LM: ANCOVA

Kolloquium für Statistik

Departement of Health Professions
Bern University of Applied Sciences

7. November 2023

Eine kategorielle und eine kontinuierliche Eingangsgrösse

Im Folgenden haben wir

- Eine kategorielle Eingangsgrösse: `group` (mit zwei Kategorien A und B)
- Eine kontinuierliche Eingangsgrösse: `height`.
- Die Zielgrösse, die abhängige Variable: `weight`.

Ziel

- Uns interessiert der Effekt von `group` auf `weight`, und wenn nötig, soll dieser für `height` kontrolliert werden.
- Klassisch werden solche Situationen Kovarianzanalysen genannt.

Means-Parameterisierung

- Die Beobachtung Y_i (`weight`) der Person i in der Gruppe j (`group`) mit Kovariable x_i (`height`) schreiben wir dann

$$\boxed{Y_i = \alpha_{j(i)} + \beta_{j(i)}\, x_i + \epsilon_i}, \qquad j = 1, 2, \quad i = 1, \ldots, n.$$

- $\alpha_{j(i)}$ ist das Intercept und $\beta_{j(i)}$ die Steigung für den `height-weight` Zusammenhang in Gruppe j.
- x_i ist die Körpergrösse der Person i.
- ϵ_i beschreibt den kombinierten Effekt aller unbekannten Einflüsse auf das Körpergewicht von Person i mit $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$.

Means-Parameterisierung

- Das Modell hat also (neben σ^2) vier Parameter (2 Gruppen mal 2 Parameter).
- Wenn die Steigungen in den zwei Gruppen verschieden sind, haben wir einen Interaktionseffekt von `height` und `group` auf `weight`.

Effekt-Parameterisierung

$$\boxed{Y_i = \beta_1 + \beta_2\, I_{B(i)} + \beta_3\, x_i + \beta_4\, x_i\, I_{B(i)} + \epsilon_i}.$$

$I_{B(i)}$ ist eine Indikatorvariable für die Zugehörigkeit zu Gruppe B

$$I_{B(i)} = \begin{cases} 1 & \text{group } B \\ 0 & \text{group } \neq B. \end{cases}$$

- β_1 ist das erwartete Gewicht für eine Person in Gruppe A und Körpergrösse Null.
- β_2 ist der erwartete Unterschied im Gewicht für eine Person in Gruppe B relativ zu Gruppe A bei Körpergrösse Null
- β_3 ist die erwartete Steigung der Regression von `weight` auf `height` in Gruppe A.
- β_4 ist der erwartete Unterschied in den Steigungen in der Gruppe B relativ zur Gruppe A.

Effekt-Parameterisierung

Die Erwartungswerte in den zwei Gruppen sind

$$\mathrm{E}(Y_i; \mathrm{group} = A) = \beta_1 + \beta_3\, x_i$$
$$\mathrm{E}(Y_i; \mathrm{group} = B) = \beta_1 + \beta_2 + \beta_3\, x_i + \beta_4\, x_i$$

Die Erwartungswerte in den zwei Gruppen bei Gewicht Null, sind

$$\mathrm{E}(Y_i \mid \mathrm{group} = A, x_i = 0) = \beta_1$$
$$\mathrm{E}(Y_i \mid \mathrm{group} = B, x_i = 0) = \beta_1 + \beta_2$$

Simulation*

Folgender Code simuliert Daten aus dem beschriebenen Modell.

```
set.seed(10)
n.groups <- 2
n.sample <- 50
n <- n.groups * n.sample  ##sample size
ind <- rep(1:n.groups, each = n.sample)  ##Indicator for group
group <- factor(ind, labels = c("A", "B"))
height <- rnorm(n, mean = 165, sd = 11.4)
covariates <- data.frame(group, height)
Xeffects <- model.matrix(~group * height)
Xmeans <- model.matrix(~group * height - height - 1)
sigma <- 2
betaM <- c(muA <- -36.475, muB <- -45.5, slopeA <- 0.615, slopeB <- 0.7)  ##Means-Param.
betaE <- c(muA, muB - muA, slopeA, slopeB - slopeA)  ##Effekt-Parm
lin.pred <- Xeffects %*% betaE
lin.pred2 <- Xmeans %*% betaM
# all.equal(lin.pred,lin.pred2) ## ist dasselbe
eps <- rnorm(n = n, mean = 0, sd = sigma)  ## add noise
weight <- lin.pred + eps  ## Zielgrösse
d.catcont <- data.frame(group, height, weight)
```

