Welcome to the 2d world!
MVE080/MMG640 Lecture 1

Sebastian Persson
sebpe@chalmers.se

October 31, 2022

The course is divided into two parts

- Part1 (week 1-3) visualisation of data in 2 dimensions
  - Responsible : Sebastian Persson, PhD-student
- Part2 (week 4-6) visualisation in 3 dimensions
  - Responsible : Klas Modin (examiner)
- Normal schedule in MVE24-25
  - Mondays 13:15-15:00
  - Wednesdays 08:00-11:45

Visualisations to process information

- Dataset1-4 same means ($\mu_x = 9$ and $\mu_y = 7.5$)
- Dataset1-4 fit a linear model "equally" well $R^2 \approx 0.67$

x	y
10	8.04
8	6.95
13	7.58
9	8.81
11	8.33
14	9.96
6	7.24
4	4.26
12	10.84
7	4.82
5	5.68

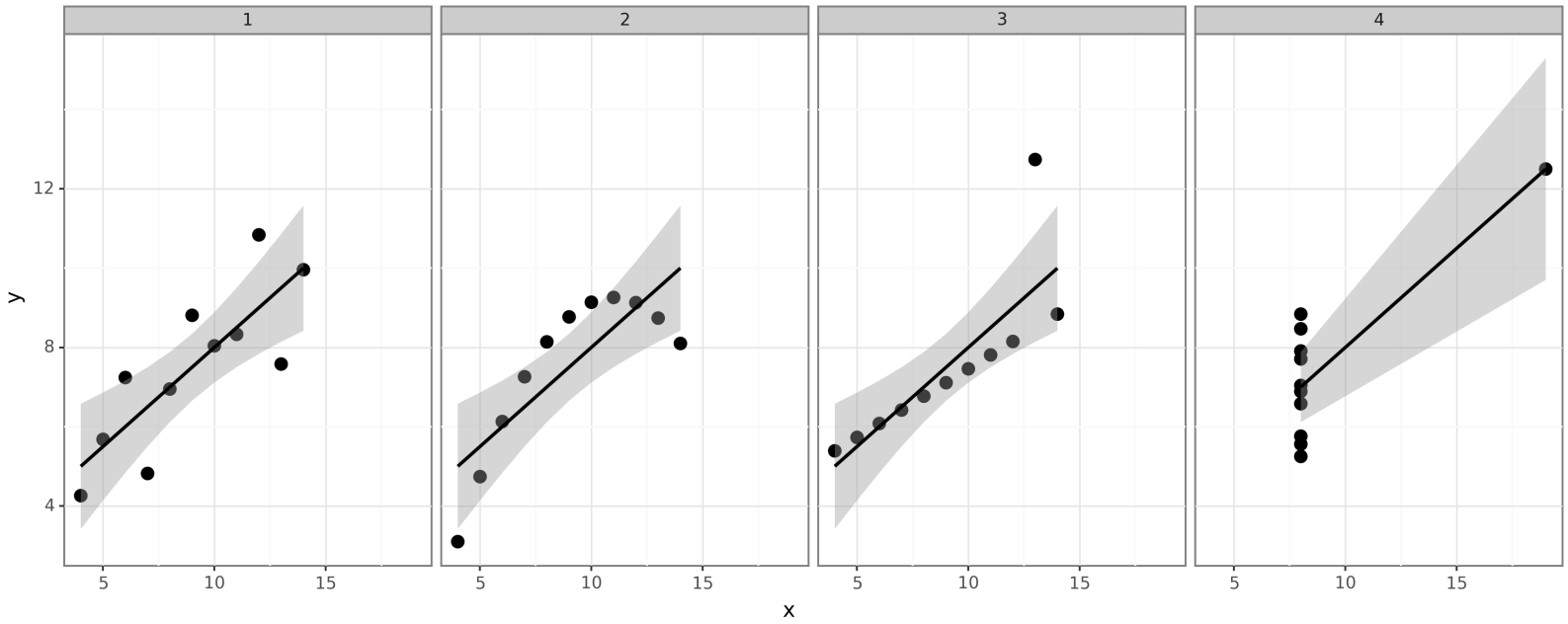
x	y
10	9.14
8	8.14
13	8.74
9	8.77
11	9.26
14	8.10
6	6.13
4	3.10
12	9.13
7	7.26
5	4.74

x	y
10	7.46
8	6.77
13	12.74
9	7.11
11	7.81
14	8.84
6	6.08
4	5.39
12	8.15
7	6.42
5	5.73

x	y
8	6.58
8	5.76
8	7.71
8	8.84
8	8.47
8	7.04
8	5.25
19	12.5
8	5.56
8	7.91
8	6.89

