Empirisch-Wissenschaftliches Arbeiten

Übung zur computergestützen Datenanalyse

© Prof. Dr. Stephan Huber

17. Juli 2024

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Vorwort

Eine PDF-Version dieser Notizen ist hier verfügbar.

Bitte beachten Sie, dass der Inhalt der PDF-Version identisch ist, sie jedoch nicht für das PDF-Format optimiert wurde. Daher könnten einige Teile nicht wie beabsichtigt erscheinen.

Wikipedia sagt:

> "Die Psychologie [...] ist eine empirische Wissenschaft"

Diese Unterlagen helfen...

- die Abfolge und die Inhalte der *Übung zur computergestützen Datenanalyse* zu überblicken,
- die Übungsaufgaben zu verstehen und zu bearbeiten und
- die Projektarbeit der Veranstaltung *Empirisch-Wissenschaftliches Arbeiten* erfolgreich zu gestalten.

Die Übung vermittelt...

- Kenntnisse der Programmiersprache R welche eine wissenschaftliche Datenbearbeitung und Datenanalyse ermöglichen.
- Kenntnisse zum programmbasierten Verfassen von wissenschaftlichen Arbeiten (Aufsätze, Bücher, Arbeitspapiere, Hausarbeiten).

Studierende lernen...

- Daten mit der Programmiersprache R und mit Hilfe der integrierten Entwicklungsumgebung RStudio einzulesen, zu bearbeiten und empirisch auszuwerten.
- Empirische Ergebnisse in ein publikationswürdiges Format zu übertragen.
- Einen APA konformes Manuskript mit Quarto, bzw. (R)Markdown, zu erstellen und dies entsprechend zu publizieren.
- Literatur entsprechend wählbaren Zitationsregeln unter Verwendung von Quarto und BibTeX in einen Aufsatz einzuarbeiten.

Studierende sollen...

- Die angeführte Literatur studieren: Ohne eigenständige Vor- und Nachbereitung lassen sich die Programmierkenntnisse nicht erlernen.
- Aktiv um Hilfe bitten: Wenn etwas unklar ist, kann ich individuell während des Kurses versuchen zu helfen. Für eine intensivere Betreuung, bitte ich mich zu kontaktieren, in die Sprechstunde zu kommen, oder eine außerordentliche Sprechstunde zu vereinbaren. Dies ist möglich und erwünscht.

- Inhaltliche Fragen und Wünsche jederzeit kommunizieren. Es besteht die Möglichkeit diese in das Curriculum aufzunehmen.

Liebe Studierende,

das Erlernen einer Programmiersprache in Verbindung mit empirischen Arbeiten ist eine Herausforderung die Vielen keinen Spaß macht. So ist es nur Verständlich, dass die Sinnhaftigkeit dieses Kurses teilweise von Studierenden angezweifelt wird. Tätigkeiten die keinen Spaß machen, sollten sinnstiftend sein oder zumindest ein monetäres Einkommen sichern. Da das Vorhandensein von empirischen Kenntnissen und einer Programmiersprache in einem Lebenslauf zweifelsfrei in der heutigen Zeit die Vermittlungsfähigkeit und die Verhandlungsposition am Arbeitsmarkt wesentlich verbessern, will ich mich hier kurz bemühen, die Sinnhaftigkeit zu thematisieren.

Ich verstehe die Abneigung gegenüber diesen Kurs: Viele haben sich nicht für ein Studium der Psychologie entschieden, um empirische Methoden und deren computergestützte Umsetzung zu erlernen. In der modernen Welt aber, insbesondere in der psychologischen Forschung, ist ein Verständnis von empirischen Methoden sowie deren computergestützten Umsetzung die praktische Voraussetzung zum Erkenntnisgewinns. Ohne dieses Verständnis verharrt man bei rein theoretische und philosophische Überlegungen ohne jede Evidenz. Eine professionell agierende Psychologin und Psychologe, sollte die Fähigkeit besitzen die Literatur in seinem Fach zu begreifen sowie in der Lage ein die Ergebnisse kritisch zu hinterfragen und/oder zu überprüfen.

Ich bemühe mich, die Veranstaltung so attraktiv wie möglich zu gestalten. Ich biete...

- ein ausführliches Skript zur Programmiersprache R an, welches
  - eine Batterie an Übungsaufgaben mit Lösungskripten und
  - eine Vielzahl an interaktive Übungen zum eigenständigen bearbeiten enthält.
- dieses Skript, welches
  - Psychologie-spezifische empirische Inhalte aufgreift und
  - Software vorstellt, welche die Erstellung der Projektarbeit erleichtert.
- mündliche Erklärungen in der Veranstaltung.
- die Möglichkeit spezifische Fragen zu stellen und Unklarheiten anzusprechen.
- individuelle Betreuung während und außerhalb der Sprechstunde.

Wenn sie Vorschläge und Wünsche bezüglich der Inhalte oder der didaktischen Aufbereitung haben, bitte ich diese auszusprechen. Konstruktive Kritik ist sehr willkommen. Ich nehme diese an und ernst. Ob Sie diesen Kurs letztendlich als gelungen betrachten, ist ihrer Wahrnehmung überlassen. Bevor Sie den Kurs aber schlecht evaluieren, bitte ich sie um Folgendes: Fragen Sie sich, ob ihr Wille und ihr Wunsch ausgeprägt genug waren, um sich ernsthaft mit den Inhalten

und den Angeboten auseinanderzusetzen und ob sie mir evtl. die Gelegenheit gegeben haben auf Ihre Wünsche einzugehen.

Abschließend wünsche ich Ihnen viel Freude mit dem Kurs und den angebotenen Unterlagen und Inhalten. Ich freue mich, diesen Kurs halten zu dürfen und zu können. Es ist mir stets eine Freude, den anwesenden Studierenden R, Quarto, BibTeX und Co. erklären zu können. Ich wünsche mir, möglichst Viele mit den dargebotenen Inhalten, das Studium zu bereichern und die Bearbeitung der Projektarbeit sowie der Abschlußarbeit zu erleichtern.

Ihr
Stephan Huber

Teil I.

Einleitung

1. Die Programmiersprache R

Ich bitte Sie, studieren sie das Skript *How to use R for data science* [Huber, 2024c].

In den ersten Wochen werden wir uns ausschließlich damit beschäftigen, die Programmiersprache R zu erlernen. Das ist ähnlich mühsam wie das Erlernen einer wirklichen Sprache. Wer keine Lust darauf hat, wird es schwer haben. Ich beispielsweise hatte in der Schule überhaupt keine Lust auf Englisch und Latein. Dementsprechend schlecht waren meine Noten. Ich musste die siebte Klasse wiederholen und bis zum Abitur waren Sprachen für mich ein nötiges Übel. Erst als ich im Studium sah, dass praktisch alle relevanten und für mich interessanten Artikel und Bücher in englischer Sprache verfasst sind, machte das Erlernen der Sprache einen Sinn für mich. Jetzt lehre ich abseits dieses Kurses ausschließlich auf Englisch und publiziere in englischer Sprache. Interesse und Freude sind mächtige Katalysatoren für Erfolg.

Das Schreiben von Code ist für die meisten Studierenden Neuland. Studierende im Jahr 2024 sind zumeist mit dem Smartphone aufgewachsen und demnach sind Sie es gewohnt, ihre Geräte (Smartphone, Tablet, Desktop-PC) ohne zur Hilfenahme einer Programmiersprache zu steuern. Das ist wunderbar: Die grafische Benutzeroberfläche heutzutage erlaubt eine effiziente und intuitive Art der Steuerung mit der Computermaus, durch Wischen, Tippen oder durch Spracheingabe. Leider hat diese Art der Steuerung massive Nachteile beim wissenschaftlich orientierten Arbeiten mit Daten. Insbesondere was die Reproduzierbarkeit der Ergebnisse und die Flexibilität des Arbeitsprozesses anbelangt, stößt man bei Applikationen ohne Code an Grenzen. Die Vor- und Nachteile von Script-basierten Arbeiten werden im Kapitel *The limitations of no-code applications* [Huber, 2024c] ausführlich erläutert.

Zusammenfassend sollten Studierende nach den ersten 5-6 Unterrichtseinheiten folgendes getan haben beziehungsweise erlernt haben:

- Installation von
  - R,
  - RStudio und der
  - gängisten Pakete.
- Wissen über…
  - den Aufbau von R Skripten.
  - die Verwendung von Funktionen, Objekten und Pakete in R.
  - die grundsätzlichen Eigenheiten der Programmiersprache R.
  - das Ausführen von Code (`Ctrl`+`Enter`, Klicken von `Run`, oder durch die Funktion `source()`).
  - die Verwendung von Pipes mit dem Pipe Operator (`|>`).
  - die Verwendung von logischen und relativen Operatoren.
  - die Funktionen des Pakets `dplyr` (`filter()`, `select()`, `mutate()`, `summarise()`, etc.)

2. Wissenschaftliche Texte schreiben

Studierende verwenden zum Verfassen wissenschaftlicher Texte gerne Microsoft Word, Apples Pages oder LibreOffice. Diese Textverarbeitungsprogramme zeigen das Dokumentenlayout bereits während des Schreibens an. Dies wird auch als "What you see is what you get" (WYSIWYG) Prinzip bezeichnet. Dieses Prinzip und die entsprechenden Anwendungen sind weit verbreitet und erscheinen vielen als alternativlos. Dies trifft jedoch keineswegs zu: Es gibt eine Vielzahl alternativer Textsatz-Systeme wie LaTeX, Markdown, R Markdown und Quarto, die beachtenswerte Vorteile bieten. Nicht ohne Grund nutzen viele professionell arbeitende Wissenschaftler und Publizisten diese Alternativen. Eine große Anzahl von Doktorarbeiten und wissenschaftlichen Aufsätzen wird mit LaTeX verfasst, und nahezu alle Herausgeber und Verlage arbeiten mit codebasierten Alternativen, die nicht dem WYSIWYG-Prinzip folgen.

Bei codebasierten Alternativen werden die Angaben zum Layout entweder an den Anfang des Textes oder direkt in den Fließtext eingefügt. Das endgültige Dokument wird erst nach der Umwandlung (auch Kompilieren oder Rendern genannt) in ein Dokumentformat wie PDF sichtbar. Dies mag zunächst gewöhnungsbedürftig sein und sicherlich weniger intuitiv als eine WYSIWYG-Benutzeroberfläche. Doch die intuitivste Lösung ist nicht zwangsläufig die beste oder einfachste. Die vielen Studienarbeiten, die ich betreuen durfte, zeigen deutlich, dass die intuitiven Features von MS Word und Pages sich mittel- und langfristig oft als zeitaufwändig erweisen und Nutzer nur unzureichend dabei unterstützen, Fehler beim Verfassen von wissenschaftlichen Arbeiten zu vermeiden. Studierende, die sich gegen eine WYSIWYG-Office-Anwendung entscheiden, sind in der Regel weniger frustriert und erfolgreicher – zumindest trifft dies auf die von mir betreuten Arbeiten zu.

Codebasierte Anwendungen ermöglichen es den Schreibenden, sich auf die eigentliche Textarbeit zu konzentrieren. Die Feinheiten des Formats und die Einhaltung von Zitationsregeln werden größtenteils automatisch von der Software übernommen. Die notwendige Anfangsinvestition, sich etwa mit Quarto vertraut zu machen, macht sich schnell bezahlt, und die Qualität der wissenschaftlichen Texte verbessert sich spürbar.

In den folgenden Unterabschnitten werde ich zunächst die typische Nutzung von WYSIWYG-Anwendungen umreißen. Anschließend werde ich die Vorteile des codebasierten Verfassens von Texten am Beispiel von Quarto beleuchten, um dann zu erläutern, wie das Verfassen von Texten mit Quarto gelingen kann.

2.1. WYSIWYG Anwendungen

Die Nutzung klassischer Textverarbeitungsprogramme wie Microsoft Word oder Apple Pages zum Verfassen wissenschaftlicher Texte ist in der studentischen Welt weit verbreitet. Obwohl diese Programme für alltägliche Schreibprojekte benutzerfreundlich sind, erzeugen sie erheblichen Mehraufwand, um den Ansprüchen wissenschaftlicher Arbeit gerecht zu werden.

Ein erstes Problem ist die Einbindung von Literatur. Die korrekte Formatierung nach verschiedenen Zitierrichtlinien ist oft alles andere als intuitiv und Fehler treten leicht auf. Dies gilt insbesondere, wenn die von der Software bereitgestellten Zitat- und Bibliografiefunktionen

nicht oder nicht richtig genutzt werden. Anstelle externer Zitationsmanager zu nutzen und sich in deren Gebrauch einzuarbeiten, verfassen viele Studierende Zitate und Literaturlisten manuell. Dies führt erfahrungsgemäß zu zahlreichen kleinen und manchmal größeren Fehlern, die vermeidbar wären.

Ein weiterer Schwachpunkt von studentischen Arbeiten ist die Einhaltung spezifischer Formatierungsvorgaben. Akademische Institutionen und Journale fordern oft eine strenge Beachtung von Formatierungsrichtlinien, inklusive der Gestaltung von Titelseiten, Kopf- und Fußzeilen, Seitenrändern und Überschriftenhierarchien. Zwar bieten Word und Pages Vorlagen und Stile an, diese müssen jedoch für jedes Dokument individuell angepasst und oft aufgrund geringfügiger Änderungen im Text modifiziert werden. Wenn eine Formatanpassung erforderlich wird, ist dies meistens nur mit großem Aufwand möglich.

Das Einfügen empirischer Ergebnisse wie statistischer Daten und Grafiken stellt eine zusätzliche Hürde dar. In Word und Pages gestaltet sich der Vorgang häufig manuell: Forschungsdaten müssen aus Statistiksoftware exportiert, als Bilder abgespeichert und anschließend in das Dokument eingebunden werden. Ändert sich etwas an den Daten, so muss dieser mühsame Prozess wiederholt werden, was den Arbeitsaufwand signifikant erhöht und die Fehleranfälligkeit steigert.

2.2. Vorteile von codebasierten Anwendungen

Der traditionelle Ansatz zum Verfassen wissenschaftlicher Texte mittels MS Word oder Pages kann für Studierende zeitaufwendig und fehleranfällig sein. Im folgenden Abschnitt möchte ich Quarto (bzw. R Markdown) vorstellen, eine moderne Alternative, die folgende Vorteile bietet:

- Mit Quarto lassen sich mühelos verschiedene Ausgabeformate generieren. So kann derselbe Text beispielsweise als Website (HTML), Manuskript (PDF, DOCX), Buch (EPUB, PDF) oder in Form von Präsentationsfolien (PDF). Diese Flexibilität erlaubt es, sich eher auf den Inhalt als auf das Format zu konzentrieren.
- Die Formatierung kann in Quarto einfach geändert werden indem bestimmte Vorlagen verwendet werden.
- Literaturreferenzen lassen sich unkompliziert einbinden, während die Einhaltung von Zitierregeln von der Software übernommen wird. Quartos Integration mit Zitationsverwaltungssystemen ermöglicht es, Literaturverweise und Bibliografien effizienter und konsistenter zu handhaben als beispielsweise in Word.
- Querverweise auf Abschnitte, Tabellen und Abbildungen lassen sich leicht erstellen.
- Die Datenanalyse und das Erstellen von Datenoutputs erfolgen direkt in Quarto. Dadurch sind dargestellte Grafiken und Tabellen stets aktuell und manuelles Nachbearbeiten entfällt, wodurch die Reproduzierbarkeit der Ergebnisse gewährleistet ist.
- Forschende können ihre Datenvisualisierungen ohne manuelle Zwischenschritte direkt in den Text einbetten.
- Versionskontrollsysteme wie Git erleichtern die Zusammenarbeit an wissenschaftlichen Dokumenten, da Änderungen nachverfolgbar sind und integriert werden können, ohne auf komplexe und konfliktträchtige Vergleichstools angewiesen zu sein.

> Lesetipp
>
> Für Interessierte empfehle ich den Onlinekurs *Introduction to Reproducible Publications with RStudio*, der explizit erläutert, wie man empirisch nachvollziehbar arbeitet. Eine etwas kompaktere Einführung bietet Bauer and Landesvatter [2023] und das Standardwerk zum

Thema stammt von Gandrud [2020].

