Exercise 12. Tests for multiple samples

Michal Béreš, Martina Litschmannová, Adéla Vrtková

Test data for a function call example

In [1]:

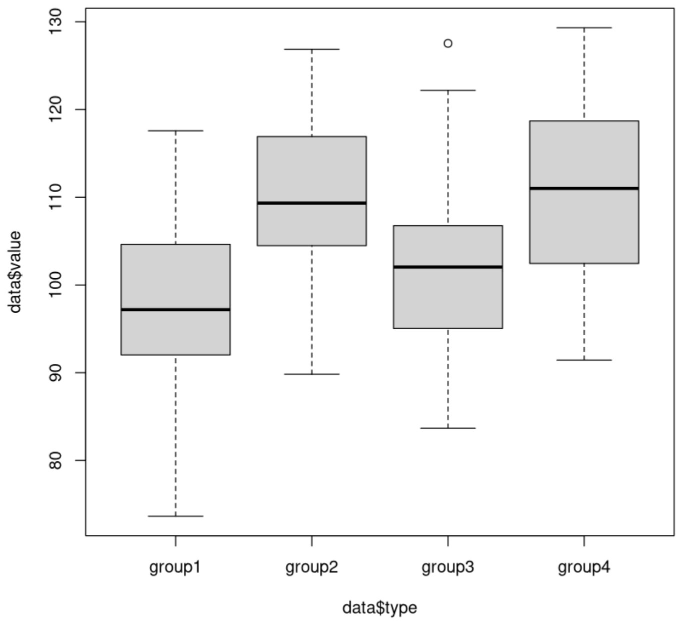
```
# I will create some data from the normal distribution with same variances
a = as.data.frame(rnorm(n = 35, mean = 100, sd = 10))
b = as.data.frame(rnorm(n = 30, mean = 108, sd = 10))
c = as.data.frame(rnorm(n = 40, mean = 104, sd = 10))
d = as.data.frame(rnorm(n = 32, mean = 112, sd = 10))

# I will rename the column name
colnames(a) = c("value")
colnames(b) = c("value")
colnames(c) = c("value")
colnames(d) = c("value")

# I will add a type for all frame data
a$type = "group1"
b$type = "group2"
c$type = "group3"
d$type = "group4"

# I glue the lines together
data = rbind(a,b,c,d)

# Convert type to type factor (needed for some tests)
data$type = as.factor(data$type)

head(data)
```

A data.frame: 6 × 2

	value	type
	<dbl>	<fct>
1	97.18982	group1
2	84.31632	group1
3	87.86475	group1
4	117.57793	group1
5	104.55862	group1
6	83.05962	group1

In [2]:

```
boxplot(data$value ~ data$type)
# if there are any outliars, I will ignore them
# I know the data is from a normal distribution!
# I also know they have the same variance sd = 10
```



Overview of tests and their functions

Comparing the measures of Variability (variances)

Bartlett test

- verifies the equality of variances
- $H_0 : \sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \dots$
- $H_A : \neg H_0$
- the assumtion data normality(and of course independence and continuity)

In [3]:

```
bartlett.test(data$value ~ data$type)
```

```
	Bartlett test of homogeneity of variances

data:  data$value by data$type
Bartlett's K-squared = 0.38409, df = 3, p-value = 0.9435
```

Levene's test

- verifies the equality of variances
- $H_0 : \sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \dots$
- $H_A : \neg H_0$
- only independence and continuity are required

In [4]:

```
car::leveneTest(data$value ~ data$type)
```

A anova: 2 × 3

	Df	F value	Pr(>F)
	<int>	<dbl>	<dbl>
group	3	0.2102347	0.8891685
	133	NA	NA

Cochran's and Hartley's test

- verifies the equality of variances
- require data normality and so-called balanced sorting
    - balanced sorting means that we have approximately the same amount of data in each group

- we will not use them

Comparing the measures of Position (means or medians)

ANOVA(analysis of variance)

- test the equality of mean values
- $H_0 : \mu_1 = \mu_2 = \mu_3 = \dots$
- $H_A : \neg H_0$
- prerequisites:
  - normality dat
  - homoskedasticity(identical variances)
  - (and of course independence and continuity)
- if we reject H_0 Post-Hoc analysis is required
  - using TukeyHSD test
  - we want the result in the form of letter scheme and effects

In [5]:

```
# ANOVA
# H0: mu1=mu2=mu3=mu4
# HA:~H0(H0 negation)

res = aov(data$value~ data$type)
summary(res)
```

```
              Df Sum Sq Mean Sq F value   Pr(>F)    
data$type      3   3962  1320.5   13.09 1.53e-07 ***
Residuals    133  13419   100.9                     
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

In [6]:

```
# Post-Hoc analysis

TukeyHSD(res)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = data$value ~ data$type)

