MAX PLANCK INSTITUTE
FOR PSYCHOLINGUISTICS

Key Practices for the Language Scientist:
Introduction to R and RStudio

NATALIA LEVSHINA

Outline

1. Why R is good for you
2. Basics of syntax
3. Installation
4. Data types and classes
5. Some useful functions
6. Quest
7. How to import and export your data.

What is R?

- A language for programming, statistics and visualizations
- Core + add-on packages
- Constantly in a flux

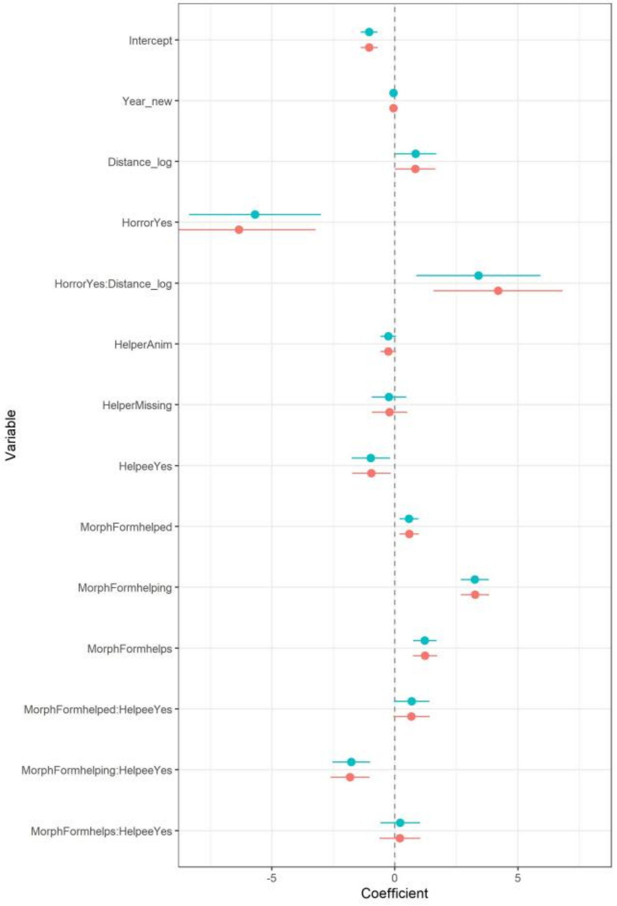
Advantages

- Free, open-source
- Comprehensive
- Many add-on packages for all kinds of purposes
- A big community
- Good documentation and help on diverse fora
- Automatization and reproducibility of work with the help of scripts

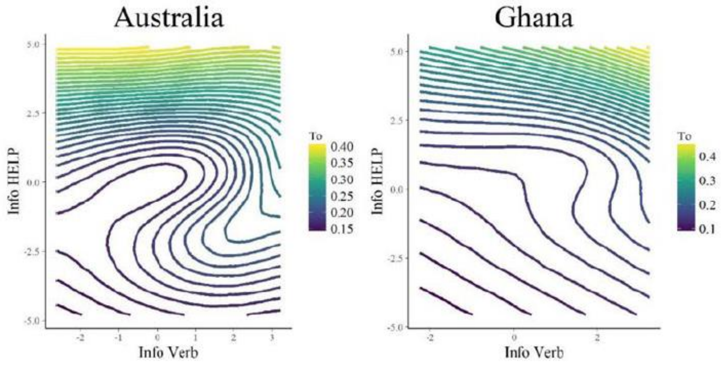
Graphs

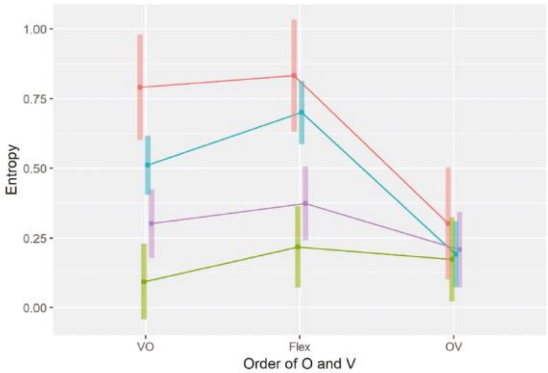
See examples at https://www.r-graph-gallery.com/