2.3. Einführung in Quarto

Quarto kann in RStudio genutzt werden, um APA-konforme Texte zu erstellen. Gehen Sie dazu bitte wie folgt vor:

- Installieren Sie R und R Studio.
- Installieren Sie Quarto folgendermaßen:

```
install.packages("quarto")
```

- Installieren Sie das Paket `tinytex`, um PDF-Dateien zu generieren:

```
install.packages("tinytex")
tinytex::install_tinytex()
```

- Es ist zudem ratsam, weitere Pakete zu installieren, die später benötigt werden könnten:

```
if (!require(pacman)) install.packages("pacman")
pacman::p_load(knitr, rmarkdown, papaja)
```

- Eignen Sie sich Kenntnisse in Markdown an. Markdown ist eine leichtgewichtige Markup-Language zur Formatierung von Klartext. Sie ist eine essenzielle Fähigkeit für die effektive Nutzung von Quarto. Beginnen Sie damit, ausreichend Markdown für die Strukturierung Ihrer Arbeit zu erlernen, einschließlich Überschriften, Listen, Links und Codeblöcken. Markdown ist schnell zu erlernen; ich empfehle dazu den Besuch von www.markdowntutorial.com und das Durcharbeiten der interaktiven Lektionen sowie des Abschnitts *Markdown Basics* auf quarto.org.
- Machen Sie sich vertraut mit Quarto. Als Lektüre dient Telford [2023]: Enough Markdown to Write a Thesis, welcher fast alles abdeckt, was für das akademische Schreiben hilfreich ist. Alternativ finden Sie umfassende Informationen zur Arbeit mit Quarto direkt auf der Webseite quarto.org/docs/guide.

Quarto und R markdown

Quarto ist ein relativ neues Werkzeug und kann als Nachfolger von R Markdown betrachtet werden. Die meisten R Markdown-Dokumente sind mit Quarto kompatibel. Allerdings bietet Quarto einige verbesserte Funktionen gegenüber R Markdown, die die Benutzerfreundlichkeit steigern. Einen detaillierten Überblick über die Unterschiede und Gemeinsamkeiten zwischen den beiden Plattformen finden Sie in [diesem Artikel] (https://quarto.org/docs/faq/rmarkdown.html).

2.4. Erste Schritte mit Quarto

- Öffnen Sie RStudio.
- Wählen Sie "File" -> "New File" -> "Quarto Document" und dann "Create".
- Speichern Sie die neue Datei in einem leeren Ordner und definieren Sie diesen Ordner als Ihr Arbeitsverzeichnis.
- Klicken Sie auf "Render".
- Besuchen Sie die Webseite *Markdown Basics*, fügen Sie etwas Markdown in Ihr Dokument ein und klicken Sie erneut auf "Render".
- Klicken Sie auf den Pfeil neben dem "Render"-Knopf. Hier können Sie andere Dateiformate auswählen und diese generieren. Probieren Sie es aus.
- Konsultieren Sie die Webseite PDF Basics und ergänzen Sie Ihren Header mit den dort gefundenen Informationen.
- Versuchen Sie das Paper von Huber and Rust [2016], das Sie hier finden, in Ihrem Dokument zu zitieren.
  - Klicken Sie dazu auf "Visual",
  - gehen Sie an die Stelle im Text, an der Sie das Paper zitieren möchten, und wählen Sie "Insert" -> "Citation".
  - Suchen Sie im Kontextmenü mithilfe der entsprechenden DOI (https://doi.org/10.1177/1536867X160 nach dem Papier und fügen Sie es ein.
- Um mit dem APA Version 7-Stil zu zitieren, schreiben Sie folgendes in den YAML-Header:

```
csl: "https://www.zotero.org/styles/apa"
```

- Wählen Sie einen anderen Zitierstil von www.zotero.org/styles. Rendern Sie dann das Dokument erneut und beobachten Sie die Unterschiede.

2.5. APA konformes Manuscript erstellen mit Quarto (apaquarto)

Um ein APA konformes Manuscript zu erstellen, empfiehlt es sich, die *Quarto Extension* `apaquarto` zu benutzen. Wie das geht wird hier genau beschrieben. Durch die Verwendung der Vorlage werden alle APA Regeln automatisch berücksichtigt. Da auch APA viel Spielraum lässt und jeder Gutachter Sonderwünsche hat, erlaubt es `apaquarto`, eine Vielzahl von Einstellungen. Beispielsweise kann in der Preamble (YAML header) die Sprache geändert werden oder der allgemeine Stil des Dokumentes verändert werden.

2.6. Vorlage zur Hausarbeit mit Quarto

Ich habe eine Vorlage erstellt, die Sie zur Erstellung ihrer Hausarbeit verwenden können. Sollte bei Ihnen etwas nicht funktionieren oder sie Hinweise zur Verbesserung haben, freue ich mich über eine Nachricht. Um die Vorlage zu verwenden, folgen Sie bitte den anweisungen auf meinem GitHub Account im Repository *temp_apa_de*:

https://github.com/hubchev/temp_apa_de

Teil II.

Anwendungen

3. Daten einlesen und aufbereiten

Dieses Dokument beschreibt exemplarisch die Datenaufbereitung der Datei 'Dataset 71.txt'. Alle Schritte werden mit R durchgeführt. Der Code ist in die Quarto Datein eingebettet. Die Ergebniss sind vollständig replizierbar. Der verwendete Code kann wieder und anderweitig verwendet werden.

> **Tipp**
>
> Die PDF Datei kann hier heruntergeladen werden: https://github.com/hubchev/ewa/raw/main/ss_24/read_in_71/doc_read_in_71.pdf.
> Um die komplette Arbeit zu replizieren und gegebenfalls auf einen anderen Datensatz anzuwenden, kann das Repository "ewa" von meinem GitHub Account heruntergeladen werden. Alle entsprechenden Dateien befinden sich im Verzeichnis "ewa/ss_24/read_in_71". Hier ist der Link zu dem entsprechenden Repository: https://github.com/hubchev/ewa/
> Wie das alles im Detail von statten geht, wurde in der Übung behandelt.

4. Zwei-Wege-ANOVA-Modellen

Dies ist eine Übung, bei der Datenmanagement mit den `dplyr`-Funktionen `pivot_longer`, `rename` und `bind_rows` geübt wird. Außerdem zeige ich, wie eine ANOVA-Analyse mit R durchgeführt und in Quarto veranschaulicht werden kann. Dabei beziehe ich mich auf den Inhalt von Childs et al. [2021, Kapitel 27].

Unser Ziel ist es, zu lernen, wie man mit Zwei-Wege-ANOVA-Modellen in R arbeitet, anhand eines Beispiels aus einem Pflanzenwettbewerbsexperiment. Der Arbeitsablauf ist sehr ähnlich wie bei der Einweg-ANOVA in R. Wir beginnen mit dem Problem und den Daten und arbeiten dann durch die Modellanpassung, die Bewertung der Annahmen, den Signifikanztest und schließlich die Darstellung der Ergebnisse.

4.1. R-Sitzung einrichten

```
rm(list = ls())

if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, rstatix, ggpubr, agricolae, tinytable, rempsyc,
               knitr, kableExtra, ggstatsplot, papaja, janitor, apa)
```

4.2. Daten einlesen

Pflanzen haben einen optimalen Boden-pH-Wert für ihr Wachstum, und dieser variiert zwischen den Arten. Folglich würden wir erwarten, dass wenn wir zwei Pflanzen im Wettbewerb zueinander unter verschiedenen pH-Werten anbauen, der Effekt des Wettbewerbs je nach Boden-pH-Wert unterschiedlich ausfallen könnte. In einer aktuellen Studie wurde das Wachstum des Grases Festuca ovina (Schaf-Schwingel) im Wettbewerb mit der Besenheide Calluna vulgaris (Heidekraut) in Böden mit unterschiedlichen pH-Werten untersucht. Calluna ist gut angepasst, auf sehr sauren Böden wie dem Millstone Grit und den Hochmoorflächen um Sheffield zu wachsen. Festuca wächst auf Böden mit einem viel breiteren pH-Bereich. Wir könnten die Hypothese aufstellen, dass Calluna in sehr sauren Böden ein besserer Konkurrent von Festuca sein wird als in mäßig sauren Böden. Hier sind die Daten: Die Spalten `pH 3.5` und `pH 5.5` enthalten das Gewicht, die Spalte `Condition` enthält die Anwesenheit oder Abwesenheit von Calluna.

Dies ist ein vollständig faktorielles Zwei-Wege-Design. Die Gesamtanzahl der unterschiedlichen Behandlungsgruppen beträgt $2 \times 2 = 4$. Für jede der Behandlungen gab es 5 Messwerte bzw. Pflanzen, was insgesamt $2 \times 2 \times 5 = 20$ Beobachtungen ergibt. Hier sind die vorliegenden Daten:

```
data_present <- data.frame(
  Condition = rep(c("Calluna Present"), each = 5),
  `ph_3_5` = c(2.76, 2.39, 3.54, 3.71, 2.49),
  `ph_5_5` = c(3.21, 4.10, 3.04, 4.13, 5.21),
  check.names = FALSE
)
data_present
```

```
        Condition ph_3_5 ph_5_5
1 Calluna Present   2.76   3.21
2 Calluna Present   2.39   4.10
3 Calluna Present   3.54   3.04
4 Calluna Present   3.71   4.13
5 Calluna Present   2.49   5.21
```

```
data_absent <- data.frame(
  Condition = rep(c("Calluna Absent"), each = 5),
  `ph_3_5`= c(4.10, 2.72, 2.28, 4.43, 3.31),
  `ph_5_5` = c(5.92, 7.31, 6.10, 5.25, 7.45),
  check.names = FALSE
)
data_absent
```

```
       Condition ph_3_5 ph_5_5
1 Calluna Absent   4.10   5.92
2 Calluna Absent   2.72   7.31
3 Calluna Absent   2.28   6.10
4 Calluna Absent   4.43   5.25
5 Calluna Absent   3.31   7.45
```

Um diese zwei Datensätze zu kombinieren, verwende ich die Funktion `bind_rows` (siehe R Dokumentation):

```
data <- bind_rows(data_present, data_absent)
data
```

```
         Condition ph_3_5 ph_5_5
1  Calluna Present   2.76   3.21
2  Calluna Present   2.39   4.10
3  Calluna Present   3.54   3.04
4  Calluna Present   3.71   4.13
5  Calluna Present   2.49   5.21
6   Calluna Absent   4.10   5.92
7   Calluna Absent   2.72   7.31
8   Calluna Absent   2.28   6.10
9   Calluna Absent   4.43   5.25
10  Calluna Absent   3.31   7.45
```

Um diesen Datensatz nun im sogenannten *Long-Format* darzustellen, verwende ich die Funktion `pivot_longer`. Dieses Format hat bei der Verwendung einiger Befehle vorteile. Wie zwischen dem *Long-Format* und den *Wide-Format* gewechselt werden kann, bitte ich Wickham and Grolemund [2023]: 5.3 Lengthening data zu entnehmen.

```
festuca <- data |>
  pivot_longer(cols = starts_with("pH"), names_to = "ph", values_to = "weight") |>
  rename(calluna = Condition) |>
  mutate(across(c(calluna, ph), as.factor))

festuca
```

```
# A tibble: 20 x 3
   calluna         ph     weight
   <fct>           <fct>   <dbl>
 1 Calluna Present ph_3_5   2.76
 2 Calluna Present ph_5_5   3.21
 3 Calluna Present ph_3_5   2.39
 4 Calluna Present ph_5_5   4.1
 5 Calluna Present ph_3_5   3.54
 6 Calluna Present ph_5_5   3.04
 7 Calluna Present ph_3_5   3.71
 8 Calluna Present ph_5_5   4.13
 9 Calluna Present ph_3_5   2.49
10 Calluna Present ph_5_5   5.21
11 Calluna Absent  ph_3_5   4.1
12 Calluna Absent  ph_5_5   5.92
13 Calluna Absent  ph_3_5   2.72
14 Calluna Absent  ph_5_5   7.31
15 Calluna Absent  ph_3_5   2.28
16 Calluna Absent  ph_5_5   6.1
17 Calluna Absent  ph_3_5   4.43
18 Calluna Absent  ph_5_5   5.25
19 Calluna Absent  ph_3_5   3.31
20 Calluna Absent  ph_5_5   7.45
```

4.3. Deskriptive Statistik

Um Aussagen über die Beziehung des pH-Werts mit der Pflanzenart tätigen zu können, sollte zunächst ein deskriptiver Blick auf die Daten getätigt werden. Lassen Sie uns also auf den Mittelwert und die Standardabweichung der vier Gruppen blicken. Dies kann tabellarisch oder grafisch geschehen.

4.3.1. Tabellarisch

Dies geht flexibel mit den Funktionen `group_by` in Kombination mit `summarize`:

```r
summary_stats <- festuca |>
  group_by(calluna, ph) |>
  summarize(
    mean = mean(weight),
    sd = sd(weight)
  ) |>
  ungroup()

summary_stats
```

```
# A tibble: 4 x 4
  calluna         ph      mean    sd
  <fct>           <fct>  <dbl> <dbl>
1 Calluna Absent  ph_3_5  3.37 0.904
2 Calluna Absent  ph_5_5  6.41 0.945
3 Calluna Present ph_3_5  2.98 0.609
4 Calluna Present ph_5_5  3.94 0.869
```

Wenn wir schließlich eine publikationswürdige Tabelle haben wollen, geht das wie folgt:

````
```{r  , echo=FALSE, warning=FALSE, message=FALSE}
#| label: tbl-desc_calluna
#| tbl-cap: Deskriptive Statistiken
#| tbl.align: left

summary_stats |>
  kable()
```
````

Das Ergebnis, ist in Tabelle 4.1 zu sehen.

Tabelle 4.1.: Deskriptive Statistiken

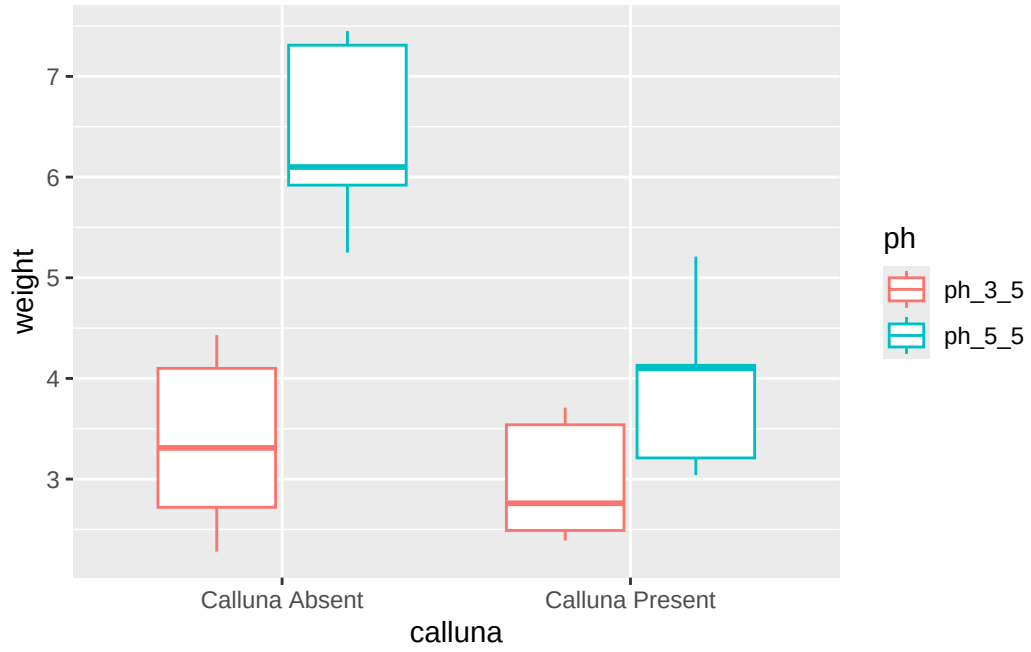
calluna	ph	mean	sd
Calluna Absent	ph_3_5	3.368	0.9042511
Calluna Absent	ph_5_5	6.406	0.9451614
Calluna Present	ph_3_5	2.978	0.6089089
Calluna Present	ph_5_5	3.938	0.8685448

4.3.2. Grafisch

Boxplots bieten einen guten Einblick in die Häufigkeitsverteilung, ohne die Grafik zu überfrachten. Bei wenigen Beobachtungen, wie in unserem Fall, können sie aber problematisch sein da die Datengrundlage (5 Beobachtungen pro Boxplot) nicht ersichtlich ist, siehe Abbildung 4.1.

```r
ggplot(data = festuca, aes(x = calluna, y = weight, colour = ph)) +
  geom_boxplot()
```

