Essential Research Toolkit for the Humanities

Week 4: R basics

Anna Pryslopska

May 2, 2022

Psycholinguistics and Cognitive Modeling Lab

Questions?

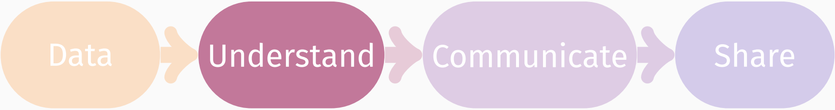
1. Change the editor theme and pane layout
2. Install & load `tidyverse`, `knitr`, `learnr`, +1
3. What is `typeof()`:
   `"Anna"`, `-10`, `FALSE`, `3.14`, `as.logical(1)`
4. Is the following true:
   `7+0i == 7`, `9 == 9.0`, `"zero" == 0L`
5. What is the output of the following operations and why
   - `10 < 1` FALSE
   - `5 != 4` TRUE
   - `1.0 == 1` TRUE
   - `4 * 9.1` 36.4
   - `"a" + 1` ⊘
   - `0/0` NaN
   - `b * 2` ⊘
   - `(1-2)/0` -Inf
   - `10 <- 20` ⊘
6. Complete the ”Data basics” tutorial from the package `learnr`
7. Report `sessionInfo()`

Table of contents

1. Where are we this week?
2. Inspecting data
3. Cleaning and transforming data
4. Homework assignment

Where are we this week?

Workflow



Data

Understand

Communicate

Share

import,
clean,
transform,
visualize,
model

document,
create
clean and
beautiful
reports

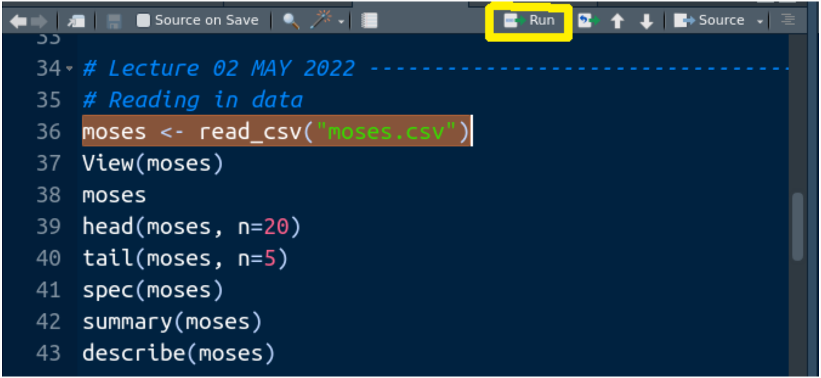
connect,
collaborate,
backup

Inspecting data

Load the data into environment

```
moses <- read_csv("moses.csv")
```

`CTRL+ENTER` or `CMD+RETURN` or click on “Run”



```
# Lecture 02 MAY 2022 ----------------------------------
# Reading in data
moses <- read_csv("moses.csv")
View(moses)
moses
head(moses, n=20)
tail(moses, n=5)
spec(moses)
summary(moses)
describe(moses)
```

You now have a data frame or tibble called `moses`.

Look at what you did

`View(moses)`	in the RStudio window
`moses`	in the console
`print(moses, n=Inf)`	in the console
`head(moses, n=20)`	first 20 rows
`tail(moses, n=5)`	last 5 rows
`spec(moses)`	column properties
`summary(moses)`	summary statistics
`describe(moses)`	summary statistics vol. 2
`colnames(moses)`	column names
`summary(NAME)`	→ calling function with one argument
`head(NAME, n=20)`	→ calling function with two arguments
`dbinom(x=6, size=9, prob=0.5)`	3 arguments in order, 2 named

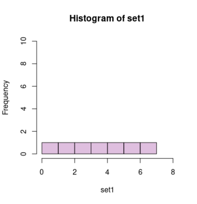
Summarize

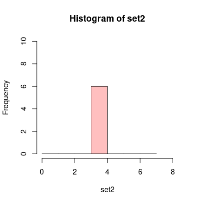
`Min.`	`min()`	minimal value
`Max.`	`max()`	maximal value
`Mean`	`mean()`	average
`1st Qu.`	`quantile()`	25%
`Median`	`quantile()`	middle number == 2nd quantile == 50%
`3rd Qu.`	`quantile()`	75%
`NA's`	TBA	nr missing data

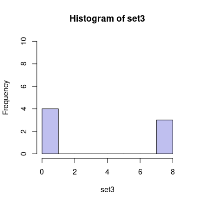
Describe

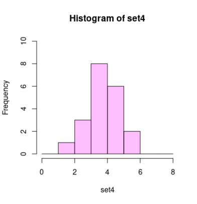
	`colnames()`	item name
`vars`	`colnames()`	item number
`n`	TBA	number of valid cases
`mean`	`mean()`	mean
`median`	`median()`	median
`min`	`min()`	minimum
`max`	`max()`	maximum
`range`	`range()`	range
`sd`	`sd()`	standard deviation ($\sqrt{variance}$)
`trimmed`		trimmed mean
`mad`		median absolute deviation
`skew`	`skew()`	skew
`kurtosis`	`kurtosi()`	kurtosis
`se`	`mean_se()`	standard error

