

Vertiefende statistische Verfahren

3. Übungsblatt SS 2024

Stefan Kolb, Joachim Walzl

Allgemeine Information

Alle Aufgaben sind mit R zu lösen, wenn nicht explizit anders angegeben. Die Berechnungen sollen nachvollziehbar und dokumentiert sein. Um die vollständige Punktezahl zu erreichen, müssen alle Ergebnisse und Fragen entsprechend interpretiert bzw. beantwortet werden. Code alleine ist nicht ausreichend! Die Abgabe erfolgt über Moodle entsprechend der Abgaberrichtlinien als pdf und Rmd File. Bitte inkludieren Sie namentlich alle beteiligten Gruppenmitglieder sowohl im Bericht als auch im Source Code. Die jeweiligen Datensätze die für diese Übung relevant sind finden Sie ebenfalls in Moodle.

1 Einfaktorielle ANOVA (händisch) [2P]

Ein Sportwissenschaftler möchte sehen, ob es einen Unterschied in der Gewichtszunahme von Sportlern gibt, die einer von drei speziellen Diäten folgen. Die Athleten werden nach dem Zufallsprinzip drei Gruppen zugewiesen und unterziehen sich für 6 Wochen der jeweiligen Diät. Die Gewichtszunahmen (in Pfund) sind angegeben. Nehmen Sie an, dass die Gewichtszunahmen normalverteilt sind und die Varianzen gleich sind. Kann der Sportwissenschaftler bei einem Signifikanzniveau von 0,05 schlussfolgern, dass es einen Unterschied zwischen den Diäten gibt? Führen Sie die ANOVA händisch durch und überprüfen Sie das Ergebnis mit `summary(aov(...))`.

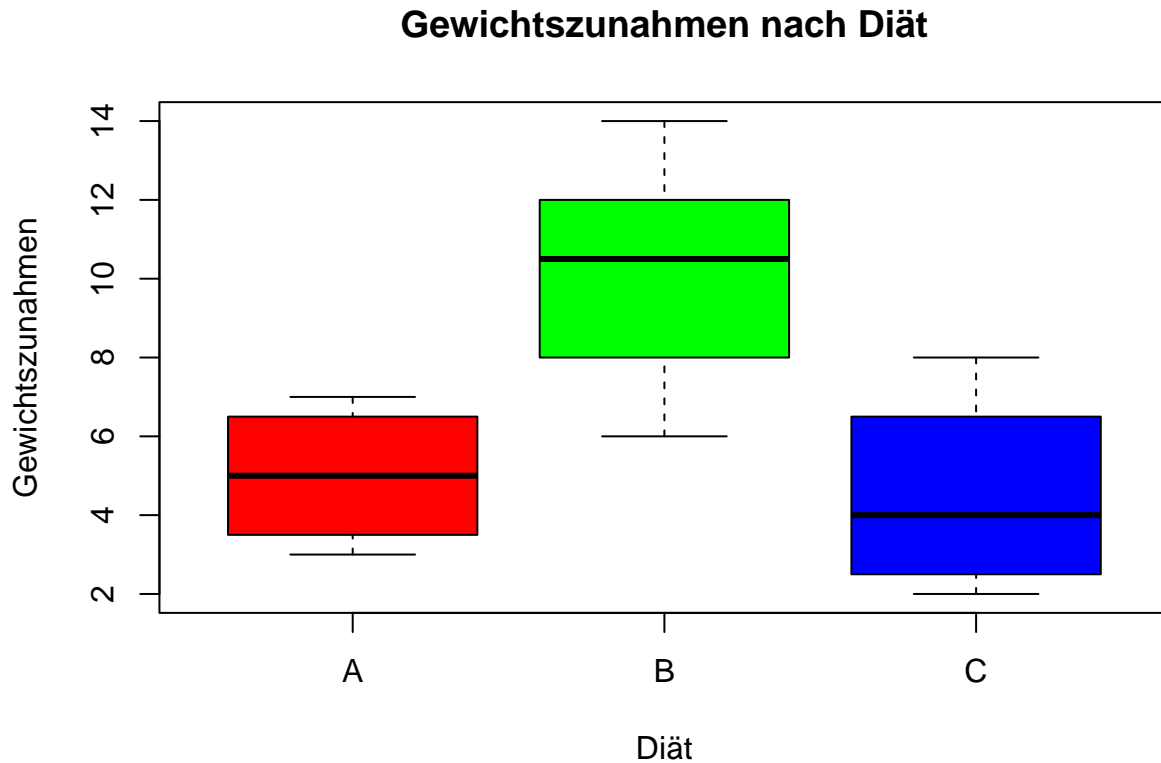
Diät	Messwerte
A	3, 6, 7, 4
B	10, 12, 11, 14, 8, 6
C	8, 3, 2, 5

```
# Einlesen der Daten
gains <- list(
  A = c(3, 6, 7, 4),
  B = c(10, 12, 11, 14, 8, 6),
  C = c(8, 3, 2, 5))

# Überblick über die Daten / Farben nach Diet
str(gains)
```

```
## List of 3
## $ A: num [1:4] 3 6 7 4
## $ B: num [1:6] 10 12 11 14 8 6
## $ C: num [1:4] 8 3 2 5
```

```
boxplot(gains, col = c("red", "green", "blue"),
        ylab = "Gewichtszunahmen",
        xlab = "Diät",
        main = "Gewichtszunahmen nach Diät")
```



```
# Hypothesen
# H0:  $\mu_a = \mu_b = \mu_c$ 
# H1:  $\mu_i \neq \mu_j$  (für mindestens ein Paar  $i, j$ )

# Berechnung des Gesamt-Mittelwertes
mean_total = mean(unlist(gains))

# Berechnung der Quadratsummen
qs_between <- 0
for (group in gains) {
  group_mean <- mean(group)
  n <- length(group)
  qs_between <- qs_between + n * (group_mean - mean_total)^2
}

paste("Die Quadratsumme zwischen den Gruppen beträgt: ", round(qs_between, 2))
```

```
## [1] "Die Quadratsumme zwischen den Gruppen beträgt: 101.1"
```

```

qs_within <- 0
for (group in gains) {
  group_mean <- mean(group)
  for (value in group) {
    qs_within <- qs_within + (value - group_mean)^2
  }
}

paste("Die Quadratsumme innerhalb der Gruppen beträgt: ", round(qs_within,2))

```

```
## [1] "Die Quadratsumme innerhalb der Gruppen beträgt: 71.83"
```

```

qs_total <- sum((unlist(gains) - mean_total)^2)

paste("Die Quadratsumme insgesamt beträgt: ", round(qs_total,2))

```

```
## [1] "Die Quadratsumme insgesamt beträgt: 172.93"
```

```

# Prüfung der Quadratsummen
round(qs_total, 2)==round(qs_between + qs_within, 2)

```

```
## [1] TRUE
```

```

# Überprüfen der Voraussetzungen

# Normalverteilung
for (group in gains) {
  print(shapiro.test(group))
}

```

```

##
## Shapiro-Wilk normality test
##
## data: group
## W = 0.94971, p-value = 0.7143
##
##
## Shapiro-Wilk normality test
##
## data: group
## W = 0.98901, p-value = 0.9866
##
##
## Shapiro-Wilk normality test
##
## data: group
## W = 0.94563, p-value = 0.6889

```

```

# Varianzhomogenität (Barlett-Test)
bartlett.test(gains)

```

```
##
## Bartlett test of homogeneity of variances
##
## data: gains
## Bartlett's K-squared = 0.61192, df = 2, p-value = 0.7364
```

```
# Freiheitsgrade
df_between <- length(gains) - 1
df_within <- length(unlist(gains)) - length(gains)
df_total <- length(unlist(gains)) - 1

# Berechnung der Varianzen (Mittlere Quadratsummen)
ms_total <- qs_total / df_total
ms_between <- qs_between / df_between
ms_within <- qs_within / df_within

# Berechnung des F-Wertes
f_value <- ms_between / ms_within
paste("Der F-Wert beträgt: ", round(f_value, 2))
```

```
## [1] "Der F-Wert beträgt: 7.74"
```

```
# kritischer F-Wert aus Tabelle
alpha <- 0.05
f_critical <- qf(alpha, df_between, df_within, lower.tail = F)
paste("Der kritische F-Wert beträgt: ", round(f_critical, 2))
```

```
## [1] "Der kritische F-Wert beträgt: 3.98"
```

```
# Ergebnis
if (f_value > f_critical) {
  print("H0 wird verworfen")
} else {
  print("H0 wird nicht verworfen")
}
```

```
## [1] "H0 wird verworfen"
```

```
# p-Wert
p_value <- pf(f_value, df_between, df_within, lower.tail = F)
paste("Der berechnete p-Wert beträgt: ", round(p_value, 5))
```

```
## [1] "Der berechnete p-Wert beträgt: 0.00797"
```

```
# Überprüfung mit R
diet <- factor(rep(names(gains), times = sapply(gains, length)))
all_gains <- unlist(gains)

aov_result <- aov(all_gains ~ diet)
summary(aov_result)
```

```
##           Df Sum Sq Mean Sq F value    Pr(>F)
## diet       2 101.10   50.55     7.74 0.00797 **
## Residuals 11  71.83    6.53
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation: In diesem Fall gibt es einen signifikanten Einfluss des Faktors Diät [$F(2,11) = 7.74$; $p < 0.05$] auf die Gewichtszunahme der Sportler. Die Ergebnisse der manuellen Berechnung, stimmen mit den Ergebnissen der ANOVA-Analyse in R überein.

2 Einfaktorielle ANOVA [2P]

Analysieren Sie die Daten zu niedrigem Geburtsgewicht von Neugeborenen in `birthwt_aov.xlsx`. Untersuchen Sie ob die ethnische Zugehörigkeit der Mutter (Variable `ethnic`, wobei 1 = "white", 2 = "black", 3 = "other") einen Einfluss auf das Geburtsgewicht von Neugeborenen hat (Variable `bwt`). Beachten Sie dabei folgende Punkte:

- i) Überprüfen Sie die Voraussetzungen.
- ii) Verwenden Sie eine geeignete grafische Darstellung der Daten / Ergebnisse.
- iii) Gibt es signifikante Unterschiede zwischen den Gruppen, wenn ja zwischen welchen Gruppen?
- iv) Achten Sie auf eine "statistisch korrekte" Formulierung des Ergebnisses.

```
# Daten einlesen
library(readxl)
birthwt <- read_excel("UE3 Daten/birthwt_aov.xlsx")

# Faktor für ethnische Zugehörigkeit
birthwt$ethnic <- factor(birthwt$ethnic, levels = c(1,2,3), labels = c("white", "black", "other"))

# Überblick über die Daten
str(birthwt)
```

```
## tibble [189 x 10] (S3: tbl_df/tbl/data.frame)
## $ low   : num [1:189] 0 0 0 0 0 0 0 0 0 0 ...
## $ age   : num [1:189] 19 33 20 21 18 21 22 17 29 26 ...
## $ lwt   : num [1:189] 182 155 105 108 107 124 118 103 123 113 ...
## $ ethnic: Factor w/ 3 levels "white","black",...: 2 3 1 1 1 3 1 3 1 1 ...
## $ smoke : num [1:189] 0 0 1 1 1 0 0 0 1 1 ...
## $ ptl    : num [1:189] 0 0 0 0 0 0 0 0 0 0 ...
## $ ht     : num [1:189] 0 0 0 0 0 0 0 0 0 0 ...
## $ ui     : num [1:189] 1 0 0 1 1 0 0 0 0 0 ...
## $ ftv    : num [1:189] 0 3 1 2 0 0 1 1 1 0 ...
## $ bwt    : num [1:189] 2523 2551 2557 2594 2600 ...
```

```
table(birthwt$ethnic)
```

```
##
## white black other
##    96    26    67
```

```
summary(birthwt$bwt)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      709    2414    2977    2945    3487    4990
```

```
# Hypothesen
```

```
# H0:  $\mu_{white} = \mu_{black} = \mu_{other}$ 
```

```
# H1:  $\mu_i \neq \mu_j$  (für mindestens ein Paar  $i, j$ )
```

```
# Überprüfen der Voraussetzungen
```

```
# Vergleich der Gruppengrößen
```

```
max(table(birthwt$ethnic))/min(table(birthwt$ethnic)) < 1.5
```

```
## [1] FALSE
```

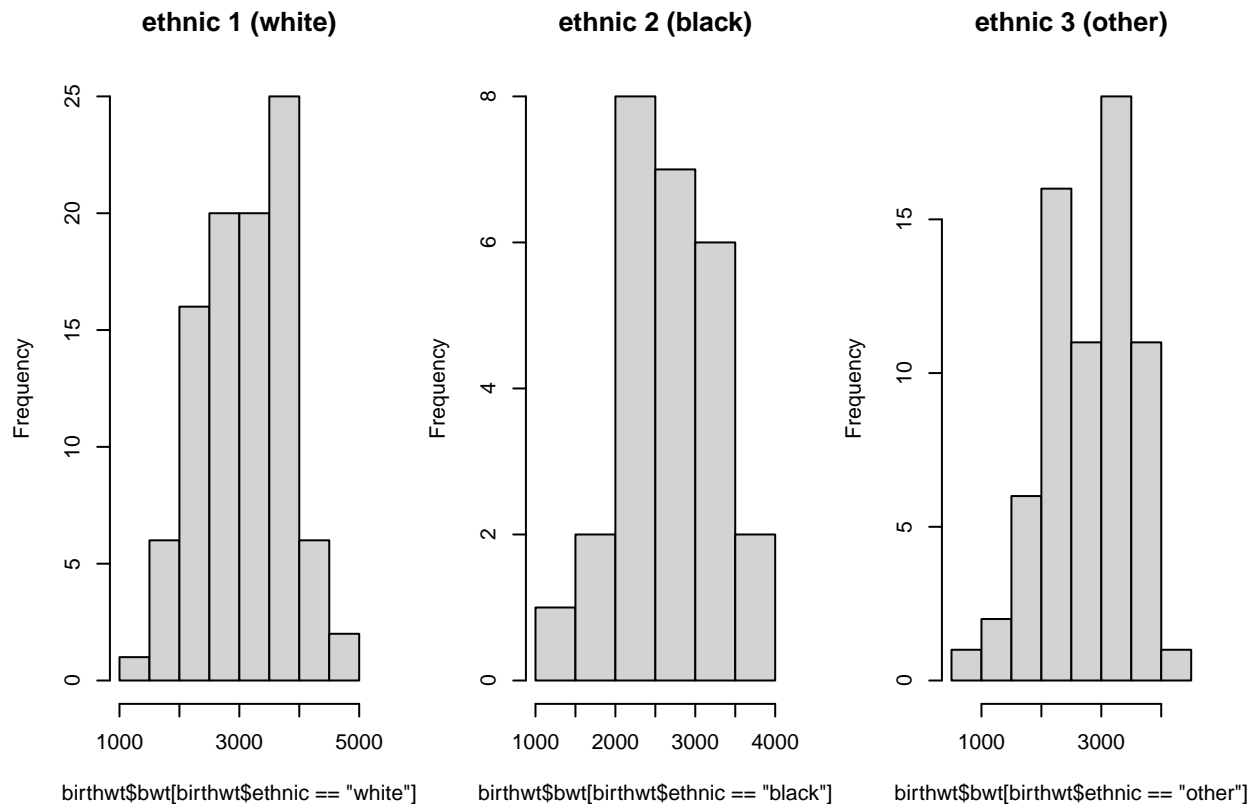
```
# Normalverteilung (für jede Gruppe)
```

```
par(mfrow = c(1,3))
```

```
hist(birthwt$bwt[birthwt$ethnic == 'white'], main = "ethnic 1 (white)")
```

```
hist(birthwt$bwt[birthwt$ethnic == 'black'], main = "ethnic 2 (black)")
```

```
hist(birthwt$bwt[birthwt$ethnic == 'other'], main = "ethnic 3 (other)")
```



```

ethnics <- unique(birthwt$ethnic)
# Shapiro-Wilk-Test
for (ethnic in ethnics) {
  print(shapiro.test(birthwt$bwt[birthwt$ethnic == ethnic]))
}

```

```

##
## Shapiro-Wilk normality test
##
## data: birthwt$bwt[birthwt$ethnic == ethnic]
## W = 0.97696, p-value = 0.8038
##
##
## Shapiro-Wilk normality test
##
## data: birthwt$bwt[birthwt$ethnic == ethnic]
## W = 0.97537, p-value = 0.2046
##
##
## Shapiro-Wilk normality test
##
## data: birthwt$bwt[birthwt$ethnic == ethnic]
## W = 0.98727, p-value = 0.4861

```

Ein visuelle Überprüfung der Normalverteilung zeigt, dass diese für alle Gruppen gegeben ist. Der Shapiro-Wilk-Test bestätigt diese Annahme, weshalb die Annahme einer Normalverteilung der Residien erfüllt ist. Ein Vergleich der Gruppengrößen zeigt, dass die Gruppen unterschiedlich groß sind, da der Quotient aus der größten und kleinsten Gruppengröße größer als 1.5 ist. Das ist jedoch kein großes Problem, da die ANOVA robust gegenüber unterschiedlichen Gruppengrößen ist, insofern die Varianzhomogenität nicht verletzt wird Wikipedia.

```

# Varianzhomogenität (Heteroskedastizität)

```

```

# Levene-Test
library(car)
leveneTest(bwt ~ ethnic, data = birthwt)

```

```

## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group  2  0.4684 0.6267
##      186

```

```

# Bartlett-Test
bartlett.test(birthwt$bwt ~ birthwt$ethnic)

```

```

##
## Bartlett test of homogeneity of variances
##
## data: birthwt$bwt by birthwt$ethnic
## Bartlett's K-squared = 0.65952, df = 2, p-value = 0.7191

```

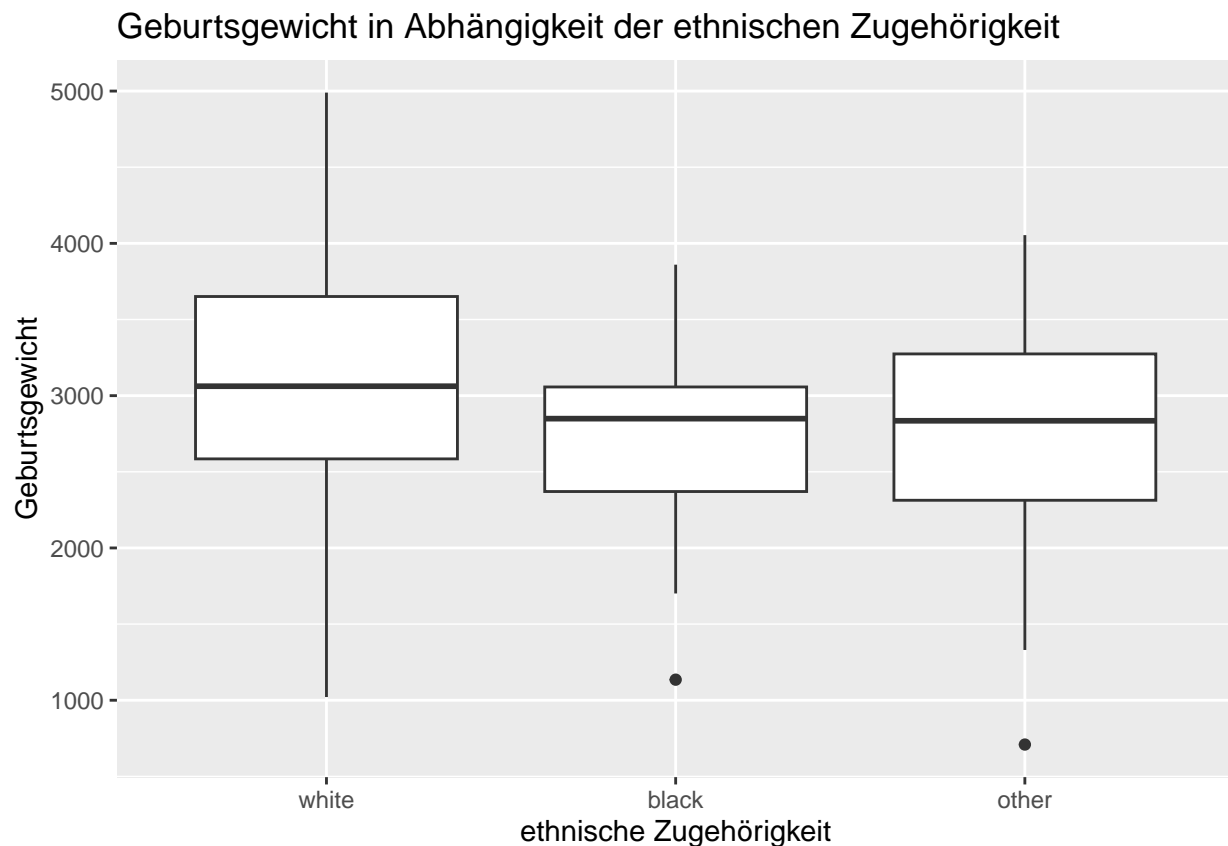
Sowohl der Levene- als auch der Bartlett-Test zeigen, dass die Varianzhomogenität gegeben ist.

Alle wesentlichen Voraussetzungen für die ANOVA sind somit gegeben und sie kann ohne Bedenken durchgeführt werden.

```
# ANOVA
aov_result <- aov(bwt ~ ethnic, data = birthwt)
summary(aov_result)

##           Df    Sum Sq Mean Sq F value    Pr(>F)    
## ethnic      2  5015725  2507863   4.913 0.00834 ** 
## Residuals 186 94953931   510505                
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Darstellung der Ergebnisse
library(ggplot2)
ggplot(birthwt, aes(x = ethnic, y = bwt)) +
  geom_boxplot() +
  labs(title = "Geburtsgewicht in Abhängigkeit der ethnischen Zugehörigkeit",
       x = "ethnische Zugehörigkeit",
       y = "Geburtsgewicht")
```



Die ethnische Zugehörigkeit hat einen signifikanten Einfluss auf das Geburtsgewicht von Neugeborenen, $F(2,186) = 4.91$, $p = 0.008$.

Um herauszufinden zwischen welchen Gruppen signifikante Unterschiede bestehen, wird nun eine Post-Hoc-Analyse durchgeführt. Als geeigneter Test wird der Tukey HSD-Test verwendet.


```

# Post-Hoc-Test
pht <- TukeyHSD(aov_result)
print(pht)

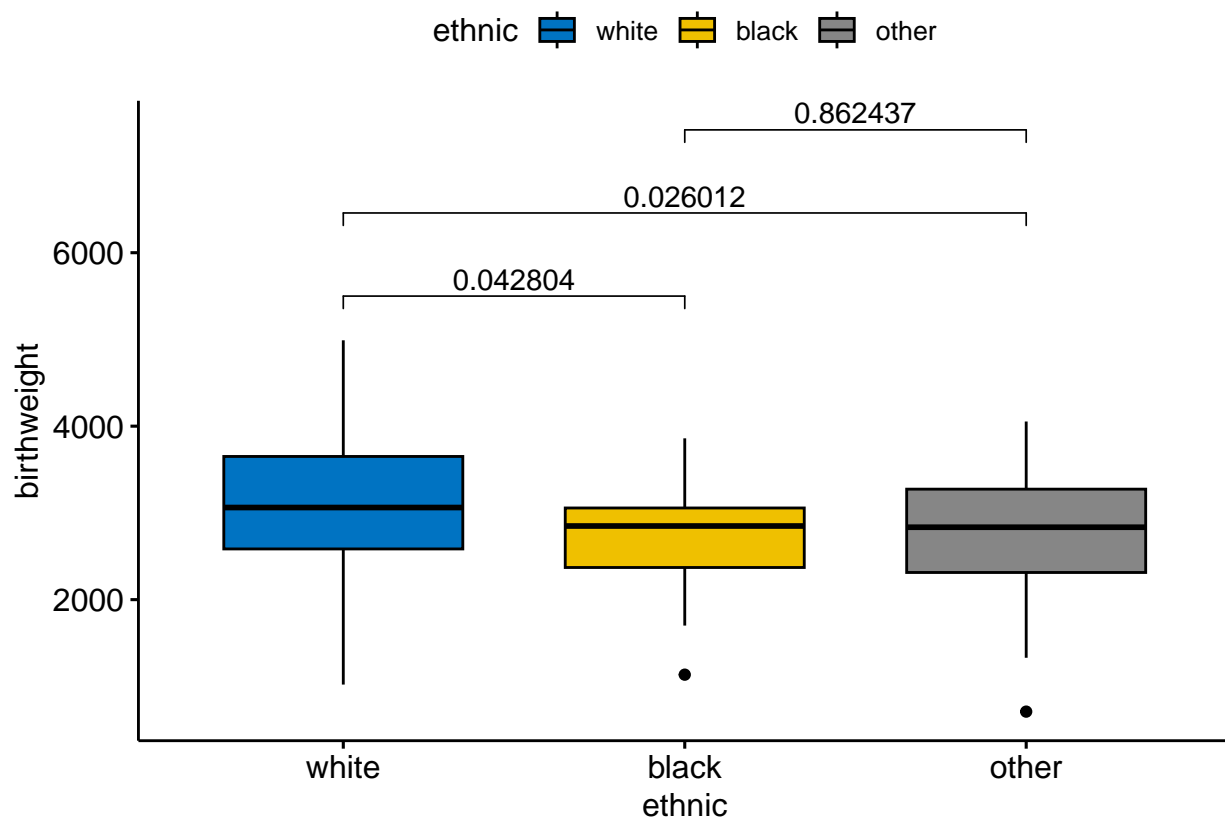
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = bwt ~ ethnic, data = birthwt)
##
## $ethnic
##          diff          lwr          upr      p adj
## black-white -383.02644 -756.2363  -9.816581 0.0428037
## other-white -297.43517 -566.1652 -28.705095 0.0260124
## other-black  85.59127 -304.4521 475.634630 0.8624372

# Darstellung der Ergebnisse
library(ggpubr)

stat.test <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "white",  "black", round(pht$ethnic[1,4],6),
  "white",  "other", round(pht$ethnic[2,4],6),
  "black",  "other", round(pht$ethnic[3,4],6))