Abbildung 4.1.: Boxplots



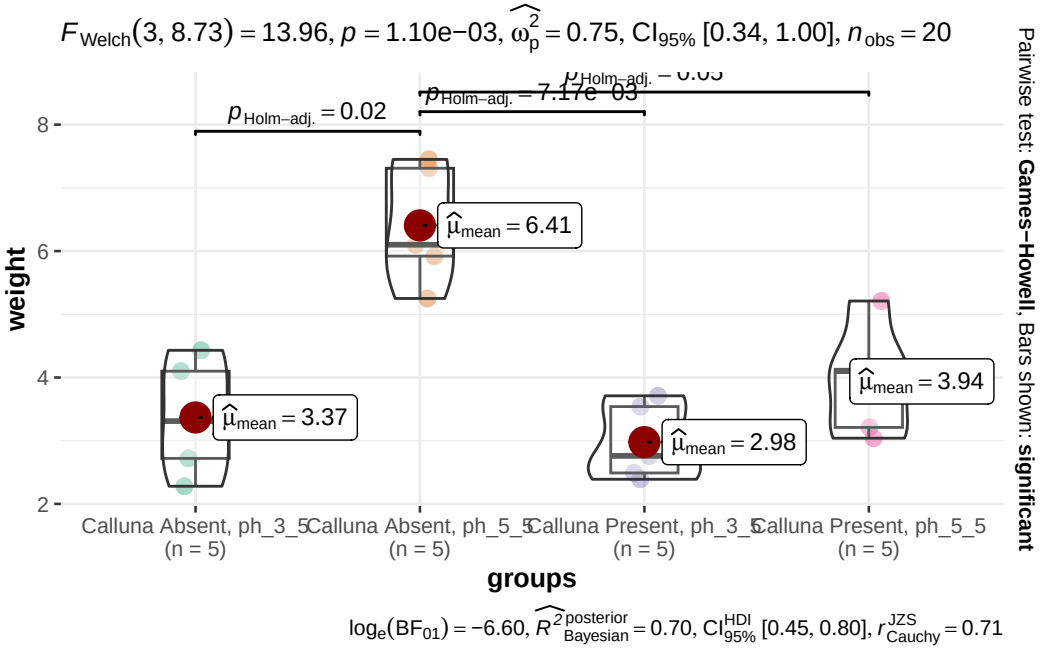
Mit der Funktion `ggbetweenstats` aus dem Paket `ggstatsplot` können die einzelnen Beobachtungen und die statistischen Test zu den Mittelwertvergleichen angezeigt werden, siehe Abbildung 4.2.

```
festuca_group <- festuca |>
  mutate(groups = paste(calluna, ph, sep = ", "))

plt <- ggbetweenstats(
  data = festuca_group,
  x = groups,
  y = weight
)

plt
```

Abbildung 4.2.: Boxplots mit `ggbetweenstats`



4.4. t-Test

4.4.1. One Sample t-test

```
t.test(festuca$weight, mu = 4)
```

```
    One Sample t-test

data:  festuca$weight
t = 0.49085, df = 19, p-value = 0.6292
alternative hypothesis: true mean is not equal to 4
95 percent confidence interval:
 3.436939 4.908061
sample estimates:
mean of x
   4.1725
```

```
t.test(festuca$weight, mu = 1)
```

```
    One Sample t-test

data:  festuca$weight
t = 9.0273, df = 19, p-value = 2.664e-08
alternative hypothesis: true mean is not equal to 1
```

```
95 percent confidence interval:
 3.436939 4.908061
sample estimates:
mean of x
   4.1725
```

4.4.2. Two sided t-test

```
t.test(data$ph_3_5, data$ph_5_5, paired = FALSE)
```

```

    Welch Two Sample t-test

data:  data$ph_3_5 and data$ph_5_5
t = -3.6529, df = 13.013, p-value = 0.002917
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.1811156 -0.8168844
sample estimates:
mean of x mean of y
    3.173     5.172
```

```
t.test(data_absent$ph_3_5, data_absent$ph_5_5, paired = FALSE)
```

```

    Welch Two Sample t-test

data:  data_absent$ph_3_5 and data_absent$ph_5_5
t = -5.1934, df = 7.9844, p-value = 0.0008343
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -4.387422 -1.688578
sample estimates:
mean of x mean of y
    3.368     6.406
```

```
t.test(data_present$ph_3_5, data_present$ph_5_5, paired = FALSE)
```

```

    Welch Two Sample t-test

data:  data_present$ph_3_5 and data_present$ph_5_5
t = -2.0237, df = 7.1669, p-value = 0.08173
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.0764347  0.1564347
sample estimates:
mean of x mean of y
    2.978     3.938
```

```r
t.test(data_absent$ph_3_5, data_present$ph_5_5, paired = FALSE)
```

```
    Welch Two Sample t-test

data:  data_absent$ph_3_5 and data_present$ph_5_5
t = -1.0165, df = 7.9871, p-value = 0.3392
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -1.8633899  0.7233899
sample estimates:
mean of x mean of y
    3.368     3.938
```

```r
t.test(data_present$ph_3_5, data_absent$ph_5_5, paired = FALSE)
```

```
    Welch Two Sample t-test

data:  data_present$ph_3_5 and data_absent$ph_5_5
t = -6.8177, df = 6.8324, p-value = 0.0002776
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -4.622899 -2.233101
sample estimates:
mean of x mean of y
    2.978     6.406
```

```r
t.test(weight ~ ph, data = festuca)
```

```
    Welch Two Sample t-test

data:  weight by ph
t = -3.6529, df = 13.013, p-value = 0.002917
alternative hypothesis: true difference in means between group ph_3_5 and group ph_5_5 is no
95 percent confidence interval:
 -3.1811156 -0.8168844
sample estimates:
mean in group ph_3_5 mean in group ph_5_5
               3.173                5.172
```

```r
t.test(weight ~ calluna, data = festuca)
```

```
    Welch Two Sample t-test

data:  weight by calluna
```

Tabelle 4.2.: ANOVA Ergebnisse

```
       Effect
1 (Intercept) F(1, 16) = 491.08, p < .001, petasq = .97 ***
2          ph F(1, 16) =  28.18, p < .001, petasq = .64 ***
3     calluna F(1, 16) =  14.40, p = .002, petasq = .47 **
4  ph:calluna F(1, 16) =   7.61, p = .014, petasq = .32 *
```

```
t = 2.2371, df = 12.893, p-value = 0.04359
alternative hypothesis: true difference in means between group Calluna Absent and group Cal
95 percent confidence interval:
 0.04785705 2.81014295
sample estimates:
 mean in group Calluna Absent mean in group Calluna Present
                        4.887                         3.458
```

4.5. ANOVA

Verwenden Sie dieses Modell, um die ANOVA zu berechnen: `weight ~ ph + calluna + ph:calluna`

```
festuca_model <- aov(weight ~ ph + calluna + ph:calluna, data = festuca)
anova(festuca_model)
```

```
Analysis of Variance Table

Response: weight
           Df  Sum Sq Mean Sq F value    Pr(>F)
ph          1 19.9800 19.9800 28.1792 7.065e-05 ***
calluna     1 10.2102 10.2102 14.4001   0.00159 **
ph:calluna  1  5.3976  5.3976  7.6126   0.01397 *
Residuals  16 11.3446  0.7090
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Diese Ergebnis können publikationswürdig mit den Funktionen `apa_print` aus dem Paket `papaja` und `kable` dargestellt werden, siehe Tabelle 4.2:

```
```{r, echo=FALSE, eval=TRUE, message=FALSE, warning=FALSE}
#| label: tbl-festuca_model
#| tbl-cap: "ANOVA Ergebnisse"

apa_anova <- apa_print(festuca_model)
knitr::kable( apa_anova$table, booktabs=T)
```
```

4.6. Diagnostics

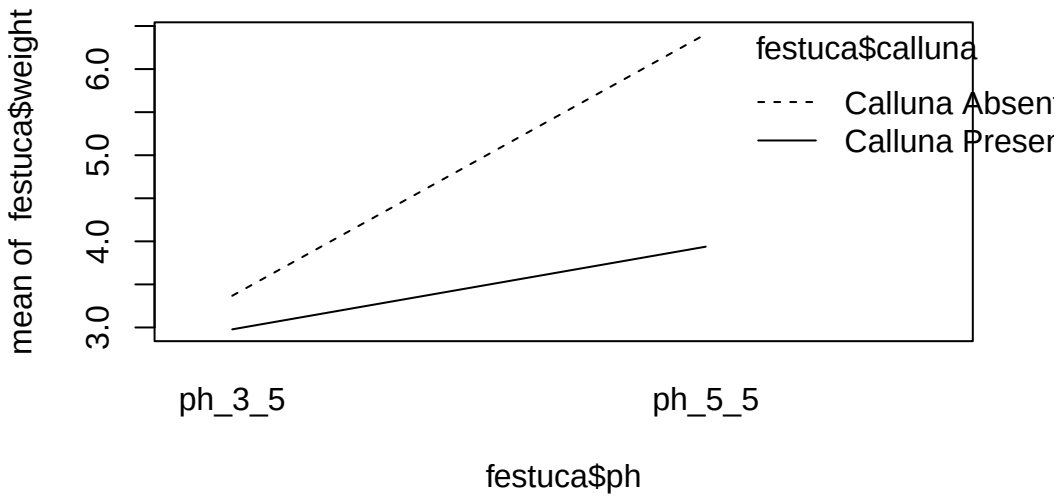
Lesen Sie Childs et al. [2021]: 27.5 Diagnostics. Außerdem ist diese Seite einen Blick wert. Dort finden Sie einige für die ANOVA-Diagnostik hilfreiche R-Funktionen.

4.7. Interaktions Diagramm

Ein Interaktionsdiagramm illustriert, wie zwei oder mehr unabhängige Variablen gemeinsam die abhängige Variable beeinflussen. Es hilft dabei, Wechselwirkungen zwischen Faktoren visuell darzustellen und besser zu verstehen, ob der Effekt einer unabhängigen Variablen von der Ausprägung einer anderen abhängt. Dies ist besonders wichtig, um mögliche Interaktionen identifizieren und interpretieren zu können, die in einer ANOVA-Analyse auftreten.

So ein Diagramm kann mit der Funktion `interaction.plot` erstellt werden:

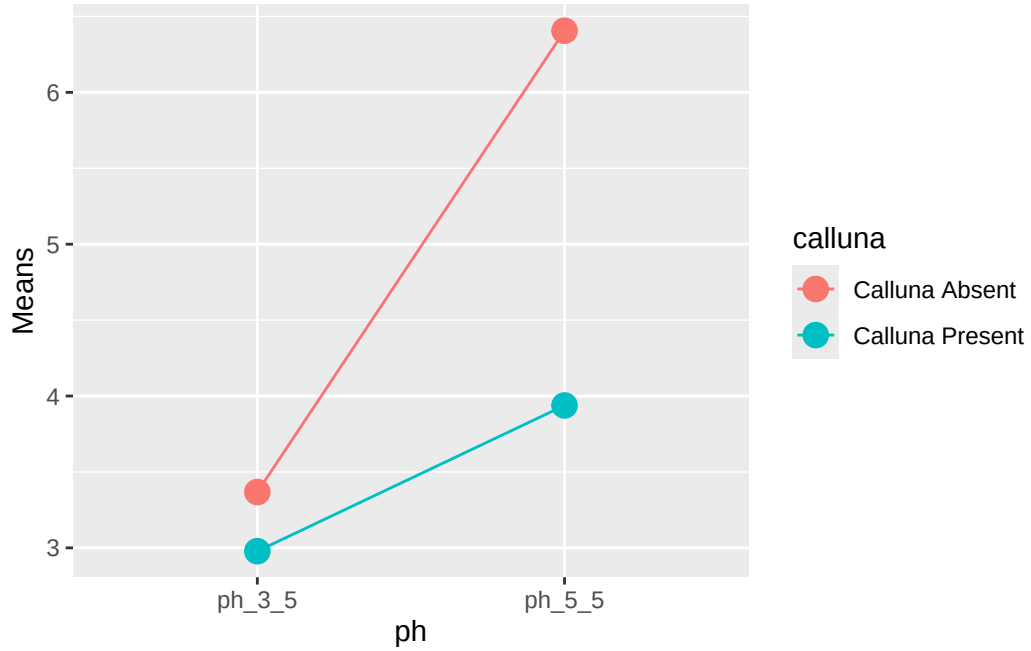
```
interaction.plot(festuca$ph, festuca$calluna, response = festuca$weight)
```



Hier ist eine viel ansprechendere und flexiblere Methode, um Interaktionsdiagramme mithilfe der tidyverse-Funktionen zu erstellen:

```
# step 1. calculate means for each treatment combination
festuca_means <-
  festuca |>
  group_by(calluna, ph) |>  # <- remember to group by *both* factors
  summarise(Means = mean(weight))
```

```
# step 2. plot these as an interaction plot
ggplot(festuca_means,
       aes(x = ph, y = Means, colour = calluna, group = calluna)) +
  geom_point(size = 4) + geom_line()
```



Bitte lesen Sie Childs et al. [2021]: 27.6.1 und berücksichtigen Sie Abbildung 4.3.

4.8. Multiple-Vergleichs-Test

Ein Multiple-Vergleichs-Test, wie der TukeyHSD-Test, wird verwendet, um nach einer ANOVA-Analyse die Unterschiede zwischen den Gruppenpaaren genauer zu untersuchen. Er hilft dabei, festzustellen, welche spezifischen Gruppen sich signifikant voneinander unterscheiden, indem er alle möglichen Paarvergleiche berücksichtigt. Dies ist besonders nützlich, um nach signifikanten Ergebnissen aus der ANOVA detailliertere Erkenntnisse zu gewinnen.

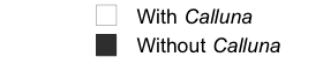
```
TukeyHSD(festuca_model, which = 'ph:calluna')
```

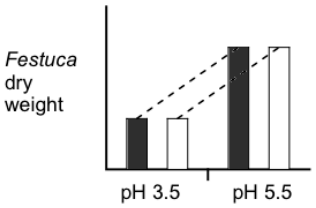
```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = weight ~ ph + calluna + ph:calluna, data = festuca)

$`ph:calluna`
                                                diff        lwr        upr
ph_5_5:Calluna Absent-ph_3_5:Calluna Absent    3.038  1.5143518  4.5616482
ph_3_5:Calluna Present-ph_3_5:Calluna Absent  -0.390 -1.9136482  1.1336482
ph_5_5:Calluna Present-ph_3_5:Calluna Absent   0.570 -0.9536482  2.0936482
ph_3_5:Calluna Present-ph_5_5:Calluna Absent  -3.428 -4.9516482 -1.9043518
ph_5_5:Calluna Present-ph_5_5:Calluna Absent  -2.468 -3.9916482 -0.9443518
ph_5_5:Calluna Present-ph_3_5:Calluna Present  0.960 -0.5636482  2.4836482
                                                  p adj
ph_5_5:Calluna Absent-ph_3_5:Calluna Absent   0.0001731
ph_3_5:Calluna Present-ph_3_5:Calluna Absent  0.8826936
ph_5_5:Calluna Present-ph_3_5:Calluna Absent  0.7117913
ph_3_5:Calluna Present-ph_5_5:Calluna Absent  0.0000443
```

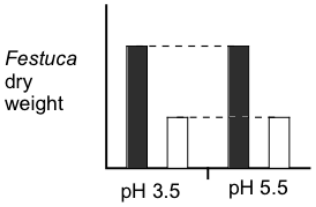
Abbildung 4.3.: Grafische Veranschaulichung des Models





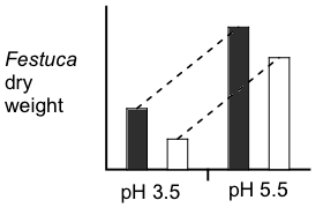
- Significant effect of soil pH
- No significant effect of *Calluna*
- No significant interaction

Festuca grows better at higher pH regardless of whether *Calluna* is present or not.



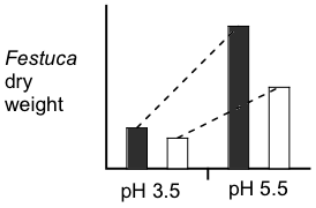
- No significant effect of soil pH
- Significant effect of *Calluna*
- No interaction

Calluna reduces *Festuca* growth, but does so equally in both soil pHs. pH itself has no effect.



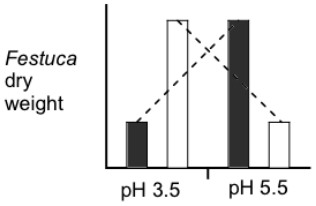
- significant effect of soil pH
- significant effect of *Calluna*
- no significant interaction

Festuca grows better in the absence of *Calluna* and also grows better at higher pH, but the effect of *Calluna* is the same magnitude at both pH levels.
.



- significant effect of soil pH
- significant effect of *Calluna*
- significant interaction

Festuca grows better in the absence of *Calluna* and also grows better at higher pH; however, *Calluna* has a much greater effect on Festuca growth at high pH than at low pH (giving the interaction).
.



- no significant effect of soil pH
- no significant effect of *Calluna*
- significant interaction

The growth of *Festuca* is the same, on average, at both pH levels (compare the means of the two bars at pH 3.5 and the two at pH 5.5), and it is also the same, on average, with and without *Calluna* (compare the means of the two white bars and of the two grey bars). However, the growth is affected by the pH / *Calluna* combination - at higher pH there is a higher growth without *Calluna*, but at lower pH it grows better with *Calluna* present.
.

