Thema 4: Mean-Plot

JSA

3/24/23

suppressPackageStartupMessages(library(tidyverse))  
suppressPackageStartupMessages(library(ggplot2))  
suppressPackageStartupMessages(library(flextable))  
suppressPackageStartupMessages(library(officer))

## Beispieldaten für Teil 4 erstellen

Wir erstellen Dummydaten, die aus 5 Variablen bestehen.

* Identifier der Studienteilnehmer
* Behandlungsarm
* Geschlecht
* Alter
* Blutdruck zu Baseline (V1)
* Blutdruck zu V2
* Blutdruck zu V3

Die Daten sind so ausgearbeitet, dass das Test-Medikament zu einer Blutdrucksenkung führt.

Dieses soll jetzt grafisch dargestellt werden.

# Startwert für Zufallszahlengenerator  
set.seed(123)  
# Zahl der Studienteilnehmer festlegen  
n <- 500  
# Folge 1 bis 500 speichern  
id <- seq(1, n)  
# Zufallswerte aus der Menge {"Test", "Reference"} mit den Wahrscheinlichkeiten (0.5, 0.5)  
treatment <- sample(c("Test", "Reference"), n, replace = TRUE, prob = c(0.5, 0.5))  
# Zufallswerte aus der Menge {"Female", "Male"} mit den Wahrscheinlichkeiten (0.5, 0.5)  
gender <- sample(c("Female", "Male"), n, replace = TRUE, prob = c(0.5, 0.5))  
# Gleichförmig verteilte Zufallszahlen aus dem Bereich 18 bis 65  
age <- round(runif(n, 18, 65), 0)  
summary(age)

Min. 1st Qu. Median Mean 3rd Qu. Max.   
 18.00 29.00 41.00 41.22 53.00 65.00

# Normalverteilte Zufallszahlen mit Mittelwert 140 und SD 5 und Offset für Test-Behandlungsarm  
bp\_v1 <- round(rnorm(n, 160, 5))  
bp\_v2 <- round(bp\_v1 + rnorm(n, 0, 2) + ifelse(treatment == "Test", rnorm(1, -5, 1), 0), 0)  
bp\_v3 <- round(bp\_v2 + rnorm(n, 0, 2) + ifelse(treatment == "Test", rnorm(1, -5, 1), 0), 0)  
# Speichern als data.frame, rechteckige Tabellenstruktur wie in einem SAS data set.  
df <- data.frame(id, treatment, gender, age, bp\_v1, bp\_v2, bp\_v3)  
# head() gibt die ersten Zeilen einer Tabelle aus  
head(df)

id treatment gender age bp\_v1 bp\_v2 bp\_v3  
1 1 Reference Male 31 168 170 172  
2 2 Test Male 46 159 155 150  
3 3 Reference Male 26 163 164 162  
4 4 Test Male 58 161 156 152  
5 5 Test Male 58 159 156 154  
6 6 Reference Male 40 159 157 158

## Deskriptive Statistiken für die Variable *Blutdruck* berechnen

Aus der Wide-Form der Daten wird die Long-Form abgeleitet.

df\_long <- df %>% pivot\_longer(cols = -c(id, treatment, gender, age), names\_to = "Visit", values\_to = "RR")  
head(df\_long)

# A tibble: 6 × 6  
 id treatment gender age Visit RR  
 <int> <chr> <chr> <dbl> <chr> <dbl>  
1 1 Reference Male 31 bp\_v1 168  
2 1 Reference Male 31 bp\_v2 170  
3 1 Reference Male 31 bp\_v3 172  
4 2 Test Male 46 bp\_v1 159  
5 2 Test Male 46 bp\_v2 155  
6 2 Test Male 46 bp\_v3 150

Die deskriptiven Statistiken für den mean-Plot werden berechnet.

plot\_df <- df\_long %>%  
 group\_by(treatment, Visit) %>%  
 summarize(m = mean(RR),  
 s = sd(RR)) %>%  
 ungroup()

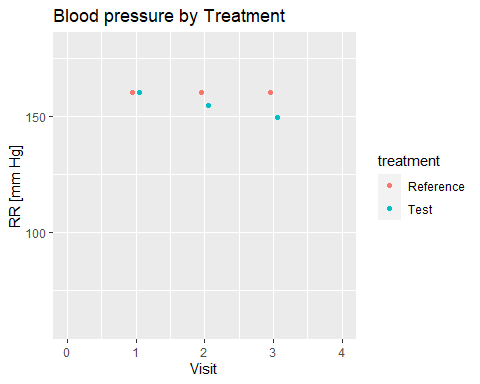
`summarise()` has grouped output by 'treatment'. You can override using the  
`.groups` argument.

plot\_df <- plot\_df %>% mutate(visitn = case\_when(  
 Visit == "bp\_v1" ~ 1,  
 Visit == "bp\_v2" ~ 2,  
 Visit == "bp\_v3" ~ 3))  
plot\_df$visitn <- ifelse(plot\_df$treatment == "Reference",   
 plot\_df$visitn - 0.05,   
 plot\_df$visitn + 0.05)  
  
head(plot\_df)