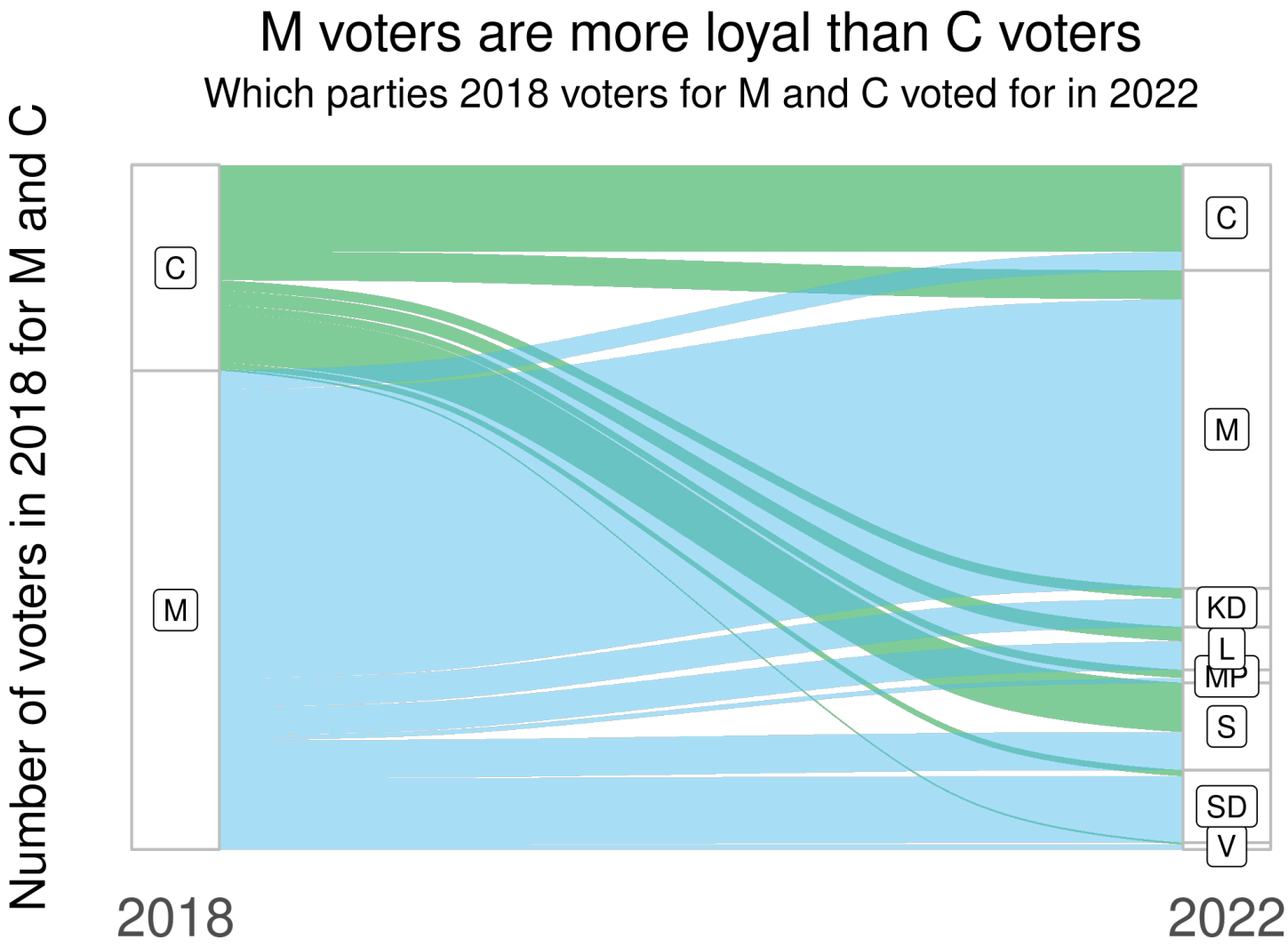
Visualisations to process information

- Only dataset1 fits linear model
- Visualisation important in preprocessing/modelling



