

Leistungsnachweis:

R you Ready? Reproduzierbare Datenaufbereitung und –analyse ein R

- Anwesenheitspflicht: max. 2x unentschuldigtes Fehlen
- Weitere Leistungen:

Mitarbeit	14 Punkte
Regelmässige Hausübungen	40 Punkte
Abschlussarbeit	46 Punkte
Total	100 Punkte

Notenskala:

100-92 Punkte = **6**

91-84 Punkte = **5.5**

83-76 Punkte = **5**

75-68 Punkte = **4.5**

67-60 Punkte = **4**

<60 Punkte = **ungenügend**

Anmerkung: Man muss bei jeder der drei Einzelleistungen (Mitarbeit, Regelmässige HÜs, Abschlussarbeit) mind. 50% der Punkte erreichen, um das Seminar erfolgreich absolvieren zu können.

Ihr könnt euch jederzeit bei uns melden, um eure Leistungen einzusehen.

Erläuterungen zu den einzelnen Punkten:

1. Mitarbeit → 14 Punkte

Mitarbeit kann unterschiedlich ausfallen, z.B.:

- Wortmeldungen im Plenum
- Aktive Teilnahme an Hands-on Sessions im Seminar
- Beteiligung am Ilias Forum

7 Mitarbeitspunkte werden von der LV-Leitung vergeben, die restlichen 7 Punkte erfolgen am Ende des Semesters über eine Selbsteinschätzung.

2. Regelmässige Hausübungen (Datenanalyseplan, Codebook, R-Übungen mit Peer-Feedback → gesamt 7x) → gesamt 40 Punkte

Details zu den Hausübungen und deren Deadlines werden in den Einheiten erläutert; einen Überblick gibt es hier:

Nr	Bekanntgabe	Inhalt	Abgabe	Peer-Feedback	Punkte
1	EH 2	Datenanalyseplan	Bis EH 4	Siehe folgende HÜ	4
2	EH 4	Peer-Feedback zum Datenanalyseplan	Bis EH 5	-	4
3	EH 5	Codebook	Bis EH 6	Siehe folgende HÜ	4
4	EH 6	Peer-Feedback zum Codebook	Bis EH 7	-	4
5	EH 7	R Übung + Peer Feedback	Bis Freitag vor EH 9	Bis EH 9	8
6	EH 10	R Übung + Peer Feedback	Bis Freitag vor EH 12	Bis EH 12	8
7	EH 12	R Übung + Peer Feedback	Bis Freitag vor EH 14	Bis EH 14	8

Bewertungsrelevant: Auseinandersetzung mit dem Stoff/den aufgetragenen Aufgaben.
Korrektheit zweitrangig!

Für individuelles Feedback oder bei Fragen bitte selbstständig auf Dozierende zukommen.

3. Abschlussprojekt → (46 Punkte)

- Überarbeiteter Datenanalyseplan (hierfür bekommt man «doppelt» Punkte – Datenanalyseplan war auch schon Teil der ersten Hausübung)
- Überarbeitetes Codebook für in R kombinierten Datensatz (dat_full – simulierte Daten; hierfür bekommt man «doppelt» Punkte – Codebook war auch schon Teil der dritten Hausübung)
- Datenfiles der simulierten Daten (7x Einzel-Daten + Datensatz dat_full + Long Datensatz)
- Quarto Analyseskripte: «Processing» und «Analysis» (.qmd + gerenderte Datei als html)

Hauptziel: Reanalyse der Studie von Grinschgl et al. (2021) mit simulierten Daten. Die Datenaufbereitung und -analyse soll im Quarto Skript kommentiert sein. Die statistischen Analysen sollen außerdem im Quarto Skript kurz interpretiert werden.

Schritte der Datenaufbereitung:

- 7 Einzel-Datensätze einlesen Einzel-Datensätze einlesen und zu dat_full mergen
- Data Wrangling, z.B.:
 - Sind Item zu rekodieren?
 - Gibt es Personen mit fehlenden Werten, die ausgeschlossen werden müssen?
 - Gibt es Personen doppelt im Datensatz?