# Boxplots (inkl. p-Werte)
ggboxplot(birthwt, x = "ethnic", y = "bwt",
  fill = "ethnic",
  palette = "jco",
  ylab = "birthweight") +
  stat_pvalue_manual(stat.test,
    y.position = 5500, step.increase = 0.2,
    label = "p.adj")

```



Der Tukey HSD-Test zeigt, dass es signifikante Unterschiede zwischen den ethnischen Gruppen “white” und “black” ($p = 0.043$) sowie “white” und “other” ($p = 0.026$) gibt. Zwischen den Gruppen “black” und “other” besteht hingegen kein signifikanter Unterschied ($p = 0.862$). Konkret wiegen schwarze Neugeborene durchschnittlich 383,03 Gramm weniger als weiße. Neugeborene aus anderen ethnischen Gruppen weisen ebenfalls ein signifikant geringeres Geburtsgewicht auf als weiße Neugeborene, mit einem durchschnittlichen Unterschied von 297,44 Gramm. Im Vergleich zwischen schwarzen und anderen ethnischen Gruppen gibt es jedoch keinen signifikanten Unterschied im Geburtsgewicht, wie die Analyse mit einem p-Wert von 0,8624 zeigt.

3 Einfaktorielle ANOVA [2P]

Verwenden Sie den bereits bekannten `Framingham.sav` Datensatz. Analysieren Sie ob es Unterschiede im BMI in Abhängigkeit von der Schulbildung gibt. Achten Sie auf Ausreißer und fehlende Daten (`NaN`, `NA's`).

i) Überprüfen Sie die Voraussetzungen.

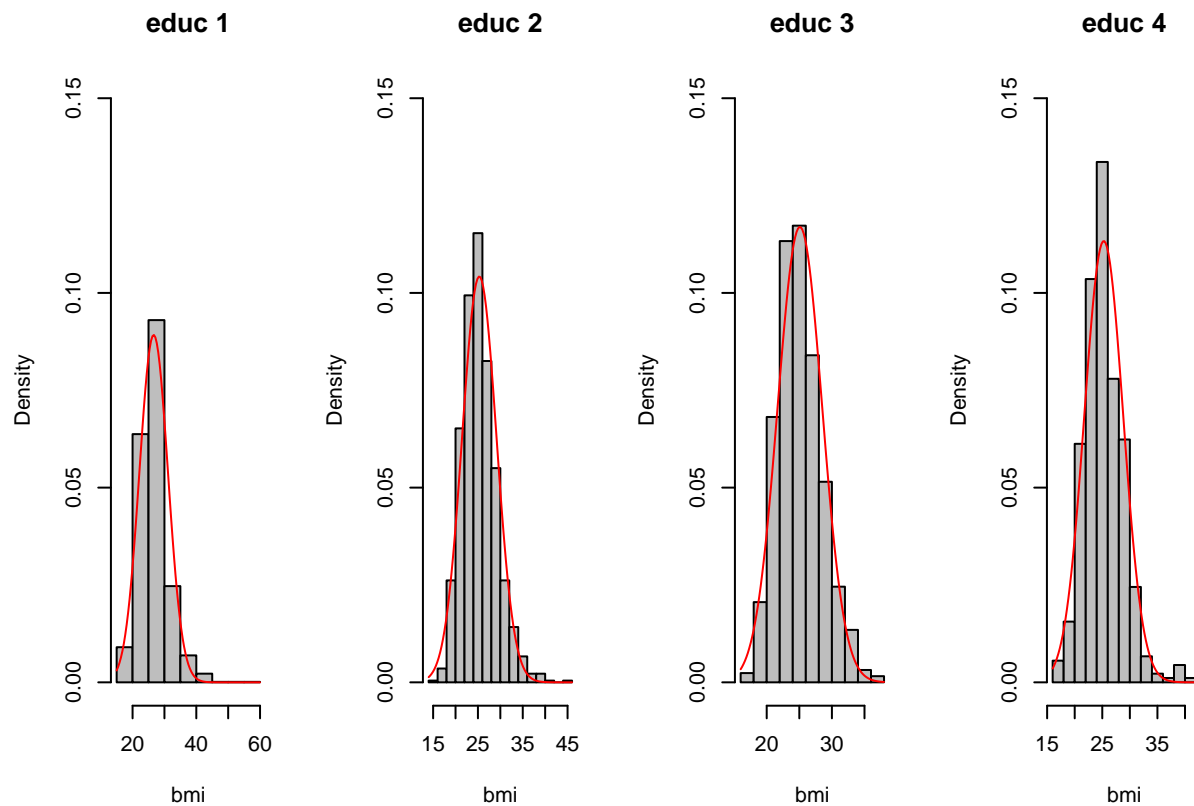
```
load_source()

# Load and clean data. Handle factor variables
framingham <- load_and_prepare_framingham("UE3 Daten/Framingham.sav")

# Check requirements for ANOVA

# -- For education levels
```

```
test_normality_of_groups(framingham, "educ", "bmi")
```



```
##
##  Shapiro-Wilk normality test
##
## data:  data$bmi[data$educ==1]
## W = 0.94863, p-value < 2.2e-16
##
##
##  Shapiro-Wilk normality test
##
## data:  data$bmi[data$educ==2]
## W = 0.97479, p-value = 4.192e-13
##
##
##  Shapiro-Wilk normality test
##
## data:  data$bmi[data$educ==3]
## W = 0.98319, p-value = 1.188e-06
##
##
##  Shapiro-Wilk normality test
##
## data:  data$bmi[data$educ==4]
## W = 0.96666, p-value = 1.419e-08
```

```
# Variance inhomogeneity (Heteroscedasticity)
```

```
# Levene test
```

```
leveneTest(bmi ~ educ, data = framingham)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
```

```
##           Df F value    Pr(>F)
```

```
## group      3  12.304 5.22e-08 ***
```

```
##           3822
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Bartlett test
```

```
bartlett.test(bmi ~ educ, data = framingham)
```

```
##
```

```
## Bartlett test of homogeneity of variances
```

```
##
```

```
## data:  bmi by educ
```

```
## Bartlett's K-squared = 89.772, df = 3, p-value < 2.2e-16
```

Die Histogramme der BMI-Werte für die verschiedenen Bildungsstufen zeigen, dass die Verteilung der BMI-Werte für jede Bildungsstufe keiner Normalverteilung folgt. Der Shapiro-Wilk-Test bestätigt diese Annahme, da die p-Werte für alle Bildungsstufen kleiner als 0,05 sind. Außerdem zeigt der Levene Test, dass die Varianz aller Gruppen nicht gleich ist. Daher sind die Voraussetzungen einer ANOVA verletzt. Allerdings reagiert der F-Test auf Abweichungen nicht so empfindlich, wenn 1) die Gruppengrößen etwa gleich sind 2) und die Gruppen hinreichend groß sind.

```
table(framingham$educ)
```

```
##
```

```
##      1      2      3      4
```

```
## 1619 1127  631  449
```

Da Bedingung 1 nicht erfüllt ist, muss von einer ANOVA abgesehen werden. Stattdessen wird ein Kruskal Wallis Test angewandt.

i) Verwenden Sie eine geeignete grafische Darstellung der Daten / Ergebnisse.

```
# Boxplot education levels
```

```
ggplot(framingham, aes(x = educ, y = bmi)) +
```

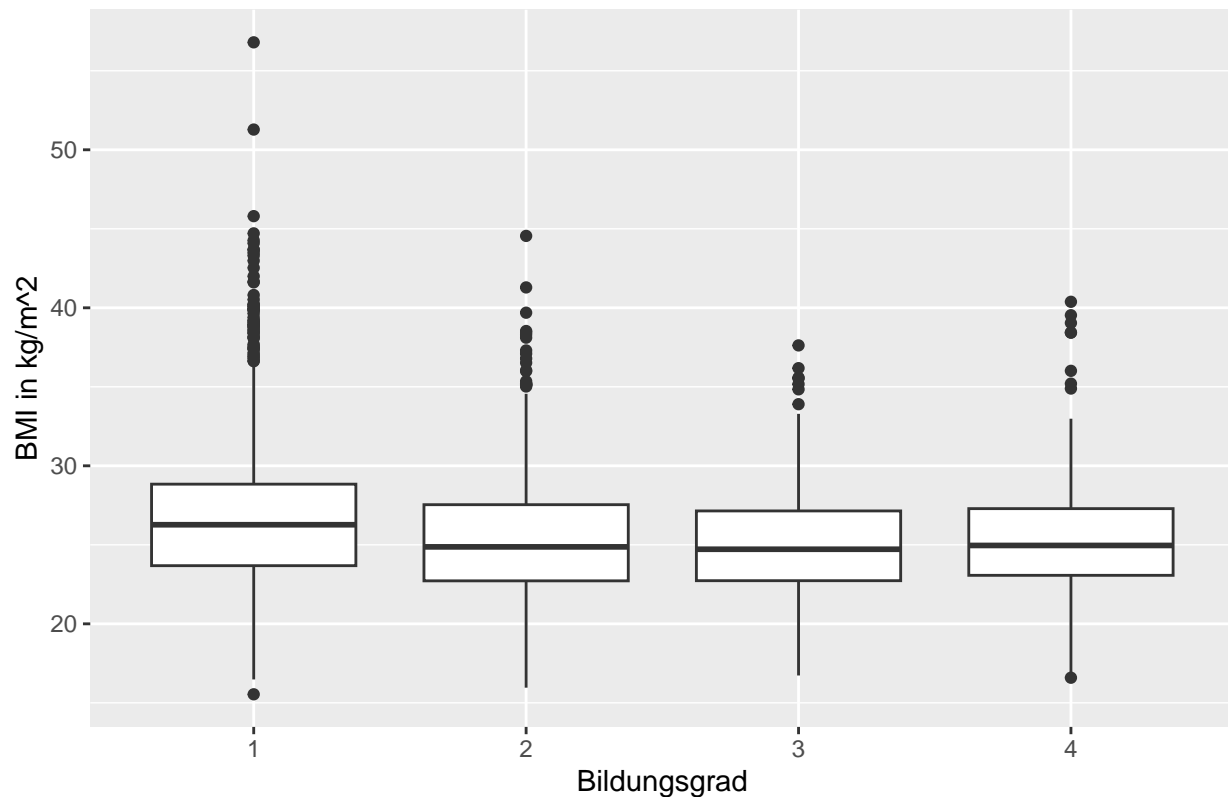
```
  geom_boxplot() +
```

```
  labs(title = "BMI in Abhängigkeit des Bildungsgrades",
```

```
        x = "Bildungsgrad",
```

```
        y = "BMI in kg/m^2")
```

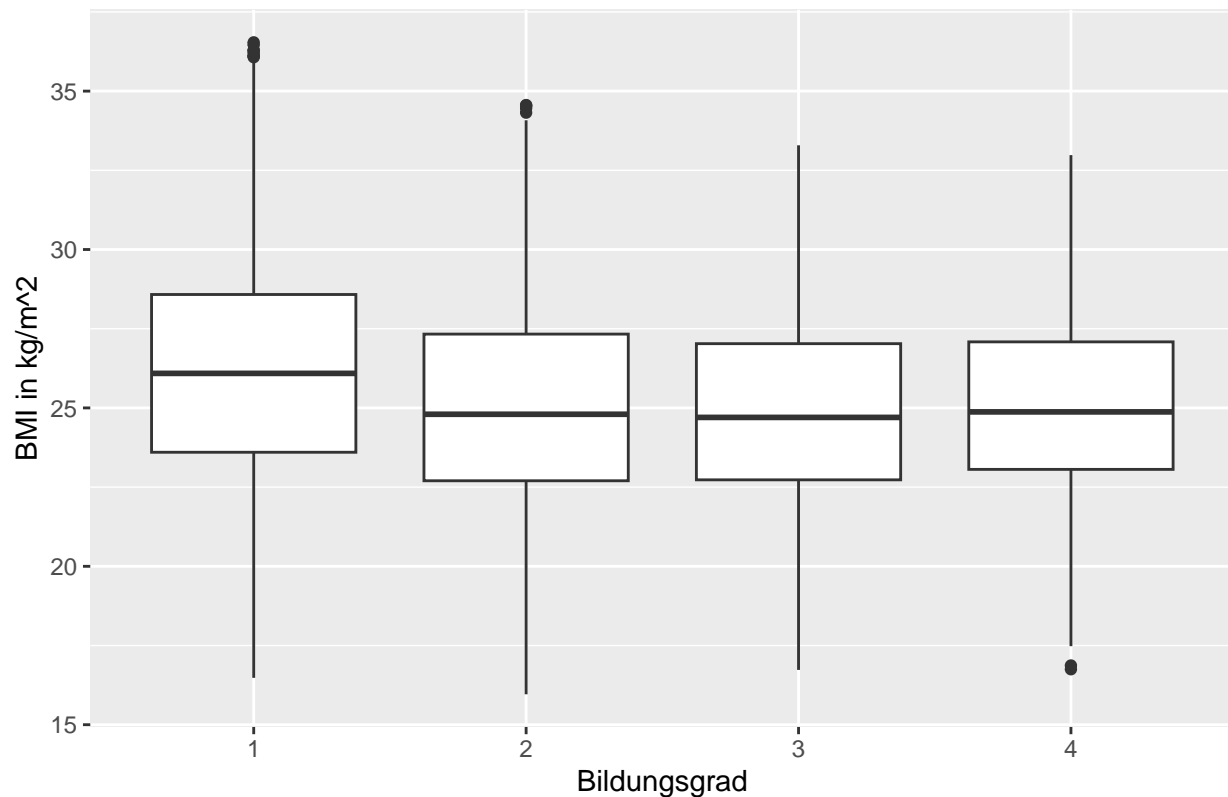
BMI in Abhängigkeit des Bildungsgrades



```
# Remove outliers (median +/- 1.5 IQR)
library(dplyr)
framingham_clean <- framingham %>%
  group_by(educ) %>%
  mutate(lower_bound = quantile(bmi, 0.25) - 1.5 * IQR(bmi),
         upper_bound = quantile(bmi, 0.75) + 1.5 * IQR(bmi)) %>%
  filter(bmi >= lower_bound & bmi <= upper_bound)

# Boxplot education levels without outliers
ggplot(framingham_clean, aes(x = educ, y = bmi)) +
  geom_boxplot() +
  labs(title = "BMI in Abhängigkeit des Bildungsgrades",
       x = "Bildungsgrad",
       y = "BMI in kg/m^2")
```

BMI in Abhängigkeit des Bildungsgrades



Die Boxplots legen die Vermutung nahe, dass zumindest der Unterschied zwischen Gruppe "Bildungsgrad 1" und den anderen Gruppen signifikant ist.

i) Gibt es signifikante Unterschiede zwischen den Bildungsstufen, wenn ja zwischen welchen Stufen?

```
# RESULTS NOT VALID! REQUIREMENTS NOT MET!
```

```
# # ANOVA
# aov_result <- aov(bmi ~ educ, data = framingham)
# summary(aov_result)

# # Show results
# library(ggplot2)
# ggplot(framingham, aes(x = educ, y = bmi)) +
#   geom_boxplot() +
#   labs(title = "BMI in Abhängigkeit des Bildungsgrades",
#         x = "Bildungsgrad",
#         y = "BMI in kg/m^2")
```

```
# RESULTS NOT VALID! REQUIREMENTS NOT MET!
```

```
# # Post hoc analysis
# pht <- TukeyHSD(aov_result)
# print(pht)

# # Show results
```

```

# library(ggpubr)

# stat.test <- tibble::tribble(
#   ~group1, ~group2, ~p.adj,
#   "1",      "2", round(pht$educ[1,4],6),
#   "1",      "3", round(pht$educ[2,4],6),
#   "1",      "4", round(pht$educ[3,4],6),
#   "2",      "3", round(pht$educ[4,4],6),
#   "2",      "4", round(pht$educ[5,4],6),
#   "3",      "4", round(pht$educ[6,4],6))

# # Determine the maximum value of the 'bmi' column
# max_bmi <- max(framingham$bmi, na.rm = TRUE)

# # Create an array of y positions starting from the maximum BMI value and incrementing by a small amount
# y_positions <- max_bmi + seq(from = 1, to = 6, by = 1)

# # Create boxplot
# ggboxplot(framingham, x = "educ", y = "bmi",
#           color = "educ",
#           palette = "jco") +
#   stat_pvalue_manual(stat.test,
#                     y.position = y_positions,
#                     step.increase = 0.2,
#                     label = "p.adj")

# Kruskal Wallis Test
# Non parametric test instead of one way ANOVA

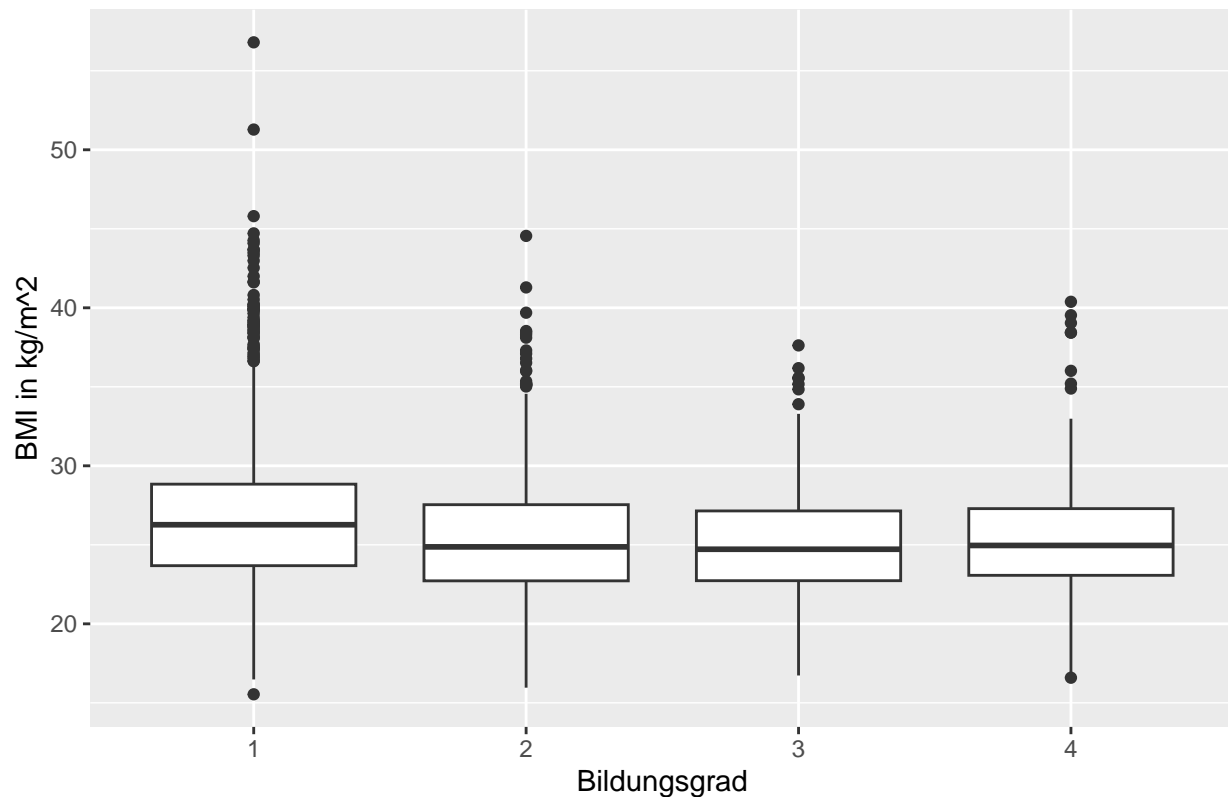
kwt_result <- kruskal.test(bmi ~ educ, data = framingham)
kwt_result

##
## Kruskal-Wallis rank sum test
##
## data:  bmi by educ
## Kruskal-Wallis chi-squared = 102.08, df = 3, p-value < 2.2e-16

# Show results
library(ggplot2)
ggplot(framingham, aes(x = educ, y = bmi)) +
  geom_boxplot() +
  labs(title = "BMI in Abhängigkeit des Bildungsgrades",
       x = "Bildungsgrad",
       y = "BMI in kg/m^2")

```

BMI in Abhängigkeit des Bildungsgrades



Da der p-Wert kleiner als 0.05 ist, kann angenommen werden, dass es einen signifikanten Unterschied zwischen den Bildungsstufen gibt.

```
# Post hoc test

# Perform pairwise Wilcoxon test
pwt_result <- pairwise.wilcox.test(framingham$bmi, framingham$educ, p.adjust.method = "BH")

# Extract p-values
p_values <- pwt_result$p.value
print(p_values)
```

```
##           1           2           3
## 2 4.143386e-15         NA         NA
## 3 3.142318e-14 0.4660284         NA
## 4 3.560653e-09 0.8529525 0.4660284
```

```
# Create data frame for manual p-value annotation
stat.test <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "1",     "2", round(p_values[1],6),
  "1",     "3", round(p_values[2],6),
  "1",     "4", round(p_values[3],6),
  "2",     "3", round(p_values[5],6),
  "2",     "4", round(p_values[6],6),
  "3",     "4", round(p_values[9],6))
```



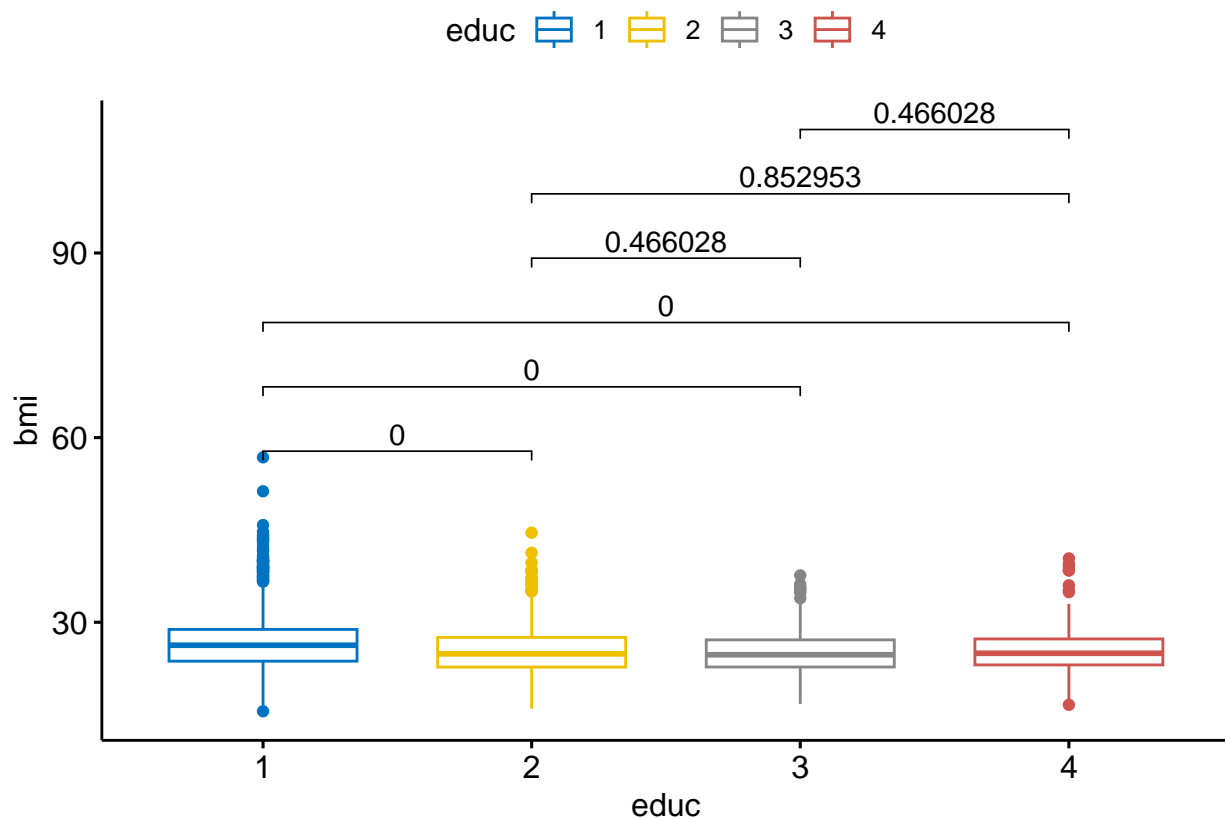
```

# Determine the maximum value of the 'bmi' column
max_bmi <- max(framingham$bmi, na.rm = TRUE)

# Create an array of y positions starting from the maximum BMI value and incrementing by a small amount
y_positions <- max_bmi + seq(from = 1, to = 6, by = 1)

# Create boxplot
ggboxplot(framingham, x = "educ", y = "bmi",
          color = "educ",
          palette = "jco") +
  stat_pvalue_manual(stat.test,
                    y.position = y_positions,
                    step.increase = 0.2,
                    label = "p.adj")

```



i) Achten Sie auf eine “statistisch korrekte” Formulierung des Ergebnisses.

Der Wilcoxon Test liefert beim Faktor “Bildungsgrad” für die Vergleichspaare (1,2), (1,3) und (1,4) einen hochsignifikanten p-Wert kleiner 0.01. D.h. der BMI der Gruppe mit Bildungsgrad 1 unterscheidet sich signifikant von den anderen Bildungsgraden, während der Unterschied zwischen den anderen Bildungsgraden (2,3), (2,4), (3,4) nicht signifikant ist. Nimmt man an, dass die Gruppen aus Populationen mit annähernd derselben Verteilung entstammen, impliziert dies, dass der Median der Gruppe mit Bildungsgrad 1 signifikant höher als der Median der anderen Gruppen ist.

4 Mehrfaktorielle ANOVA [2P]

Sie führen eine Studie bezüglich des Einflusses unterschiedlicher Diäten und Aktivitätslevels auf den Erfolg bei der Gewichtsabnahme durch. Jedem Probanden wird eine Diät und ein Aktivitätslevel zugewiesen und die Differenz zum Ausgangsgewicht nach 2 Monaten gemessen (in kg). Die Daten der Studie sind in `weightloss.sav` zusammengefasst. Analysieren Sie ob die beiden Faktoren einen Einfluss auf das Gewicht haben.

- i) Überprüfen Sie die Voraussetzungen.
- ii) Verwenden Sie eine geeignete grafische Darstellung der Daten / Ergebnisse.
- iii) Gibt es signifikante Haupteffekte sowie Interaktionseffekte (inkl. Interaktions-Plot)?
- iv) Achten Sie auf eine "statistisch korrekte" Formulierung des Ergebnisses.

