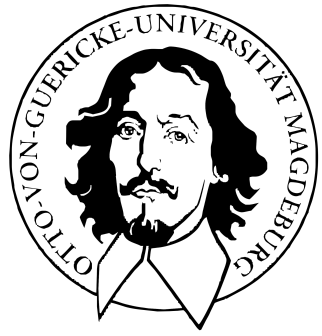


Interventionsforschung

MSc Klinische Psychologie und Psychotherapie

MSc Umweltpsychologie / Mensch-Technik-Interaktion

SoSe 2026

Prof. Dr. Dirk Ostwald

(2) Kovarianzanalyse

Allgemeines lineares Modell

Parallelgruppendesign

Einfache lineare Regression

Kovarianzanalyse

Selbstkontrollfragen

Allgemeines lineares Modell

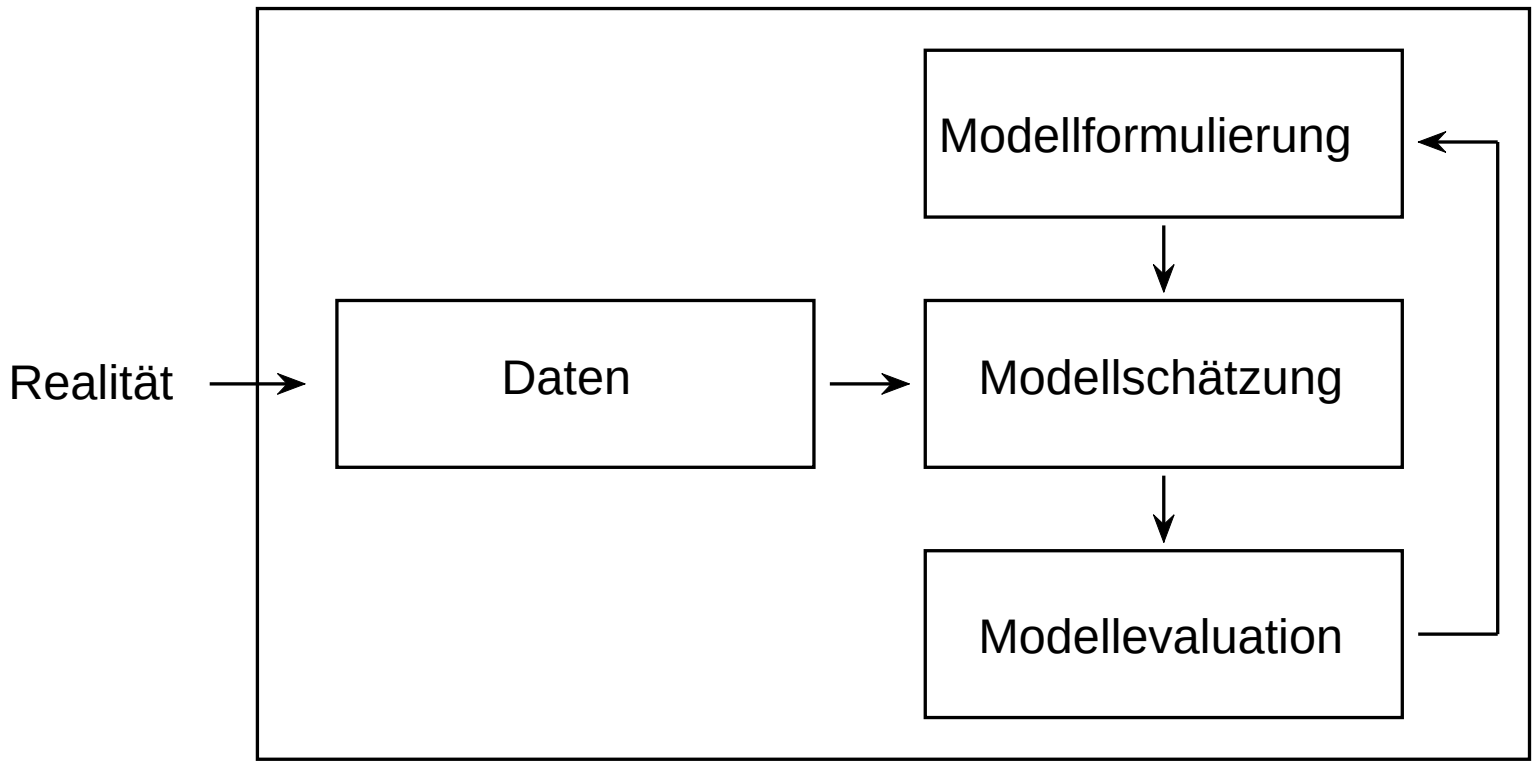
Parallelgruppendesign

Einfache lineare Regression

Kovarianzanalyse

Selbstkontrollfragen

Modellbasierter Realismus



Modellformulierung

$$y = X\beta + \varepsilon, \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{1}$$

Modellschätzung

$$\hat{\beta} = (X^T X)^{-1} X^T y, \quad \hat{\sigma}^2 = \frac{(y - X\hat{\beta})^T (y - X\hat{\beta})}{n - p} \tag{2}$$

Modellevaluation

$$T = \frac{c^T \hat{\beta} - c^T \beta_0}{\sqrt{\hat{\sigma}^2 c^T (X^T X)^{-1} c}}, \quad F = \frac{(\hat{\varepsilon}_0^T \hat{\varepsilon}_0 - \hat{\varepsilon}^T \hat{\varepsilon})/p_1}{\hat{\varepsilon}^T \hat{\varepsilon}/(n - p)} \tag{3}$$

Möglichkeiten der Minderung des Einflusses von Störvariablen in empirischen Studien

Möglichkeiten der Minderung des Einflusses von Störvariablen in empirischen Studien

- Neben den UVen beeinflussen auch andere Störvariablen die Werte der AVen
- Störvariablen können bekannt sein, es sind aber auch unbekannte Störvariablen denkbar

Experimentelle Kontrolle

- Treatment und Kontrollbedingungen finden unter kontrollierten Störvariablen-Bedingungen statt
- Die Kontrolle von Störvariablen schränkt die Generalisierbarkeit von Ergebnissen ein
- Ergebnisse gelten nur für die spezifisch gewählten Bedingungen der Störvariablen

Randomisierung

- Proband:innen werden zufällig auf die Experimentalbedingungen aufgeteilt
- Man hofft auf den Ausgleich von Störeffekten im Durchschnitt
- Randomisierung kann sowohl bekannte als auch unbekannte Störvariablen kontrollieren

Statistische Anpassung im Rahmen des Allgemeinen Linearen Modells (Kovarianzanalyse)

- Im Rahmen der Datenanalyse werden Störvariablen modellbasiert posthoc kontrolliert
- Voraussetzung ist die Messung der Störvariablen während der Studiendurchführung
- Voraussetzung ist weiterhin ein gewisses Maß an Unabhängigkeit von Störvariablen und UVen

Linden (2016)

Datenvariabilitätszerlegung

Variabilität der Werte einer abhängigen Variable

- Die Werte einer AV unterscheiden sich im Allgemeinen zwischen experimentellen Einheiten.
- Die Schwankungen der Werte einer AV bezeichnet man als *Datenvariabilität*.
- Ziel jeder Datenanalyse ist die Erklärung von Datenvariabilität durch Zerlegung.

Datenvariabilität und Allgemeines lineares Modell

- Eine spezielle Art der Quantifizierung von Datenvariabilität ist die Stichprobenvarianz.
- Einen additiven Zugang zur Dekomposition von Datenvariabilität bietet das ALM.
- Den folgenden Überlegungen entspricht datenanalytisch insbesondere die Kovarianzanalyse.

Gesamtvariabilität = Primärvariabilität + Residualvariabilität

Primärvariabilität

Systematische Veränderung der AVen, die allein auf Variation der UVen zurückzuführen ist.

Residualvariabilität = Sekundärvariabilität + Zufallsfehler

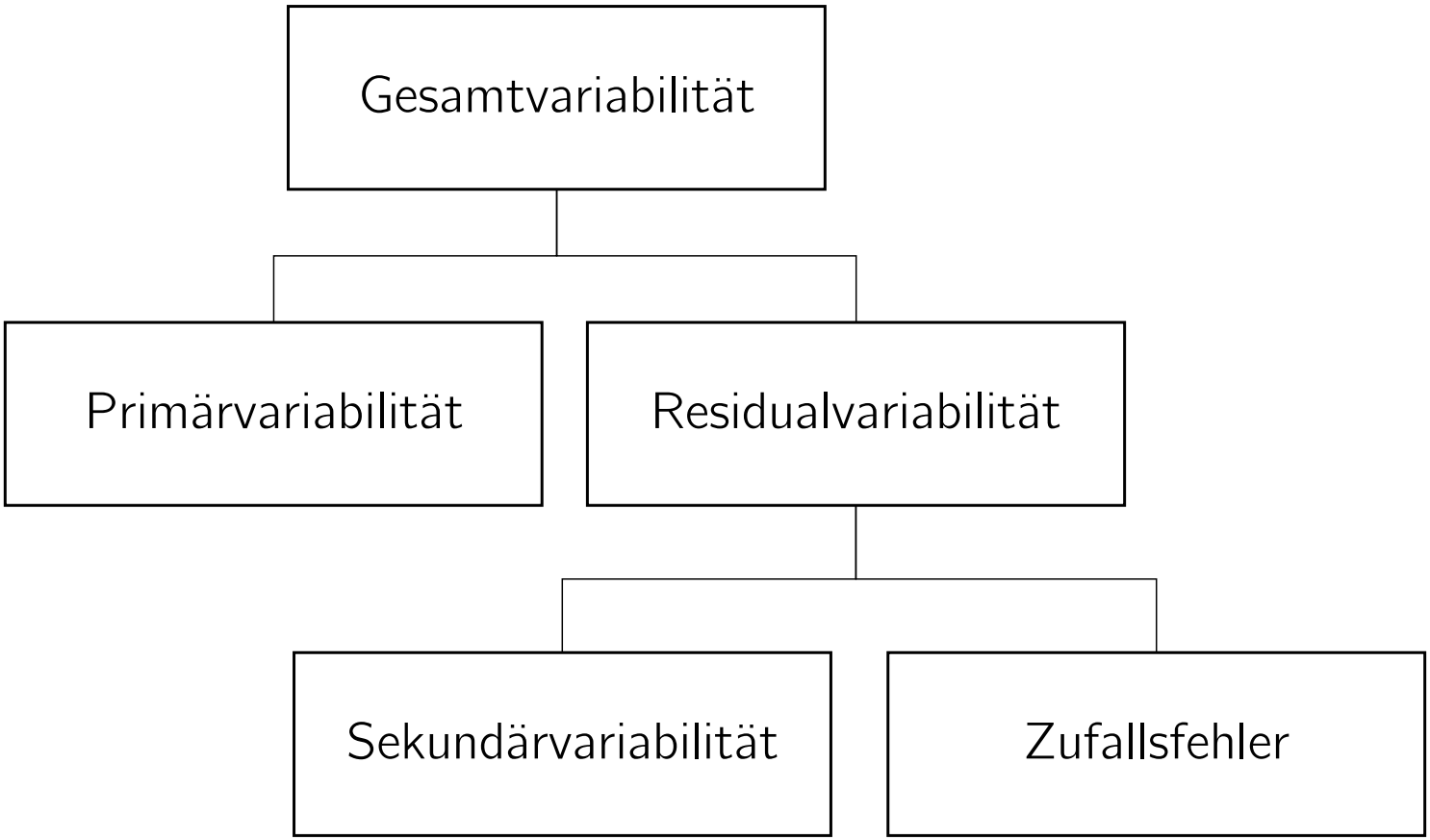
Sekundärvariabilität

Systematische Veränderung der AVen, die auf die Wirkung von unkontrollierten Störvariablen, nicht aber auf die Variation der UVen, zurückzuführen ist.

Zufallsfehler

Unsystematische Veränderung der AVen, die weder auf die Variation der UVen, noch auf den Einfluss von Störvariablen zurückzuführen ist.

Allgemeines lineares Modell



Allgemeines lineares Modell (ALM)

- Mathematischer Framework, in dem viele lineare statistische Methoden formuliert werden können
- Daten als lineare Kombination systematischer Einflüsse plus normalverteilte Zufallsfehler
- Normalverteilte Zufallsfehler als summarische Repräsentation aller nicht-modellierten Einflüsse auf die Daten
- Verschiedenen Verfahren unterscheiden sich bezüglich Designmatrix und Betaparameterinterpretation

Kategoriale Designs

- Spalten der Designmatrix repräsentieren experimentelle Faktoren
- Level von Faktoren werden mit 1en und 0en, manchmal -1en, kodiert
- Betaparameter repräsentieren Gruppenerwartungswerte oder Kontraste.
- $\Rightarrow$ T-Tests, Einfaktorielle Varianzanalyse, Mehrfaktorielle Varianzanalyse

Kontinuierliche Designs

- Spalten der Designmatrix werden als Regressoren, Prädiktoren oder Kovariaten bezeichnet
- Betaparameter repräsentieren Interzepte und Steigungen
- Enge Bezüge zur Korrelationsanalyse, insbesondere zur partiellen Korrelation
- $\Rightarrow$ Einfache lineare Regression, Multiple lineare Regression

Kombination kategorialer und kontinuierlicher Designs

- Integration beider Designs in einer gemeinsamen Designmatrix
- Betaparameter repräsentieren Gruppenerwartungswerte, Kontraste, Interzepte und Steigungen
- $\Rightarrow$ Kovarianzanalyse und Moderationsanalyse

Definition (Allgemeines lineares Modell)

Es sei

$$y = X\beta + \varepsilon, \tag{4}$$

wobei

- y ein n-dimensionaler beobachtbarer Zufallsvektor ist, der *Daten* genannt wird,
- $X \in \mathbb{R}^{n \times p}$ mit $n > p$ eine vorgegebene Matrix ist, die *Designmatrix* genannt wird,
- $\beta \in \mathbb{R}^p$ ein unbekannter Parametervektor ist, der *Betaparametervektor* genannt wird,
- ε ein n-dimensionaler nicht-beobachtbarer Zufallsvektor ist, der *Zufallsfehler* genannt wird und für den angenommen wird, dass mit einem unbekannten Varianzparameter $\sigma^2 > 0$ gilt, dass

$$\varepsilon \sim N\left(0_n, \sigma^2 I_n\right). \tag{5}$$

Dann heißt (4) *Allgemeines lineares Modell (ALM)*.

Allgemeines lineares Modell

Bemerkungen

- Wir nehmen durchgängig an, dass $X \in \mathbb{R}^{n \times p}$ vollen Spaltenrang hat, also dass $\text{rg}(X) = p$.
- y ist ein Zufallsvektor, weil er aus der Addition des Zufallsvektors ε zu dem Vektor $X\beta \in \mathbb{R}^n$ resultiert.
- Wir nennen $X\beta \in \mathbb{R}^n$ den *deterministischen Modellaspekt* und ε den *probabilistischen Modellaspekt*.
- $n \in \mathbb{N}$ bezeichnet durchgängig die Anzahl an Datenpunkten.
- $p \in \mathbb{N}$ bezeichnet durchgängig die Anzahl an Betaparametern.
- Die Gesamtzahl an Parametern des ALMs ist $p + 1$ (p Betaparameterkomponenten und 1 Varianzparameter).
- Der Betaparametervektor wird auch *Gewichtsvektor* oder *Effektvektor* genannt.
- Weil der Kovarianzmatrixparameter von ε als sphärisch angenommen wird, sind die $\varepsilon_1, \ldots, \varepsilon_n$ unabhängige normalverteilte Zufallsvariablen mit identischem Varianzparameter; weil zusätzlich der Erwartungswertparameter von ε als 0_n angenommen wird, sind die $\varepsilon_1, \ldots, \varepsilon_n$ auch identisch normalverteilte Zufallsvariablen.
- Für jede Komponente $y_i, i = 1, \ldots, n$ von y impliziert (4) nach Definition des Matrixprodukts, dass

$$y_i = x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{ip}\beta_p + \varepsilon_i \text{ mit } \varepsilon_i \sim N\left(0, \sigma^2\right), \tag{6}$$

  wobei $x_{ij} \in \mathbb{R}$ das ijte Element der Designmatrix X bezeichnet.

