

Statistik für Digital Humanities

Mehrstufige Modelle & Wachstumsmodelle / Trendanalysen

Dr. Jochen Tiepmar

Institut für Informatik
Computational Humanities
Universität Leipzig

24. August 2020

[Letzte Aktualisierung: 24/08/2020, 15:36]

Hierarchische Daten

Bisher:

- flach
- 1 Level / Stufe
- Gruppeneinteilungen
- Gruppen als unabhängig voneinander angenommen

Jetzt:

- Hierarchisch
- >1 Level / Stufe Multilevel
- Gruppeneinteilungen \approx Kontextvariablen
- Daten innerhalb einer Kontextvariable sind **verschachtelte Variablen** nested
- 2-stufig, 3-stufig, . . .
- Gruppenabhängigkeiten als Modell

Beispiel

2-Stufig, verschachtelte Variable unabhängig:

Schulklasse	A	B	C	D	E	F	Level 2
Englishnote pro Kind	1	3	4	2	3	2	Stufe 1
	2	3	5	3	4	2	
	3	4	5	1		5	
	1		2	2		3	
			3	2		2	

- Klasse A und D unterrichtet von native Speaker
- → Noten also abhängig von der Kontextvariable

Beispiel

3-Stufig, verschachtelte Variable unabhängig:

Schule	OBA	OBA	KEM	KEM	STU	STU	Stufe 3
Schulklasse	A	B	C	D	E	F	Stufe 2
Englishnote pro Kind	1	3	4	2	3	2	Level 1
	2	3	5	3	4	2	
	3	4	5	1		5	
	1		2	2		3	
			3	2		2	

– 2 Kontextvariablen *Schule* und *Schulklasse*

Beispiel

3-Stufig, verschachtelte Variable abhängig: Messwiederholung

Schulklasse	A	A	B	B	C	C	Stufe 3
Kind	1	2	3	4	5	6	Level 2
Englishnote fortlaufend	1	3	4	2	3	2	Level 1
	2	3	5	3	4	2	
	3	4	5	1		5	
	1		2	2		3	
			3	2		2	

- Noten fortlaufend im Jahr pro Kind als abhängige Variable
→ Motivation, Lehrbias,...
- Beachte, dass sich nur die Beschreibung geändert hat
- mit *Schule* ergibt sich Level 4

Gruppen vs. Hierarchie

Warum Hierarchie nicht als Gruppenzuordnung verstehen wie bisher?

- Mehr-stufig
- Abhängige Einteilungen
- Je stärker **Intraklassen-Korrelation (ICC)**, desto eher Hierarchie

Beispiel Arachnophobie Siehe t-Test

Gruppen vs. Hierarchie

Proband	Gruppe	Angst
1	Real	30
2	Real	35
3	Real	45
4	Bild	40
5	Bild	35
6	Bild	50

vs.

Proband	Herkunft	Angst
1	Dorf A	30
2	Dorf A	35
3	Stadt B	45
4	Stadt C	40
5	Dorf D	35
6	Stadt E	50

Die Abhängigkeit selbst ist bei Hierarchien Teil des Modells.

Intraklassen-Korrelation (ICC)

- Anteil der Variation, die durch Klasse erklärt wird
- Variation in den Klassen vs. Variation zwischen den Klassen
- Korrelationswert $-1 \dots 1$
- hoher \pm ICC \rightarrow :
 - \rightarrow Variation in Klassen gering
 - \rightarrow Variation zwischen Klassen hoch
 - \rightarrow \rightarrow Kontextvariable hat hohen Einfluss
 - \rightarrow \rightarrow Hierarchie

(komplexe) Berechnung Siehe Field, A.P. (2005): *Intraclass Correlation*
(Moodle)

Vorteile Mehrstufiger Modelle

Potentielle Lösung für folgende Probleme:

- Heterogene Anstiege der Regressionslinien Siehe ANCOVA
- Problematische Abhängigkeit der Messwerte (je nach Anwendung) bezogen auf höherstufige Variablen
- Fehlende Werte, die besonders bei abhängigen Designs teure Reparaturen (Löschungen) erfordern
 - Aber R mitteilen (bspw. `na.action=na.exclude`)