```
ph_5_5:Calluna Present-ph_5_5:Calluna Absent  0.0014155
ph_5_5:Calluna Present-ph_3_5:Calluna Present 0.3079685
```

```
HSD.test(festuca_model, trt = c("ph", "calluna"), console = TRUE)
```

```
Study: festuca_model ~ c("ph", "calluna")

HSD Test for weight

Mean Square Error:  0.709035

ph:calluna,  means

                       weight       std r        se  Min  Max  Q25  Q50  Q75
ph_3_5:Calluna Absent   3.368 0.9042511 5 0.3765727 2.28 4.43 2.72 3.31 4.10
ph_3_5:Calluna Present  2.978 0.6089089 5 0.3765727 2.39 3.71 2.49 2.76 3.54
ph_5_5:Calluna Absent   6.406 0.9451614 5 0.3765727 5.25 7.45 5.92 6.10 7.31
ph_5_5:Calluna Present  3.938 0.8685448 5 0.3765727 3.04 5.21 3.21 4.10 4.13

Alpha: 0.05 ; DF Error: 16
Critical Value of Studentized Range: 4.046093

Minimun Significant Difference: 1.523648

Treatments with the same letter are not significantly different.

                       weight groups
ph_5_5:Calluna Absent   6.406      a
ph_5_5:Calluna Present  3.938      b
ph_3_5:Calluna Absent   3.368      b
ph_3_5:Calluna Present  2.978      b
```

4.9. Schlussfolgerungen ziehen und Ergebnisse präsentieren

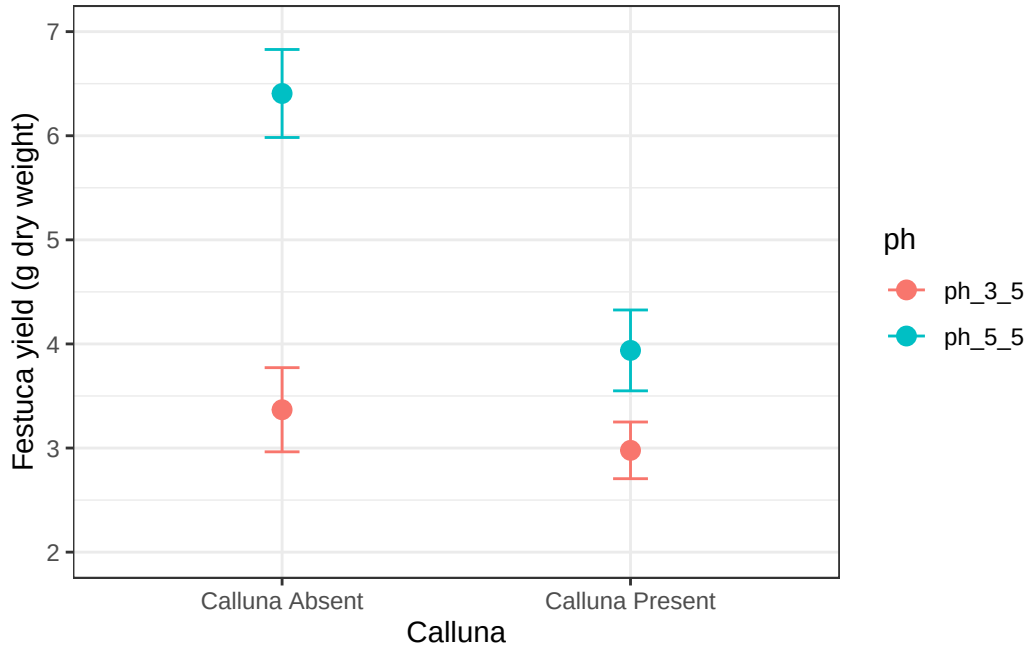
Hier sind einige Code-Beispiele, wie die oben gezeigten Diagramme viel schöner gestaltet werden könnten.

```
# step 1. calculate means for each treatment combination
festuca_stats <-
  festuca |>
  group_by(calluna, ph) %>% # <- remember to group by the two factors
  summarise(means = mean(weight), SEs = sd(weight)/sqrt(n()))
```

```
`summarise()` has grouped output by 'calluna'. You can override using the
`.groups` argument.
```

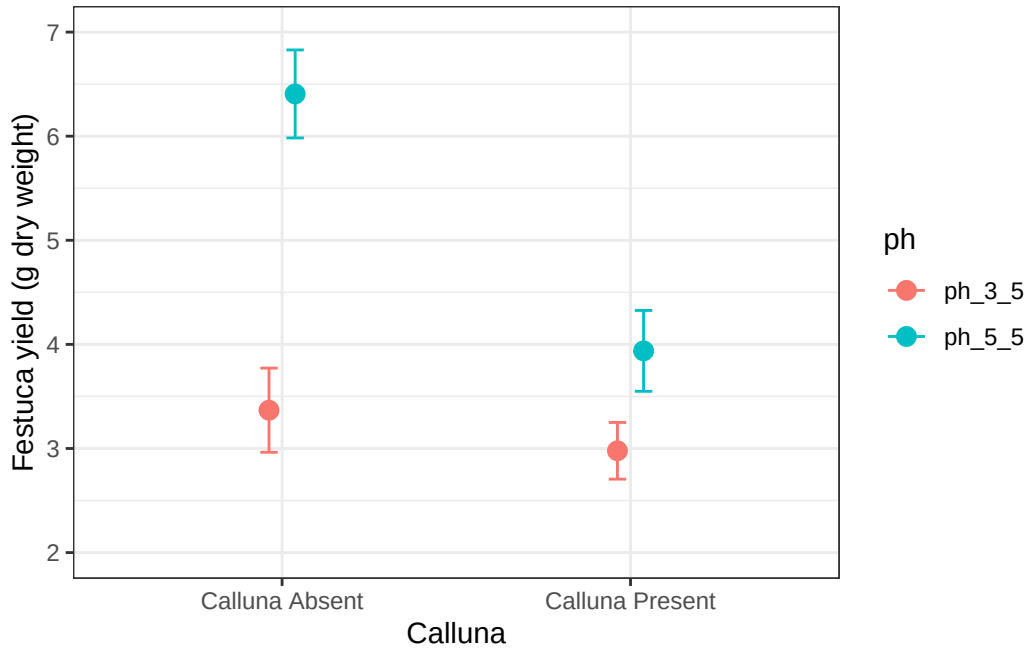
```
# step 1. calculate means for each treatment combination
festuca_stats <-
  festuca |>
  group_by(calluna, ph) %>% # <- remember to group by the two factors
  summarise(means = mean(weight), ses = sd(weight)/sqrt(n()))
```

```
# step 2. plot these as an interaction plot
ggplot(festuca_stats,
       aes(x = calluna, y = means, colour = ph,
           ymin = means - ses, ymax = means + ses)) +
  # this adds the mean
  geom_point(size = 3) +
  # this adds the error bars
  geom_errorbar(width = 0.1) +
  # controlling the appearance
  scale_y_continuous(limits = c(2, 7)) +
  xlab("Calluna") + ylab("Festuca yield (g dry weight)") +
  # use a more professional theme
  theme_bw()
```

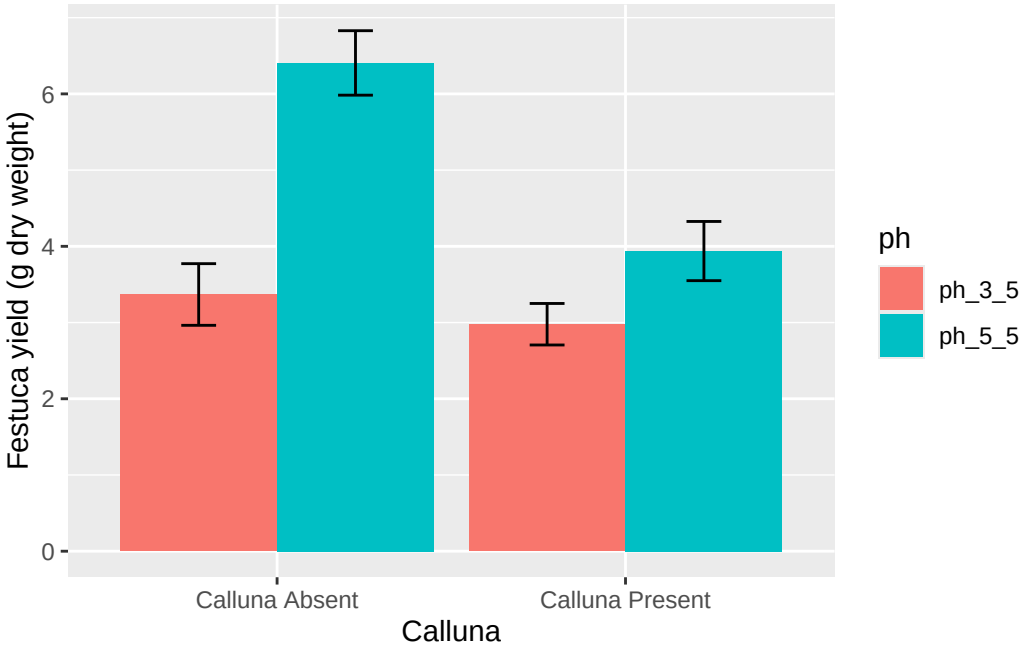



```
# define a position adjustment
pos <- position_dodge(0.15)
# make the plot
ggplot(festuca_stats,
       aes(x = calluna, y = means, colour = ph,
           ymin = means - ses, ymax = means + ses)) +
  # this adds the mean (shift positions with 'position =')
  geom_point(size = 3, position = pos) +
  # this adds the error bars (shift positions with 'position =')
  geom_errorbar(width = 0.1, position = pos) +
  # controlling the appearance
```

```
  scale_y_continuous(limits = c(2, 7)) +
  xlab("Calluna") + ylab("Festuca yield (g dry weight)") +
  # use a more professional theme
  theme_bw()
```




```
ggplot(festuca_stats,
       aes(x = calluna, y = means, fill = ph,
           ymin = means - ses, ymax = means + ses)) +
  # this adds the mean
  geom_col(position = position_dodge()) +
  # this adds the error bars
  geom_errorbar(position = position_dodge(0.9), width=.2) +
  # controlling the appearance
  xlab("Calluna") + ylab("Festuca yield (g dry weight)")
```


Festuca yield (g dry weight)
0
2
4
6
Calluna Absent
Calluna Present
Calluna
ph
ph_3_5
ph_5_5

5. ANOVA Ergebnisse und Quarto

In den folgenden zwei Abschnitten präsentiere ich zwei Dokumente. Beide Dokumente zeigen exemplarisch auf, wie ANOVA Analysen mit R durchgeführt und mit Hilfe von Quarto veranschaulicht werden können.

Um die dargestellten Ergebniss zu replizieren und den Code gegebenfalls auf einen anderen Datensatz anzuwenden, kann das Repository "ewa" von meinem GitHub Account heruntergeladen werden. Alle entsprechenden Dateien befinden sich im entsprechenden Unterverzeichnis "ewa/ss_24". Hier ist der Link zu dem entsprechenden Repository: https://github.com/hubchev/ewa/

> Tipp
>
> Die PDF Datei kann hier heruntergeladen werden: https://github.com/hubchev/ewa/raw/main/ss_24/desc_aov/desc_aov.pdf.
> Die dazu gehörende Quarto Datei sowie alle sonstigen Dateien, sind auf meinem GitHub Account zu finden [Huber, 2024b]: https://github.com/hubchev/ewa/
> Wie das alles im Detail von statten geht, wurde in der Übung behandelt.

6. Regression

Please consider my lecture notes concerning **Regression Analysis** which you find here [Huber, 2024a].

Moreover, I highly recommend reading Wysocki et al. [2022] which is freely available here: https://journals.sagepub.com/doi/10.1177/25152459221095823. They explain how difficult it is to use regression analysis to dentify a causal impact. The main insights of the paper are nicely summarized here: https://osf.io/38mxq.

6.1. Making regression tables using `apa_table`

Here is an example how to use `apa_table` from the `papaja` package to make regression output tables.

```
rm(list = ls())

if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, stargazer, kableExtra, papaja, haven, tinytable)
```

```
# Load the mtcars dataset
data("mtcars")

# Fit a linear regression model
m1 <- lm(mpg ~ wt + hp, data = mtcars)
m2 <- lm(mpg ~ wt , data = mtcars)

# Summary of the model
summary(m1)
```

```
Call:
lm(formula = mpg ~ wt + hp, data = mtcars)

Residuals:
   Min     1Q Median     3Q    Max
-3.941 -1.600 -0.182  1.050  5.854

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 37.22727    1.59879  23.285  < 2e-16 ***
wt          -3.87783    0.63273  -6.129 1.12e-06 ***
hp          -0.03177    0.00903  -3.519  0.00145 **
---
```

Tabelle 6.1.: A full regression table.

term	estimate	conf.int	statistic	df	p.value
Intercept	37.23	[33.96, 40.50]	23.28	29	< .001
Wt	-3.88	[-5.17, -2.58]	-6.13	29	< .001
Hp	-0.03	[-0.05, -0.01]	-3.52	29	.001

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.593 on 29 degrees of freedom
Multiple R-squared:  0.8268,    Adjusted R-squared:  0.8148
F-statistic: 69.21 on 2 and 29 DF,  p-value: 9.109e-12
```

```
apa_lm <- apa_print(m1)
```

````
```{r  , echo=FALSE, warning=FALSE, message=FALSE}
#| label: tbl-reg_class
#| tbl-cap: Deskriptive Statistiken
#| tbl.align: left

tt(apa_lm$table)
```
````

6.2. Data

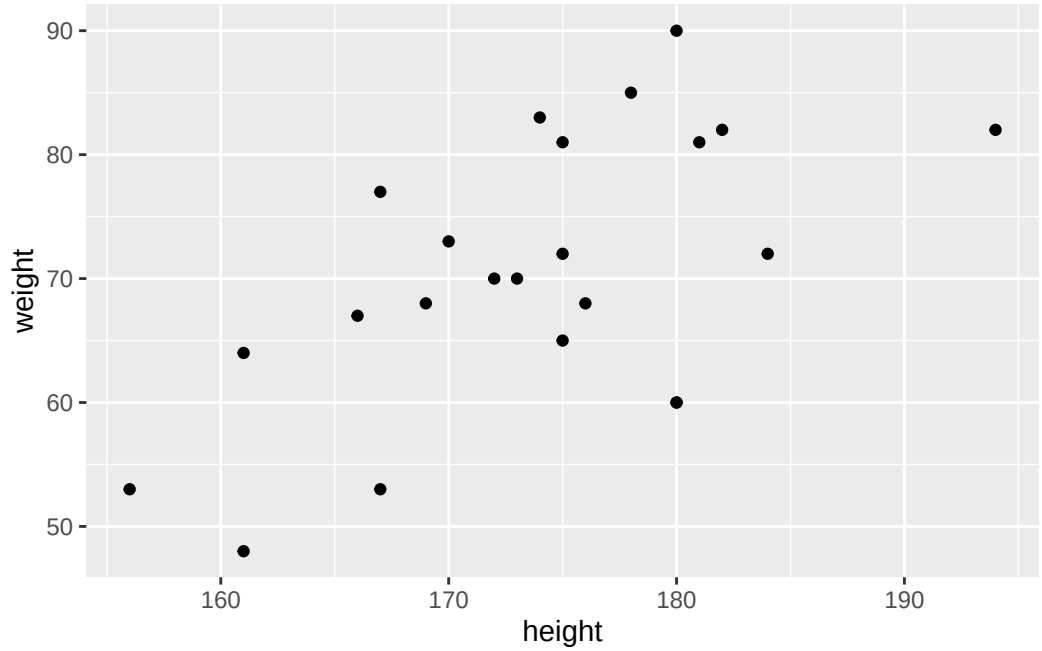
In the statistic course of WS 2020, I asked 23 students about their weight, height, sex, and number of siblings:

```
classdata <- read.csv("https://raw.githubusercontent.com/hubchev/courses/main/dta/classdata
head(classdata)
```

```
  id sex weight height siblings row
1  1   w     53    156        1   g
2  2   w     73    170        1   g
3  3   m     68    169        1   g
4  4   w     67    166        1   g
5  5   w     65    175        1   g
6  6   w     48    161        0   g
```

