

Intro to CASA Coding Group

25/02/2020

Outline for today

1. Purpose of group
2. Our tools: Languages/software we will learn
3. Examples of coding applications for speech
4. Download R, RStudio, Praat
5. Intro to RStudio
6. Topics for the term

Purpose of this group

Purpose of this group

- Develop skills that make it easier to do our job well as speech researchers
- Create a community that comes together to make it easier to learn this stuff

Purpose of this group

Computer Coding: Writing something in a language a computer can understand in order to tell the computer to do a specific thing or set of things.

Why bother telling a computer what to do when we can just do it ourselves?

Purpose of this group

Why bother telling a computer what to do when we can just do it ourselves?

- Automate repetitive tasks like...
 - opening/closing/saving files
- Minimize human error in data preparation
 - Renaming things in a spreadsheet
- Keep a careful log of how we did our analyses
 - Code = instructions
- “Reproducible research”
- Fun! (seriously!)

Our tools: Logistics

Project Website

- Website: <https://casa-lab.com/coding-group/>
- Slack channel: <https://casa-lab-ub.slack.com/>
 - [Invite to join Slack channel](#) (I will send this out via email after today)

Our tools: Languages & Software

First: Some terminology

Coding: Writing in a language a computer can understand

Scripting: A type of coding that tells a specific program exactly what actions to take

Programming: Writing code that serves to actually create another program (an app, software, etc)

Scripts: Text files containing code.

- Scripting, coding, and programming are sometimes used interchangeably

First: Some terminology

Functions: A certain named format of code that outlines a procedure. Often this allows several lines of code to be executed with a single line of code (by using the name of the function)

- For example, in Excel, you may use functions like `=sum(2, 2)`. `sum()` is the function that takes input (in this case, numbers), and performs an a specific action (adds them).

Calling: Invoke a function by using the name of the function and specifying parameters.

- For example: I “call the sum function” when I type it out with its inputs and execute it in excel.

Our tools

1. R and R Studio
2. Praat



- “R is a free software environment for statistical computing and graphics.”
- [Download here](#)



Studio[®]

- RStudio is a handy interface that helps you use R.
- [Download Desktop version](#)



Praat

- “Doing phonetics by computer”: Praat is a powerful software program that also has its own specialized language for writing scripts
- Praat = “Speech” in Dutch
- Looks like it hasn’t been updated since 1995 but it has and it’s great
- “World’s worst programming language”
 - *don’t let the haters get you down*
- [Download here](#)

