LM: Kategorielle Eingangsgrössen

Kolloquium für Statistik

Departement of Health Professions
Bern University of Applied Sciences

4. Oktober 2023

Zwei kategorielle Eingangsgrössen

- Zwei kategorielle Eingangsgrössen, einen Faktor A mit I Kategorien und einen Faktor B mit J Kategorien
- $I \times J$ Stichproben S_{ij} mit Grösse n_{ij} und $n = \sum_{i,j} n_{ij}$.

		Faktor B					
		B_1	B_2	$\cdots$	B_j	$\cdots$	B_J
	A_1	S_{11}	S_{12}	$\cdots$	S_{1j}	$\cdots$	S_{1J}
	A_2	S_{21}	S_{22}	$\cdots$	S_{2j}	$\cdots$	S_{2J}
Faktor A	$\vdots$						$\vdots$
	A_i			$\cdots$	S_{ij}	$\cdots$	S_{iJ}
	$\vdots$						$\vdots$
	A_I	S_{I1}	S_{I2}	$\cdots$	S_{Ij}	$\cdots$	S_{IJ}

- Klassisch nennt man ein solches Design eine zweifaktorielle Varianzanalyse.

Means-Parameterisierung

- Die abhängige Variable Y_{ij} setzt sich zusammen aus dem linearen Prädiktor μ_{ij} und einem Fehler ϵ_{ijk}.
- Die μ_{ij}, die $I \times J$ Erwartungswerte sind dann die Parameter in der Means-Parameterisierung,

$$\boxed{Y_{ijk} = \mu_{ij} + \epsilon_{ijk}} \qquad i = 1, \ldots, I, j = 1, \ldots, J, k = 1, \ldots, n_{ij}$$

Effekt-Parameterisierung

Für die Effekt-Parameterisierung ersetzen wir die μ_{ij} durch die Summe aus

- Gesamtmittel μ
- Haupteffekte von A, α_i
- Haupteffekte von B, β_j
- Interaktionseffekte $(\alpha\beta)_{ij}$

Das Modell schreiben wir dann

$$\boxed{Y_{ijk} = \underbrace{\mu + \alpha_i + \beta_j + (\alpha\beta)_{ij}}_{\mu_{ij}} + \epsilon_{ijk}}, \qquad i = 1, \ldots, I, j = 1, \ldots, J, k = 1, \ldots, n_{ij}$$

Anzahl Parameter

- Das Modell hat $1 + I + J + (IJ)$ Parameter, also mehr Parameter als erlaubt.
- Modell identifizierbar machen, z.B. mit den sum-to-zero-Bedingungen:
  - $\sum_i \alpha_i = 0$
  - $\sum_j \beta_j = 0$
  - $\sum_i (\alpha\beta)_{ij} = 0$ für alle j
  - $\sum_j (\alpha\beta)_{ij} = 0$ für alle i
- Durch die Zusatzbedingungen haben wir dann $1 + (I-1) + (J-1) + (I-1)(J-1) = I \times J$ Parameter.

Effekt-Parameterisierung in R

In R ist standardmässig nicht die sum-to-zero Parameterisierung eingestellt, sondern die Referenzgruppe-Parameterisierung:

- $\alpha_1 = 0$
- $\beta_1 = 0$
- $(\alpha\beta)_{1j} = 0$ für alle j
- $(\alpha\beta)_{i1} = 0$ für alle i

Prinzip der Marginalität

- Die Faktoren A und B haben einen Interaktionseffekt, wenn die erwartete Veränderung in der Zielgrösse bei einer Veränderung der Stufe auf dem Faktor A von der Stufe des Faktors B abhängt, und umgekehrt.
- Wenn jede Veränderung der Stufe des Faktors A über alle Stufen vom Faktor B konstant ist, dann haben die beiden Faktoren keinen Interaktionseffekt, sie agieren additiv.
- Die Haupteffekte sind dann wohldefiniert und diese zu testen macht Sinn.

Prinzip der Marginalität

- Das Testen von Haupteffekten in Gegenwart von Interaktionseffekten verletzt das Prinzip der Marginalität.
- Es ist gemäss diesem Prinzip falsch, Haupteffekte zu interpretieren in der Gegenwart von Interaktionseffekten, oder Haupteffekte aus einem Modell zu entfernen, wenn Interaktionseffekte signifikant sind.