Theorem (Datenverteilung des Allgemeinen Linearen Modells)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{7}$$

das ALM. Dann gilt

$$y \sim N(\mu, \sigma^2 I_n) \text{ mit } \mu := X\beta \in \mathbb{R}^n. \tag{8}$$

Beweis

Mit dem Theorem zur linear-affinen Transformation multivariater Normalverteilungen gilt für $\varepsilon \sim N(0_n, \sigma^2 I_n)$ und $y := I_n\varepsilon + X\beta$, dass

$$y \sim N\left(I_n 0_n + X\beta, I_n(\sigma^2 I_n)I_n^T\right) = N(X\beta, \sigma^2 I_n) = N(\mu, \sigma^2 I_n) \text{ mit } \mu := X\beta \in \mathbb{R}^n. \tag{9}$$

Bemerkungen

- Im ALM sind die Daten y also ein n-dimensionaler normalverteilter Zufallsvektor mit Erwartungswertparameter $\mu = X\beta \in \mathbb{R}^n$ und Kovarianzmatrixparameter $\sigma^2 I_n \in \mathbb{R}^{n\times n}$.
- Die Komponenten $y_1, ..., y_n$ von y, also die Datenpunkte, sind damit unabhängige, aber im Allgemeinen nicht identisch verteilte, normalverteilte Zufallsvariablen der Form $y_i \sim N\left(\mu_i, \sigma^2\right)$ für $i = 1, ..., n$.

Theorem (Betaparameterschätzer)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{10}$$

das ALM und es sei

$$\hat{\beta} := \left(X^T X\right)^{-1} X^T y. \tag{11}$$

der *Betaparameterschätzer*. Dann gilt, dass $\hat{\beta}$ die Summe der Abweichungsquadrate minimiert,

$$\hat{\beta} = \arg\min_{\tilde{\beta}} (y - X\tilde{\beta})^T (y - X\tilde{\beta}), \tag{12}$$

und dass $\hat{\beta}$ ein unverzerrter Maximum-Likelihood Schätzer von $\beta \in \mathbb{R}^p$ ist.

Bemerkungen

- Das Theorem gibt eine Formel an, um β anhand von Designmatrix und Daten zu schätzen.
- Da $\hat{\beta}$ die Summe der Abweichungsquadrate minimiert, heißt $\hat{\beta}$ auch Kleinste-Quadrate (KQ) Schätzer.
- Die $\tilde{\beta}$-Notation des Maximierungsarguments dient lediglich zur Abgrenzung vom w.a.u. β.
- Als ML Schätzer ist $\hat{\beta}$ weiterhin konsistent, asymptotisch normalverteilt und asymptotisch effizient.
- Wir sehen später, dass $\hat{\beta}$ sogar normalverteilt ist.
- Außerdem hat $\hat{\beta}$ die "kleinste Varianz" in der Klasse der linearen unverzerrten Schätzer von β.
- Letztere Eigenschaft ist Kernaussage des *Gauss-Markov Theorems*, auf das wir hier nicht näher eingehen wollen.
- Für eine Diskussion und einen Beweis des Gauss-Markov Theorems siehe z.B. S. R. Searle (1971), Kapitel 3.

Theorem (Frequentistische Verteilung des Betaparameterschätzers)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{13}$$

das ALM. Weiterhin sei

$$\hat{\beta} := \left(X^T X\right)^{-1} X^T y \tag{14}$$

der Betaparameterschätzer. Dann gilt

$$\hat{\beta} \sim N\left(\beta, \sigma^2 (X^T X)^{-1}\right). \tag{15}$$

Bemerkungen

- Es gilt also wie bereits gesehen $\mathbb{E}(\hat{\beta}) = \beta$ und außerdem $\mathbb{C}(\hat{\beta}) = \sigma^2 (X^T X)^{-1}$.
- Die Varianzen der Komponenten von $\hat{\beta}$ sind die Diagonalelemente von $\mathbb{C}(\hat{\beta})$, also

$$\mathbb{V}(\hat{\beta}_i) = (\sigma^2 (X^T X)^{-1})_{ii} \text{ für } i = 1, \ldots p. \tag{16}$$

- Die Streuung von $\hat{\beta}$ hängt von σ^2 und der Designmatrix X ab. σ^2 ist ein experimentell nicht zu beinflussender wahrer, aber unbekannter, Parameter
- X dagegen kann so gewählt werden, um zum Beispiel die Diagonalelemente von $\mathbb{C}(\hat{\beta})$ bei festem σ^2 zu minimieren.

Theorem (Varianzparameterschätzer)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{17}$$

das ALM in generativer Form. Dann ist

$$\hat{\sigma}^2 := \frac{(y - X\hat{\beta})^T (y - X\hat{\beta})}{n - p} \tag{18}$$

ein unverzerrter Schätzer von $\sigma^2 > 0$.

Bemerkungen

- Es handelt sich bei $\hat{\sigma}^2$ *nicht* um einen Maximum Likelihood Schätzer von σ^2.
- Für einen Beweis siehe z.B. S. R. Searle (1971), Kapitel 3 oder Rencher and Schaalje (2008), Kapitel 7.
- Mit Definition des Residuenvektors und der Residuen bieten sich für $\hat{\sigma}^2$ auch folgende Schreibweisen an:

$$\hat{\sigma}^2 = \frac{\hat{\varepsilon}^T \hat{\varepsilon}}{n - p} = \frac{1}{n - p} \sum_{i=1}^{n} \hat{\varepsilon}_i^2 = \frac{1}{n - p} \sum_{i=1}^{n} \left(y_i - (X\hat{\beta})_i \right)^2 \tag{19}$$

- σ^2 wird also durch eine skalierte Residualquadratsumme geschätzt.
- Der Maximum Likelihood Schätzer des Varianzparameters ist $\hat{\sigma}^2_{\text{ML}} := \frac{1}{n} \hat{\varepsilon}^T \hat{\varepsilon}$.

Theorem (Frequentistische Verteilung des Varianzparameterschätzers)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{20}$$

das ALM. Weiterhin sei

$$\hat{\sigma}^2 = \frac{(y - X\hat{\beta})^T (y - X\hat{\beta})}{n - p} \tag{21}$$

der Varianzparameterschätzer. Dann gilt

$$\frac{n-p}{\sigma^2} \hat{\sigma}^2 \sim \chi^2(n-p) \tag{22}$$

Bemerkungen

- Wir verzichten auf einen Beweis. Da es sich bei $(y - X\hat{\beta})^T (y - X\hat{\beta})$ um eine Summe quadrierter normalverteilter Zufallsvariablen handelt, liegt die χ^2-Verteilung im Lichte der χ^2-Transformation zumindest nahe.

Definition (T-Statistik)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N\left(0_n, \sigma^2 I_n\right) \tag{23}$$

das ALM. Weiterhin seien

$$\hat{\beta} := (X^T X)^{-1} X^T y \text{ und } \hat{\sigma}^2 := \frac{(y - X\hat{\beta})^T (y - X\hat{\beta})}{n - p} \tag{24}$$

die Betaparameter- und Varianzparameterschätzer, respektive. Dann ist für einen *Kontrastgewichtsvektor* $c \in \mathbb{R}^p$ und einen Parameter $\beta_0 \in \mathbb{R}^p$ die *T-Statistik* definiert als

$$T := \frac{c^T \hat{\beta} - c^T \beta_0}{\sqrt{\hat{\sigma}^2 c^T (X^T X)^{-1} c}}. \tag{25}$$

Bemerkungen

- Die T-Statistik hängt via $\hat{\beta}$ und $\hat{\sigma}^2$ von den Daten y ab.
- Der Kontrastgewichtsvektor projiziert $\hat{\beta}$ auf einen Skalar $c^T \hat{\beta} \in \mathbb{R}$.
- Die Wahl p-dimensionaler Einheitsvektoren für c erlaubt die Auswahl einzelner Komponenten von $\hat{\beta}$ bzw. β_0.
- Eine generelle Wahl von c erlaubt die Evaluation beliebiger Linearkombinationen von $\hat{\beta}$ bzw. β_0.

Bemerkungen (fortgeführt)

Die Wahl von $\beta_0 \in \mathbb{R}^p$ erlaubt es, die T-Statistik unterschiedlich einzusetzen:

- Wählt man $\beta_0 := 0_p$, so erhält man mit der T-Statistik eine Deskriptivstatistik, die es erlaubt, geschätzte Regressoreffekte, also Komponenten oder Linearkombinationen von $\hat{\beta}$, im Sinne eines Signal-zu-Rauschen Verhältnisses in Bezug zu der durch $\hat{\sigma}^2$ quantifizierten Residualdatenvariabilität zu setzen. Der Nenner der T-Statistik stellt dabei sicher, dass insbesondere die adäquate (Ko)Standardabweichung der entsprechenden Betaparameterkomponentenkombination als Bezugsgröße dient, da es sich bei $\hat{\sigma}^2 \left(X^T X\right)^{-1}$ bekanntlich um die Kovarianz des Betaparameterschätzers handelt. Folgende erste Intuition ist in diesem Kontext hilfreich:

$$T = \frac{\text{Geschätzte Effektstärke}}{\text{Geschätzte stichprobenumfangskalierte Datenvariabilität}} \tag{26}$$

- Wählt man für $\beta_0 = \beta$, also den wahren, aber unbekannten, Betaparameterwert, so eröffnet die T-Statistik die Möglichkeit, für einzelne Komponenten des Betaparametervektors Konfidenzintervalle zu bestimmen.
- Deklariert man schließlich $\beta_0 \in \Theta_0$ im Kontext eines Testszenarios als das Element einer Nullhypothese Θ_0, so eröffnet die T-Statistik die Hypothesentest-basierte Inferenz über Betaparameterkomponenten und ihrer Linearkombinationen.

Theorem (Frequentistische Verteilung der T-Statistik)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{27}$$

das ALM. Weiterhin seien

$$\hat{\beta} := (X^T X)^{-1} X^T y \text{ und } \hat{\sigma}^2 := \frac{(y - X\hat{\beta})^T (y - X\hat{\beta})}{n - p} \tag{28}$$

die Betaparameter- und Varianzparameterschätzer, respektive. Schließlich sei für einen Kontrastgewichtsvektor $c \in \mathbb{R}^p$ und einen Parameter $\beta_0 \in \mathbb{R}^p$

$$T := \frac{c^T \hat{\beta} - c^T \beta_0}{\sqrt{\hat{\sigma}^2 c^T (X^T X)^{-1} c}} \tag{29}$$

die T-Statistik. Dann gilt

$$T \sim t(\delta, n - p) \text{ mit } \delta := \frac{c^T \beta - c^T \beta_0}{\sqrt{\sigma^2 c^T (X^T X)^{-1} c}}. \tag{30}$$

Bemerkungen

- Wir verzichten auf einen Beweis.
- T ist eine Funktion der Parameterschätzer, δ ist eine Funktion der wahren, aber unbekannten, Parameter
- Für $c^T \beta = c^T \beta_0$, also bei Zutreffen der Nullhypothese, gilt $\delta = 0$ und damit $T \sim t(n - p)$.
- Unter der Annahme des Zutreffens der Nullhypothese wird $t(n - p)$ zur Bestimmung von p-Werten genutzt.
- Für $c^T \beta \neq c^T \beta_0$ kann die Verteilung von T zur Herleitung von Powerfunktionen genutzt werden.

Theorem (Konfidenzintervalle für Betaparameterkomponenten)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{31}$$

das ALM, $\hat{\beta}$ und $\hat{\sigma}^2$ seien die Betaparameter- und Varianzparameterschätzer, respektive und für ein $\delta \in]0,1[$ sei

$$t_\delta := \Psi^{-1}\left(\frac{1+\delta}{2}; n-p\right). \tag{32}$$

Schließlich sei für $j = 1, \ldots p$

$$\lambda_j := \left(\left(X^T X\right)^{-1}\right)_{jj} \text{ das } j\text{te Diagonalelement von } \left(X^T X\right)^{-1}. \tag{33}$$

Dann ist für $j = 1, \ldots p$

$$\kappa_j := \left[\hat{\beta}_j - \hat{\sigma}\sqrt{\lambda_j} t_\delta, \hat{\beta}_j + \hat{\sigma}\sqrt{\lambda_j} t_\delta\right] \tag{34}$$

ein δ-Konfidenzintervall für die $j$te Komponente β_j des Betaparameters $\beta = (\beta_1, \ldots, \beta_p)^T$.

Bemerkungen

- Intuitiv gilt im Vergleich zum Konfidenzintervall für den Erwartungswertparameter bei Normalverteilung

$$\hat{\beta}_j \approx \bar{y}, \hat{\sigma} \approx S, \sqrt{\lambda_j} \approx \sqrt{n^{-1}} \text{ und } t_\delta = t_\delta. \tag{35}$$

Theorem (Konfidenzintervalle für Betaparameterkontraste)

Es sei

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n) \tag{36}$$

das ALM, $\hat{\beta}$ und $\hat{\sigma}^2$ seien die Betaparameter- und Varianzparameterschätzer, respektive, und für ein $\delta \in]0, 1[$ sei

$$t_\delta := \Psi^{-1}\left(\frac{1+\delta}{2}; n-p\right). \tag{37}$$

Ferner sei $c \in \mathbb{R}^p$ ein Kontrastgewichtsvektor und

$$\lambda_c := c^T \left(X^T X\right)^{-1} c. \tag{38}$$

Dann ist

$$\kappa_c := \left[c^T\hat{\beta} - \hat{\sigma}\sqrt{\lambda_c}t_\delta, c^T\hat{\beta} + \hat{\sigma}\sqrt{\lambda_c}t_\delta\right] \tag{39}$$

ein δ-Konfidenzintervall für den Betaparameterkontrast $c^T\beta$.

Bemerkungen

- Das vorige Theorem ergibt sich als Spezialfall mit dem Einheitsvektor $c = e_j$.