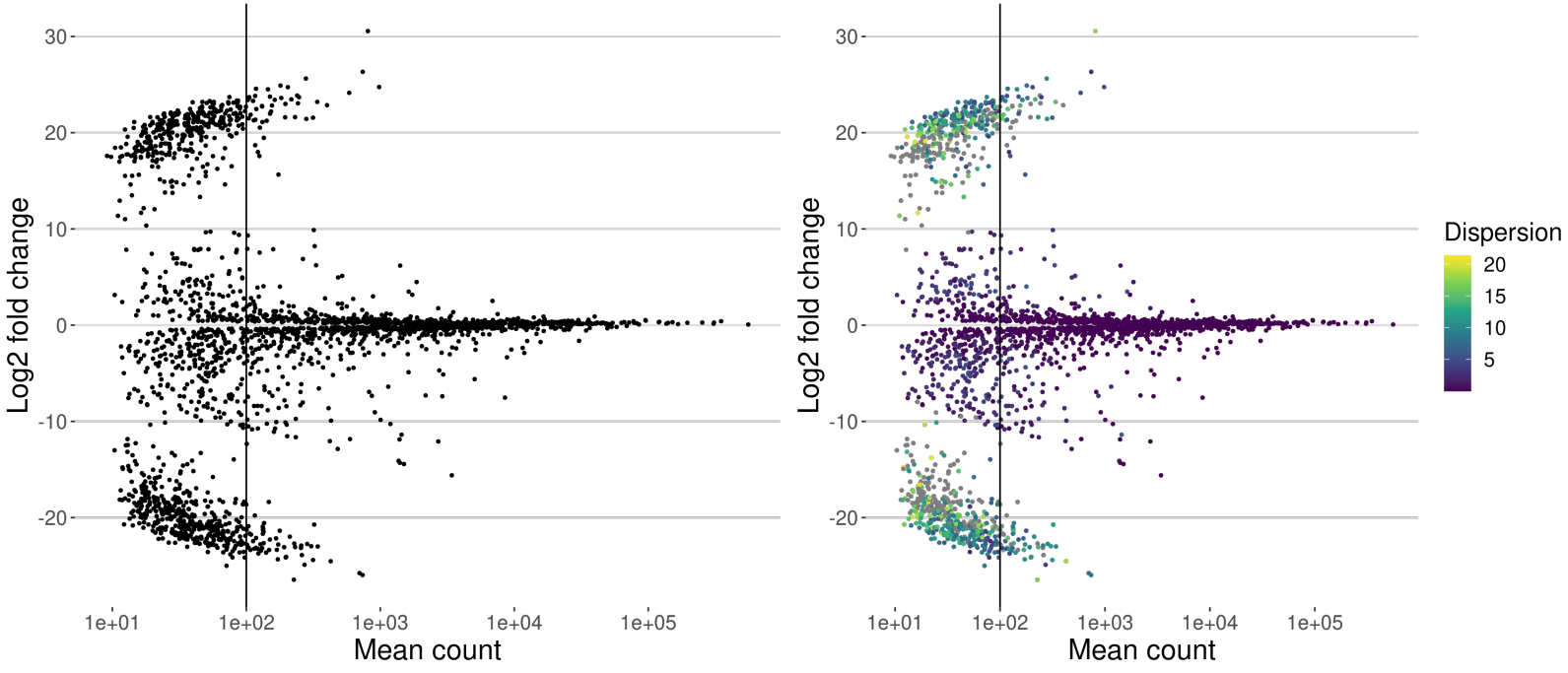
Visualisations can convey complex messages

- Albeit bigger M voters are more loyal than C voters



Visualisations to explore hypotheses

- Hypothsis: Outliers have small propensities (variance)
  - Hypothesis rejected from plotting



We need good visuals to...