Beispieldaten

Beispiel Kosmetische Chirurgie

- Post_QoL → Outcomevariable, Quality of Life nach Operation
- Base_QoL → Adjustierungsvariable, Quality of Life vor Operation
- Surgery → 1 = fertig, 0 = Warteliste (Kontrollgruppe)
- Clinic → Besuchte Klinik
- Age → Alter
- BDI → Beck Depression Inventory
- Reason → 1 = Körperlich, 0 = Kosmetisch
- Reason_Text → Begründung als Text
- Gender → 1 = Mann, 0 = Frau

- *Clinic* als Kontextvariable zu jeder Person
- Jeweilige Messwerte für jede Person

Daten Siehe Moodle

Feste vs. Zufällige Koeffizienten

Feste Koeffizienten *fixed*

- $\hat{Y} = (b_0 + b_1 * X_1)$
- Schnittpunkt und Koeffizienten konstant

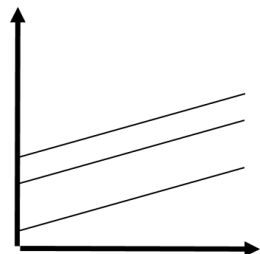
Zufällige Koeffizienten

- $\hat{Y} = ((b_0 + u_0) + (b_1 + u_1) * X_1)$
- Schnittpunkt und Koeffizienten variabel
- u_i ist die Variabilität des Koeffizienten i

Zufälliger Anstieg und zufälliger Schnittpunkt

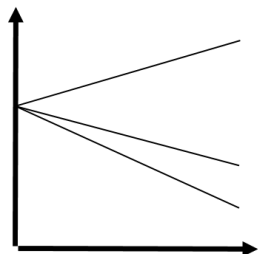
Jede der Linien entspricht jetzt einem Kontext (Klinik)

Fester Anstieg, Zufälliger Schnittpkt



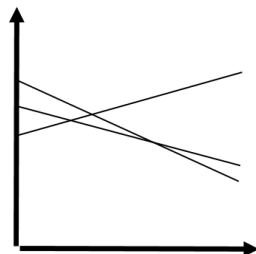
$$\hat{Y} = ((b_0 + u_0) + b_1 * X_1)$$

Zufälliger Anstieg, Fester Schnittpkt



$$\hat{Y} = (b_0 + (b_1 + u_1) * X_1)$$

Zufälliger Anstieg, Zufälliger Schnittpkt



$$\hat{Y} = ((b_0 + u_0) + (b_1 + u_1) * X_1)$$

Das Mehrstufige Modell

- Wir starten mit einem einfachen linearen Modell

$$Post_QoL_i = b_0 + b_1 * Surgery_i + e_i$$

$$Post_QoL_{Sam} = b_0 + b_1 * Surgery_{Sam}$$

- *Clinic* als Kontext, Zufälliger Schnittpunkt $Post_QoL_i = b_{0,j} + b_1 * Surgery_{i,j} + e_{i,j}$

$$b_{0,j} = b_0 + u_{0,j}$$

$$Post_QoL_{Sam} = b_{0,Clinic7} + b_1 * Surgery_{Sam,Clinic7}$$

$$b_{0,Clinic7} = b_0 + u_{0,Clinic7}$$

- Zufälliger Anstieg $Post_QoL_{Sam} = b_{0,Clinic7} + b_{1,Clinic7} * Surgery_{Sam,Clinic7}$

$$b_{0,Clinic7} = b_0 + u_{0,Clinic7} \quad b_{1,Clinic7} = b_1 + u_{1,Clinic7}$$

- Mehr Prädiktoren

$$Post_QoL_{Sam} = b_{0,Clinic7} + b_{1,Clinic7} * Surgery_{Sam,Clinic7} + b_2 * Base_QoL_{Sam,Clinic7}$$

$$b_{0,Clinic7} = b_0 + u_{0,Clinic7} \quad b_{1,Clinic7} = b_1 + u_{1,Clinic7}$$

Das Mehrstufige Modell

- Wir haben ein mehrstufiges Modell

$$Post_QoL_{Sam} = b_{0,Clinic7} + b_{1,Clinic7} * Surgery_{Sam,Clinic7} + b_2 * Base_QoL_{Sam,Clinic7}$$

$$b_{0,Clinic7} = b_0 + u_{0,Clinic7} \quad b_{1,Clinic7} = b_1 + u_{1,Clinic7}$$

$$Post_QoL_{Sam} =$$

$$(b_0 + u_{0,Clinic7}) + (b_1 + u_{1,Clinic7}) * Surgery_{Sam,Clinic7} + b_2 * Base_QoL_{Sam,Clinic7}$$

- *Clinic* = Kontext & Zufälliger Schnittpunkt
- *Surgery* = Zufälliger Koeffizient
- *Base_QoL* = Fester Koeffizient

Im Grunde fügen wir wie bei multipler Regression Parameter hinzu und entscheiden bei jedem, ob dieser fest oder zufällig sein soll.