# A tibble: 6 × 5  
 treatment Visit m s visitn  
 <chr> <chr> <dbl> <dbl> <dbl>  
1 Reference bp\_v1 160. 4.79 0.95  
2 Reference bp\_v2 160. 4.97 1.95  
3 Reference bp\_v3 160. 5.37 2.95  
4 Test bp\_v1 160. 5.38 1.05  
5 Test bp\_v2 155. 5.68 2.05  
6 Test bp\_v3 149. 5.95 3.05

## Plot erstellen

plt <- ggplot(data = plot\_df, aes(x = visitn, y = m, color = treatment)) +  
 geom\_point() +  
 ggtitle("Blood pressure by Treatment") +  
 xlab("Visit") +  
 xlim(0, 4) +  
 ylab("RR [mm Hg]") +  
 ylim(60, 180)  
plt



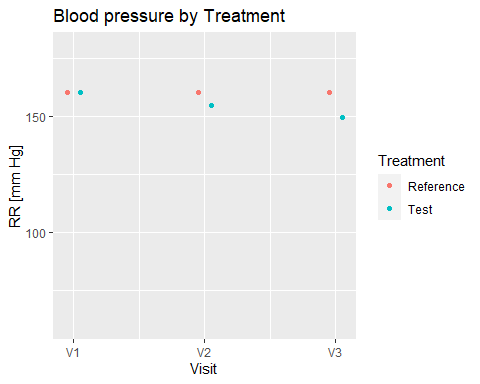
Die Legende wird umbenannt.

Die x-Ticks werden sinnvoller gelabelt.

plt <- plt + scale\_color\_discrete(name = "Treatment", labels = c("Reference", "Test")) +  
 scale\_x\_continuous(breaks = c(1, 2, 3), labels= c("V1", "V2", "V3"))

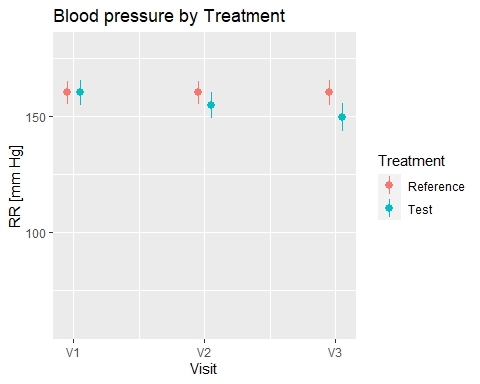
Scale for x is already present.  
Adding another scale for x, which will replace the existing scale.

plt



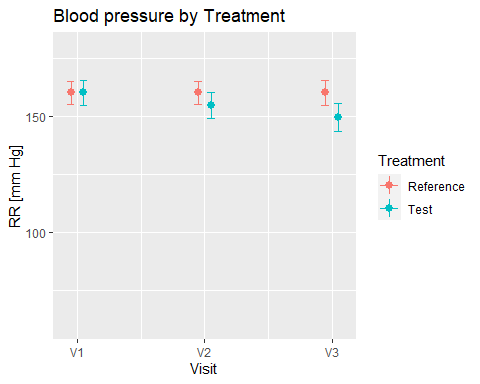
Die Standardabweichung wird als Errorbar ergänzt.

plt <- plt + geom\_pointrange(aes(ymin = m - s, ymax = m + s))  
plt



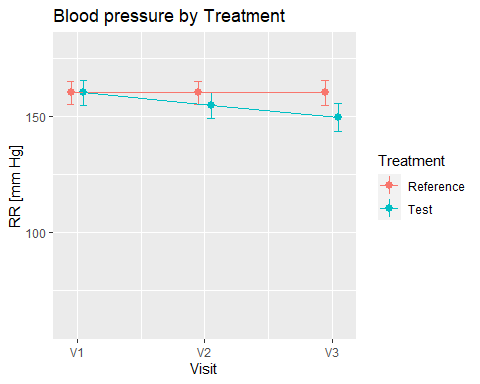
Es kann noch die Verzierung für den Errorbar ergänzt werden.

plt <- plt + geom\_segment(aes(x = visitn - 0.03, xend = visitn + 0.03, y = m + s, yend = m + s)) +  
 geom\_segment(aes(x = visitn - 0.03, xend = visitn + 0.03, y = m - s, yend = m - s))  
plt



Und jetzt wird noch die Verbindungslinie ergänzt.

plt <- plt + geom\_line()  
plt



## Plot speichern

### Plot als Grafik speichern

ggsave(filename = "t4\_meanplot.png", plot = plt)

Saving 5 x 4 in image

### Plot als docx speichern

txt <- "Und hier kommt noch weiterer Text für das Dokument."  
  
my\_doc <- read\_docx() %>%   
 body\_add\_par("Erster importierter Plot im Word-Dokument.") %>%  
 body\_add\_par("Zweite Zeile.") %>%  
 body\_add\_par("") %>%  
 body\_add\_par(txt) %>%  
 body\_add\_par("") %>%  
 body\_add\_gg(plt)  
print(my\_doc, target = "t4\_meanplot.docx")