```
# Laden der Daten
library(foreign)
weightloss <- read.spss("UE3 Daten/weightloss.sav", to.data.frame = TRUE)

head(weightloss)
```

```
##   id      diet      exercise wloss
## 1  1 Vegetarian 60 minutes per day 13.96
## 2  2      None 30 minutes per day -2.12
## 3  3      None 60 minutes per day  6.92
## 4  4    Atkins 30 minutes per day  0.68
## 5  5 Vegetarian 60 minutes per day  4.68
## 6  6      None 30 minutes per day  0.36
```

```
summary(weightloss)
```

```
##           id           diet           exercise           wloss
## Min.      : 1.00      None      :60      None           :60      Min.      : -6.440
## 1st Qu.: 45.75    Atkins      :60    30 minutes per day:60    1st Qu.:  1.220
## Median : 90.50    Vegetarian:60    60 minutes per day:60    Median :  4.640
## Mean     : 90.50                                     Mean     :  4.488
## 3rd Qu.:135.25                                     3rd Qu.:  7.320
## Max.     :180.00                                     Max.     :14.760
```

```
attr(weightloss, "variable.labels")
```

```
##                                     id
##                                     ""
##                                     diet
##          "Diet assigned to participant"
##                                     exercise
##          "Exercise level assigned to participant"
##                                     wloss
## "Observed weight loss in kilos over last 2 months"
```

```
# Hypothesen
```

```
# H0 (Diät):  $\mu_{noDiet} = \mu_{vegetarian} = \mu_{atkins}$ 
# H0 (Aktivität):  $\mu_{none} = \mu_{30} = \mu_{60}$ 
# H0 (Interaktion): Alle Mittelwertunterschiede werden durch die Haupteffekte erklärt.
```

Nun werden die Voraussetzungen für die ANOVA überprüft.

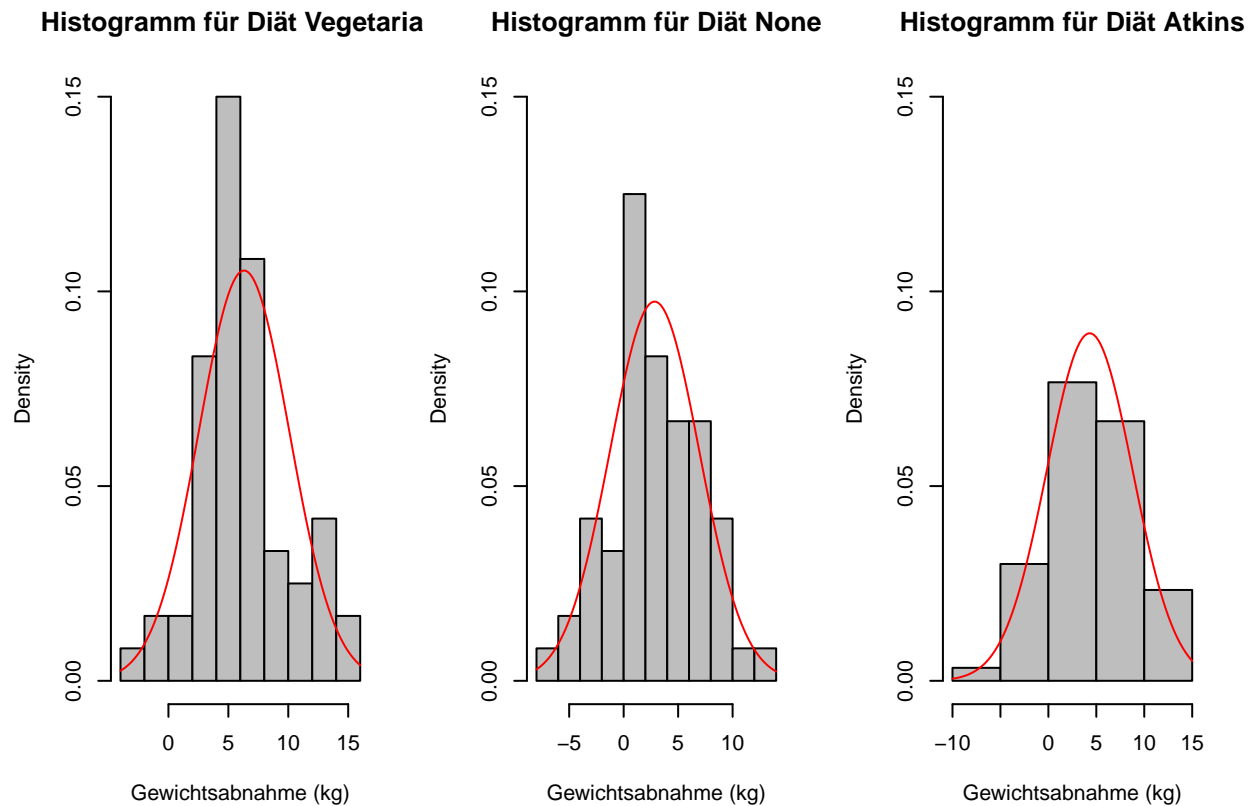
```
# Überprüfen der Voraussetzungen

# -- für Diet-Gruppen
unique_diets <- unique(weightloss$diet)

# Normalverteilung
par(mfrow = c(1, 3))
for(diet in unique_diets) {
  # Daten für die spezifische Diät
  diet_data <- weightloss$wloss[weightloss$diet == diet]

  # Histogramm erstellen
  hist(diet_data, probability = TRUE,
       main = paste("Histogramm für Diät", diet),
       xlab = "Gewichtsabnahme (kg)",
       col = "gray", border = "black",
       ylim = c(0, 0.15))

  # Normalverteilungskurve hinzufügen
  curve(dnorm(x, mean = mean(diet_data), sd = sd(diet_data)), add = TRUE, col = "red")
}
```



```

# Shapiro-Wilk-Test
for (diet in unique_diets) {
  print(shapiro.test(weightloss$wloss[weightloss$diet == diet]))
}

##
## Shapiro-Wilk normality test
##
## data: weightloss$wloss[weightloss$diet == diet]
## W = 0.96877, p-value = 0.1271
##
##
## Shapiro-Wilk normality test
##
## data: weightloss$wloss[weightloss$diet == diet]
## W = 0.98733, p-value = 0.7893
##
##
## Shapiro-Wilk normality test
##
## data: weightloss$wloss[weightloss$diet == diet]
## W = 0.98252, p-value = 0.5436

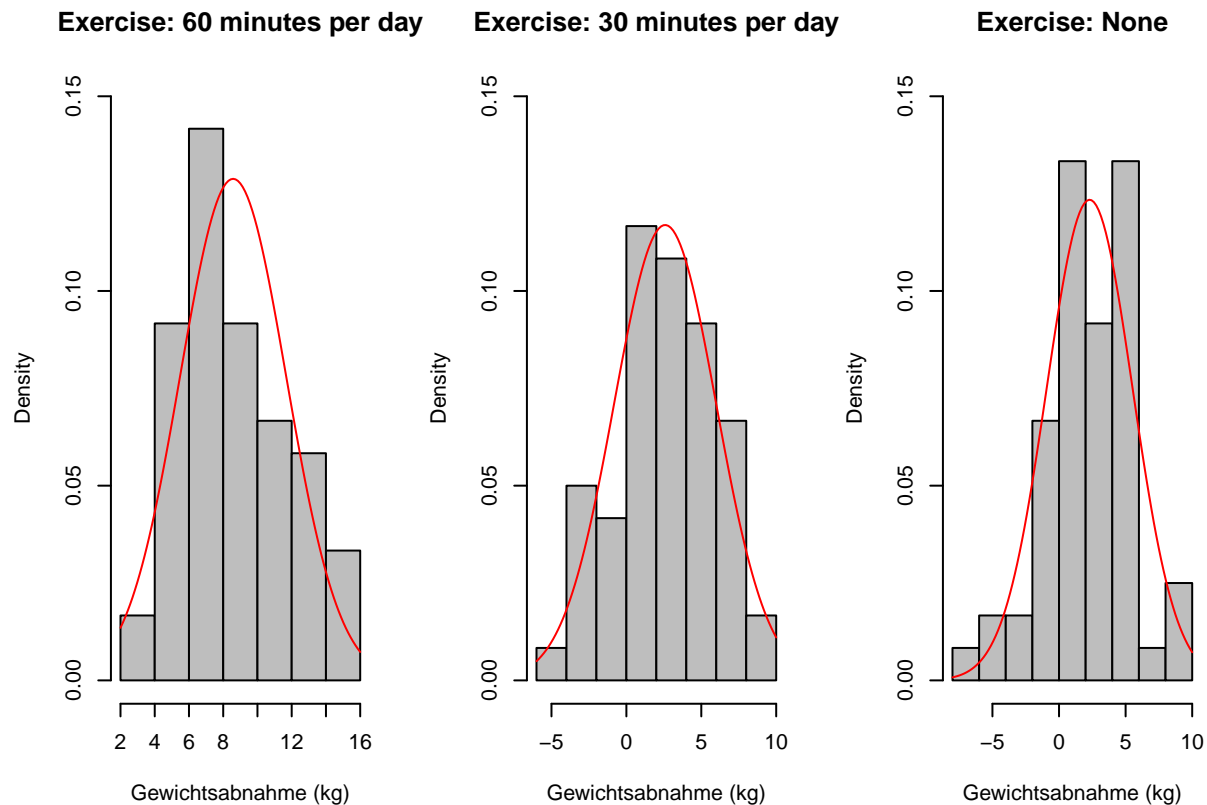
# -- für Activity-Gruppen
unique_activities <- unique(weightloss$exercise)

# Normalverteilung
par(mfrow = c(1, 3))
for(activity in unique_activities) {
  # Daten für die spezifische Aktivität
  activity_data <- weightloss$wloss[weightloss$exercise == activity]

  # Histogramm erstellen
  hist(activity_data, probability = TRUE,
        main = paste("Exercise:", activity),
        xlab = "Gewichtsabnahme (kg)",
        col = "gray", border = "black",
        ylim = c(0, 0.15))

  # Normalverteilungskurve hinzufügen
  curve(dnorm(x, mean = mean(activity_data), sd = sd(activity_data)), add = TRUE, col = "red")
}

```



```
# Shapiro-Wilk-Test
for (activity in unique_activities) {
  print(shapiro.test(weightloss$wloss[weightloss$exercise == activity]))
}
```

```
##
## Shapiro-Wilk normality test
##
## data: weightloss$wloss[weightloss$exercise == activity]
## W = 0.96325, p-value = 0.06792
##
## Shapiro-Wilk normality test
##
## data: weightloss$wloss[weightloss$exercise == activity]
## W = 0.98855, p-value = 0.8468
##
## Shapiro-Wilk normality test
##
## data: weightloss$wloss[weightloss$exercise == activity]
## W = 0.98445, p-value = 0.6417
```

```
# Varianzhomogenität (Heteroskedastizität)
```

```
# Levene-Test
leveneTest(wloss ~ diet, data = weightloss)

## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group  2  1.4859 0.2291
##      177
```

```
leveneTest(wloss ~ exercise, data = weightloss)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group  2  0.3179 0.7281
##      177
```

```
leveneTest(wloss ~ diet * exercise, data = weightloss)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group  8  0.7178 0.6756
##      171
```

```
# Bartlett-Test
bartlett.test(wloss ~ diet, data = weightloss)
```

```
##
## Bartlett test of homogeneity of variances
##
## data:  wloss by diet
## Bartlett's K-squared = 1.613, df = 2, p-value = 0.4464
```

```
bartlett.test(wloss ~ exercise, data = weightloss)
```

```
##
## Bartlett test of homogeneity of variances
##
## data:  wloss by exercise
## Bartlett's K-squared = 0.54989, df = 2, p-value = 0.7596
```

Die Überprüfung der Varianzhomogenität mittels Levene-Test und Bartlett-Test ergab keine signifikanten Unterschiede in den Varianzen zwischen den Diät- und Aktivitätsgruppen. Somit erfüllen die Daten die Voraussetzung der Varianzhomogenität für die Durchführung einer mehrfaktoriellen ANOVA. Es ist also statistisch angemessen, die Haupteffekte sowie den Interaktionseffekt der beiden Variablen zu analysieren.

```
# Durchführung der ANOVA
summary(aov(wloss~diet+exercise,data=weightloss))
```

```
##           Df Sum Sq Mean Sq F value    Pr(>F)
## diet       2   363.9    182.0    21.16 5.91e-09 ***
## exercise   2  1508.9     754.4    87.72 < 2e-16 ***
## Residuals 175  1505.2       8.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#mit Interaktionseffekte
```

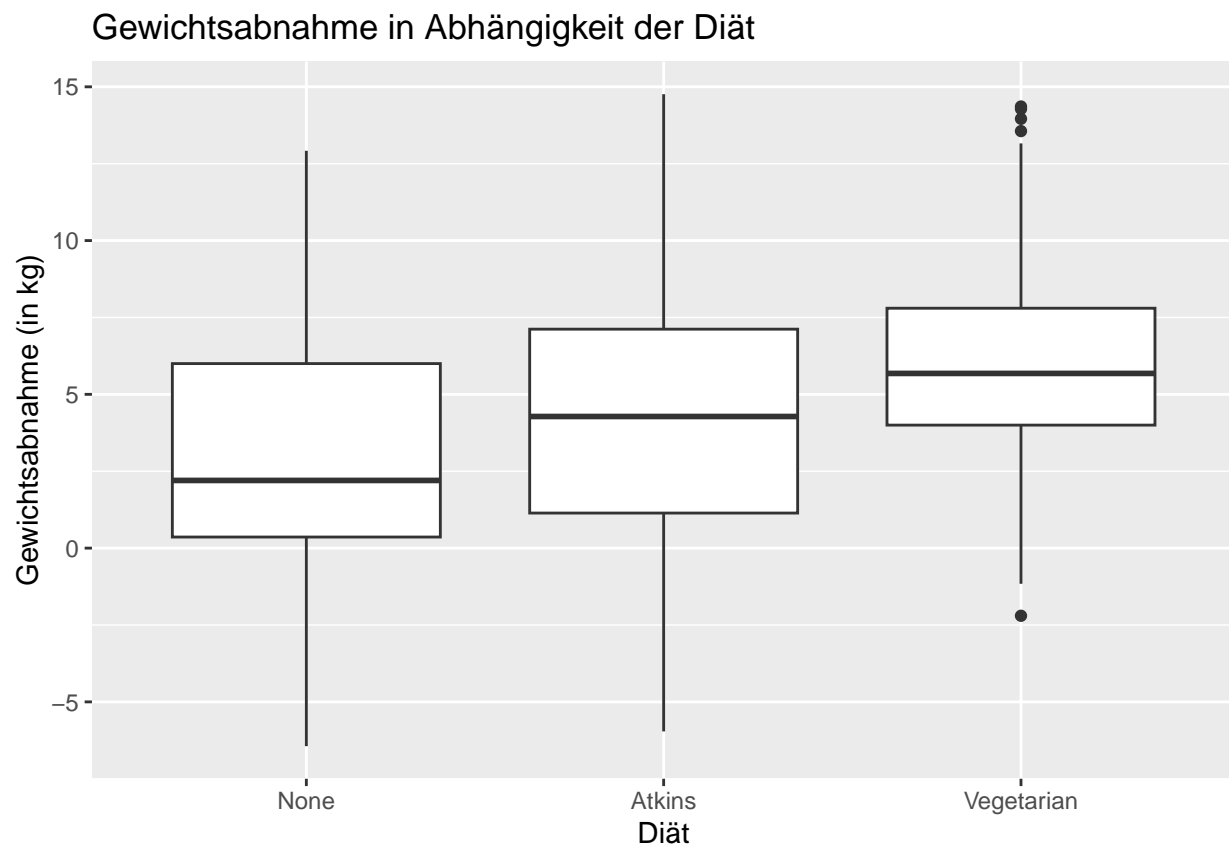
```
summary(aov(wloss~diet*exercise,data=weightloss))
```

```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## diet           2   363.9    182.0   21.124 6.33e-09 ***
## exercise       2  1508.9     754.4   87.589 < 2e-16 ***
## diet:exercise   4    32.3      8.1    0.937   0.444
## Residuals     171  1472.9      8.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ergebnisse der ANOVA

```
# Boxplot (diets)
```

```
ggplot(weightloss, aes(x = diet, y = wloss)) +
  geom_boxplot() +
  labs(title = "Gewichtsabnahme in Abhängigkeit der Diät",
       x = "Diät",
       y = "Gewichtsabnahme (in kg)")
```

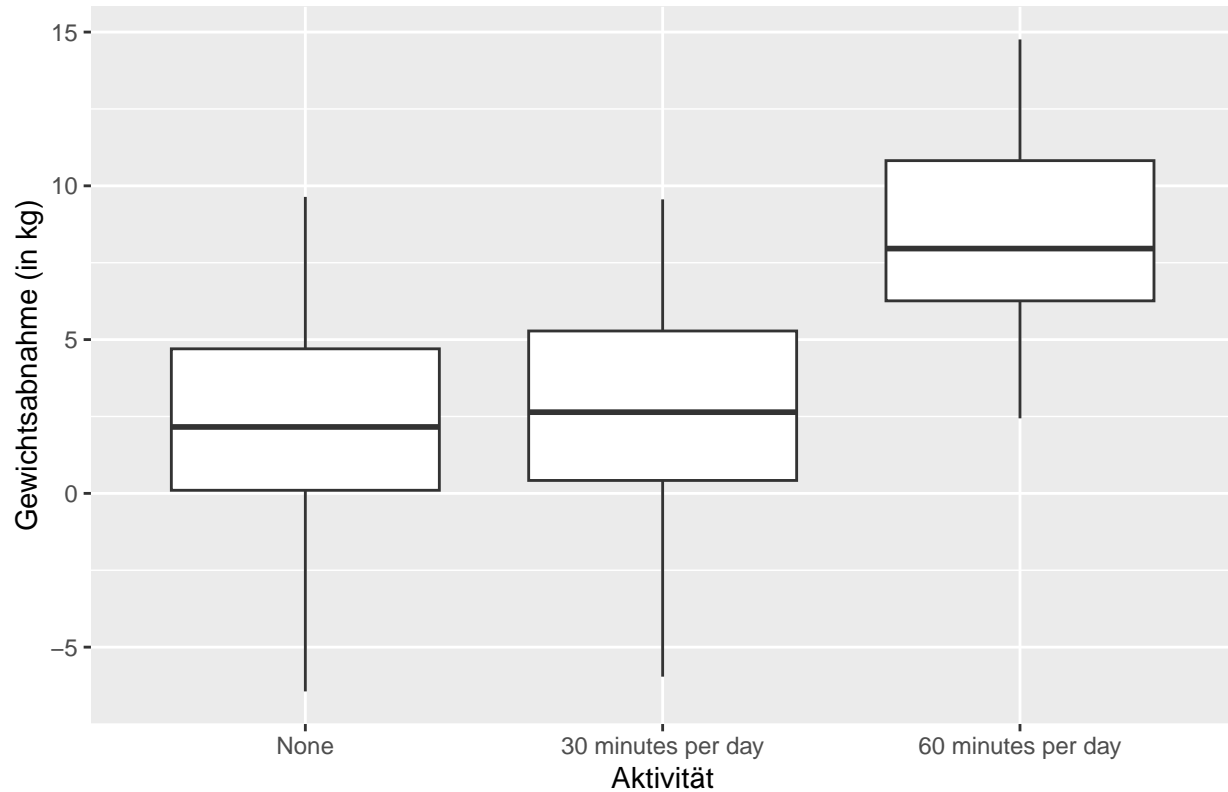


```
# Boxplot (activities)
```

```
ggplot(weightloss, aes(x = exercise, y = wloss)) +
  geom_boxplot() +
  labs(title = "Gewichtsabnahme in Abhängigkeit der Aktivität",
```

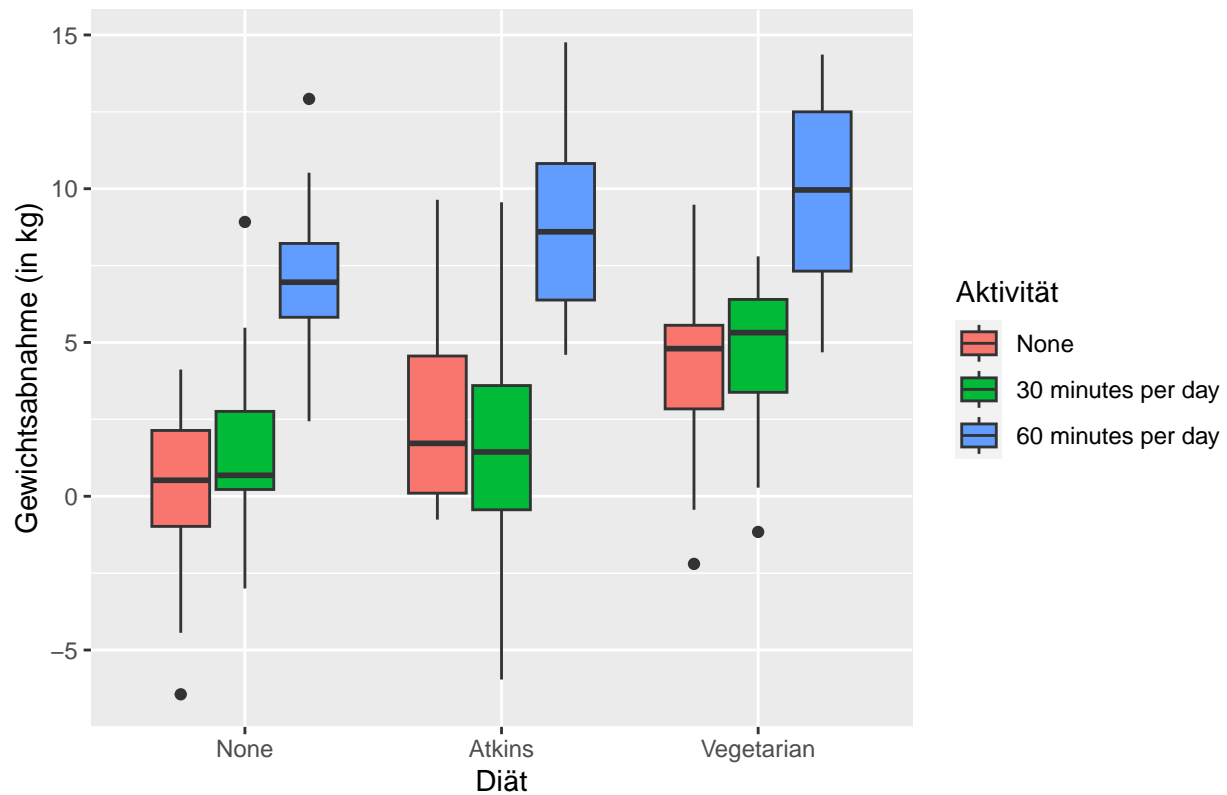
```
x = "Aktivität",  
y = "Gewichtsabnahme (in kg)"
```

Gewichtsabnahme in Abhängigkeit der Aktivität



```
# Boxplot (diets & activities)  
ggplot(weightloss, aes(x = diet, y = wloss, fill = exercise)) +  
  geom_boxplot() +  
  labs(title = "Gewichtsabnahme in Abhängigkeit der Diät und Aktivität",  
        x = "Diät",  
        y = "Gewichtsabnahme (in kg)",  
        fill = "Aktivität")
```

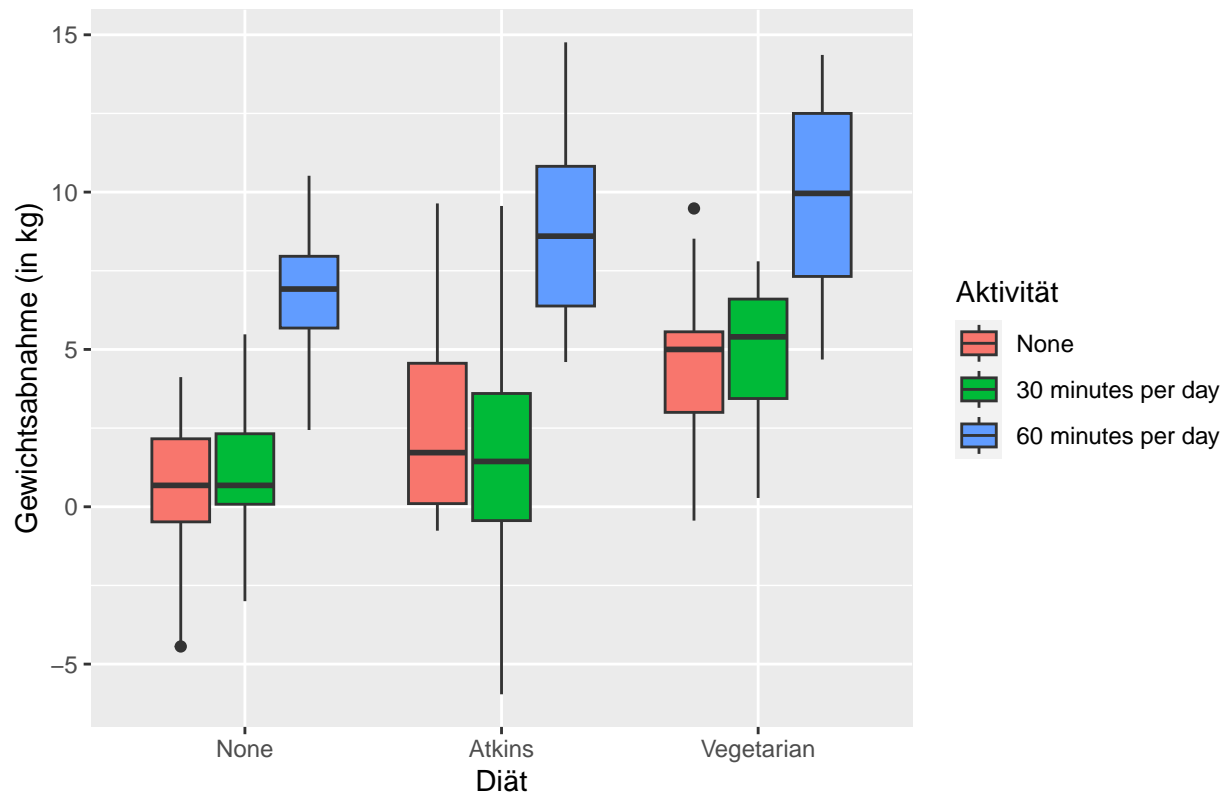

Gewichtsabnahme in Abhängigkeit der Diät und Aktivität



```
# Ausreißer bereinigen (median +/- 1.5 IQR)
library(dplyr)
weightloss_clean <- weightloss %>%
  group_by(diet, exercise) %>%
  mutate(lb = quantile(wloss, 0.25) - 1.5 * IQR(wloss),
         ub = quantile(wloss, 0.75) + 1.5 * IQR(wloss)) %>%
  filter(wloss >= lb & wloss <= ub)

# Boxplot (diets & activities)
ggplot(weightloss_clean, aes(x = diet, y = wloss, fill = exercise)) +
  geom_boxplot() +
  labs(title = "Gewichtsabnahme in Abhängigkeit der Diät und Aktivität (ausreißerbereinigt)",
       x = "Diät",
       y = "Gewichtsabnahme (in kg)",
       fill = "Aktivität")
```

Gewichtsabnahme in Abhängigkeit der Diät und Aktivität (ausreißerbereinigt)



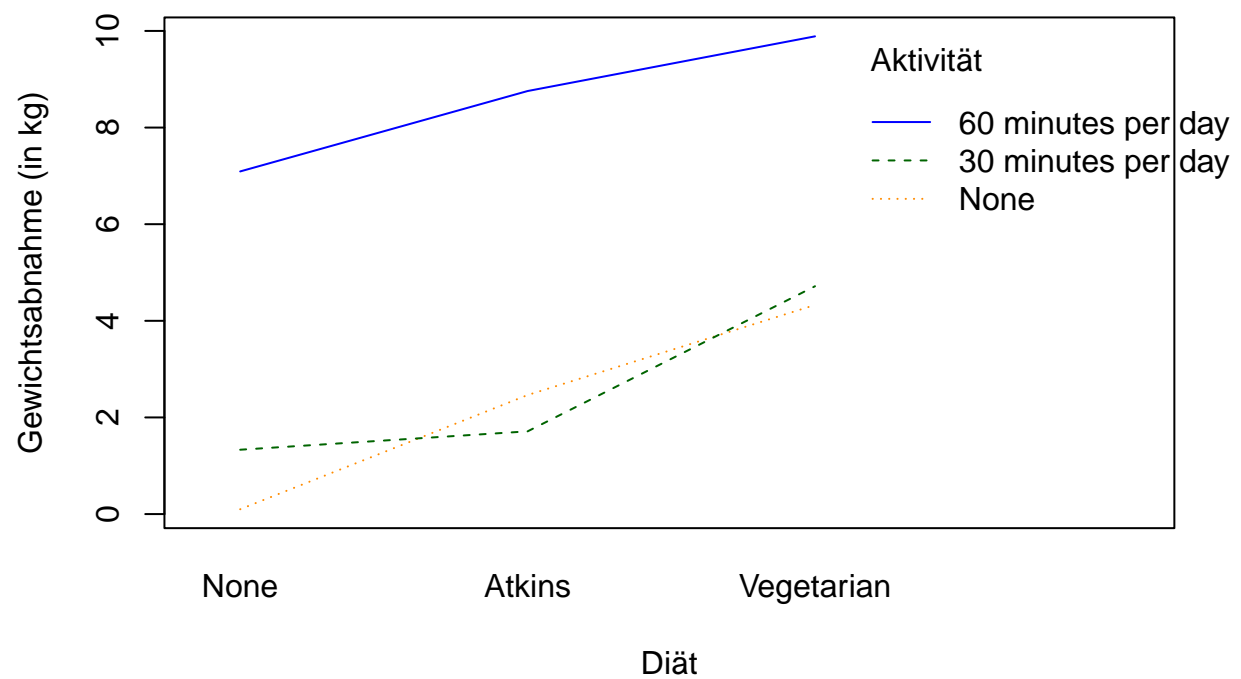
i) Gibt es signifikante Haupteffekte sowie Interaktionseffekte (inkl. Interaktions-Plot)?

Der Faktor Diet (Df=2) hat einen deutlichen Einfluss auf die Gewichtsabnahme mit einem F-Wert von 21.124 und einem hochsignifikanten p-Wert (6.33e-09).