Fitness

$$-2LL = -2 * \text{LogLikelihood}$$

- Je kleiner, desto besser
- χ^2 -Verteilung

Akaike Information Criterion (AIC)

- Beachtet Parsimony
- Je kleiner, desto besser

Schwarz's Bayesian Criterion (BIC)

- Konservativer als AIC
- Wichtiger bei großen Stichproben und wenigen Prädiktoren

Auswahl zufälliger Koeffizienten

- Start: Alle Koeffizienten fest
- Randomisiere passenden Koeffizient
- Teste auf Änderung

$$\chi^2_{change} = (-2LL_{old}) - (-2LL_{new})$$

$$df_{change} = \text{Anzahl Parameter}_{old} - \text{Anzahl Parameter}_{new}$$

Wichtig: Uneingeschränkte Maximum Likelihood nutzen und alle Effekte in *new* müssen auch in *old* sein

Raudenbush, S.W. & Byrk, A.S. (2002): *Hierarchical linear models*

Twisk, J.W.R. (2006): *Applied multilevel analysis*

Varianz-Kovarianz-Struktur

- Angabe nötig für R bei Messwiederholung und bei zufälligen Effekten
- Laut Rücksprache mit Andy Field eher *überlegenswert* als *nötig*
- = Form der Varianz-Kovarianz Matrix
- Diagonal = Varianz In *Clinic*
- Off-Diagonal = Kovarianz Zwischen *Clinic*
- In der Praxis einfach alle berechnen und besten Fit wählen

Typen von Varianz-Kovarianz-Strukturen

Varianzkomponenten

	C 1	C 2	C 3	C 4
C 1	1	0	0	0
C 2	0	1	0	0
C 3	0	0	1	0
C 4	0	0	0	1

Diagonal

	C 1	C 2	C 3	C 4
C 1	s_1^2	0	0	0
C 2	0	s_2^2	0	0
C 3	0	0	s_3^2	0
C 4	0	0	0	s_4^2

Autoregressive Struktur (Rang 1)

	C 1	C 2	C 3	C 4
C 1	1	p	p^2	p^3
C 2	p^3	1	p	p^2
C 3	p^2	p^3	1	p
C 4	p	p^2	p^3	1

Kovarianz nimmt ab

Unstrukturiert

	C 1	C 2	C 3	C 4
C 1	s_1^2	$s_{2,1}$	$s_{3,1}$	$s_{4,1}$
C 2	$s_{1,2}$	s_2^2	$s_{2,3}$	$s_{2,4}$
C 3	$s_{3,1}$	$s_{3,2}$	s_3^2	$s_{3,4}$
C 4	$s_{4,1}$	$s_{4,2}$	$s_{4,3}$	s_4^2

Kein Muster

Annahmen

- Annahmen der linearen Regression sollten gelten
- Unabhängigkeit muss nicht gelten für hochstufigen Variablen
- Normalverteilung der randomisierten Koeffizienten über das gesamte Modell
- Multikollinearität problematisch bei Interstufen-Interaktionen, aber lösbar durch Zentrierung Kreft, I.G.G. & de Leeuw, J. (1998): *Introducing multilevel modelling*

Stichprobengröße

Kreft, I.G.G. & de Leeuw, J. (1998): *Introducing multilevel modelling*

- Je mehr, desto besser
- Bei Interklassen-Interaktionen mindestens 20 Kontexte (Klassen) und "nicht zu kleine Gruppengrößen"

Twisk, J.W.R. (2006): *Applied multilevel analysis*

- Standardstichprobengrößen können gelten wenn korrigiert um Multilevel-Aspekte
 - 2 vorgeschlagene Korrekturen resultieren in stark unterschiedlichen Werten
- ... Die Experten sind sich uneinig

