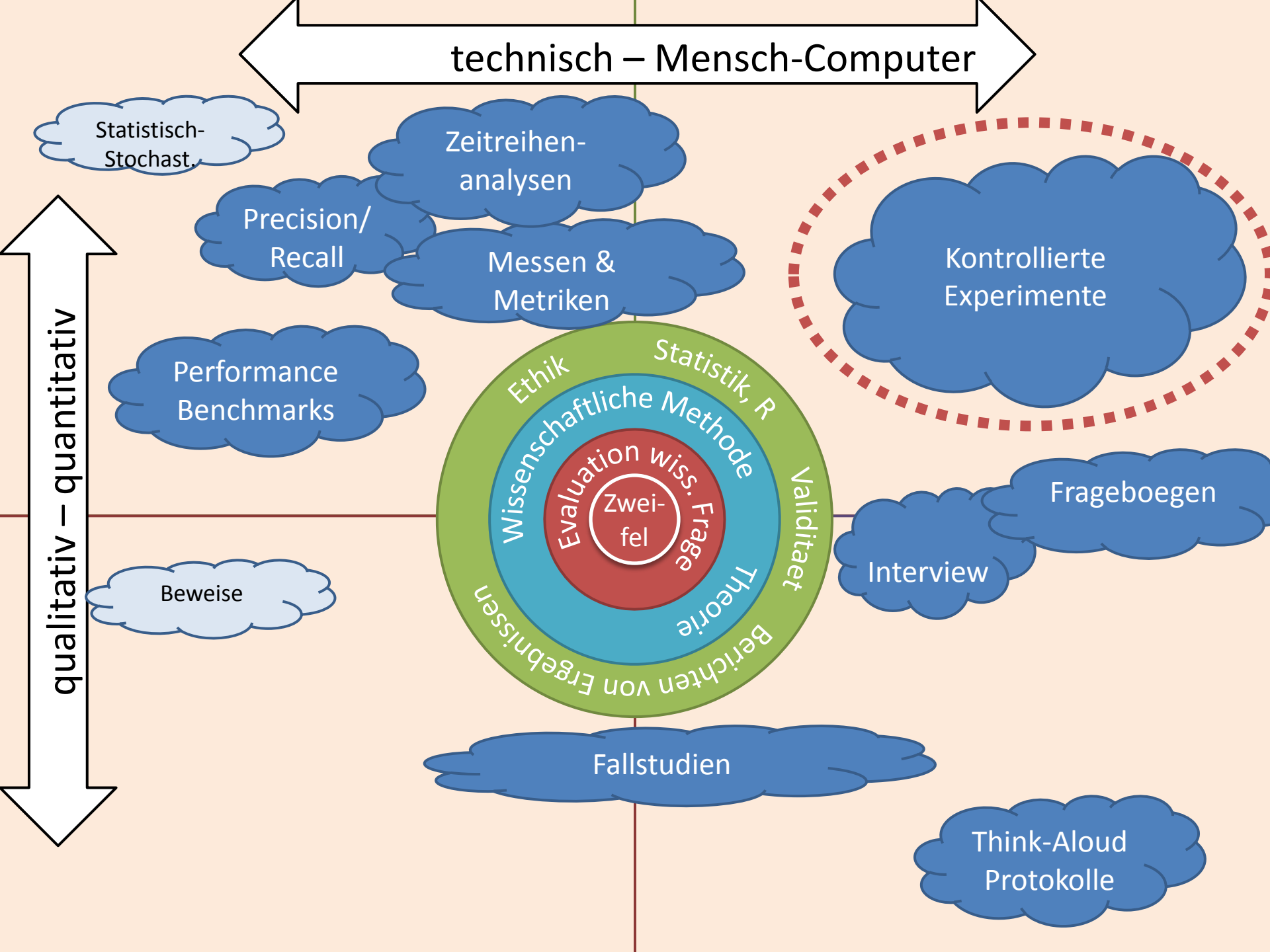
Empirische Methoden für Informatiker
Teil 5: Kontrollierte Experimente

Christian Kästner
Stefan Hanenberg


technisch – Mensch-Computer
qualitativ – quantitativ
Statistisch-Stochast.
Zeitreihen-analysen
Precision/ Recall
Messen & Metriken
Kontrollierte Experimente
Performance Benchmarks
Ethik
Statistik, R
Validitaet
Theorie
Berichten von Ergebnissen
Wissenschaftliche Methode
Evaluation wiss. Frage
Zwei-fel
Frageboegen
Interview
Beweise
Fallstudien
Think-Aloud Protokolle

Agenda

- Einführung Kontrollierte Experimente
- Stoervariablen
- Experimentelle Designs
  - Within-Subject / Between-Subject
  - Latin Square
- Validitaet und Berichten von Ergebnissen

Phasen von Kontrollierten Experimenten

1. Bestimmung der Hypothese(n)
   - Verhältnis von abhängigen und unabhängigen Variablen
2. Experimentaufbau
   - Erstellung von Aufgaben, Fragebögen, etc.
   - Experimentelles Design: Anzahl von Gruppen, Zuordnung von Gruppen, etc.
3. Experimentdurchführung
   - Räumlich / zeitliche Ähnlichkeit, etc.
4. Statistische Analyse
   - Gemessenes Verhältnis von abhängigen und unabhängigen Variablen / Vergleich mit Hypothesen
5. Interpretation / Diskussion

Aufgaben / Diskussion

- Aufgabe: Vergleich Java <-> Perl
- Idee: Zwei Entwicklerteams à 5 Personen schreiben einen HTML 4.0 Browser, Team 1 in Java, Team 2 in Perl

- Diskutieren Sie, warum die so erzielten Ergebnisse problematisch sein könnten

Experimentelles Design
Ziele / Aufgaben / Probleme

10x

- Variation zwischen individuellen Programmierern mit 7 Jahren Berufserfahrung [Sackman et al 1968]
  - Initial Coding Time 20:1
  - Debugging Times 25:1
  - Program size 5:1
  - Program Execution Speed 10:1
- Diverse Studien bestaetigen Faktor 3x bis 10x
- Aehnliche Ergebnisse auch fuer ganze Teams
  - Microsoft Excel (50 staff years, 649KLOC) vs. Lotus 123 (260 staff years, 400KLOC, delivered 2 years late)

Steve McConnell. **What does 10x mean? Measuring variations in programmer productivity**. In Making Software. O'Reilly 2011