Daten

- Wir haben nun einen Datensatz `d.catcont`
- $n = 100$ Beobachtungen auf `weight` und mit unabhängigen Variablen `height` und `group` (zweiwertig)

```
str(d.catcont)
```

```
## 'data.frame': 100 obs. of  3 variables:
##  $ group : Factor w/ 2 levels "A","B": 1 1 1 1 1 1 1 1 1 1 ...
##  $ height: num  165 163 149 158 168 ...
##  $ weight: num  63.6 64.5 53.3 62.2 65.8 ...
```

```
head(d.catcont)
```

```
##   group height weight
## 1     A    165   63.6
## 2     A    163   64.5
## 3     A    149   53.3
## 4     A    158   62.2
## 5     A    168   65.8
## 6     A    169   68.9
```

Beschreiben

```
psych::describe(d.catcont[, -1])
```

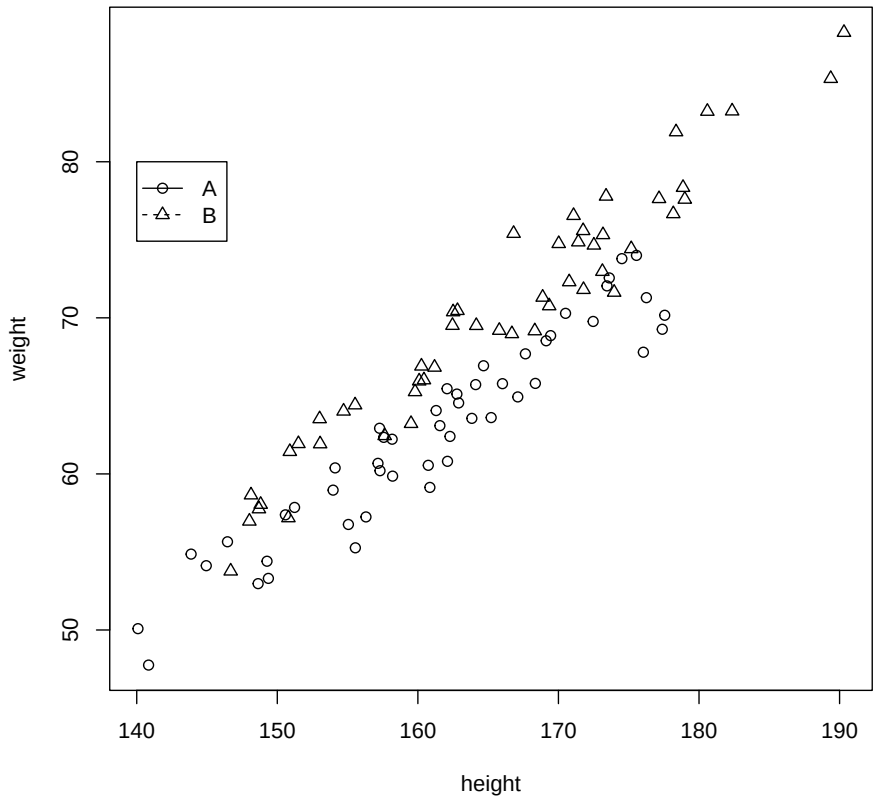
```
##        vars   n  mean    sd median trimmed   mad   min   max range skew kurtosis   se
## height    1 100 163.4 10.73  162.8   163.4 12.13 140.1 190.3  50.2 0.03    -0.56 1.07
## weight    2 100  66.4  8.08   65.8    66.2  8.17  47.8  88.3  40.5 0.22    -0.26 0.81
```

```
by(d.catcont[, -1], d.catcont$group, psych::describe)
```

```
## d.catcont$group: A
##        vars  n  mean   sd median trimmed  mad   min max range skew kurtosis   se
## height    1 50 161.1 9.88    162   161.4 9.86 140.1 178  37.5 -0.2    -0.75 1.40
## weight    2 50  62.7 6.32     63    62.8 7.02  47.8  74  26.2 -0.2    -0.70 0.89
## ------------------------------------------------------------------------------------------------------
## d.catcont$group: B
##        vars  n  mean    sd median trimmed   mad   min   max range skew kurtosis   se
## height    1 50 165.8 11.13  166.8     166 10.50 146.7 190.3  43.6 0.06    -0.81 1.57
## weight    2 50  70.1  7.96   70.4      70  8.28  53.8  88.3  34.5 0.06    -0.61 1.13
```