Examples of coding applications for speech research

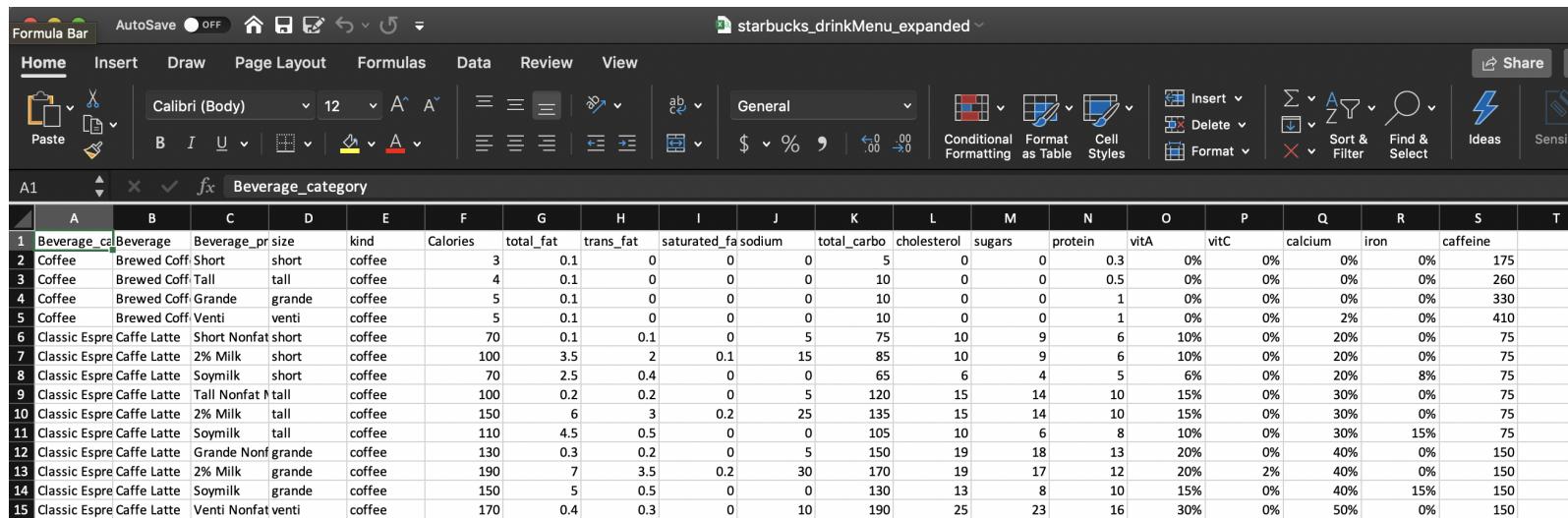
1. Data preparation in

Example: Starbucks data

1. Start with a data set you have in Excel
2. "Read" it into R
3. Do things to it like...
 - Instantly calculate means values

Let's look together

1. Data preparation: Raw data



The screenshot shows a Microsoft Excel spreadsheet titled "starbucks_drinkMenu_expanded". The data is organized into a table with 15 rows and 21 columns. The columns are labeled from A to T, and the rows are numbered 1 to 15. The first row contains the column headers: Beverage_cat, Beverage, Beverage_pr, size, kind, Calories, total_fat, trans_fat, saturated_fa, sodium, total_carbo, cholesterol, sugars, protein, vitA, vitC, calcium, iron, and caffeine. The data rows represent various Starbucks drink items, such as different sizes of coffee, lattes, and espressos, with their respective nutritional information and caffeine content.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	Beverage_cat	Beverage	Beverage_pr	size	kind	Calories	total_fat	trans_fat	saturated_fa	sodium	total_carbo	cholesterol	sugars	protein	vitA	vitC	calcium	iron	caffeine	
2	Coffee	Brewed Coff	Short	short	coffee	3	0.1	0	0	0	5	0	0	0.3	0%	0%	0%	0%	0%	175
3	Coffee	Brewed Coff	Tall	tall	coffee	4	0.1	0	0	0	10	0	0	0.5	0%	0%	0%	0%	0%	260
4	Coffee	Brewed Coff	Grande	grande	coffee	5	0.1	0	0	0	10	0	0	1	0%	0%	0%	0%	0%	330
5	Coffee	Brewed Coff	Venti	venti	coffee	5	0.1	0	0	0	10	0	0	1	0%	0%	2%	0%	0%	410
6	Classic Espre	Caffe Latte	Short	Nonfat short	coffee	70	0.1	0.1	0	5	75	10	9	6	10%	0%	20%	0%	0%	75
7	Classic Espre	Caffe Latte	2% Milk	short	coffee	100	3.5	2	0.1	15	85	10	9	6	10%	0%	20%	0%	0%	75
8	Classic Espre	Caffe Latte	Soymilk	short	coffee	70	2.5	0.4	0	0	65	6	4	5	6%	0%	20%	8%	0%	75
9	Classic Espre	Caffe Latte	Tall	Nonfat tall	coffee	100	0.2	0.2	0	5	120	15	14	10	15%	0%	30%	0%	0%	75
10	Classic Espre	Caffe Latte	2% Milk	tall	coffee	150	6	3	0.2	25	135	15	14	10	15%	0%	30%	0%	0%	75
11	Classic Espre	Caffe Latte	Soymilk	tall	coffee	110	4.5	0.5	0	0	105	10	6	8	10%	0%	30%	15%	0%	75
12	Classic Espre	Caffe Latte	Grande Nonfat	grande	coffee	130	0.3	0.2	0	5	150	19	18	13	20%	0%	40%	0%	0%	150
13	Classic Espre	Caffe Latte	2% Milk	grande	coffee	190	7	3.5	0.2	30	170	19	17	12	20%	2%	40%	0%	0%	150
14	Classic Espre	Caffe Latte	Soymilk	grande	coffee	150	5	0.5	0	0	130	13	8	10	15%	0%	40%	15%	0%	150
15	Classic Espre	Caffe Latte	Venti Nonfat	venti	coffee	170	0.4	0.3	0	10	190	25	23	16	30%	0%	50%	0%	0%	150