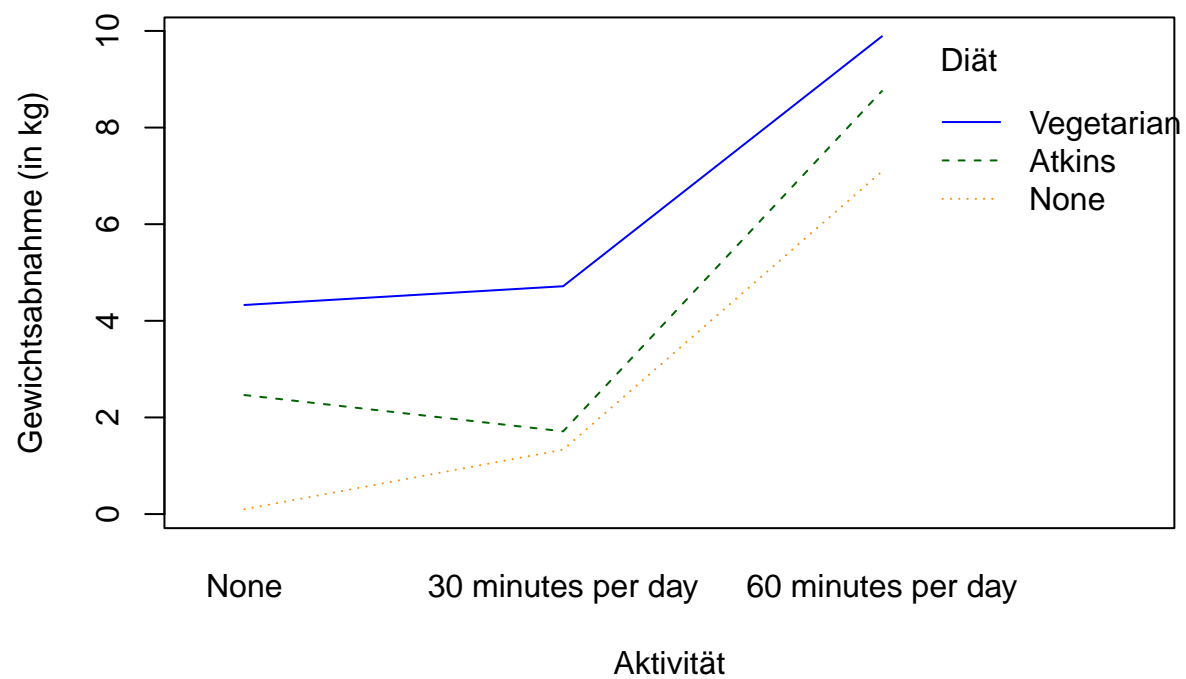
Das Bewegungsausmaß (Df=2) hatte einen noch stärkeren Einfluss auf die Gewichtsabnahme mit einem F-Wert von 87.589 und einem ebenfalls hochsignifikanten p-Wert ($< 2e-16$).

Die Interaktion zwischen Diät und Bewegung (Df = 4) war nicht signifikant mit einem F-Wert von 0.937 und einem p-Wert von 0.444. Dies bedeutet, dass die Wechselwirkung zwischen Diät und Bewegung keinen signifikanten Beitrag zur Erklärung der Gewichtsabnahme liefert.

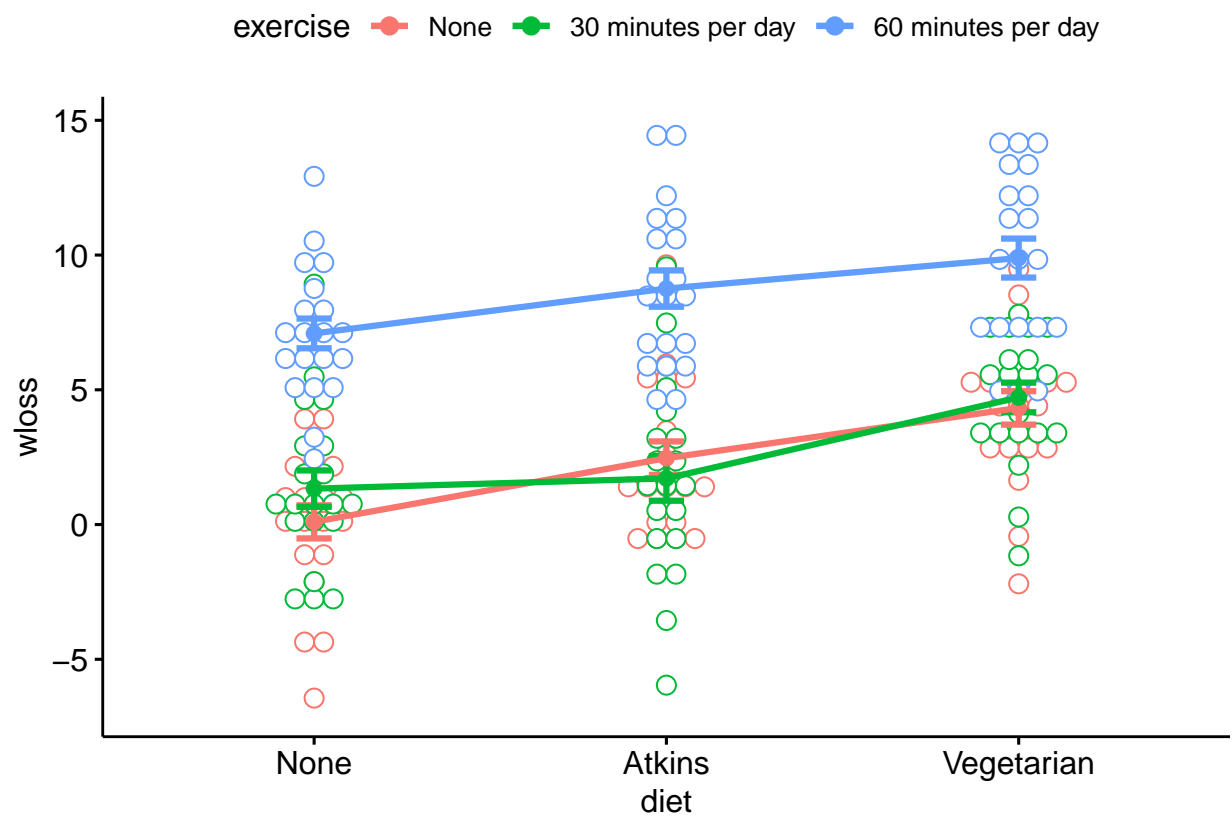
```
# Interaktionsplots (simple)
ip_act <- interaction.plot(weightloss$diet, weightloss$exercise, weightloss$wloss,
  col = c("darkorange", "darkgreen", "blue"),
  pch = c(1, 2, 3), fixed = FALSE, xlab = "Diät",
  ylab = "Gewichtsabnahme (in kg)", trace.label = "Aktivität")
```



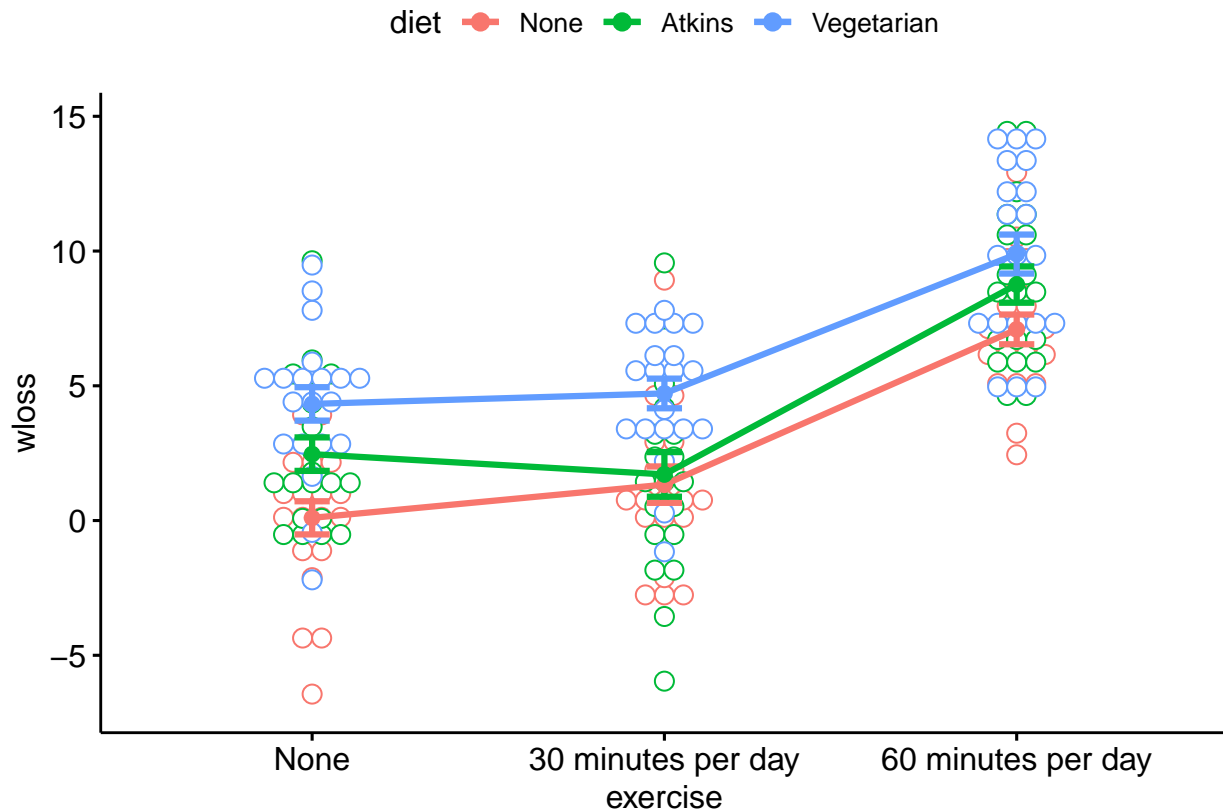
```
ip_diet <- interaction.plot(weightloss$exercise, weightloss$diet, weightloss$wloss,
  col = c("darkorange", "darkgreen", "blue"),
  pch = c(1, 2, 3), fixed = FALSE, xlab = "Aktivität",
  ylab = "Gewichtsabnahme (in kg)", trace.label = "Diät")
```



```
# Interaktionsplot (ggpubr)
library(ggpubr)
ggline(weightloss, x = "diet", y = "wloss", color = "exercise", size = 1.1,
       add = c("mean_se", "dotplot"))
```



```
ggline(weightloss, x = "exercise", y = "wloss", color = "diet", size = 1.1,
       add = c("mean_se", "dotplot"))
```



Interaktionsplot 1: Dieser Plot zeigt die Gewichtsabnahme aufgeteilt nach Diättyp und gruppiert nach Aktivitätsniveau. Die drei Linien repräsentieren die unterschiedlichen Aktivitätsniveaus. Auffallend ist hier, dass sich die Linien für keine Aktivität und die des mittleren Aktivitätsniveaus bei der Atkins-Diet kreuzen. Dies könnte auf eine leichte semidisordinale Interaktion, in diesem Fall hindeuten. Allerdings war die Interaktion in der ANOVA statistisch nicht signifikant ($p = 0.444$), somit ist diese visuell wahrgenommene Interaktion möglicherweise nicht stark genug, um statistische Signifikanz zu erreichen.

Interaktionsplot 2: Auf diesem Plot ist die Gewichtsabnahme aufgeteilt nach Aktivitätsniveau und gruppiert nach Diättyp. Die drei Linien repräsentieren die unterschiedlichen Diättypen. Die Linien zeigen sehr ähnliche Trends auf, und kreuzen sich nicht. Dies unterstützt das ANOVA-Ergebnis, dass keine signifikante Interaktion vorliegt. Die Linie für "60 minutes per day" zeigt durchwegs die höchsten Werte, was die starke Wirkung eines erhöhten Aktivitätsniveaus auf die Gewichtsabnahme unterstreicht, unabhängig von der gewählten Diät.

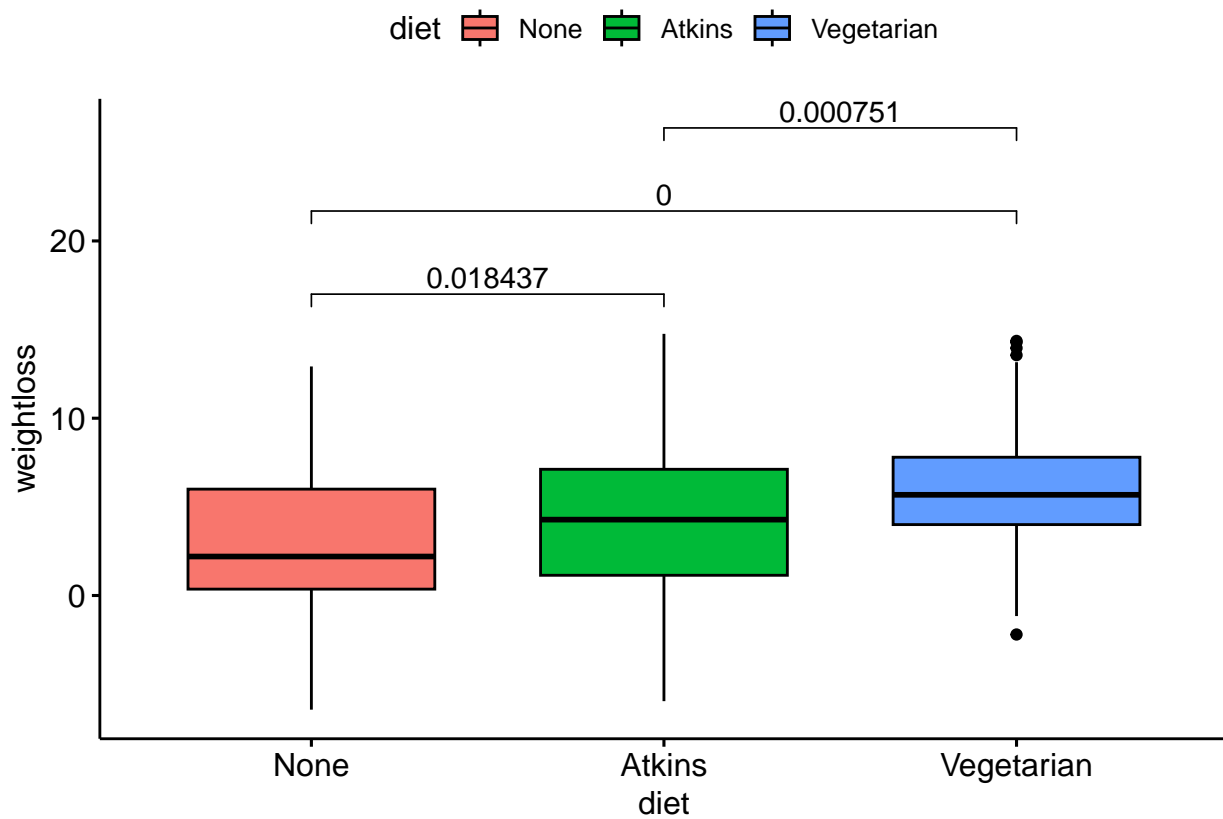
Um die Unterschiede zwischen den Gruppen genauer zu untersuchen, werden folglich Post-hoc-Tests durchgeführt. Hierfür eignen sich beispielsweise der Tukey's HSD-Test.

```
# Post-hoc Tests
pht_diet <- TukeyHSD(aov(wloss~diet*exercise,data=weightloss),which="diet")
pht_act <- TukeyHSD(aov(wloss~diet*exercise,data=weightloss),which="exercise")
```

```
# Darstellung der Post-hoc Test Ergebnisse
stat.test_diet <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "Atkins", "None", round(pht_diet$diet[1,4],6),
  "Vegetarian", "None", round(pht_diet$diet[2,4],6),
  "Vegetarian", "Atkins", round(pht_diet$diet[3,4],6))
```

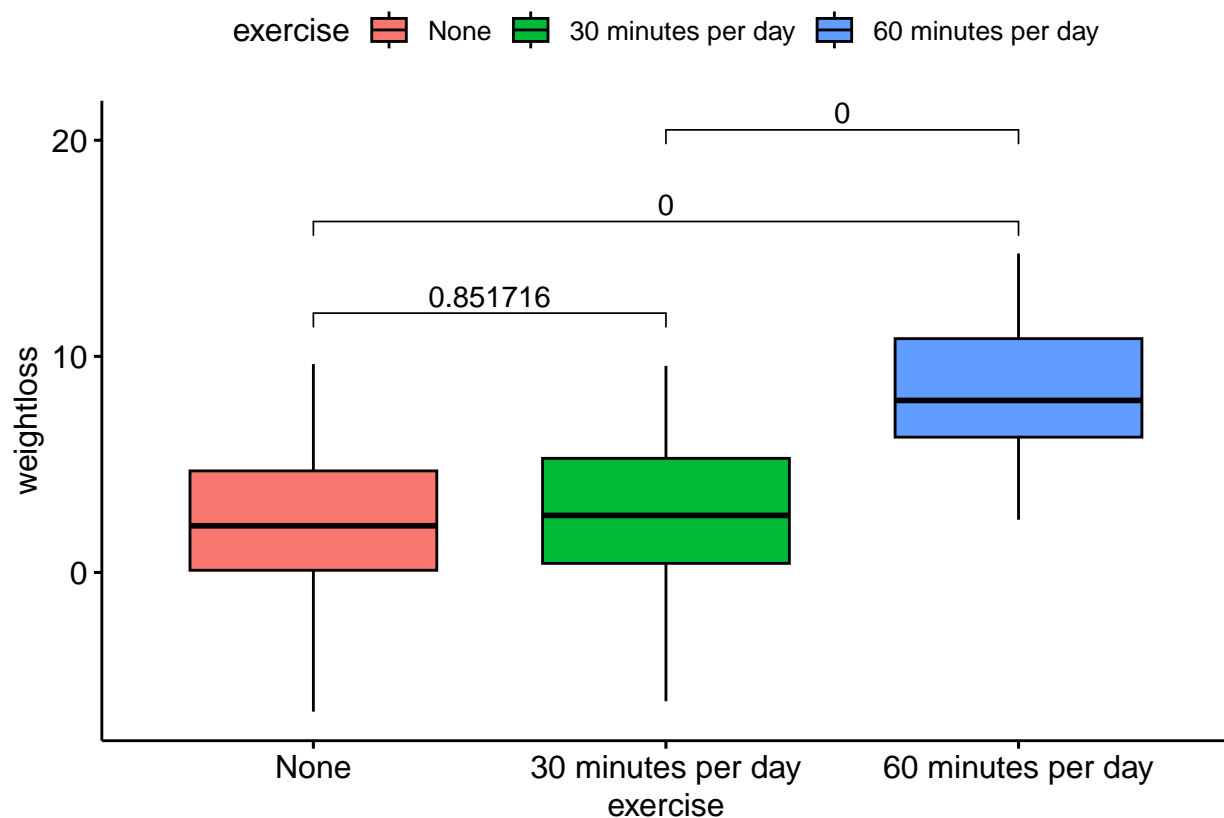
```
# Boxplot
```

```
ggboxplot(weightloss, x = "diet", y = "wloss",
  fill = "diet",
  palette = "Okabe-Ito",
  ylab = "weightloss") +
  stat_pvalue_manual(stat.test_diet,
    y.position = 17, step.increase = 0.2,
    label = "p.adj")
```



```
stat.test_act <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "30 minutes per day", "None", round(pht_act$exercise[1,4],6),
  "60 minutes per day", "None", round(pht_act$exercise[2,4],6),
  "60 minutes per day", "30 minutes per day", round(pht_act$exercise[3,4],6))

# Boxplot
ggboxplot(weightloss, x = "exercise", y = "wloss",
  fill = "exercise",
  palette = "Okabe-Ito",
  ylab = "weightloss") +
  stat_pvalue_manual(stat.test_act,
    y.position = 12, step.increase = 0.2,
    label = "p.adj")
```



Interpretation: Eine 2x3 ANOVA wurde durchgeführt, um den Einfluss unterschiedlicher Diäten (Keine, Atkins, Vegetarisch) und Aktivitätsniveaus (Keine, 30 min/Tag, 60 min/Tag) auf die Gewichtsabnahme zu untersuchen. Es wurde ein signifikanter Haupteffekt des Faktors Diät [$F(2,171)=21.12, p < 0.01$], als auch ein signifikanter Haupteffekt des Faktors Aktivitätsniveau [$F(2,171)=87.59, p < 0.01$] festgestellt werden. Es konnte kein signifikanter Interaktionseffekt zwischen Diät und Aktivitätsniveau nachgewiesen werden [$F(4,171) = 0.937; p = 0.444$].

Post-hoc-Tests Analysen zeigten signifikanten Unterschiede zwischen den einzelnen Diätgruppen und Aktivitätsniveaus.

Personen, die eine vegetarische Diät befolgten, verloren durchschnittlich 3,47 kg mehr als jene ohne spezielle Diät und um 2 kg mehr als Personen, die der Atkins-Diät folgen. Diese Unterschiede sind statistisch signifikant, was die vegetarische Diät als besonders effektiv herausstellt. Die Atkins-Diät zeigt eine überlegene Wirkung im Vergleich zu keiner Diät mit einer signifikant höheren mittleren Gewichtsabnahme von 1,47 kg.

Die tägliche Dauer der körperlichen Aktivität spielt ebenfalls eine entscheidende Rolle bei der Gewichtsreduktion. Die Analyse ergab, dass 60 Minuten tägliche Bewegung zu einer signifikant höheren Gewichtsabnahme von 6,28 kg im Vergleich zu keiner Bewegung führte. Selbst im Vergleich zu 30 Minuten täglicher Aktivität ist das längere Training mit einer zusätzlichen Gewichtsabnahme von fast 6 kg deutlich effektiver. Im Gegensatz dazu zeigt die geringere Dauer von 30 Minuten keine signifikante Wirkung gegenüber keiner Bewegung.

5 Mehrfaktorielle ANOVA [2P]

Verwenden Sie den erneut den `Framingham.sav` Datensatz. Analysieren Sie den systolischen Blutdruck `sysbp` abhängig von Geschlecht und Bildungsstufe. Achten Sie auf Ausreißer und fehlende Daten (`NaN`, `NA's`).

i) Überprüfen Sie die Voraussetzungen.

```
load_source()

# Open a new graphics device
dev.new()

# Load and clean data. Handle factor variables
framingham <- load_and_prepare_framingham("UE3 Daten/Framingham.sav")

# Check requirements for ANOVA

# -- For education levels

test_normality_of_groups(framingham, "educ", "sysbp")
```

```
##
## Shapiro-Wilk normality test
##
## data: data$sysbp[data$educ==1]
## W = 0.94267, p-value < 2.2e-16
##
##
## Shapiro-Wilk normality test
##
## data: data$sysbp[data$educ==2]
## W = 0.92898, p-value < 2.2e-16
##
##
## Shapiro-Wilk normality test
##
## data: data$sysbp[data$educ==3]
## W = 0.92049, p-value < 2.2e-16
##
##
## Shapiro-Wilk normality test
##
## data: data$sysbp[data$educ==4]
## W = 0.93238, p-value = 2.203e-13
```

```
# -- For sex

test_normality_of_groups(framingham, "sex", "sysbp")
```

```
##
## Shapiro-Wilk normality test
##
## data: data$sysbp[data$sex==1]
## W = 0.94294, p-value < 2.2e-16
##
##
## Shapiro-Wilk normality test
##
```

```
## data: data$sysbp[data$sex==2]
## W = 0.93418, p-value < 2.2e-16
```

```
# Variance inhomogeneity (Heteroscedasticity)
```

```
# Levene test
```

```
leveneTest(sysbp ~ educ, data = framingham)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value    Pr(>F)
## group   3  7.1555 8.656e-05 ***
##      3822
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
leveneTest(sysbp ~ sex, data = framingham)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value    Pr(>F)
## group   1 63.979 1.652e-15 ***
##      3824
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
leveneTest(sysbp ~ educ * sex, data = framingham)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value    Pr(>F)
## group   7 13.607 < 2.2e-16 ***
##      3818
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Bartlett test
```

```
bartlett.test(sysbp ~ educ, data = framingham)
```

```
##
## Bartlett test of homogeneity of variances
##
## data: sysbp by educ
## Bartlett's K-squared = 26.286, df = 3, p-value = 8.311e-06
```

```
bartlett.test(sysbp ~ sex, data = framingham)
```

```
##
## Bartlett test of homogeneity of variances
##
## data: sysbp by sex
## Bartlett's K-squared = 101.81, df = 1, p-value < 2.2e-16
```

Die Histogramme der Blutdruck-Werte für die verschiedenen Bildungsstufen zeigen, dass die Verteilung der Blutdruck-Werte für jede Bildungsstufe nicht normal ist. Der Shapiro-Wilk-Test bestätigt diese Annahme, da die p-Werte für alle Bildungsstufen kleiner als 0,05 sind. Außerdem zeigt der Levene Test, dass die Varianz aller Gruppen nicht gleich ist. Dasselbe gilt für die Gruppen nach Geschlecht. Daher sind die Voraussetzungen einer ANOVA verletzt. Allerdings reagiert der F-Test auf Abweichungen nicht so empfindlich, wenn 1) die Gruppengrößen etwa gleich sind 2) und die Gruppen hinreichend groß sind.

```
table(framingham$educ)
```

```
##
##      1      2      3      4
## 1619 1127  631  449
```

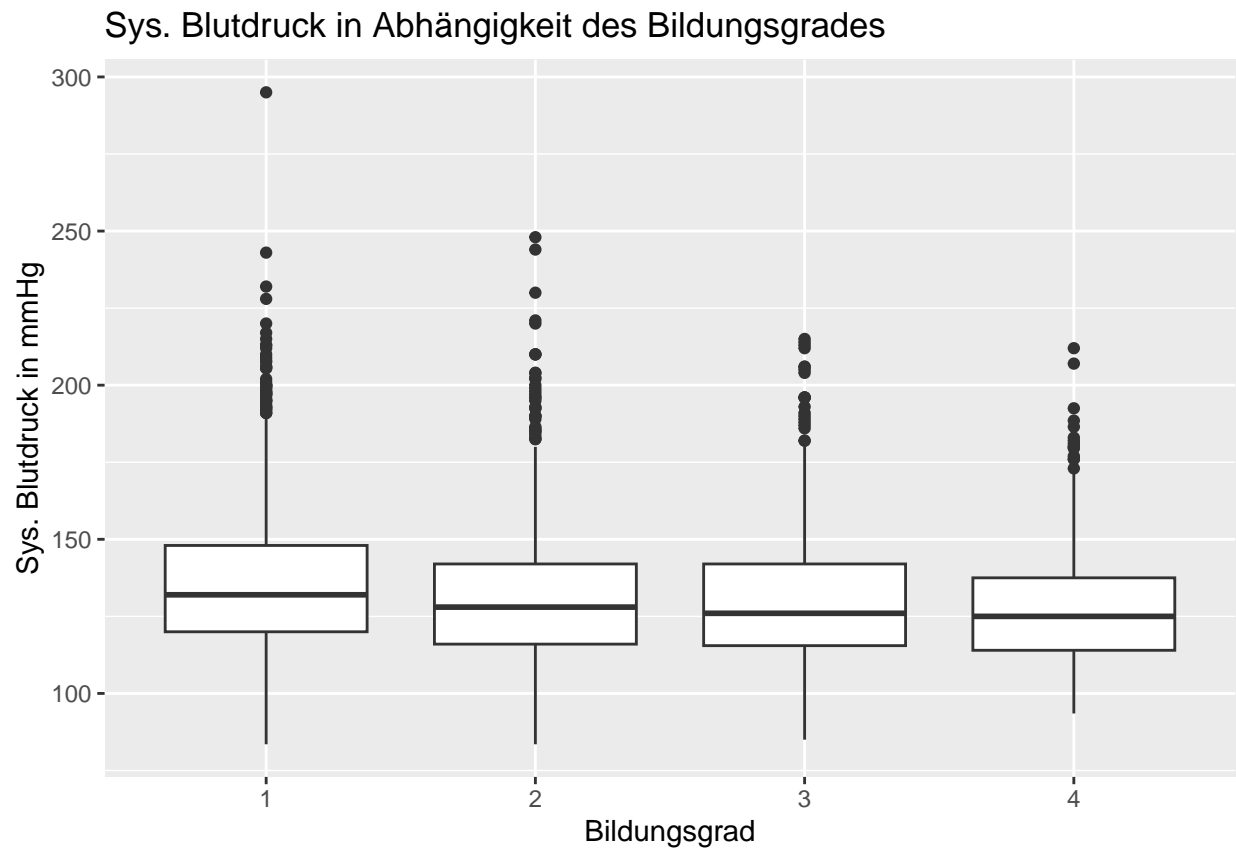
```
table(framingham$sex)
```

```
##
##      1      2
## 1731 2095
```