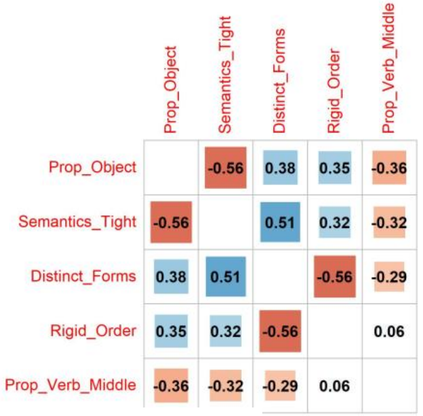
Examples from linguistics



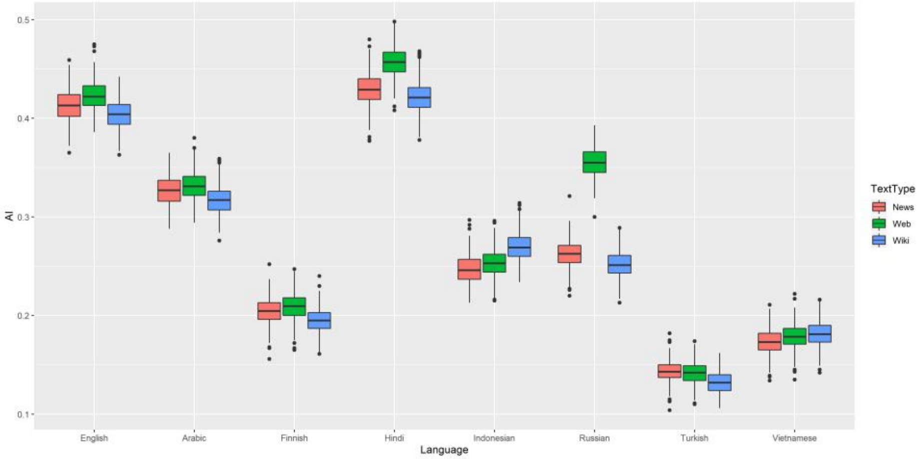
Model
● ML_big
● Bayes_big

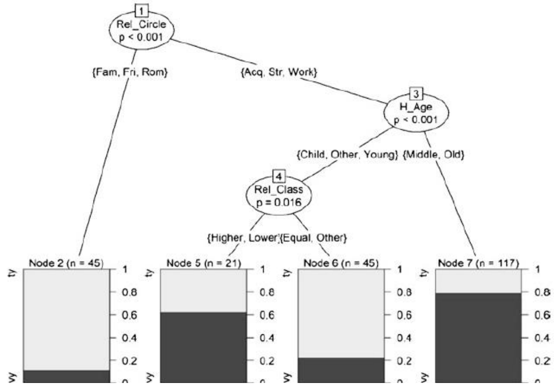






	Prop_Object	Semantics_Tight	Distinct_Forms	Rigid_Order	Prop_Verb_Middle
Prop_Object		-0.56	0.38	0.35	-0.36
Semantics_Tight	-0.56		0.51	0.32	-0.32
Distinct_Forms	0.38	0.51		-0.56	-0.29
Rigid_Order	0.35	0.32	-0.56		0.06
Prop_Verb_Middle	-0.36	-0.32	-0.29	0.06	





Disadvantages

- Command-line interface (but RStudio is hybrid)
- Syntax:
  - no typos are allowed
  - difficult to learn
  - a lot of variability across packages
- Lower speed in comparison with some other languages

Installing R and RStudio

- R: https://www.r-project.org/ or https://cran.r-project.org/ (choose a mirror)
- RStudio: https://rstudio.com/products/rstudio/download/ (the free version is enough)

Data classes

- Vectors: numeric, character, logical, factors
- Data frames
- Matrices
- Lists
- …

Vectors

- One-dimensional, contain only elements of the same type (numbers, characters, etc.)

- See the script.

Exercise

1. Create a character vector Name with the first names of your family members or friends.

2. Create a character vector FavPet with their favourite pets.

3. Create a numeric vector Age with their age.

4. Create a logical vector Pizza with the TRUE and FALSE values depending on whether they like pizza or not.

5. Change the character vector FavPet into a vector with factors. How many levels do you have?

Data frames

A data frame is a two-dimensional object with rows (e.g. subjects or stimuli) and columns (variables) of any data type.

Exercise

Using your previous objects, create a data frame with your family members or friends.

R functions

- Functions have a fixed structure: They start with the function name, followed by their arguments between round brackets ()

- If you don’t specify all arguments, the default settings will be used:

```
> sort(RT)
[1] 455 512 667 773
> sort(RT, decreasing = FALSE)
[1] 455 512 667 773
> sort(RT, decreasing = TRUE)
[1] 773 667 512 455
```

Exercise

How to fix the bug? Name two methods.

```
> head(2, RT)
Error in checkHT(n, dx <- dim(x)) :
  invalid 'n' - must have length one when dim(x) is
NULL, got 4
```

Tip: you can access the help page of a function by adding a question mark before its name: `?head`

Randomness and reproducibility

- Random sampling:

```
> sample(1:100, 10)
 [1] 44  6 87 56 52 96 57 65 49 54
```

- To reproduce the results later, choose any number **immediately** before sampling and use it later:

```
> set.seed(1)
> sample(1:100, 10)
 [1] 68 39  1 34 87 43 14 82 59 51
```

Reproducing the same ‘random’ numbers later:

```
> set.seed(1)
> sample(1:100, 10)
 [1] 68 39  1 34 87 43 14 82 59 51
```

A quest

1. Take the natural logarithm of 22026.47.
2. Type in R: set.seed(x), where x is the result of step 1.
3. Create a random sample of 10 numbers from 1 to 100. Use the function “sample()”. (See previous page, if needed!)
4. Find the 1st element. This will be your y.
5. Take the y^{th} letter in the English alphabet. Write it down.
6. Open the help page of the function read.table() and find the subsection “See also”. Find the first R function mentioned in that subsection. Remove the first letter and write down the result.
7. Find R citation information using citation(). Take the 3rd word and write down the letter.
8. Put all words together!

Important!

See R Script for examples:

- Spaces usually don’t matter (but don’t separate <- !)
  - Also, use _ to connect elements of a variable name, e.g. my_data <- 5 instead of my data <- 5.
- Signs == and = have different meanings!
- R is case sensitive.
- No typos!
- Remember to use quotation marks around character strings, unless they are object names.

Addition: importing and exporting your data

1. To export your data frame as a tab-separated file:

```
> write.table(…[the name of your data frame]…, file =
mydata.txt", quote = FALSE, sep= "\t", row.names= FALSE)
```

2. To read the file back in R:

```
> mydata <- read.table(file = file.choose(), header =
TRUE)
```

Other options: read.csv(), write.csv(), read.csv2() and write.csv2() for comma-separated and semicolon-separated files,

Exercise

- Save the data frame with information about your friends in two formats: tab-separated and comma-separated.

- Read the files into R under different names.