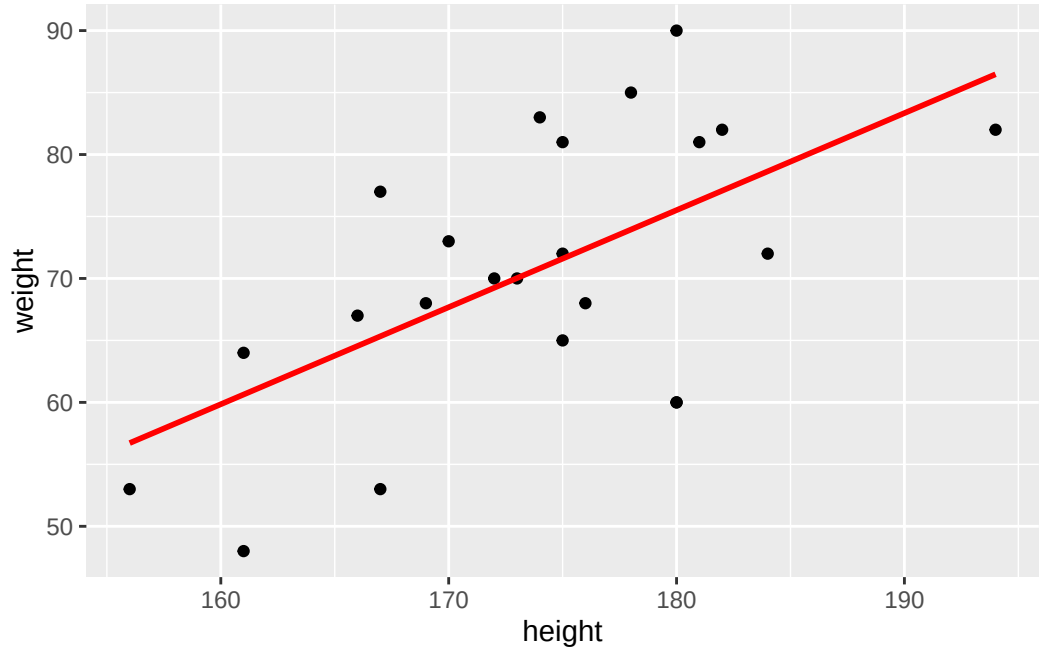
6.3. First look at data

```
ggplot(classdata, aes(x=height, y=weight)) + geom_point()
```



6.4. Include a regression line:

```
ggplot(classdata, aes(x=height, y=weight)) +
  geom_point() +
  stat_smooth(formula=y~x, method="lm", se=FALSE, colour="red", linetype=1)
```



6.5. Regression: Distinguish male/female by including a seperate constant:

```
## baseline regression  model
model  <- lm(weight ~ height + sex , data = classdata )
show(model)
```

```
Call:
lm(formula = weight ~ height + sex, data = classdata)

Coefficients:
(Intercept)       height         sexw
   -29.5297       0.5923      -5.7894
```

```
interm <- model$coefficients[1]
slope  <- model$coefficients[2]
interw <- model$coefficients[1]+model$coefficients[3]
```

```
summary(model)
```

```
Call:
lm(formula = weight ~ height + sex, data = classdata)

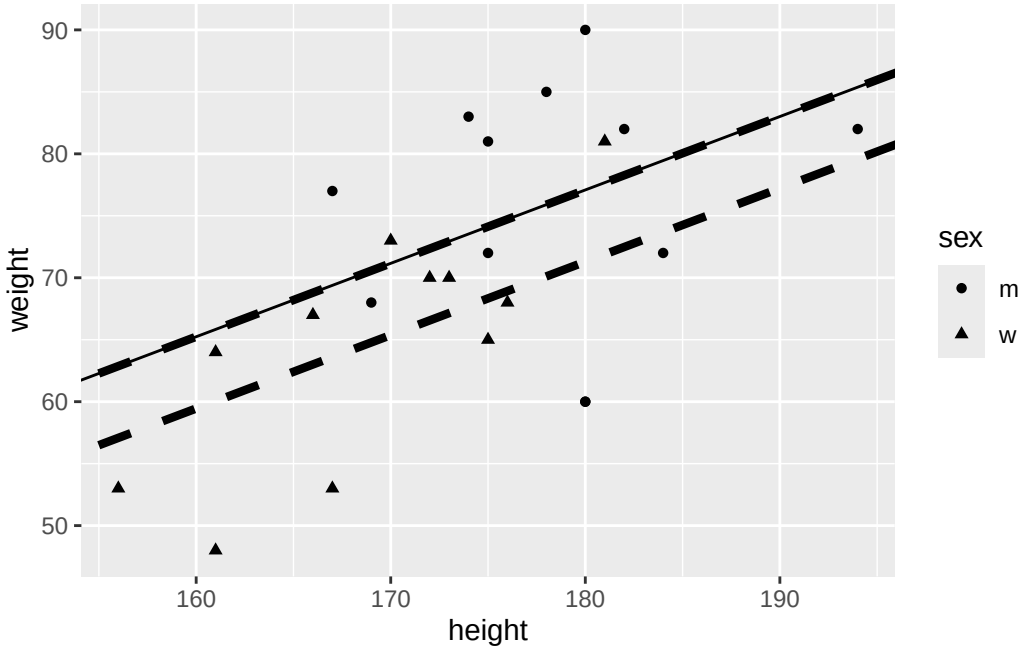
Residuals:
    Min      1Q  Median      3Q     Max
-17.086  -3.730   2.850   7.245  12.914

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -29.5297    47.6606  -0.620   0.5425
height        0.5923     0.2671   2.217   0.0383 *
sexw         -5.7894     4.4773  -1.293   0.2107
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.942 on 20 degrees of freedom
Multiple R-squared:  0.4124,    Adjusted R-squared:  0.3537
F-statistic: 7.019 on 2 and 20 DF,  p-value: 0.004904
```

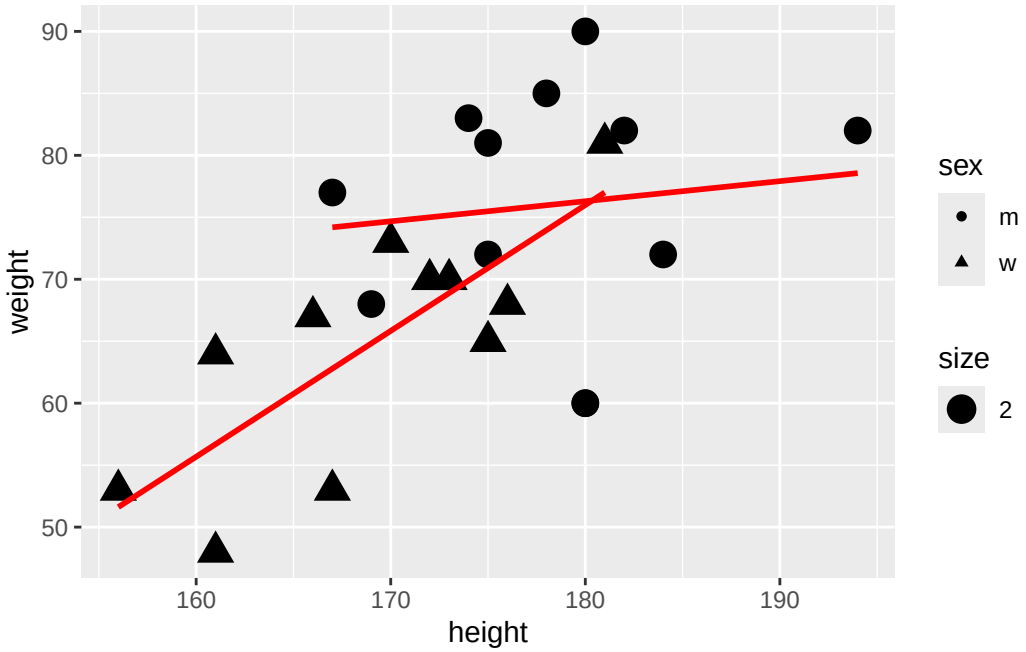
```
ggplot(classdata, aes(x=height, y=weight, shape = sex)) +
  geom_point() +
  geom_abline(slope = slope, intercept = interw, linetype = 2, size=1.5)+
  geom_abline(slope = slope, intercept = interm, linetype = 2, size=1.5) +
  geom_abline(slope = coef(model)[[2]], intercept = coef(model)[[1]])
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```



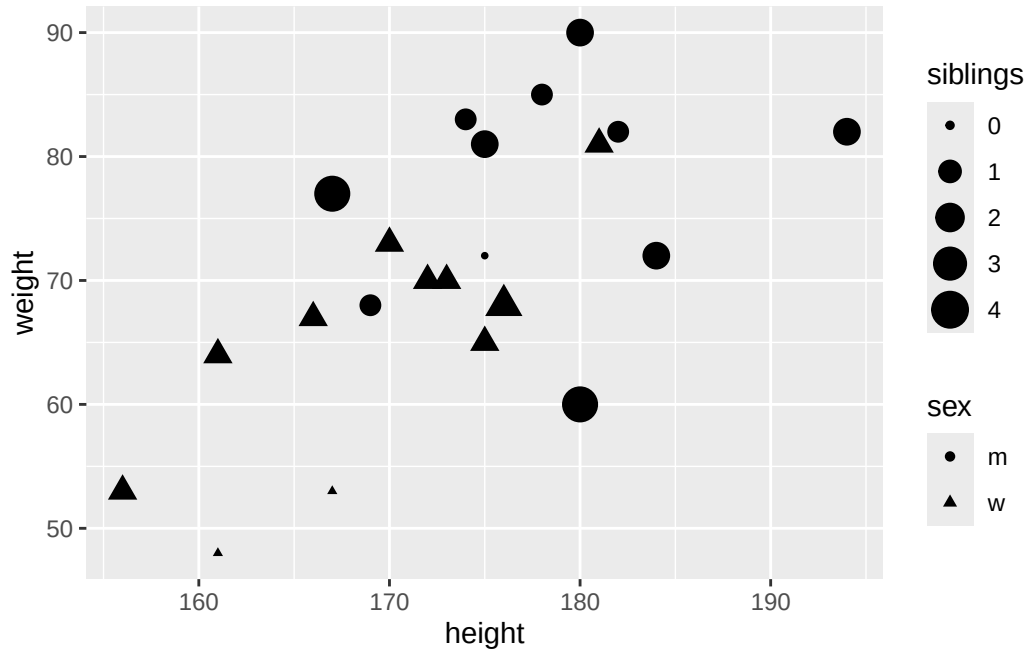
That does not look good. Maybe we should introduce also different slopes for male and female.

```
ggplot(classdata, aes(x=height, y=weight, shape = sex)) +
  geom_point( aes(size = 2)) +
  stat_smooth(formula = y ~ x,  method = "lm",
              se = FALSE, colour = "red", linetype = 1)
```

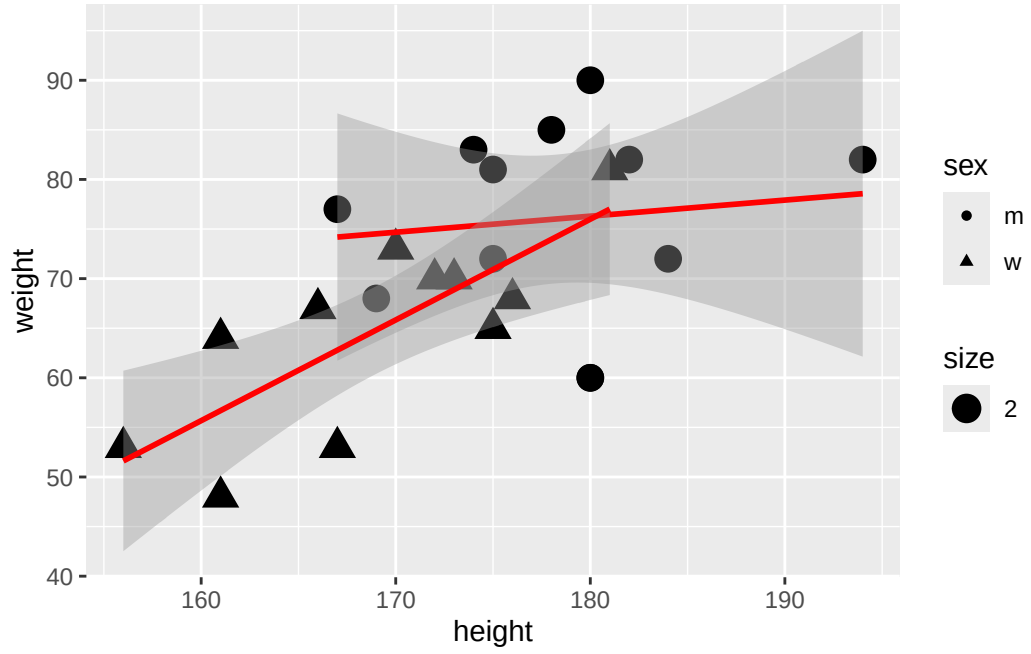


6.6. Can we use other available variables such as siblings?

```
ggplot(classdata, aes(x=height, y=weight, shape = sex)) +
  geom_point( aes(size = siblings))
```




```
ggplot(classdata, aes(x=height, y=weight, shape = sex)) +
  geom_point( aes(size = 2)) +
  stat_smooth(formula = y ~ x,
              method = "lm",
              se = T,
              colour = "red",
              linetype = 1)
```



6.7. Let us look at regression output:

```
m1 <- lm(weight ~ height , data = classdata )
m2 <- lm(weight ~ height + sex , data = classdata )
m3 <- lm(weight ~ height + sex + height * sex , data = classdata )
m4 <- lm(weight ~ height + sex + height * sex + siblings , data = classdata )
m5 <- lm(weight ~ height + sex + height * sex , data = subset(classdata, siblings < 4 ))
```

6.8. Interpretation of the results

- We can make predictions about the impact of height on male and female
- As both, the intercept and the slope differs for male and female we should interpret the regressions seperately:
- One centimeter more for **MEN** is *on average* and *ceteris paribus* related with 0.16 kg more weight.
- One centimeter more for **WOMEN** is *on average* and *ceteris paribus* related with 1.01 kg more weight.

6.9. Regression diagnostics

Linear Regression makes several assumptions about the data, the model assumes that:

- The relationship between the predictor (x) and the dependent variable (y) has linear relationship.
- The residuals are assumed to have a constant variance.
- The residual errors are assumed to be normally distributed.

Tabelle 6.2.: Regression

	Dependent variable:				
			weight		
	Model-1	Model-2	Model-3	Model-4	Model-5
	(1)	(2)	(3)	(4)	(5)
height	0.78***	0.59**	0.16	0.16	0.28
	(0.23)	(0.27)	(0.36)	(0.37)	(0.39)
sexw		−5.79	−153.96*	−161.92*	−134.51
		(4.48)	(88.96)	(91.68)	(90.65)
siblings				−1.16	
				(2.05)	
height:sexw			0.85	0.89	0.74
			(0.51)	(0.53)	(0.52)
Constant	−65.44	−29.53	47.14	50.27	27.69
	(39.35)	(47.66)	(64.81)	(66.23)	(70.36)
Observations	23	23	23	23	21
R^2	0.36	0.41	0.49	0.50	0.57
Adjusted R^2	0.33	0.35	0.41	0.38	0.50
Residual Std. Error	9.08	8.94	8.57	8.73	8.04
F Statistic	11.98***	7.02***	6.02***	4.44**	7.59***
Note:	*p<0.1; **p<0.05; ***p<0.01				
	Here are my notes.				

- Error terms are independent and have zero mean.

More on regression Diagnostics can be found Applied Statistics with R: 13 Model Diagnostics

6.9.1. Check assumptions

When performing regression analysis, it is crucial to validate that the underlying assumptions of the model are met. These assumptions include linearity, independence, homoscedasticity (constant variance of residuals), absence of multicollinearity, and normality of residuals. Diagnosing these assumptions helps ensure the reliability and validity of the model.

In this section, we will explore how to perform regression diagnostics in R using the performance and see packages, which provide comprehensive tools for evaluating model assumptions and performance. Here is a sample code to illustrate these concepts:

```
# Load the required packages using pacman
pacman::p_load(performance, see)

# Check for heteroscedasticity (non-constant variance of residuals)
check_heteroscedasticity(m4)
```

```
OK: Error variance appears to be homoscedastic (p = 0.630).
```

```
# Check for multicollinearity (correlations among predictors)
check_collinearity(m4)
```

```
Model has interaction terms. VIFs might be inflated.
  You may check multicollinearity among predictors of a model without
  interaction terms.