1. Data preparation: Data prep script

1_prep_data.R

```
#####
# Helper script for analyzing Starbucks drink data
#####

# Setup ----

# Load packages that contain functions we will use
library(tidyverse)
library(plyr)

# Load data ----
starbucks <- read.csv("1_materials/starbucks_drinkMenu_expanded.csv")
```

1. Data preparation: Data prep script (Continued)

1_prep_data.R

```
# Create new columns ----
# Create a "caffeine" column that is numeric
starbucks <- starbucks %>%
  mutate(caffeine = revalue(caffeine, replace = c(
    "varies" = NA, "Varies" = NA)),
    caffeine_num = as.numeric(as.character(caffeine)))

# Is caffeine content over 100 mg? If so, label it "YES", otherwise, "NO"
starbucks <- starbucks %>%
  mutate(too_much_caffeine = ifelse(caffeine_num > 100, "YES", "NO"))

starbucks %>% select(caffeine_num, too_much_caffeine) %>% head()
```

2. Data visualization in

2_figures.R

3. Writing in

Using R Markdown to write:

- Notes & reports
- Papers, articles, theses
- Presentations (like this one!)
- Websites, blog posts!

R Markdown allows you to incorporate *code* AND regular text using simple “markdown” syntax (more on that later).

4. Automating repetitive tasks in

For example...

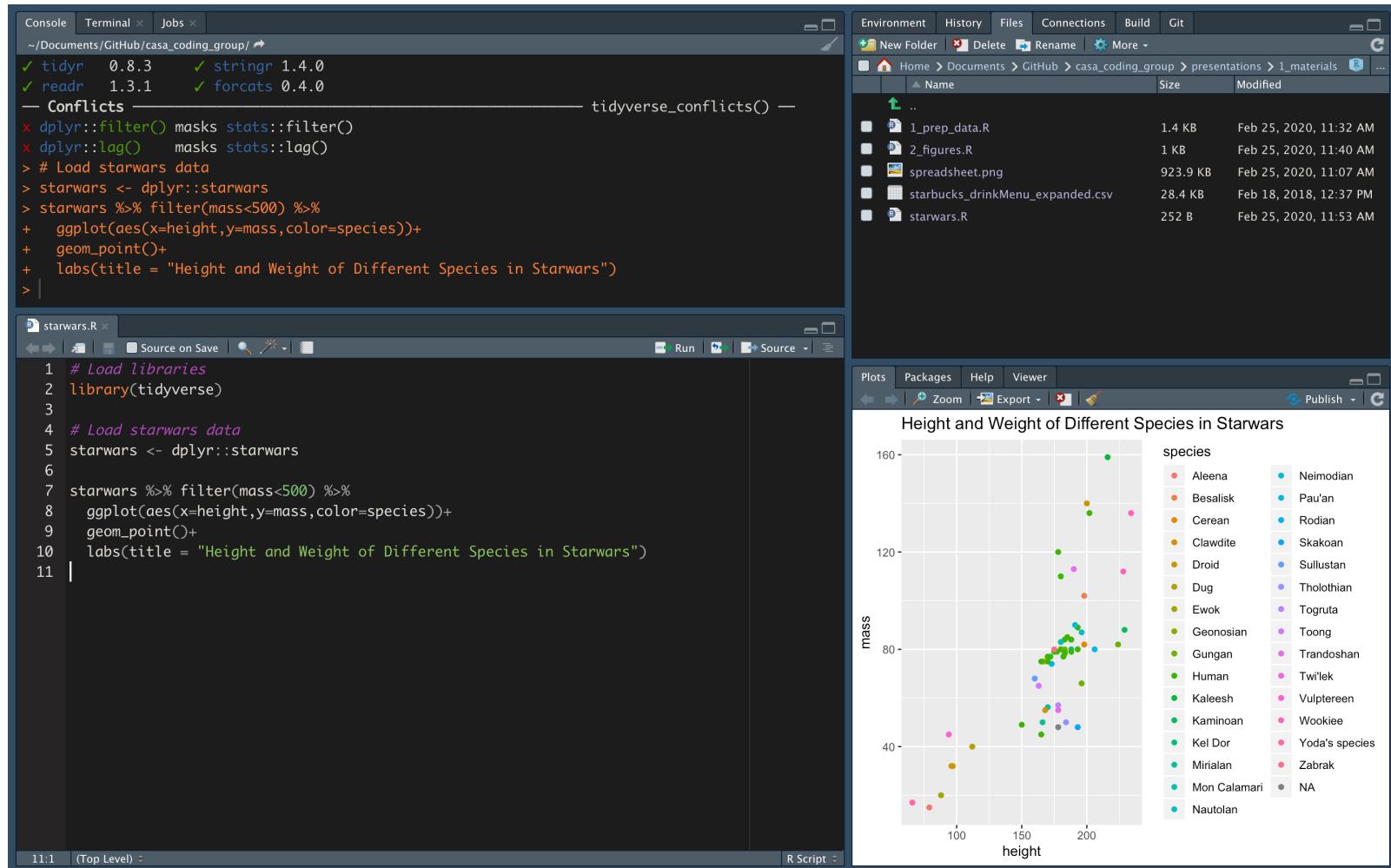
- Automatically create TextGrids for all .wav files in a directory
- Automatically adjust Praat TextGrid boundaries for all files in a directory