Beschreiben

```
plot(weight ~ height, data = d.catcont, pch = as.numeric(group))
legend(140, 80, legend = levels(group), lty = c(1, 2), pch = c(1, 2))
```



Model anpassen

- Die Grössen β_1 und $\beta_1 + \beta_2$ stellen das Intercept dar für Gruppe A beziehungsweise Gruppe B, das erwartete Gewicht bei Körpergrösse Null.
- Damit das Intercept besser interpretierbar ist, macht es Sinn, die Variable `height` vorab zu zentrieren.
- Das heisst, dass man von jedem Wert den Mittelwert subtrahiert. Das wird nur eine Auswirkung haben auf die Interpretation des Intercept.
- `scale()`

```
d.catcont$heightCent <- scale(d.catcont$height, scale = FALSE)
```

Model anpassen

Wir passen das Modell an, zuerst mit nicht-, dann mit zentrierter Körpergrösse.

```
modraw <- lm(weight ~ group * height, d.catcont)
summary(modraw)$coef
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -35.5185     4.5592   -7.79 7.86e-12
## groupB         -9.3987     6.1742   -1.52 1.31e-01
## height          0.6094     0.0282   21.57 1.36e-38
## groupB:height   0.0845     0.0378    2.24 2.75e-02
```

```
mod <- lm(weight ~ group * heightCent, d.catcont)
summary(mod)$coef
```

```
##                   Estimate Std. Error t value  Pr(>|t|)
## (Intercept)        64.0776     0.2839  225.67 1.22e-132
## groupB              4.4191     0.4004   11.04  9.01e-19
## heightCent          0.6094     0.0282   21.57  1.36e-38
## groupB:heightCent   0.0845     0.0378    2.24  2.75e-02
```

Model anpassen

```
plot(weight ~ height, data = d.catcont, pch = as.numeric(group))
legend(140, 80, legend = levels(group), lty = c(1, 2), pch = c(1, 2))
abline(a = coef(modraw)[1], b = coef(modraw)[3], lty = 1)
abline(a = coef(modraw)[1] + coef(modraw)[2], b = coef(modraw)[3] + coef(modraw)[4], lty = 2)


plot(weight ~ heightCent, data = d.catcont, pch = as.numeric(group))
legend(-20, 80, legend = levels(group), lty = c(1, 2), pch = c(1, 2))
abline(a = coef(mod)[1], b = coef(mod)[3], lty = 1)
abline(a = coef(mod)[1] + coef(mod)[2], b = coef(mod)[3] + coef(mod)[4], lty = 2)
```