Central tendency









Set	Values	Mean	Median	SD
1	1,2,3,4,5,6,7	4	4	2
2	4,4,4,4,4,4	4	4	0
3	1,1,1,1,8,8,8	4	1	4
4	2,2,3,3,3,3,3,3,3,4,4,4,4,4,4,5,5,5,5,6	4	4	1

What a mess

- too much information IP
- too little information condition 1? 2?? 100???
- missing information NA
- inconsistent information

  Q: Margaret Thatcher was the former president/prime minister of which country?

  A: uk, the uk, england, united kingdom, great britain...

Kehr-Woche

Nach der Hausordnung sind im wöchentlichen Wechsel, beginnend mit jedem Sonntag

Hinter dem Haus-Hauseingang-Bürgersteig-Mülleimerplätze.

Im Hause: [illegible] Putzen der Kellertreppen u. Kellervorplätze

Nach dem Kaminfegen An 4 Stellen (2 im Kellervorraum, 1 neben der Haustür u. 1 i. d. Waschküche) Ruß entfernen. Hausmülleimer!!

Im Winter: Leerung des Wasserbehälters im Kellervorraum. Eisbildung (Wasserablauf) am Eingang zum Haus verhindern!

Beim Schneeräumen kein Salz [illegible]

In dieser Woche ist die Reihe an Ihnen

[illegible] Samstag abend Kehrwechsel

Clean up after yourselves

select meaningful columns	`select(WHERE, WHAT)`
remove missing values	`na.omit(WHERE)`
choose or remove data	`filter(WHERE, TRUE CONDITION)`
reorder values	`arrange(WHERE, HOW)`
create values	`mutate(WHERE, NEW = FUNCTION(OLD))`

Functions are executed, results are displayed, but nothing is saved.

Selecting

Tidyverse

```
select(moses, ID, Item, Condition, Answer)
```

```
select(moses, c(ID, Item, Condition, Answer))
```

```
select(moses, c(ID, Item:Answer))
```

base R

```
moses$ID
```

```
moses[ , "ID"]
```

```
moses[ , c("ID", "Item", "Condition", "Answer")]
```

```
moses[ , c(1,4:6)]
```

Both

`c()` = concatenate, i.e. combine, join, bundle up

Create a new data frame with columns: ID, Item, Condition, Answer

Missing data

`na.omit(moses)`	everywhere
`na.omit(moses$Item)`	only in the items
`na.omit(moses[ , "Item"])`	only in the items
`na.omit(moses[ , 4])`	only in the items

```
is.na(WHERE)
is.na(select(moses, Item))
```

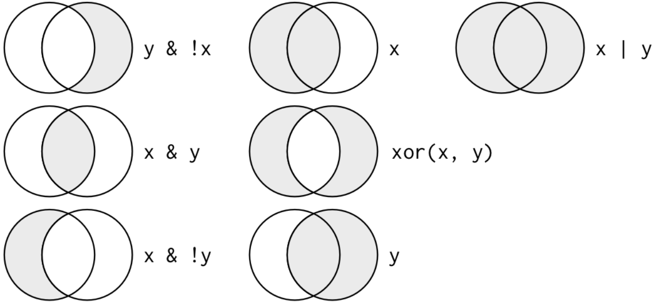
Create a new data frame from the previous one with no NAs.

Coding basics: R as a calculator

addition	`+`
subtraction	`-`
division	`/`
multiplication	`*`
power	`^`
equals	`==`
not equals	`!=`
greater than	`>`
greater than or equal	`>=`
less than	`<`
less than or equal	`<=`
range	`NR1:NR2`
identify element	`VALUE %in% OBJECT`

Coding basics: Logic

negation	!
logical *and*	&
logical *or*	|



Wickham and Grolemund (2016)

Filter (out)

`filter(moses, Condition == 1)`	condition 1
`filter(moses, Condition %in% 1)`	condition 1
`filter(moses, Condition >= 1 & Condition < 2)`	condition 1
`filter(moses, Condition == 1 | Condition == 2)`	conditions 1–2
`filter(moses, Condition %in% 1:2)`	conditions 1–2
`filter(moses, Condition == 1:2)`	conditions 1–2
`filter(moses, Condition < 100)`	conditions 1–2
`filter(moses, Condition %in% c(1, 2))`	conditions 1–2

Create a new data frame from the previous one with conditions 1–2.

(Re)arrange

`arrange(moses, Item)`	item
`arrange(moses, Item, Condition)`	item, then condition
`arrange(moses, desc(List))`	list, descending
`arrange(moses, desc(is.na(Answer)))`	

Create a new data frame from the previous one and sort it by participant ID.