Prinzip der Marginalität*

- In R `anova(mod)` sind standardmässig "Type I Sum of Squares" eingestellt (sequentielle Tests!)
  - $SS(A)$ for factor A
  - $SS(A, B) - SS(A)$ for factor B
  - $SS(AB, A, B) - SS(A, B)$ for interaction AB
- "Type III Sum of Squares" (`car::Anova(mod,type=3)`) verletzen z.T. standardmässig das Prinzip.
  - $SS(A, B, AB) - SS(B, AB)$ for factor A
  - $SS(A, B, AB) - SS(A, AB)$ for factor B
  - Für Type III Sum of Squares: Setze sum-to-zero für coding matrix: `options(contrasts=c("contr.sum","contr.poly"))`, sonst wird Unsinn berechnet!
- Bei balancierten Daten geben beide SS dasselbe Resultat.

Beispiel: Simulation Daten aus einem 3×2 zweifaktoriellen Design.*

```r
set.seed(22)
nage <- 3  ## Anzahl Kategorien Altersgruppe
ntherapy <- 2  ## Anzahl Kategorien Therapieart
n <- 200  ## Totale Sample Size
age <- factor(sample(c("child", "young", "old"), size = n, replace = TRUE, prob = c(1, 2, 3)), levels = c("child", "young", "old"))
therapy <- factor(sample(c("Ctrl", "Trt"), size = n, replace = TRUE))
beta1 <- 40  ## Referenz
betaAge <- c(10, 20)  ## betaAge2 und betaAge3
betaTr <- c(10)  ## betaTreat
alphabeta <- c(-0, 0)  ## youngTrt und oltTrt, Interaktionseffekte=0
parameter <- c(beta1, betaAge, betaTr, alphabeta)  ## Wahrer Parametervektor
sigma <- 12  ## Noise SD
epsilon <- rnorm(n, 0, sigma)  ## Fehler
X <- model.matrix(~age * therapy)  ## Design-Matrix
response <- as.numeric(X %*% parameter + epsilon)  ## Y=Xbeta+epsilon, Daten ziehen aus Modell
d.cat2 <- data.frame(response, age, therapy)
```

Daten

Wir haben nun einen Datensatz `d.cat2` mit $n = 200$ Beobachtungen auf den Faktoren age (dreiwertig) und `therapy` (zweiwertig.)

```
str(d.cat2)

## 'data.frame': 200 obs. of  3 variables:
##  $ response: num  78.5 45.6 11.1 51.5 20.6 ...
##  $ age     : Factor w/ 3 levels "child","young",..: 3 3 1 2 1 2 2 2 3 3 ...
##  $ therapy : Factor w/ 2 levels "Ctrl","Trt": 2 2 1 2 1 2 2 2 2 2 ...

head(d.cat2)

##   response   age therapy
## 1     78.5   old     Trt
## 2     45.6   old     Trt
## 3     11.1 child    Ctrl
## 4     51.5 young     Trt
## 5     20.6 child    Ctrl
## 6     43.4 young     Trt
```

Ziel

- Wir wollen den Effekt von `therapy` auf die Zielgrösse quantifizieren
- und für age kontrollieren, wenn nötig.

Beschreiben

- Tabellarisch: `table()`, `xtabs()`, `by()`, `aggregate()`, `tapply()`
- Design

```
addmargins(xtabs(~therapy + age))

##        age
## therapy child young old Sum
##    Ctrl    20    25  54  99
##    Trt     10    34  57 101
##    Sum     30    59 111 200
```

- Graphisch: `interaction.plot()` (Abbildung 1).
- Die Altersgruppen innerhalb der Treatmentgruppen sind nicht gleich gross.
- Die graphische Darstellung deutet auf keinen Interaktionseffekt hin.

Beschreiben

```
interaction.plot(x.factor = age, trace.factor = therapy, response = response, trace.label = "treatment", xlab = "age group", ylab = "r
```

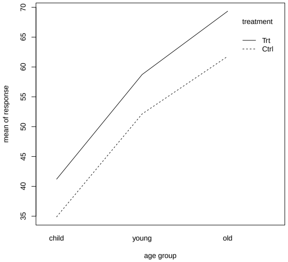

Abbildung: Beobachtete Durchschnitte bezüglich Faktoren age und therapy.