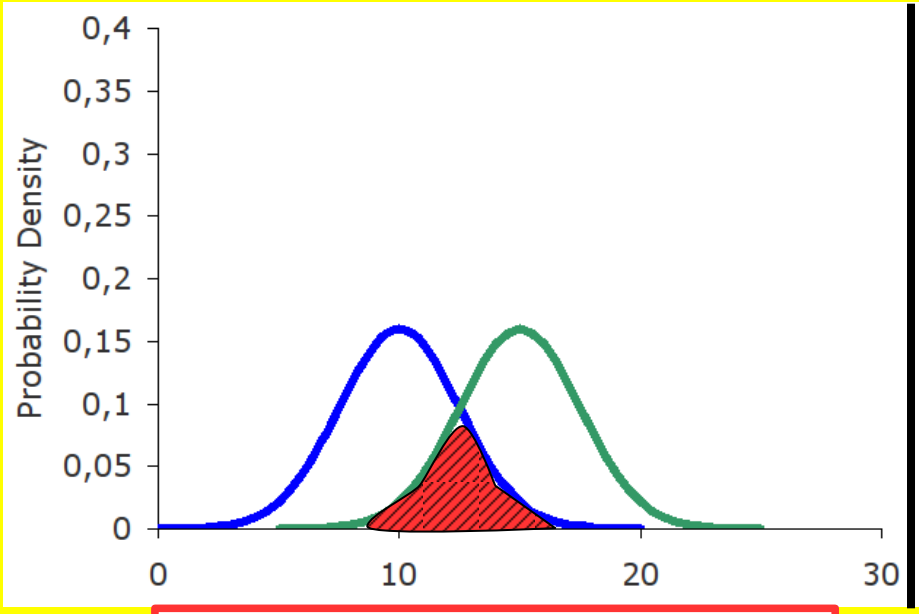
Experimente mit Menschen

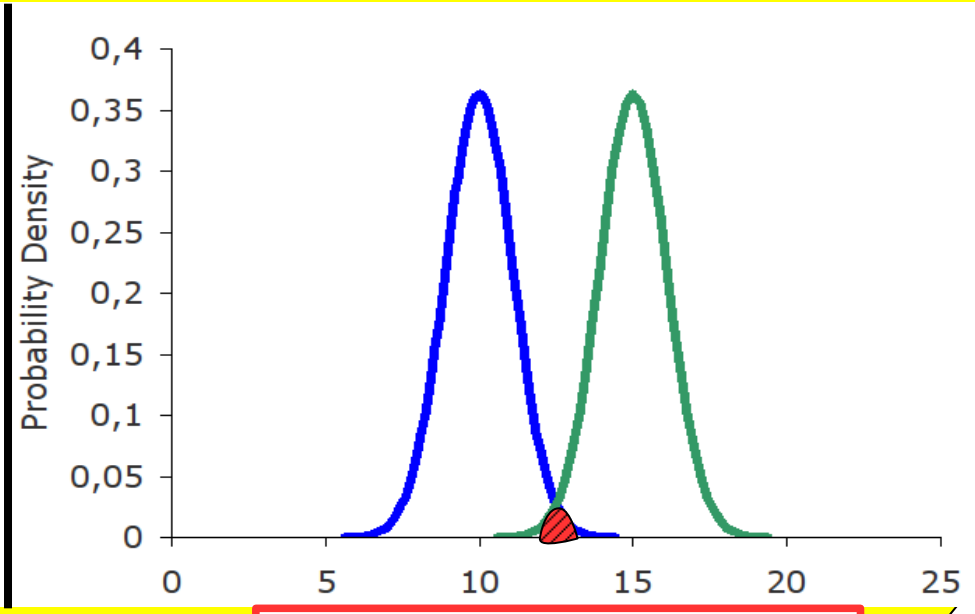
- Hohe Streuung
  - 10x Unterschied zwischen guten und schlechten Programmierern
- Bewusste Steuerung moeglich
  - bewusst Fehler einbauen, besonders anstrengen
- Motivation, Lerneffekte
- Schwer zu rekrutieren, typischerweise wenige Probanden, nicht automatisierbar, teuer

Streuung

- Beispiel: Zwei Gruppen (Messreihen) im Vergleich (Unterschiedshypothese)

Beispiel 1: Gleiche Effektgröße, Unterschiedliche Streuung





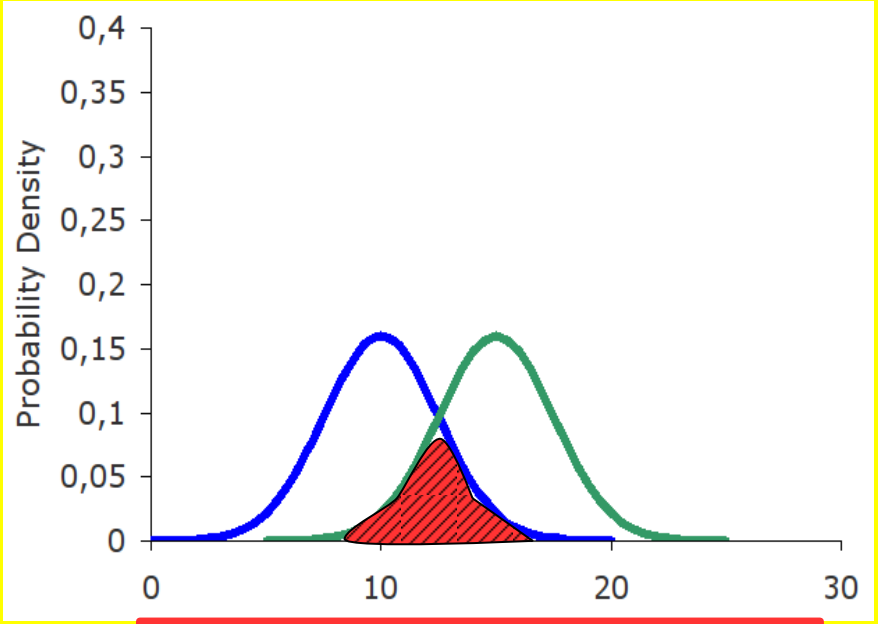
Large overlap
=> no (significant) difference

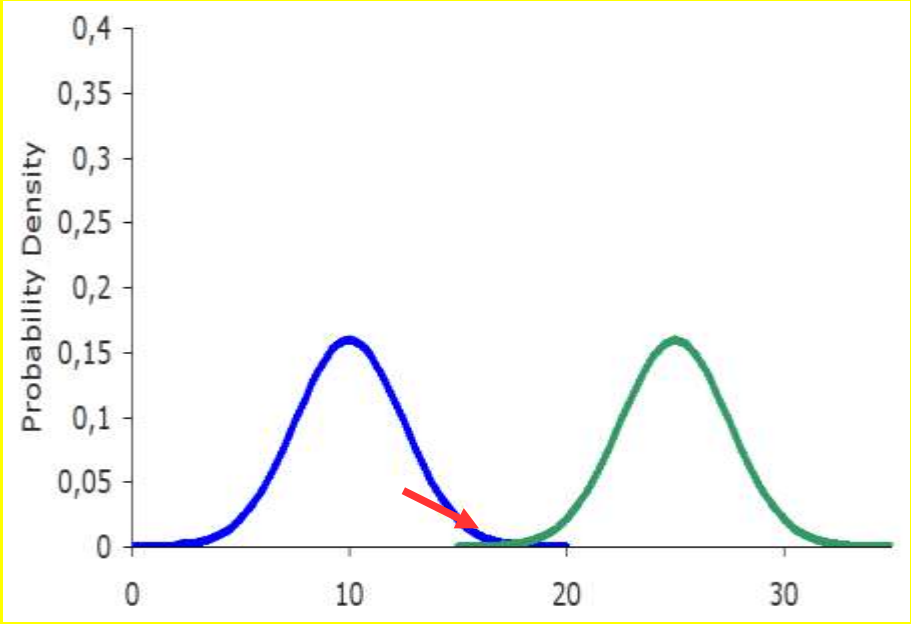
Small overlap
=> (significant) difference