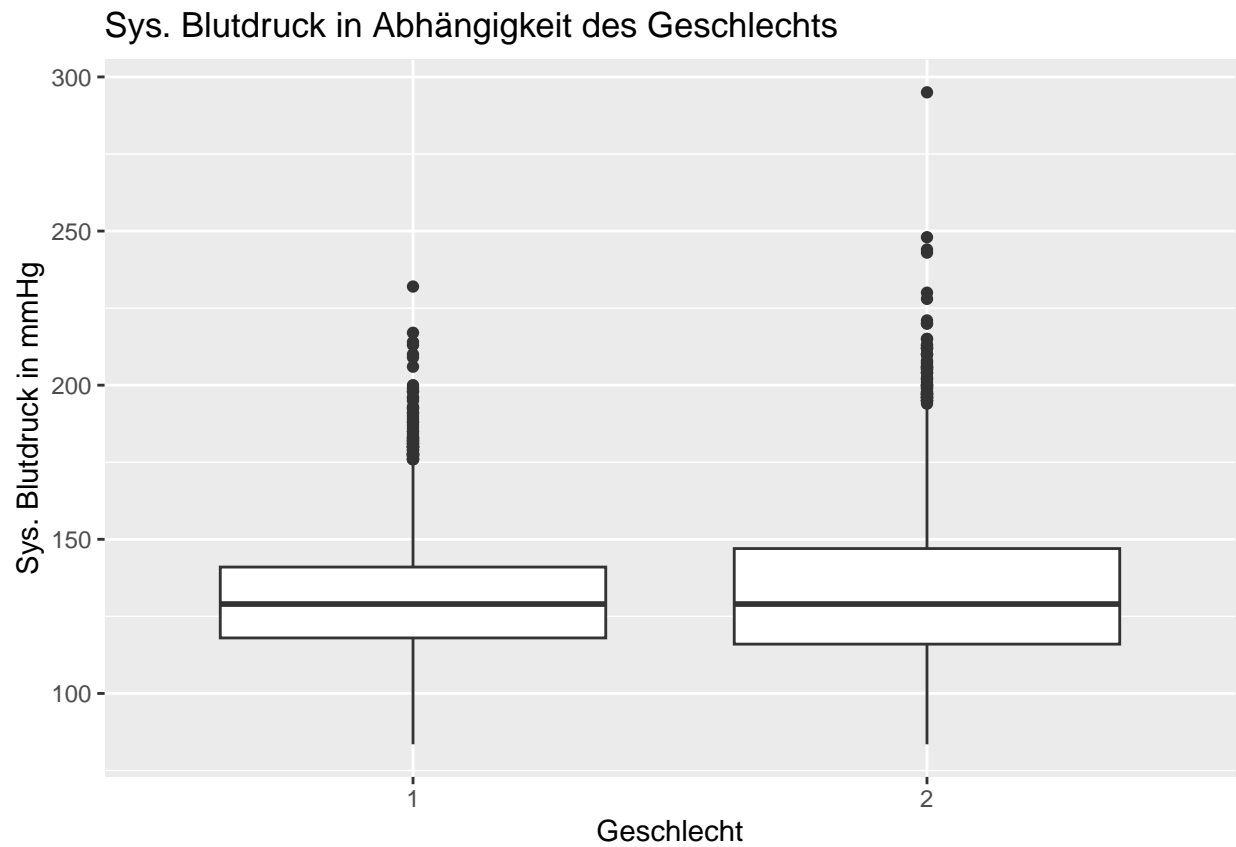
Da Bedingung 1 nicht erfüllt ist, kommt keine zweifaktorielle ANOVA infrage. Stattdessen wird für jeden Faktor ein Kruskal Wallis Test angewandt. Für die Untersuchung des Einflusses der Kombination von Geschlecht und Bildungsgrad wird ein Scheirer-Ray-Hare-Test durchgeführt.

i) Verwenden Sie eine geeignete grafische Darstellung der Daten / Ergebnisse.

```
# Boxplot education levels
ggplot(framingham, aes(x = educ, y = sysbp)) +
  geom_boxplot() +
  labs(title = "Sys. Blutdruck in Abhängigkeit des Bildungsgrades",
       x = "Bildungsgrad",
       y = "Sys. Blutdruck in mmHg")
```

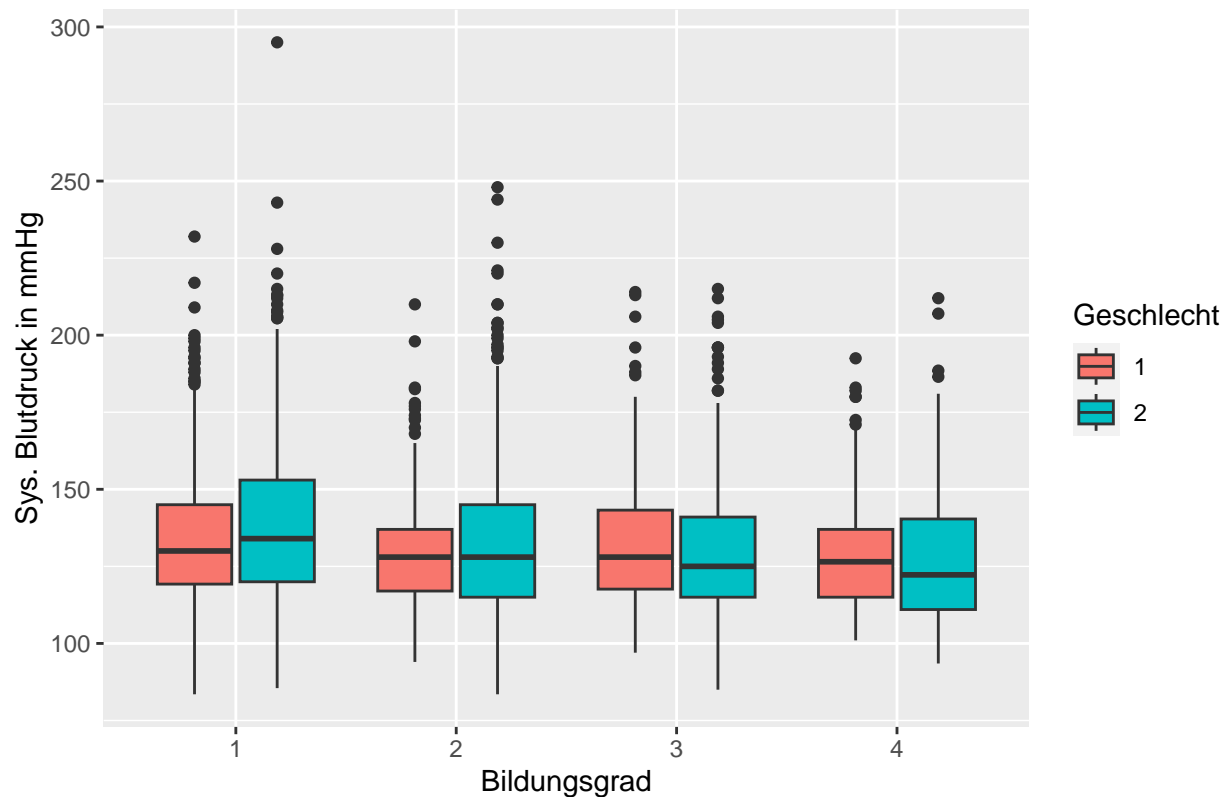


```
# Boxplot sexes
ggplot(framingham, aes(x = sex, y = sysbp)) +
  geom_boxplot() +
  labs(title = "Sys. Blutdruck in Abhängigkeit des Geschlechts",
        x = "Geschlecht",
        y = "Sys. Blutdruck in mmHg")
```



```
# Boxplot education levels & sexes
ggplot(framingham, aes(x = educ, y = sysbp, fill = sex)) +
  geom_boxplot() +
  labs(title = "Sys. Blutdruck in Abhängigkeit des Bildungsgrades und Geschlechts",
        x = "Bildungsgrad",
        y = "Sys. Blutdruck in mmHg",
        fill = "Geschlecht")
```

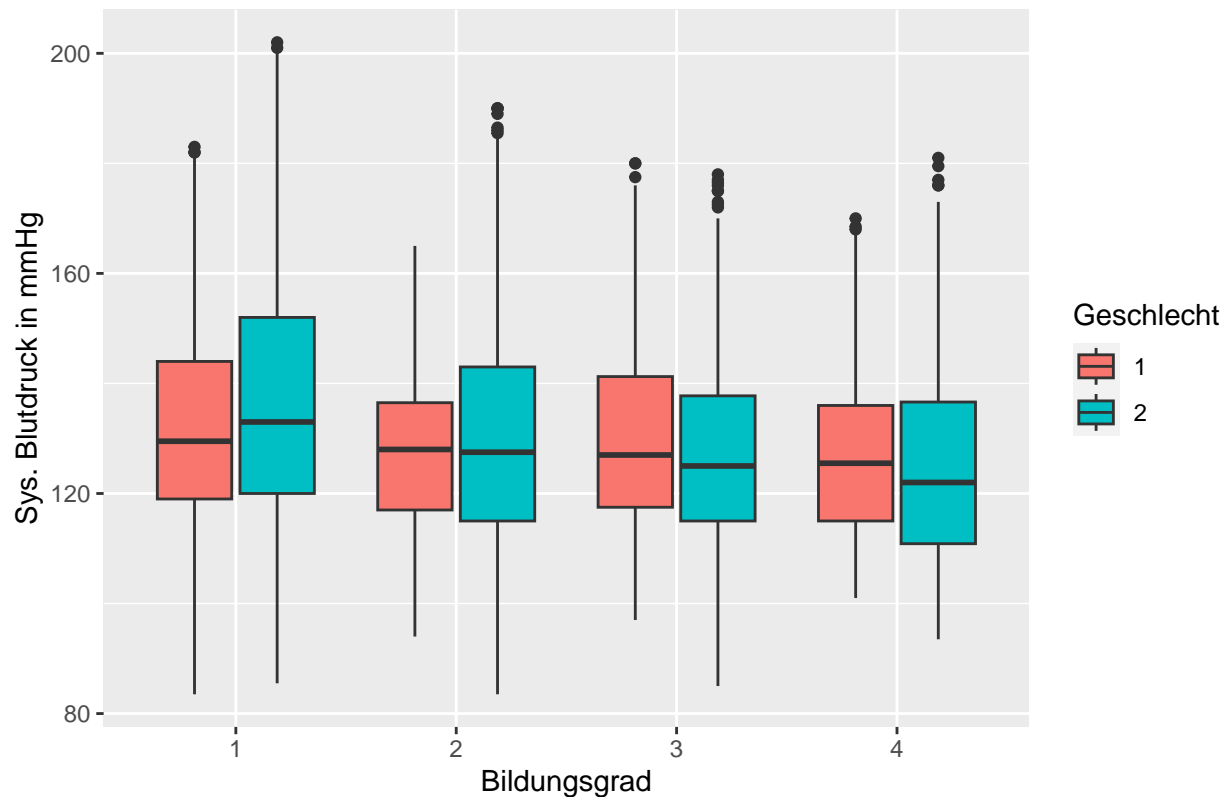
Sys. Blutdruck in Abhängigkeit des Bildungsgrades und Geschlechts



```
# Remove outliers (median +/- 1.5 IQR)
library(dplyr)
framingham_clean <- framingham %>%
  group_by(educ, sex) %>%
  mutate(lb = quantile(sysbp, 0.25) - 1.5 * IQR(sysbp),
         ub = quantile(sysbp, 0.75) + 1.5 * IQR(sysbp)) %>%
  filter(sysbp >= lb & sysbp <= ub)

# Boxplot (diets & activities)
ggplot(framingham_clean, aes(x = educ, y = sysbp, fill = sex)) +
  geom_boxplot() +
  labs(title = "Sys. Blutdruck in Abhängigkeit des Bildungsgrades und Geschlechts",
       x = "Bildungsgrad",
       y = "Sys. Blutdruck in mmHg",
       fill = "Geschlecht")
```

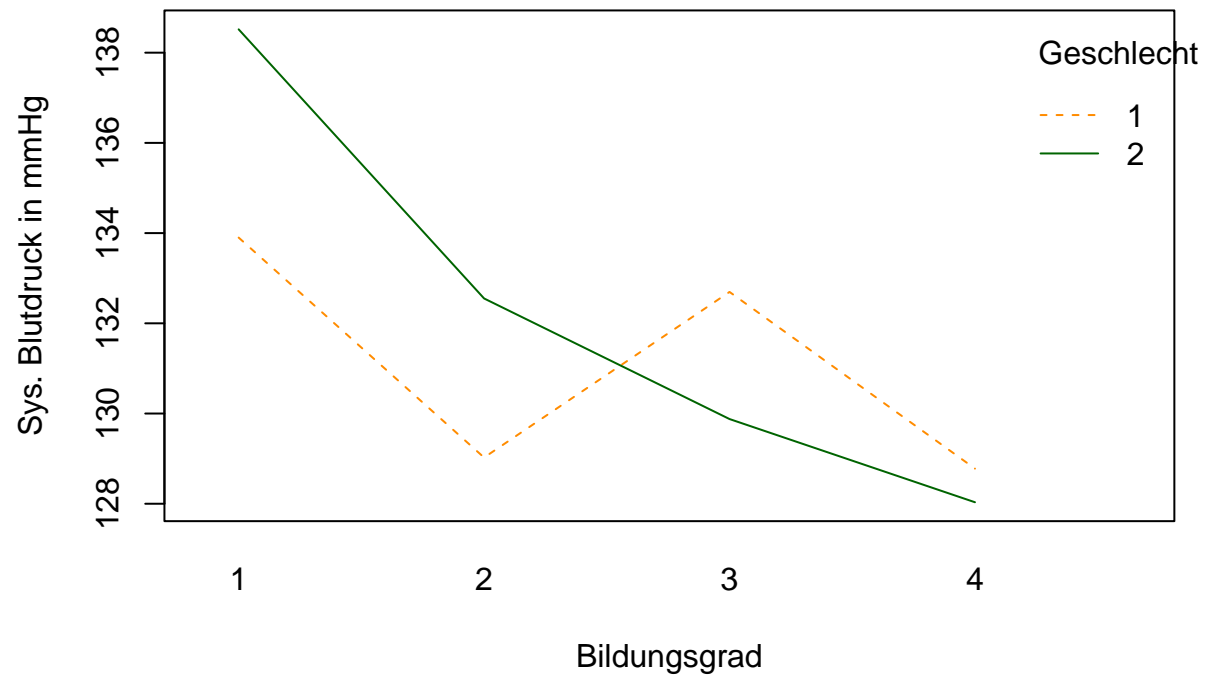
Sys. Blutdruck in Abhängigkeit des Bildungsgrades und Geschlechts



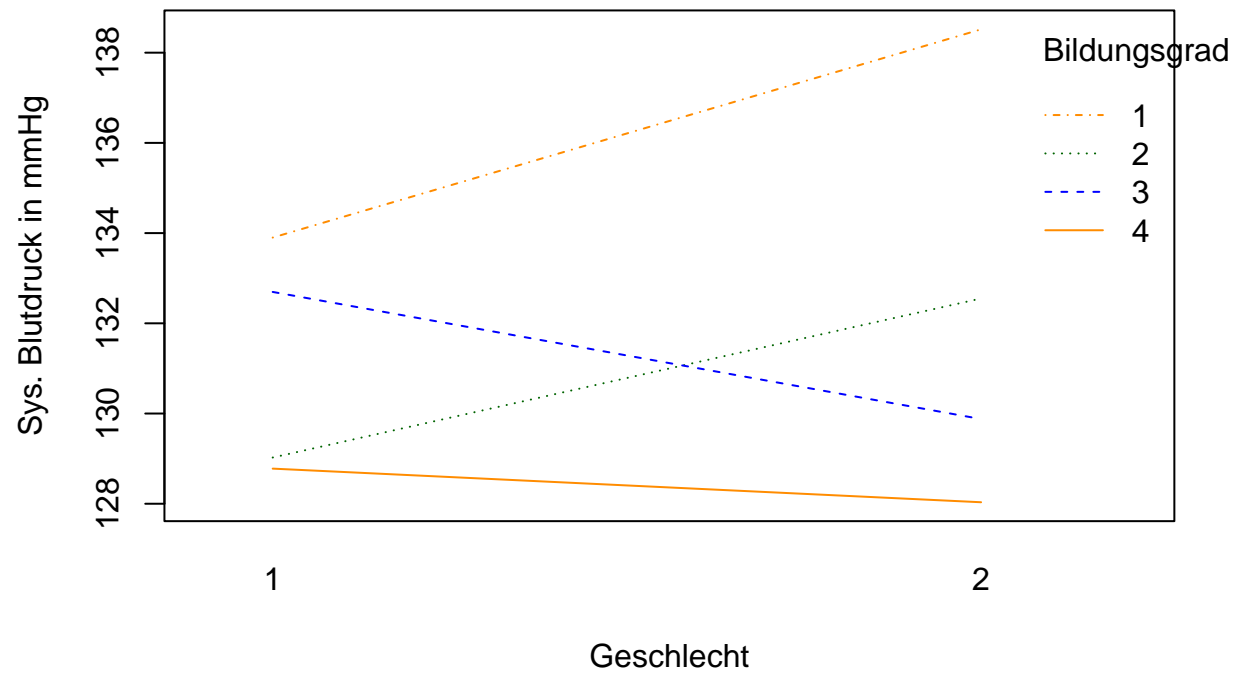
Die visuelle Analyse legt die Vermutung nahe, dass Geschlecht und Bildungsgrad wechselwirken, da die gedachten Verläufe durch die Mediane für Männer- und Frauengruppen sich zu überschneiden scheinen.

i) Gibt es signifikante Haupteffekte sowie Interaktionseffekte (inkl. Interaktions-Plot)?

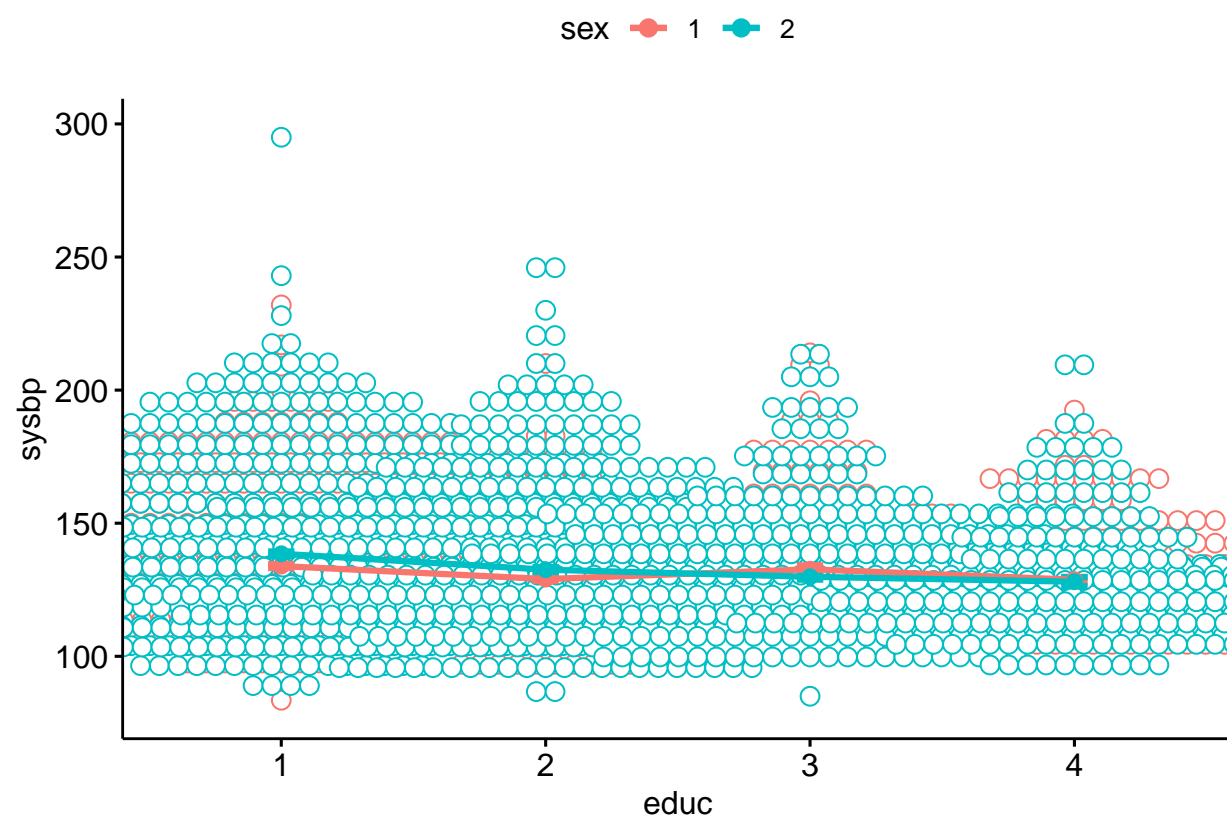
```
# Interaction plots (simple)
ip_educ <- interaction.plot(framingham$educ, framingham$sex, framingham$sysbp,
  col = c("darkorange", "darkgreen", "blue"),
  pch = c(1, 2, 3), fixed = FALSE, xlab = "Bildungsgrad",
  ylab = "Sys. Blutdruck in mmHg", trace.label = "Geschlecht")
```



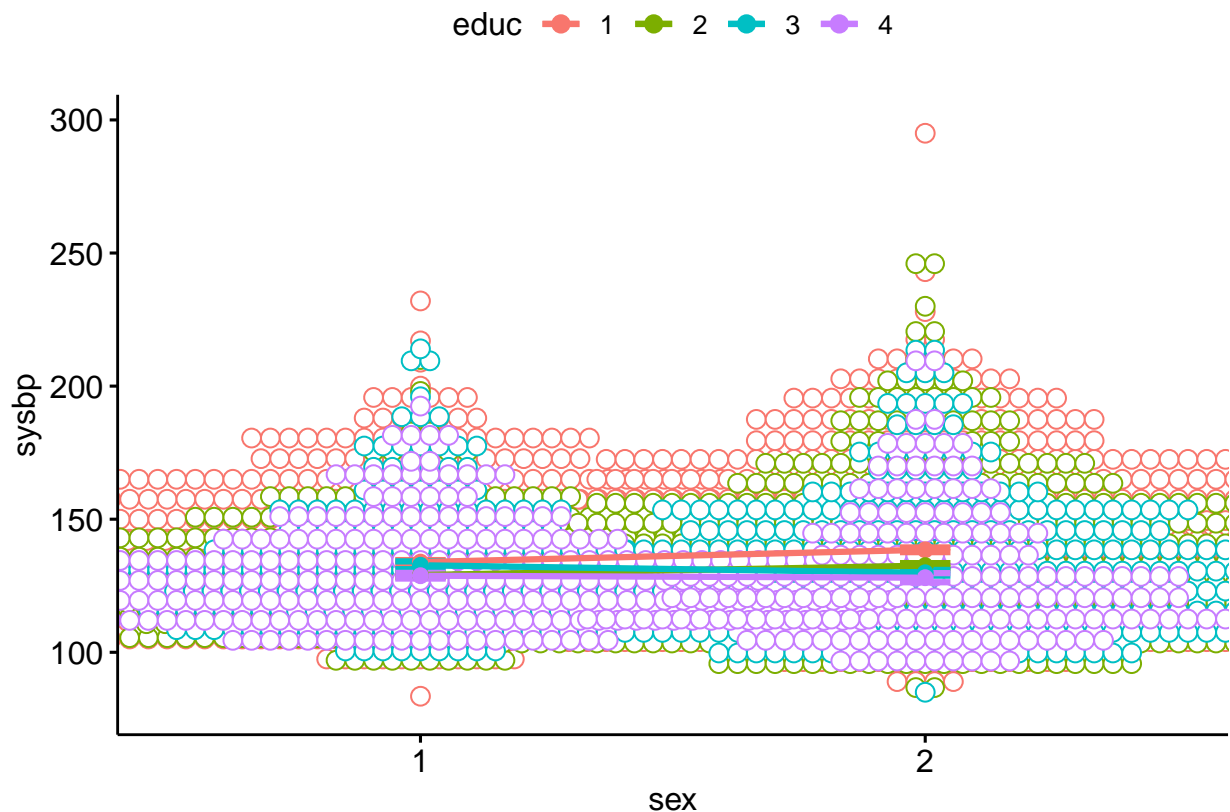
```
ip_sex <- interaction.plot(framingham$sex, framingham$educ, framingham$sysbp,
  col = c("darkorange", "darkgreen", "blue"),
  pch = c(1, 2, 3), fixed = FALSE, xlab = "Geschlecht",
  ylab = "Sys. Blutdruck in mmHg", trace.label = "Bildungsgrad")
```

```
# Interaction plots (ggpubr)
ggline(framingham, x = "educ", y = "sysbp", color = "sex", size = 1.1,
       add = c("mean_se", "dotplot"))
```



```
ggline(framingham, x = "sex", y = "sysbp", color = "educ", size = 1.1,
       add = c("mean_se", "dotplot"))
```



Die Interaktionsplots zeigen, dass die Wechselwirkung disordinal ist, da es in beiden Plots (x-Achse: Bildungsgrad bzw. Geschlecht) Überschneidungen in den Verläufen durch die Gruppenmittelwerte gibt.

```
# RESULTS NOT VALID! REQUIREMENTS NOT MET!
```

```
## Post hoc tests
# pht_educ <- TukeyHSD(aov(sysbp~educ*sex,data=framingham),which="educ")
# pht_sex <- TukeyHSD(aov(sysbp~educ*sex,data=framingham),which="sex")
```

```
# RESULTS NOT VALID! REQUIREMENTS NOT MET!
```

```
## Show the post hoc test results
# stat.test_educ <- tibble::tribble(
#   ~group1, ~group2, ~p.adj,
#   "1",      "2", round(pht$educ[1,4],6),
#   "1",      "3", round(pht$educ[2,4],6),
#   "1",      "4", round(pht$educ[3,4],6),
#   "2",      "3", round(pht$educ[4,4],6),
#   "2",      "4", round(pht$educ[5,4],6),
#   "3",      "4", round(pht$educ[6,4],6))
```

```
## Determine the maximum value of the 'bmi' column
# max_sysbp <- max(framingham$sysbp, na.rm = TRUE)
```

```
## Create an array of y positions starting from the maximum BMI value and incrementing by a small amount
# y_positions <- max_sysbp + seq(from = 1, to = 6, by = 1)
```

```

# # Boxplot
# ggboxplot(framingham, x = "educ", y = "sysbp",
#           fill = "educ",
#           palette = "Okabe-Ito",
#           ylab = "Sys. Blutdruck in mmHg") +
#   stat_pvalue_manual(stat.test_educ,
#                     y.position = y_positions,
#                     step.increase = 0.2,
#                     label = "p.adj")

```

```

# stat.test_sex <- tibble::tribble(
#   ~group1, ~group2, ~p.adj,
#   "1",      "2", round(pht_sex$sex[1,4],6))

```

```

# # Boxplot
# ggboxplot(framingham, x = "sex", y = "sysbp",
#           fill = "sex",
#           palette = "Okabe-Ito",
#           ylab = "Sys. Blutdruck in mmHg") +
#   stat_pvalue_manual(stat.test_sex,
#                     y.position = max_sysbp,
#                     label = "p.adj")

```

```

# Perform Kruskal-Wallis test for educ
kruskal_educ <- kruskal.test(sysbp ~ educ, data = framingham)
kruskal_educ

```

```

##
## Kruskal-Wallis rank sum test
##
## data: sysbp by educ
## Kruskal-Wallis chi-squared = 69.133, df = 3, p-value = 6.546e-15

```

```

# Post hoc tests

```

```

# Perform pairwise Wilcoxon tests
wilcox_educ <- pairwise.wilcox.test(framingham$sysbp, framingham$educ, p.adjust.method = "BH")

```

```

# Extract p-values
p_values_educ <- wilcox_educ$p.value

```

```

# Create data frames for manual p-value annotation
stat.test_educ <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "1",      "2", round(p_values_educ[1],6),
  "1",      "3", round(p_values_educ[2],6),
  "1",      "4", round(p_values_educ[3],6),
  "2",      "3", round(p_values_educ[5],6),
  "2",      "4", round(p_values_educ[6],6),
  "3",      "4", round(p_values_educ[9],6))

```

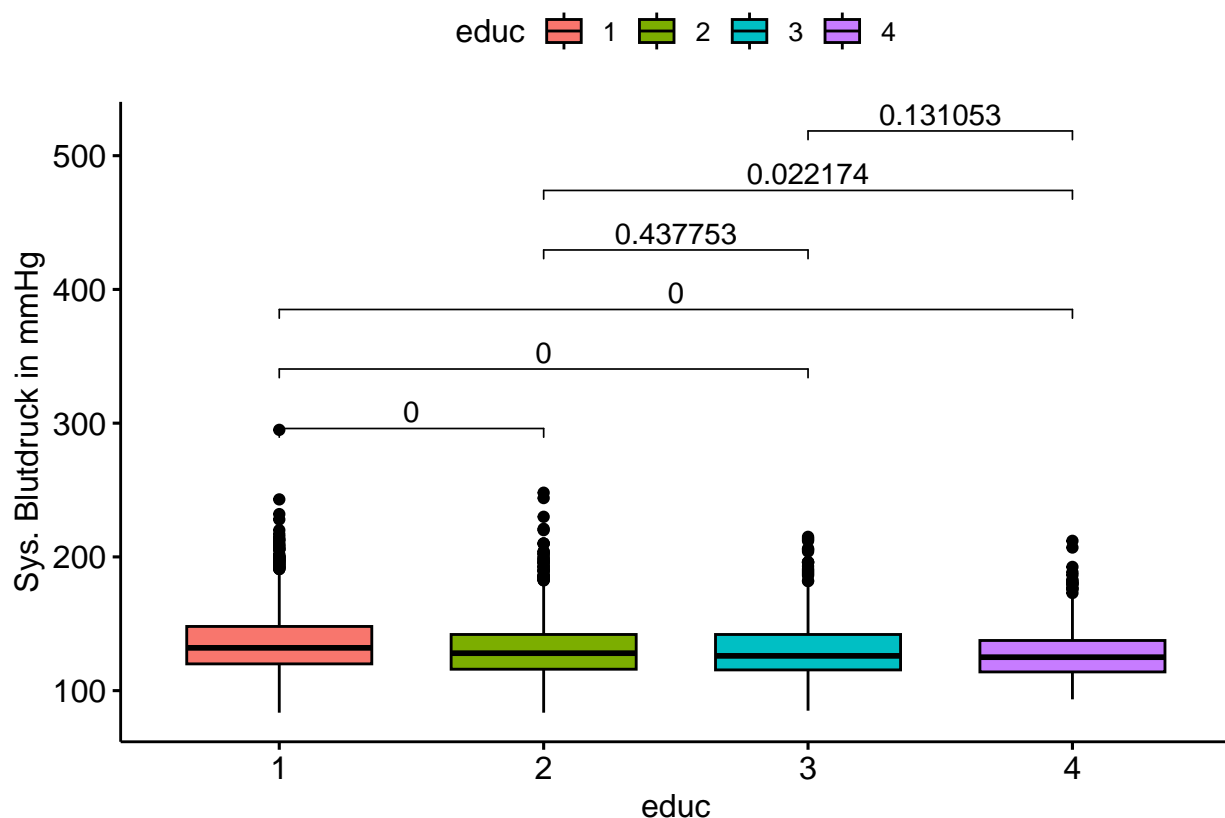
```

# Determine the maximum value of the 'sysbp' column
max_sysbp <- max(framingham$sysbp, na.rm = TRUE)

# Create an array of y positions starting from the maximum sysbp value and incrementing by a small amount
y_positions <- max_sysbp + seq(from = 1, to = 6, by = 1)

# Boxplot
ggboxplot(framingham, x = "educ", y = "sysbp",
          fill = "educ",
          palette = "Okabe-Ito",
          ylab = "Sys. Blutdruck in mmHg") +
  stat_pvalue_manual(stat.test_educ,
                    y.position = y_positions,
                    step.increase = 0.2,
                    label = "p.adj")

```



Der Kruskal Wallis Test zeigt, dass ein signifikanter Unterschied im BMI zwischen den Gruppen mit unterschiedlichem Bildungsgrad (1 bis 4) besteht (p Wert < 0.05).

