2: Tidy data

Videregående kvantitative metoder i studiet af politisk adfærd

Frederik Hjorth
fh@ifs.ku.dk
fghjorth.github.io
@fghjorth

Institut for Statskundskab
Københavns Universitet

13. september 2017

Leeper, *Really Introductory Introduction*:

- Getting started
  - brug af R som regnemaskine: fx. `(2+4)/7`
  - parsing errors ctr. syntax errors
  - nye vektorer: fx. `dice <- c(2,2,3,4)`
  - ekstrahering fra vektorer: fx. `dice[1:3]`
  - ny data frame: fx. `df <- data.frame(dice,number=1:4)`
  - data framens struktur: `str(df)`
  - centrale tendenser: `summary(df)`

- Real data
  - installer pakker: `install.packages()`
  - indlæs pakker: `library()`
  - importér data: `import()` fra `rio`-pakken

- Randomness
  - sample fra en vektor: `sample()`
- Plots
  - pakke: `ggplot2`
  - fx. `ggplot(iris,aes(x=Sepal.Length)) + geom_histogram`

- Basic programming tools
  - funktioner: fx. `ftoc <- function(f){ c<-((f-35)*5)/9 ; print(c) }`
  - for loops: fx. `for (i in 1:10) print(i*i)`

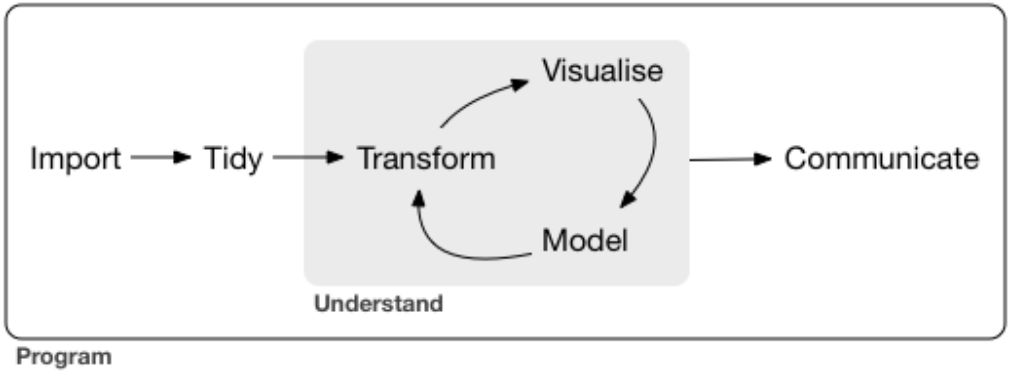
Swirl øvelse 1.1-1.4

- assignment
- vektorer
- 'recycling' af vektorer
- working directories
- talrækker
- atomiske vektorer ctr. lister
- logiske operatorer og vektorer
- character vektorer
- `paste()`

Tænkeren bag 'tidy data': Hadley Wickham

i alt forfatter til $\approx$ 63 R-pakker (a.k.a. 'hadleyverse')

Et typisk data science workflow:



Centrale Hadley-pakker til databehandling:

- ggplot2 (visualisering)
- dplyr (databehandling)
- readr (import af csv o. lign.)
- purrr (funktioner og loops)
- tibble (smartere version af data frames)

$\rightarrow$ alle disse kan loades samtidig med `library(tidyverse)`

Andre nyttige hadleyverse-pakker:

- haven (import af data fra Stata, SPSS etc.)
- rvest (web scraping)
- stringr (behandling af character-objekter)
- lubridate (dato-objekter)
- forcats (faktorer)

Messy data:

	treatmenta	treatmentb
John Smith	–	2
Jane Doe	16	11
Mary Johnson	3	1

Tidy data:

person	treatment	result
John Smith	a	–
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Principper i tidy data:

1. Hver variabel udgør en kolonne
2. Hver observation udgør en række
3. Hver enhed udgør en tabel

NB: good for data entry $\neq$ tidy $\rightarrow$ det typiske datasæt er messy!

De fire verber i databehandling:

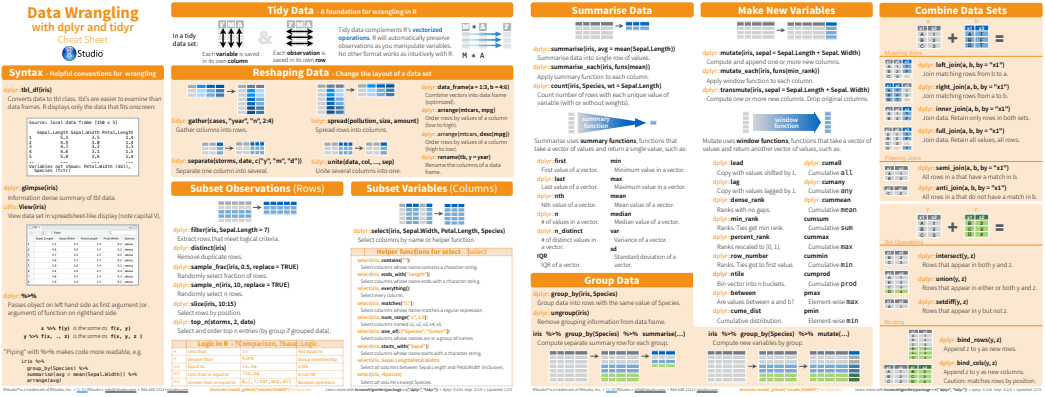
1. *filter*: udvælg observationer
2. *transform*: tilføje eller omkode variable
3. *aggregate*: opsummere værdier til én eller færre
4. *sort*: ændre observationers rækkefølge