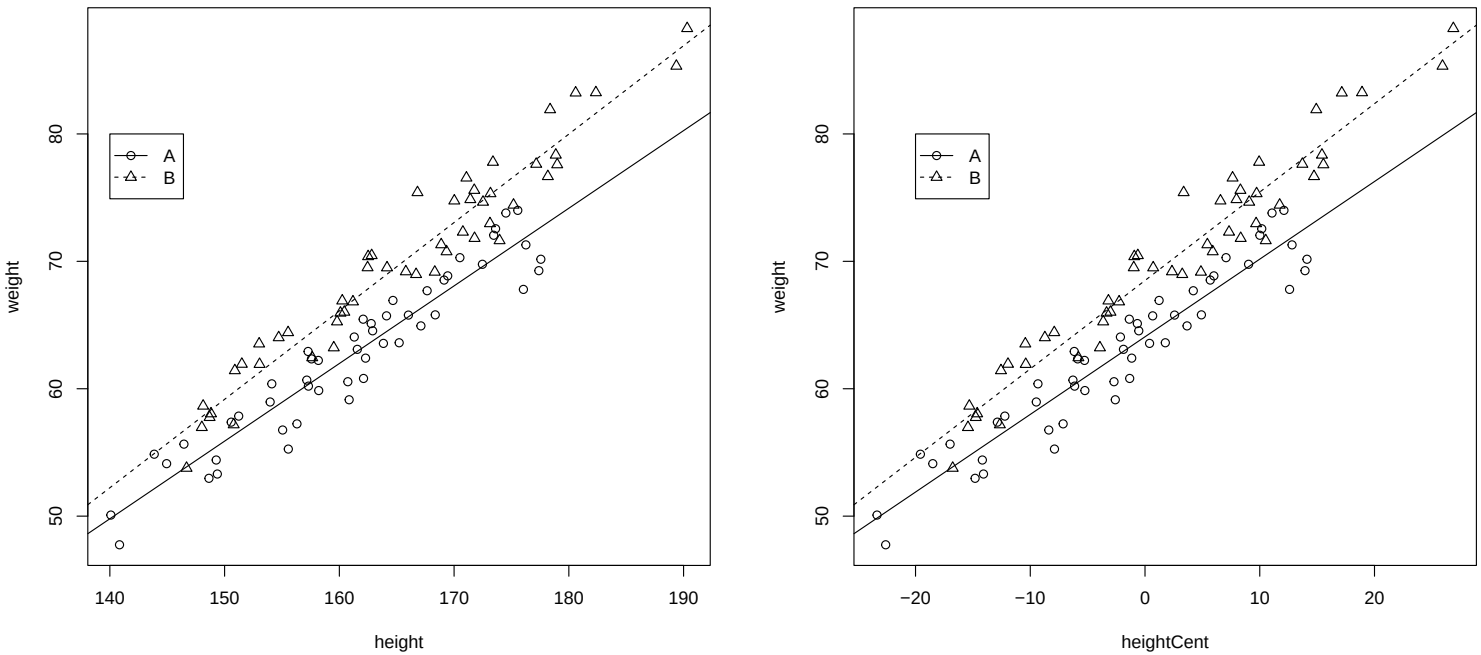


Abbildung: Angepasstes Modell. Rechts: mit zentrierter Körpergrösse

Interpretation

- $\hat{\beta}_1 = 64.078$ ist jetzt eine Schätzung für das erwartete Gewicht bei Durchschnittsgrösse für eine Person in Gruppe A
- $\hat{\beta}_2 = 4.419$ ist die Schätzung für die Differenz im erwarteten Gewicht bei Durchschnittsgrösse zwischen Gruppe B und Gruppe A
- $\hat{\beta}_3 = 0.609$ ist die Schätzung für die Steigung für eine Person in Gruppe A
- $\hat{\beta}_4 = 0.085$ ist die Schätzung für den Unterschied in der Steigung für eine Person aus der Gruppe B relativ zu einer Person in der Gruppe A

Interaktionseffekt

- Wir wollen jetzt testen, ob der Interaktionseffekt signifikant ist.
- Wir sehen das eigentlich schon im `summary()`-Output.
- Wir machen aber diesen Test wieder explizit über `anova()` oder `drop1()`.

```
modMain <- update(mod, . ~ . - group:heightCent)
anova(modMain, mod)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ group + heightCent
## Model 2: weight ~ group * heightCent
##   Res.Df RSS Df Sum of Sq    F Pr(>F)
## 1     97 385
## 2     96 366  1      19.1 5.01  0.027
```

```
drop1(mod, test = "F")
```

```
## Single term deletions
##
## Model:
## weight ~ group * heightCent
##                  Df Sum of Sq RSS AIC F value Pr(>F)
## <none>                        366 138
## group:heightCent  1      19.1 385 141    5.01  0.027
```