Effektgroesse

- Beispiel: Zwei Gruppen (Messreihen) im Vergleich (Unterschiedshypothese)

Beispiel 2: Unterschiedliche Effektgröße, Gleiche Streuung





Large overlap
=> no (significant) difference

Small overlap
=> (significant) difference

Probleme bei experimentellem Design

- Unterschied zwischen Techniken ggf. aufgrund zu geringer Effektgröße oder zu großer Streuung nicht mehr messbar
  - insb. Programmieraufgaben (10x – Abweichung zwischen Entwicklern), Modellieraufgaben, etc.
  - weniger bei Fragebögen, Reaktionszeiten (Hicksches Gesetz), etc.
- Strategien:
  - Streuung einschränkt
  - Effektgröße betont
- Reduzierung der Streuung
  - Größere Stichprobe, entsprechende Designs
  - Staerkere Selektion

Externe vs. Interne Validitaet

- *„Schreiben Sie einen Browser….“*
  - Wahrscheinlich sehr große Streuung
  - Aber dafür praktische Relevanz
- *„Schreiben Sie Hello World…“*
  - Wahrscheinlich kleine Streuung
  - Geringe praktische Relevanz

- Ziel bei der Aufgabenstellung
  - Zwar Augenmerk auf Relevanz, aber Hauptaugenmerk auf Effektgröße und Streuung

Aufgabenstellung - Praktisches Vorgehen

- „Erfinden“ von Aufgaben, in denen großer Effekt vermutet wird (eigentlich Widerspruch zur Idee des Hypothesentestens!)
  - -> Favorable Case
- Beobachten, ob Haupteffekt gemessen wurde
  - Wenn nein, dann Indiz gefunden, dass vermuteter Effekt nicht existiert
  - Wenn ja, dann weitere Experimente, ob Effekt auch bei „realistischeren“ Aufgaben auftritt
- Aufgaben mit grossem Effekt oft nur bedingt interessant (offensichtlich)

Stoervariablen kontrollieren (Ziel: Streuung verkleinern)

Stoervariablen

- Programmiererfahrung, Ausbildung
- Domaenenwissen, Vorkenntnisse zu Werkzeugen
- Abstraktions-/Problemloese-/Kommunikations-/Gedaechtniss-Fähigkeiten, Intelligenz
- Kultur, Geschlecht
- Motivation, Muedigkeit, Einstellung
- Lesegeschwindigkeit
- Bevorzugung einer Methode, Voreingenommenheit

Kontrollstrategien

- Zufall
  - hohe Streuung, viele Probanden
- Balancieren / Matching
  - vorherige Messung
- Stoervariable als unabhaengige Variable
  - vorherige Messung, aufwendige Designs und Evaluierung, viele Probanden, hohe externe Validitaet
- Konstant halten
  - z.B. gleiche Umgebung, aehnliche Aufgaben, Probandenselektion, keine Generalisierung (gerine externe Validitaet)
- Nachtraegliche Analyse
  - Messung waehrend der Durchfuehrung, kann nachtraeglich bestaetigen dass Ergebnisse unbrauchbar sind

muss Gruppenbildung beeinflussen koennen

muss messbar sein

Stoervariablen

- Programmiererfahrung, Ausbildung
- Domaenenwissen, Vorkenntnisse zu Werkzeugen
- Abstraktions-/Problemloese-/Kommunikations-/Gedaechtniss-Fähigkeiten, Intelligenz
- Kultur, Geschlecht
- Motivation, Muedigkeit, Einstellung
- Lesegeschwindigkeit
- Bevorzugung einer Methode, Voreingenommenheit

- Zufall
- Balancieren/Matching
- unabhaengige Variable
- Konstant halten
- Nachtraegliche Analyse

Experimentelles Design

Experimentelle Designs (1)

- Fragestellung
  - Wie baue ich einen Versuch auf?
- Faktoren
  - Welche Arten von Aufgaben habe ich für Probanden?
  - Wie lang sind die Aufgaben (wahrscheinlich)?
  - Wie viele Probanden habe ich?
  - …
- Um angemessenes Design zu wählen, muss bekannt sein
  - Erwartete Effektgröße
  - Stichprobengröße

Experimentelle Designs (2)

- Generell
  - Je größer Effekt und je größer Stichprobe, desto unwichtiger das Experimentelle Design!
- Aber
  - Typische Softwaretechnikexperimente haben nur sehr geringe Stichprobengröße

- => Experimentelles Design aktuell sehr wichtig

Experimentelle Designs (3)

- Bei Versuchsaufbauen gibt es (mind.)
  - **Effektgruppe**: Gruppe die unter dem Einfluß des Effekts steht (bzw. stehen soll)
  - **Kontrollgruppe** als “Vergleichswert”

  - Terminologie von Effekt- bzw. Kontrollgruppe stammt aus der Medizin/Sozialwissenschaften. Für z.B. die Programmierung ist die Terminologie etwas „problematisch“
  - Bsp: OO vs. Prozedurale Programmierung – die eine Hälfte der Probanden bearbeitet eine Aufgabe oo, die andere prozedural. Welche Gruppe ist nun die Kontrollgruppe?

2-Gruppen Between-Subject Design (Einfaktorieller Versuchsplan)

2-Gruppen Between-Subject

- Charakteristika:
  - Sehr einfaches Design
  - Probanden werden in zwei Gruppen aufgeteilt
  - Jede Gruppe entspricht einem Treatment der freien Variable
  - Messungen der einzelnen Gruppen werden miteinander verglichen
- Probleme
  - Balancierung der Gruppen
  - Große Effektgröße oder viele Probanden notwendig
- Einfache Analyse
  - T-Test
  - Mann-Whitney-U-Test

Group	Levels
A	Comment
B	No comment

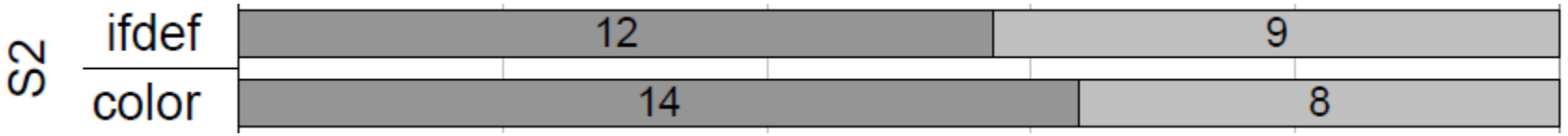
2-Gruppen Between-Subject: Beispiel (1)

- Research question
  - Programmverstaendniss schneller bei Farben als Ifdefs
- Programming Tasks
  - Fehler suchen in Aufgabe
- Setup
  - Independent variable: Quelltextdarstellung
    - Treatment 1: Farben
    - Treatment 2: Ifdefs (Kontrollgruppe)
  - Measurement
    - Time required until error found

2-Gruppen Between-Subject: Beispiel (2)

- “To test RH1, we conducted a Mann-Whitney-U test, because the response times are not normally distributed (as revealed a Shapiro-Wilk test). Since the correctness of a solution can have an influence on response time (e.g., a subject may deliberately enter a wrong solution just to be faster), we omitted response times for wrong answers. Our sample is large enough to compensate the missing cases. The observed differences for the task regarding response time is significant, such that subjects who worked with the color version were faster ($p < 0.001$)“

2-Gruppen Between-Subject: Beispiel (2)



- “For the number of correctly solved tasks (RH3), we conducted X^2 test, which checks whether the observed frequencies significantly differ from expected frequencies under the assumption that the null hypothesis is valid (i.e., that no differences between number of correct answers exist). We found no significant differences in the correctness for any task.”

