

Bericht 3 - ANWEISUNGEN

Konsolidierung der neuen Kenntnisse

Daniela Palleschi

2024-02-07

Inhaltsverzeichnis

Bericht 3	1
1 Einrichten	2
1.1 Rendern	2
1.2 YAML	2
1.3 Pakete	2
1.4 Daten laden	3
1.5 Datenuntersuchung	3
2 Base R	3
2.1 Filter	3
2.2 Codebeschreibung	4
2.3 base R-tidyverse ‘Übersetzung’	4
2.4 base R Zusammenfassung	4
2.5 Zusammenfassen mit tidyverse	4
3 Plots	5
3.1 Mehrteilige Darstellung	5
3.2 Fehlerbalken-Diagramm	5
3.3 Plots drucken	5
3.4 Interpretation des Plots	7
4 Reflexion	7
Referenzen	7

Bericht 3

Der Zweck dieses Berichts ist es, das, was wir in den letzten Wochen gelernt haben, mit dem zu konsolidieren, was wir zuvor gelernt haben. Sie werden einen Datensatz aus einer

veröffentlichten Arbeit laden, die sich mit der Untersuchung von Stimmbruchzeiten befasst (Sonderegger, Bane, und Graff 2017; Sonderegger 2023) und die auf Moodle verfügbar ist. Wenn Sie nicht in diesem Kurs auf Moodle eingeschrieben sind, ist der Datensatz auch frei auf dem Open Science Framework unter <https://osf.io/cmh3p> verfügbar.

Sie können Ihr eigenes Quarto-Skript von Grund auf erstellen oder ein leeres Skript verwenden, das ich mit den Fragen und den entsprechenden Überschriften erstellt habe. Sie können dieses Skript auf Moodle oder online [hier](#) finden.

1 Einrichten

Dieser Bericht wird Ihnen als unvollständiges Quarto-Skript zur Verfügung gestellt. Ihre Aufgabe ist es, die Anweisungen zu befolgen, um das Skript zu vervollständigen. Sie finden das Skript auf Moodle, oder [hier](#).

1.1 Rendern

Bevor Sie fortfahren, überprüfen Sie, ob Sie das Skript als PDF rendern können. Es sollte laufen, auch wenn unten unvollständige Codeabschnitte stehen, da die YAML `eval: false` enthält.

1.2 YAML

Ändern Sie die YAML so, dass:

- `eval: true`
- `Autor: "IHR NAME"`

Nun können Sie das Dokument erst rendern, wenn alle Code-Bausteine darunter fertiggestellt sind, weil darunter unvollständige Code-Bausteine sind und Sie `eval: true` gesetzt haben.

1.3 Pakete

Laden Sie die Pakete `tidyverse`, `gghalves`, `patchwork`, und `here` ein.

Listing 1 Kopieren Sie diesen Code und führen Sie ihn aus

```
df_vot <-  
  read_csv(here::here("daten", "vot_rmlid_2023.csv"))
```

1.4 Daten laden

Nachfolgend ein Codeabschnitt, der einen Datensatz aus Sonderegger, Bane, und Graff (2017) lädt und in Sonderegger (2023) besprochen wird. **Wichtig:** Sie müssen die Daten bereits heruntergeladen haben, entweder von Moodle oder vom [Open Science Framework](#), und sie in Ihrem ‘daten’-Ordner gespeichert haben.

Der Datensatz enthält Stimmeinsatzzeiten in Millisekunden (VOTs; `vot`) für wortanfängliche Stoppkonsonanten (`phone`: p/t/k/b/d/g) in der Rede von englischen Muttersprachlern, die an der Reality-TV-Show Big Brother UK (2008, Staffel 9) teilnahmen. Wichtig ist, dass “[der Zeitpunkt des Stimmeinsatzes] im Englischen der primäre akustische Hinweis darauf ist, ob ein Stop phonologisch [stimmhaft] (b/d/g) oder [stimmlos] (p/t/k) ist” (Sonderegger 2023, 98). Wir werden die Unterschiede in der VOT als Funktion der Stimmhaftigkeit und des Artikulationsortes (`place`: alveolar/labial/velar) in einer Untergruppe von Teilnehmern untersuchen.