Interaktionseffekt

- Das heisst, der Effekt von `group` auf `weight` hängt vom Wert auf `heightCent` ab (oder, was quantitativ äquivalent ist, aber nicht unsere Frage: Der Effekt von `heightCent` auf `weight` hängt von `group` ab).
- Das Modell ohne Interaktionseffekt wird verworfen.
- Der Interaktionseffekt ist damit signifikant.

Interaktionseffekt

- R macht sequentielle Tests (Type I-Quadratsummen).
- Es gibt nun Varianzanalysen mit Type III-Quadratsummen, die **Haupteffekte testen, kontrolliert für Interaktionseffekte.**
- Das sind nicht-sequentielle Tests, die aber umstritten sind und zu viel Verwirrung führen.
- Wir folgen dem Prinzip der Marginalität und gehen nicht weiter darauf ein.

Effekt von group, kontrolliert für heightCent

- Dazu müssen wir aber die Residualvarianz des Interaktionsmodells nehmen.
- Sequentielle ANOVA (Reihenfolge ist wichtig)

```
anova(lm(weight ~ heightCent * group, d.catcont))  ##richtig: F=120
```

```
## Analysis of Variance Table
##
## Response: weight
##                   Df Sum Sq Mean Sq F value Pr(>F)
## heightCent         1   5612    5612 1471.39 <2e-16
## group              1    460     460  120.59 <2e-16
## heightCent:group   1     19      19    5.01  0.027
## Residuals         96    366       4
```

```
anova(lm(weight ~ heightCent + group, d.catcont))  ##falsch: F=116
```

```
## Analysis of Variance Table
##
## Response: weight
##             Df Sum Sq Mean Sq F value Pr(>F)
## heightCent   1   5612    5612    1413 <2e-16
## group        1    460     460     116 <2e-16
## Residuals   97    385       4
```

Kontraste

- Zu den Intervallschätzungen kommen wir mit `confint()`.
- Damit wir auch eine Intervallschätzung für die Steigung in der Gruppe *B* haben (Punktschätzung 0.609+0.085), können wir die Funktion `emtrends()` aus dem Packet `emmeans` brauchen.

```
cbind(summary(mod)$coef, confint(mod))
```

```
##                   Estimate Std. Error t value  Pr(>|t|)    2.5 % 97.5 %
## (Intercept)        64.0776     0.2839  225.67 1.22e-132 63.51394 64.641
## groupB              4.4191     0.4004   11.04  9.01e-19  3.62425  5.214
## heightCent          0.6094     0.0282   21.57  1.36e-38  0.55329  0.665
## groupB:heightCent   0.0845     0.0378    2.24  2.75e-02  0.00959  0.159
```