5. Running experiments in

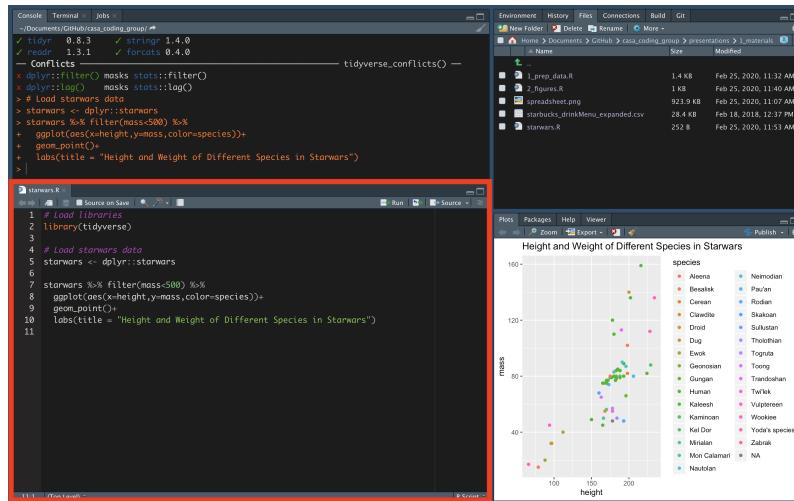
Intelligibility experiment in Praat

Intro to RStudio

RStudio layout

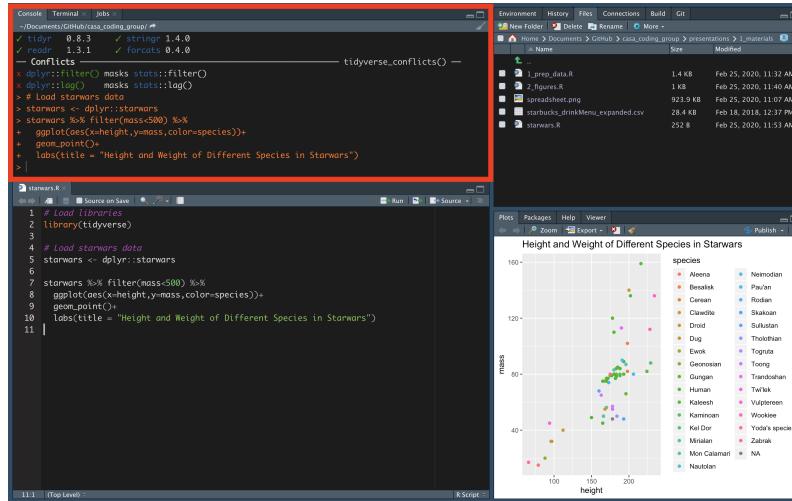


RStudio layout: Source pane



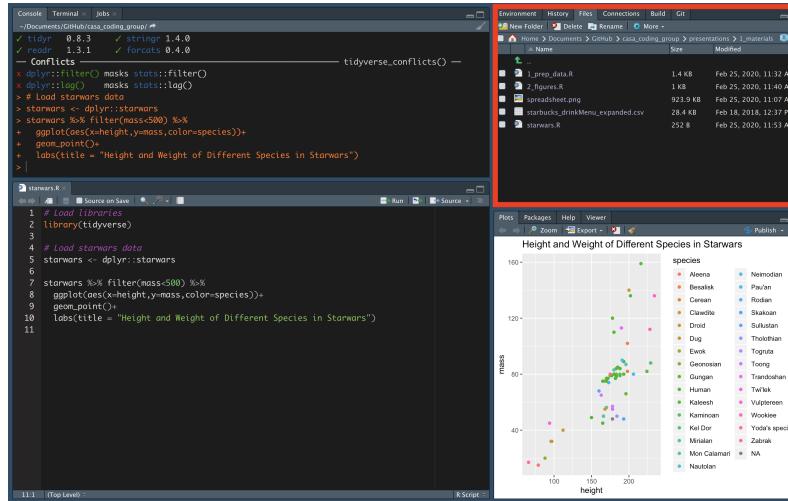
This is where you'll edit and run your scripts.

RStudio layout: Console pane



This is where code, error messages, warnings, etc. show up when you run code

RStudio layout: Files/Environment pane

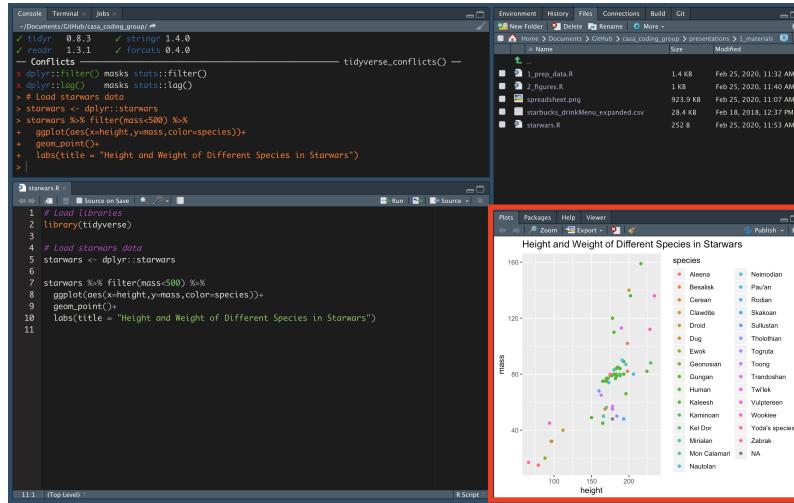


Here you can see...

1. Files in your directory ("Files")
2. Variables in your environment ("Environment")

- This is anything you have created in R.
- Saving your work to a script allows you to recreate these variables again later.

RStudio layout: Plots/Packages/Help/Viewer pane



- This is where plots you create will show up when you call them (automatically in “Plots”)
- You can also...
 - search help documentation (“Help”)
 - search for packages (“Packages”)

Topics for the term

My thoughts:

- Next three meet ups: R
- Final meet ups: Praat

Potential meet up flow: **Basic skill** + **fun skill** per meet up

- **Intro to R/R Markdown** + **Using emoji in R**
- **Cleaning data** + **Making boxplots**
- **Making figures in R** + **Making animated figures in R!**

Break up into small groups for a couple of minutes to discuss what you would like to see at this group

Schedule for the rest of the term

Date	Time	Location	Topic
2/25	4pm	Cary Hall 42	Intro to group + RStudio
3/10	4pm	TBD	TBD
3/24	4pm	TBD	TBD
4/7	4pm	TBD	TBD
4/21	4pm	TBD	TBD
5/5	4pm	TBD	TBD

DO TRY THIS AT HOME



Intro to R

Try this:

1. Sign up for an account on [udemy.com](https://www.udemy.com)
2. Sign up for the “[R basics: R programming language](https://www.udemy.com/course/r-basics/)” course on udemy
3. Watch videos 1, 2, 3, and 9.
 - 1 - R basics (3 min)
 - 2 - A walkthrough of downloading R & Rstudio (5 min)
 - 3 - the Rstudio interface (19 min)
 - 9 - Three common mistakes in R beginners (11 min)