Der Wilcoxon Test liefert beim Faktor "Bildungsgrad" (1 bis 4) für die Vergleichspaare (1,2), (1,3), (1,4) und (2,3) einen signifikanten p-Wert kleiner 0.05. Nimmt man an, dass die Gruppen aus Populationen mit annähernd derselben Verteilung entstammen, impliziert dies, dass der Median der Gruppe mit Bildungsgrad 1 signifikant höher als der Median der anderen Gruppen ist, und dass der Median der Gruppe mit Bildungsgrad 2 signifikant höher als der Median der Gruppen 4 ist.

```

# Perform Kruskal-Wallis test for sex
kruskal_sex <- kruskal.test(sysbp ~ sex, data = framingham)
kruskal_sex

##
## Kruskal-Wallis rank sum test
##
## data: sysbp by sex
## Kruskal-Wallis chi-squared = 1.7312, df = 1, p-value = 0.1883

# Post hoc tests

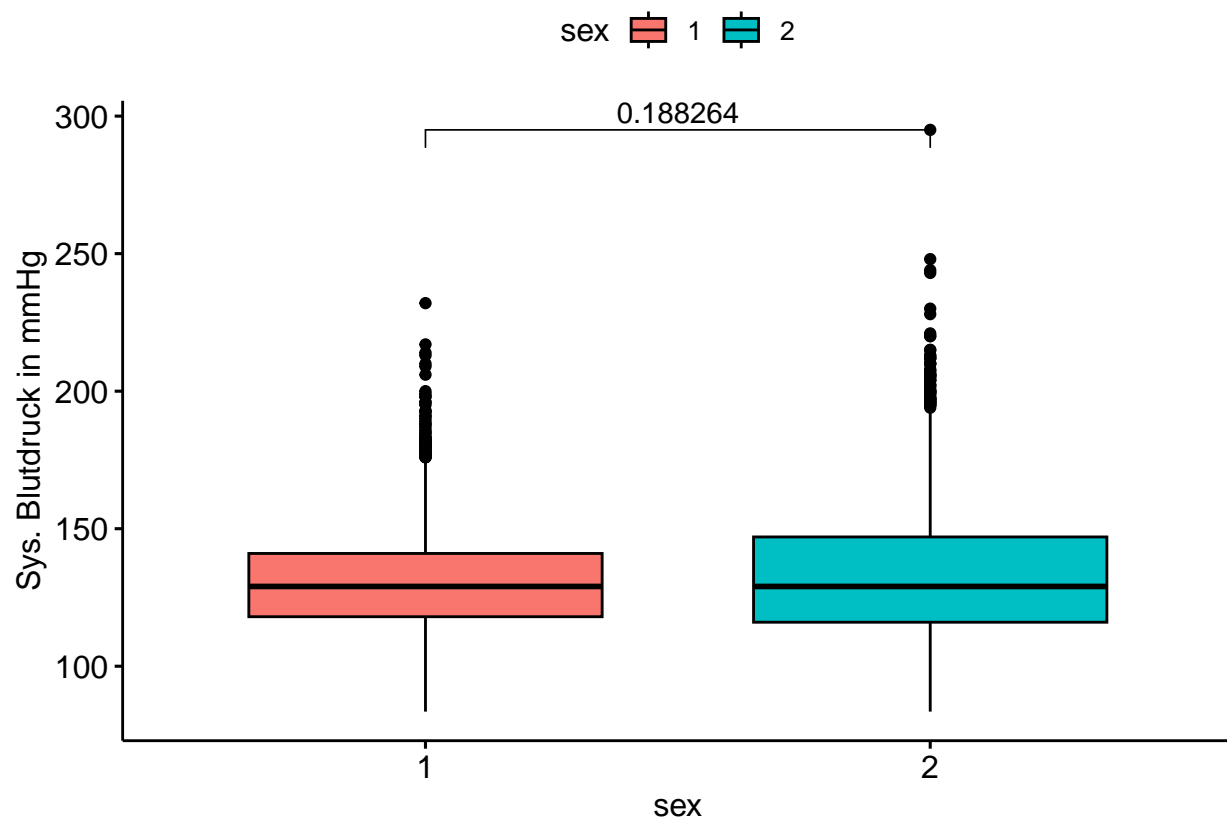
# Perform pairwise Wilcoxon tests
wilcox_sex <- pairwise.wilcox.test(framingham$sysbp, framingham$sex, p.adjust.method = "BH")

# Extract p-values
p_values_sex <- wilcox_sex$p.value

# Create data frames for manual p-value annotation
stat.test_sex <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "1",      "2", round(p_values_sex[1],6))

# Boxplot
ggboxplot(framingham, x = "sex", y = "sysbp",
  fill = "sex",
  palette = "Okabe-Ito",
  ylab = "Sys. Blutdruck in mmHg") +
  stat_pvalue_manual(stat.test_sex,
    y.position = max_sysbp,
    label = "p.adj")

```



Der Kruskal Wallis Test zeigt keinen signifikanten Unterschied im Blutdruck zwischen Geschlecht Mann und Frau (p Wert > 0.05). Der anschließende Wilcoxon Test bestätigt das Ergebnis.

```
# Perform the Scheirer-Ray-Hare test to account for interactions
result <- scheirerRayHare(sysbp ~ educ + sex, data = framingham)
```

```
##
## DV: sysbp
## Observations: 3826
## D: 0.9998357
## MS total: 1220175
```

```
# Print the results
print(result)
```

```
##           Df      Sum Sq      H  p.value
## educ       3  84603146  69.348 0.000000
## sex        1   2375068   1.947 0.162930
## educ:sex    3   15105969  12.382 0.006182
## Residuals 3818 4564582112
```

i) Achten Sie auf eine “statistisch korrekte” Formulierung des Ergebnisses.

Der Scheirer-Ray-Hare-Test liefert einen signifikanten p-Wert (<0.05) für die Wechselwirkung zwischen Bildungsgrad und Geschlecht. Dieses Ergebnis steht mit der Wechselwirkung, die im Interaktionsplot visuell

dargestellt wurde, im Einklang. Außerdem sind die p-Werte für die Haupteffekte (p-Wert für Bildungsgrad < 0.01, für Geschlecht: 0.163) des SRH Tests kompatibel mit den p-Werten der Kruskal Wallis Tests der einzelnen Faktoren (p-Wert für Bildungsgrad < 0.01, für Geschlecht = 0.188).

[Bonus]

Analysieren Sie die erhobenen Daten bezüglich der Usability der LETHE App in der Kontroll- und Interventionsgruppe. Betrachten Sie Kontroll- und Interventionsgruppe getrennt voneinander. Verwenden Sie dazu die zur Verfügung gestellten .csv Dateien, wobei die Spalte „Score“ den SUS darstellt. Untersuchen Sie, ob das jeweilige Land einen Einfluss auf die Usability der LETHE App hat. Verwenden Sie eine geeignete grafische Darstellung der Daten und statistische Verfahren. Gibt es signifikante Unterschiede zwischen den Ländern, wenn ja, zwischen welchen Ländern? Formulieren Sie die Ergebnisse „statistisch korrekt“.

```
# Read in data

read_files <- function(directory) {
  # Get a list of all CSV files in the directory
  file_list <- list.files(directory, pattern = "*.csv", full.names = TRUE)

  # Initialize an empty data frame to store the results
  result <- data.frame()

  # Loop over the files
  for (file in file_list) {
    # Read the file into a data frame
    data <- read.csv(file)

    # Extract the group and country from the file name
    file_info <- strsplit(basename(file), "_")[[1]]
    group <- file_info[1]
    country <- file_info[3]

    # Keep only the 'Score' column and add the 'Group' and 'Country' columns
    data <- data.frame(Score = data$Score, Group = group, Country = country)

    # Append the data frame to the result
    result <- rbind(result, data)
  }

  return(result)
}

# Use the function to read the files from a directory
data_sus <- read_files("ue3_Bonus_LETHE_SUS/LETHE_SUS_data")

# Convert 'Group' and 'Country' to factors
data_sus$Group <- factor(data_sus$Group)
data_sus$Country <- factor(data_sus$Country)

# Print the updated data frame
data_sus
```

```
##      Score      Group Country
```


## 1	62.5	Control Austria
## 2	100.0	Control Austria
## 3	85.0	Control Austria
## 4	97.5	Control Austria
## 5	92.5	Control Austria
## 6	55.0	Control Austria
## 7	80.0	Control Austria
## 8	100.0	Control Austria
## 9	95.0	Control Austria
## 10	100.0	Control Austria
## 11	85.0	Control Austria
## 12	90.0	Control Austria
## 13	95.0	Control Austria
## 14	92.5	Control Austria
## 15	57.5	Control Austria
## 16	72.5	Control Austria
## 17	72.5	Control Austria
## 18	92.5	Control Finland
## 19	60.0	Control Finland
## 20	52.5	Control Finland
## 21	67.5	Control Finland
## 22	70.0	Control Finland
## 23	72.5	Control Finland
## 24	97.5	Control Finland
## 25	97.5	Control Finland
## 26	62.5	Control Finland
## 27	72.5	Control Finland
## 28	77.5	Control Finland
## 29	65.0	Control Finland
## 30	65.0	Control Finland
## 31	67.5	Control Finland
## 32	65.0	Control Finland
## 33	75.0	Control Finland
## 34	55.0	Control Italy
## 35	32.5	Control Italy
## 36	47.5	Control Italy
## 37	70.0	Control Italy
## 38	40.0	Control Italy
## 39	92.5	Control Italy
## 40	97.5	Control Italy
## 41	90.0	Control Italy
## 42	62.5	Control Italy
## 43	90.0	Control Italy
## 44	85.0	Control Italy
## 45	90.0	Control Italy
## 46	65.0	Control Sweden
## 47	87.5	Control Sweden
## 48	62.5	Control Sweden
## 49	95.0	Control Sweden
## 50	50.0	Control Sweden
## 51	50.0	Control Sweden
## 52	70.0	Control Sweden
## 53	47.5	Control Sweden
## 54	45.0	Control Sweden

## 55	37.5	Control	Sweden
## 56	85.0	Control	Sweden
## 57	82.5	Control	Sweden
## 58	57.5	Intervention	Austria
## 59	97.5	Intervention	Austria
## 60	85.0	Intervention	Austria
## 61	65.0	Intervention	Austria
## 62	65.0	Intervention	Austria
## 63	95.0	Intervention	Austria
## 64	0.0	Intervention	Austria
## 65	52.5	Intervention	Austria
## 66	90.0	Intervention	Austria
## 67	55.0	Intervention	Austria
## 68	97.5	Intervention	Austria
## 69	52.5	Intervention	Austria
## 70	97.5	Intervention	Austria
## 71	75.0	Intervention	Austria
## 72	60.0	Intervention	Austria
## 73	97.5	Intervention	Austria
## 74	72.5	Intervention	Austria
## 75	50.0	Intervention	Finland
## 76	70.0	Intervention	Finland
## 77	95.0	Intervention	Finland
## 78	57.5	Intervention	Finland
## 79	55.0	Intervention	Finland
## 80	77.5	Intervention	Finland
## 81	42.5	Intervention	Finland
## 82	87.5	Intervention	Finland
## 83	82.5	Intervention	Finland
## 84	67.5	Intervention	Finland
## 85	62.5	Intervention	Finland
## 86	80.0	Intervention	Finland
## 87	55.0	Intervention	Finland
## 88	85.0	Intervention	Finland
## 89	70.0	Intervention	Finland
## 90	75.0	Intervention	Finland
## 91	40.0	Intervention	Finland
## 92	87.5	Intervention	Finland
## 93	75.0	Intervention	Finland
## 94	62.5	Intervention	Italy
## 95	70.0	Intervention	Italy
## 96	65.0	Intervention	Italy
## 97	82.5	Intervention	Italy
## 98	67.5	Intervention	Italy
## 99	77.5	Intervention	Italy
## 100	50.0	Intervention	Italy
## 101	75.0	Intervention	Italy
## 102	37.5	Intervention	Italy
## 103	77.5	Intervention	Italy
## 104	75.0	Intervention	Italy
## 105	47.5	Intervention	Italy
## 106	100.0	Intervention	Italy
## 107	85.0	Intervention	Italy
## 108	52.5	Intervention	Italy

```
## 109 80.0 Intervention Italy
## 110 62.5 Intervention Italy
## 111 67.5 Intervention Italy
## 112 50.0 Intervention Sweden
## 113 100.0 Intervention Sweden
## 114 80.0 Intervention Sweden
## 115 42.5 Intervention Sweden
## 116 72.5 Intervention Sweden
## 117 80.0 Intervention Sweden
## 118 70.0 Intervention Sweden
## 119 100.0 Intervention Sweden
## 120 57.5 Intervention Sweden
## 121 40.0 Intervention Sweden
## 122 45.0 Intervention Sweden
## 123 50.0 Intervention Sweden
```

Alle .csv Dateien werden eingelesen. Das Dataframe enthält 3 Spalten “Score”, “Group” und “Country”. Die Spalten “Group” und “Country” werden in Faktoren umgewandelt.

```
# Select subset of data by group type

# Subset the data for the Intervention group
intervention_data <- subset(data_sus, Group == "Intervention")

# Subset the data for the Control group
control_data <- subset(data_sus, Group == "Control")
```

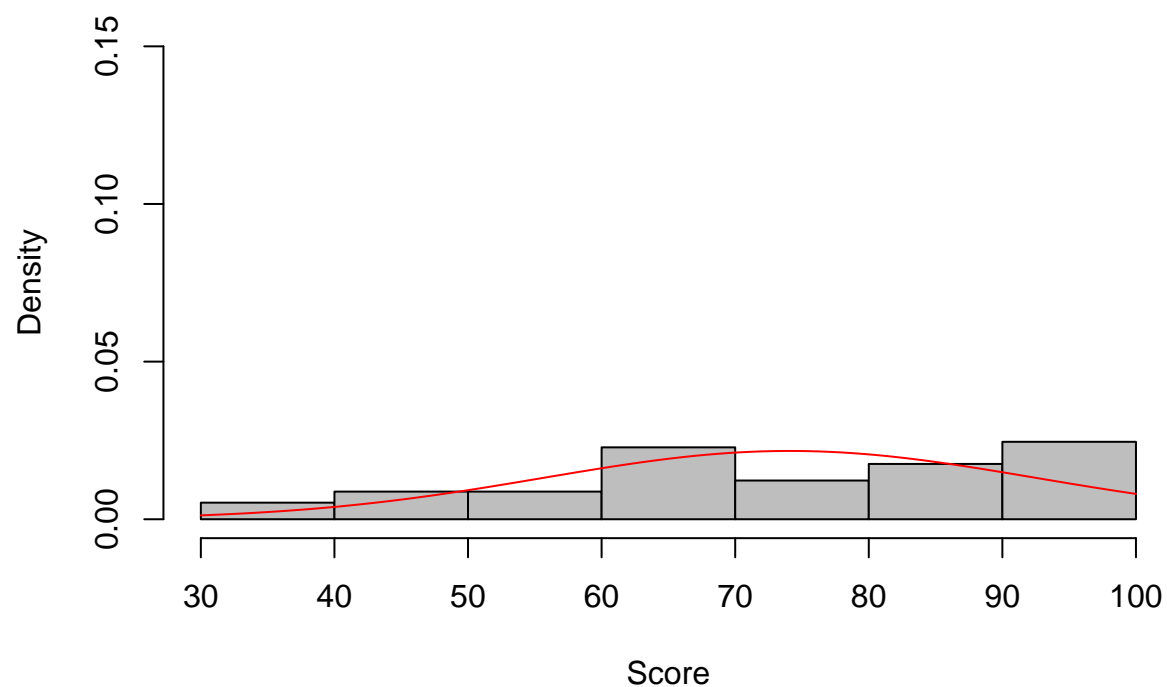
Für die getrennte Analyse von Kontroll- und Interventionsgruppe werden die Daten entsprechend unterteilt.

```
# Analyze Control Group

# -- For Whole Control Group

test_normality_of_groups(control_data, "Group", "Score")
```

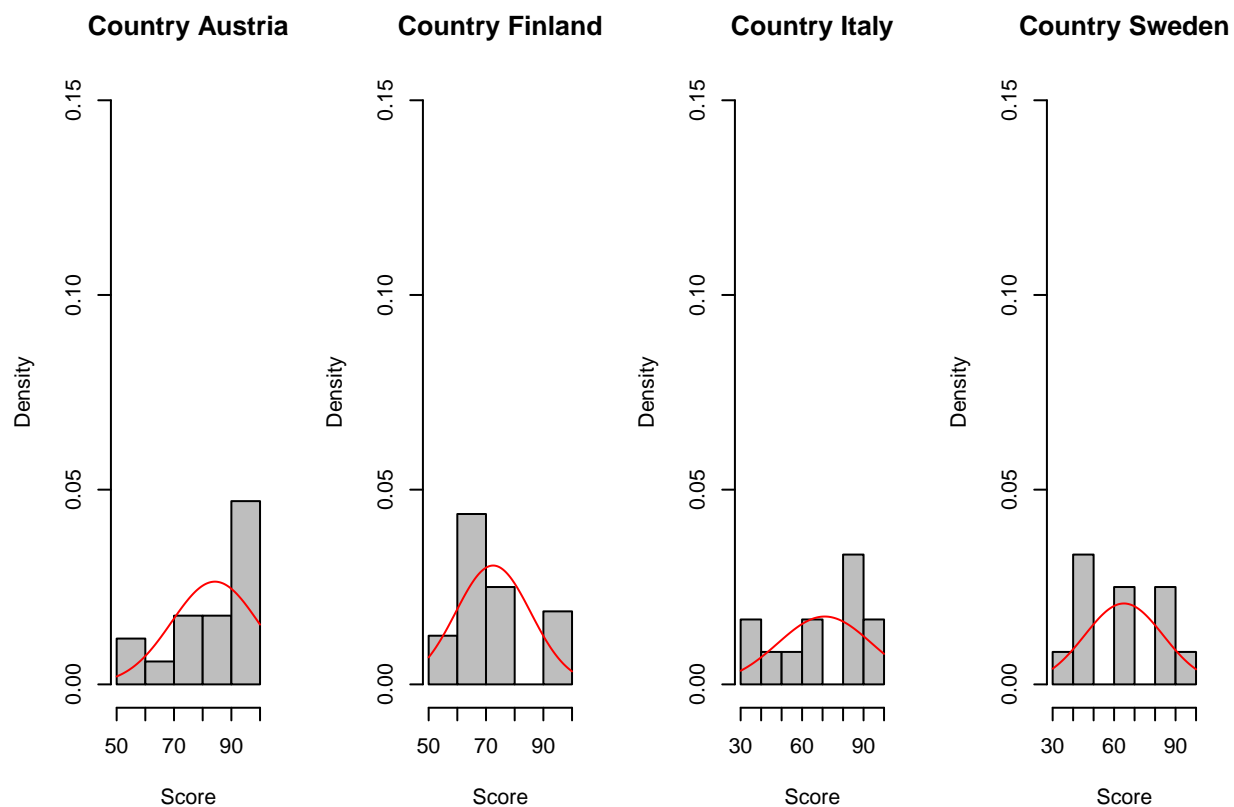
Group Control



```
##  
## Shapiro-Wilk normality test  
##  
## data: data$Score[data$Group==Control]  
## W = 0.94784, p-value = 0.01572
```

```
# -- For Subgroup Countries in Control Group
```

```
test_normality_of_groups(control_data, "Country", "Score")
```



```
##
##  Shapiro-Wilk normality test
##
## data:  data$Score[data$Country==Austria]
## W = 0.87728, p-value = 0.02874
##
##
##  Shapiro-Wilk normality test
##
## data:  data$Score[data$Country==Finland]
## W = 0.88503, p-value = 0.04652
##
##
##  Shapiro-Wilk normality test
##
## data:  data$Score[data$Country==Italy]
## W = 0.88798, p-value = 0.111
##
##
##  Shapiro-Wilk normality test
##
## data:  data$Score[data$Country==Sweden]
## W = 0.9324, p-value = 0.4062
```

```
# Variance inhomogeneity (Heteroscedasticity)

# Levene test
leveneTest(Score ~ Country, data = control_data)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group 3  2.6629 0.05731 .
##      53
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Bartlett test
bartlett.test(Score ~ Country, data = control_data)
```

```
##
## Bartlett test of homogeneity of variances
##
## data:  Score by Country
## Bartlett's K-squared = 4.7073, df = 3, p-value = 0.1945
```

Für die gesamte Kontrollgruppe gilt, dass die Normalverteilung der Scores nicht gegeben ist. Dies wird durch den Shapiro-Wilk-Test bestätigt ($p < 0.05$). Ebenso liegt für die Subgruppen Österreich und Finnland keine Normalverteilung vor ($p < 0.05$). Die Varianz der Scores über die Ländergruppen ist homogen, wie der Levene- und Bartlett-Test zeigen ($p > 0.05$). Da die Normalverteilung der Scores aller Ländergruppen nicht gegeben, ist eine Voraussetzung für die ANOVA verletzt.

```
table(control_data$Country)
```

```
##
## Austria Finland    Italy  Sweden
##      17      16      12      12
```

Zu wenige Daten bedeuten, dass eine ANOVA endgültig unzulässig ist. Es wird ein Kruskal Wallis Test angewandt.

```
# Perform Kruskal-Wallis test for educ
kruskal_country <- kruskal.test(Score ~ Country, data = control_data)
kruskal_country
```

```
##
## Kruskal-Wallis rank sum test
##
## data:  Score by Country
## Kruskal-Wallis chi-squared = 9.0494, df = 3, p-value = 0.02864
```

```
# Post hoc tests
```

```
# Perform pairwise Wilcoxon tests
wilcox_country <- pairwise.wilcox.test(control_data$Score, control_data$Country, p.adjust.method = "BH")
```

```
## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen
```

```
# Extract p-values
p_values_country <- wilcox_country$p.value

print(p_values_country)
```

```
##           Austria  Finland      Italy
## Finland 0.10863903         NA         NA
## Italy   0.15854102 0.8889787         NA
## Sweden  0.04659353 0.3135669 0.4818974
```

```
# Convert p-value matrix to a vector
p_values_country_vec <- as.vector(p_values_country)

# Remove NA values (these are for comparisons with themselves)
p_values_country <- p_values_country_vec[!is.na(p_values_country_vec)]

# Create data frames for manual p-value annotation
stat.test_country <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "Austria", "Finland", round(p_values_country[1],6),
  "Austria", "Italy", round(p_values_country[2],6),
  "Austria", "Sweden", round(p_values_country[3],6),
  "Finland", "Italy", round(p_values_country[4],6),
  "Finland", "Sweden", round(p_values_country[5],6),
  "Italy", "Sweden", round(p_values_country[6],6))

p_values_country[8]
```

```
## [1] NA
```

```
stat.test_country
```

```
## # A tibble: 6 x 3
##   group1 group2 p.adj
```

```
##   <chr>   <chr>   <dbl>
## 1 Austria Finland 0.109
## 2 Austria Italy   0.159
## 3 Austria Sweden 0.0466
## 4 Finland Italy   0.889
## 5 Finland Sweden 0.314
## 6 Italy   Sweden 0.482
```

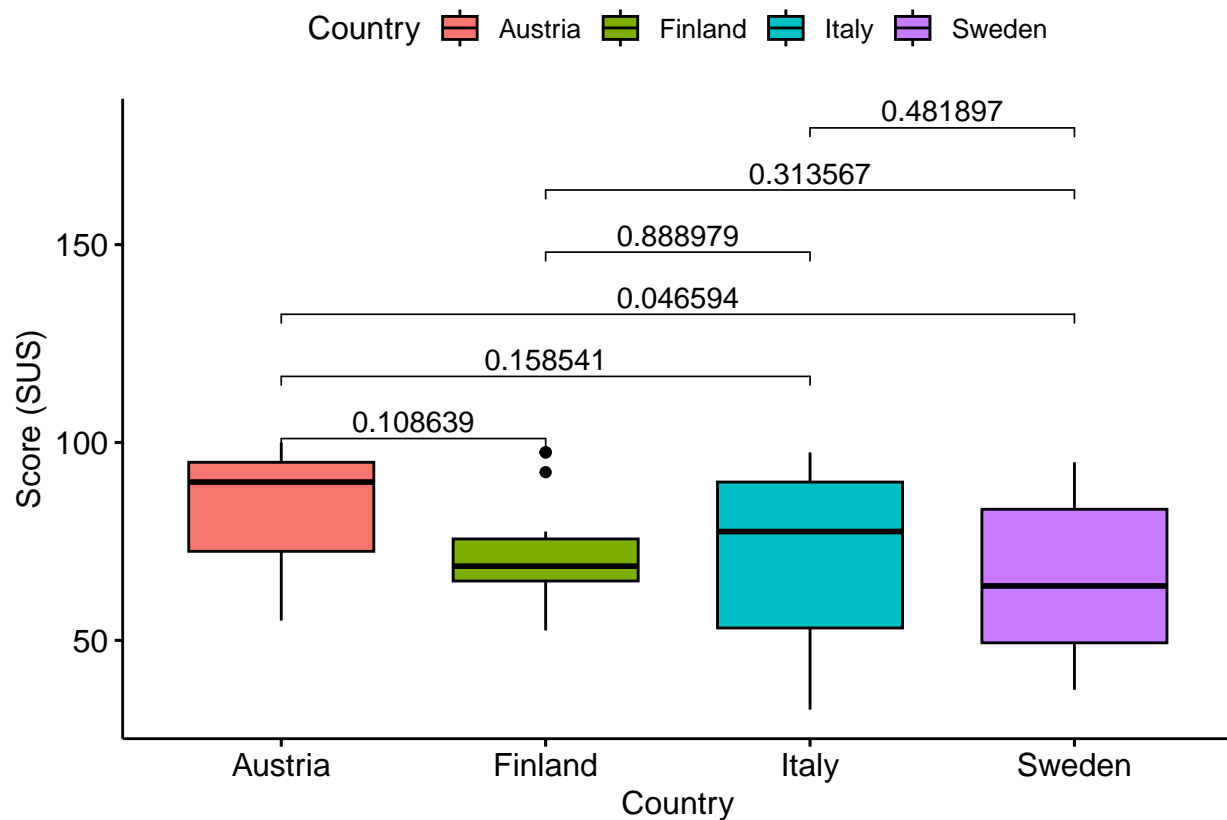
```
# Determine the maximum value of the 'sysbp' column
max_score <- max(control_data$Score, na.rm = TRUE)
```

```
# Create an array of y positions starting from the maximum sysbp value and incrementing by a small amount
y_positions <- max_score + seq(from = 1, to = 6, by = 1)
```

```
y_positions
```

```
## [1] 101 102 103 104 105 106
```

```
# Boxplot
ggboxplot(control_data, x = "Country", y = "Score",
  fill = "Country",
  palette = "Okabe-Ito",
  ylab = "Score (SUS)") +
  stat_pvalue_manual(stat.test_country,
    y.position = y_positions,
    step.increase = 0.2,
    label = "p.adj")
```

Der Kruskal Wallis Test zeigt, dass es einen signifikanten Unterschied in der Usability der LETHE App zwischen den Ländern bei der Kontrollgruppe gibt ($p < 0.05$).