1. Efficiently communicate to colleagues, managers and/or stakeholders
2. Efficiently explore/model/understand data

Hence after this part I want you to:

1. Given a dataset visualise the data in a truthful and readable format.
2. Be able to rapidly produce visuals.

Lecture outline

1. Introduction, grammar of graphics, tidy data and programming environment
2. Visualising amounts and distributions
3. Colors, themes and proportions
4. Associations and time series
5. Geospatial data and uncertainty
6. The truthful art

Weekly homework related to lecture content provided as notebooks.

Course litterature

- Wilke, Claus O. Fundamentals of data visualization: a primer on making informative and compelling figures. O'Reilly Media, 2019. *https://clauswilke.com/dataviz/*
- Cairo, Alberto. The truthful art: Data, charts, and maps for communication. New Riders, 2016. (do not buy)
- Wickham H (2016). ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York. ISBN 978-3-319-24277-4, *https://ggplot2-book.org/*.
- Franconeri, Steven L., et al. "The science of visual data communication: What works." Psychological Science in the public interest 22.3 (2021): 110-161. *https://journals.sagepub.com/doi/full/10.1177/15291006211051956*

Grammar of graphics

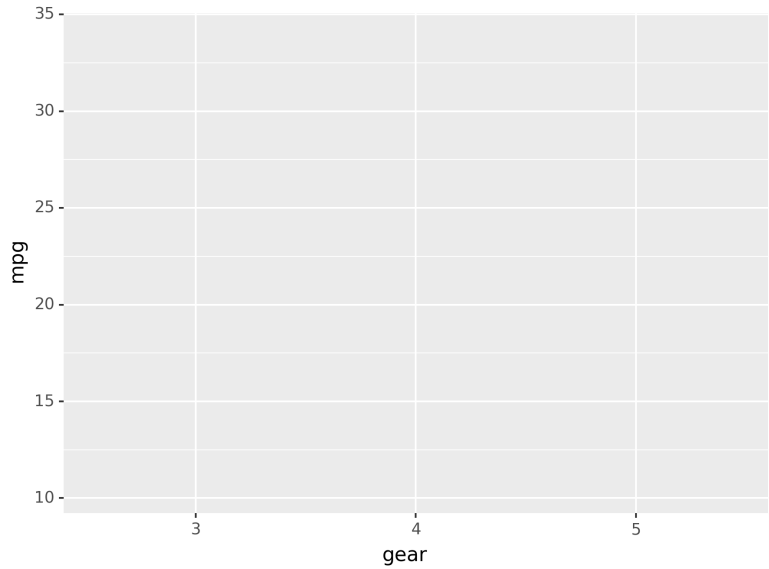
Framework to describe fundamental features of a graph

1. A dataset and mapping from variables to aesthetics
2. Layer(s) composed of geometrical objects, positions adjustments and optionally data+mapping
3. A scale for each aesthetic mapping.
4. A coordinate system
5. Faceting system (cover later in the course)
6. A theme

Dataset and mapping

Nothing to see yet as we have only provided a dataset and aesthetically mapped the variables gear to x-axis and mpg to y-axis in the *mtcars* dataset

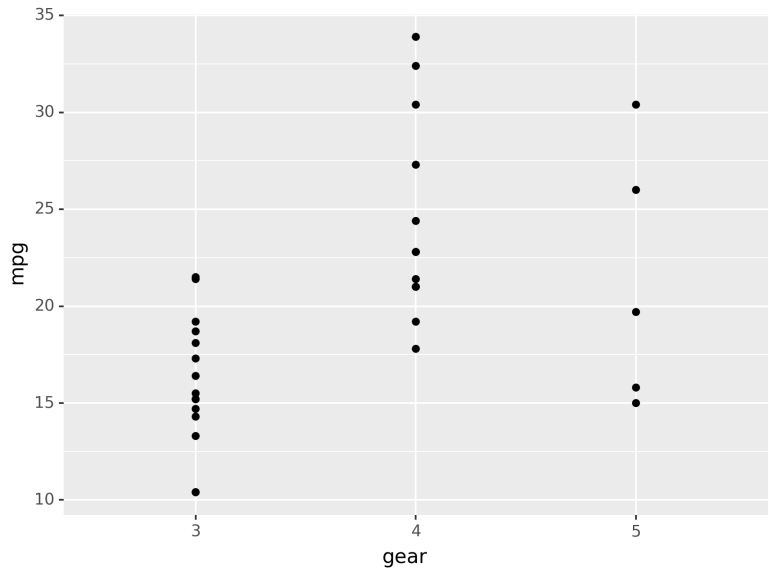
```
import numpy as np
import pandas as pd
from plotnine import *
from plotnine.data import mtcars

(ggplot(mtcars, aes("wt", "mpg")))
```