Allgemeines lineares Modell

Parallelgruppendesign

Einfache lineare Regression

Kovarianzanalyse

Selbstkontrollfragen

Die **R** Implementation des Allgemeinen linearen Modells

- In R wird das ALM mit der Funktion `lm()` formuliert und geschätzt
- `lm()` wird üblicherweise in der Form `alm = lm(formula, data)` aufgerufen
- `lm()` schätzt das durch `formula` spezifizierte Modell anhand der Daten im Dataframe `data`
- Zur Modellevaluation können verschiedene Funktionen auf `alm` angewendet werden
- `summary(alm)` gibt vor allem Schätzer und Konfidenzintervalle aus
- `aov(alm)` gibt ANOVA Tabellen aus
- Eine ausführliche Dokumentation von `lm()` geben Chambers and Hastie (1992)

Allgemeines lineares Modell

Modelle der Form $y = X\beta + \varepsilon$ mit $\varepsilon \sim N(0_n, \sigma^2 I_n)$ werden in R symbolisch durch `formulas` dargestellt

```
Daten ~ Term 1 + Term 2 + ... + Term k
```

- Der `~` Operator trennt die linke Seite und rechte Seite einer `formula`
- `Daten` wird zur Identifikation der abhängigen Variable y genutzt
- `Term 1 + Term 2 + ... + Term k` dient der Spezifikation der Spalten der Designmatrix X
- Die `formulas` Syntax geht zurück auf Wilkinson and Rogers (1973) und Chambers and Hastie (1992)

Terme können numerische Prädiktoren oder kategoriale Faktoren (R `factors`) sein

Die `formula` Syntax ist symbolisch, zur Laufzeit müssen die Prädiktoren und Faktoren nicht spezifiziert sein

Essentielle Operatoren in `formulas` sind `+` und `:`

- `+` fügt der Designmatrix Prädiktoren hinzu, `:` dient der Spezifikation von Interaktionen

Nichtessentielle Operatoren in `formulas` sind `*`, `/`, `%in%`, `-` und `^`

- Für zwei Faktoren `f1` und `f2` gilt beispielsweise `f1*f2 = f1 + f2 + f1:f2`

Allgemeines lineares Modell

Beispiele mit `f1`, `f2` als R `factors` und `x1`, `x2` als numerische Vektoren

```
formula(y ~ f1)          # Spezifikation eines einfaktoriellen ANOVA Designs mithilfe der formula() Funktion
y ~ f1                   # Aufruf der formula() Funktion ist aber nicht nötig, R erkennt formulas auch so
y ~ f1 + f2              # Additives zweifaktorielles ANOVA Design
y ~ f1 + f2 + f1:f2      # Zweifaktorielles ANOVA Design mit Interaktion
y ~ f1 + x1              # Additives einfaktorielles ANCOVA Design mit einer Kovariate

formula(y ~ x1)          # Spezifikation einer einfachen linearen Regresssion mithilfe der formula() Funktion
y ~ x1                   # Aufruf der formula() Funktion ist aber nicht nötig, R erkennt formulas auch so
y ~ 1 + x1               # Explizite Interzeptdefinition bei einfacher linearer Regression, nicht nötig
y ~ 0 + x1               # Verzicht auf Interzeptdefinition bei einfacher lineare Regression
y ~ 1 + x1 + x2          # Multiple Regression mit zwei Regressoren und expliziter Interzeptdefinition
y ~ f1 + x1 + f1:x1      # Einfaktorielles ANCOVA Design mit einer Kovariate und Interaktion
```

Wir betrachten im Folgenden die durch diese `formulas` erzeugten Designmatrizen $X \in \mathbb{R}^{n \times p}$.

Allgemeines lineares Modell

Parallelgruppen = Einfaktorielles ANOVA Design mit zwei Faktorleveln

```r
n   = 12                                          # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2))       # Faktorlevel der Datenpunkte
y   = rnorm(n)                                    # Primäre Zielvariable
D   = data.frame(y = y, f1 = f1)                  # Dataframe
M   = lm(y ~ f1, D)                               # Modellevaluation
X   = model.matrix(M)                             # Designmatrix
```

```
   (Intercept) f12
1            1   0
2            1   0
3            1   0
4            1   0
5            1   0
6            1   0
7            1   1
8            1   1
9            1   1
10           1   1
11           1   1
12           1   1
attr(,"assign")
[1] 0 1
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"
```

Einfaktorielles ANOVA Design mit drei Faktorleveln

```r
n   = 12                                          # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,2,2,2,2,3,3,3,3))       # Faktorlevel der Datenpunkte
y   = rnorm(n)                                    # Primäre Zielvariable
D   = data.frame(y = y, f1 = f1)                  # Dataframe
M   = lm(y ~ f1, D)                               # Modellevaluation
X   = model.matrix(M)                             # Designmatrix
```

```
   (Intercept) f12 f13
1            1   0   0
2            1   0   0
3            1   0   0
4            1   0   0
5            1   1   0
6            1   1   0
7            1   1   0
8            1   1   0
9            1   0   1
10           1   0   1
11           1   0   1
12           1   0   1
attr(,"assign")
[1] 0 1 1
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"
```

Allgemeines lineares Modell

Zweifaktorielles additives ANOVA Design mit jeweils zwei Faktorleveln

```r
n   = 12                                          # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2))       # Faktor-1-Level der Datenpunkte
f2  = as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2))       # Faktor-2-Level der Datenpunkte
y   = rnorm(n)                                    # Primäre Zielvariable
D   = data.frame(y = y, f1 = f1, f2 = f2)         # Dataframe
M   = lm(y ~ f1 + f2, D)                          # Modellevaluation
X   = model.matrix(M)                             # Designmatrix
```

```
   (Intercept) f12 f22
1            1   0   0
2            1   0   0
3            1   0   0
4            1   0   1
5            1   0   1
6            1   0   1
7            1   1   0
8            1   1   0
9            1   1   0
10           1   1   1
11           1   1   1
12           1   1   1
attr(,"assign")
[1] 0 1 2
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"

attr(,"contrasts")$f2
[1] "contr.treatment"
```

Allgemeines lineares Modell

Zweifaktorielles additives ANOVA Design mit Interaktion

```r
n   = 12                                                    # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2))                 # Faktor-1-Level der Datenpunkte
f2  = as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2))                 # Faktor-2-Level der Datenpunkte
y   = rnorm(n)                                              # Primäre Zielvariable
D   = data.frame(y = y, f1 = f1, f2 = f2)                   # Dataframe
M   = lm(y ~ f1 + f2 + f1:f2, D)                            # Modellevaluation
X   = model.matrix(M)                                       # Designmatrix
```

```
   (Intercept) f12 f22 f12:f22
1            1   0   0       0
2            1   0   0       0
3            1   0   0       0
4            1   0   1       0
5            1   0   1       0
6            1   0   1       0
7            1   1   0       0
8            1   1   0       0
9            1   1   0       0
10           1   1   1       1
11           1   1   1       1
12           1   1   1       1
attr(,"assign")
[1] 0 1 2 3
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"

attr(,"contrasts")$f2
[1] "contr.treatment"
```

Einfache Lineare Regression

```r
n   = 12                                    # Anzahl Datenpunkte
x1  = 1:n                                   # Regressor
y   = rnorm(n)                              # Primäre Zielvariable
D   = data.frame(y = y, x1 = x1)            # Dataframe
M   = lm(y ~ x1, D)                         # Modellevaluation
X   = model.matrix(M)                       # Designmatrix
```

```
   (Intercept) x1
1            1  1
2            1  2
3            1  3
4            1  4
5            1  5
6            1  6
7            1  7
8            1  8
9            1  9
10           1 10
11           1 11
12           1 12
attr(,"assign")
[1] 0 1
```

Einfache Lineare Regression mit expliziter Interzeptdefinition

```r
n   = 12                                    # Anzahl Datenpunkte
x1  = 1:n                                   # Regressor
y   = rnorm(n)                              # Primäre Zielvariable
D   = data.frame(y = y, x1 = x1)            # Dataframe
M   = lm(y ~ 1 + x1, D)                     # Modellevaluation
X   = model.matrix(M)                       # Designmatrix
```

```
   (Intercept) x1
1            1  1
2            1  2
3            1  3
4            1  4
5            1  5
6            1  6
7            1  7
8            1  8
9            1  9
10           1 10
11           1 11
12           1 12
attr(,"assign")
[1] 0 1
```

Allgemeines lineares Modell

Einfache Lineare Regression mit Verzicht auf Interzeptdefinition

```r
n   = 12                                    # Anzahl Datenpunkte
x1  = 1:n                                   # Regressor
y   = rnorm(n)                              # Primäre Zielvariable
D   = data.frame(y = y, x1 = x1)            # Dataframe
M   = lm(y ~ 0 + x1, D)                     # Modellevaluation
X   = model.matrix(M)                       # Designmatrix
```

```
   x1
1   1
2   2
3   3
4   4
5   5
6   6
7   7
8   8
9   9
10 10
11 11
12 12
attr(,"assign")
[1] 1
```

Multiple Regression mit zwei Regressoren und expliziter Interzeptdefinition

```r
n   = 12                                    # Anzahl Datenpunkte
x1  = 1:n                                   # Regressor 1
x2  = (1:n)+5                               # Regressor 2
y   = rnorm(n)                              # Primäre Zielvariable
D   = data.frame(y = y, x1 = x1, x2 = x2)   # Dataframe
M   = lm(y ~ 1 + x1 + x2, D)                # Modellevaluation
X   = model.matrix(M)                       # Designmatrix
```

```
   (Intercept) x1 x2
1            1  1  6
2            1  2  7
3            1  3  8
4            1  4  9
5            1  5 10
6            1  6 11
7            1  7 12
8            1  8 13
9            1  9 14
10           1 10 15
11           1 11 16
12           1 12 17
attr(,"assign")
[1] 0 1 2
```

Allgemeines lineares Modell

Additives einfaktorielles ANCOVA Design mit einer Kovariate

```r
n   = 12                                          # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2))       # Faktorlevel der Datenpunkte
x1  = 1:n                                         # Kovariatenwerte der Datenpunkte
y   = rnorm(n)                                    # Primäre Zielvariable
D   = data.frame(y = y, f1 = f1, x1 = x1)         # Dataframe
M   = lm(y ~ f1 + x1, D)                          # Modellevaluation
X   = model.matrix(M)                             # Designmatrix
```

```
   (Intercept) f12 x1
1            1   0  1
2            1   0  2
3            1   0  3
4            1   0  4
5            1   0  5
6            1   0  6
7            1   1  7
8            1   1  8
9            1   1  9
10           1   1 10
11           1   1 11
12           1   1 12
attr(,"assign")
[1] 0 1 2
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"
```

Einfaktorielles ANCOVA Design mit einer Kovariate und Interaktion

```r
n   = 12                                          # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2))       # Faktorlevel der Datenpunkte
x1  = 1:n                                         # Kovariatenwerte der Datenpunkte
y   = rnorm(n)                                    # Primäre Zielvariable
D   = data.frame(y = y, f1 = f1, x1 = x1)         # Dataframe
M   = lm(y ~ f1 + x1 + f1:x1, D)                  # Modellevaluation
X   = model.matrix(M)                             # Designmatrix
```

```
   (Intercept) f12 x1 f12:x1
1            1   0  1      0
2            1   0  2      0
3            1   0  3      0
4            1   0  4      0
5            1   0  5      0
6            1   0  6      0
7            1   1  7      7
8            1   1  8      8
9            1   1  9      9
10           1   1 10     10
11           1   1 11     11
12           1   1 12     12
attr(,"assign")
[1] 0 1 2 3
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"
```

Allgemeines lineares Modell

Parallelgruppendesign

Einfache lineare Regression

Kovarianzanalyse

Selbstkontrollfragen

Parallelgruppendesign

= Parallelgruppendesign mit Post-Messung

= Zweistichproben-T-Test-Design

= Einfaktorielle Varianzanalyse mit zwei Faktorleveln

Designannahmen

- Zwei Gruppen/Stichproben randomisierter experimenteller Einheiten.
- Üblicherweise eine Kontrollgruppe und eine Treatmentgruppe
- Einmalige Messung der Outcomevariable an jeder experimentellen Einheit

Modellannahmen

- Gruppenspezifische Normalverteilungen $N\left(\beta_0, \sigma^2\right)$ und $N\left(\beta_0 + \beta_1, \sigma^2\right)$
- β_0, β_1 und σ^2 unbekannt
- Annahme eines identischen Varianzparameters σ^2 für beide Gruppen
- Ziel ist ein inferentieller Vergleich von β_1 mit 0

Parallelgruppendesign

Anwendungsbeispiel Klinische Psychologie

Beeinflusst die Selbstoffenbarung von Therapeut:innen die Bereitschaft zur Therapieaufnahme?

Randomisierte Zuweisung von $n = 16$ Proband:innen zu zwei Gruppen

- Kontrollgruppe: Video ohne Selbstoffenbarung durch Therapeut:in
- Treatmentgruppe: Video mit Selbstoffenbarung durch Therapeut:in

$\Rightarrow$ Messung der Bereitschaft zur Therapieaufnahme

Anwendungsbeispiel Umweltpsychologie

Beeinflusst die Exposition gegenüber einer spezifischen Umgebung das unmittelbare Stresserleben?

Randomisierte Zuweisung von $n = 16$ Proband:innen zu zwei Gruppen

- Kontrollgruppe: Video mit natürlicher Umgebung
- Treatmentgruppe: Video mit urbaner Umgebung

$\Rightarrow$ Messung des subjektiven Stresserlebens

Parallelgruppendesign

Strukturelle Modellform

Für $i = 1, \ldots, n$ Proband:innen seien y_i die Daten.

Dann hat das Parallelgruppendesign-Modell die strukturelle Modellform

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \tag{40}$$

mit

- $x_i := 0$ für Proband:in i in Kontrollgruppe
- $x_i := 1$ für Proband:in i in Treatmentgruppe
- $\varepsilon_i \sim N(0, \sigma^2)$ u.i.v.

Parameterbedeutungen

β_0	Erwartungswert der Kontrollgruppendaten
β_1	Erwartungswertunterschied zwischen Kontrollgruppe- und Treatmentgruppe
σ^2	Datenvariabilität

Designmatrixform für $n_0 = n_1 = 8$ Proband:innen in Kontrollgruppe bzw. Treatmentgruppe

$$y = X\beta + \varepsilon \Leftrightarrow \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \\ y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{15} \\ y_{16} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \\ \varepsilon_7 \\ \varepsilon_8 \\ \varepsilon_9 \\ \varepsilon_{10} \\ \varepsilon_{11} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{14} \\ \varepsilon_{15} \\ \varepsilon_{16} \end{pmatrix} \tag{41}$$

mit

$$\varepsilon_i \sim N(0, \sigma^2) \text{ u.i.v. für } i = 1, ..., 16 \Leftrightarrow \varepsilon \sim N(0_{16}, \sigma^2 I_{16}) \tag{42}$$

Datengeneration

```r
library(MASS)                                          # multivariate Normalverteilung
set.seed(0)                                            # Zufallszahlengeneratorzustand
p       = 2                                            # Anzahl Gruppen
n_0     = 8                                            # Proband:innen in Kontrollgruppe
n_1     = 8                                            # Proband:innen in Treatmentgruppe
n       = n_0 + n_1                                    # Gesamtanzahl an Proband:innen
X       = cbind(1, c(rep(0, n_0), rep(1, n_1)))        # Treatment-Control-Designmatrix
beta    = matrix(c(5,2), nrow = p)                     # Betaparameter
s_eps   = 1                                            # Varianzparameter
eps     = mvrnorm(1, rep(0,n), s_eps*diag(n))          # Fehlervektor
y       = X %*% beta + eps                             # Primäre Zielvariable
TRM     = c(rep(1, n_0), rep(2, n_1))                  # Treatmentfaktor
D       = data.frame(TRM = TRM, Y =  y)                # Dataframe
write.csv(D, "2-Daten/pgd.csv", row.names = FALSE)     # Speichern
```

