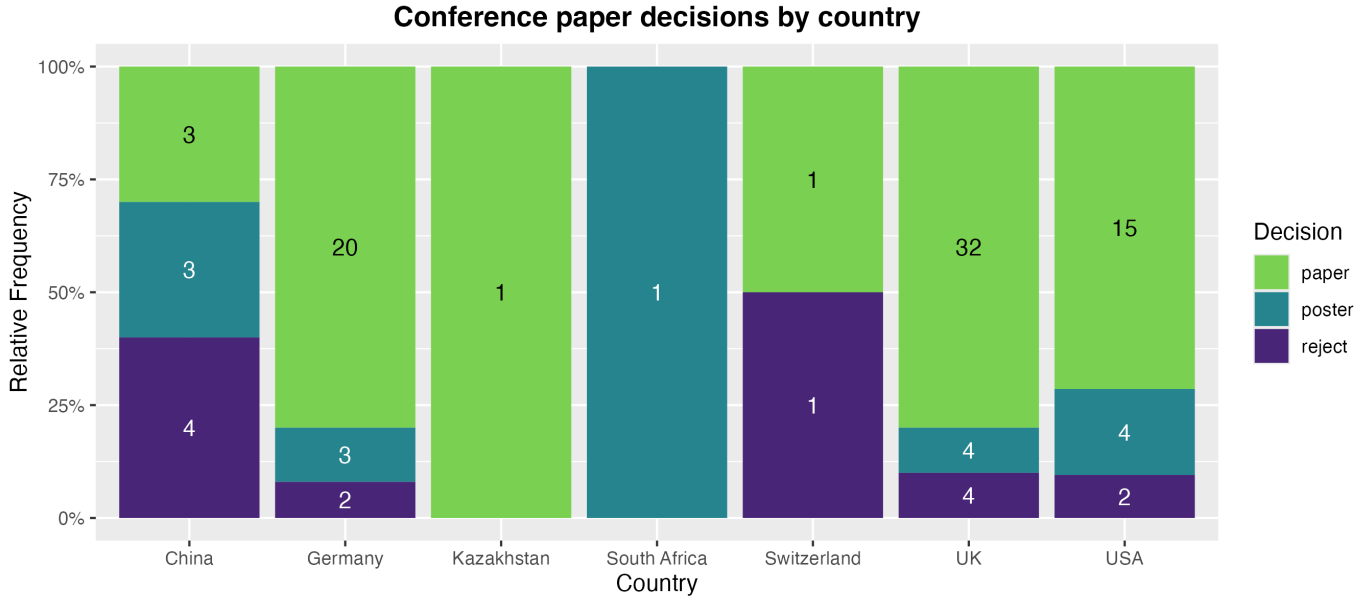
- Display categorical data as frequencies **and** percentages, making sure that the number of observations is included.



(Freeman et al. 2008: 4)

Further tips

- Display categorical data as frequencies **and** percentages, making sure that the number of observations is included.

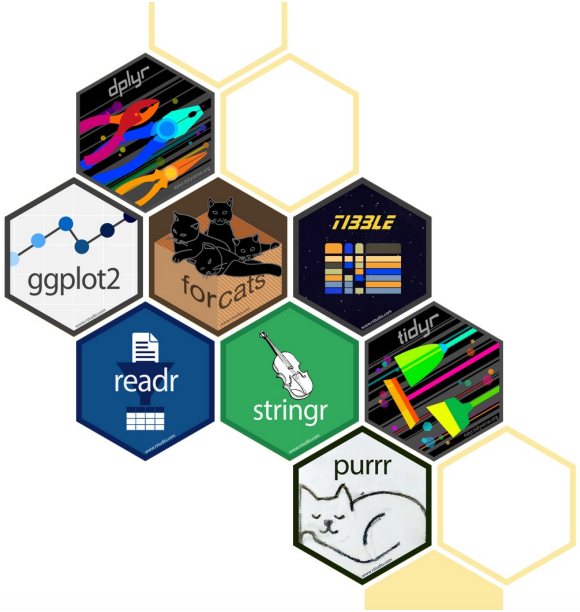


(Freeman et al. 2008: 4)