Probleme between-subject design

- Ausbalancieren der Gruppen problematisch
- Streuung zwischen Probanden i.d.R. höher als Haupteffekt
  - Ergebnis: kein messbarer Unterschied, s.o.
  - Randomisierung braucht große Gruppen
  - Nachträgliches Rausrechnen von Stoervariablen kaum möglich
  - Konstanthalten unrealistisch (zu wenig Probandenauswahl)
- Typische Strategie: Balancieren nach Programmiererfahrung u.ae.
  - Fragebogen + Matching

2-Gruppen Within-Subject Design (Einfaktorieller Versuchsplan*)

* Einfaktoriell = 1 Unabhaengige Variable

Within-Subject Design

- Hintergrund
  - Individuelle Differenzen sollen berücksichtig werden
  - Differenzen zwischen den Probanden sollen geringes Gewicht haben
- Charakteristika
  - Jeder Proband ist in Effekt- als auch Kontrollgruppe (jeder Proband macht jede Aufgabe 2x)
  - Messungen der einzelnen Probanden werden miteinander verglichen
- Einfache Analyse
  - Wilcoxon-Test / T-Test für Abhaengige Stichproben

One Group	Session 1	Session 2
	Comment	No Comment

One Group	Session 1	Session 2
	With exercise	Without exercise

77	11
69	61
32	21
48	91
88	37
35	25
71	64
19	28
65	4
75	70
52	73
13	68
72	16
50	

One Group	Session 1	Session 2
	With pill	Without pill

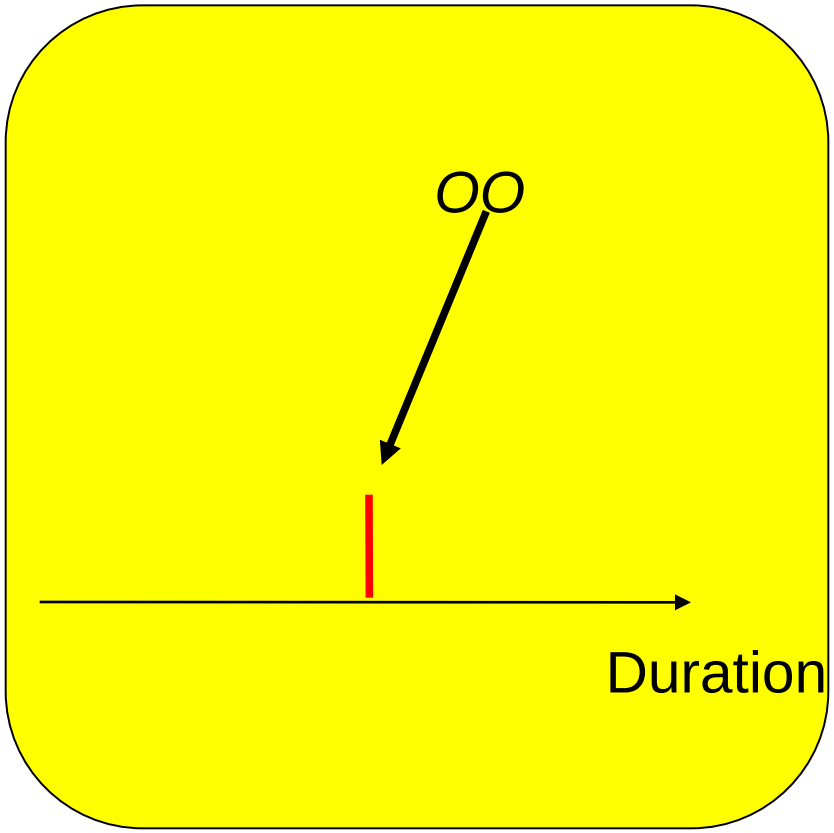
Probleme within-subject design

- Zweiter Durchlauf schneller wegen **Lerneffekt** (auch carryover effect), nicht wegen zweiten Treatment
- 2x die identische Aufgabe loesen
  - Konzept bereits im ersten Durchlauf entwickelt, im zweiten “nur noch” umsetzen
  - Exakte Wiederholung im Softwarebereich unrealistisch (Kreative Taetigkeit)
  - Fast nur bei sehr mechanischen Aufgaben anwendbar
- Andere Effekte: zunehmende Muedigkeit, abnehmende Motivation, …