Datensatz

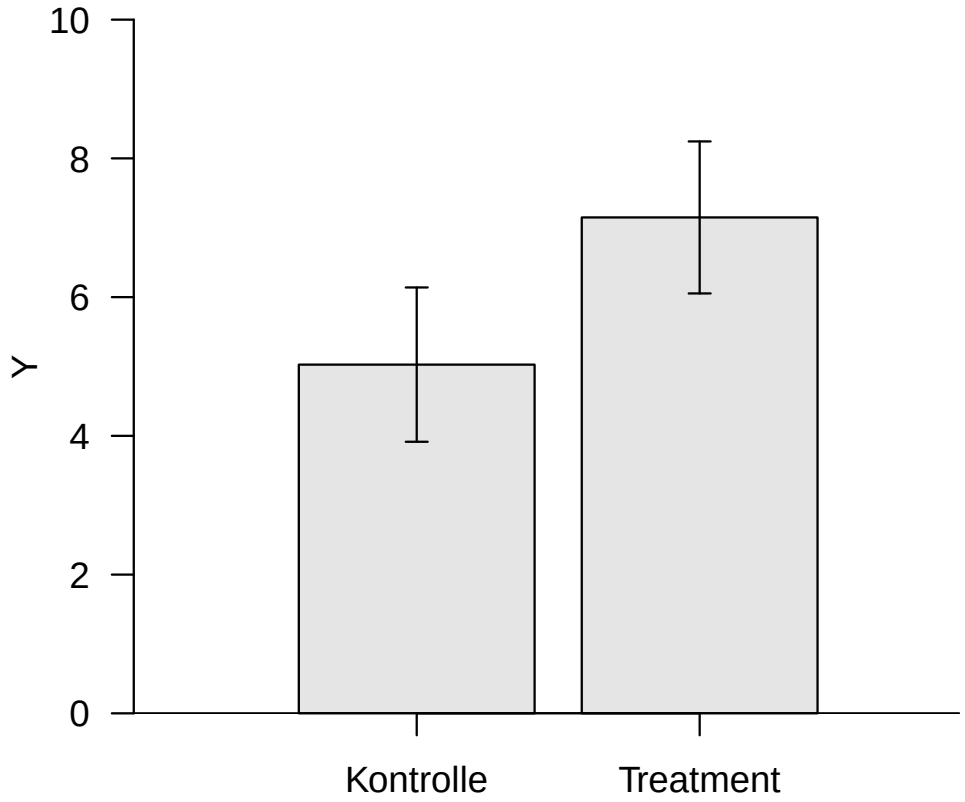
TRM	Y
1	4.6
1	4.7
1	4.7
1	3.9
1	4.2
1	5.8
1	7.4
1	5.0
2	6.7
2	6.1
2	5.5
2	7.4
2	8.3
2	8.3
2	6.7
2	8.3

Deskriptivstatistik

```r
library(dplyr)                                                  # dplyr für einfache Datengruppierung
D   = read.csv("2-Daten/pgd.csv")                               # Dateneinlesen
DS  = D %>% group_by(TRM) %>%  summarise(av = mean(Y, na.rm = TRUE), # Gruppenmittelwerte
                                         sd = sd(Y, na.rm = TRUE),   # Gruppenstandardabweichungen
                                         .groups = "drop")           # Gruppierungsaufhebung
print(DS)
```

```
# A tibble: 2 x 3
    TRM    av    sd
  <int> <dbl> <dbl>
1     1  5.03  1.11
2     2  7.15  1.10
```

Visualisierung



Modellschätzung und Modellevaluation

```
D           = read.csv("2-Daten/pgd.csv")                          # Dateneinlesen
n_g         = length(unique(D$TRM))                                # Anzahl Gruppen
n_0         = sum(D$TRM == 1)                                      # Proband:innen in Kontrollgruppe
n_1         = sum(D$TRM == 2)                                      # Proband:innen in Treatmentgruppe
n           = n_0 + n_1                                            # Gesamtanzahl an Proband:innen
p           = 2                                                    # Anzahl Betaparameter
X           = kronecker(matrix(c(1,1,0,1), ncol = 2), rep(1,n_0)) # Designmatrix bei n_0 = n_1
y           = D$Y                                                  # Primäre Zielvariable
beta_hat    = solve(t(X) %*% X) %*% t(X) %*% y                     # Betaparameterschätzer
eps_hat     = y - X %*% beta_hat                                   # Prädiktionsfehler
sigsqr_hat  = (t(eps_hat) %*% eps_hat)/(n-p)                       # Varianzparameterschätzer
delta       = 0.95                                                 # Konfidenzbedingung
t_delta     = qt((1+delta)/2,n-p)                                  # \Psi^{-1}((1+\delta)/2,n-p)
lambda      = diag(solve(t(X) %*% X))                              # \lambda_j Werte
ses         = sqrt(sigsqr_hat)*sqrt(lambda)                        # Betaparameterstandardfehlerschätzer
tvals       = beta_hat/ses                                         # T Werte
kappa_u     = beta_hat - ses*t_delta                               # untere KI Grenze
kappa_o     = beta_hat + ses*t_delta                               # obere  KI Grenze
```

Modellschätzung und Modellevaluation

```
# Ergebnisausgabe
print(
data.frame(
"Estimate"   = beta_hat,
"Std.Error"  = ses,
"t.value"    = beta_hat/ses,
"p.value"    = 2*(1-pt(abs(tvals), n-p)),
"KI.u"       = kappa_u,
"KI.o"       = kappa_o,
row.names    = c("beta_0_hat", "beta_1_hat")),
digits = 4)
```

```
           Estimate Std.Error t.value   p.value  KI.u  KI.o
beta_0_hat    5.027    0.3903  12.879 3.754e-09 4.190 5.864
beta_1_hat    2.122    0.5520   3.844 1.788e-03 0.938 3.306
```

Modellschätzung und Modellevaluation mit `lm()`

```
D           = read.csv("2-Daten/pgd.csv", head = T)             # Dataframe
D$TRM       = as.factor(D$TRM)                                  # R Faktor Kodierung
M           = lm(Y ~ TRM, data = D)                             # ALM Schätzung
coefs       = summary(M)$coefficients                           # Betaparameterschätzer
ci          = confint(M, level  = 0.95)                         # Konfidenzintervalle
print(cbind(coefs, KI.u = ci[, 1], KI.o = ci[, 2]), digits = 4) # Ausgabe
```

```
            Estimate Std. Error t value  Pr(>|t|)  KI.u  KI.o
(Intercept)    5.027     0.3903  12.879 3.754e-09 4.190 5.864
TRM2           2.122     0.5520   3.844 1.788e-03 0.938 3.306
```

Dokumentation

Anwendungsbeispiel Klinische Psychologie

Die Bereitschaft zur Therapieaufnahme war in der Gruppe mit Selbstoffenbarung durch die Therapeut:in höher ($M = 7.15$, $SD = 1.10$) als in der Gruppe ohne Selbstoffenbarung ($M = 5.03$, $SD = 1.11$); die ALM-Analyse zeigte einen signifikanten positiven Effekt der Selbstoffenbarung, $b = 2.12$, $SE = 0.55$, 95%-KI [0.94, 3.31], $t(14) = 3.84$, $p = .002$.

Anwendungsbeispiel Umweltpsychologie

Das subjektive Stresserleben war in der Gruppe mit Exposition gegenüber einer urbanen Umgebung höher ($M = 7.15$, $SD = 1.10$) als in der Gruppe mit Exposition gegenüber einer natürlichen Umgebung ($M = 5.03$, $SD = 1.11$); die ALM-Analyse ergab einen signifikanten Unterschied zwischen den Bedingungen, $b = 2.12$, $SE = 0.55$, 95%-KI [0.94, 3.31], $t(14) = 3.84$, $p = .002$.

Allgemeines lineares Modell

Parallelgruppendesign

Einfache lineare Regression

Kovarianzanalyse

Selbstkontrollfragen

Einfache lineare Regression

Designannahmen

- Eine Gruppe experimenteller Einheiten
- Einmalige Messung einer UV und einer AV an jeder experimentellen Einheit
- Frage nach dem linear-affinen Zusammenhang univariater UV und AV

Modellannahmen

- Annahme einer linear-affinen funktionalen Abhängigkeit der Form

$$y = \beta_0 + \beta_1 x + \varepsilon \text{ mit } \varepsilon \sim N(0, \sigma^2) \tag{43}$$

- β_0: Schnittpunkt von Gerade und y-Achse (Interzeptparameter)
- β_1: y-Differenz pro x-Einheitsdifferenz (Steigungsparameter)
- β_0, β_1 und σ^2 unbekannt.
- Ziel ist oft insbesondere ein inferentieller Vergleich von β_1 mit 0

Einfache lineare Regression

Anwendungsbeispiel Klinische Psychologie

Hängen allgemeines interpersonelles Vertrauen und die Bereitschaft zur Therapieaufnahme zusammen?

$n = 16$ Proband:innen wird ein Video einer Therapiesitzung gezeigt

$\Rightarrow$ Messung des allgemeinen interpersonellen Vertrauens (UV)

$\Rightarrow$ Messung der Bereitschaft zur Therapieaufnahme (AV)

Anwendungsbeispiel Umweltpsychologie

Hängen generelle Naturverbundenheit und subjektives Stresserleben zusammen?

$n = 16$ Proband:innen werden in einer urbanen Umgebung befragt

$\Rightarrow$ Messung der generellen Naturverbundenheit (UV)

$\Rightarrow$ Messung des subjektiven Stresserlebens (AV)

Einfache lineare Regression

Strukturelle Modellform

Für $i = 1, \ldots, n$ Proband:innen seien y_i die Daten der AV und x_i die Daten der UV.

Dann hat das Modell der einfachen linearen Regression die strukturelle Modellform

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \tag{44}$$

mit

- $\varepsilon_i \sim N(0, \sigma^2)$ u.i.v.

Parameterbedeutungen

β_0	Erwartungswert der AV bei $x_i = 0$
β_1	Erwartungswertunterschied der AV pro x_i-Einheitsdifferenz
σ^2	Datenvariabilität

Designmatrixform für $n = 16$ Proband:innen

$$y = X\beta + \varepsilon \Leftrightarrow \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \\ y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{15} \\ y_{16} \end{pmatrix} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ 1 & x_4 \\ 1 & x_5 \\ 1 & x_6 \\ 1 & x_7 \\ 1 & x_8 \\ 1 & x_9 \\ 1 & x_{10} \\ 1 & x_{11} \\ 1 & x_{12} \\ 1 & x_{13} \\ 1 & x_{14} \\ 1 & x_{15} \\ 1 & x_{16} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \\ \varepsilon_7 \\ \varepsilon_8 \\ \varepsilon_9 \\ \varepsilon_{10} \\ \varepsilon_{11} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{14} \\ \varepsilon_{15} \\ \varepsilon_{16} \end{pmatrix} \tag{45}$$

mit

$$\varepsilon_i \sim N(0, \sigma^2) \text{ u.i.v. für } i = 1, ..., 16 \Leftrightarrow \varepsilon \sim N(0_{16}, \sigma^2 I_{16}) \tag{46}$$

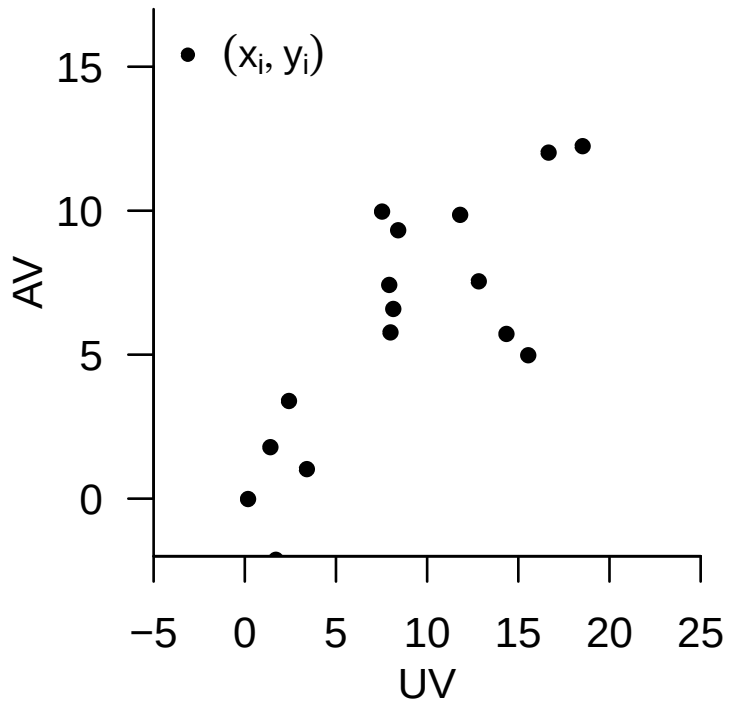
Datengeneration

```r
library(MASS)                                       # multivariate Normalverteilung
set.seed(0)                                         # Zufallszahlengeneratorzustand
n       = 16                                        # Gesamtanzahl an Proband:innen
x       = mvrnorm(1, 1:n, 4*diag(n))                # Zufälliger Prädiktor (keine Modellannahme!)
X       = matrix(c(rep(1,n), x), ncol = 2)          # Designmatrix
beta    = matrix(c(2,.5), nrow = 2)                 # Betaparameter
s_eps   = 15                                        # Varianzparameter
eps     = mvrnorm(1, rep(0,n), s_eps*diag(n))       # Fehlervektor
y       = X %*% beta + eps                          # Primäre Zielvariable
D       = data.frame(UV = X[,2], AV = y)            # Dataframe
write.csv(D, "2-Daten/elr.csv", row.names = FALSE)  # Speichern
```

Datensatz

UV	AV
0.2	0.0
1.4	1.8
2.4	3.4
1.7	-2.1
3.4	1.0
7.5	10.0
11.8	9.9
8.0	5.8
8.4	9.3
8.1	6.6
7.9	7.4
12.8	7.5
15.5	5.0
16.7	12.0
14.3	5.7
18.5	12.2

Visualisierung



Modellschätzung und Modellevaluation

```r
D           = read.csv("2-Daten/elr.csv")                  # Dateneinlesen
y           = D$AV                                         # Primäre Zielvariable
n           = nrow(D)                                      # Gesamtanzahl an Proband:innen
X           = matrix(c(rep(1,n), D$UV), ncol = 2)          # Designmatrix
p           = ncol(X)                                      # Anzahl Betaparameter
beta_hat    = solve(t(X) %*% X) %*% t(X) %*% y             # Betaparameterschätzer
eps_hat     = y - X %*% beta_hat                           # Prädiktionsfehler
sigsqr_hat  = (t(eps_hat) %*% eps_hat)/(n-p)               # Varianzparameterschätzer
delta       = 0.95                                         # Konfidenzbedingung
t_delta     = qt((1+delta)/2,n-p)                          # \Psi^{-1}((1+\delta)/2,n-p)
lambda      = diag(solve(t(X) %*% X))                      # \lambda_j Werte
ses         = sqrt(sigsqr_hat)*sqrt(lambda)                # Betaparameterstandardfehlerschätzer
tvals       = beta_hat/ses                                 # T Werte
kappa_u     = beta_hat - ses*t_delta                       # untere KI Grenze
kappa_o     = beta_hat + ses*t_delta                       # obere  KI Grenze
```

Modellschätzung und Modellevaluation

```
# Ergebnisausgabe
print(
data.frame(
"Estimate"   = beta_hat,
"Std.Error"  = ses,
"t.value"    = beta_hat/ses,
"p.value"    = 2*(1-pt(abs(tvals), n-p)),
"KI.u"       = kappa_u,
"KI.o"       = kappa_o,
row.names    = c("beta_0_hat", "beta_1_hat")),
digits = 4)
```