Zentrierung von Variablen

- Transformation von Variablen zu Abweichung von (beliebigen) Fixpunkt
- Zentrierung um 0: Von jedem Wert Mittelwert abziehen (Siehe z-Score)
- Grand-Mean Zentrierung oder Gruppen-Mean Zentrierung

Bei Mehrstufigen Modellen:

- Nur auf Level 1
- Sinnvoll wenn 0-Punkt sinnlos (Herzschlagrate)
- Ändert Schnittpunkt der Linie zum Durchschnittswert

Zentrierung von Variablen

- Tendenziell stabiler, Vorhersagen sind unabhängig
- Grand-Mean Zentrierung
 - vorhergesagte Werte, Fitness und Residuen bleiben
 - Koeffizienten ändern sich, aber sind zu vorher transformierbar
 - Modell bleibt gleich, Interpretation ändert sich
- Gruppen-Mean Zentrierung ändert Modell komplett, dann sollten Gruppenmittelwerte als Level 2 Variablen hinzugefügt werden, falls das nicht Untersuchungsgegenstand ist

Kreft, I.G.G. & de Leeuw, J. (1998): *Introducing multilevel modelling*

Kreft, I.G.G. & de Aiken, L.S. (1995): *The effect of different forms of centering in hierarchical linear models*

Zentrierung von Variablen

Auswahl von Forschung abhängig

Guidelines nach Enders, C.K. & Tofighi, D. (2007): *Centering predictor variables in cross-sectional multilevel models: A new look on an old issue*

:

- Gruppenmittelwert bei Untersuchung von Level 1 Variablen (*Surgery* → *Post_QoL*)
- Grand Mean bei Untersuchung von Level 2 Variablen mit Kontrolle der Level 1 Kovariaten (*Clinic* → *Post_QoL* unter Kontrolle von *Surgery*)
- Beide bei Untersuchung von Differentialeinfluss zwischen Level 1 und 2 (*Clinic* → *Post_QoL* verschieden von *Surgery*)
- Gruppenmittelwert bei Untersuchung von Interklassen-Interaktionen (*Clinic* × *Surgery* → *Post_QoL*)

Mehrstufige Modelle in R

Datenexploration

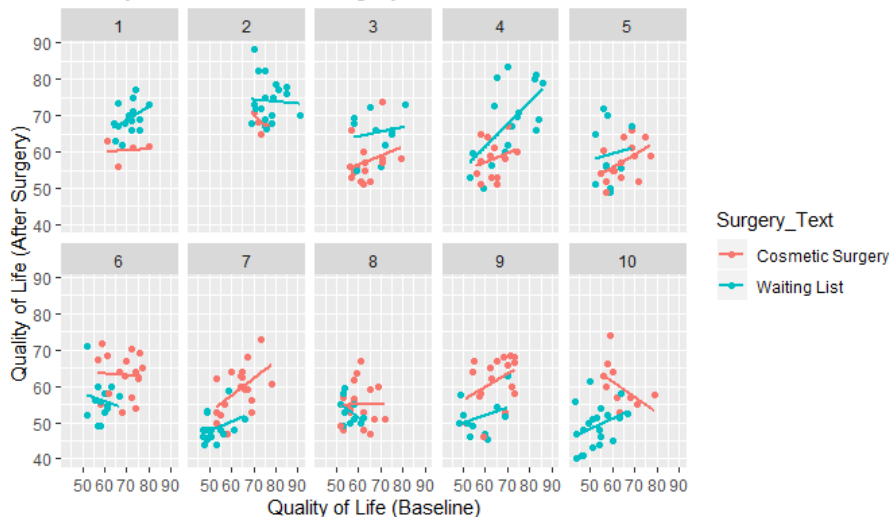
```
library(ggplot2)

surgeryData = read.delim("Cosmetic Surgery.dat", header = TRUE)
pgrid <- ggplot(surgeryData, aes(Base_QoL, Post_QoL))
pgrid + geom_point(aes(colour = Surgery_Text))
  + geom_smooth(aes(colour = Surgery_Text), method = "lm", se = F)
  + facet_wrap(~Clinic, ncol = 5)
  + labs(x = "Quality of Life (Pre Surgery, Baseline)"
, y = "Quality of Life (After Surgery)")
  + ggtitle("Quality of Life Pre-Post Surgery at 10 Clinics")
```