Beschreiben

Wir beschreiben die Daten pro Gruppe auf `therapy` und dann pro Kombination auf `therapy` und `age`

```
psych::describeBy(response, list(therapy), mat = TRUE)

##     item group1 vars   n mean   sd median trimmed  mad  min  max range   skew kurtosis   se
## X11    1   Ctrl    1  99 53.9 16.3   54.4    54.9 15.6 -4.0 99.7 103.7 -0.617    1.386 1.64
## X12    2    Trt    1 101 63.0 14.3   64.2    63.4 15.8 21.6 93.3  71.6 -0.326   -0.273 1.43

## by(response,therapy,psych::describe) ##Alternative, weniger kompakt
```

```
psych::describeBy(response, list(therapy, age), mat = TRUE)

##     item group1 group2 vars  n mean   sd median trimmed  mad  min  max range    skew kurtosis   se
## X11    1   Ctrl  child    1 20 34.9 16.5   38.0    36.3 16.4 -4.0 56.6  60.6 -0.6681   -0.584 3.69
## X12    2    Trt  child    1 10 41.2 11.8   44.4    41.5 13.7 21.6 58.4  36.8 -0.1773   -1.375 3.72
## X13    3   Ctrl  young    1 25 52.1 11.7   50.9    51.9 15.2 34.1 71.6  37.5  0.2074   -1.326 2.33
## X14    4    Trt  young    1 34 58.7 12.4   55.2    58.3 13.9 40.9 81.5  40.6  0.3188   -1.273 2.12
## X15    5   Ctrl    old    1 54 61.8 11.5   60.9    61.4 11.6 38.6 99.7  61.2  0.5691    0.929 1.56
## X16    6    Trt    old    1 57 69.3 10.9   69.3    69.5 11.2 45.6 93.3  47.7 -0.0521   -0.341 1.44

## by(response,list(therapy,age),psych::describe) ##Alternative, weniger kompakt
```

Beschreiben

Wir fassen die 6 Durchschnitte $\hat{\mu}_{ij}$ zusammen mit den

- ungewichteten Mittel (Durchschnitte der Durchschnitte)
- gewichteten Mittel (Durchschnitte in Therapiegruppen unabhängig von age).

	child	young	old	unweighted.mean	weighted.mean
Ctrl	34.89	52.13	61.78	49.60	53.91
Trt	41.20	58.74	69.35	56.43	62.99

Tabelle: Ungewichtete und Gewichtete Mittel

“Durchschnitte”

- Die beiden Durchschnitte (je für `Ctrl` und `Trt`) in den beiden letzten Kolonnen sind nur gleich, wenn das Design balanciert ist, wenn die Altersgruppen gleich gross sind, im Allgemeinen ist das aber – wie hier – nicht der Fall.
- Wenn der Durchschnitt von allen in `Ctrl` (`Trt`) die Kontrollgruppe (Treatmentgruppe) repräsentiert (das gewichtete Mittel), dann testen wir den Effekt von `therapy` mit den Werten auf `age` in unserer Stichprobe, d.h. wir ignorieren die Variable `age`.
- Wenn wir die Mittelwerte der Mittelwerte brauchen, dann testen wir den Effekte von `therapy` kontrolliert für `age`.

Modell anpassen

- Wir passen jetzt ein Modell an mit Haupt- und Interaktionseffekten.
- Die Modell-Formel kann man auf zwei verschiedene Arten eingeben, mit

```
## ~A + B + A:B
```

oder mit

```
## ~A * B
```

```
modelInt <- lm(response ~ therapy * age, data = d.cat2)
summary(modelInt)

##
## Call:
## lm(formula = response ~ therapy * age, data = d.cat2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -38.89  -8.48  -0.72   8.07  37.93
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           34.889      2.702   12.91  < 2e-16
## therapyTrt             6.314      4.681    1.35     0.18
## ageyoung              17.245      3.626    4.76  3.8e-06
## ageold                26.893      3.164    8.50  5.0e-15
## therapyTrt:ageyoung    0.288      5.661    0.05     0.96
## therapyTrt:ageold      1.251      5.213    0.24     0.81
##
## Residual standard error: 12.1 on 194 degrees of freedom
## Multiple R-squared:  0.442,Adjusted R-squared:  0.428
## F-statistic: 30.7 on 5 and 194 DF,  p-value: <2e-16
```

Effekt-Parameterisierung in R

`child` und `Ctrl` sind die beiden Referenzkategorien. Relativ zu diesen sind dann alle Effekte zu interpretieren.