Layers

- Point is a geometrical object
- stat and identity have defaults values

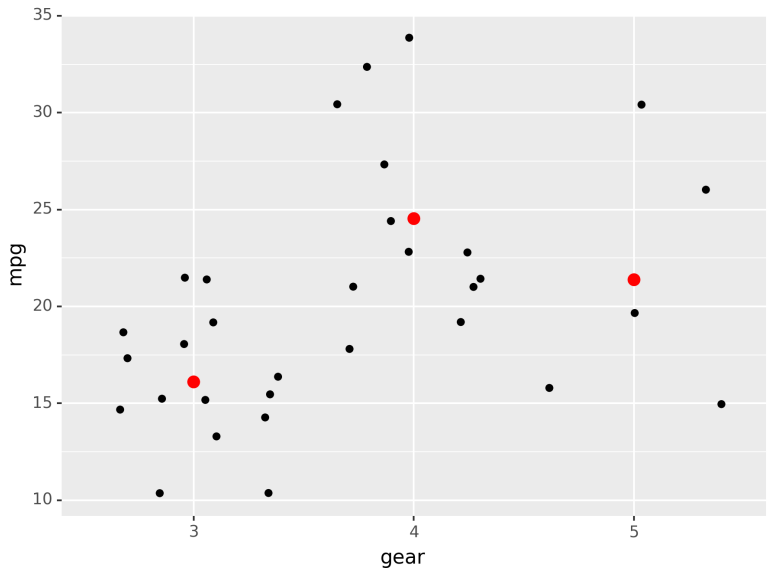
```
import numpy as np
import pandas as pd
from plotnine import *
from plotnine.data import mtcars

(ggplot(mtcars, aes('gear', 'mpg'))
    + geom_point(stat = "identity",
                 position = "identity"))
```



Layers

- stat - compute and plot functions of data
- position - jitter and/or move around data

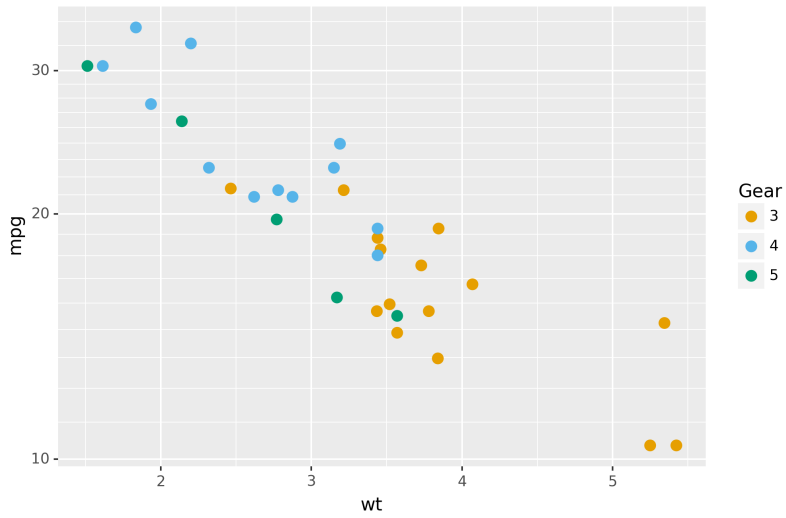
```
import numpy as np
import pandas as pd
from plotnine import *
from plotnine.data import mtcars

(ggplot(mtcars, aes('gear', 'mpg'))
    + geom_point(position = "jitter")
    + geom_point(stat = "summary",
                 fun_y = np.mean,
                 position = "identity",
                 size = 3.0,
                 color="red"))
```



Scale

- Map from data space to aesthetic space
  - Colors, shapes and/or linetypes
  - Axis (discrete, log, etc. . .)