One Group	Session 1	Session 2
	Comment	No Comment

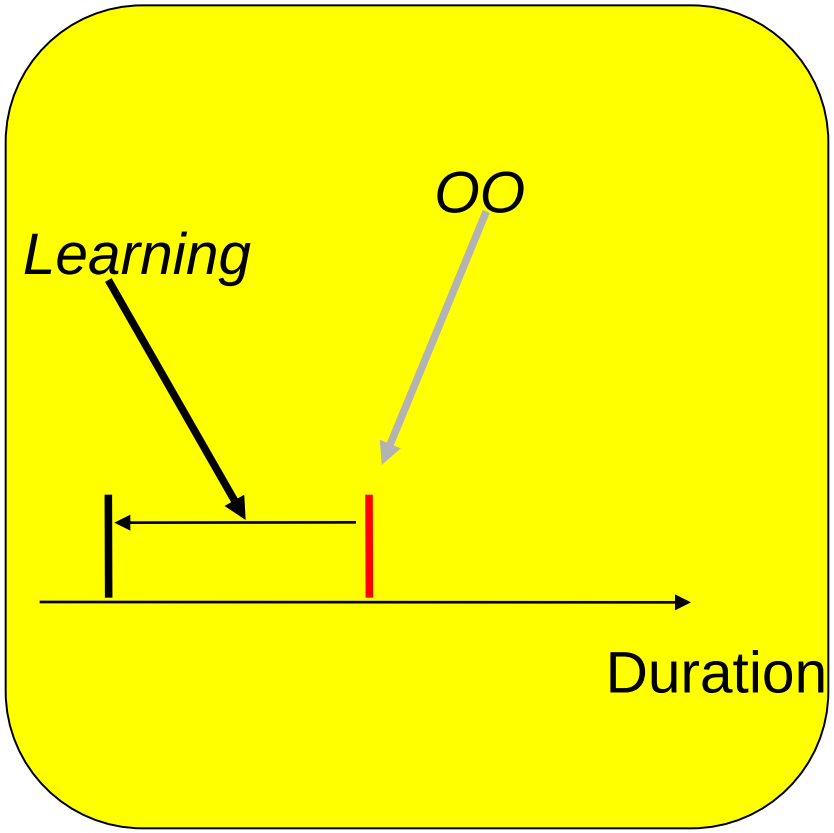
Carryover Effects

- A first



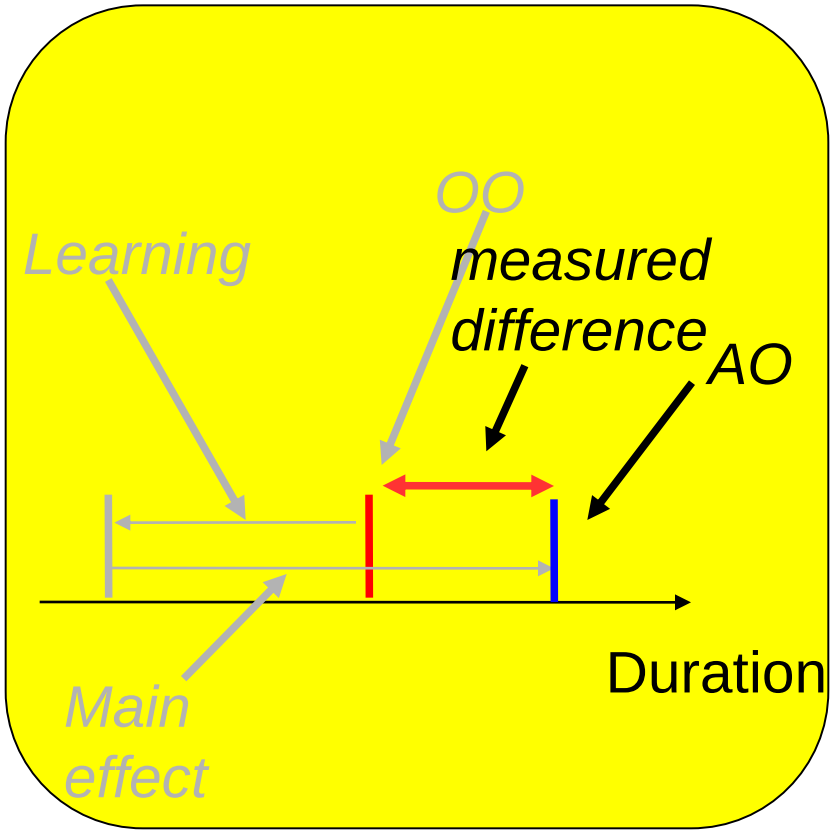
Carryover Effects

- A first



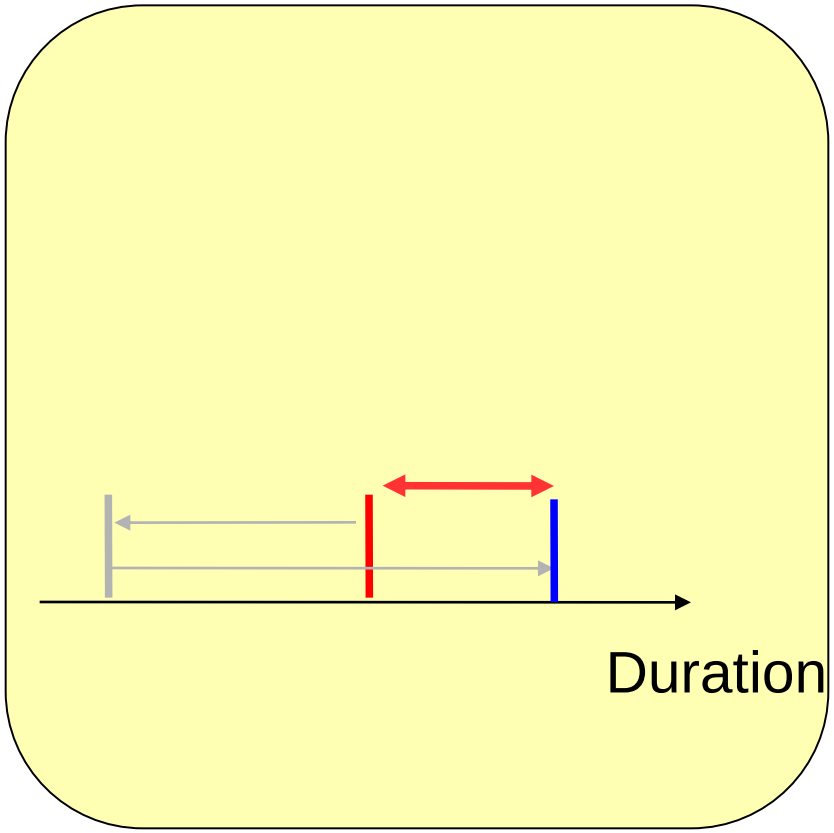
Carryover Effects

- A first

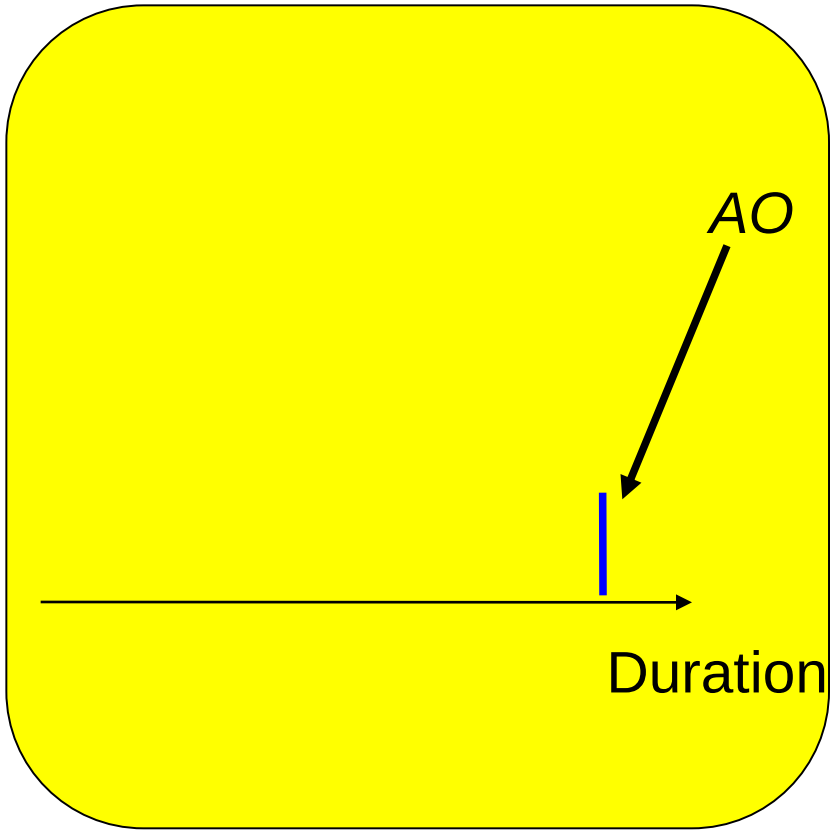


Carryover Effects

- A first

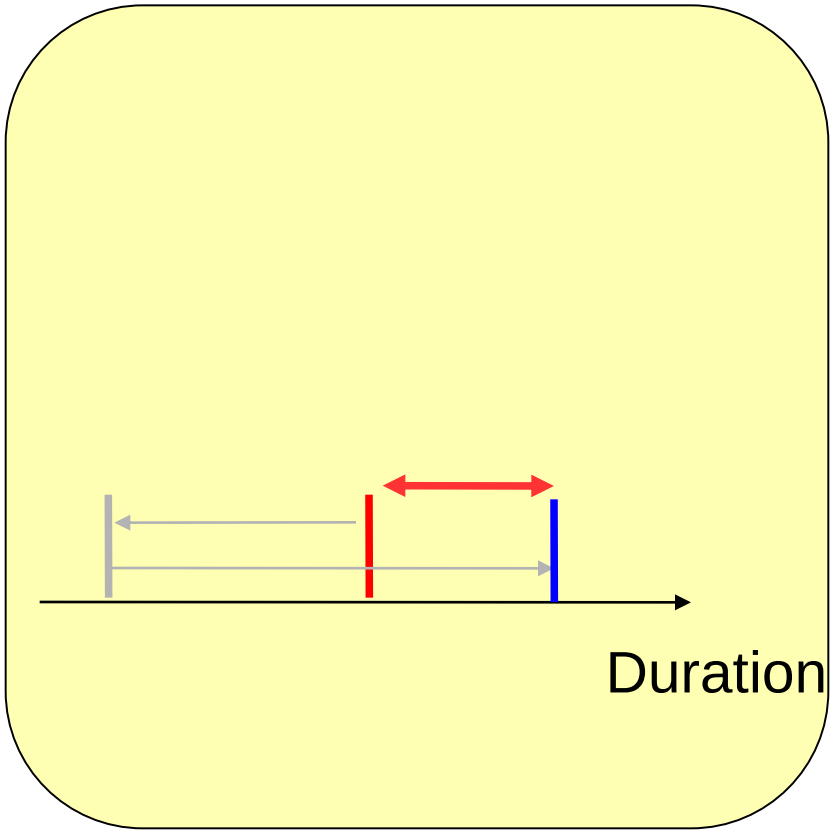


- B first

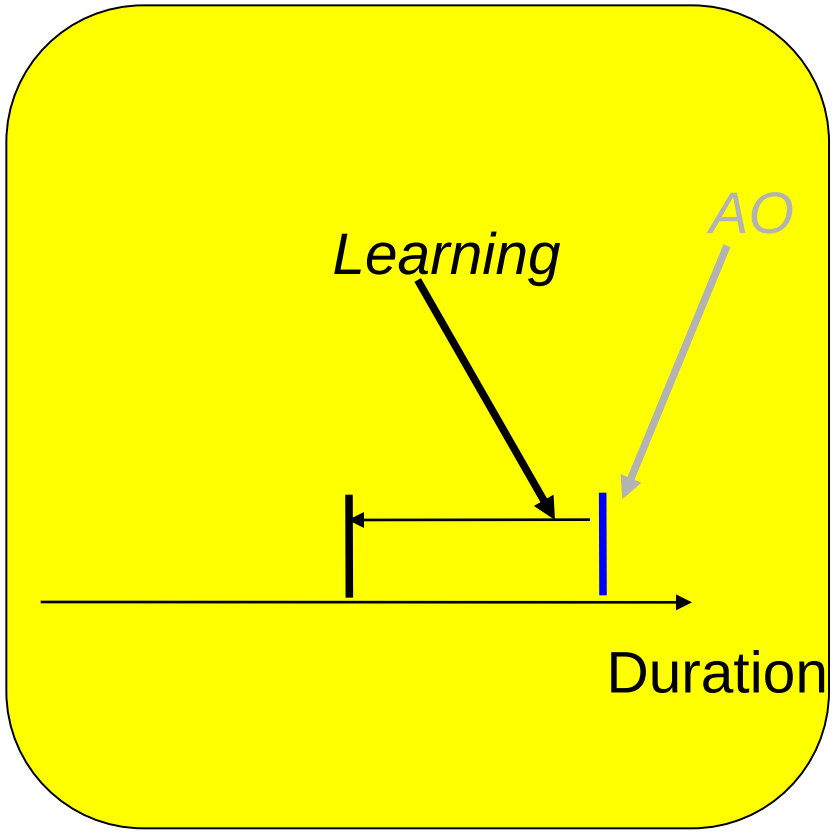


Carryover Effects

- A first

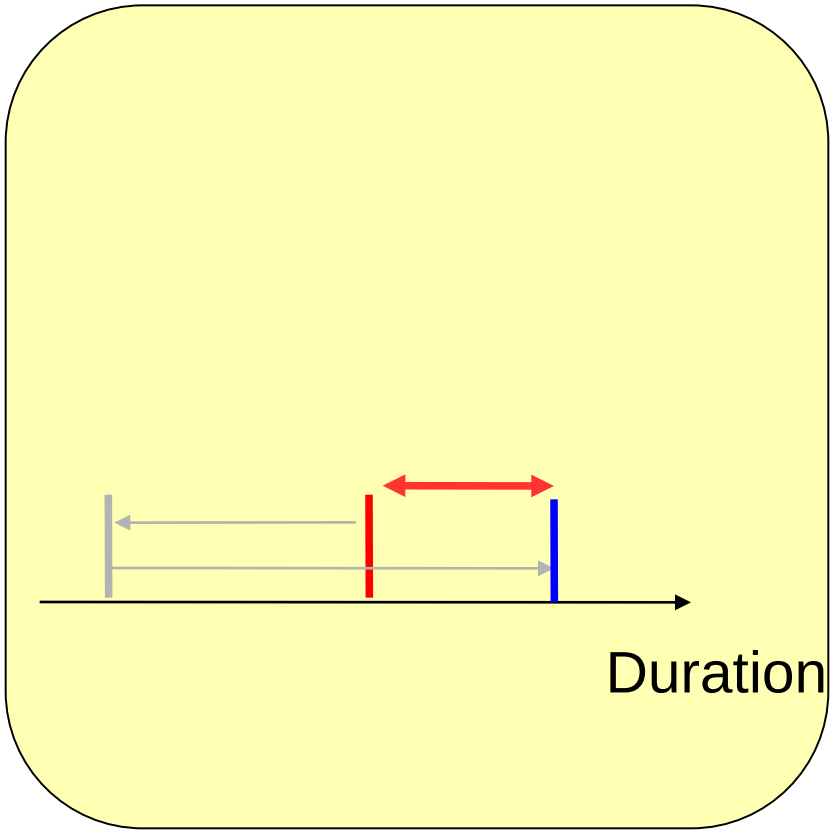


- B first

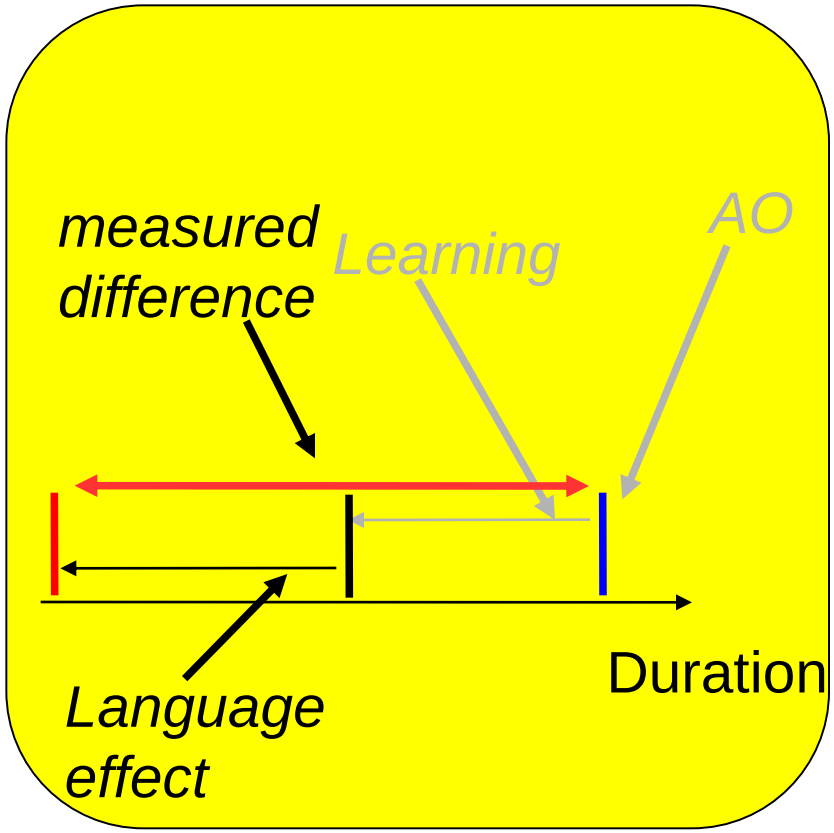


Carryover Effects

- A first



- B first