Basics of ggplot

Base R and Tidyverse

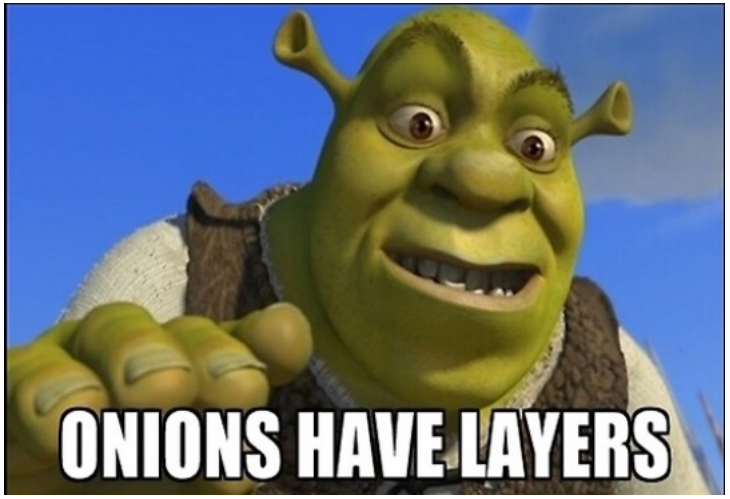
What is the Tidyverse?



- family of packages developed by RStudio/Posit
- implement an own "dialect" of R
- still fully compatible with base R, but adding more syntax possibilities