Mehrstufige Modelle in R

Datenexploration

Quality of Life Pre-Post Surgery at 10 Clinics



Mehrstufige Modelle in R

Sinnhaftigkeit von Mehrstufigem Modell bestimmen

```
library(nlme)
```

```
# Baseline Modell, Abschätzung über Schnittpunkt, Beachte method = "ML"  
interceptOnly <- gls(Post_QoL~1, data = surgeryData, method = "ML")  
# Ablesen von AIC (2017.104) und BIC (2024.365)  
summary(interceptOnly)
```

```
# Clinic als zufälliger Effekt, Abschätzung über Schnittpunkt pro Klinik  
randomInterceptOnly <- lme(Post_QoL~1, data = surgeryData, random = ~1|Clinic,  
  method = "ML")  
# Ablesen von AIC (1911.473) und BIC (1922.334)  
summary(randomInterceptOnly)
```

```
# -2LL für Signifikanztest  
logLik(interceptOnly)*-2  
>'log Lik.' 2013.124 (df=2)  
logLik(randomInterceptOnly)*-2  
>'log Lik.' 1905.473 (df=3)
```

Mehrstufige Modelle in R

Wir stellen fest:

- AIC und BIC zeigen Verbesserung des Modellfits
- $\chi^2_{change} = 2013.124 - 1905.473 = 107.65$, $df = 3 - 2 = 1$
 χ^2 Tabelle $\rightarrow 3.84(p = 0.05)$, $6.63(p = 0.01) \rightarrow$ Verbesserung sehr signifikant

Mehrstufiges Modell sinnvoll

Aktuelles Modell: $Post_QoL_{i,j} = (b_0 + u_0) + e_{i,j}$

Jetzt bauen wir

$Post_QoL_{i,j} = (b_0 + u_0) + (b_1 + u_1) * Surgery_{i,j} + b_2 * Base_QoL_{i,j} + e_{i,j}$

Mehrstufige Modelle in R

Prädiktoren hinzufügen

Surgery

```
randomInterceptSurgery <-lme(Post_QoL~Surgery, data = surgeryData,  
  random = ~1|Clinic, method = "ML")  
summary(randomInterceptSurgery) # Ablesen von AIC (1910.137) und BIC (1924.619)
```

Base_QoL

```
randomInterceptSurgeryQoL <-lme(Post_QoL~Surgery + Base_QoL, data = surgeryData  
  , random = ~1|Clinic, method = "ML")  
summary(randomInterceptSurgeryQoL) # Ablesen von AIC (1847.490) und BIC (1865.592)
```

Signifikanztest der Änderung

```
anova(randomInterceptOnly, randomInterceptSurgery, randomInterceptSurgeryQoL)
```

Mehrstufige Modelle in R

Prädiktoren hinzufügen

	Model	df	AIC	BIC	logLik	Test	L.Ratio
randomInterceptOnly	1	3	1911.473	1922.334	-952.7364		
randomInterceptSurgery	2	4	1910.137	1924.619	-951.0686	1 vs 2	3.33564
randomInterceptSurgeryQoL	3	5	1847.490	1865.592	-918.7450	2 vs 3	64.64721
			p-value				
randomInterceptOnly							
randomInterceptSurgery			0.0678				
randomInterceptSurgeryQoL			<.0001				

Wir stellen fest:

- Hinzufügen von *Surgery* hatte keinen signifikanten Effekt
- Hinzufügen von *Surgery* und *Base_QoL* hatte signifikanten Effekt

Mehrstufige Modelle in R

Surgery randomisieren

```
addRandomSlope<-lme(Post_QoL~Surgery + Base_QoL, data = surgeryData
  , random = ~Surgery|Clinic, method = "ML")
anova(randomInterceptSurgeryQoL,addRandomSlope)
```

	Model	df	AIC	BIC	logLik	Test	L.Ratio
randomInterceptSurgeryQoL	1	5	1847.490	1865.592	-918.7450		
addRandomSlope	2	7	1812.624	1837.966	-899.3119	1 vs 2	38.86626
							p-value
randomInterceptSurgeryQoL							
addRandomSlope							<.0001

Wir stellen fest:

- Randomisieren von *Surgery* zeigt signifikante Verbesserung
- Der Zusammenhang zwischen *Surgery* und *Post_QoL* unter kontrolliertem *Base_QoL* ist sehr unterschiedlich pro *Clinic*