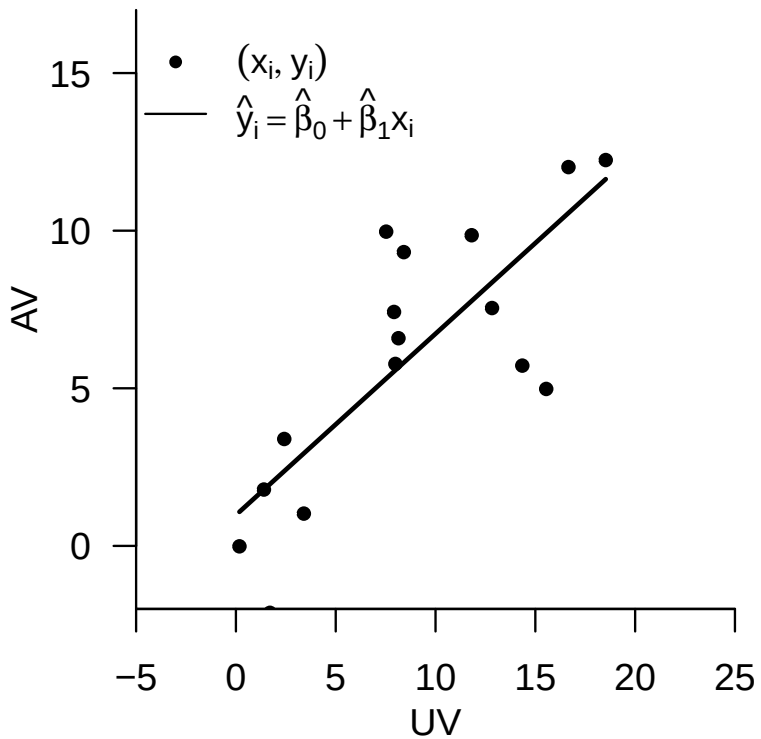
```
           Estimate Std.Error t.value   p.value    KI.u   KI.o
beta_0_hat   0.9786    1.2562   0.779 0.4489300 -1.7156 3.6728
beta_1_hat   0.5752    0.1213   4.740 0.0003162  0.3149 0.8354
```

Modellschätzung und Modellevaluation mit `lm()`

```r
D           = read.csv("2-Daten/elr.csv", head = T)             # Dataframe
M           = lm(AV ~ UV, data = D)                             # ALM Schätzung
coefs       = summary(M)$coefficients                           # Betaparameterschätzer
ci          = confint(M, level  = 0.95)                         # Konfidenzintervalle
print(cbind(coefs, KI.u = ci[, 1], KI.o = ci[, 2]), digits = 4) # Ausgabe
```

```
            Estimate Std. Error t value  Pr(>|t|)    KI.u   KI.o
(Intercept)   0.9786     1.2562   0.779 0.4489300 -1.7156 3.6728
UV            0.5752     0.1213   4.740 0.0003162  0.3149 0.8354
```

Visualisierung



Dokumentation

Anwendungsbeispiel Klinische Psychologie

Das allgemeine interpersonelle Vertrauen sagte die Bereitschaft zur Therapieaufnahme positiv vorher; eine lineare Regressionsanalyse zeigte einen signifikanten positiven Regressionskoeffizienten, $b = 0.58$, $SE = 0.12$, 95%-KI [0.31, 0.84], $t(14) = 4.74$, $p < .001$.

Anwendungsbeispiel Umweltpsychologie

Die generelle Naturverbundenheit sagte das subjektive Stresserleben positiv vorher; eine lineare Regressionsanalyse zeigte einen signifikanten positiven Regressionskoeffizienten, $b = 0.58$, $SE = 0.12$, 95%-KI [0.31, 0.84], $t(14) = 4.74$, $p < .001$.

Allgemeines lineares Modell

Parallelgruppendesign

Einfache lineare Regression

Kovarianzanalyse

Selbstkontrollfragen

Kovarianzanalyse

Überblick

- Analysis of Covariance (ANCOVA)
- Kombination von Parallelgruppendesign und Regression
- Kontrolle von (unkontrollierten) Störvariablen bei Schätzung von Treatmenteffekten
- Gemessene Werte einer Störvariable werden als *Kovariate* in das Modell aufgenommen
- "Herausrechnen der Kovarianz von Störvariable und abhängiger Variable"
- Kategoriale und kontinuierliche Prädiktoren in einer Designmatrix
- *Adjustierte Gruppenmittelwerte* als modellbasiert-korrigierte Deskriptivstatistiken
- Adjustierte Gruppenmittelwerte aka *Estimated Marginal Means (EMMs)*

Kovarianzanalyse $\approx$ Additive Berücksichtigung von Kovariaten im ALM

Moderationsanalyse $\approx$ Interaktive Berücksichtigung von Kovariaten im ALM

Kovarianzanalyse

Anwendungsbeispiel Klinische Psychologie

Beeinflusst die Selbstoffenbarung von Therapeut:innen die Bereitschaft zur Therapieaufnahme?

Randomisierte Zuweisung von $n = 40$ Proband:innen zu zwei Gruppen

- Kontrollgruppe: Video ohne Selbstoffenbarung durch Therapeut:in
- Treatmentgruppe: Video mit Selbstoffenbarung durch Therapeut:in

$\Rightarrow$ Messung der Bereitschaft zur Therapieaufnahme als primäre Zielvariable

$\Rightarrow$ Messung des allgemeinen interpersonellen Vertrauens als Kovariate

Anwendungsbeispiel Umweltpsychologie

Beeinflusst die Exposition gegenüber einer spezifischen Umgebung das unmittelbare Stresserleben?

Randomisierte Zuweisung von $n = 40$ Proband:innen zu zwei Gruppen

- Kontrollgruppe: Video mit Naturumgebung
- Treatmentgruppe: Video mit urbaner Umgebung

$\Rightarrow$ Messung des subjektiven Stresserlebens als primäre Zielvariable

$\Rightarrow$ Messung der generellen Naturverbundenheit als Kovariate

Kovarianzanalysemodell für ein Parallelgruppendesign mit zwei Gruppen und einer Kovariate

In einem Kovarianzanalysemodell für ein Parallelgruppendesign mit zwei Gruppen und einer Kovariate modelliert man die Daten y_i von n Proband:innen

$$y_i \sim N(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}, \sigma^2) \tag{47}$$

basierend auf dem Treatmentindikator x_{i1} und den Kovariatenwerten x_{i2} für $i = 1, ..., n$. Der Treatmentindikator ist dabei als $x_{i1} = 0$ für Proband:innen in der Kontrollgruppe und $x_{i1} = 1$ für Proband:innen in der Treatmentgruppe kodiert.

Der Parameter β_0 modelliert den Erwartungswert der Zielvariable für die Kontrollgruppe bei einem Kovariatenwert von $x_{i2} = 0$. Der Parameter β_1 modelliert den Erwartungswertunterschied zwischen Treatment- und Kontrollgruppe bei gleichem Kovariatenwert. Der Parameter β_2 quantifiziert den Beitrag der Kovariate zum Erwartungswert der Zielvariable.

Man beachte, dass dieses Modell eine einfache lineare Regression für jede der beiden Gruppen definiert, bei der β_0 und $\beta_0 + \beta_1$ als gruppenspezifische Interzeptparameter interpretiert werden können und der Steigungsparameter β_2 für beide Gruppen identisch ist.

Strukturelle Modellform

Für $i = 1, \ldots, n$ Proband:innen seien y_i die Daten der Zielvariable und x_{i2} die Daten der Kovariate

Dann hat das Kovarianzanalysemodell für ein Parallelgruppendesign mit einer Kovariate die strukturelle Modellform

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \varepsilon_i \tag{48}$$

mit

- $x_{i1} = 0$ für Proband:in i in Kontrollgruppe,
- $x_{i1} = 1$ für Proband:in i in Treatmentgruppe,
- $\varepsilon_i \sim N(0, \sigma^2)$ u.i.v.

Parameterbedeutungen

β_0	Erwartungswert der AV bei $x_{i1} = 0$ und $x_{i2} = 0$
β_1	Erwartungswertunterschied der AV zwischen Treatment- und Kontrollgruppe
β_2	Erwartungswertunterschied der AV pro x_{i2}-Einheitsdifferenz
σ^2	Datenvariabilität

Designmatrixform für insgesamt $n = 16$ Proband:innen mit $n_0 = n_1 = 8$

$$y = X\beta + \varepsilon \Leftrightarrow \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \\ y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{15} \\ y_{16} \end{pmatrix} = \begin{pmatrix} 1 & 0 & x_{12} \\ 1 & 0 & x_{22} \\ 1 & 0 & x_{32} \\ 1 & 0 & x_{42} \\ 1 & 0 & x_{52} \\ 1 & 0 & x_{62} \\ 1 & 0 & x_{72} \\ 1 & 0 & x_{82} \\ 1 & 1 & x_{92} \\ 1 & 1 & x_{102} \\ 1 & 1 & x_{112} \\ 1 & 1 & x_{122} \\ 1 & 1 & x_{132} \\ 1 & 1 & x_{142} \\ 1 & 1 & x_{152} \\ 1 & 1 & x_{162} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \\ \varepsilon_7 \\ \varepsilon_8 \\ \varepsilon_9 \\ \varepsilon_{10} \\ \varepsilon_{11} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{14} \\ \varepsilon_{15} \\ \varepsilon_{16} \end{pmatrix} \tag{49}$$

mit

$$\varepsilon_i \sim N(0, \sigma^2) \text{ u.i.v. für } i = 1, ..., 16 \Leftrightarrow \varepsilon \sim N(0_{16}, \sigma^2 I_{16}) \tag{50}$$

Datengeneration

```r
library(MASS)                                                    # Multivariate Normalverteilung
set.seed(0)                                                      # Zufallszahlengeneratorzustand
p       = 2                                                      # Anzahl Gruppen
n_0     = 20                                                     # Proband:innen in Kontrollgruppe
n_1     = 20                                                     # Proband:innen in Treatmentgruppe
n       = n_0 + n_1                                              # Gesamtzahl Datenpunkte
x       = round(mvrnorm(1, 5*rep(1,n), 4*diag(n)), digits = 1)   # Kovariate
X       = cbind(1, c(rep(0, n_0), rep(1, n_1)), x)               # Designmatrix
beta    = matrix(c(10,5,-1), nrow = ncol(X))                     # Betaparameter
s_eps   = 4                                                      # Varianzparameter
eps     = mvrnorm(1, rep(0,n), s_eps*diag(n))                    # Fehlervektor
y       = X %*% beta + eps                                       # Primäre Zielvariable
TRM     = c(rep(1, n_0), rep(2, n_1))                            # Gruppenvariable
D       = data.frame(TRM = TRM, x = x, Y = y)                    # Dataframe
write.csv(D, "2-Daten/kva-1.csv", row.names = FALSE)             # Datenspeicherung
```

Datensatzauszug

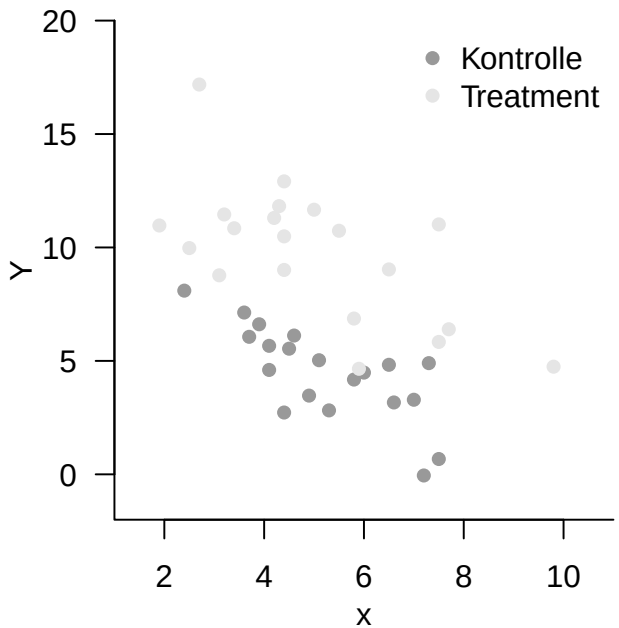
	TRM	x	Y
1	1	4.4	2.7
2	1	7.5	0.7
3	1	4.1	5.7
4	1	7.0	3.3
5	1	7.3	4.9
6	1	6.5	4.8
7	1	3.7	6.1
8	1	4.1	4.6
9	1	3.9	6.6
10	1	4.5	5.5
21	2	2.5	10.0
22	2	5.9	4.7
23	2	3.2	11.5
24	2	5.5	10.7
25	2	4.2	11.3
26	2	4.4	10.5
27	2	4.4	9.0
28	2	2.7	17.2
29	2	3.4	10.8
30	2	6.5	9.0

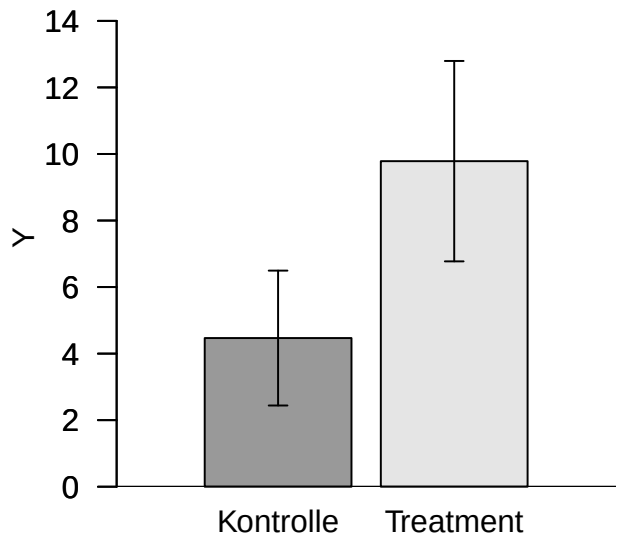
Deskriptivstatistik

```r
library(dplyr)                                                    # dplyr für einfache Datengruppierung
D   = read.csv("2-Daten/kva-1.csv")                               # Dateneinlesen
DS  = D %>% group_by(TRM) %>% summarise(avY = mean(Y, na.rm = TRUE),    # Gruppenmittelwerte Y
                                        sdY = sd(Y, na.rm = TRUE),      # Gruppenstandardabweichungen Y
                                        avx = mean(x, na.rm = TRUE),    # Gruppenmittelwerte x
                                        sdx = sd(x, na.rm = TRUE),      # Gruppenstandardabweichungen x
                                        .groups = "drop")               # Gruppierungsaufhebung
print(DS)
```

```
# A tibble: 2 x 5
    TRM   avY   sdY   avx   sdx
  <int> <dbl> <dbl> <dbl> <dbl>
1     1  4.47  2.03  5.22  1.45
2     2  9.78  3.01  4.99  2.05
```

Visualisierung





Theorem (Parameterschätzung im Kovarianzanalysemodell)

Gegeben sei das Kovarianzanalysemodell für ein Parallelgruppendesign mit einer Kovariate in Designmatrixform

$$y = X\beta + \varepsilon \text{ mit } y := \begin{pmatrix} y_{01} \\ \vdots \\ y_{0n_0} \\ y_{11} \\ \vdots \\ y_{1n_1} \end{pmatrix}, \quad X := \begin{pmatrix} 1 & 0 & x_{01} \\ \vdots & \vdots & \vdots \\ 1 & 0 & x_{0n_0} \\ 1 & 1 & x_{11} \\ \vdots & \vdots & \vdots \\ 1 & 1 & x_{1n_1} \end{pmatrix} \in \mathbb{R}^{n\times 3}, \quad \beta := \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} \in \mathbb{R}^3 \text{ und } \sigma^2 > 0. \tag{51}$$

Dann ergibt sich mit

$$\bar{x}_0 := \frac{1}{n_0}\sum_{j=1}^{n_0} x_{0j}, \quad \bar{x}_1 := \frac{1}{n_1}\sum_{j=1}^{n_1} x_{1j}, \quad \bar{y}_0 := \frac{1}{n_0}\sum_{j=1}^{n_0} y_{0j}, \quad \bar{y}_1 := \frac{1}{n_1}\sum_{j=1}^{n_1} y_{1j} \tag{52}$$

für den Betaparameterschätzer

$$\hat{\beta} = \begin{pmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \end{pmatrix} = \begin{pmatrix} \bar{y}_0 - \hat{\beta}_2\bar{x}_0 \\ (\bar{y}_1 - \bar{y}_0) - \hat{\beta}_2(\bar{x}_1 - \bar{x}_0) \\ \frac{\sum_{j=1}^{n_0}(x_{0j}-\bar{x}_0)(y_{0j}-\bar{y}_0)+\sum_{j=1}^{n_1}(x_{1j}-\bar{x}_1)(y_{1j}-\bar{y}_1)}{\sum_{j=1}^{n_0}(x_{0j}-\bar{x}_0)^2+\sum_{j=1}^{n_1}(x_{1j}-\bar{x}_1)^2}. \end{pmatrix} \tag{53}$$

Bemerkung

- Man beachte, dass die Gruppenzugehörigkeit der Daten- und Kovariatenwerte hier explizit notiert ist.