# Check for Multicollinearity

Low Correlation

     Term  VIF        VIF 95% CI Increased SE Tolerance Tolerance 95% CI
   height 2.90 [  1.93,    4.90]         1.70      0.34     [0.20, 0.52]
 siblings 1.30 [  1.07,    2.37]         1.14      0.77     [0.42, 0.94]

High Correlation

       Term    VIF        VIF 95% CI Increased SE Tolerance Tolerance 95% CI
        sex 633.56 [359.01, 1118.64]        25.17  1.58e-03     [0.00, 0.00]
 height:sex 597.51 [338.60, 1054.98]        24.44  1.67e-03     [0.00, 0.00]
```

```
# Check for normality of residuals
check_normality(m4)
```

```
OK: residuals appear as normally distributed (p = 0.086).
```

```
# Check for outliers in the model
check_outliers(m4)
```

```
OK: No outliers detected.
- Based on the following method and threshold: cook (0.816).
- For variable: (Whole model)
```

```
# Evaluate overall model performance
model_performance(m4)
```

```
# Indices of model performance

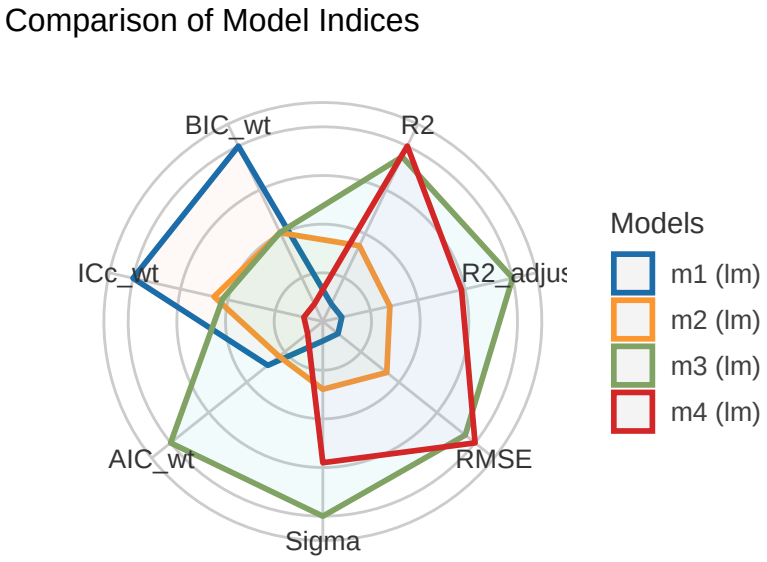
AIC     |    AICc |     BIC |    R2 | R2 (adj.) |  RMSE | Sigma
-----------------------------------------------------------------
171.282 | 176.532 | 178.095 | 0.496 |     0.385 | 7.719 | 8.726
```

```
# Compare performance of multiple models and rank them
compare_performance(m1, m2, m3, m4, rank = TRUE, verbose = FALSE)
```

```
# Comparison of Model Performance Indices

Name | Model |    R2 | R2 (adj.) |  RMSE | Sigma | AIC weights | AICc weights | BIC weights
-------------------------------------------------------------------------------------------
m3   |    lm | 0.487 |     0.407 | 7.788 | 8.568 |       0.381 |        0.240 |       0.241
m4   |    lm | 0.496 |     0.385 | 7.719 | 8.726 |       0.172 |        0.046 |       0.062
m2   |    lm | 0.412 |     0.354 | 8.338 | 8.942 |       0.215 |        0.260 |       0.240
m1   |    lm | 0.363 |     0.333 | 8.680 | 9.084 |       0.232 |        0.454 |       0.457
```

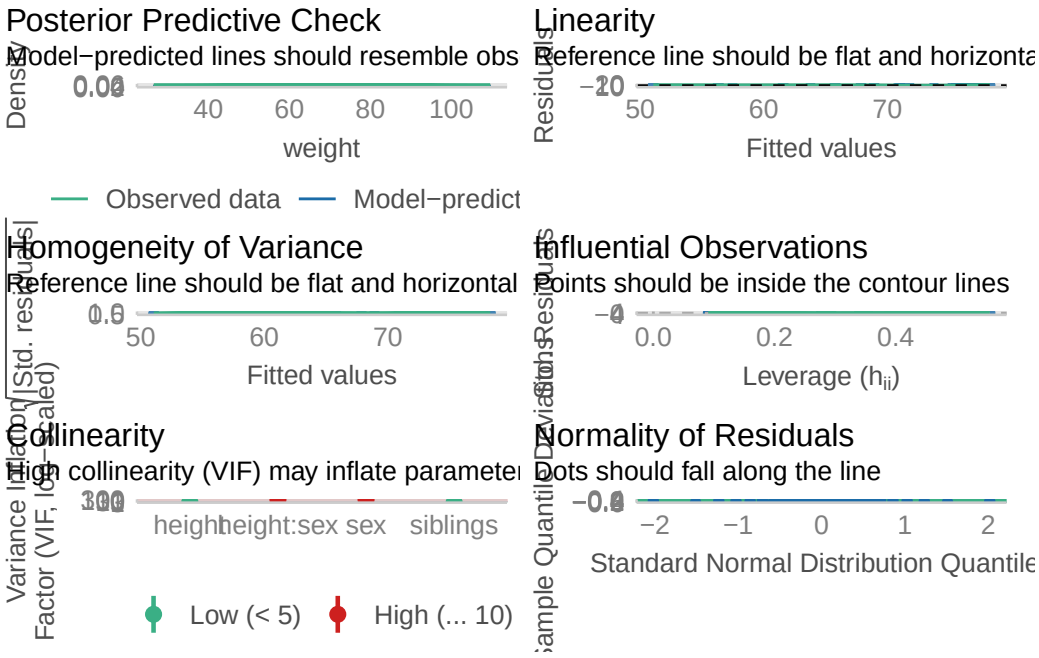
```
# Plot the performance comparison of multiple models
plot(compare_performance(m1, m2, m3, m4, rank = TRUE, verbose = FALSE))
```



```
# Perform statistical tests on the model performance
test_performance(m1, m2, m3, m4)
```

```
Name | Model |    BF | df | df_diff | Chi2 |     p
-----------------------------------------------------
m1   |    lm |       |  3 |         |      |
m2   |    lm | 0.525 |  4 |    1.00 | 1.85 | 0.174
m3   |    lm |  1.00 |  5 |    1.00 | 3.14 | 0.076
m4   |    lm | 0.256 |  6 |    1.00 | 0.41 | 0.524
Models were detected as nested (in terms of fixed parameters) and are compared in sequential
```

```
# Comprehensive check of the model's assumptions
check_model(m4)
```



7. Descriptive Statistics of the NRW80+ Dataset

In this chapter, I illustrate the process of importing NRW80+ data [see Zank et al., 2022] into R. Additionally, I present descriptive statistics and graphical visualizations to gain insights into Likert-scaled surveys. The paper adheres to the APA style, implementing the R template provided by the 'papaja' package [Aust and Barth, 2023].

7.1. Technical Note

In the following, I load (and install) packages that I use later on and I show information about my R session with `sessionInfo()`.

```
# (Install and) load pacman package
if (!require(pacman)) install.packages("pacman")

# load packages that are already installed and install packages that are not
# installed yet and then load them:
pacman::p_load(tinylabels,
               papaja,
               haven,
               labelled,
               janitor,
               skimr,
               rstatix,
               HH,
               likert,
               expss,
               tidyr,
               ggstats,
               psych,
               sjlabelled,
               sjmisc,
               tidyverse,
               MASS,
               dplyr,
               magick,
               tinytable)

# sessionInfo()
```

7.2. Import Data

I host a R script on my GitHub account (see https://raw.githubusercontent.com/hubchev/courses/main/scr/readin_GESIS.R) that explains how to import the NRW80+ data. I have manually saved the data, `gesis.RData`, in a subfolder named `data`.

7.3. How to Use the NRW80+ Data

7.3.1. Load and Subset Data

I load the data and select some variables that are of particular interest to me.

```
getwd()
```

```
[1] "/home/sthu/Dropbox/hsf/courses/ewa"
```

```
load("/home/sthu/Dropbox/hsf/23-ws/ewa/data/gesis.RData")
df <- dfdta |>
  select(starts_with("alter"),
         ALT_agegroup,
         ALT_sex,
         famst1, famst7,
         demtectcorr,
         kogstat,
         final,
         geschlecht)

# Remove the common prefix from all variables
df <- df |>
  mutate_all(~ set_label(., gsub("^Alternserleben: ", "", get_label(.))))
```

For simplification, let us focus on the questions that refer to the "Experience of Ageing" and create a new dataset `df_alterl` that contains only those questions:

```
df_alterl <- df |>
  select(alterl1,
         alterl2,
         alterl3,
         alterl4,
         alterl5,
         alterl6,
         alterl7,
         alterl8,
         alterl9,
         alterl10) |>
  drop_unused_labels()

# to remove unused labels you can use drop_unused_labels():
```

```
df_alterl_un <- df_alterl |>
  drop_unused_labels()

summary(df_alterl)
```

```
    alterl1          alterl2          alterl3          alterl4
 Min.   :-2.000   Min.   :-2.000   Min.   :-2.000   Min.   :-2.000
 1st Qu.: 1.000   1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 2.000
 Median : 3.000   Median : 4.000   Median : 2.000   Median : 3.000
 Mean   : 2.656   Mean   : 3.282   Mean   : 2.349   Mean   : 2.763
 3rd Qu.: 4.000   3rd Qu.: 4.000   3rd Qu.: 3.000   3rd Qu.: 4.000
 Max.   : 5.000   Max.   : 5.000   Max.   : 5.000   Max.   : 5.000
    alterl5          alterl6          alterl7          alterl8
 Min.   :-2.00   Min.   :-2.000   Min.   :-2.000   Min.   :-2.000
 1st Qu.: 2.00   1st Qu.: 2.000   1st Qu.: 2.000   1st Qu.: 1.000
 Median : 3.00   Median : 4.000   Median : 3.000   Median : 3.000
 Mean   : 2.99   Mean   : 3.405   Mean   : 3.237   Mean   : 2.712
 3rd Qu.: 4.00   3rd Qu.: 5.000   3rd Qu.: 4.000   3rd Qu.: 4.000
 Max.   : 5.00   Max.   : 5.000   Max.   : 5.000   Max.   : 5.000
    alterl9          alterl10
 Min.   :-2.000   Min.   :-2.000
 1st Qu.: 2.000   1st Qu.: 1.000
 Median : 3.000   Median : 2.000
 Mean   : 2.969   Mean   : 2.305
 3rd Qu.: 4.000   3rd Qu.: 3.000
 Max.   : 5.000   Max.   : 5.000
```

7.3.2. Get an Overview by Counting

7.3.2.1. `table()` of R base

With the `table()` function, you can count how many observations of each unique value a variable contains:

```
table(df_alterl$alterl1)
```

```
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        80          6        390        266        451        511        159
```

To do that for each variable of a dataset is easy using `~`, the pipe operator, and `map()` of the package `purrr` [Wickham and Henry, 2023]:

```
df_alterl |>
  map(~ table(.))
```

```
$alterl1
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        80          6        390        266        451        511        159

$alterl2
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        36          4        196        245        379        648        355

$alterl3
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        20          3        500        577        403        244        116

$alterl4
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
       122          8        222        260        527        543        181

$alterl5
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
       101          4        199        211        452        680        216

$alterl6
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        19          3        149        324        358        537        473

$alterl7
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        20          2        145        362        471        525        338

$alterl8
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        20          3        516        350        325        340        309

$alterl9
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        83         10        261        228        425        564        292

$alterl10
.
Weiß nicht Verweigert  Gar nicht  Ein wenig      Mäßig      Stark Sehr stark
        44          7        537        433        486        251        105
```

Using `proportions()` returns the conditional proportions:

```
df_alterl |>
  map(~ proportions(table(.)))
```

```
$alterl1
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.042941492 0.003220612 0.209339775 0.142780462 0.242082662 0.274288782
 Sehr stark
0.085346216

$alterl2
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.019323671 0.002147075 0.105206656 0.131508320 0.203435319 0.347826087
 Sehr stark
0.190552872

$alterl3
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.010735373 0.001610306 0.268384326 0.309715513 0.216317767 0.130971551
 Sehr stark
0.062265164

$alterl4
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.065485776 0.004294149 0.119162641 0.139559850 0.282877080 0.291465378
 Sehr stark
0.097155126

$alterl5
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.054213634 0.002147075 0.106816962 0.113258186 0.242619431 0.365002684
 Sehr stark
0.115942029

$alterl6
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.010198604 0.001610306 0.079978529 0.173913043 0.192163178 0.288244767
 Sehr stark
0.253891573

$alterl7
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.010735373 0.001073537 0.077831455 0.194310252 0.252818035 0.281803543
 Sehr stark
```

```
0.181427805

$alterl8
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.010735373 0.001610306 0.276972625 0.187869028 0.174449812 0.182501342
 Sehr stark
0.165861514

$alterl9
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.044551798 0.005367687 0.140096618 0.122383253 0.228126677 0.302737520
 Sehr stark
0.156736447

$alterl10
.
 Weiß nicht  Verweigert   Gar nicht   Ein wenig       Mäßig       Stark
0.023617821 0.003757381 0.288244767 0.232420827 0.260869565 0.134728932
 Sehr stark
0.056360709
```

7.3.2.2. `tabyl()` of `janitor`

With `tabyl()` which is part of `janitor` [Firke, 2023], we can get both nicely:

```
df_alterl |>
  tabyl(alterl1)
```

```
 alterl1   n     percent
      -2  80 0.042941492
      -1   6 0.003220612
       1 390 0.209339775
       2 266 0.142780462
       3 451 0.242082662
       4 511 0.274288782
       5 159 0.085346216
```

```
df_alterl |>
  map(~ tabyl(.))
```

```
$alterl1
  .   n     percent
 -2  80 0.042941492
 -1   6 0.003220612
  1 390 0.209339775
  2 266 0.142780462
  3 451 0.242082662
```

```
  4 511 0.274288782
  5 159 0.085346216

$alterl2
  .   n     percent
 -2  36 0.019323671
 -1   4 0.002147075
  1 196 0.105206656
  2 245 0.131508320
  3 379 0.203435319
  4 648 0.347826087
  5 355 0.190552872

$alterl3
  .   n     percent
 -2  20 0.010735373
 -1   3 0.001610306
  1 500 0.268384326
  2 577 0.309715513
  3 403 0.216317767
  4 244 0.130971551
  5 116 0.062265164

$alterl4
  .   n     percent
 -2 122 0.065485776
 -1   8 0.004294149
  1 222 0.119162641
  2 260 0.139559850
  3 527 0.282877080
  4 543 0.291465378
  5 181 0.097155126

$alterl5
  .   n     percent
 -2 101 0.054213634
 -1   4 0.002147075
  1 199 0.106816962
  2 211 0.113258186
  3 452 0.242619431
  4 680 0.365002684
  5 216 0.115942029

$alterl6
  .   n     percent
 -2  19 0.010198604
 -1   3 0.001610306
  1 149 0.079978529
  2 324 0.173913043
  3 358 0.192163178
  4 537 0.288244767
```

```
  5 473 0.253891573

$alterl7
  .   n     percent
 -2  20 0.010735373
 -1   2 0.001073537
  1 145 0.077831455
  2 362 0.194310252
  3 471 0.252818035
  4 525 0.281803543
  5 338 0.181427805

$alterl8
  .   n     percent
 -2  20 0.010735373
 -1   3 0.001610306
  1 516 0.276972625
  2 350 0.187869028
  3 325 0.174449812
  4 340 0.182501342
  5 309 0.165861514

$alterl9
  .   n     percent
 -2  83 0.044551798
 -1  10 0.005367687
  1 261 0.140096618
  2 228 0.122383253
  3 425 0.228126677
  4 564 0.302737520
  5 292 0.156736447

$alterl10
  .   n     percent
 -2  44 0.023617821
 -1   7 0.003757381
  1 537 0.288244767
  2 433 0.232420827
  3 486 0.260869565
  4 251 0.134728932
  5 105 0.056360709
```

7.3.2.3. `frq()` of `sjmisc`

As the variables `df_alterl1` are factors. Thus, we can use the `sjmisc` package, see Lüdecke [2018] and the cheatsheet of `sjmisc` http://strengejacke.de/sjmisc-cheatsheet.pdf. Also worth a reading is `browseVignettes("sjmisc")`.

For example, we can use `frq()` for nice frequency tables:

```
df_alterl |>
  map(~ frq(. , show.na = T))
```

```
$alterl1
Beziehungen und andere Menschen mehr schätzen (x) <numeric>
# total N=1863 valid N=1863 mean=2.66 sd=1.61

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht |  80 |  4.29 |    4.29 |   4.29
    2 |  Ein wenig |   6 |  0.32 |    0.32 |   4.62
    3 |      Mäßig | 390 | 20.93 |   20.93 |  25.55
    4 |      Stark | 266 | 14.28 |   14.28 |  39.83
    5 | Sehr stark | 451 | 24.21 |   24.21 |  64.04
    6 |       <NA> | 511 | 27.43 |   27.43 |  91.47
    7 |       <NA> | 159 |  8.53 |    8.53 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl2
Gesundheit mehr Aufmerksamkeit widmen (x) <numeric>
# total N=1863 valid N=1863 mean=3.28 sd=1.45