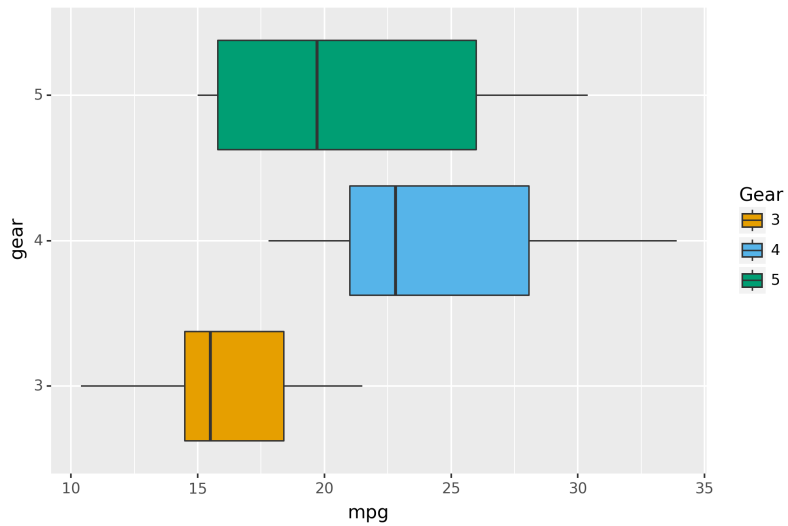
```
import numpy as np
import pandas as pd
from plotnine import *
from plotnine.data import mtcars
# A nice color palette
cbPalette = ["#E69F00", "#56B4E9",
             "#009E73", "#F0E442",
             "#0072B2", "#D55E00",
             "#CC79A7", "#999999"]

(ggplot(mtcars, aes("wt", "mpg",
                    color = "gear"))
    + geom_point(size=3.0)
    + scale_y_log10() # Scale data axis
    + labs(x = "wt", y = "mpg")
    + scale_color_manual(values = cbPalette,
                         name = "Gear"))
```



Coordinates

- Places position aesthetics correctly on a plot
  - Common to flip coordinate system

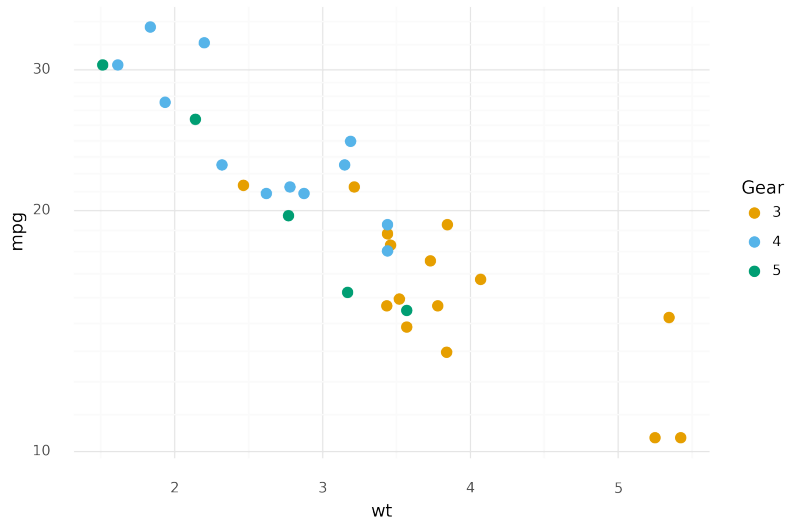
```
import numpy as np
import pandas as pd
from plotnine import *
from plotnine.data import mtcars
# A nice color palette
cbPalette = ["#E69F00", "#56B4E9",
             "#009E73", "#F0E442",
             "#0072B2", "#D55E00",
             "#CC79A7", "#999999"]

(ggplot(mtcars, aes('gear', 'mpg',
                    fill = "gear"))
    + geom_boxplot()
    + coord_flip()
    + scale_fill_manual(values=cbPalette,
                        name = "Gear"))
```