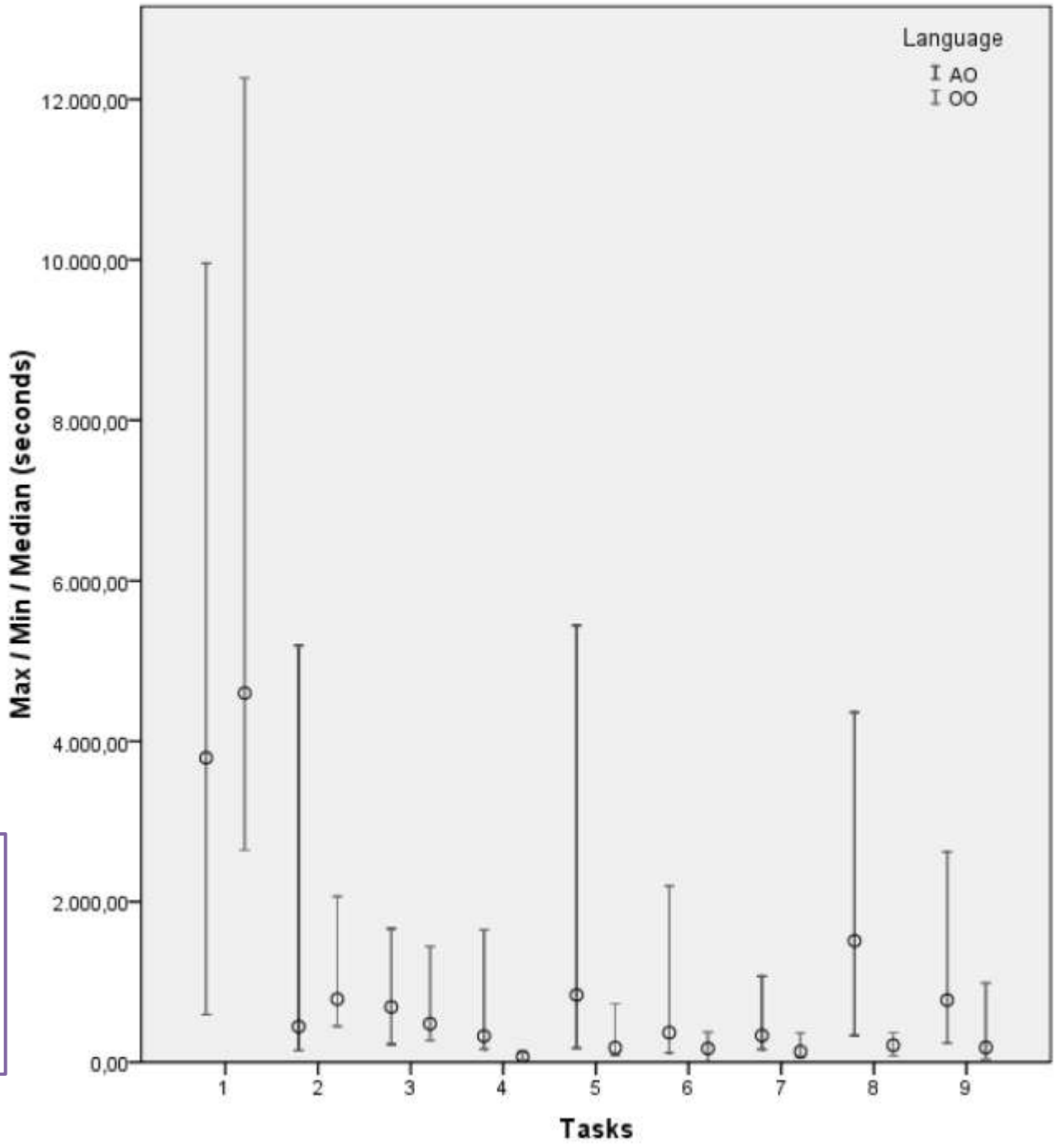


Diskussion “sehr aehnliche Aufgaben”

- Wenn Wiederholung mit identischen Aufgaben unmoeglich
- Sehr aehnliche Aufgaben finden
  - insb. bei mechanischen, nicht-kreativen Aufgaben
  - gut argumentieren dass sie wirklich vergleichbar sind
  - wenn moeglich mit Vortests belegen
  - Durchschnitt ueber viele aehnliche Aufgaben

- z.B. Behaupten zwei Bug-Finding Aufgaben seien sehr aehnlich -> kaum zu belegen

One Group	Session 1	Session 2
	Comment	No Comment



Hanenberg, Kleinschmager, Josupeit-Walter: Does aspect-oriented programming increase the development speed for crosscutting code? An empirical study. ESEM 2009: 156-167

Crossover Design AB/BA (Einfaktorieller Versuchsplan)

Zwei Gruppen-Design (AB / BA)

- Within-Subject Design mit 2 Gruppen
- Lerneffekte in beide Richtungen
  - Annahme: Lerneffekt sind in beide Richtungen gleich gross und gleichen sich daher aus
- Statistischer Test über beide Gruppen gleichzeitig

- Lerneffekt nicht bestimmbar/rausrechenbar
- Lerneffekt sollte kleiner sein als Haupteffekt; oft unrealistisch bei identischen Aufgaben

Group	Session 1	Session 2
A	Comment	No Comment
B	No comment	Comment

Mehrfaktorielle Versuchsplaene* (Designs mit mehreren Aufgaben)