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht |  36 |  1.93 |    1.93 |   1.93
    2 |  Ein wenig |   4 |  0.21 |    0.21 |   2.15
    3 |      Mäßig | 196 | 10.52 |   10.52 |  12.67
    4 |      Stark | 245 | 13.15 |   13.15 |  25.82
    5 | Sehr stark | 379 | 20.34 |   20.34 |  46.16
    6 |       <NA> | 648 | 34.78 |   34.78 |  80.94
    7 |       <NA> | 355 | 19.06 |   19.06 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl3
geistige Leistungsfähigkeit nimmt ab (x) <numeric>
# total N=1863 valid N=1863 mean=2.35 sd=1.28

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht |  20 |  1.07 |    1.07 |   1.07
    2 |  Ein wenig |   3 |  0.16 |    0.16 |   1.23
    3 |      Mäßig | 500 | 26.84 |   26.84 |  28.07
    4 |      Stark | 577 | 30.97 |   30.97 |  59.04
    5 | Sehr stark | 403 | 21.63 |   21.63 |  80.68
```

```
    6 |       <NA> | 244 | 13.10 |   13.10 |  93.77
    7 |       <NA> | 116 |  6.23 |    6.23 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl4
mehr Erfahrung, um Dinge und Menschen einzuschätzen (x) <numeric>
# total N=1863 valid N=1863 mean=2.76 sd=1.72

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht | 122 |  6.55 |    6.55 |   6.55
    2 |  Ein wenig |   8 |  0.43 |    0.43 |   6.98
    3 |      Mäßig | 222 | 11.92 |   11.92 |  18.89
    4 |      Stark | 260 | 13.96 |   13.96 |  32.85
    5 | Sehr stark | 527 | 28.29 |   28.29 |  61.14
    6 |       <NA> | 543 | 29.15 |   29.15 |  90.28
    7 |       <NA> | 181 |  9.72 |    9.72 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl5
besseres Gespür, was wichtig ist (x) <numeric>
# total N=1863 valid N=1863 mean=2.99 sd=1.66

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht | 101 |  5.42 |    5.42 |   5.42
    2 |  Ein wenig |   4 |  0.21 |    0.21 |   5.64
    3 |      Mäßig | 199 | 10.68 |   10.68 |  16.32
    4 |      Stark | 211 | 11.33 |   11.33 |  27.64
    5 | Sehr stark | 452 | 24.26 |   24.26 |  51.91
    6 |       <NA> | 680 | 36.50 |   36.50 |  88.41
    7 |       <NA> | 216 | 11.59 |   11.59 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl6
Einschränkung der Aktivitäten (x) <numeric>
# total N=1863 valid N=1863 mean=3.40 sd=1.38

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht |  19 |  1.02 |    1.02 |   1.02
    2 |  Ein wenig |   3 |  0.16 |    0.16 |   1.18
    3 |      Mäßig | 149 |  8.00 |    8.00 |   9.18
    4 |      Stark | 324 | 17.39 |   17.39 |  26.57
    5 | Sehr stark | 358 | 19.22 |   19.22 |  45.79
```

```
    6 |       <NA> | 537 | 28.82 |   28.82 |  74.61
    7 |       <NA> | 473 | 25.39 |   25.39 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl7
weniger Energie (x) <numeric>
# total N=1863 valid N=1863 mean=3.24 sd=1.32

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht |  20 |  1.07 |    1.07 |   1.07
    2 |  Ein wenig |   2 |  0.11 |    0.11 |   1.18
    3 |      Mäßig | 145 |  7.78 |    7.78 |   8.96
    4 |      Stark | 362 | 19.43 |   19.43 |  28.40
    5 | Sehr stark | 471 | 25.28 |   25.28 |  53.68
    6 |       <NA> | 525 | 28.18 |   28.18 |  81.86
    7 |       <NA> | 338 | 18.14 |   18.14 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl8
Abhängigkeit von der Hilfe Anderer (x) <numeric>
# total N=1863 valid N=1863 mean=2.71 sd=1.53

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht |  20 |  1.07 |    1.07 |   1.07
    2 |  Ein wenig |   3 |  0.16 |    0.16 |   1.23
    3 |      Mäßig | 516 | 27.70 |   27.70 |  28.93
    4 |      Stark | 350 | 18.79 |   18.79 |  47.72
    5 | Sehr stark | 325 | 17.44 |   17.44 |  65.16
    6 |       <NA> | 340 | 18.25 |   18.25 |  83.41
    7 |       <NA> | 309 | 16.59 |   16.59 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl9
Freiheit, Tage nach eigenem Willen zu verleben (x) <numeric>
# total N=1863 valid N=1863 mean=2.97 sd=1.68

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht |  83 |  4.46 |    4.46 |   4.46
    2 |  Ein wenig |  10 |  0.54 |    0.54 |   4.99
    3 |      Mäßig | 261 | 14.01 |   14.01 |  19.00
    4 |      Stark | 228 | 12.24 |   12.24 |  31.24
    5 | Sehr stark | 425 | 22.81 |   22.81 |  54.05
```

```
    6 |       <NA> | 564 | 30.27 |   30.27 |  84.33
    7 |       <NA> | 292 | 15.67 |   15.67 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>

$alterl10
Motivation fällt schwerer (x) <numeric>
# total N=1863 valid N=1863 mean=2.31 sd=1.38

Value |      Label |   N | Raw % | Valid % | Cum. %
--------------------------------------------------
   -2 | Weiß nicht |   0 |  0.00 |    0.00 |   0.00
   -1 | Verweigert |   0 |  0.00 |    0.00 |   0.00
    1 |  Gar nicht |  44 |  2.36 |    2.36 |   2.36
    2 |  Ein wenig |   7 |  0.38 |    0.38 |   2.74
    3 |      Mäßig | 537 | 28.82 |   28.82 |  31.56
    4 |      Stark | 433 | 23.24 |   23.24 |  54.80
    5 | Sehr stark | 486 | 26.09 |   26.09 |  80.89
    6 |       <NA> | 251 | 13.47 |   13.47 |  94.36
    7 |       <NA> | 105 |  5.64 |    5.64 | 100.00
 <NA> |       <NA> |   0 |  0.00 |    <NA> |   <NA>
```

7.3.3. First Summary Statistics

7.3.3.1. Using `summary()` and `get_summary_stats()`

First, I am interested in the class of the data and some very basic summary statistics.

```
summary(df)
```

```
    alterl1          alterl2          alterl3          alterl4
 Min.   :-2.000   Min.   :-2.000   Min.   :-2.000   Min.   :-2.000
 1st Qu.: 1.000   1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 2.000
 Median : 3.000   Median : 4.000   Median : 2.000   Median : 3.000
 Mean   : 2.656   Mean   : 3.282   Mean   : 2.349   Mean   : 2.763
 3rd Qu.: 4.000   3rd Qu.: 4.000   3rd Qu.: 3.000   3rd Qu.: 4.000
 Max.   : 5.000   Max.   : 5.000   Max.   : 5.000   Max.   : 5.000

    alterl5          alterl6          alterl7          alterl8
 Min.   :-2.00   Min.   :-2.000   Min.   :-2.000   Min.   :-2.000
 1st Qu.: 2.00   1st Qu.: 2.000   1st Qu.: 2.000   1st Qu.: 1.000
 Median : 3.00   Median : 4.000   Median : 3.000   Median : 3.000
 Mean   : 2.99   Mean   : 3.405   Mean   : 3.237   Mean   : 2.712
 3rd Qu.: 4.00   3rd Qu.: 5.000   3rd Qu.: 4.000   3rd Qu.: 4.000
 Max.   : 5.00   Max.   : 5.000   Max.   : 5.000   Max.   : 5.000

    alterl9          alterl10        alter_int        alter_cont
 Min.   :-2.000   Min.   :-2.000   Min.   : 80.00   Min.   : 80.11
 1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 82.00   1st Qu.: 82.99
 Median : 3.000   Median : 2.000   Median : 86.00   Median : 86.59
 Mean   : 2.969   Mean   : 2.305   Mean   : 86.48   Mean   : 86.98
```

```
 3rd Qu.: 4.000   3rd Qu.: 3.000   3rd Qu.: 90.00   3rd Qu.: 90.56
 Max.   : 5.000   Max.   : 5.000   Max.   :102.00   Max.   :102.92
                                   NA's   :6        NA's   :6
   alterl_m1       alterl_m2         alterp        ALT_agegroup
 Min.   :1.000   Min.   :1.000   Min.   :-4.000   Min.   :1.000
 1st Qu.:2.600   1st Qu.:2.200   1st Qu.:-4.000   1st Qu.:1.000
 Median :3.200   Median :2.800   Median :-4.000   Median :2.000
 Mean   :3.168   Mean   :2.877   Mean   : 2.632   Mean   :1.883
 3rd Qu.:3.800   3rd Qu.:3.600   3rd Qu.:-4.000   3rd Qu.:3.000
 Max.   :5.000   Max.   :5.000   Max.   :99.000   Max.   :3.000
 NA's   :16      NA's   :14
    ALT_sex          famst1           famst7        demtectcorr
 Min.   :1.000   Min.   :-1.000   Min.   :-3.000   Min.   :-11.000
 1st Qu.:1.000   1st Qu.: 1.000   1st Qu.:-3.000   1st Qu.: -1.000
 Median :2.000   Median : 4.000   Median : 0.000   Median :  0.000
 Mean   :1.502   Mean   : 2.765   Mean   :-1.179   Mean   : -1.742
 3rd Qu.:2.000   3rd Qu.: 4.000   3rd Qu.: 0.000   3rd Qu.:  0.000
 Max.   :2.000   Max.   : 5.000   Max.   : 1.000   Max.   :  2.000

    kogstat          final         geschlecht
 Min.   :-4.00   Min.   :81.00   Min.   :1.000
 1st Qu.:-4.00   1st Qu.:81.00   1st Qu.:1.000
 Median :-4.00   Median :81.00   Median :2.000
 Mean   :-3.21   Mean   :81.09   Mean   :1.502
 3rd Qu.:-4.00   3rd Qu.:81.00   3rd Qu.:2.000
 Max.   : 7.00   Max.   :82.00   Max.   :2.000
```

```
sumstat_alter <- df |>
  get_summary_stats(
    alterl1,
    alterl2,
    alterl3,
    alterl4,
    alterl5,
    alterl6,
    alterl7,
    alterl8,
    alterl9,
    alterl10,
    type = "five_number")
```

```
Warning: attributes are not identical across measure variables; they will be
dropped
```

```
sumstat_alter
```

```
# A tibble: 10 x 7
   variable     n   min   max    q1 median    q3
   <fct>    <dbl> <dbl> <dbl> <dbl>  <dbl> <dbl>
```

```
 1 alterl1   1863    -2     5      1      3      4
 2 alterl2   1863    -2     5      2      4      4
 3 alterl3   1863    -2     5      1      2      3
 4 alterl4   1863    -2     5      2      3      4
 5 alterl5   1863    -2     5      2      3      4
 6 alterl6   1863    -2     5      2      4      5
 7 alterl7   1863    -2     5      2      3      4
 8 alterl8   1863    -2     5      1      3      4
 9 alterl9   1863    -2     5      2      3      4
10 alterl10  1863    -2     5      1      2      3
```

7.3.3.2. Using `psych::describe()`

A powerful alternative for descriptive summary statistics is provided by the function `describe()` of the `psych` package [William Revelle, 2023].

```
sumstat_alter_psych <- df |>
  select(starts_with("alterl")) |>
  select(-ends_with("m1"), -ends_with("m2")) |>
  psych::describe() |>
  as_tibble(rownames="Question")  |>
  select(-skew, -kurtosis, -range, -vars)

sumstat_alter_psych
```

```
# A tibble: 10 x 10
   Question      n  mean    sd median trimmed   mad   min   max     se
   <chr>     <dbl> <dbl> <dbl>  <dbl>   <dbl> <dbl> <dbl> <dbl>  <dbl>
 1 alterl1    1863  2.66  1.61      3    2.76  1.48    -2     5 0.0374
 2 alterl2    1863  3.28  1.45      4    3.43  1.48    -2     5 0.0336
 3 alterl3    1863  2.35  1.28      2    2.28  1.48    -2     5 0.0296
 4 alterl4    1863  2.76  1.72      3    2.96  1.48    -2     5 0.0398
 5 alterl5    1863  2.99  1.66      3    3.20  1.48    -2     5 0.0385
 6 alterl6    1863  3.40  1.38      4    3.54  1.48    -2     5 0.0321
 7 alterl7    1863  3.24  1.32      3    3.33  1.48    -2     5 0.0306
 8 alterl8    1863  2.71  1.53      3    2.68  1.48    -2     5 0.0355
 9 alterl9    1863  2.97  1.68      3    3.14  1.48    -2     5 0.0389
10 alterl10   1863  2.31  1.38      2    2.28  1.48    -2     5 0.0321
```

7.3.3.3. Using `summarize()` and the `tidyverse`

As you may be aware, the `tidyverse` package provides powerful and flexible functions such as `filter`, `select`, `group_by`, and `summarize`. Here is an example demonstrating how these functions can be utilized to create descriptive statistic tables:

```
descriptives <- dfdta |>
  # filter(alterl1 > 0) |>
  group_by(geschlecht)  |>
  summarize(
```

Tabelle 7.1.: Summary Statistics: Experience of Ageing.

variable	n	min	max	q1	median	q3
alterl1	1863	-2	5	1	3	4
alterl2	1863	-2	5	2	4	4
alterl3	1863	-2	5	1	2	3
alterl4	1863	-2	5	2	3	4
alterl5	1863	-2	5	2	3	4
alterl6	1863	-2	5	2	4	5
alterl7	1863	-2	5	2	3	4
alterl8	1863	-2	5	1	3	4
alterl9	1863	-2	5	2	3	4
alterl10	1863	-2	5	1	2	3

Note: This table contains all variables of ‘alterl*‘.

```
    Mean = mean(alterl1)
    , Count = n()
    , SD = sd(alterl1)
    , Min = min(alterl1)
    , Max = max(alterl1)
  )

descriptives
```

```
# A tibble: 2 x 6
  geschlecht    Mean Count    SD Min             Max
  <dbl+lbl>    <dbl> <int> <dbl> <dbl+lbl>       <dbl+lbl>
1 1 [Männlich]  2.71   927  1.50 -2 [Weiß nicht] 5 [Sehr stark]
2 2 [Weiblich]  2.60   936  1.72 -2 [Weiß nicht] 5 [Sehr stark]
```

7.3.4. Make Tables using `tt()`

```
```{r, echo=FALSE, eval=TRUE, message=FALSE, warning=FALSE}
#| label: tbl-tabrstatix
#| tbl-cap: "Summary Statistics: Experience of Ageing."

tt(sumstat_alter,  output = "markdown",
   note = "Note: This table contains all variables of `alterl*`.")
```
```

```
```{r, echo=FALSE, eval=TRUE, message=FALSE, warning=FALSE}
#| label: tbl-tabsumstatalterpsych
#| tbl-cap: "Summary Statistics: Experience of Ageing (psych)"
```
```

Tabelle 7.2.: Summary Statistics: Experience of Ageing (psych)

variable	n	min	max	q1	median	q3
alterl1	1863	-2	5	1	3	4
alterl2	1863	-2	5	2	4	4
alterl3	1863	-2	5	1	2	3
alterl4	1863	-2	5	2	3	4
alterl5	1863	-2	5	2	3	4
alterl6	1863	-2	5	2	4	5
alterl7	1863	-2	5	2	3	4
alterl8	1863	-2	5	1	3	4
alterl9	1863	-2	5	2	3	4
alterl10	1863	-2	5	1	2	3

Note: This table contains all variables of ‘alterl*‘.

Tabelle 7.3.: Summary Statistics: Experience of Ageing (psych)

Question	n	mean	sd	median	trimmed	mad	min	max	se
alterl1	1863	2.655931	1.613659	3	2.757210	1.4826	-2	5	0.03738568
alterl2	1863	3.281804	1.449666	4	3.429913	1.4826	-2	5	0.03358626
alterl3	1863	2.348900	1.278429	2	2.277666	1.4826	-2	5	0.02961898
alterl4	1863	2.763285	1.716885	3	2.963783	1.4826	-2	5	0.03977726
alterl5	1863	2.990338	1.661439	3	3.196512	1.4826	-2	5	0.03849266
alterl6	1863	3.404724	1.384050	4	3.537894	1.4826	-2	5	0.03206605
alterl7	1863	3.236715	1.320460	3	3.325956	1.4826	-2	5	0.03059276
alterl8	1863	2.712292	1.534387	3	2.684775	1.4826	-2	5	0.03554909
alterl9	1863	2.969404	1.677112	3	3.142186	1.4826	-2	5	0.03885578
alterl10	1863	2.305421	1.383735	2	2.284373	1.4826	-2	5	0.03205875

```
tt(sumstat_alter,  output = "markdown",
   note = "Note: This table contains all variables of `alterl*`.")
```

```
```{r, echo=FALSE, eval=TRUE, message=FALSE, warning=FALSE}
#| label: tbl-tabsumstatalterpsychbal
#| tbl-cap: "Summary Statistics: Experience of Ageing (psych)"

tt(sumstat_alter_psych, output = "markdown",
  note = "This table contains all variables of `alterl*` and only observations where all que
```
```

Tabelle 7.4.: Experience of Ageing: Valuing Relationships and Other People More (By Gender)

geschlecht	Mean	Count	SD	Min	Max
1	2.713053	927	1.500062	-2	5
2	2.599359	936	1.717715	-2	5

```
```{r, echo=FALSE, eval=TRUE, message=FALSE, warning=FALSE}
#| label: tbl-tabdescriptives
#| tbl-cap: "Experience of Ageing: Valuing Relationships and Other People More (By Gender)"

tt(descriptives, output = "markdown")
```
```

Table Tabelle 7.1 was created with the function `get_summary_stats()` of the `rstatix` package [Kassambara, 2023], Tables Tabelle 7.2 and Tabelle 7.3 were created with the function `describe()` of the `psych` package [William Revelle, 2023], and Table Tabelle 7.4 was created with the function `summarize()` of the `dplyr` package [Wickham et al., 2023].

7.3.5. Use the Likert Scale using `gglikert()`

We have seen that the data contain not only the five different (Likert scaled) answers. Thus, let us remove all values that have, in one or multiple questions, no answer of the Likert scale. The cleaned dataset is named `df_alterl_balance`.