- Gibt es unmögliche Werte in den Daten, z. B. -999? Wie können diese Daten ersetzt werden?
- Gibt es Variablen, die nicht den korrekten Datentyp haben, z. B. numerische Variablen, die als Charakter angezeigt werden? Schaut euch diese Variablen genau an! (Beispiel: *drei* = 3, *stimme voll und ganz zu* = höchster Wert)
- Umbenennen von Variablen, die nicht im *snake_case*-Format sind
- Neue Variablen berechnen (mmq_mean)
- Wide to long Umwandlung
- Variablen zu factor() umwandeln, falls zutreffend (in analysis Skript)

Durchzuführende Analysen für die Reproduzierung von Grinschgl et al., 2021:

- Deskriptive Statistik (*M* & *SD*) berechnen so wie in Table 1 (siehe Grinschgl et al., 2021)
- 2x3 mixed ANOVA für subj. Leistungseinschätzungen mit post-hoc t-Tests, inkl. Effektstärken (η^2 und Cohen's d)
- One-Way ANOVA für jede Offloading Variable (3x), inkl. Effektstärke (η^2)
- One-Way ANOVA für Trial Duration in Pattern Copy Task, inkl. Effektstärke (η^2)
- One-Way ANOVA für MMQ mit post-hoc t-Tests, inkl. Effektstärken (η^2 und Cohen's d)
- One-Way ANOVA für Arbeitsgedächtnisleistung im Feature Switch Detection Task, inkl. Effektstärke (η^2)
- 1 Tabelle oder Abbildung nach Wahl erstellen
- Optional: Testen der Voraussetzungen bei jeder Analyse – wenn diese gemacht wird, soll es auch im Datenanalyseplan beschrieben werden und die Analysen bei Voraussetzungsverletzungen entsprechend angepasst werden.

Bewertungsrelevant:

- Vollständigkeit
 - Datenanalyseplan: Wurde versucht alle notwendigen Fragen zu beantworten?
 - Codebook: Wurde alle Variablen mit den notwendigen Informationen beschrieben?
 - Datenfiles: Wurden alle Datensätze abgegeben?
 - R-Skripte: Wurden alle Datenaufbereitungsschritte und Analysen (so wie im Datenanalyseplan beschrieben) durchgeführt? Sind die Datenaufbereitungsschritte und Analysen auch in einer gerenderten Datei dargestellt?
- Reproduzierbarkeit
 - Datenanalyseplan: Wurden die Fragen in einer nachvollziehbaren Art und Weise beschrieben?
 - Codebook: Wurden alle relevanten Informationen zu einer Variable angeführt?

- Datenfiles: Sind die Dateien und darin enthaltene Variablen nachvollziehbar benannt?
- R-Skripte: Sind die einzelnen Schritte der Datenaufbereitung ausreichend kommentiert, sodass man diese nachvollziehen kann (Skript «Processing»)? Sind die Analysen kommentiert, sodass man nachvollziehen kann welche Schritte durchgeführt wurden (Skript «Analysis»)?
- Weiteres für die Skripte («Processing» und «Analysis»):
 - Wurden die Daten richtig aufbereitet?
 - Kann man die Skripte ohne Fehlermeldungen laufen lassen und rendern ODER sollte es Fehlermeldungen geben, sind diese ausreichend dokumentiert?
 - Wurden die Style Guidelines eingehalten? Hier kommt es vor allem auf einen einheitlichen Style an.
 - Sind die Schritte der Datenaufbereitung/Analyse sinnvoll gruppiert (Aufbereitung in Skript «Processing», Analyse in Skript «Analysis»)?
 - Weisen die Skripte Redundanzen auf?
 - Laufen die Skripte ohne Anpassungen vorzunehmen (d.h. es werden z.B. relative Pfade verwendet)?
 - Sind die Analysen korrekt und wurden diese richtig interpretiert?

Auch hier gilt: Auseinandersetzung mit dem Stoff/den aufgetragenen Aufgaben ist wesentlich. Korrektheit zweitrangig!

Das Abschlussprojekt ist eine individuelle Leistung.