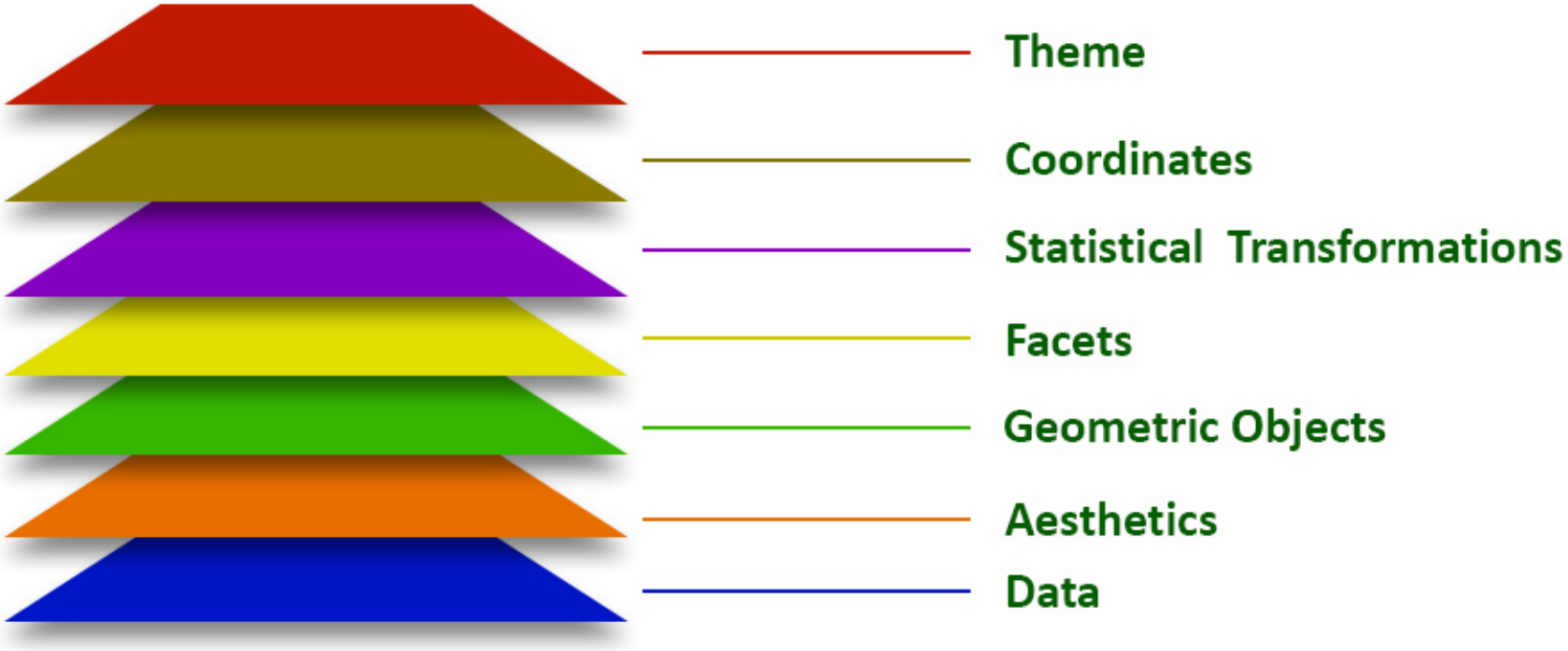
The syntax of ggplot

- A ggplot consists of three components
  - the **data,**
  - a set of **aesthetic mappings**,
  - at least one **layer** (usually created with the geom function) describing how to render each observation.



(Wickham et al. 2023)

Main Components of the Grammar of Graphics



Creating a ggplot

- A ggplot is built **layer by layer**
- We start out with the **data** and the **aesthetic mappings**
- Basic syntax:

```
p <- ggplot(data, aes(x = ..., y = ..., group = ...))
```

- We specify the **geometric objects** to plot, e.g.

```
p <- p + geom_line() # lineplot
```

- Optional: We customize the **scales** (position, color, size) and/or change the **theme** of the plot

```
p <- p + scale_color_grey() + theme_minimal()
```

A basic ggplot

- **First step: generating fake data**

Try to create a dataframe with two columns x and y, with x containing the numbers from 1 to 100 and y 100 normally-distributed random numbers (`rnorm(100)`).