1.5 Datenuntersuchung

Machen Sie sich mit dem Datensatz vertraut, wie Sie möchten. Sie müssen Ihre Arbeit für diese Aufgabe nicht zeigen, aber wenn Sie es tun, bedenken Sie bitte, dass, wenn Sie Code im Skript verwenden, der die Daten ausgibt (z.B. einfach `df_vot`), er alle Zeilen ($n = 25154$) erzeugt, wenn Sie das Dokument rendern, was viele Seiten zu Ihrem gerenderten Dokument hinzufügt. Versuchen Sie deshalb, mit `head()` nur die ersten 6 Zeilen auszudrucken (zuerst in [Chapter 4](#) besprochen), oder öffnen Sie einfach den Datensatzbetrachter durch Doppelklick auf seinen Namen im Fenster ‘Umgebung’ (‘Environment’).

2 Base R

In diesem Abschnitt werden Sie Code in Base R und/oder Tidyverse interpretieren und schreiben.

2.1 Filter

Filtern Sie mit base R die Daten so, dass nur die Sprecher `rex`, `michael`, `sara` und `lisa` enthalten sind. Tipp: Verwenden Sie `[,]`, wobei Zeilen links und Spalten rechts definiert sind. Wir wollen alle Spalten, also müssen Sie nichts rechts vom Komma einfügen.

Listing 2 Vervollständigen Sie diesen Code

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

2.2 Codebeschreibung

Beschreiben Sie in Worten, was der folgende R-Basiscode bewirkt:

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

2.3 base R-tidyverse ‘Übersetzung’

Schreiben Sie nun den gleichen Code in tidyverse und drucken Sie das Ergebnis aus.

Listing 3 Vervollständigen Sie diesen Code

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

2.4 base R Zusammenfassung

Erstellen Sie mit base R eine Zusammenfassung mit dem Namen `sum_vot`, die den Mittelwert und die Standardabweichung der Stimmausgangszeiten sowie die Anzahl der Beobachtungen (mit `length()`) des gesamten Datensatzes enthält.

Listing 4 Vervollständigen Sie diesen Code

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

2.5 Zusammenfassen mit tidyverse

Machen Sie nun dasselbe mit tidyverse, aber produzieren Sie diese Werte nach Sprecher, Telefon, Ort und Stimmlage. Nennen Sie es `sum_speaker`.

Listing 5 Vervollständigen Sie diesen Code

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

3 Plots

Für diese Aufgaben werden Sie zwei Diagramme erstellen, indem Sie Funktionen und Argumente verwenden, die in der Sitzung Datenvisualisierung 4 behandelt wurden.

3.1 Mehrteilige Darstellung

Erstellen Sie einen mehrteiligen Plot pro Sprecher (x-Achse) und Sprechzeit (y-Achse), der einen Scatter Plot, Boxplot und ein Dodged Histogram enthält, wobei die Facetten pro Teilnehmer in einer einzigen Zeile angeordnet sind (Hinweis: `facet_wrap()` hat ein Argument `nrow =`, mit dem die Anzahl der Zeilen definiert werden kann). Siehe [Abbildung in Abschnitt 13.1.4 aus dem Web-Book](#) zur Inspiration. Nennen Sie diese Darstellung `fig_vot`.

Listing 6 Vervollständigen Sie diesen Code

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

3.2 Fehlerbalken-Diagramm

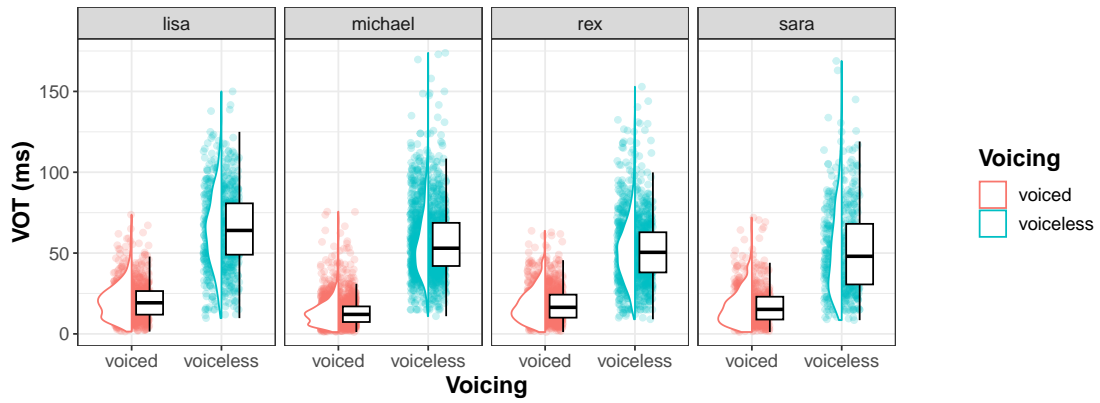
Erzeugen Sie ein Fehlerdiagramm der Zusammenfassung nach Sprechern, mit Intonation auf der x-Achse, Farbe, Form und Gruppe nach Ort der Artikulation und einer Facette pro Teilnehmer, angeordnet in einer einzigen Zeile (Hinweis: `facet_wrap()` hat ein Argument `nrow =`, das die Anzahl der Zeilen festlegen kann). Fügen Sie `geom_line()` ein, mit der lokalen Ästhetik `linetype` an Stelle. Stellen Sie sicher, dass Sie `position_dodge()` verwenden, um sicherzustellen, dass sich die Fehlerbalken nicht überlappen. Sobald Sie die Darstellung erstellt haben, fügen Sie Beschriftungen und Themenanpassungen hinzu, die so gut wie möglich zu `?@fig-errorbar` passen. Nennen Sie diese Darstellung `fig_vot_error`.