Beweis

Wir halten zunächst fest, dass

$$X^T = \begin{pmatrix} 1 & \cdots & 1 & 1 & \cdots & 1 \\ 0 & \cdots & 0 & 1 & \cdots & 1 \\ x_{01} & \cdots & x_{0n_0} & x_{11} & \cdots & x_{1n_1} \end{pmatrix}. \tag{54}$$

Damit ergibt sich

$$X^T X = \begin{pmatrix} n_0 + n_1 & n_1 & \sum_{j=1}^{n_0} x_{0j} + \sum_{j=1}^{n_1} x_{1j} \\ n_1 & n_1 & \sum_{j=1}^{n_1} x_{1j} \\ \sum_{j=1}^{n_0} x_{0j} + \sum_{j=1}^{n_1} x_{1j} & \sum_{j=1}^{n_1} x_{1j} & \sum_{j=1}^{n_0} x_{0j}^2 + \sum_{j=1}^{n_1} x_{1j}^2 \end{pmatrix} \tag{55}$$

und

$$X^T y = \begin{pmatrix} \sum_{j=1}^{n_0} y_{0j} + \sum_{j=1}^{n_1} y_{1j} \\ \sum_{j=1}^{n_1} y_{1j} \\ \sum_{j=1}^{n_0} x_{0j} y_{0j} + \sum_{j=1}^{n_1} x_{1j} y_{1j} \end{pmatrix}. \tag{56}$$

Beweis (fortgeführt)

Aus den ersten beiden Komponenten der Normalgleichungen

$$X^T X\hat{\beta} = X^T y. \tag{57}$$

folgt damit

$$(n_0 + n_1)\hat{\beta}_0 + n_1\hat{\beta}_1 + \hat{\beta}_2 \left(\sum_{j=1}^{n_0} x_{0j} + \sum_{j=1}^{n_1} x_{1j} \right) = \sum_{j=1}^{n_0} y_{0j} + \sum_{j=1}^{n_1} y_{1j}, \tag{58}$$

$$n_1\hat{\beta}_0 + n_1\hat{\beta}_1 + \hat{\beta}_2 \sum_{j=1}^{n_1} x_{1j} = \sum_{j=1}^{n_1} y_{1j}. \tag{59}$$

Subtraktion der zweiten von der ersten Gleichung liefert

$$n_0\hat{\beta}_0 + \hat{\beta}_2 \sum_{j=1}^{n_0} x_{0j} = \sum_{j=1}^{n_0} y_{0j}, \tag{60}$$

also

$$\hat{\beta}_0 = \bar{y}_0 - \hat{\beta}_2\bar{x}_0. \tag{61}$$

Setzt man dies in die zweite Gleichung ein, so erhält man

$$n_1(\bar{y}_0 - \hat{\beta}_2\bar{x}_0) + n_1\hat{\beta}_1 + \hat{\beta}_2 \sum_{j=1}^{n_1} x_{1j} = \sum_{j=1}^{n_1} y_{1j}. \tag{62}$$

Beweis (fortgeführt)

Wegen $\sum_{j=1}^{n_1} x_{1j} = n_1 \bar{x}_1$ und $\sum_{j=1}^{n_1} y_{1j} = n_1 \bar{y}_1$ folgt

$$\bar{y}_0 - \hat{\beta}_2 \bar{x}_0 + \hat{\beta}_1 + \hat{\beta}_2 \bar{x}_1 = \bar{y}_1, \tag{63}$$

also

$$\hat{\beta}_1 = (\bar{y}_1 - \bar{y}_0) - \hat{\beta}_2 (\bar{x}_1 - \bar{x}_0). \tag{64}$$

Zur Bestimmung von $\hat{\beta}_2$ betrachten wir die dritte Normalgleichung

$$\hat{\beta}_0 \left(\sum_{j=1}^{n_0} x_{0j} + \sum_{j=1}^{n_1} x_{1j} \right) + \hat{\beta}_1 \sum_{j=1}^{n_1} x_{1j} + \hat{\beta}_2 \left(\sum_{j=1}^{n_0} x_{0j}^2 + \sum_{j=1}^{n_1} x_{1j}^2 \right) = \sum_{j=1}^{n_0} x_{0j} y_{0j} + \sum_{j=1}^{n_1} x_{1j} y_{1j}. \tag{65}$$

Einsetzen von

$$\hat{\beta}_0 = \bar{y}_0 - \hat{\beta}_2 \bar{x}_0 \quad \text{und} \quad \hat{\beta}_1 = (\bar{y}_1 - \bar{y}_0) - \hat{\beta}_2 (\bar{x}_1 - \bar{x}_0) \tag{66}$$

liefert nach Ausmultiplizieren

$$(\bar{y}_0 - \hat{\beta}_2 \bar{x}_0) \left(\sum_{j=1}^{n_0} x_{0j} + \sum_{j=1}^{n_1} x_{1j} \right) + \big((\bar{y}_1 - \bar{y}_0) - \hat{\beta}_2 (\bar{x}_1 - \bar{x}_0) \big) \sum_{j=1}^{n_1} x_{1j} \tag{67}$$

$$+ \hat{\beta}_2 \left(\sum_{j=1}^{n_0} x_{0j}^2 + \sum_{j=1}^{n_1} x_{1j}^2 \right) = \sum_{j=1}^{n_0} x_{0j} y_{0j} + \sum_{j=1}^{n_1} x_{1j} y_{1j}. \tag{68}$$

Beweis (fortgeführt)

Unter Verwendung von

$$\sum_{j=1}^{n_0} x_{0j} = n_0\bar{x}_0, \quad \sum_{j=1}^{n_1} x_{1j} = n_1\bar{x}_1, \quad \sum_{j=1}^{n_0} y_{0j} = n_0\bar{y}_0, \quad \sum_{j=1}^{n_1} y_{1j} = n_1\bar{y}_1 \tag{69}$$

vereinfacht sich dies zu

$$n_0\bar{x}_0\bar{y}_0 + n_1\bar{x}_1\bar{y}_1 - \hat{\beta}_2\left(n_0\bar{x}_0^2 + n_1\bar{x}_1^2\right) + \hat{\beta}_2\left(\sum_{j=1}^{n_0} x_{0j}^2 + \sum_{j=1}^{n_1} x_{1j}^2\right) = \sum_{j=1}^{n_0} x_{0j}y_{0j} + \sum_{j=1}^{n_1} x_{1j}y_{1j}. \tag{70}$$

Also gilt

$$\hat{\beta}_2 = \frac{\sum_{j=1}^{n_0} x_{0j}y_{0j} - n_0\bar{x}_0\bar{y}_0 + \sum_{j=1}^{n_1} x_{1j}y_{1j} - n_1\bar{x}_1\bar{y}_1}{\sum_{j=1}^{n_0} x_{0j}^2 - n_0\bar{x}_0^2 + \sum_{j=1}^{n_1} x_{1j}^2 - n_1\bar{x}_1^2}. \tag{71}$$

Beweis (fortgeführt)

Mit der Identität

$$\sum_{j=1}^{n_0}(x_{0j}-\bar{x}_0)(y_{0j}-\bar{y}_0)=\sum_{j=1}^{n_0}x_{0j}y_{0j}-n_0\bar{x}_0\bar{y}_0 \tag{72}$$

sowie

$$\sum_{j=1}^{n_1}(x_{1j}-\bar{x}_1)(y_{1j}-\bar{y}_1)=\sum_{j=1}^{n_1}x_{1j}y_{1j}-n_1\bar{x}_1\bar{y}_1 \tag{73}$$

und

$$\sum_{j=1}^{n_0}(x_{0j}-\bar{x}_0)^2=\sum_{j=1}^{n_0}x_{0j}^2-n_0\bar{x}_0^2,\quad \sum_{j=1}^{n_1}(x_{1j}-\bar{x}_1)^2=\sum_{j=1}^{n_1}x_{1j}^2-n_1\bar{x}_1^2 \tag{74}$$

folgt schließlich

$$\hat{\beta}_2=\frac{\sum_{j=1}^{n_0}(x_{0j}-\bar{x}_0)(y_{0j}-\bar{y}_0)+\sum_{j=1}^{n_1}(x_{1j}-\bar{x}_1)(y_{1j}-\bar{y}_1)}{\sum_{j=1}^{n_0}(x_{0j}-\bar{x}_0)^2+\sum_{j=1}^{n_1}(x_{1j}-\bar{x}_1)^2}. \tag{75}$$

Theorem (Darstellung der Gruppenmittel im Kovarianzanalysemodell)

Gegeben sei das Kovarianzanalysemodell für ein Parallelgruppendesign mit einer Kovariate und sein Betaparameterschätzer $\hat{\beta}$. Dann gilt für die Gruppenmittelwerte, dass

$$\bar{y}_0 = \hat{\beta}_0 + \hat{\beta}_2 \bar{x}_0 \text{ und } \bar{y}_1 = \hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 \bar{x}_1, \tag{76}$$

und für die Gruppenmittelwertsdifferenz ergibt sich

$$\bar{y}_1 - \bar{y}_0 = \hat{\beta}_1 + \hat{\beta}_2 (\bar{x}_1 - \bar{x}_0). \tag{77}$$

Bemerkungen

- Das Theorem ist für die Idee der adjustierten Gruppenmittelwerte von zentraler Bedeutung
- Das Theorem zeigt, dass die Differenz der Gruppenmittelwerte $\bar{y}_1 - \bar{y}_0$ als Summe
  - eines gruppenspezifischen Effekts $\hat{\beta}_1$ und
  - eines Effekts der Kovariate $\hat{\beta}_2(\bar{x}_1 - \bar{x}_0)$

  dargestellt werden kann.
- Der Term $\hat{\beta}_2(\bar{x}_1 - \bar{x}_0)$ beschreibt dabei den Beitrag der Kovariatenmittelwertsdifferenz zur unadjustierten Gruppenmittelwertsdifferenz.

Beweis

Anhand des Theorems zur Parameterschätzung im Kovarianzanalysemodell gilt

$$\hat{\beta}_0 = \bar{y}_0 - \hat{\beta}_2 \bar{x}_0 \text{ und } \hat{\beta}_1 = (\bar{y}_1 - \bar{y}_0) - \hat{\beta}_2(\bar{x}_1 - \bar{x}_0). \tag{78}$$

Durch Umstellen der ersten Gleichung erhält man

$$\bar{y}_0 = \hat{\beta}_0 + \hat{\beta}_2 \bar{x}_0. \tag{79}$$

Ferner folgt aus der zweiten Gleichung

$$\bar{y}_1 - \bar{y}_0 = \hat{\beta}_1 + \hat{\beta}_2(\bar{x}_1 - \bar{x}_0), \tag{80}$$

also

$$\bar{y}_1 = \bar{y}_0 + \hat{\beta}_1 + \hat{\beta}_2(\bar{x}_1 - \bar{x}_0) = \hat{\beta}_0 + \hat{\beta}_2 \bar{x}_0 + \hat{\beta}_1 + \hat{\beta}_2(\bar{x}_1 - \bar{x}_0) = \hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 \bar{x}_1. \tag{81}$$

Modellformulierung zur Analyse des Beispieldatensatzes

M1 | Parallelgruppendesign ohne Berücksichtigung der Kovariate

$$y_i \sim N(\mu_i, \sigma^2) \text{ mit } \mu_i := \beta_0 + \beta_1 x_{i1} \text{ und } \sigma^2 > 0$$

$$\Leftrightarrow y \sim N(X\beta, \sigma^2 I_n) \text{ mit } X := \begin{pmatrix} 1 & 0 \\ \vdots & \vdots \\ 1 & 0 \\ 1 & 1 \\ \vdots & \vdots \\ 1 & 1 \end{pmatrix}, \beta := \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} \text{ und } \sigma^2 > 0 \quad (82)$$

M2 | Kovarianzanalysedesign mit Berücksichtigung der Kovariate

$$y_i \sim N(\mu_i, \sigma^2) \text{ mit } \mu_i := \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} \text{ und } \sigma^2 > 0$$

$$\Leftrightarrow y \sim N(X\beta, \sigma^2 I_n) \text{ mit } X := \begin{pmatrix} 1 & 0 & x_1 \\ \vdots & \vdots & \vdots \\ 1 & 0 & x_{n_0} \\ 1 & 1 & x_{n_0+1} \\ \vdots & \vdots & \vdots \\ 1 & 1 & x_{n_0+n_1} \end{pmatrix}, \beta := \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} \text{ und } \sigma^2 > 0 \quad (83)$$

Modellschätzung und Modellevaluation

```
D          = read.csv("2-Daten/kva-1.csv")                                  # Datensatz
n_0        = sum(D$TRM == 1)                                                # Proband:innen in Kontrollgruppe
n_1        = sum(D$TRM == 2)                                                # Proband:innen in Treatmentgruppe
y          = D$Y                                                            # Primäre Zielvariable
n          = length(y)                                                      # Gesamtanzahl Datenpunkte
XS         = list()                                                         # Modell 1 und 2 Liste
CS         = list()                                                         # Kontrastgewichtsvektorenliste
XS[[1]]    = matrix(c(rep(1,n_0),rep(1,n_1),                                # \beta_0    Regressor
                      rep(0,n_0),rep(1,n_1)), nrow = n)                     # \beta_1 Regressor
XS[[2]]    = matrix(c(rep(1,n_0),rep(1,n_1),                                # \beta_0    Regressor
                      rep(0,n_0),rep(1,n_1),                                # \beta_1 Regressor
                      D$x),nrow = n)                                        # x        Regressor
CS[[1]]    = matrix(c(0,1)   , nrow = 2)                                    # Kontrastgewichtsvektor \beta_1
CS[[2]]    = matrix(c(0,1,0), nrow = 3)                                     # Kontrastgewichtsvektor \beta_1
B          = list()                                                         # Betaparameterschätzerliste
S          = rep(NaN,2)                                                     # Varianzparameterschätzervektor
T          = rep(NaN,2)                                                     # T-Statistikvektor
for(i in 1:2){                                                              # Modell 1 und 2 Iterationen
  X           = XS[[i]]                                                     # Designmatrix
  p           = ncol(X)                                                     # Anzahl Betaparameter
  beta_hat    = solve(t(X) %*% X) %*% t(X) %*% y                            # Betaparameterschätzer vollständiges Modell
  eps_hat     = y - X %*% beta_hat                                          # Residuenvektor
  sigsqr_hat  = (t(eps_hat) %*% eps_hat)/(n-p)                              # Varianzparameterschätzer
  c           = CS[[i]]                                                     # Kontrastgewichtsvektor
  t_num       = t(c) %*% beta_hat                                           # Zähler der T-Teststatistik
  t_den       = sqrt(sigsqr_hat %*% t(c) %*% solve(t(X) %*% X) %*% c)       # Nenner der T-Teststatistik
  t           = t_num/t_den                                                 # Wert der T-Teststatistik
  B[[i]]      = beta_hat                                                    # Betaparameterschätzer
  S[i]        = sigsqr_hat                                                  # Varianzparameterschätzer
  T[i]        = t                                                           # T-Statistik
}
```

Ergebnisse

Wahre, aber unbekannte Werte: $\beta = (10, 5, -1)^T$ und $\sigma^2 = 4$.

```
M1 | Betaparameterschätzer    :  4.47 5.32
M2 | Betaparameterschätzer    :  9.63 5.08 -0.99
M1 | Varianzparameterschätzer :  6.59
M2 | Varianzparameterschätzer :  3.62
M1 | T-Statistik TRM 2        :  6.55
M2 | T-Statistik TRM 2        :  8.42
```

Im Kovarianzanalysemodell M2 ist der Varianzparameterschätzer kleiner als in M1.