Die Post hoc Analyse besagt, dass es innerhalb der Kontrollgruppe zwischen den Länder-Subgruppen keine signifikanten Unterschiede gibt außer zwischen Österreich und Schweden ($p < 0.05$). Dies bedeutet, dass sich die Verteilung des SUS Scores zwischen den Ländern bei der Kontrollgruppe nicht signifikant unterscheidet außer zwischen Österreich und Schweden.

```
## Get the length of the p_values_country vector
n <- length(p_values_country)

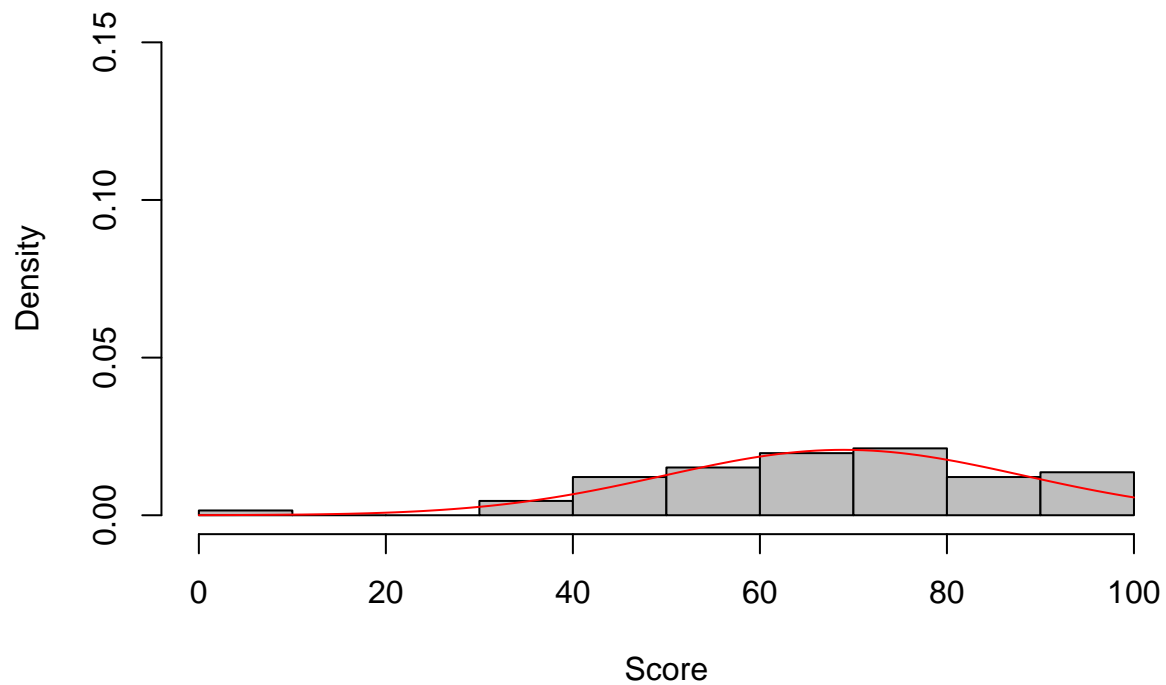
## Loop over the indices from 1 to n
for (i in 1:n) {
  # # Print the index and the corresponding p-value
  # print(paste("Index:", i, "P-value:", round(p_values_country[i], 6)))
# }
```

```
# Analyze Intervention Group

# -- For Whole Intervention Group

test_normality_of_groups(intervention_data, "Group", "Score")
```

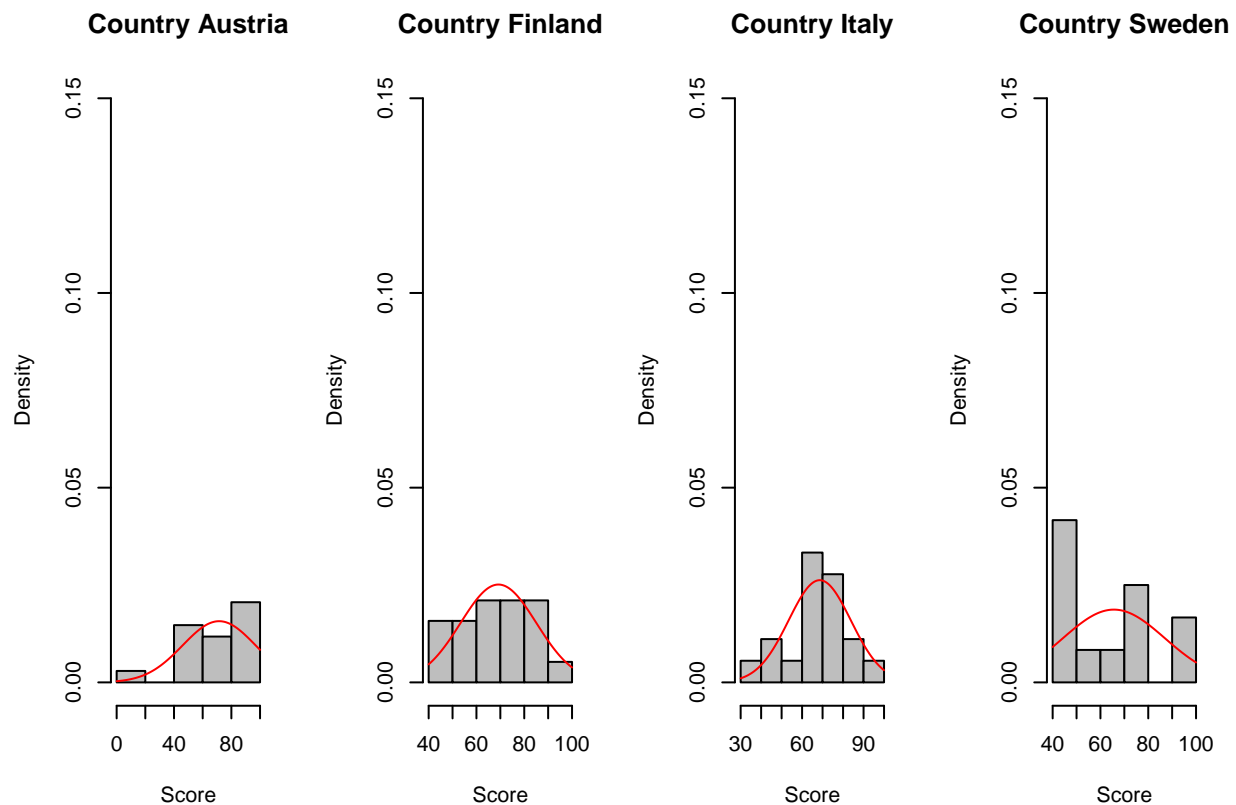
Group Intervention



```
##  
## Shapiro-Wilk normality test  
##  
## data: data$Score[data$Group==Intervention]  
## W = 0.95966, p-value = 0.0309
```

```
# -- For Subgroup Countries in Intervention Group
```

```
test_normality_of_groups(intervention_data,"Country","Score")
```



```
##
##  Shapiro-Wilk normality test
##
## data:  data$Score[data$Country==Austria]
## W = 0.85477, p-value = 0.01269
##
##
##  Shapiro-Wilk normality test
##
## data:  data$Score[data$Country==Finland]
## W = 0.96415, p-value = 0.6563
##
##
##  Shapiro-Wilk normality test
##
## data:  data$Score[data$Country==Italy]
## W = 0.9795, p-value = 0.9447
##
##
##  Shapiro-Wilk normality test
##
## data:  data$Score[data$Country==Sweden]
## W = 0.90608, p-value = 0.19
```

```
# Variance inhomogeneity (Heteroscedasticity)
```

```
# Levene test
```

```
leveneTest(Score ~ Country, data = intervention_data)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
```

```
##      Df F value Pr(>F)
```

```
## group 3  1.8306 0.1508
```

```
##      62
```

```
# Bartlett test
```

```
bartlett.test(Score ~ Country, data = intervention_data)
```

```
##
```

```
## Bartlett test of homogeneity of variances
```

```
##
```

```
## data: Score by Country
```

```
## Bartlett's K-squared = 5.8348, df = 3, p-value = 0.1199
```

Für die gesamte Interventionsgruppe gilt, dass die Normalverteilung der Scores nicht gegeben ist. Dies wird durch den Shapiro-Wilk-Test bestätigt ($p < 0.05$). Ebenso liegt für die Subgruppe Österreich keine Normalverteilung vor ($p < 0.05$). Die Varianz der Scores über die Ländergruppen ist homogen, wie der Levene- und Bartlett-Test zeigen ($p > 0.05$). Da die Normalverteilung der Scores aller Ländergruppen nicht gegeben ist, ist eine Voraussetzung für die ANOVA verletzt.

```
table(intervention_data$Country)
```

```
##
```

```
## Austria Finland Italy Sweden
```

```
##      17      19      18      12
```

Zu wenige Daten bedeuten, dass eine ANOVA endgültig unzulässig ist. Es wird ein Kruskal Wallis Test angewandt.

```
# Perform Kruskal-Wallis test for educ
```

```
kruskal_country <- kruskal.test(Score ~ Country, data = intervention_data)
```

```
kruskal_country
```

```
##
```

```
## Kruskal-Wallis rank sum test
```

```
##
```

```
## data: Score by Country
```

```
## Kruskal-Wallis chi-squared = 1.2512, df = 3, p-value = 0.7408
```

```
# Post hoc tests
```

```
# Perform pairwise Wilcoxon tests
```

```
wilcox_country <- pairwise.wilcox.test(intervention_data$Score, intervention_data$Country, p.adjust.meth
```

```
## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen
```

```
# Extract p-values
p_values_country <- wilcox_country$p.value

print(p_values_country)
```

```
##           Austria  Finland      Italy
## Finland 0.7329331      NA         NA
## Italy   0.7329331 0.7724098      NA
## Sweden  0.7329331 0.7329331 0.7329331
```

```
# Convert p-value matrix to a vector
p_values_country_vec <- as.vector(p_values_country)

# Remove NA values (these are for comparisons with themselves)
p_values_country <- p_values_country_vec[!is.na(p_values_country_vec)]

# Create data frames for manual p-value annotation
stat.test_country <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "Austria", "Finland", round(p_values_country[1],6),
  "Austria", "Italy", round(p_values_country[2],6),
  "Austria", "Sweden", round(p_values_country[3],6),
  "Finland", "Italy", round(p_values_country[4],6),
  "Finland", "Sweden", round(p_values_country[5],6),
  "Italy", "Sweden", round(p_values_country[6],6))

p_values_country[8]
```

```
## [1] NA
```

```
stat.test_country
```

```
## # A tibble: 6 x 3
##   group1 group2 p.adj
```

```
##   <chr>   <chr>   <dbl>
## 1 Austria Finland 0.733
## 2 Austria Italy   0.733
## 3 Austria Sweden 0.733
## 4 Finland Italy   0.772
## 5 Finland Sweden 0.733
## 6 Italy   Sweden 0.733
```

```
# Determine the maximum value of the 'sysbp' column
```

```
max_score <- max(intervention_data$Score, na.rm = TRUE)
```

```
# Create an array of y positions starting from the maximum sysbp value and incrementing by a small amount
```

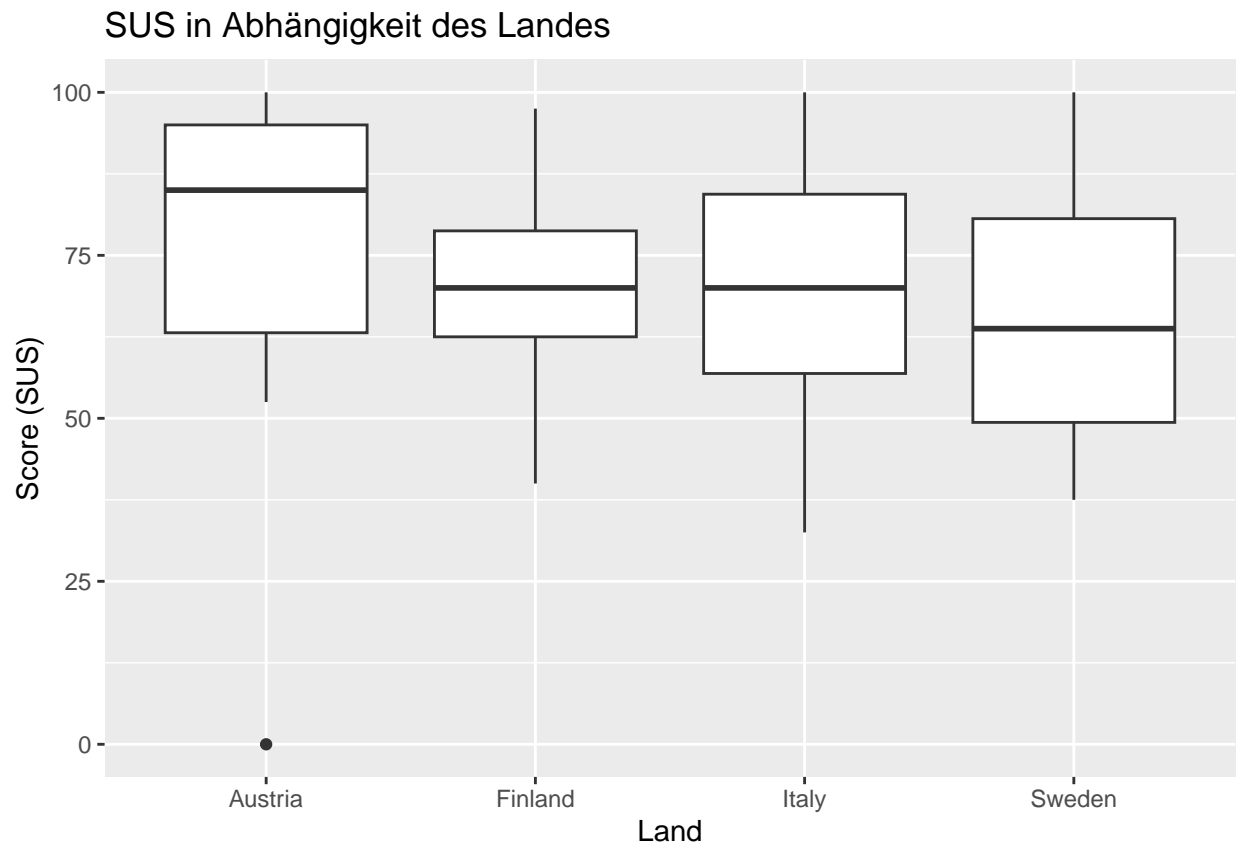
```
y_positions <- max_score + seq(from = 1, to = 6, by = 1)
```

```
y_positions
```

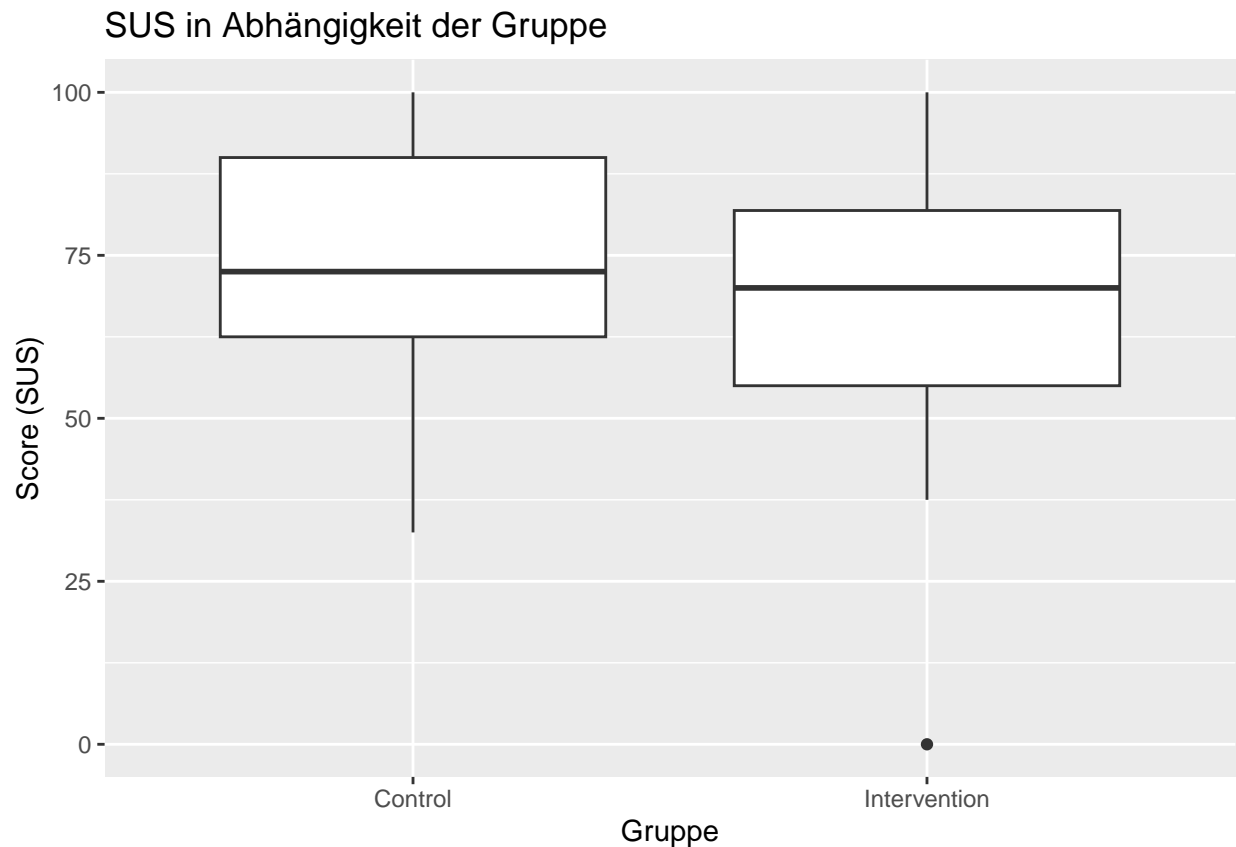
```
## [1] 101 102 103 104 105 106
```

```
# Boxplot
```

```
ggboxplot(intervention_data, x = "Country", y = "Score",
  fill = "Country",
  palette = "Okabe-Ito",
  ylab = "Score (SUS)") +
  stat_pvalue_manual(stat.test_country,
    y.position = y_positions,
    step.increase = 0.2,
    label = "p.adj")
```

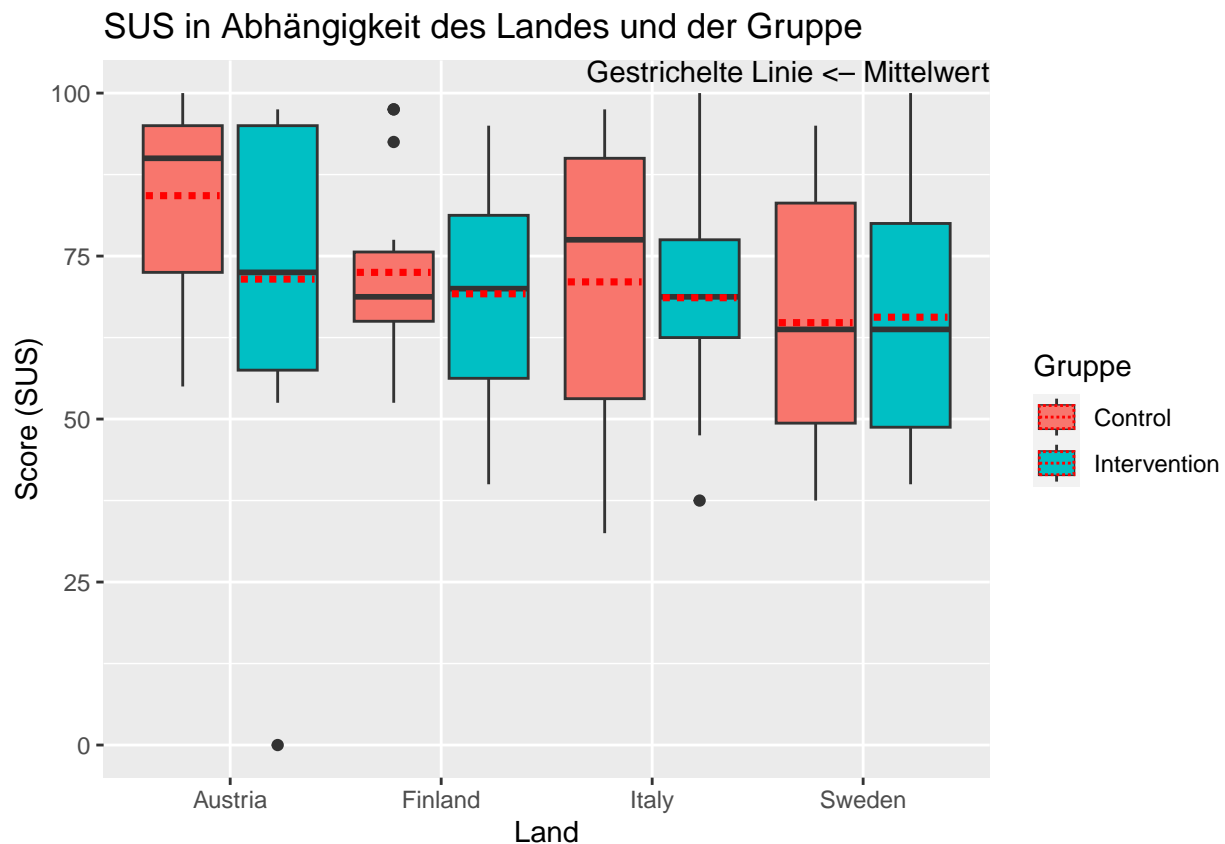
```
# Boxplot group levels
ggplot(data_sus, aes(x = Group, y = Score)) +
  geom_boxplot() +
  labs(title = "SUS in Abhängigkeit der Gruppe",
       x = "Gruppe",
       y = "Score (SUS)")
```

```
# Boxplot country and group levels with mean lines
ggplot(data_sus, aes(x = Country, y = Score, fill = Group)) +
  geom_boxplot(position = position_dodge(0.9)) +
  stat_summary(
    fun = mean,
    geom = "crossbar",
    width = 0.7,
    position = position_dodge(0.9),
    color = "red",
    size = 0.5,
    linetype = "11"
  ) +
  annotate(
    "text",
    x = Inf,
    y = Inf,
    label = "Gestrichelte Linie <- Mittelwert",
    hjust = 1,
    vjust = 1
  ) +
  labs(
    title = "SUS in Abhängigkeit des Landes und der Gruppe",
    x = "Land",
    y = "Score (SUS)",
    fill = "Gruppe"
  ) +
```

```
ylim(min(data_sus$Score), max(data_sus$Score))
```

```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```



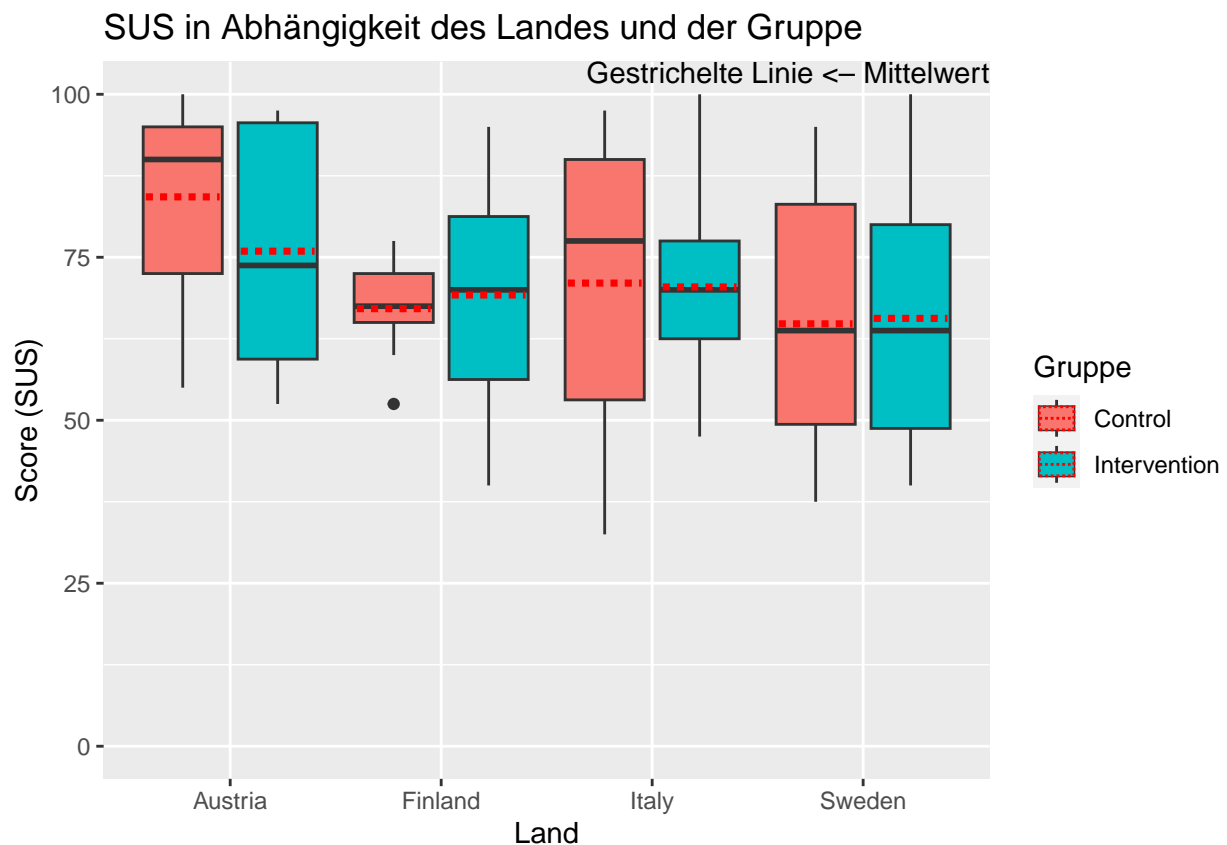
```
# Remove outliers (median +/- 1.5 IQR)
library(dplyr)
data_sus_clean <- data_sus %>%
  group_by(Country, Group) %>%
  mutate(lb = quantile(Score, 0.25) - 1.5 * IQR(Score),
         ub = quantile(Score, 0.75) + 1.5 * IQR(Score)) %>%
  filter(Score >= lb & Score <= ub)

# Boxplot country and group levels with mean lines
ggplot(data_sus_clean, aes(x = Country, y = Score, fill = Group)) +
  geom_boxplot(position = position_dodge(0.9)) +
  stat_summary(
    fun = mean,
    geom = "crossbar",
    width = 0.7,
```

```

position = position_dodge(0.9),
color = "red",
size = 0.5,
linetype = "11"
) +
annotate(
  "text",
  x = Inf,
  y = Inf,
  label = "Gestrichelte Linie <- Mittelwert",
  hjust = 1,
  vjust = 1
) +
labs(
  title = "SUS in Abhängigkeit des Landes und der Gruppe",
  x = "Land",
  y = "Score (SUS)",
  fill = "Gruppe"
) +
ylim(min(data_sus$Score), max(data_sus$Score))

```



Die von Ausreißern bereinigten Boxplots zeigen variierende Verhältnisse zwischen Kontroll- und Interventionsgruppe in den Ländern. Für Österreich und Italien fällt der Median der Kontrollgruppe höher aus als der der Interventionsgruppe. Für Finnland fällt der Median der Interventionsgruppe höher aus als der der Kontrollgruppe. Für Schweden gibt es nur einen minimalen Unterschied zwischen den Medianen der Kontroll- und Interventionsgruppe.