Create and mutate

Tidyverse

`mutate(moses, Class = TRUE)`	new column
`mutate(moses, Number = 1:598)`	
`mutate(moses, Lists = List + 1)`	calculate column
`mutate(moses, List1 = List == 1)`	evaluate column
`mutate(moses, Condition = as.character(Condition))`	overwrite column
`mutate(moses, List1 = NULL)`	remove column

Create and mutate

base R

```
moses$Class <- TRUE
moses$Number <- 1:598
moses$Lists <- moses$List + 1
moses$List1 <- moses$List == 1
moses$Condition <- as.character(moses$Condition)
moses$List1 <- NULL
```

💾 Assignment saves, so be careful! This code deletes `List1` and permanently changes `Condition`.

Cleaning and transforming data

Stuttgarter Nachrichten www.stuttgarter-nachrichten.de

Task: Tidy up the data

💾 remember to save in the environment!

1. select relevant columns (ID, Item, Condition, Answer)
2. remove missing data
3. arrange by Item and Condition
4. recode inconsistent information

`unique(VALUES)`	show all *unique* values
`unique(moses$Answer)`	plain list
`unique(select(moses, Answer))`	only as many as fit on screen
`print(unique(select(moses, Answer)), n=Inf)`	show all

Clean up on aisle “Answer”

```
dont_know <- c("don't know", "don’t know", "don't
know his name", "dont know", 'idk', "i forget")

cant_answer <- c("can't answer", "can't say",
"can’t say", "cant say", "none")

armstrong <- c("neal armstrong", "neil armstrong",
"armstrong")
```

…

Consolidate the answers for:
`everest, madrid, manchester, nobel, olympics, platypus, prince, printing, roman, sagrada, santa, scholz, shakespeare, squirrel, switzerland, ten, two, uk, usa, valentines, whale`

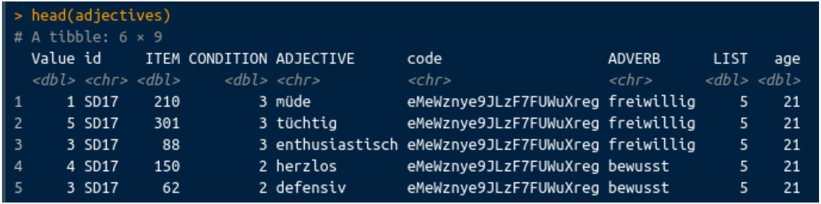
⚠ use `arrange()`, `filter()`, `select()`, and `unique()`

Homework assignment

Homework assignment due May 9

Next week: More data manipulation, pipelines, documentation, tidy code, and getting help

- Complete assignment 2 (→ ILIAS)
- Tidy up the adjectives data
    - Download the file `adjectives.csv` from ILIAS
    - Examine the data
    - Look for mistakes (missing data, values that don't fit etc.) given the information about the data (next slide)
    - Remove missing and incorrect values
    - Which variables/columns seem most important? Save a new data frame with just the relevant columns
    - Arrange the data by participant, item, and condition



```
> head(adjectives)
# A tibble: 6 × 9
  Value id     ITEM CONDITION ADJECTIVE      code                 ADVERB      LIST   age
  <dbl> <chr> <dbl>     <dbl> <chr>          <chr>                <chr>      <dbl> <dbl>
1     1 SD17    210         3 müde           eMeWznye9JLzF7FUWuXreg freiwillig     5    21
2     5 SD17    301         3 tüchtig        eMeWznye9JLzF7FUWuXreg freiwillig     5    21
3     3 SD17     88         3 enthusiastisch eMeWznye9JLzF7FUWuXreg freiwillig     5    21
4     4 SD17    150         2 herzlos        eMeWznye9JLzF7FUWuXreg bewusst        5    21
5     3 SD17     62         2 defensiv       eMeWznye9JLzF7FUWuXreg bewusst        5    21
```

· Value	acceptability rating to the sentence on 1–7 scale
· id	participant ID 1–63
· ITEM	sentence ID 1–360
· CONDITION	sentence group 1–3
· ADJECTIVE	adjective used in the sentence
· code	random letters and numbers
· ADVERB	adverb used in the sentence
· LIST	version of experiment 1–6
· age	age of participant in years

Logic exercise

Your world has four individuals:
🐔 🦆 🦕 🐊

Two are of the type `bird`
🐔 🦆

Two are of the type `can swim`
🦆 🐊

Using only basic logical expressions (negation `!`, and `&`, or `|`) and the two groups, describe the groups on the right, as in the first example. Tip: a Venn diagram as on slide 14 might help.

🦕 🐊 `!bird`

🐔 🦕

🦆

🐊

🦕

🐔

🐔 🦆 🐊

🦆 🦕 🐊

🐔 🦆 🦕 🐊

∅ (i.e. exclude all)