Jetzt bauen wir

$$Post_QoL_i = (b_0 + u_0) + (b_1 + u_1) * Surgery_{i,j} + b_2 * Base_QoL_{i,j} + b_3 * Reason_{i,j} + b_4 * Reason_{i,j} * Surgery_{i,j} + e_{i,j}$$

Mehrstufige Modelle in R

Interaktion *Reason* × *Surgery*

```
addReason<-lme(Post_QoL~Surgery + Base_QoL + Reason, data = surgeryData,  
  random = ~Surgery|Clinic, method = "ML")  
addReason<-update(addRandomSlope, .~. + Reason)
```

```
addReasonxSurgery<-lme(Post_QoL~Surgery + Base_QoL + Reason + Reason:Surgery,  
  data = surgeryData, random = ~Surgery|Clinic, method = "ML")
```

```
anova(addRandomSlope, addReason, addReasonxSurgery)
```

	Model	df	AIC	BIC	logLik	Test	L.Ratio	p-value
addRandomSlope	1	7	1812.624	1837.966	-899.3119			
addReason	2	8	1810.825	1839.788	-897.4124	1 vs 2	3.798961	0.0513
addReasonxSurgery	3	9	1807.045	1839.629	-894.5226	2 vs 3	5.779555	0.0162

Wir stellen fest:

- Hinzufügen von *Reason* bringt knapp keine signifikante Verbesserung bei $p = 0.05$
- Hinzufügen von *Reason* × *Surgery* bringt signifikante Verbesserung bei $p = 0.05$

Mehrstufige Modelle in R

Modellanalyse

```
summary(addReasonxSurgery)
```

```
...  
Fixed effects: Post_QoL ~ Surgery + Base_QoL + Reason + Reason:Surgery  
                Value Std.Error DF   t-value p-value  
(Intercept)    42.51781  3.875317 262  10.971441  0.0000  
Surgery         -3.18768  2.185367 262  -1.458646  0.1459  
Base_QoL        0.30536  0.053125 262   5.747835  0.0000  
Reason         -3.51515  1.140934 262  -3.080939  0.0023  
Surgery:Reason  4.22129  1.700269 262   2.482718  0.0137  
Correlation:  
...  
Standardized Within-Group Residuals:  
...
```

Wir stellen fest:

- *Surgery* sagt *Post_QoL* nicht signifikant voraus
- *Base_QoL*, *Reason*, *Surgery:Reason* sagen *Post_QoL* signifikant voraus

Mehrstufige Modelle in R

Interpretation

Reason

- 1 = Gesundheitlich, 0 = Kosmetisch, $t = -3.080939$
- *Post_QoL* sinkt mit Erhöhung von 0 auf 1

Mehrstufige Modelle in R

Interpretation

Reason unter Beachtung der Kontrollgruppe

```
physicalSubset<- surgeryData$Reason==1
cosmeticSubset<-surgeryData$Reason==0
physicalModel<-lme(Post_QoL~Surgery + Base_QoL, data = surgeryData,
  random = ~Surgery|Clinic, subset= physicalSubset, method = "ML")
cosmeticModel<-lme(Post_QoL~Surgery + Base_QoL, data = surgeryData,
  random = ~Surgery|Clinic, subset= cosmeticSubset, method = "ML")
```

```
summary(physicalModel)
```

```
...
```

```
b=1.20,t=0.57,p=0.57
```

```
summary(cosmeticModel)
```

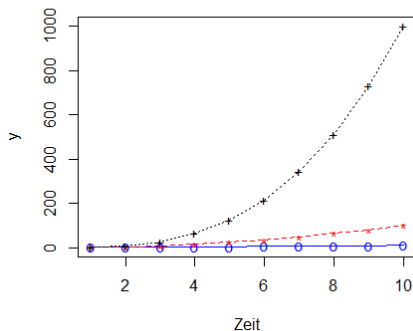
```
...
```

```
b=-4.31,t=-1.89,p=0.06
```

Verglichen mit Kontrollgruppe

- Verschlechterung der Lebensfreude bei kosmetischen Gründen
- Verbesserung der Lebensfreude bei körperlichen Gründen
- Keins von beiden hat signifikante Vorhersagekraft

Wachstumsmodelle



Exponentielle Verläufe: Im Lehrbuch Polynomiell(?)