```

# load_source()

# # -- For group levels

# test_normality_of_groups(data_sus,"Group", "Score")

# # -- For country levels

# test_normality_of_groups(data_sus,"Country","Score")

# # Variance inhomogeneity (Heteroscedasticity)

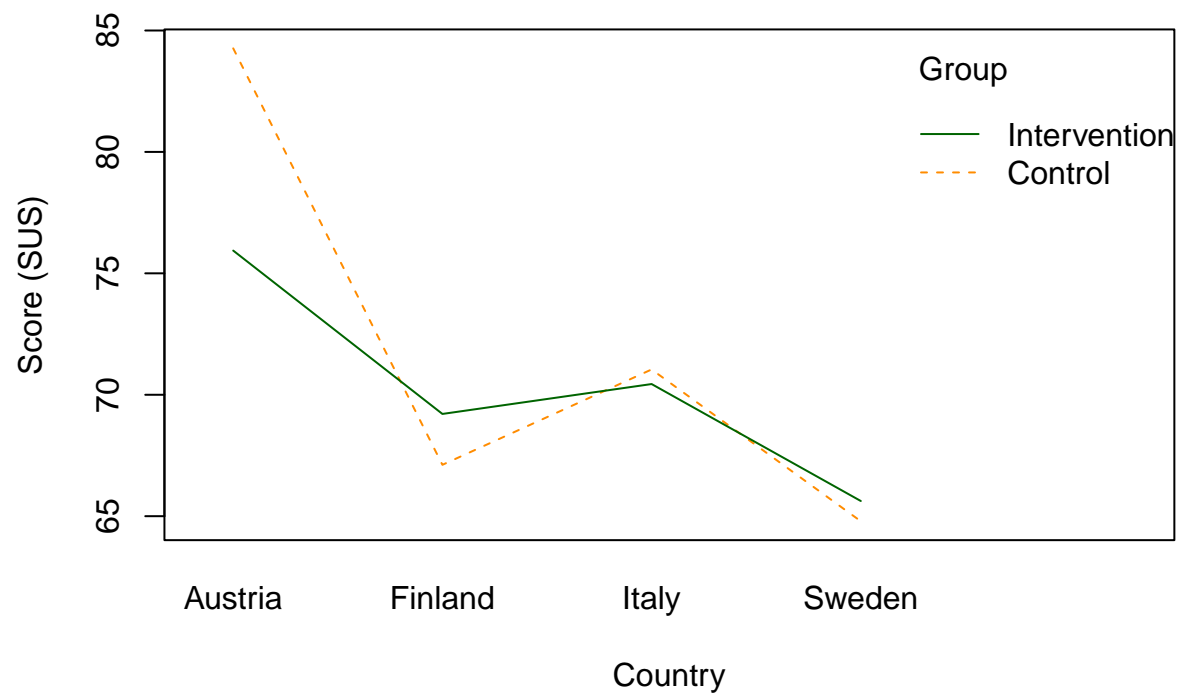
# # Levene test
# leveneTest(Score ~ Group, data = data_sus)
# leveneTest(Score ~ Country, data = data_sus)
# leveneTest(Score ~ Group * Country, data = data_sus)

# # Bartlett test
# bartlett.test(Score ~ Group, data = data_sus)
# bartlett.test(Score ~ Country, data = data_sus)

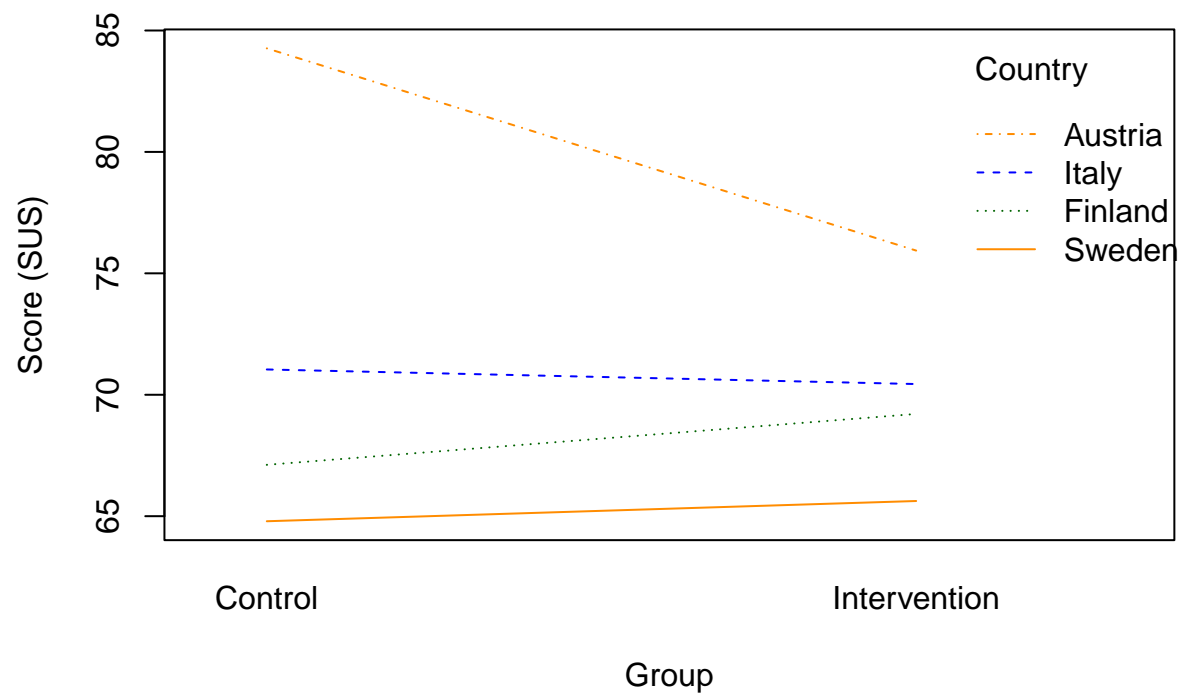
data_sus <- data_sus_clean

# Interaction plots (simple)
ip_educ <- interaction.plot(data_sus$Country, data_sus$Group, data_sus$Score,
  col = c("darkorange", "darkgreen", "blue"),
  pch = c(1, 2, 3), fixed = FALSE, xlab = "Country",
  ylab = "Score (SUS)", trace.label = "Group")

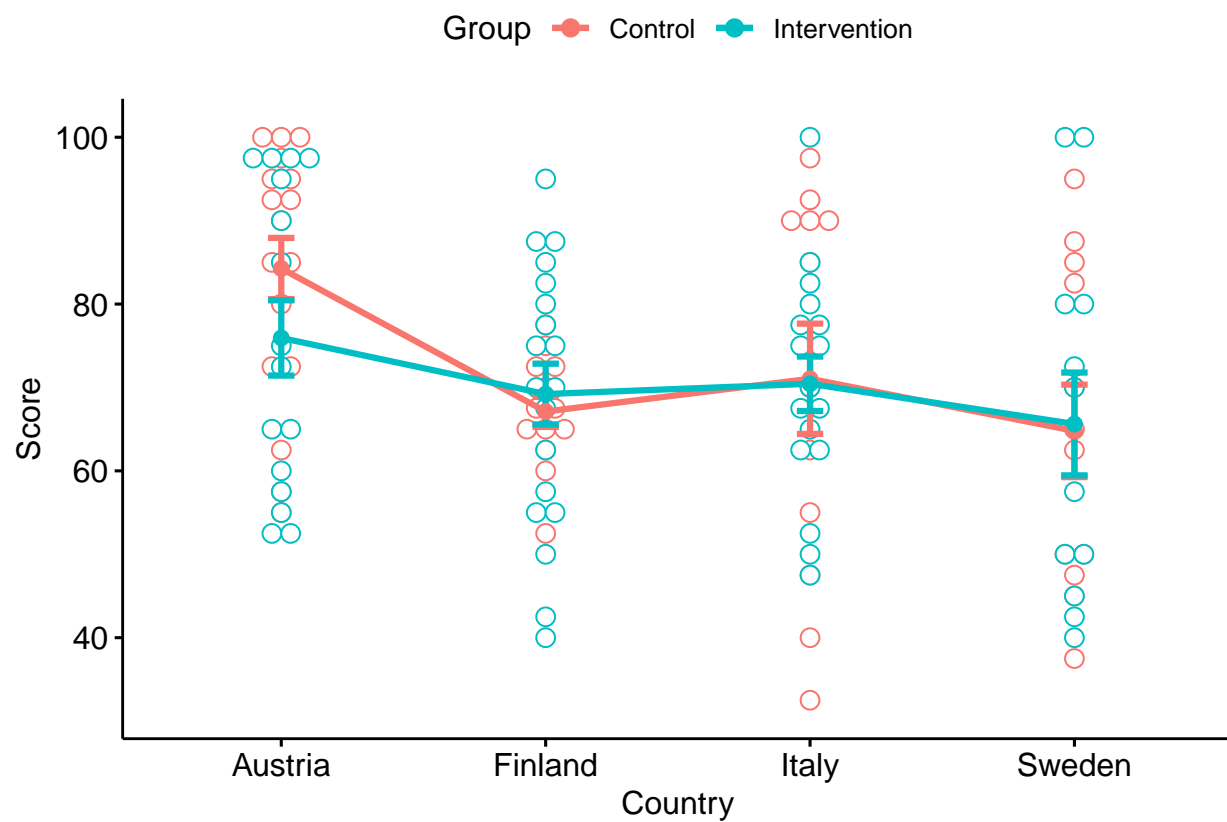
```



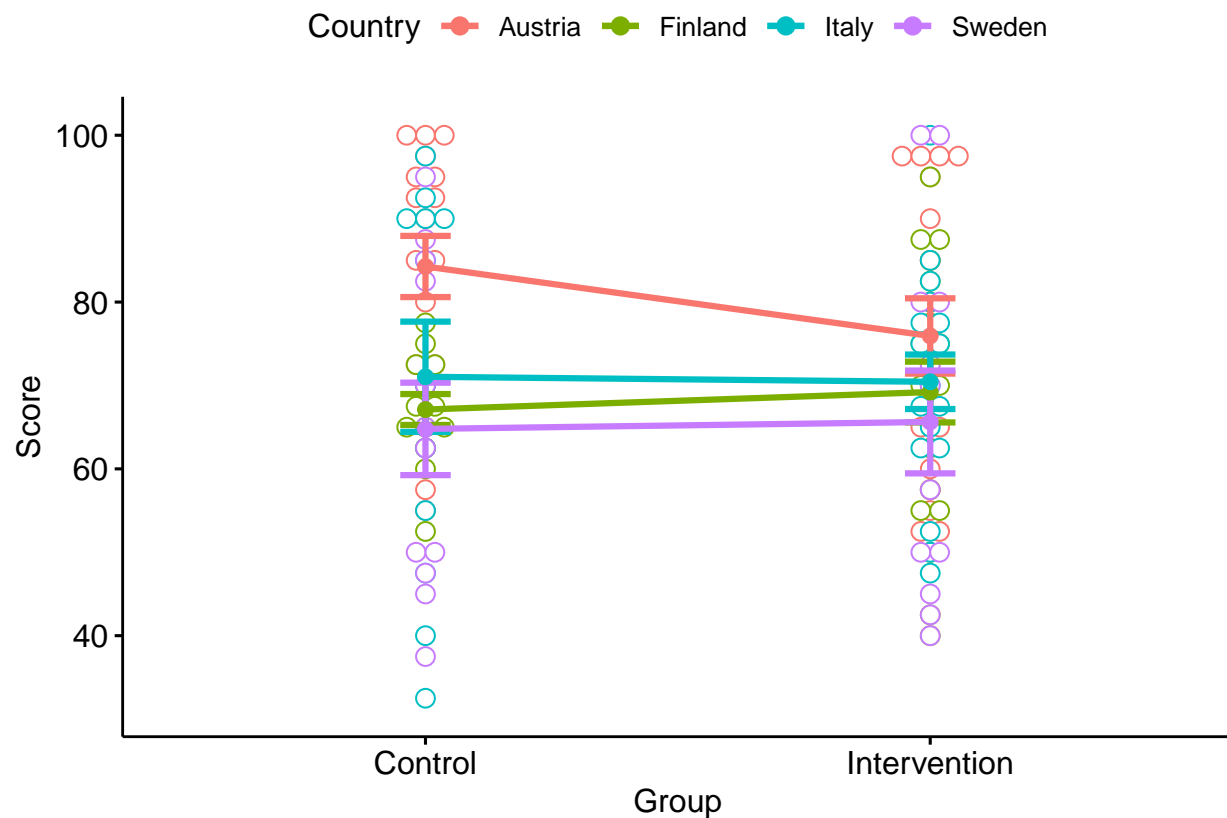
```
ip_sex <- interaction.plot(data_sus$Group, data_sus$Country, data_sus$Score,  
  col = c("darkorange", "darkgreen", "blue"),  
  pch = c(1, 2, 3), fixed = FALSE, xlab = "Group",  
  ylab = "Score (SUS)", trace.label = "Country")
```



```
# Interaction plots (ggpubr)
ggline(data_sus, x = "Country", y = "Score", color = "Group", size = 1.1,
       add = c("mean_se", "dotplot"))
```



```
ggline(data_sus, x = "Group", y = "Score", color = "Country", size = 1.1,
       add = c("mean_se", "dotplot"))
```



Die Interaktionsplots zeigen für die Ausreißer bereinigten Daten, dass die Wechselwirkung semidisordinal ist. Beim Ländervergleich auf der x-Achse überschneiden sich die Verläufe für Kontroll- und Interventionsgruppe. Beim Gruppenvergleich auf der x-Achse gibt es keine Überschneidungen, aber negative wie positive Steigungen der Verdünnungslinien.

Ab hier ignorieren

Vergleich: Für Gesamt Kontroll vs Intervention

```
# Perform Kruskal-Wallis test for educ
kruskal_group <- kruskal.test(Score ~ Group, data = data_sus)
kruskal_group

##
## Kruskal-Wallis rank sum test
##
## data: Score by Group
## Kruskal-Wallis chi-squared = 0.52107, df = 1, p-value = 0.4704

# Post hoc tests
```



```

# Perform pairwise Wilcoxon tests
wilcox_group <- pairwise.wilcox.test(data_sus$Score, data_sus$Group, p.adjust.method = "BH")

# Extract p-values
p_values_group <- wilcox_group$p.value

print(p_values_group)

```

```

##           Control
## Intervention 0.4720479

```

```

# Convert p-value matrix to a vector
p_values_group_vec <- as.vector(p_values_group)

# Remove NA values (these are for comparisons with themselves)
p_values_group <- p_values_group_vec[!is.na(p_values_group_vec)]

# Create data frames for manual p-value annotation
stat.test_group <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "Control", "Intervention", round(p_values_group[1],6)
)

stat.test_group

```

```

## # A tibble: 1 x 3
##   group1 group2      p.adj
##   <chr>  <chr>    <dbl>
## 1 Control Intervention 0.472

```

```

# Determine the maximum value of the 'Score' column
max_score <- max(data_sus$Score, na.rm = TRUE)

# Create an array of y positions starting from the maximum Score value and incrementing by a small amount
y_positions <- max_score + seq(from = 1, to = 1, by = 1)

y_positions

```

```

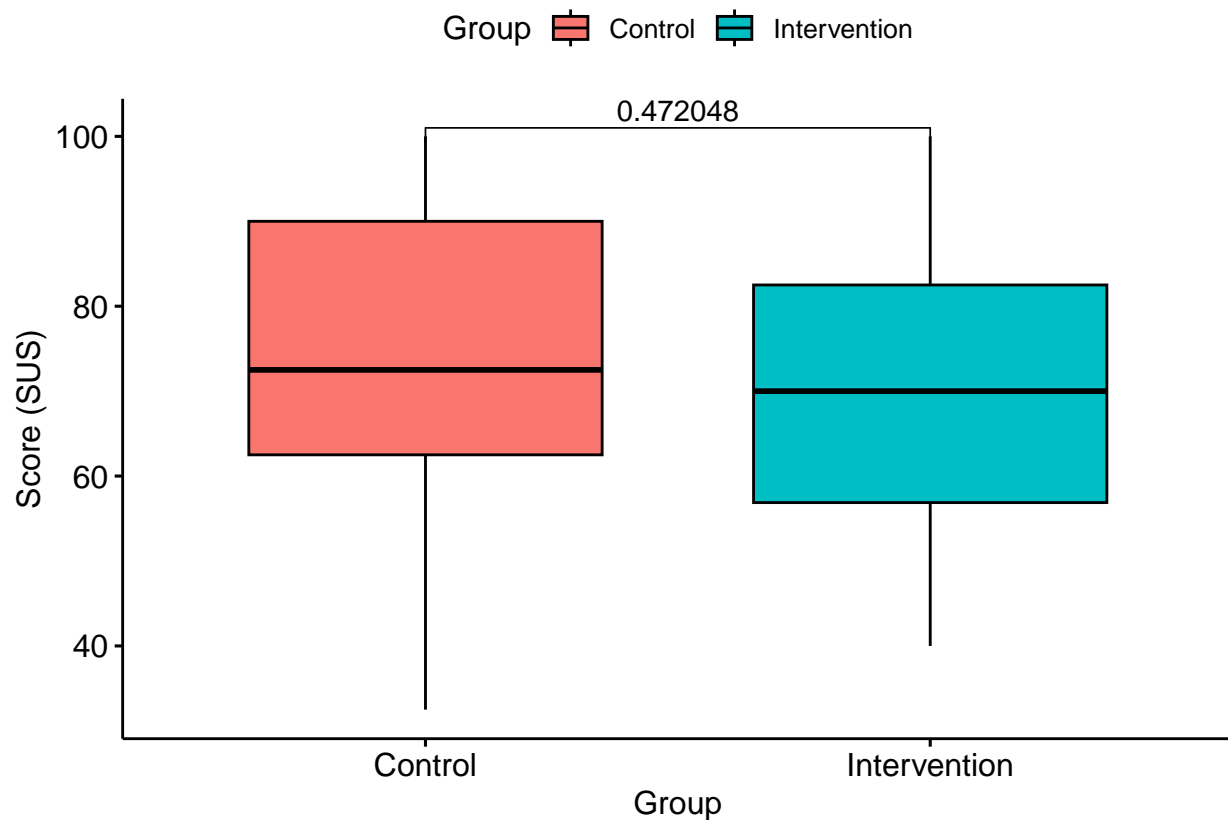
## [1] 101

```

```

# Boxplot
ggboxplot(data_sus, x = "Group", y = "Score",
  fill = "Group",
  palette = "Okabe-Ito",
  ylab = "Score (SUS)") +
  stat_pvalue_manual(stat.test_group,
    y.position = y_positions,
    step.increase = 0.2,
    label = "p.adj")

```



Vergleich: Für Österreich Kontroll vs. Intervention

```
# Filter data_sus by Austria
austria_data <- dplyr::filter(data_sus, Country == "Austria")

# Perform Kruskal-Wallis test for educ
kruskal_group <- kruskal.test(Score ~ Group, data = austria_data)
kruskal_group

##
## Kruskal-Wallis rank sum test
##
## data: Score by Group
## Kruskal-Wallis chi-squared = 1.6468, df = 1, p-value = 0.1994

# Post hoc tests

# Perform pairwise Wilcoxon tests
wilcox_group <- pairwise.wilcox.test(austria_data$Score, austria_data$Group, p.adjust.method = "BH")

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen
```

```

# Extract p-values
p_values_group <- wilcox_group$p.value

print(p_values_group)

```

```

##           Control
## Intervention 0.2057944

```

```

# Convert p-value matrix to a vector
p_values_group_vec <- as.vector(p_values_group)

# Remove NA values (these are for comparisons with themselves)
p_values_group <- p_values_group_vec[!is.na(p_values_group_vec)]

# Create data frames for manual p-value annotation
stat.test_group <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "Control", "Intervention", round(p_values_group[1],6)
)

stat.test_group

```

```

## # A tibble: 1 x 3
##   group1 group2      p.adj
##   <chr>  <chr>    <dbl>
## 1 Control Intervention 0.206

```

```

# Determine the maximum value of the 'Score' column
max_score <- max(austria_data$Score, na.rm = TRUE)

# Create an array of y positions starting from the maximum Score value and incrementing by a small amount
y_positions <- max_score + seq(from = 1, to = 1, by = 1)

y_positions

```

```

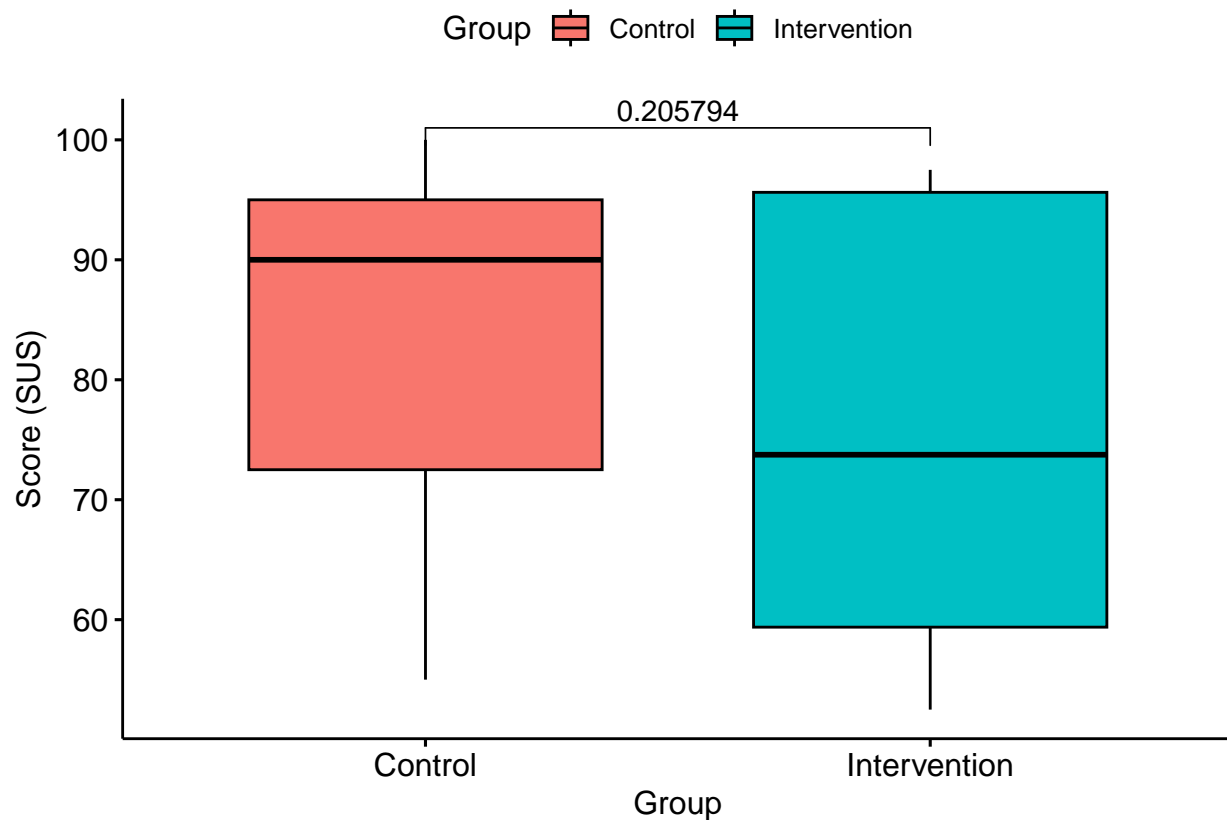
## [1] 101

```

```

# Boxplot
ggboxplot(austria_data, x = "Group", y = "Score",
  fill = "Group",
  palette = "Okabe-Ito",
  ylab = "Score (SUS)") +
  stat_pvalue_manual(stat.test_group,
    y.position = y_positions,
    step.increase = 0.2,
    label = "p.adj")

```



Vergleich: Für alle Länder Kontroll vs Intervention

```
# Get unique countries
countries <- unique(data_sus$Country)

# Initialize an empty list to store plots
plots <- list()

# For each country
for (country in countries) {
  # Filter data_sus by country
  country_data <- dplyr::filter(data_sus, Country == country)

  # Perform Kruskal-Wallis test for educ
  kruskal_group <- kruskal.test(Score ~ Group, data = country_data)

  # Post hoc tests

  # Perform pairwise Wilcoxon tests
  wilcox_group <- pairwise.wilcox.test(country_data$Score,
                                       country_data$Group,
                                       p.adjust.method = "BH")

  # Extract p-values
  p_values_group <- wilcox_group$p.value
```

```

# Convert p-value matrix to a vector
p_values_group_vec <- as.vector(p_values_group)

# Remove NA values (these are for comparisons with themselves)
p_values_group <- p_values_group_vec[!is.na(p_values_group_vec)]

# Create data frames for manual p-value annotation
stat.test_group <- tibble::tribble(
  ~group1, ~group2, ~p.adj,
  "Control", "Intervention", round(p_values_group[1],6)
)

# Determine the maximum value of the 'Score' column
max_score <- max(country_data$Score, na.rm = TRUE)

# Create an array of y positions starting from the maximum Score value and incrementing by a small amount
y_positions <- max_score + seq(from = 1, to = 1, by = 1)

# Boxplot
plot <- ggboxplot(country_data, x = "Group", y = "Score",
  fill = "Group",
  palette = "Okabe-Ito",
  ylab = "Score (SUS)") +
  stat_pvalue_manual(stat.test_group,
    y.position = y_positions,
    step.increase = 0.2,
    label = "p.adj")

# Add the plot to the list
plots[[country]] <- plot
}

```

```

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

```

```

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

```

```

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

```

```

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): kann bei
## Bindungen keinen exakten p-Wert Berechnen

```

```

# Print all plots
for (country in countries) {
  print(plots[[country]])
}

```