	child	young	old
Control	μ_{11}	μ_{12}	μ_{13}
Treatment	μ_{21}	μ_{22}	μ_{23}

- `(Intercept)`: Geschätzter Erwartungswert für `child` in `Ctrl`: $\hat{\mu}_{11}$
- `therapyTrt`: Geschätzter erwarteter Abstand von `Trt` zu `Ctrl` für `child`: $\hat{\mu}_{21} - \hat{\mu}_{11}$
- `ageyoung`: Geschätzter erwarteter Abstand von `young` zu `child` für `Ctrl`: $\hat{\mu}_{12} - \hat{\mu}_{11}$
- `ageold`: Geschätzter erwarteter Abstand von `old` zu `child` für `Ctrl`: $\hat{\mu}_{13} - \hat{\mu}_{11}$
- `therapyTrt:ageyoung`: Geschätzter erwarteter Unterschied im Unterschied `Trt-Ctrl` für `young` versus `child`: $\hat{\mu}_{22} - \hat{\mu}_{12} - (\hat{\mu}_{21} - \hat{\mu}_{11})$
- `therapyTrt:ageold`: Geschätzter erwarteter Unterschied im Unterschied `Trt-Ctrl` für `old` versus `child`: $\hat{\mu}_{23} - \hat{\mu}_{13} - (\hat{\mu}_{21} - \hat{\mu}_{11})$

Interaktionen

- Interaktionen sind **Unterschiede von Unterschieden**. Wenn vorhanden, dann haben wir nicht-parallele Linien im Interaktionsplot.
- Punktschätzungen für den Therapieeffekt hängen vom Alter ab. Dieser ist
  - 6.314 für `child`
  - $6.314 + 0.288$ für `young`
  - $6.314 + 1.251$ für `old`
- Der Durchschnitt von diesen drei Effekten ist gleich dem Unterschied der ungewichteten Mittel in Tabelle 1.
- Man kann den Therapieeffekt (Punktschätzung) auch kurz und knackig mit Indikatorvariablen schreiben:

$$6.314 + 0.288 \times I_{young} + 1.251 \times I_{old}$$

Testen von Hypothesen

Wir passen nun auch ein Modell an ohne den Interaktionseffekt und die beiden Modelle mit nur einer Eingangsgrösse:

```
(modelMain <- update(modelInt, ~. - therapy:age))

##
## Call:
## lm(formula = response ~ therapy + age, data = d.cat2)
##
## Coefficients:
## (Intercept)   therapyTrt     ageyoung       ageold
##       34.62         7.11        17.22        27.39
```

```
(modelTherapy <- update(modelMain, ~. - age))

##
## Call:
## lm(formula = response ~ therapy, data = d.cat2)
##
## Coefficients:
## (Intercept)   therapyTrt
##       53.91         9.08
```

```
(modelAge <- update(modelMain, ~. - therapy))

##
## Call:
## lm(formula = response ~ age, data = d.cat2)
##
## Coefficients:
## (Intercept)     ageyoung       ageold
##        37.0         18.9         28.7
```

Testen vom Interaktionseffekt

Ist der Effekt von `therapy` abhängig von der Altersgruppe? `anova()` oder `drop1()`.

```
anova(modelMain, modelInt)  ## explizit über Modellvergleich

## Analysis of Variance Table
##
## Model 1: response ~ therapy + age
## Model 2: response ~ therapy * age
##   Res.Df   RSS Df Sum of Sq    F Pr(>F)
## 1    196 28351
## 2    194 28337  2      13.7 0.05   0.95
```

```
anova(modelInt)  ## geht auch, da sequentiell und INTERAKTIONSEFFEKT AM SCHLUSS

## Analysis of Variance Table
##
## Response: response
##              Df Sum Sq Mean Sq F value    Pr(>F)
## therapy       1   4117    4117   28.19 0.0000003
## age           2  18313    9157   62.69   < 2e-16
## therapy:age   2     14       7    0.05      0.95
## Residuals   194  28337     146
```

```
drop1(modelInt, test = "F")  ## drop1 respektiert das Prinzip der Marginalität, Haupteffekte werden nicht getestet.

## Single term deletions
##
## Model:
## response ~ therapy * age
##             Df Sum of Sq   RSS  AIC F value Pr(>F)
## <none>                   28337 1003
## therapy:age  2      13.7 28351  999    0.05   0.95
```