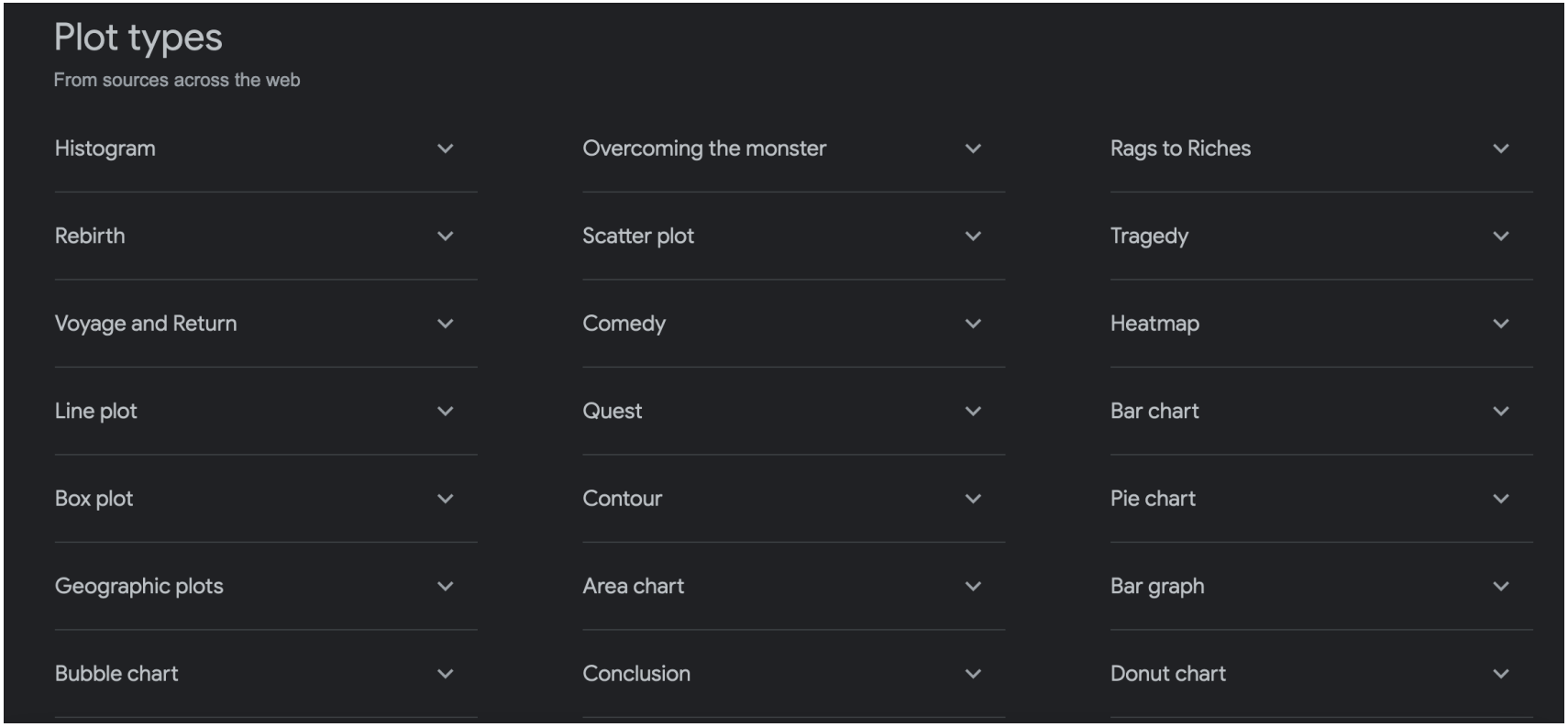
- **Second step: visualizing the data**

Plot x against y using base R first and then using ggplot.

- **Third step: customizing the plot**

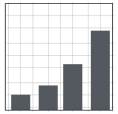
Play around with different scale configurations and themes.

Plot types

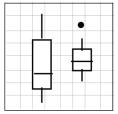

Plot types
From sources across the web
Histogram
Rebirth
Voyage and Return
Line plot
Box plot
Geographic plots
Bubble chart
Overcoming the monster
Scatter plot
Comedy
Quest
Contour
Area chart
Conclusion
Rags to Riches
Tragedy
Heatmap
Bar chart
Pie chart
Bar graph
Donut chart

Different Geoms (Plot Type) in ggplot2

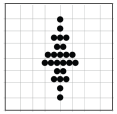
Two Variables (X,Y)
- Discrete X, continuous Y
- Visualise distribution of Y with respect to X



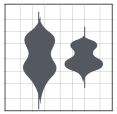
geom_col()
- heights of bars represent values



geom_boxplot()
- summarise distribution using median, hinges and whiskers

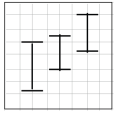


geom_jitter()
- adds jitter to prevent overplotting

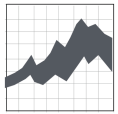


geom_violin()
- mirrored density plot (smoothed distribution)

Visualising Errors and Uncertainties

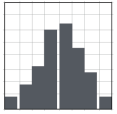


geom_errorbar()
- uncertainty in continuous Y against discrete X

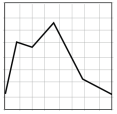


geom_ribbon()
- uncertainty in continuous Y against continuous X

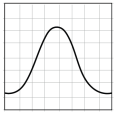
One Variable (X)
- Continuous X
- Visualise distribution of X



geom_histogram()
- divide X into bins and count no. observation

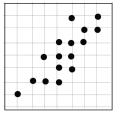


geom_freqpoly()
- display counts with lines
- able to overlay multiple distributions