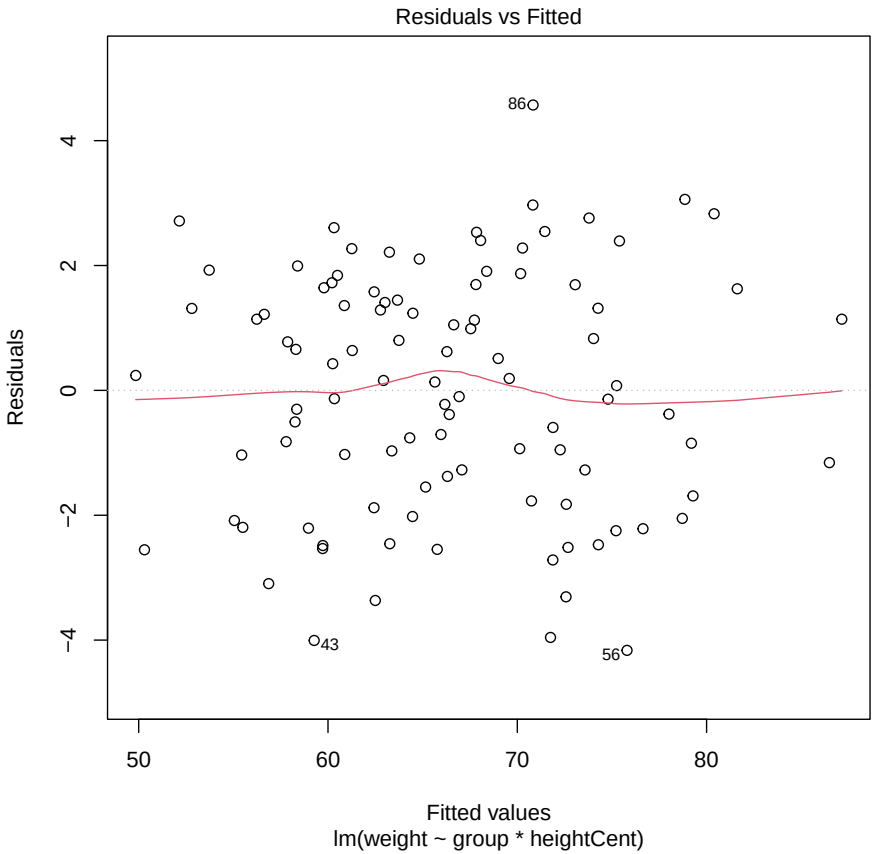
```
emmeans::emtrends(mod, revpairwise ~ group, var = "heightCent", infer = c(TRUE, TRUE))
```

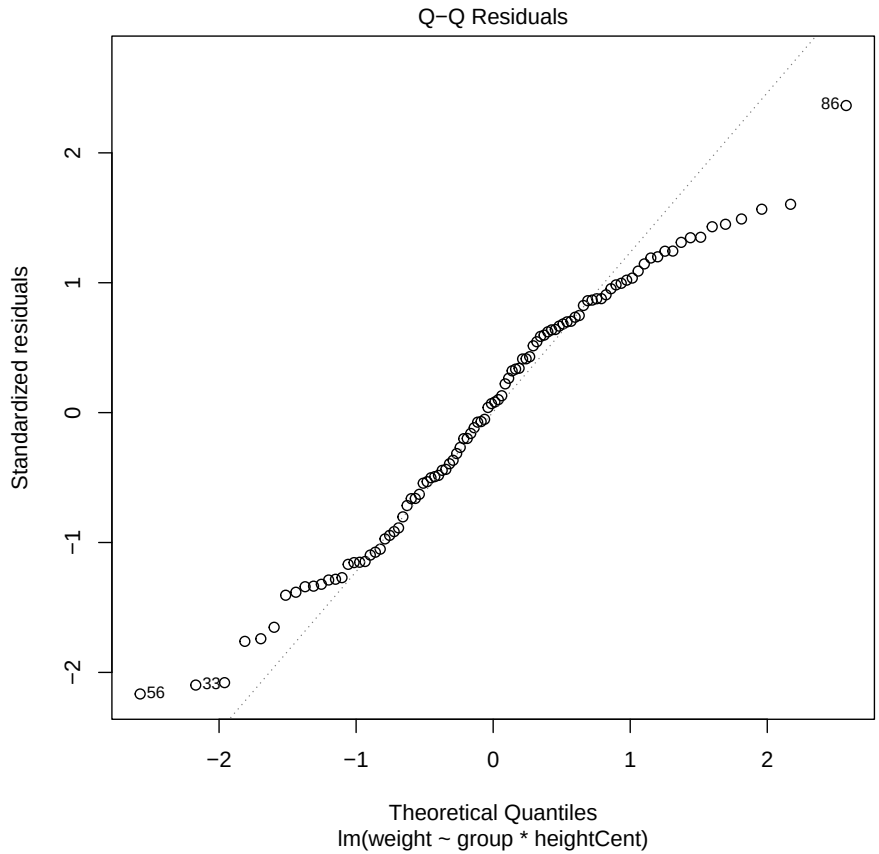
```
## $emtrends
##  group heightCent.trend     SE df lower.CL upper.CL t.ratio p.value
##  A                0.609 0.0283 96    0.553    0.665  21.570  <.0001
##  B                0.694 0.0251 96    0.644    0.744  27.690  <.0001
##
## Confidence level used: 0.95
##
## $contrasts
##  contrast estimate     SE df lower.CL upper.CL t.ratio p.value
##  B - A      0.0845 0.0378 96  0.00959     0.16   2.239  0.0275
##
## Confidence level used: 0.95
```