```
df_alterl_balance <- df_alterl %>%
  rowwise() %>%
  mutate(has_negative = ifelse(any(c(across(alterl1:alterl10)) < 0), 1, 0)) |>
  filter(has_negative == 0) |>
  select(starts_with("alter")) |>
  as_tibble()
```

Using the `gglikert()` of the `ggstats` package [Larmarange, 2023] allows us to draw nice graphs. I highly recommend reading the vignette of the package in the R documentation which you get with `vignette("gglikert")`.

Figures Abbildung 7.1 and Abbildung 7.3 shows the proportions of answers using `df_alterl` data and Figures Abbildung 7.2 and Abbildung 7.4 does so using the `df_alterl_balance` data whereby the latter to show the proportions stacked. Do you see any difference and can you explain the differences?

As we are interested in the differences of the two samples, it makes sense to look as the summary statistics for the `df_alter_balance` sample. This is shown in Table Tabelle 7.3.

Abbildung 7.1.: Experience of Ageing: Proportions of Answers (df_alterl)

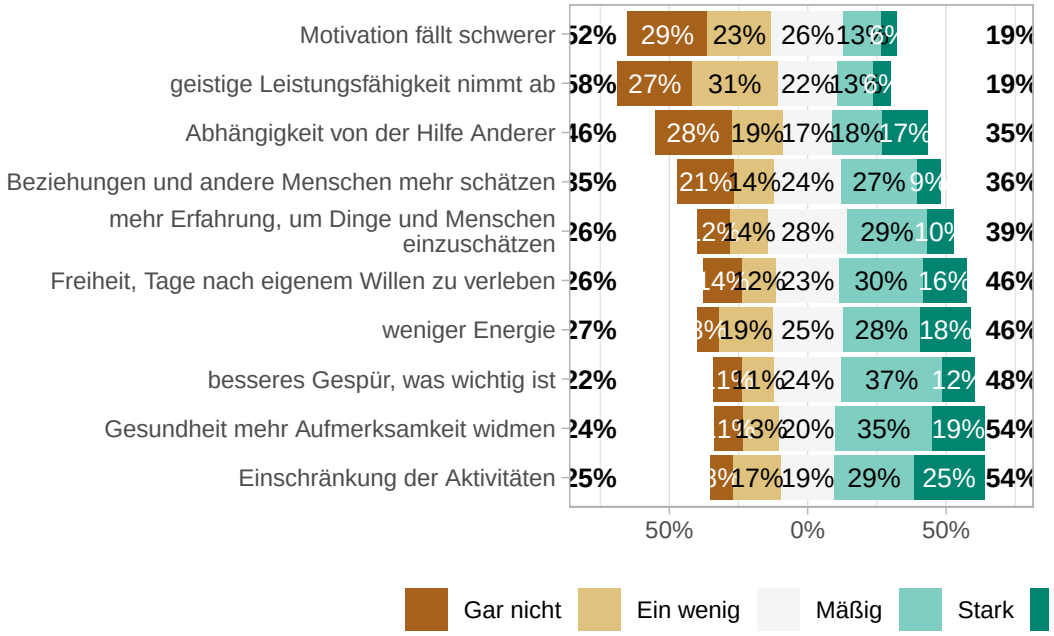


Abbildung 7.2.: Experience of Ageing: Proportions of Answers (df_alterl_balance)

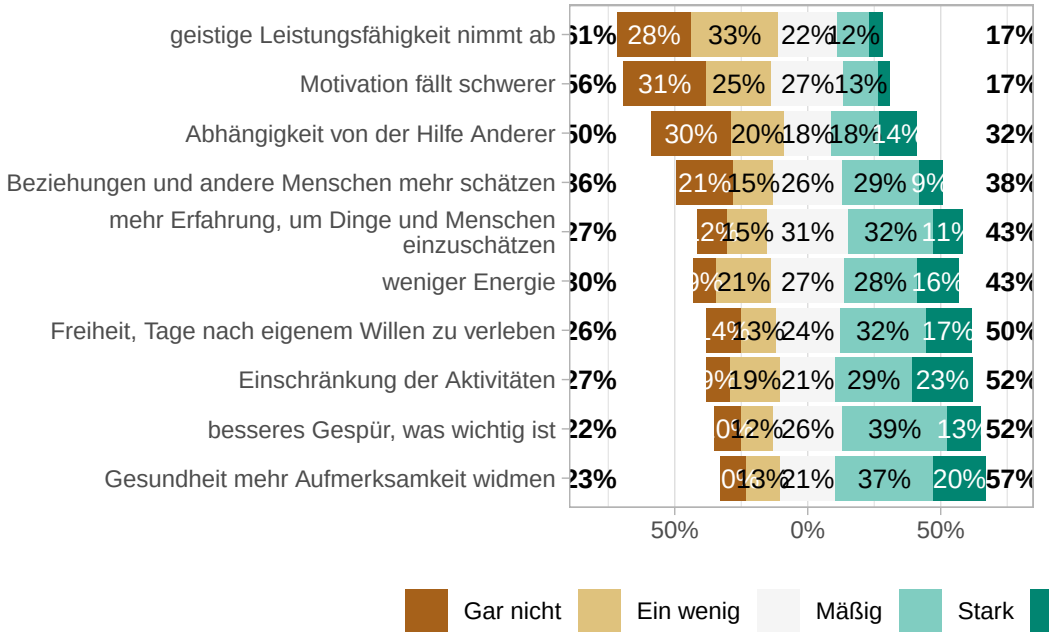


Abbildung 7.3.: Experience of Ageing: Proportions of Answers - Stacked (df_alter)

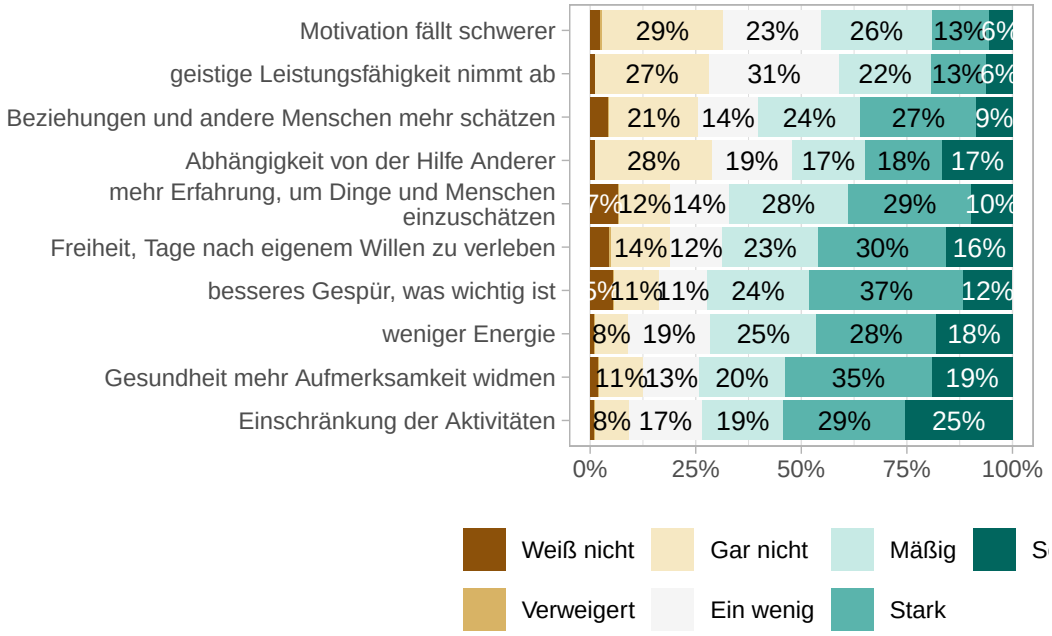
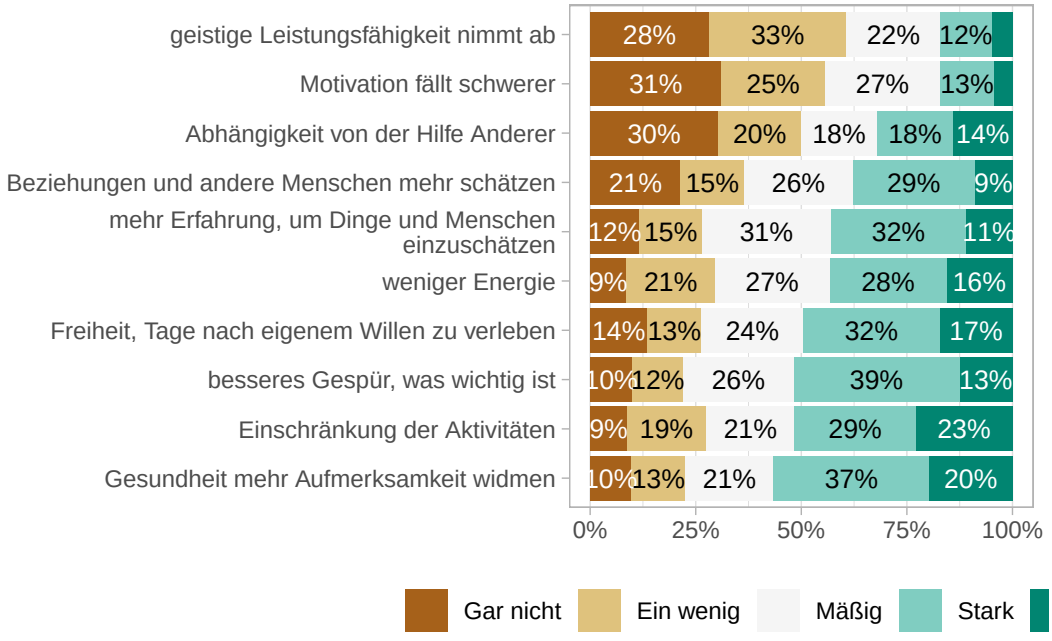


Abbildung 7.4.: Experience of Ageing: Proportions of Answers - Stacked (df_alterl_balance)



7.4. Cross-Referencing in R Markdown

In adherence to the APA style guidelines [Association et al., 2022], it is imperative to reference all figures and tables by their respective numbers within the text. Avoid using generic phrases like "the table above" or "the figure below." Additionally, refrain from hard-coding the numbers for a more dynamic and standardized approach. Xie et al. [2023] explains concisely how to do that with R Markdown, see: https://bookdown.org/yihui/rmarkdown-cookbook/cross-ref.html.

For example, I can refer to Table Tabelle 7.1 with `@tbl-tabrstatix` because I have specified the corresponding label in the R code-chunk, see:

7.5. Exercises

1. With `knitr::purl("desc_NRW80.Rmd")` you can extract the whole R code from the R Markdown file and write it into the R script `desc_NRW80.R`. Try it.
2. The dataset `gesis.RData` comes with two different tibbles: `dfsav` and `dfdta`. Is there a difference between these two when it comes to the statistics that are shown in this paper? To check that, rename the pdf file `desc_NRW80.pdf`, change the code in Section @ref(sec-load) so that you are using the other data (`df <- dfdta |> ...` vs. `df <- dfsav |> ...`), knit the Rmd again, and compare the stats.
3. Check possible differences in the `gglikert` plots when using `df_alterl_un` instead of `df_alterl`.
4. The stats above show that dealing with missing or non-standard answers is a crucial thing. Please read chapter *Missing Values* of Wickham and Grolemund [2023], see: https://r4ds.hadley.nz/missing-values.
5. The labels of the variables `alterl1:alterl10` have "Alternserleben:" at the beginning. This is not necessary and overloads the graphs. Please change the labels for all graphs using the following code in the respective place in the rmd and then knit it again.

```
# Remove the common prefix from all variables
df <- df |>
  mutate_all(~ set_label(., gsub("^Alternserleben: ", "", get_label(.))))
```

Literatur

American Psychological Association et al. *Publication manual of the American psychological association*. Number 1. : American Psychological Association, 2022.

Frederik Aust and Marius Barth. *papaja: Prepare reproducible APA journal articles with R Markdown*, 2023. URL https://github.com/crsh/papaja. R package version 0.1.2.

Paul C. Bauer and Camille Landesvatter. Writing a reproducible paper with rstudio and quarto. Technical report, 2023. URL https://doi.org/10.31219/osf.io/ur4xn.

Dylan Z. Childs, Bethan J. Hindle, and Philip H. Warren. Aps 240: Data analysis and statistics with r. online, 2021. URL https://dzchilds.github.io/stats-for-bio.

Sam Firke. *janitor: Simple Tools for Examining and Cleaning Dirty Data*, 2023. URL https://CRAN.R-project.org/package=janitor. R package version 2.2.0.

Christopher Gandrud. *Reproducible research with R and R studio*. Chapman and Hall/CRC, 3 edition, 2020.

Stephan Huber. Quantitative methods, 2024a. URL https://hubchev.github.io/qm/.

Stephan Huber. Empirisch-wissenschaftlich arbeiten (ewa). GitHub repository, 2024b. URL https://github.com/hubchev/ewa.

Stephan Huber. How to use R for data science, 2024c. URL https://hubchev.github.io/ds/.

Stephan Huber and Christoph Rust. Calculate travel time and distance with openstreetmap data using the open source routing machine (osrm). *The Stata Journal*, 16(2):416–423, 2016.

Alboukadel Kassambara. *rstatix: Pipe-Friendly Framework for Basic Statistical Tests*, 2023. URL https://CRAN.R-project.org/package=rstatix. R package version 0.7.2.

Joseph Larmarange. *ggstats: Extension to 'ggplot2' for Plotting Stats*, 2023. URL https://CRAN.R-project.org/package=ggstats. R package version 0.5.1.

Daniel Lüdecke. sjmisc: Data and variable transformation functions. *Journal of Open Source Software*, 3(26):754, 2018. doi: 10.21105/joss.00754.

Richard J Telford. Enough markdown to write a thesis, 9 2023. URL https://biostats-r.github.io/biostats/quarto/.

Hadley Wickham and Garrett Grolemund. R for data science (2e), 2023. URL https://r4ds.hadley.nz/.

Hadley Wickham and Lionel Henry. *purrr: Functional Programming Tools*, 2023. URL https://CRAN.R-project.org/package=purrr. R package version 1.0.1.

Hadley Wickham, Romain François, Lionel Henry, Kirill Müller, and Davis Vaughan. *dplyr: A Grammar of Data Manipulation*, 2023. URL https://CRAN.R-project.org/package=dplyr. R package version 1.1.2.

William Revelle. *psych: Procedures for Psychological, Psychometric, and Personality Research*. Northwestern University, Evanston, Illinois, 2023. URL https://CRAN.R-project.org/package=psych. R package version 2.3.9.

Anna C Wysocki, Katherine M Lawson, and Mijke Rhemtulla. Statistical control requires causal justification. *Advances in Methods and Practices in Psychological Science*, 5(2), 2022. URL https://doi.org/10.1177/25152459221095823.

Yihui Xie, Christophe Dervieux, and Emily Riederer. R markdown cookbook. online, 2023. URL https://bookdown.org/yihui/rmarkdown-cookbook/.

Susanne Zank, Christiane Woopen, Michael Wagner, Christian Rietz, and Roman Kaspar. Quality of life and well-being of very old people in nrw (representative survey nrw80+) - cross-section wave 1. GESIS, Cologne. ZA7558 Data file Version 2.0.0, 2022.