Bericht 3

Daniela Palleschi

2024-06-25

Inhaltsverzeichnis

Bericht 3

The purpose of this report is to consolidate what we have learned over the last few weeks with what we have learned previously. You will load a dataset from a published paper that investigates voiced-onset times (Sonderegger, 2023) that is available on Moodle. If you are not enrolled in this course on Moodle, the dataset is also freely available on the Open Science Framework at https://osf.io/cmh3p.

You can create your own Quarto script from scratch or use a blank script I created with the questions and corresponding headings. You can find this script on Moodle or online here.

1 Set-up

This report is shared with you as an incomplete Quarto script. Your task is to follow the instructions to complete the script. The instructions are preceded by a `>`, in order to differentiate them from your own text that you will include to describe your data. Please keep the instructions in the script so that it is clear what each task is trying to achieve. You may delete this paragraph if you wise.

1.1 Render

Before continuing, check that you can render the script as PDF. It should run even though below there are incomplete code chunks, because the YAML includes `eval: false`.

1.2 YAML

Change the YAML so that:

- `eval: true`
- `author: "YOUR NAME"`

Now you will not be able to render the document until all the code chunks below are completed, because there are incomplete code chunks below and you have set `eval: true`.

1.3 Packages

Load in the packages `tidyverse`, `gghalves`, `patchwork`, and `here`.

1.4 Load data

Below is a code chunk that loads in a data set from Sonderegger et al. (2017) and discussed in Sonderegger (2023). **Wichtig:** Sie müssen die Daten bereits heruntergeladen haben, entweder von Moodle oder vom Open Science Framework, und sie in Ihrem 'daten'-Ordner gespeichert haben.

Listing 1 `Copy and run this code`

```
df_vot <-
  read_csv(here::here("daten", "vot_rmld_2023.csv"))
```

The dataset contains voice onset times in milliseconds (VOTs; `vot`) for word-initial stop consonants (`phone`: p/t/k/b/d/g) in speech from native English speaker contestants in a reality television show, Big Brother UK (2008, Season 9). Importantly, "[voice onset time] is the primary acoustic cue in English signaling whether a stop is phonologically [voiced] (b/d/g) or [voiceless] (b/d/g)" (Sonderegger, 2023, p. 98). We will be looking at differences in VOT as a function of voicing and place of articulation (`place`: alveolar/labial/velar) across a subset of contestants.

1.5 Data exploration

Explore the dataset however you like to get familiar with it. You do not need to show your work for this task, but if you do please keep in mind that if you use code in the script that prints the data (e.g., running simply `df_vot`), it will produce all the rows (n = 25154) when you render the document, adding many pages to your rendered document. For this reason, try using `head()` to print only the first 6 rows (first discussed in Chatper 4), or simply opening the dataset viewer by double clicking on its name in the 'Environment' pane.

2 Base R

In this section you will interpret and write code in base R and/or the tidyverse.

2.1 Filter

Using base R, filter the data to only include the speakers `rex`, `michael`, `sara`, and `lisa`. Tip: you will want to use `[,]`, where rows are defined on the left and columns on the right. We want all columns, so you don't need to include anything to the right of the comma.

Listing 2 `Complete this code`

```
df_vot <-
  ...
```

2.2 Code description

Describe in words what the following base R code achieves:

```
df_vot[df_vot$vot > 75 & df_vot$voicing == "voiced", 1]
```

2.3 base R-tidyverse 'translation'

Now write the same code in tidyverse and print the result.

Listing 3 `Complete this code`

```
df_vot |>
  ...
```

2.4 base R summary

Using base R, create a summary called `sum_vot` which contains the mean and standard deviation voice onset times, and number of observations (using `length()`) of the whole dataset.

Listing 4 `Complete this code`

```
sum_vot <-
  ...
```

2.5 Tidyverse summary

Now, using the tidyverse, do the same but produce these values by speaker, phone, place, and voicing. Call it `sum_speaker`.

Listing 5 `Complete this code`

```
sum_speaker <-
  ...
```

3 Plots

For these tasks you will produce two plots using functions and arguments covered in the Data Visualisation 4 session.

3.1 Multi-part plot

Produce a multi-part plot per speaker (x-axis) voice-onset times (y-axis), which contains a scatter plot, boxplot, and dodged histogram, with facets per participant arranged in a single row (Hint: `facet_wrap()` has an argument `nrow =` that can define the number of rows). See Figure in Section 13.1.4 from the web-book for inspiration. Call this plot `fig_vot`.