3.3 Plots drucken

Drucken Sie Ihre Diagramme übereinander mit dem Paket `patchwork`. Fügen Sie die Beschriftungen A und B ein (wir haben gesehen, wie man das zweimal macht: in [Kapitel 10](#) und [Kapitel 13](#)). Sie sollten etwa so aussehen wie [Abbildung 1 A und B](#).

A

Distribution of VOTs



B

Mean VOTs (w/ SD)

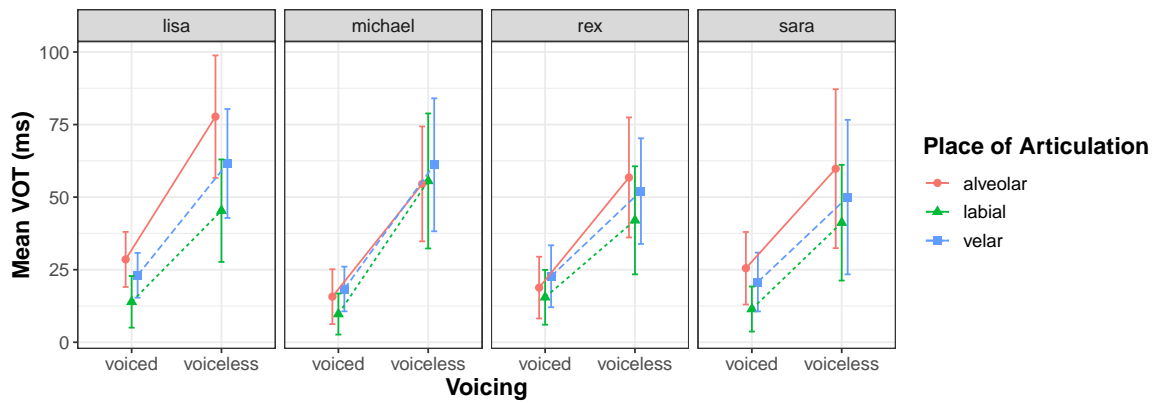


Abbildung 1: Reproduce this plot.

Listing 7 Vervollständigen Sie diesen Code

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

3.4 Interpretation des Plots

Interpretieren Sie Ihre Diagramme. Betrachten Sie die Unterschiede zwischen den Sprechern, die Intonation und den Ort der Artikulation (und ob einer dieser Unterschiede in Abhängigkeit von einer anderen Variable variiert, z. B. ob die Unterschiede bei Intonation und Ort der Artikulation für alle Sprecher gleich sind).

4 Reflexion

Reflektieren Sie über Ihre Reise durch diesen Kurs. Was haben Sie besonders interessant gefunden? Was war schwierig? Womit tun Sie sich noch schwer? Könnten Sie sich vorstellen, das Gelernte in Zukunft bei Ihrer Arbeit anzuwenden? Wie könnte der Kurs in Zukunft verbessert werden?

Referenzen

- Sonderegger, Morgan. 2023. *Regression Modeling for Linguistic Data*.
- Sonderegger, Morgan, Max Bane, und Peter Graff. 2017. „The Medium-Term Dynamics of Accents on Reality Television“. *Language* 93 (3): 598–640. <https://doi.org/10.1353/lan.2017.0038>.