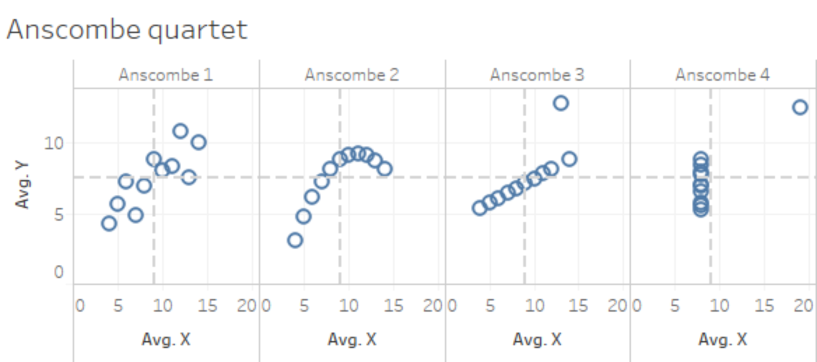
2. Data Wrangling and Programming in R

POL 212: Quantitative Analysis I

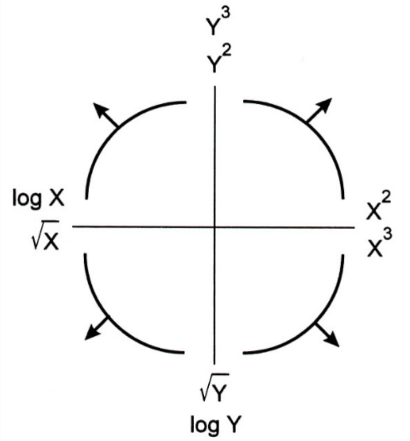
Winter 2024

Getting to know your data

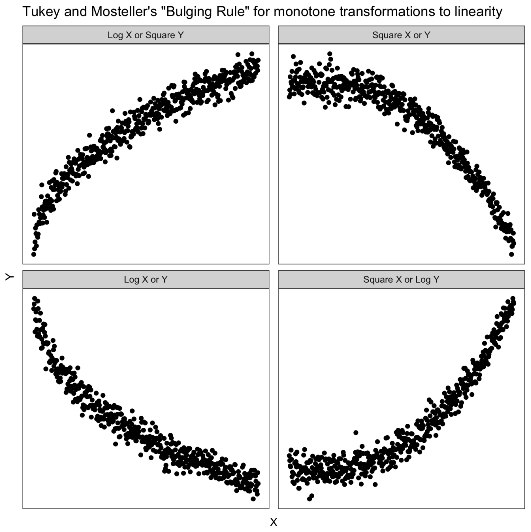


Linear regression and variable transformations

Mosteller and Tukey's **bulging rule**:



Linear regression and variable transformations



Linear regression and variable transformations

A quick example:

R Code

```
library(car)
scatterplot(prestige ~ income, data=Prestige)
scatterplot(prestige ~ log(income), data=Prestige)
```

Nominal and ordinal inputs

Quick review: nominal, ordinal, interval, and ratio-level variables (levels of measurement).

- Using dummy predictor variables only: Equivalent to analysis of variance (ANOVA), tests whether there is a significant difference between groups.
- **Example:** Is average income different between college and non-college educated persons? $Y_i = \alpha + \beta_1 C_i + u_i$, where Y_i is income for individual i and C_i is coded 1 for college degree and 0 otherwise.
- In this case, the hypothesis test for β_1 will be equivalent to the result you get for a difference-of-means test.
  - Mean for non-college educated: α
  - Mean for college educated: $\alpha + \beta_1$

Nominal and ordinal inputs

What about *nominal* predictors with more than two categories?

- Standard practice is to break the variable into series of $K-1$ dummy variables, where K is the total number of categories in the variable.
- **Crucial point:** You will need to select one of the categories as the reference category—it doesn't get a dummy.
- **Example:** Party identification (D, R, I) can be split into two dummy variables: D (1 if a Democrat, 0 if not) and R (1 if a Republican, 0 if not).
- In this case, the hypothesis test for β_1 will be equivalent to the result for a difference-of-means test.
  - Each party label will have its own β, which represents the effect relative to the omitted reference category.
  - These need to be coded as a `factor` in `R`.

Nominal and ordinal inputs

What about *ordinal* predictors with more than two categories?

- These are trickier, and very much up to the discretion of the researcher.
  - One approach is to treat them just like nominal variables: dummy them out. This complicates things a bit, but allows for nonlinear (and even non-monotone) relationships with the outcome variable.
  - But it is also not uncommon to see these treated as interval/ratio-level variables. This simplifies interpretation, but of course rests on an interval/ratio-level assumption (i.e., that the relationship between the predictor and outcome variables is linear and monotonic).
- In `R`, you can control this by coding the ordinal variable's class as `factor` or `numeric`.

Rescaling variables

- Before estimating a model with income as a variable, you should think about what unit you want to use. \$1? \$1,000?
- If you rescale the variable beforehand, then you can sensibly interpret the results afterward. Just remember, the true β_2 tells us for a one-unit change in the input variable, how many units will the outcome change *on average*. You have to know what a unit is, though.
- For example, suppose X_i is **years** of education and Y_i is **dollars** of income. How would you interpret the slope coefficient for this estimated model?

$$Y_i = 20000 + 1000 X_i + \hat{u}_i$$

Standardizing variables

- A common rescaling strategy is to *standardize* our variables.
- The formula is simple:

$$X_i^* \quad = \quad \frac{X_i - \bar{X}}{\sigma_X}$$

- Each rescaled variable has a 0 mean and a standard deviation of 1. Straightforward interpretation.
- Consequently the regression coefficients are referred to as “standardized coefficients.”
- Can simply use the base `scale()` function in `R`.