$\Rightarrow$ Der Kovariatenregressor erklärt Datenvarianz.

Die Betaparameterschätzer für den Effekt von TRM $= 2$ sind in M1 und M2 ähnlich.

$\Rightarrow$ Das Signal-zu-Rauschen Verhältnis für den Gruppeneffekt ist im Kovarianzanalysemodell M2 höher.

Modellschätzung und Modellevaluation mit `lm()`

```
D           = read.csv("2-Daten/kva-1.csv")                    # Dataframe
D$TRM       = as.factor(D$TRM)                                 # R Faktor Kodierung
M1          = lm(Y ~ TRM, data = D)                            # ALM Schätzung Modell 1
M2          = lm(Y ~ TRM + x, data = D)                        # ALM Schätzung Modell 2
coefs1      = summary(M1)$coefficients                         # Betaparameterschätzer Modell 1
coefs2      = summary(M2)$coefficients                         # Betaparameterschätzer Modell 2
ci1         = confint(M1, level  = 0.95)                       # Konfidenzintervalle Modell 1
ci2         = confint(M2, level  = 0.95)                       # Konfidenzintervalle Modell 2
```

Modellschätzung und Modellevaluation mit `lm()`

Ergebnisse M1

	Estimate	Std. Error	t value	Pr(>|t|)	KI.u	KI.o
(Intercept)	4.467	0.5740	7.783	2.201e-09	3.305	5.629
TRM2	5.317	0.8117	6.550	1.006e-07	3.673	6.960

Ergebnisse M2

	Estimate	Std. Error	t value	Pr(>|t|)	KI.u	KI.o
(Intercept)	9.6262	1.004	9.589	1.423e-11	7.592	11.6603
TRM2	5.0795	0.603	8.424	3.954e-10	3.858	6.3014
x	-0.9874	0.174	-5.673	1.739e-06	-1.340	-0.6347

Adjustierte Gruppenmittelwerte

Mit dem Theorem zur Darstellung der Gruppenmittel im Kovarianzanalysemodell gelten

$$\bar{y}_0 = \hat{\beta}_0 + \hat{\beta}_2 \bar{x}_0 \text{ und } \bar{y}_1 = \hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 \bar{x}_1 \text{ sowie } \bar{y}_1 - \bar{y}_0 = \hat{\beta}_1 + \hat{\beta}_2(\bar{x}_1 - \bar{x}_0). \tag{84}$$

Die Gruppenmittel werden also bei unterschiedlichen Kovariatenwerten ausgewertet, $\bar{y}_0$ bei $\bar{x}_0$ und $\bar{y}_1$ bei $\bar{x}_1$. Weiterhin hängt die Gruppenmittelwertdifferenz $\bar{y}_1 - \bar{y}_0$ explizit von der Kovariatenmittelwertdifferenz $\bar{x}_1 - \bar{x}_0$ ab.

- Sind sowohl $\bar{x}_1 - \bar{x}_0$ als auch $\hat{\beta}_2$ groß, so kann $\bar{y}_1 - \bar{y}_0$ auch dann groß sein, wenn $\hat{\beta}_1$ klein ist.
- Gruppenmittelwertsdifferenzen können also durch Kovariatenmittelwertsdifferenzen induziert sein.
- Ist entweder $\bar{x}_1 - \bar{x}_0$ oder $\hat{\beta}_2$ gleich Null, so gilt $\bar{y}_1 - \bar{y}_0 = \hat{\beta}_1$.

Um diese Überlagerung von Gruppen- und Kovariateneffekten zu vermeiden, werden die Gruppenmittelwerte im Kovarianzanalysemodell üblicherweise bei einem gemeinsamen Referenzwert x^* der Kovariate, zum Beispiel ihrem Gesamtmittelwert $\bar{x}$, ausgewertet. Die so *adjustierten Gruppenmittelwerte* ergeben sich damit zu

$$\bar{y}_0^* = \hat{\beta}_0 + \hat{\beta}_2 x^*, \text{ und } \bar{y}_1^* = \hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 x^*. \tag{85}$$

Für die adjustierte Gruppenmittelwertsdifferenz gilt damit

$$\bar{y}_1^* - \bar{y}_0^* = \hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 x^* - \hat{\beta}_0 - \hat{\beta}_2 x^* = \hat{\beta}_1. \tag{86}$$

Man beachte allerdings, dass die Schätzung von $\hat{\beta}_1$ im Kovarianzanalysemodell unabhängig von $\bar{x}_1 - \bar{x}_0$ ist, da $\hat{\beta}_1$ die Kovariatenmittelwertdifferenz $\bar{x}_1 - \bar{x}_0$ bereits berücksichtigt. Das Bestimmen adjustierter Gruppenmittelwerte ist also nicht notwendig, um den Gruppeneffekt $\hat{\beta}_1$ zu schätzen, sondern dient lediglich der Vermeidung von Darstellungswidersprüchen, wenn gleichzeitig Gruppenmittelwerte und Gruppeneffekte berichtet werden.

Adjustierte Gruppenmittelwerte

Das Theorem zu Konfidenzintervallen für Betaparameterkontraste lässt sich direkt auf adjustierte Gruppenmittelwerte anwenden, weil ein adjustierter Gruppenmittelwert im Kovarianzanalysemodell ebenfalls ein linearer Kontrast der Betaparameter ist. Speziell gilt für einen gemeinsamen Referenzwert x^*, dass

$$\bar{y}_0^* = \hat{\beta}_0 + \hat{\beta}_2 x^* \text{ und } \bar{y}_1^* = \hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 x^*. \tag{87}$$

Schreibt man dies in ALM-Notation, so ergeben sich die Kontrastvektoren

$$c_0^T := (1, 0, x^*) \text{ und } c_1^T := (1, 1, x^*), \tag{88}$$

Das Theorem zu Konfidenzintervallen für Betaparameterkontraste liefert damit mit

$$t_\delta := \Psi^{-1}\left(\frac{1+\delta}{2}; n-p\right) \text{ und} \lambda_{c_i} = c_i^T (X^T X)^{-1} c_i \tag{89}$$

das δ-Konfidenzintervall

$$\left[c_i^T \hat{\beta} - t_\delta \hat{\sigma}\sqrt{\lambda_{c_i}}, c_i^T \hat{\beta} + t_\delta \hat{\sigma}\sqrt{\lambda_{c_i}}\right]. \tag{90}$$

Eine Diskussion von adjustierten Gruppenmittelwerten findet sich beispielsweise in S. R. Searle, Speed, and Milliken (1980) und Senn (2006). In **R** werden adjustierte Gruppenmittelwerte üblicherweise mit der Funktion `emmeans()` aus dem gleichnamigen Paket berechnet.

Adjustierte Gruppenmittelwerte

```
D           = read.csv("2-Daten/kva-1.csv")             # Datensatz
beta_hat    = B[[2]]                                    # Betaparameterschätzer Modell 2
x_star      = mean(D$x)                                 # Gesamtmittelwert der Kovariate
X           = XS[[2]]                                   # Designmatrix Modell 2
n           = nrow(X)                                   # Gesamtanzahl Datenpunkte
p           = ncol(X)                                   # Anzahl Betaparameter
df          = n - p                                     # Freiheitsgrade
V_beta_hat  = S[2] * solve(t(X) %*% X)                  # Kovarianzmatrix der Schätzer
L           = rbind(c(1, 0, x_star), c(1, 1, x_star))   # Kontraste für adjustierte Gruppenmittelwerte
emm         = L %*% beta_hat                            # adjustierte Gruppenmittelwerte
emm_se      = sqrt(diag(L %*% V_beta_hat %*% t(L)))     # Standardfehler der EMMs
t_crit      = qt(.975, df = df)                         # t Kritischer Wert
emm_ci_u    = emm - t_crit * emm_se                     # untere KI Grenze
emm_ci_o    = emm + t_crit * emm_se                     # obere KI Grenze
out         = cbind(emmean   = emm,                     # Ausgabe EMMs
                    SE       = emm_se,                  # Standardfehler der EMMs
                    df       = df,                      # Freiheitsgradparameter
                    lower.CL = emm_ci_u,                # untere KI Grenze
                    upper.CL = emm_ci_o)                # obere  KI Grenze
rownames(out) = c("Kontrolle", "Treatment")               # Zeilennamen
print(round(out, digits = 3))                           # Ausgabe
```

```
                   SE df
Kontrolle 4.586 0.426 37 3.723  5.449
Treatment 9.665 0.426 37 8.802 10.528
```

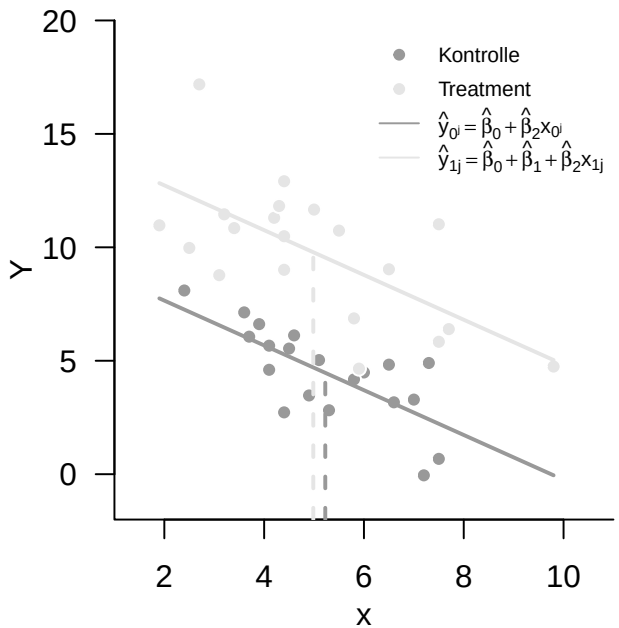
Adjustierte Gruppenmittelwerte

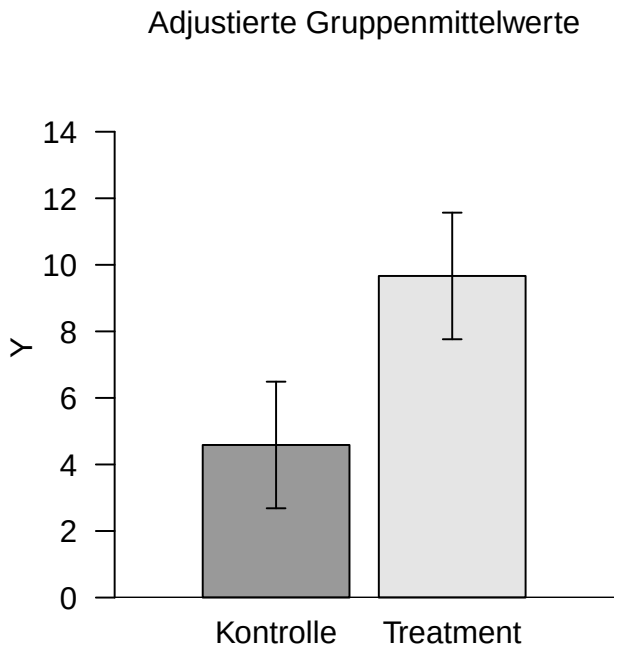
```
library(emmeans)
D           = read.csv("2-Daten/kva-1.csv")         # Datensatz
D$TRM       = as.factor(D$TRM)                      # R Faktor Kodierung
M2          = lm(Y ~ TRM + x, data = D)             # ALM Schätzung Modell 2
x_star      = mean(D$x)                             # Gesamtmittelwert der Kovariate
emmeans(M2, ~ TRM, at = list(x = x_star))           # EMMs an x*
```

```
 TRM emmean    SE df lower.CL upper.CL
 1     4.59 0.426 37     3.72     5.45
 2     9.67 0.426 37     8.80    10.53

Confidence level used: 0.95
```

Adjustierte Gruppenmittelwerte





Dokumentation

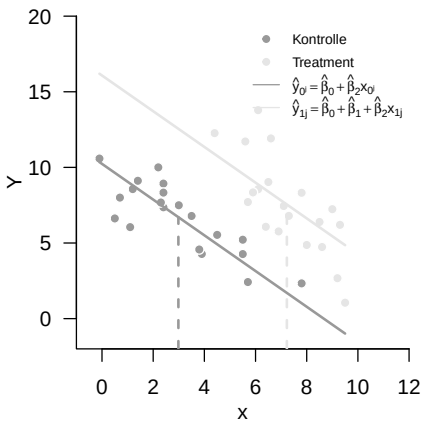
Anwendungsbeispiel Klinische Psychologie

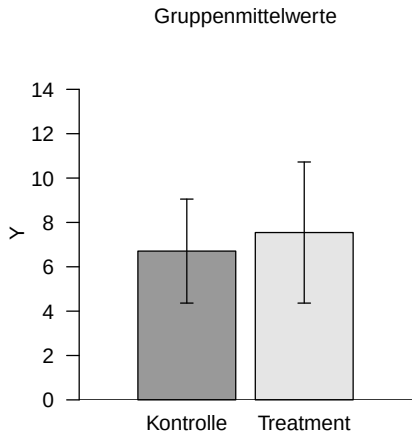
Deskriptiv war die Bereitschaft zur Therapieaufnahme in der Gruppe mit Selbstoffenbarung durch die Therapeut:in höher (M = 9.78, SD = 3.01) als in der Gruppe ohne Selbstoffenbarung (M = 4.47, SD = 2.03). Unter Adjustierung für das allgemeine interpersonelle Vertrauen als Kovariate ergaben sich am Gesamtmittelwert der Kovariate ($\bar{x} = 5.11$) geschätzte marginale Mittelwerte von 9.67 (SE = 0.43, 95%-KI [8.80, 10.53]) beziehungsweise 4.59 (SE = 0.43, 95%-KI [3.72, 5.45]); die Kovarianzanalyse zeigte einen signifikanten positiven Gruppeneffekt der Selbstoffenbarung, b = 5.08, SE = 0.60, 95%-KI [3.86, 6.30], t(37) = 8.42, p < .001.