Listing 6 `Complete this code`

```
fig_vot <-
  ...
```

3.2 Errorbar plot

Produce an errorbar plot of the by-speaker summary, with voicing on the x-axis, colour, shape, and group by place of articulation, and a facet per participant, arranged in a single row (Hint: `facet_wrap()` has an argument `nrow =` that can define the number of rows). Include `geom_line()`, with the local aesthetic `linetype` by place. Make sure you use `position_dodge()` to make sure the errorbars do not overlap. Once you have created the plot, add labels and theme customisations to match **?@fig-errorbar** as best you can. Call this plot `fig_vot_error`.

Listing 7 `Complete this code`

```
fig_vot_error <-
  ...
```

3.3 Print plots

Print your plots one on top of the other using the `patchwork` package. Include the labels `A` and `B` (we saw how to do this twice: in Chapter 10 and Chapter 13). They should look something like Abbildung 1 A and B.

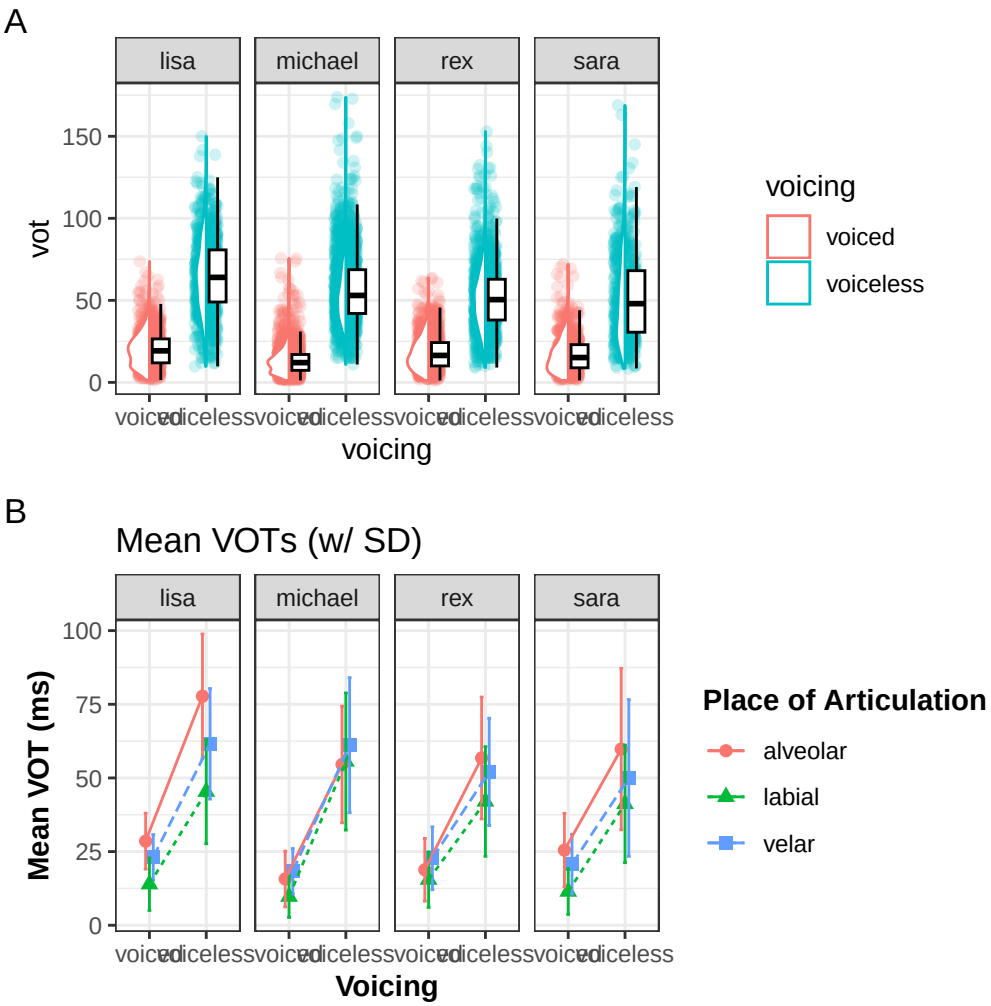

Abbildung 1: Reproduce this plot.

3.4 Plot interpretation

Interpret your plots. Consider the differences between speakers, voicing, and of place of articulation (and if any of these differences varied as a function of another

variable, e.g., are the differences of voicing and place of articulation the same for all speakers?).

4 Reflection

Reflect on your journey through this course. What did you find particularly interesting? What was difficult? What do you still struggle with? Do you see yourself applying what you've learned in future work? How might the course be improved moving forward?

References

Sonderegger, M. (2023). *Regression Modeling for Linguistic Data*.

Sonderegger, M., Bane, M., & Graff, P. (2017). The Medium-Term Dynamics of Accents on Reality Television. *Language*, *93*(3), 598–640. https://doi.org/10.1353/lan.2017.0038