Residuenanalyse

Die Residuenanalyse sieht hier gut aus (TA-plot) (weil man ja aus einem Modell mit entsprechenden Annahmen simuliert hat). n ist relativ gross, daher ist der nicht sehr schöne QQ-Plot kein Problem.

```
plot(mod, which = c(1, 2))
```





Vorhersage

- Wir haben bis jetzt vor allem Parameter geschätzt und Hypothesen diesbezüglich getestet.
- In der Wissenschaft will man aber häufig ein angepasstes Modell dahingehend benutzen, um mit neuen Werten auf den Prädiktoren neue Werte auf der abhängigen Variablen vorherzusagen.
- Das ist in R implementiert mit der `predict()`-Funktion.

Vorhersage

- Es gibt zwei Arten von Vorhersagen, mittlere Vorhersagen oder individuelle Vorhersagen.
- Achtung: Es geht jetzt nicht mehr um Unsicherheit(en) für einzelne Parameter, sondern für erwartete oder individuelle Y's.
- Wir möchten nun mit unserem angepassten Modell `mod` für ausgewählte Körpergrössen und Gruppe neue Beobachtungen vorhersagen.

Vorhersage

```
new <- data.frame(group = c("A", "B", "A"), height = c(170, 180, 190))
new
```

```
##   group height
## 1     A    170
## 2     B    180
## 3     A    190
```

Zuerst machen wir eine Vorhersage für eine neue Beobachtung $Y_{new} \mid X_{new} = x_{new}$. Das machen wir mit dem Argument `interval="prediction"`.

```
pred <- predict(modraw, newdata = new, interval = "prediction")
cbind(new, pred)
```

```
##   group height  fit  lwr  upr
## 1     A    170 68.1 64.1 72.0
## 2     B    180 80.0 76.0 84.0
## 3     A    190 80.3 76.0 84.5
```

Vorhersage

Jetzt machen wir eine Vorhersage für den Erwartungswert $\mathrm{E}(Y_{new} \mid X_{new} = x_{new})$. Das machen wir mit dem Argument `interval="confidence"`.

```
pred2 <- predict(modraw, newdata = new, interval = "confidence")
cbind(new, pred2)
```

```
##   group height  fit  lwr  upr
## 1     A    170 68.1 67.3 68.8
## 2     B    180 80.0 79.1 80.9
## 3     A    190 80.3 78.6 82.0
```

- Die Unsicherheit für die Vorhersage einer individuellen Beobachtung ist immer grösser als die Unsicherheit für die Vorhersage eines Erwartungswertes.
- Im Gegensatz zu letzterem kommt bei der individuellen Vorhersage immer noch die Fehlervarianz σ^2 dazu.