→ vi kan foretage (næsten) al databehandling med `dplyr`

Implementering af de fire verber i `dplyr`:

	verbum	funktion i dplyr
1	filter	`filter()` til rækker, `select()` til kolonner
2	transform	`mutate()`
3	aggregate	`group_by()` og `summarise()`
4	sort	`arrange()`

Uvurderligt opslagsværk til databehandling: *data wrangling cheat sheet*



ligger også på fagets hjemmeside under *extras*


%>%
magrittr
Ceci n'est pas un pipe.

Data i øvelsen til næste gang: pakkedownloads på The Comprehensive R Archive Network (CRAN) d. 8. juli 2014

```
> cran
# A tibble: 225,468 x 11
       X       date     time    size r_version r_arch      r_os      package version country ip_id
   <int>      <chr>    <chr>   <int>     <chr>  <chr>     <chr>        <chr>   <chr>   <chr> <int>
 1     1 2014-07-08 00:54:41   80589     3.1.0 x86_64   mingw32    htmltools   0.2.4      US     1
 2     2 2014-07-08 00:59:53  321767     3.1.0 x86_64   mingw32     tseries 0.10-32      US     2
 3     3 2014-07-08 00:47:13  748063     3.1.0 x86_64 linux-gnu       party  1.0-15      US     3
 4     4 2014-07-08 00:48:05  606104     3.1.0 x86_64 linux-gnu       Hmisc  3.14-4      US     3
 5     5 2014-07-08 00:46:50   79825     3.0.2 x86_64 linux-gnu      digest   0.6.4      CA     4
 6     6 2014-07-08 00:48:04   77681     3.1.0 x86_64 linux-gnu randomForest   4.6-7      US     3
 7     7 2014-07-08 00:48:35  393754     3.1.0 x86_64 linux-gnu         plyr   1.8.1      US     3
 8     8 2014-07-08 00:47:30   28216     3.0.2 x86_64 linux-gnu      whisker   0.3-2      US     5
 9     9 2014-07-08 00:54:58    5928        <NA>   <NA>      <NA>         Rcpp  0.10.4      CN     6
10    10 2014-07-08 00:15:35 2206029     3.0.2 x86_64 linux-gnu     hflights     0.1      US     7
# ... with 225,458 more rows
```

Vi ønsker at gøre flg.:

1. gruppér observationerne efter pakke
2. udregn nogle summary stats for hver pakke
3. reducér til pakker downloadet i mere end 60 lande
4. sortér efter land og pakkestørrelse

Klodset løsning I: Gem hvert skridt i et nyt objekt ($\rightarrow$ rod)

```
by_package <- group_by(cran, package)

pack_sum <- summarize(by_package,
                      count = n(),
                      unique = n_distinct(ip_id),
                      countries = n_distinct(country),
                      avg_bytes = mean(size))

top_countries <- filter(pack_sum, countries > 60)

result1 <- arrange(top_countries, desc(countries), avg_bytes)
```

Klodset løsning II: Skriv funktionerne indlejret (nested) i hinanden ($\rightarrow$ totalt ulæseligt)

```
result2 <-
  arrange(
    filter(
      summarize(
        group_by(cran,
                 package
        ),
        count = n(),
        unique = n_distinct(ip_id),
        countries = n_distinct(country),
        avg_bytes = mean(size)
      ),
      countries > 60
    ),
    desc(countries),
    avg_bytes
```

Smart løsning: brug pipede funktioner ($\rightarrow$ enkelt, letlæseligt)

```
result3 <-
  cran %>%
  group_by(package) %>%
  summarize(count = n(),
            unique = n_distinct(ip_id),
            countries = n_distinct(country),
            avg_bytes = mean(size)
  ) %>%
  filter(countries > 60) %>%
  arrange(desc(countries), avg_bytes)
```

bemærk: den originale data frame 'pipes' ned gennem funktionerne, så angives kun én gang

Øvelse: databehandling med dplyr

1. load swirl med `library(swirl)`
2. hent kurset "Getting and Cleaning Data"
   - bør fungere med `install_course("Getting and Cleaning Data")`
   - ellers download filen og kør `install_course()`
3. løs øvelse 1 (dplyr verber)
4. løs øvelse 2 (piping fra 69 pct.)

Næste gang:

- OLS brush-up
- AP 3.1+3.2
- læs Newman (kun mhp. metoden), derefter Solt
- lektie: færdiggør modul 1+2 i "Getting and Cleaning Data"

Tak for i dag!