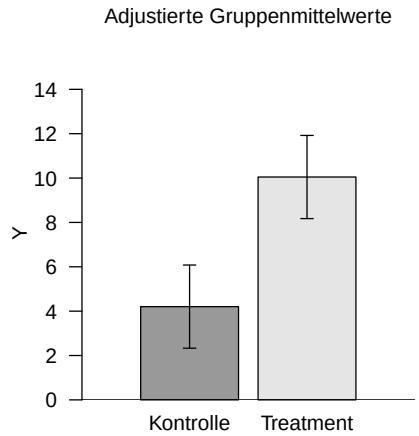
Anwendungsbeispiel Umweltpsychologie

Deskriptiv war das subjektive Stresserleben in der Gruppe mit Exposition gegenüber einer urbanen Umgebung höher (M = 9.78, SD = 3.01) als in der Gruppe mit Exposition gegenüber einer Naturumgebung (M = 4.47, SD = 2.03). Unter Adjustierung für die generelle Naturverbundenheit als Kovariate ergaben sich am Gesamtmittelwert der Kovariate ($\bar{x} = 5.11$) geschätzte marginale Mittelwerte von 9.67 (SE = 0.43, 95%-KI [8.80, 10.53]) beziehungsweise 4.59 (SE = 0.43, 95%-KI [3.72, 5.45]); die Kovarianzanalyse ergab einen signifikanten Unterschied zwischen den Bedingungen, b = 5.08, SE = 0.60, 95%-KI [3.86, 6.30], t(37) = 8.42, p < .001.

Datenszenario mit maskierten Gruppeneffekten







Datenszenario mit maskierten Gruppeneffekten

Wahre, aber unbekannte Werte: $\beta = (10, 5, -1)^T$ und $\sigma^2 = 4$.

```
M1 | Betaparameterschätzer    :  6.71 0.84
M2 | Betaparameterschätzer    :  10.23 5.84 -1.18
M1 | Varianzparameterschätzer :  7.81
M2 | Varianzparameterschätzer :  3.51
M1 | T-Statistik TRM 2        :  0.95
M2 | T-Statistik TRM 2        :  6.23
```

Im Kovarianzanalysemodell ist der Varianzparameterschätzer kleiner als in Modell 1.

$\Rightarrow$ Der Kovariatenregressor erklärt Datenvarianz.

Die Betaparameterschätzer für den Gruppeneffekt unterscheiden sich in M1 und M2 deutlich.

$\Rightarrow$ M1 unterschätzt den Gruppeneffekt, weil Kovariateneffekte und Gruppeneffekte überlagert werden.

Datenszenario mit maskierten Gruppeneffekten

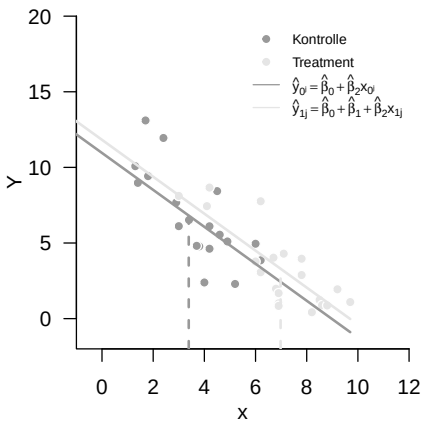
Anwendungsbeispiel Klinische Psychologie

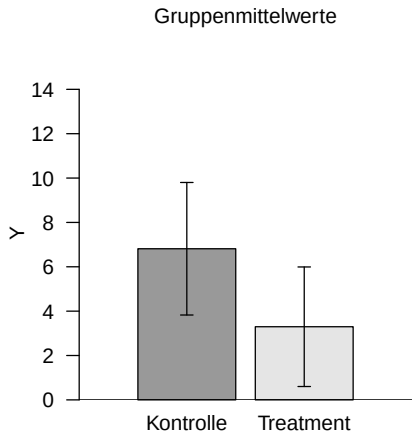
Deskriptiv war die Bereitschaft zur Therapieaufnahme in der Gruppe mit Selbstoffenbarung durch die Therapeut:in höher (M = 7.54, SD = 3.18) als in der Gruppe ohne Selbstoffenbarung (M = 6.71, SD = 2.34). Unter Adjustierung für das allgemeine interpersonelle Vertrauen als Kovariate ergaben sich am Gesamtmittelwert der Kovariate ($\bar{x} = 5.11$) geschätzte marginale Mittelwerte von 10.05 (SE = 0.55, 95%-KI [8.92, 11.17]) beziehungsweise 4.20 (SE = 0.55, 95%-KI [3.08, 5.33]); die Kovarianzanalyse zeigte einen signifikanten positiven Gruppeneffekt der Selbstoffenbarung, b = 5.84, SE = 0.94, 95%-KI [3.94, 7.74], t(37) = 6.23, p < .001.

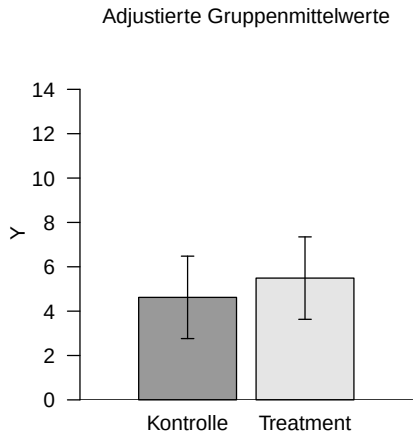
Anwendungsbeispiel Umweltpsychologie

Deskriptiv war das subjektive Stresserleben in der Gruppe mit Exposition gegenüber einer urbanen Umgebung höher (M = 7.54, SD = 3.18) als in der Gruppe mit Exposition gegenüber einer Naturumgebung (M = 6.71, SD = 2.34). Unter Adjustierung für die generelle Naturverbundenheit als Kovariate ergaben sich am Gesamtmittelwert der Kovariate ($\bar{x} = 5.11$) geschätzte marginale Mittelwerte von 10.05 (SE = 0.55, 95%-KI [8.92, 11.17]) beziehungsweise 4.20 (SE = 0.55, 95%-KI [3.08, 5.33]); die Kovarianzanalyse ergab einen signifikanten Unterschied zwischen den Bedingungen, b = 5.84, SE = 0.94, 95%-KI [3.94, 7.74], t(37) = 6.23, p < .001.

Datenszenario mit scheinbarem Gruppeneffekt







Datenszenario mit scheinbarem Gruppeneffekt

Wahre, aber unbekannte Werte: $\beta = (10, 0, -1)^T$ und $\sigma^2 = 4$.

```
M1 | Betaparameterschätzer    :  6.81 -3.52
M2 | Betaparameterschätzer    :  10.95 0.87 -1.22
M1 | Varianzparameterschätzer :  8.09
M2 | Varianzparameterschätzer :  3.45
M1 | T-Statistik TRM 2        :  -3.91
M2 | T-Statistik TRM 2        :  1.03
```

Im Kovarianzanalysemodell ist der Varianzparameterschätzer kleiner als in M1.

$\Rightarrow$ Der Kovariatenregressor erklärt Datenvarianz.

Der in M1 sichtbare Gruppenunterschied verschwindet in M2 weitgehend.

$\Rightarrow$ Der unadjustierte Gruppenunterschied wird durch die Kovariate erklärt, nicht durch einen echten Gruppeneffekt.

Datenszenario mit scheinbarem Gruppeneffekt

Anwendungsbeispiel Klinische Psychologie

Deskriptiv war die Bereitschaft zur Therapieaufnahme in der Gruppe ohne Selbstoffenbarung durch die Therapeut:in höher (M = 6.81, SD = 2.99) als in der Gruppe mit Selbstoffenbarung (M = 3.30, SD = 2.70). Unter Adjustierung für das allgemeine interpersonelle Vertrauen als Kovariate ergaben sich am Gesamtmittelwert der Kovariate ($\bar{x} = 5.18$) geschätzte marginale Mittelwerte von 4.62 (SE = 0.51, 95%-KI [3.58, 5.66]) beziehungsweise 5.49 (SE = 0.51, 95%-KI [4.45, 6.53]); die Kovarianzanalyse ergab keinen signifikanten Gruppeneffekt der Selbstoffenbarung, b = 0.87, SE = 0.85, 95%-KI [-0.84, 2.58], t(37) = 1.03, p = .311.

Anwendungsbeispiel Umweltpsychologie

Deskriptiv war das subjektive Stresserleben in der Gruppe mit Exposition gegenüber einer Naturumgebung höher (M = 6.81, SD = 2.99) als in der Gruppe mit Exposition gegenüber einer urbanen Umgebung (M = 3.30, SD = 2.70). Unter Adjustierung für die generelle Naturverbundenheit als Kovariate ergaben sich am Gesamtmittelwert der Kovariate ($\bar{x} = 5.18$) geschätzte marginale Mittelwerte von 4.62 (SE = 0.51, 95%-KI [3.58, 5.66]) beziehungsweise 5.49 (SE = 0.51, 95%-KI [4.45, 6.53]); die Kovarianzanalyse ergab keinen signifikanten Unterschied zwischen den Bedingungen, b = 0.87, SE = 0.85, 95%-KI [-0.84, 2.58], t(37) = 1.03, p = .311.

Allgemeines lineares Modell

Parallelgruppendesign

Einfache lineare Regression

Kovarianzanalyse

Selbstkontrollfragen

Selbstkontrollfragen

1. Geben Sie die Definition des Allgemeinen Linearen Modells wieder und erläutern Sie sie.
2. Geben Sie das Theorem zur Datenverteilung des Allgemeinen Linearen Modells wieder.
3. Geben Sie das Theorem zum Betaparameterschätzer wieder und erläutern Sie es.
4. Geben Sie das Theorem zur Frequentistischen Verteilung des Betaparameterschätzers wieder.
5. Geben Sie das Theorem zum Varianzparameterschätzer wieder und erläutern Sie es.
6. Geben Sie das Theorem zur Frequentistischen Verteilung des Varianzparameterschätzers wieder.
7. Geben Sie die Definition der T-Statistik wieder und erläutern Sie sie in Abhängigkeit von β_0.
8. Geben Sie das Theorem zur Frequentistischen Verteilung der T-Statistik wieder.
9. Geben Sie das Theorem zu den Konfidenzintervallen für Betaparameterkomponenten wieder.
10. Geben Sie das Theorem zu den Konfidenzintervallen für Betaparameterkontraste wieder.
11. Erläutern Sie die Syntax und Semantik der R `lm()` Funktion.
12. Erläutern Sie die Syntax und Semantik von R `formulas`.
13. Erläutern Sie das Anwendungsszenario eines Parallelgruppendesigns mit Post-Messung.
14. Geben Sie die strukturelle und die Designmatrixform des Parallelgruppen-Modells wieder.
15. Erläutern Sie das Anwendungsszenario einer einfachen linearen Regression.
16. Geben Sie die strukturelle und die Designmatrixform des Modells der einfachen linearen Regression wieder.
17. Erläutern Sie die Grundidee der Kovarianzanalyse.
18. Geben Sie die strukturelle Form und die Designmatrixform eines Kovarianzanalysemodells für ein Parallelgruppendesign mit einer Kovariate wieder.
19. Geben Sie das Theorem zur Darstellung der Gruppenmittelwerte im Kovarianzanalysemodell wieder.
20. Erläutern Sie die Bedeutung von adjustierten Gruppenmittelwerten im Kovarianzanalysemodell.

Anhang

Definition (Kronecker-Produkt)

Es seien $A \in \mathbb{R}^{m \times n}$ und $B \in \mathbb{R}^{p \times q}$. Dann ist das *Kronecker-Produkt* von A und B definiert als die Abbildung

$$\otimes : \mathbb{R}^{m \times n} \times \mathbb{R}^{p \times q} \to \mathbb{R}^{mp \times nq},\ (A, B) \mapsto \otimes(A, B) := A \otimes B \tag{91}$$

mit

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}. \tag{92}$$

Bemerkungen

- Das Kronecker-Produkt kann zur Konstruktion von Matrizen mit repetitiver Struktur genutzt werden.
- In der Form $1_n \otimes M$ setzt das Kronecker-Produkt die Matrix M n-mal übereinander.
- In der Form $M \otimes 1_n$ repliziert das Kronecker-Produkt jede Zeile von M n-mal.
- In der Form $I_n \otimes M$ erzeugt das Kronecker-Produkt eine Blockdiagonalmatrix M auf der Diagonale.

Beispiel (1)

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{pmatrix} \tag{93}$$

```
M       = matrix(c(1,1,                          # Matrix
                   1,2,
                   1,3),
                   byrow = TRUE, nrow = 3)
j_2     = matrix(rep(1,2)      , nrow = 2)       # 1_2 Vektor
X       = kronecker(j_2,M)                       # Kronecker-Produkt
```

```
     [,1] [,2]
[1,]    1    1
[2,]    1    2
[3,]    1    3
[4,]    1    1
[5,]    1    2
[6,]    1    3
```

Kronecker-Produkt

Beispiel (2)

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix} \tag{94}$$

```
M       = matrix(c(1,0,0,0,                         # Matrix
                   1,1,0,0,
                   1,0,1,0,
                   1,0,0,1),
                   byrow = TRUE, nrow = 4)
j_2     = matrix(rep(1,2)      , nrow = 2)          # 1_2 Vektor
X       = kronecker(M,j_2)                          # Kronecker-Produkt
```

```
     [,1] [,2] [,3] [,4]
[1,]    1    0    0    0
[2,]    1    0    0    0
[3,]    1    1    0    0
[4,]    1    1    0    0
[5,]    1    0    1    0
[6,]    1    0    1    0
[7,]    1    0    0    1
[8,]    1    0    0    1
```

Beispiel (3)

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 1 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix} \tag{95}$$

```
M       = matrix(c(2,1,                         # Matrix
                   1,2),
                   byrow = TRUE, nrow = 2)
I_2     = diag(2)                               # I_2
X       = kronecker(I_2,M)                      # Kronecker-Produkt
```

```
     [,1] [,2] [,3] [,4]
[1,]    2    1    0    0
[2,]    1    2    0    0
[3,]    0    0    2    1
[4,]    0    0    1    2
```

Referenzen

Chambers, John M., and Trevor Hastie, eds. 1992. *Statistical Models in S*. Wadsworth & Brooks/Cole Computer Science Series. Pacific Grove, Calif: Wadsworth & Brooks/Cole Advanced Books & Software.

Linden, Wim J. van der, ed. 2016. *Handbook of Item Response Theory*. Chapman & Hall/CRC Statistics in the Social and Behavioral Sciences Series. Boca Raton: CRC Press. https://doi.org/10.1201/9781315374512.

Rencher, Alvin C., and G. Bruce Schaalje. 2008. *Linear Models in Statistics*. 2nd ed. Hoboken, N.J: Wiley-Interscience.

Searle, S R, F M Speed, and G A Milliken. 1980. "Population Marginal Means in the Linear Model: An Alternative to Least Squares Means." *The American Statistician* 34 (4): 216–22.

Searle, S. R. 1971. *Linear Models*. Wiley Classics Library. New York, NY: Wiley.

Senn, Stephen. 2006. "Change from Baseline and Analysis of Covariance Revisited." *Statistics in Medicine* 25 (24): 4334–44. https://doi.org/10.1002/sim.2682.

Wilkinson, G. N., and C. E. Rogers. 1973. "Symbolic Description of Factorial Models for Analysis of Variance." *Applied Statistics* 22 (3): 392. https://doi.org/10.2307/2346786.