Testen vom Haupteffekt

- Der Interaktionseffekt ist nicht signifikant, d.h. wir können das Modell ohne Interaktion nicht verwerfen.
- Wir können jetzt Hypothesen testen bezüglich den Haupteffekten.
- Wir testen den Effekt von `therapy`, kontrolliert für age:

```
anova(modelAge, modelMain)  ## als Modellvergleich

## Analysis of Variance Table
##
## Model 1: response ~ age
## Model 2: response ~ therapy + age
##   Res.Df   RSS Df Sum of Sq    F   Pr(>F)
## 1    197 30817
## 2    196 28351  1      2467 17.1 0.000054
```

```
anova(lm(response ~ age + therapy, data = d.cat2))  ## als sequentielle Anova

## Analysis of Variance Table
##
## Response: response
##            Df Sum Sq Mean Sq F value   Pr(>F)
## age         2  19964    9982    69.0  < 2e-16
## therapy     1   2467    2467    17.1 0.000054
## Residuals 196  28351     145
```

Sequentielle Tests: Reihenfolge beachten

Achtung, die Reihenfolge ist wichtig, **wenn man nur das grosse Modell eingibt** in `anova()`. Dazu das Modell ohne Interaktion (Haupteffektmodel) mit anderer Reihenfolge:

```
anova(lm(response ~ therapy + age, data = d.cat2))  ## Haupteffektmodel mit neuer Reihenfolge.

## Analysis of Variance Table
##
## Response: response
##            Df Sum Sq Mean Sq F value     Pr(>F)
## therapy     1   4117    4117    28.5 0.00000026
## age         2  18313    9157    63.3    < 2e-16
## Residuals 196  28351     145
```

- Nicht-kontrollierter Effekt von `therapy`
- Effekt von age, kontrolliert für `therapy`

Welche Residualvarianz?*

- Da der Interaktionseffekt nicht signifikant ist, sollten wir auch das Haupteffektmodell brauchen, um den für Alter kontrollierten Therapieeffekt zu testen.
- Bei Abwesenheit von Interaktion sollte die Varianz, die zum Interaktionseffekt gehört, dem Zufall zugeordnet werden.

Grösse des für age kontrollierten Effekts von `therapy`

- Der Therapieeffekt ist also unabhängig von der Altersgruppe.
- Das war auch im Interaktionsplot ersichtlich. Die Profile sind annähernd parallel.
- age bleibt aber vorerst im Modell als potentieller Confounder. Der für Alter kontrollierte Effekt von `therapy` (der über den Effekt von age hinausgehende Effekt) ist 7.109 (Punktschätzung):

```
summary(modelMain)$coef

##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    34.62       2.27   15.26 3.73e-35
## therapyTrt      7.11       1.72    4.13 5.38e-05
## ageyoung       17.22       2.73    6.31 1.83e-09
## ageold         27.39       2.49   10.98 3.55e-22
```

Nicht kontrollierter Effekt von `therapy`

- Der nicht für age kontrollierte Effekt von `therapy` ist 9.08 (Punktschätzung) und entspricht dem Unterschied der gewichteten Durchschnitte in Tabelle 1.
- Hier wird die Variable age in der Analyse ignoriert.

```
summary(modelTherapy)$coef

##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    53.91       1.54   34.94 1.29e-86
## therapyTrt      9.08       2.17    4.18 4.38e-05
```

- was äquivalent wäre mit

```
t.test(response ~ therapy, var.equal = TRUE)

##
##  Two Sample t-test
##
## data:  response by therapy
## t = -4, df = 198, p-value = 0.00004
## alternative hypothesis: true difference in means between group Ctrl and group Trt is not equal to 0
## 95 percent confidence interval:
##  -13.36  -4.79
## sample estimates:
## mean in group Ctrl  mean in group Trt
##               53.9               63.0
```

Kontraste

- Kontraste sind Linearkombinationen von Parametern mit der Bedingung, dass die Summe der Koeffizienten Null ist. Mit Konstanten $a_1, a_2, \ldots, a_t$ und Parametern $\theta_1, \theta_2, \ldots, \theta_t$ ist ein Kontrast also

$$L = \sum_{i=1}^{t} a_i \theta_i, \qquad \text{mit} \sum_{i=1}^{t} a_i = 0.$$

- Mit `emmeans()` kann man die Zwischengruppenunterschiede für alle Altersgruppen berechnen, jetzt aber mit Konfidenzintervallen und Tests.
- Bei Abwesenheit von Interaktionseffekten sind hier die Therapieeffekte für alle Altersgruppen gleich.