* Mehrfaktoriell = 2..n unabhangige Variablen

Latin Square Design

- Aehnlich zu AB-BA, aber mit aehnlichen statt identischen Aufgaben
- Aufgabe behandelt wie weitere unabhaengige Variable
- Unterschied erklaerbar durch Haupteffekt oder unterschiedliche Aufgaben
- Annahme: Keine/kleine Interaktionseffekte

Group	Aufgabe 1	Aufgabe 2
A	Comment	No Comment
B	No comment	Comment

4/8-Gruppen Between-Subject Design (1)

- Charakteristika (für z.B. Vergleich 2er Sprachen)
  - Probanden werden in 4 Gruppen verteilt
  - 2 (ähnliche) Aufgaben je Gruppe
  - Gruppen sind Kombination aus Aufgaben und Sprachen
- Sehr viele Gruppen noetig, d.h. viele Probanden
  - gleich Between-Subject Design moeglich
  - Interaktionen und Lerneffekte ermittelbar, aber i.d.R. nicht von Interesse

Aufgabe 1/Sprache 1, dann Aufgabe 2/Sprache 2	Aufgabe 1/Sprache 1, dann Aufgabe 2/Sprache 1
Aufgabe 1/Sprache 2, dann Aufgabe 2/Sprache 1	Aufgabe 1/Sprache 2, dann Aufgabe 2/Sprache 2

Aufgabe 2/Sprache 1, dann Aufgabe 1/Sprache 2	Aufgabe 2/Sprache 1, dann Aufgabe 1/Sprache 1
Aufgabe 2/Sprache 2, dann Aufgabe 1/Sprache 1	Aufgabe 2/Sprache 2, dann Aufgabe 1/Sprache 2

Latin Square, 4-Gruppen Between Subject, 8-Gruppen Between Subject

4-Gruppen Between-Subject Design (2)

- Aufwendige Analyse
  - Varianzanalyse (ANOVA)

Diskussion Designs

- Between Subject Design (Einfaktorial)
  - Einfachstes Design, einfache Auswertung
  - Benoetigt viele Probanden oder Balancieren von Gruppen
- Within-Subject Design (Einfaktorial)
  - Einfaches Design, einfache Auswertung,
  - Wiederholung identischer Aufgaben ohne Lerneffekte bei kreativen Aufgaben kaum moeglich; Lerneffekte nicht messbar
- Mehrfaktoriale Plaene
  - Relativ einfach, erlaubt mehre Aufgaben als weitere unabhaengige Variable
  - Komplizierte Auswertung
  - Latin Square
    - Relativ wenig Probanden, einschraenkende Annahmen bezgl. Interaktionen
  - 4/8-Gruppen Designs
    - Viele Effekte/Interaktionen ermittelbar, hohe externe Validitaet
    - Benoetigt viele Probanden, komplizierte Auswertung

Threats to Validity

Validität (Threats to Validity)

- Interne Validität
  - In sich schluessig? Alternativerklaerungen ausschliesbar?
  - insb.: Stoervariablen kontrolliert?
- Externe Validität
  - Allgemeingültigkeit, Verallgemeinerungsfähigkeit?
  - Waren die Aufgaben realistisch und typisch?
  - Waren die Probanden repraesentativ?
- Konstrukt Validitaet
  - Waren die Messmethoden fuer die abhaengige Variable angemessen?
  - Sagt die abhaengige Variable was ueber die Fragestellung aus?

Ergebnisse Berichten

Richtlinien

- Experiment Planning
  - Goals
  - Experimental units (samples, groups, randomization; begruendet)
  - Experimental material (tasks; begruendet)
  - Hypothesis, parameters, and variables (unless exploratory)
  - Design (between-subject design etc; begruendet)
  - Analysis procedure (planned analysis)
- Execution
  - Preparation, Deviations
- Analysis
  - Descriptive statistics
  - Hypothesis testing
- Discussion
  - Evaluation of results and implications
  - Threats to validity

Fuer Details und Erklaerungen siehe: Jedlitschka, Ciolkowski, and Pfahl. **Reporting experiments in software engineering**. In *Guide to advanced empirical software engineering* (Kapitel 8). Springer 2008

Empfehlungen an Gutachter

- Don’t expect perfection
- Don’t expect a chapter out of a statistics textbook
- Don’t expect decisive answers
- Don’t reject “obvious” results
  - opinions, there may be skeptics; should be interesting though
- Don’t be casual about asking authors to redo their experiments
- Don’t dismiss a paper merely for using students as subjects
- Don’t reject negative results
- Don’t reject repetition of experiments

Walter Tichy. **Hints for reviewing empirical work in software engineering**. Empirical Software Engineering 5, 309-312. 2000

Aufgabe: Paper-Diskussion

- Lesen Sie zum 9.7 folgenden Artikel
  - Stefan Hanenberg: **An experiment about static and dynamic type systems: doubts about the positive impact of static type systems on development time**. In *Proc. OOPSLA, ACM,* 2010, p. 22-35
- Bereiten Sie sich auf eine Gruppendiskussion vor
  - Welches Experimentelle Design wurde verwendet und warum?
  - Welche Stoervariablen wurden beruecksichtigt, wie wurde mit ihnen umgegangen? Wurden Gruppen balanciert?
  - Welche Threats to Validity wurden diskutiert? Bewerten Sie interne und externe Validitaet.

Zusammenfassung

- Faktor Mensch fuehrt zu neuen Herausforderungen
- Stoervariablen kontrollieren
- Experimentelle Designs
  - Within-Subject / Between-Subject
  - Mehrfaktorielle Designs
- Validitaet und Berichten von Ergebnissen

Literatur

- Bortz, Döring; **Forschungsmethoden und Evaluation**, Aufl. 4, Springer, 2006. Insb. Kapitel 8
- Bortz, Schuster. **Statistik fuer Human- und Sozialwissenschaftler**, Aufl. 7, Springer 2010. Insb. Kapitel 12ff
- Jedlitschka, Ciolkowski, and Pfahl. **Reporting experiments in software engineering**. In Guide to advanced empirical software engineering (Kapitel 8). Springer 2008

Beispielliteratur

- Between-Subject Design
  - Feigenspan, Kästner, Apel, Liebig, Schulze, Dachselt, Papendieck, Leich, and Saake. Do Background Colors Improve Program Comprehension in the #ifdef Hell? Empirical Software Engineering, 2012.
  - Stefan Hanenberg: An experiment about static and dynamic type systems: doubts about the positive impact of static type systems on development time. OOPSLA 2010, 22-35
  - Marc Bartsch, Rachel Harrison: An exploratory study of the effect of aspect-oriented programming on maintainability. Software Quality Journal 16(1), 2008, 23-44
- Within-Subject Designs
  - Hanenberg, Kleinschmager, Josupeit-Walter: Does aspect-oriented programming increase the development speed for crosscutting code? An empirical study. ESEM 2009: 156-167
  - Feigenspan, Kästner, Apel, Liebig, Schulze, Dachselt, Papendieck, Leich, and Saake. Do Background Colors Improve Program Comprehension in the #ifdef Hell? Empirical Software Engineering, 2012.
  - Kleinschmager, Hanenberg, Robbes, Tanter, Stefik, Do Static Type Systems Improve the Maintainability of Software Systems? An Empirical Study, ICPC 2012, pp. 153-162