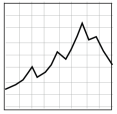


geom_density()
- smoothed version of the histogram

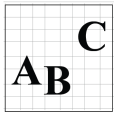
Two Variables (X,Y)
- Continuous X, continuous Y
- Visualise relationship between X and Y



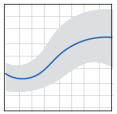
geom_point()
- scatterplot of X vs Y



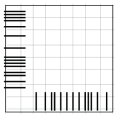
geom_line()
- connect data points, ordered by X
- alt: geom_path()



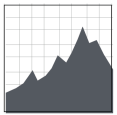
geom_text()
- labelling data points



geom_smooth()
- add smoothed curve
- helps to see trends



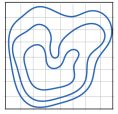
geom_rug()
- supplement 2D plot with 1D distribution along X and Y



geom_area()
- can be stacked to see cumulative contribution

Contour Plots
- Representing a third dimension using contours

geom_density2d()
- contour represents 2D density of data points



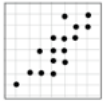
geom_contour()
- contour represents z-axis value / height

Heinrich Heine Universität Düsseldorf

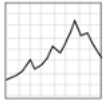
Plot types

Which plot types for which purpose?

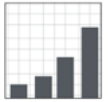
- **Scatterplots**
  - show / explore correlations between two variables
  - metric data on both the x- and the y-axis



- **Line plots**
  - Lineplots are useful for showing e.g. change over time
  - Count variable on y-axis, (at least) ordinal variable on x-axis



- **Barplots**
  - useful to show counts of categorical variables (e.g. number of men vs. number of women in parliament)...
  - summary statistics (usually: means) of metric variables across different categories (e.g. mean height of humans vs. Klingons)



More plot types

Beeswarm plots

- Packages *beeswarm* and *ggbeeswarm*
- can be combined with violin or boxplots

Interactive plots

Heinrich Heine
Universität Düsseldorf

- e.g. (*gg*)*plotly* packge
- and *shinyplots* (shinyplots.io)

Some tips and tricks

Heinrich Heine
Universität Düsseldorf

- Cheat sheets
- Code snippets
- Google/Stackoverflow is your friend ☺

Hands-on examples...

Task 1

Heinrich Heine
Universität Düsseldorf

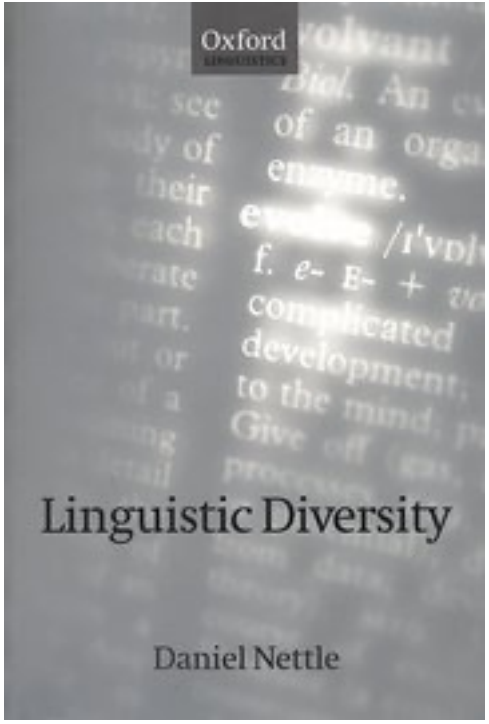
Fake dataset: conference acceptancy by country

- Plot the number of accepted abstracts by country as a stacked barplot showing percentages and absolute frequencies.
- (Fake data are generated in the code)

Task 2

Authentic dataset: Nettle (1999)

- hypothesis: linguistic diversity is correlated with climate factors
  - fertile environments → less reason to travel → less language contact → less linguistic diversity, and vice versa
- measured ecolocial risk using a country's Mean Growing Season (MGS) → how many months per year can you grow crops in the country?



(see Winter 2019: 34)