Treatment effect unadjusted

```
em <- emmeans(modelTherapy, revpairwise ~ therapy)
summary(em, infer = c(TRUE, TRUE))  ## damit mit CIs und p-werten

## $emmeans
##  therapy emmean   SE  df lower.CL upper.CL t.ratio p.value
##  Ctrl      53.9 1.54 198     50.9       57  34.900  <.0001
##  Trt       63.0 1.53 198     60.0       66  41.200  <.0001
##
## Confidence level used: 0.95
##
## $contrasts
##  contrast   estimate   SE  df lower.CL upper.CL t.ratio p.value
##  Trt - Ctrl     9.08 2.17 198     4.79     13.4   4.180  <.0001
##
## Confidence level used: 0.95
```

Treatment effect adjusted for age

```
em <- emmeans(modelMain, revpairwise ~ therapy)
summary(em, infer = c(TRUE, TRUE))  ## damit mit CIs und p-werten

## $emmeans
##  therapy emmean   SE  df lower.CL upper.CL t.ratio p.value
##  Ctrl      49.5 1.27 196       47     52.0  38.900  <.0001
##  Trt       56.6 1.33 196       54     59.2  42.500  <.0001
##
## Results are averaged over the levels of: age
## Confidence level used: 0.95
##
## $contrasts
##  contrast   estimate   SE  df lower.CL upper.CL t.ratio p.value
##  Trt - Ctrl     7.11 1.72 196     3.71     10.5   4.130  0.0001
##
## Results are averaged over the levels of: age
## Confidence level used: 0.95
```

Prinzip der Marginalität! (Note!)

```
em <- emmeans(modelInt, revpairwise ~ therapy)

## NOTE: Results may be misleading due to involvement in interactions

summary(em, infer = c(TRUE, TRUE))  ## damit mit CIs und p-werten

## $emmeans
##  therapy emmean   SE  df lower.CL upper.CL t.ratio p.value
##  Ctrl      49.6 1.33 194     47.0     52.2  37.400  <.0001
##  Trt       56.4 1.54 194     53.4     59.5  36.500  <.0001
##
## Results are averaged over the levels of: age
## Confidence level used: 0.95
##
## $contrasts
##  contrast   estimate   SE  df lower.CL upper.CL t.ratio p.value
##  Trt - Ctrl     6.83 2.04 194     2.81     10.8   3.350  0.0010
##
## Results are averaged over the levels of: age
## Confidence level used: 0.95
```

Zwischengruppenunterschiede pro Altersgruppe

```
em <- emmeans(modelInt, revpairwise ~ therapy | age)
summary(em, infer = c(TRUE, TRUE))  ## damit mit CIs und p-werten

## $emmeans
## age = child:
##  therapy emmean   SE  df lower.CL upper.CL t.ratio p.value
##  Ctrl      34.9 2.70 194     29.6     40.2  12.900  <.0001
##  Trt       41.2 3.82 194     33.7     48.7  10.800  <.0001
##
## age = young:
##  therapy emmean   SE  df lower.CL upper.CL t.ratio p.value
##  Ctrl      52.1 2.42 194     47.4     56.9  21.600  <.0001
##  Trt       58.7 2.07 194     54.6     62.8  28.300  <.0001
##
## age = old:
##  therapy emmean   SE  df lower.CL upper.CL t.ratio p.value
##  Ctrl      61.8 1.64 194     58.5     65.0  37.600  <.0001
##  Trt       69.3 1.60 194     66.2     72.5  43.300  <.0001
##
## Confidence level used: 0.95
##
## $contrasts
## age = child:
##  contrast   estimate   SE  df lower.CL upper.CL t.ratio p.value
##  Trt - Ctrl     6.31 4.68 194   -2.918     15.6   1.350  0.1790
##
## age = young:
##  contrast   estimate   SE  df lower.CL upper.CL t.ratio p.value
##  Trt - Ctrl     6.60 3.18 194    0.321     12.9   2.070  0.0395
##
## age = old:
##  contrast   estimate   SE  df lower.CL upper.CL t.ratio p.value
##  Trt - Ctrl     7.56 2.30 194    3.038     12.1   3.300  0.0012
##
## Confidence level used: 0.95
```

Predictions*

```
prInt <- effects::predictorEffects(modelInt, partial.residuals = TRUE)
prMain <- effects::predictorEffects(modelMain, partial.residuals = TRUE)
# plot(prInt,lines=list(multiline=TRUE))
plot(prInt, partial.residual = list(pch = "*", col = "#FF00FF80"))
plot(prMain, partial.residual = list(pch = "*", col = "#FF00FF80"))
```