Vorhersage

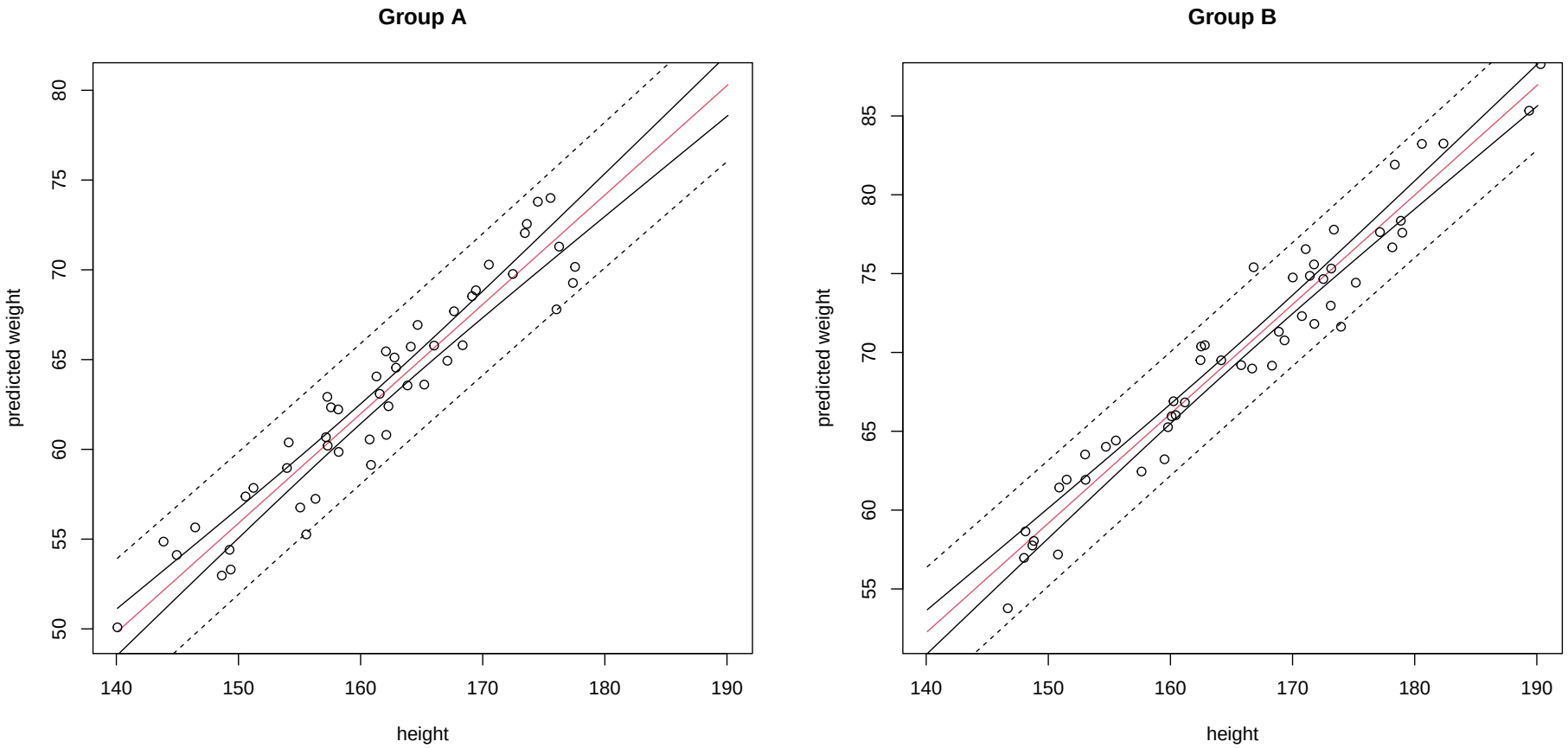


Abbildung: Vorhersage: 95% Konfidenzgrenzen für Erwartungswerte (durchgezogen) und 95% Vorhersagegrenzen für individuelle Beobachtungen (gestrichelt).

Vorhersage

- Die Unsicherheit von Vorhersagen **kann sehr gross sein** und wird oft ungenügend beachtet.
- Als Beispiel nehmen wir das Problem der Vorhersage von Maximaler Herzfrequenz (HR) durch das Alter.
- Ein kleine Studie mit Werten auf `Age` und `HR`:

```
Age <- c(18, 23, 25, 35, 65, 54, 34, 56, 72, 19, 23, 42, 18, 39, 37)
HR <- c(202, 186, 187, 180, 156, 169, 174, 172, 153, 199, 193, 174, 198, 183, 178)
modHR <- lm(HR ~ Age)
```

Vorhersage

Man sieht schön, dass einfache Formeln wie 220 – *Alter* usw. für die meisten Menschen nicht funktionieren.