Nichtsdestotrotz können Sie sich mit Ihrem Peerpartner/Ihrer Peerpartnerin austauschen (siehe unten).

Formelle Vorgaben:

- Das Skript ist übersichtlich, enthält keine Redundanzen sowie ungewöhnlichen Formatierungen. Es können die Vorgaben von [APA 7](#) angewendet werden.
- Das Skript enthält sinnvolle Überschriften und eine sinnvolle Gliederung.
- Dokumente können entweder auf Deutsch oder Englisch eingereicht werden. Als Zeitform gilt das Präsens. Die Unterlagen sind in der Wir-Form zu formulieren.
- Kommentare des Codes begründen das Vorgehen sowie die Auswahl der Funktionen und sind somit für aussenstehende Psycholog: innen leicht nachvollziehbar.
- Mögliche Warn- und Fehlermeldungen sind **sichtbar** und **nicht ausgeschaltet**.
- Eine Mögliche Anwendung von ChatGPT oder anderen LLMs wird dokumentiert und über einen Link oder ein getrenntes Dokument eingereicht.
- Mögliche Tabellen und Abbildungen aus [Grinschgl et al. \(2021\)](#) sind sinngemäss und vollständig repliziert. In der Formatierung darf es leicht Abweichungen geben.
- Die Dokumente enthalten lediglich Inhalte mit Bezug zur Endabgabe. Jegliche anderen Inhalte (beispielsweise von der Endabgabe unabhängige Hands on! - Aufgaben) sind zu entfernen.

Deadline Abschlussprojekt: 11.01.26

Abgabe via Ilias als ZIP-Ordner (inkl. Dokumentation der Verwendung von LLMs & nach vorgegebener Ordnerstruktur)

Benennung Zip-Ordner: vorname_nachname_abschlussarbeit.zip

Für individuelles Feedback zum Abschlussprojekt bitte selbstständig auf die Dozierende zukommen.

Bedeutung von «Korrektheit zweitrangig»:

Auseinandersetzung mit dem Stoff bedeutet:

- Die Studierenden zeigen, dass sie sich aktiv mit den Aufgaben auseinandergesetzt haben, indem sie ihre Lösungswege ausführlich (im Quarto Analyseskript) kommentieren und reflektieren.
- Sie sollen erkennbar eigene Denkprozesse darstellen, auch wenn die Analysen nicht immer zu korrekten Ergebnissen führen.
- Fehler in den Analysen sind nicht negativ, solange die Studierenden diese reflektieren und mögliche Lösungsideen zumindest theoretisch skizzieren.
- Sollte man keine eigenen Lösungsideen haben, kann man externe Ressourcen befragen (z.B. Google, LLMs) und dies genauso dokumentieren.

Beispiele:

- Wenn eine Analyse nicht das erwartete Ergebnis liefert, sollten die Studierenden dies kommentieren und mögliche Ursachen oder alternative Vorgehensweisen darlegen.
- Wenn die Interpretation von Ergebnissen unsicher ist, sollten die Studierenden dies darlegen und reflektieren und gegebenenfalls externe Ressourcen heranziehen, um Unterstützung zu bekommen.

Was wird nicht erwartet:

- Die perfekte Korrektheit aller Berechnungen oder Skripte.
- Vollständig fehlerfreie Analysen (solange Fehler dokumentiert und reflektiert werden).

Richtlinien zur Zusammenarbeit mit Peers:

Erlaubte Zusammenarbeit:

- Austausch über allgemeine Fragen zur Datenanalyse, wie z.B. Methodenwahl, oder zur Strukturierung der R-Skripte.
- Gemeinsames Diskutieren von Problemen beim Ausführen von R-Code (z.B. Fehlermeldungen verstehen).
- Peer-Feedback zu Klarheit, Struktur und Vollständigkeit des Datenanalyseplans oder des Codebooks.

Nicht erlaubte Zusammenarbeit:

- Direkter Austausch von vollständigen R-Skripten, Quarto-Dokumenten oder Codebooks.
- Kopieren von Peer-Formulierungen oder Analyseplänen.
- Gemeinsames Erarbeiten der gesamten Analyse oder der Skripte.