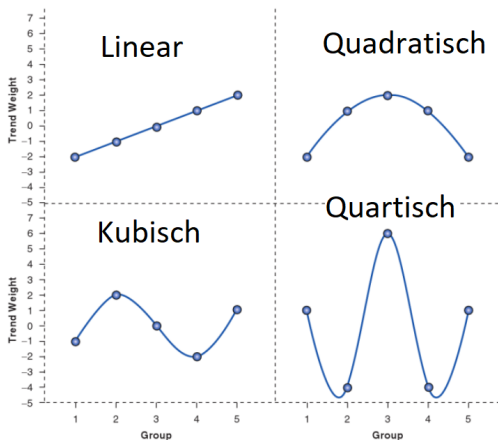
0 = x^1 First Order / Linear

* = x^2 Second Order / Quadratisch

+ = x^3 Third Order / Kubisch

Wachstumsmodelle

Polynomielle Verläufe:



Wachstumsmodelle

Wir betrachten also Wachstum über Zeit

- Potenz maximal *Zeitpunkte* - 1
 - 3 Zeitpunkte → maximal x^2
- *Zeit* als Prädiktorvariable in Mehrstufigem Modell

Frame Title

Achtung: Im Lehrbuch ist das Skript buggy. Mehr Infos dazu hier: <https://de.coredump.biz/questions/41240174/error-using-nlme-malformed-factor>

Wachstumsmodelle in R

Datenexploration (Datensatz Siehe Moodle)

```
satisfactionData = read.delim("Honeymoon Period.dat", header = TRUE)

restructuredData<-melt(satisfactionData, id = c("Person", "Gender"),
  measured = c("Satisfaction_Base", "Satisfaction_6_Months",
    "Satisfaction_12_Months", "Satisfaction_18_Months"))
names(restructuredData)<-c("Person", "Gender", "Time", "Life_Satisfaction")
# Bugfix
restructuredData$Time.Numeric <- with(restructuredData,
  ifelse(Time == "Satisfaction_Base", 0,
    ifelse(Time == "Satisfaction_6_Months", 1,
      ifelse(Time == "Satisfaction_12_Months", 2,
        ifelse(Time == "Satisfaction_18_Months", 3, NA))))))
```

Person	Gender	Time	Life_Satisfaction
1	0	Satisfaction_Base	6
...
123	0	Satisfaction_Base	5
1	0	Satisfaction_6_Months	6
...
123	0	Satisfaction_6_Months	6
...	...	Satisfaction_12_Months	...
...	...	Satisfaction_18_Months	...

Wachstumsmodelle in R

Modellierung und Fitnessbewertung

Beachte, dass wir Fehlwerte haben

```
intercept <-glms(Life_Satisfaction~1, data = restructuredData, method = "ML"
, na.action = na.exclude)
randomIntercept <-lme(Life_Satisfaction ~1, data = restructuredData
, random = ~1|Person, method = "ML", na.action = na.exclude
, control = list(opt="optim")) # Alternative Schätzungsmethode
# Wenn Modell nicht konvergiert
# kann opt und maxIter helfen

anova(intercept, randomIntercept)
```

	Model	df	AIC	BIC	logLik	Test	L.Ratio	p-value
intercept	1	2	2064.053	2072.217	-1030.0263			
randomIntercept	2	3	1991.396	2003.642	-992.6978	1 vs 2	74.65704	<.0001

Signifikante Verbesserung durch Randomisierung des Schnittpunktes nach *Person*

Wachstumsmodelle in R

Hinzufügen von Zeit

Beachte, dass wir Fehlwerte haben

```
#Zeit als fester Effekt
```

```
timeRI <- lme(Life_Satisfaction ~ Time.Numeric, data = restructuredData,  
             random = ~1|Person,  
             method = "ML", na.action = na.exclude, control = list(opt = "optim"))
```

```
#Zeit als zufälliger/veränderlicher Effekt
```

```
timeRS <- lme(Life_Satisfaction ~ Time.Numeric, data = restructuredData,  
             random = ~Time.Numeric|Person,  
             method = "ML", na.action = na.exclude, control = list(opt = "optim"))
```

timeRS variiert jetzt nach Zeit pro Person, der Effekt von *time* variiert also pro *Person*

Wachstumsmodelle in R

Messwiederholung → Kovarianz-Struktur festlegen

```
ARModel <- update(timerS, correlation = corAR1(0, form = ~Time.Numeric|Person))  
#An dieser Stelle kracht es im Lehrbuchskript
```