Theme

- Specifies fonts, ticks, panel strips and background.
  - Change axis text-size here

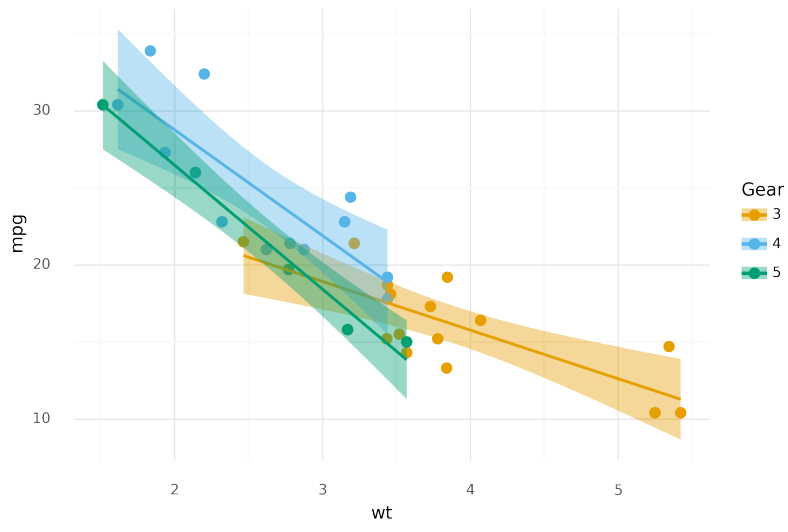
```
import numpy as np
import pandas as pd
from plotnine import *
from plotnine.data import mtcars
# A nice color palette
cbPalette = ["#E69F00", "#56B4E9",
             "#009E73", "#F0E442",
             "#0072B2", "#D55E00",
             "#CC79A7", "#999999"]

(ggplot(mtcars, aes("wt", "mpg",
                    color = "gear"))
    + geom_point(size=3.0)
    + labs(x = "wt", y = "mpg")
    + scale_color_manual(values = cbPalette,
                         name = "Gear")
    + theme_minimal())
```



Why use grammar of graphics?

- Flexible and intuitive (layer based)
- Mature libraries with large user bases
  - ggplot2 (R) and plotnine (Python)
- Powerful visuals from few lines of code

```
(ggplot(mtcars, aes("wt", "mpg",
                    color = "gear",
                    fill = "gear"))
    + geom_point(size=3.0)
    + geom_smooth(method="lm")
    + labs(x = "wt", y = "mpg")
    + scale_fill_manual(values = cbPalette,
                        name = "Gear")
    + scale_color_manual(values = cbPalette,
                        name = "Gear")
    + theme_minimal())
```



Tidy data

Both ggplot2 and plotnine work with tidy data

Tidy data

Both ggplot2 and plotnine work with tidy data

"Happy families are all alike; every unhappy family is unhappy in its own way." – Leo Tolstoy (Anna Karenina)

Tidy data

Both ggplot2 and plotnine work with tidy data

"Happy families are all alike; every unhappy family is unhappy in its own way." – Leo Tolstoy (Anna Karenina)

"Tidy datasets are all alike, but every messy dataset is messy in its own way." – Hadley Wickham

What is tidy data

1. Each variable has its own column
2. Each observation must have its own row
3. Each value must have its own cell

Site	1999	2000
Stockholm	13	21
Gothenburg	85	31
London	77	15

What is tidy data

1. Each variable has its own column
2. Each observation must have its own row
3. Each value must have its own cell

Site	1999	2000
Stockholm	13	21
Gothenburg	85	31
London	77	15

Site	year	cases
Stockholm	1999	13
Stockholm	2000	21
Gothenburg	1999	85
Gothenburg	2000	31
London	1999	77
London	2000	15

Programming environment

- I recommend to use Python with Anaconda
  - Latter part of the course is in Python
  - Widely used language
  - Anaconda great package manager (use yml-file)
- I recommend plotnine
  - Mature grammar of graphics library
  - Several themes and geoms
  - Similar to ggplot2 in R
- I recommend to work in Jupyter notebooks
  - Weekly homework distributed as notebooks
- I can also help with R.

For the next lecture. . .

Why is this plot ugly and hard to read?