Creating indices

- You will also frequently encounter situations where you have multiple indicators of the same underlying concept (e.g., ideology, partisanship, democracy, human rights).
- You can use more advanced scaling techniques to combine information from these variables, but (especially if you have good reason to believe these indicators comprise a single concept/dimension/etc.) my best advice is to simply use a **summated rating scale** (i.e., take the average).
- Why? Measurement and the theory of errors.

Creating indices

- Of course, make sure all items are on the same scale!

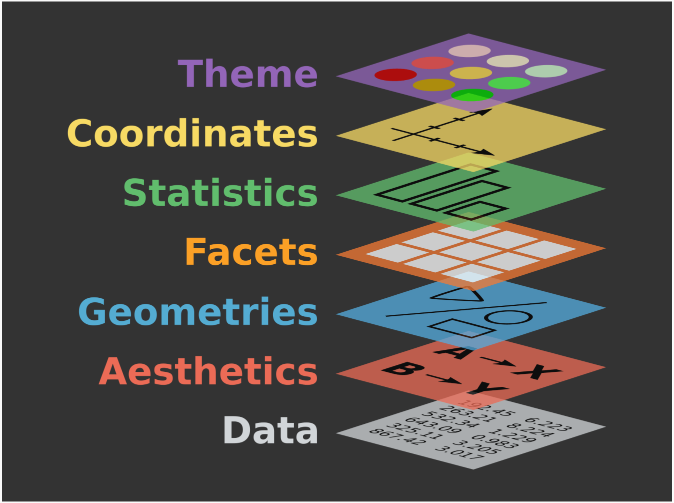
```
rescale01 <- function(x){
(x - min(x, na.rm=TRUE))/
(max(x, na.rm=TRUE) - min(x, na.rm=TRUE))}
```

- One popular measure of the constructed scale's *reliability* is Cronbach's α:

$$\alpha = \frac{p\bar{r}}{1+\bar{r}(p-1)}$$

where p is the number of items and $\bar{r}$ is the mean bivariate correlation among the items. Can be calculated using `psych::alpha` in `R`.

ggplot: Grammar of Graphics



ggplot: Grammar of Graphics

A quick example:

R Code

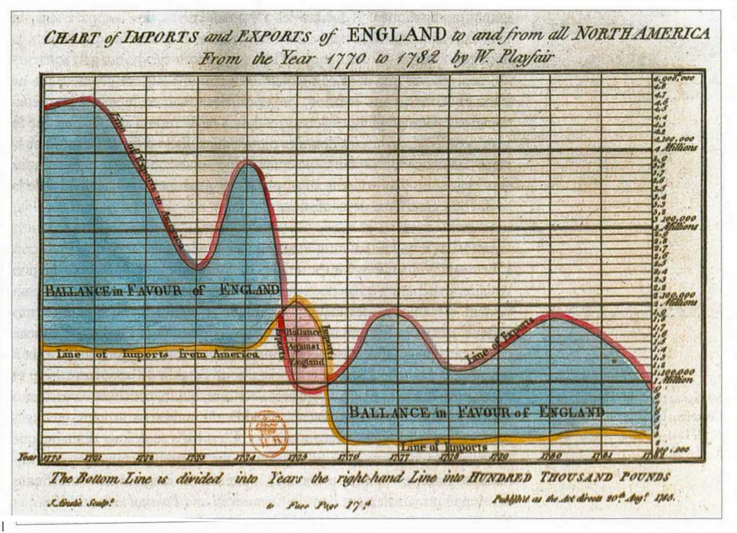
```
library(ggplot2)
ggplot(data=diamonds, aes(x=carat, y=price, color=clarity)) +
geom_point() +
scale_color_brewer(type="seq", palette="Reds") +
facet_wrap(~clarity, ncol=4) +
guides(color="none")
```

ggplot: Grammar of Graphics

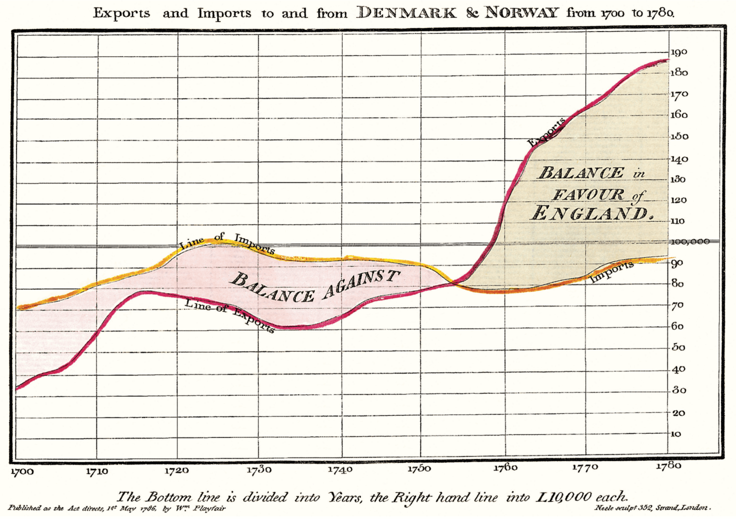
Some resources:

1. https://r-graph-gallery.com
2. https://exts.ggplot2.tidyverse.org/gallery
3. https://bookdown.org/content/b298e479-b1ab-49fa-b83d-a57c2b034d49
4. https://r-charts.com/ggplot2
5. https://github.com/erikgahner/awesome-ggplot2

Data visualization and the data-ink ratio



Data visualization and the data-ink ratio



tidyverse

dplyr
readr
purrr
tidyverse
ggplot2
tidyr
TIBBLE
www.rstudio.com