Wachstumsmodelle in R

Modelle analysieren

```
anova(intercept, randomIntercept, timeRI, timeRS, ARModel)  
summary(ARModel)
```

	Model	df	AIC	BIC	logLik	Test	L.Ratio	p-value
intercept	1	2	2064.053	2072.217	-1030.0263			
randomIntercept	2	3	1991.396	2003.642	-992.6978	1 vs 2	74.65704	<.0001
timeRI	3	4	1871.728	1888.057	-931.8642	2 vs 3	121.66714	<.0001
timeRS	4	6	1874.626	1899.120	-931.3131	3 vs 4	1.10224	0.5763
ARModel	5	7	1872.891	1901.466	-929.4453	4 vs 5	3.73564	0.0533

Wir stellen fest

- Zufälliger Schnittpunkt und *Time* verbesserten das Modell
- Zufällige *Time* verbessert das Modell nicht weiter
- Hinzufügen der Varianz-Kovarianz Struktur brachte knapp keine signifikante Verbesserung

Wachstumsmodelle in R

Modelle analysieren

```
...  
Fixed effects: Life_Satisfaction ~ Time.Numeric  
                Value Std.Error DF   t-value p-value  
(Intercept)    7.131470 0.21260192 322  33.54377    0  
Time.Numeric  -0.870087 0.07929275 322 -10.97310    0  
...
```

Wir stellen fest

- Zeit hat hochsignifikanten Einfluss (negativen) Einfluss auf die Lebensfreude verliebter Menschen

Wachstumsmodelle in R

Test auf polynomiale Trends

```
timeQuadratic<-update(ARModel, .~. + I(Time.Numeric^2))
timeCubic <-update(timeQuadratic, .~. + I(Time.Numeric^3))
anova(ARModel, timeQuadratic, timeCubic)
```

	Model	df	AIC	BIC	logLik	Test	L.Ratio	p-value
ARModel	1	7	1872.891	1901.466	-929.4453			
timeQuadratic	2	8	1817.544	1850.202	-900.7720	1 vs 2	57.34661	<.0001
timeCubic	3	9	1816.161	1852.901	-899.0808	2 vs 3	3.38245	0.0659

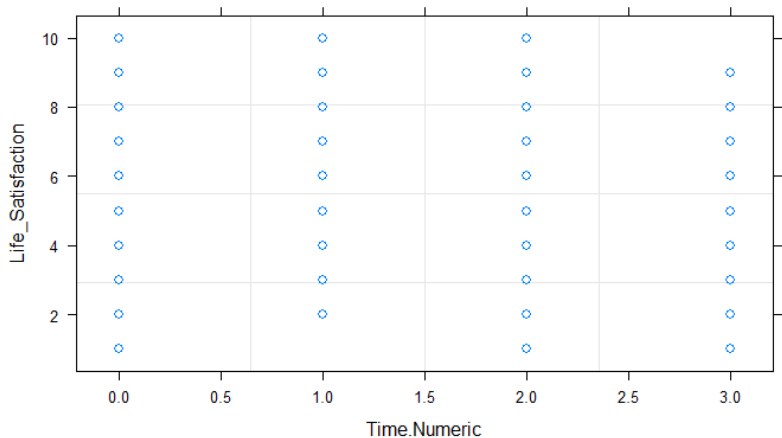
Wir stellen fest

- Das Hinzufügen von quadratischem Trend verbessert das Modell, die Daten verlaufen also tendenziell quadratisch polynomial
- Das Hinzufügen von kubischem Trend verbessert das Modell nicht

Wachstumsmodelle in R

Quadratischer Trend in den Daten

```
plot(ARModel, Life_Satisfaction ~ Time.Numeric)
```



Zusammenfassung

- Hierarchische Daten vs. Gruppen
- Mehrstufige Modelle
 - Lineare Modelle mit variablen Schnittpunkt und Koeffizienten
 - Besser bei störenden Abhängigkeiten
 - Lösung bei Fehlern und heterogenen Regressionslinien
 - Fitness mittels AIC, BIC und χ^2_{change}
 - Varianz-Kovarianzstruktur
 - Zentrierung
- Wachstumsmodelle / Zeittrends
 - Polynomielles & Exponentielles Wachstum
 - Zeitverlauf modelliert als mehrstufigem Modell