$`data$type`
                   diff        lwr        upr     p adj
group2-group1 11.366326   4.863966 17.8686865 0.0000707
group3-group1  4.371996  -1.676893 10.4208849 0.2413579
group4-group1 13.611689   7.219680 20.0036978 0.0000009
group3-group2 -6.994330 -13.306336 -0.6823245 0.0235030
group4-group2  2.245362  -4.396184  8.8869089 0.8154219
group4-group3  9.239693   3.041426 15.4379593 0.0009369
```

In [7]:

```
# effect computation
library(dplyr)

# overall average
mean_overall = mean(data$value)
mean_overall

# averages in groups
effects = data %>% group_by(type) %>%
    summarize(mean_group = mean(value))

# effects
effects$effect = effects$mean_group - mean_overall

# list them sorted
effects %>% arrange(desc(effect))
```

```
Attaching package: ‘dplyr’
```

```
The following objects are masked from ‘package:stats’:

    filter, lag


The following objects are masked from ‘package:base’:

    intersect, setdiff, setequal, union

```

```
104.686863481787
```

A tibble: 4 × 3

type	mean_group	effect
<fct>	<dbl>	<dbl>
group4	111.35371	6.666844
group2	109.10835	4.421482
group3	102.11402	-2.572848
group1	97.74202	-6.944844

In [8]:

```r
# letter scheme, library rcompanion

# install.packages("rcompanion")
posthoc = TukeyHSD(res)

# how to get the matrix of values out of the result
matrix_posthoc = posthoc[[1]]
matrix_posthoc
# now we make a dataframe with columns of pairs and pvalues
posthoc_DF = data.frame(pairs = rownames(matrix_posthoc),
                        pval = matrix_posthoc[,'p adj'])
posthoc_DF
```

A matrix: 6 × 4 of type dbl

	diff	lwr	upr	p adj
group2-group1	11.366326	4.863966	17.8686865	7.065272e-05
group3-group1	4.371996	-1.676893	10.4208849	2.413579e-01
group4-group1	13.611689	7.219680	20.0036978	9.272762e-07
group3-group2	-6.994330	-13.306336	-0.6823245	2.350298e-02
group4-group2	2.245362	-4.396184	8.8869089	8.154219e-01
group4-group3	9.239693	3.041426	15.4379593	9.368725e-04

A data.frame: 6 × 2

	pairs	pval
	<chr>	<dbl>
group2-group1	group2-group1	7.065272e-05
group3-group1	group3-group1	2.413579e-01
group4-group1	group4-group1	9.272762e-07
group3-group2	group3-group2	2.350298e-02
group4-group2	group4-group2	8.154219e-01
group4-group3	group4-group3	9.368725e-04

In [9]:

```r
rcompanion::cldList(pval ~ pairs,
        data = posthoc_DF,
        threshold = 0.05)
```

A data.frame: 4 × 3

Group	Letter	MonoLetter
<chr>	<chr>	<chr>
group2	a	a

Group	Letter	MonoLetter
<chr>	<chr>	<chr>
group3	b	b
group4	a	a
group1	b	b

Kruskal - Wallis test

- verifiesthe equality of medians
- $H_0 : X_{0.5,1} = X_{0.5,2} = X_{0.5,3} = \dots$
- $H_A : \neg H_0$
- prerequisites:
  - data symmetry
  - (and of course independence and continuity)
- if we reject H_0 Post-Hoc analysis is required
  - using the Dunn test
    - method = "bonferroni"
  - we want the result in the form of letter scheme and effects

In [10]:

```
# KW test
# H0: X0.5,1=X0.5,2=X0.5,3=X0.5,4
# HA:~H0(H0 negation)

kruskal.test(data$value ~ data$type)
```

```
        Kruskal-Wallis rank sum test

data:  data$value by data$type
Kruskal-Wallis chi-squared = 30.394, df = 3, p-value = 1.14e-06
```

In [11]:

```
# Post-Hoc analysis

# install.packages("FSA")
FSA::dunnTest(data$value ~ data$type,    # FSA library
              method="bonferroni")
```

```
Registered S3 methods overwritten by 'FSA':
  method       from
  confint.boot car
  hist.boot    car

Dunn (1964) Kruskal-Wallis multiple comparison

  p-values adjusted with the Bonferroni method.

```

```
       Comparison         Z      P.unadj        P.adj
1 group1 - group2 -4.037296 5.407089e-05 3.244253e-04
2 group1 - group3 -1.496284 1.345796e-01 8.074778e-01
3 group2 - group3  2.725139 6.427437e-03 3.856462e-02
4 group1 - group4 -4.774603 1.800624e-06 1.080374e-05
5 group2 - group4 -0.642524 5.205330e-01 1.000000e+00
6 group3 - group4 -3.463621 5.329565e-04 3.197739e-03
```

In [12]:

```
# effects

# overall median
median_overall = median(data$value)
median_overall

# medians in groups
effects = data %>% group_by(type) %>%
    summarize(median_group = median(value))

# effects
effects$effect = effects$median_group - median_overall
```

```
# list them sorted
effects %>% arrange(desc(effect))
```

104.625102209417

A tibble: 4 × 3

type	median_group	effect
<fct>	<dbl>	<dbl>
group4	111.01275	6.387651
group2	109.33532	4.710219
group3	102.04492	-2.580181
group1	97.18982	-7.435283

In [13]:

```
# letter scheme, library rcompanion

# install.packages("rcompanion")
posthoc = FSA::dunnTest(data$value ~ data$type,    # FSA library
              method="bonferroni")

# how to get the matrix of values out of the result
posthoc_DF = posthoc$res
posthoc_DF
# its in the data frame form already
```

A data.frame: 6 × 4

Comparison	Z	P.unadj	P.adj
<chr>	<dbl>	<dbl>	<dbl>
group1 - group2	-4.037296	5.407089e-05	3.244253e-04
group1 - group3	-1.496284	1.345796e-01	8.074778e-01
group2 - group3	2.725139	6.427437e-03	3.856462e-02
group1 - group4	-4.774603	1.800624e-06	1.080374e-05
group2 - group4	-0.642524	5.205330e-01	1.000000e+00
group3 - group4	-3.463621	5.329565e-04	3.197739e-03

In [14]:

```
rcompanion::cldList(P.adj ~ Comparison,
        data = posthoc_DF,
        threshold = 0.05)
```

A data.frame: 4 × 3

Group	Letter	MonoLetter
<chr>	<chr>	<chr>
group1	a	a
group2	b	b
group3	a	a
group4	b	b

Examples

Example 1.

122 patients who underwent heart surgery were randomly divided into three groups

- **Group 1:** Patients received 50% nitrous oxide and 50% oxygen mixed continuously for 24 hours.
- **Group 2:** Patients received 50% nitric oxide and 50% oxygen only during surgery.
- **Group 3:** Patients received no nitrous oxide but received 35-50% oxygen for 24 hours.

The data in the sheet 1 of testy_vicevyberove.xlsx file correspond to the folic acid salt concentrations in the red blood cells in all three groups 24 hours after the surgery. Verify that the observed differences between the folic acid salt concentrations are statistically significant, i.e. that there is an effect of the composition of the mixture on the monitored parameter.

In [15]:

```
acid = readxl::read_excel("data/testy_vicevyberove.xlsx", sheet=1)
colnames(acid) = c("Group 1","Group 2","Group 3")    # rename columns
head(acid)
```

A tibble: 6 × 3

Group 1	Group 2	Group 3
<dbl>	<dbl>	<dbl>
234	267	351
306	301	284
261	253	291
255	278	270
267	216	205
304	188	318

In [16]:

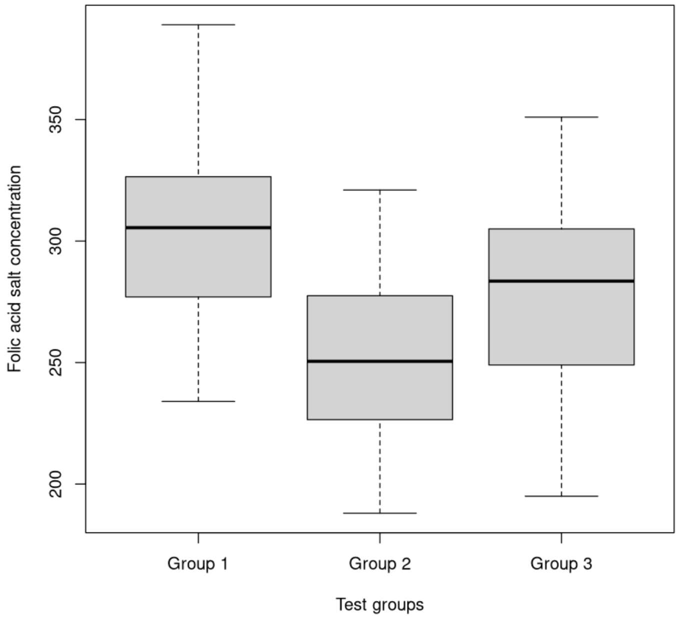
```
# conversion to standard data format
acid.s = stack(acid)
colnames(acid.s) = c("values","group")
acid.s = na.omit(acid.s)
head(acid.s)
```

A data.frame: 6 × 2

	values	group
	<dbl>	<fct>
1	234	Group 1
2	306	Group 1
3	261	Group 1
4	255	Group 1
5	267	Group 1
6	304	Group 1

In [17]:

```
boxplot(acid.s$values ~ acid.s$group, xlab = "Test groups", ylab = "Folic acid salt concentration")
# Data do not contain any outliars
```



In [18]:

```
# we test the normality using S-W. test
```

```
acid.s %>% group_by(group) %>%
    summarise(pval_SW = shapiro.test(values)$p.value)
```

A tibble: 3 × 2

group	pval_SW
<fct>	<dbl>
Group 1	0.9767689
Group 2	0.7705138
Group 3	0.5249177

In [19]:

```
# Information needed to set rounding

acid.s %>% group_by(group) %>%
    summarise(len = length(values), st.dev = sd(values))

# sd is rounded to 3 valid digits
# sd and position measures are rounded to tenths
```

A tibble: 3 × 3

group	len	st.dev
<fct>	<int>	<dbl>
Group 1	40	33.78468
Group 2	40	34.02291
Group 3	42	38.49036

In [20]:

```
# equality of variance
s2 = acid.s %>% group_by(group) %>%
        summarise(var = sd(values)^2)
s2 # sampling variances

max(s2$var)/min(s2$var)
# According to the box chart and information on the ratio of the largest and smallest
# variances(<2) we do not assume that the variances differ statistically significantly
```

A tibble: 3 × 2

group	var
<fct>	<dbl>
Group 1	1141.404
Group 2	1157.558
Group 3	1481.508

1.29796942864859

In [21]:

```
# The assumption of normality was not rejected -> Bartlett's test

bartlett.test(acid.s$values ~ acid.s$group)

# At the significance level of 0.05, there are no statistically significant differences in variances
```

```
	Bartlett test of homogeneity of variances

data:  acid.s$values by acid.s$group
Bartlett's K-squared = 0.87826, df = 2, p-value = 0.6446
```

In [22]:

```
# We want to compare the mean values of independent samples from normal distributions
# with same variances -> ANOVA
# The aov() command requires data in the standard data format

results = aov(acid.s$values ~ acid.s$group)
summary(results)

# At the significance level of 0.05, there are statistically significant differences in mean values
```

```
             Df Sum Sq Mean Sq F value   Pr(>F)    
acid.s$group  2  56502   28251   22.35 5.73e-09 ***
```

```
Residuals    119 150401    1264
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

In [23]:

```
# post-hoc analysis
TukeyHSD(results)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = acid.s$values ~ acid.s$group)

$`acid.s$group`
                     diff        lwr       upr     p adj
Group 2-Group 1 -53.15000 -72.017226 -34.28277 0.0000000
Group 3-Group 1 -26.90833 -45.549597  -8.26707 0.0024094
Group 3-Group 2  26.24167   7.600403  44.88293 0.0031788
```

In [24]:

```
# effect computation

# overall average
mean_overall = mean(acid.s$values)
mean_overall

# averages in groups
effects = acid.s %>% group_by(group) %>%
    summarize(mean_group = mean(values))

# effects
effects$effect = effects$mean_group - mean_overall

# list them sorted
effects %>% arrange(desc(effect))
```

277.385245901639

A tibble: 3 × 3

group	mean_group	effect
<fct>	<dbl>	<dbl>
Group 1	304.0750	26.6897541
Group 3	277.1667	-0.2185792
Group 2	250.9250	-26.4602459

In [25]:

```
# letter scheme, library rcompanion

# make a dataframe with columns of pairs and pvalues
matrix_posthoc = TukeyHSD(results)[[1]]
posthoc_DF = data.frame(pairs = rownames(matrix_posthoc),
                   pval = matrix_posthoc[,'p adj'])
# letter scheme
rcompanion::cldList(pval ~ pairs,
        data = posthoc_DF,
        threshold = 0.05)
```

A data.frame: 3 × 3

Group	Letter	MonoLetter
<chr>	<chr>	<chr>
Group2	a	a
Group3	b	b
Group1	c	c

Example 2.

Three breeds of rabbits are bred on the farm. An experiment was performed on sheet 2 of testy_vicevyberove.xlsx, the aim of which was to find out whether, even if we keep all the rabbits for the same time and under the same conditions (food, environment), there is a statistically significant difference between breeds in rabbit weights. Verify.

In [26]:

```
rabbits = readxl::read_excel("data/testy_vicevyberove.xlsx", sheet=2)
colnames(rabbits) = c("Vienna","Czech","Kalif")    # rename columns
head(rabbits)
```

```
A tibble: 6 × 3
Vienna  Czech   Kalif
<dbl>   <dbl>   <dbl>
 4.125  3.518   4.902
 3.923  3.464   5.228
 4.046  3.337   4.950
 4.247  3.669   5.054
 3.869  3.642   5.048
 4.094  3.440   4.970
```

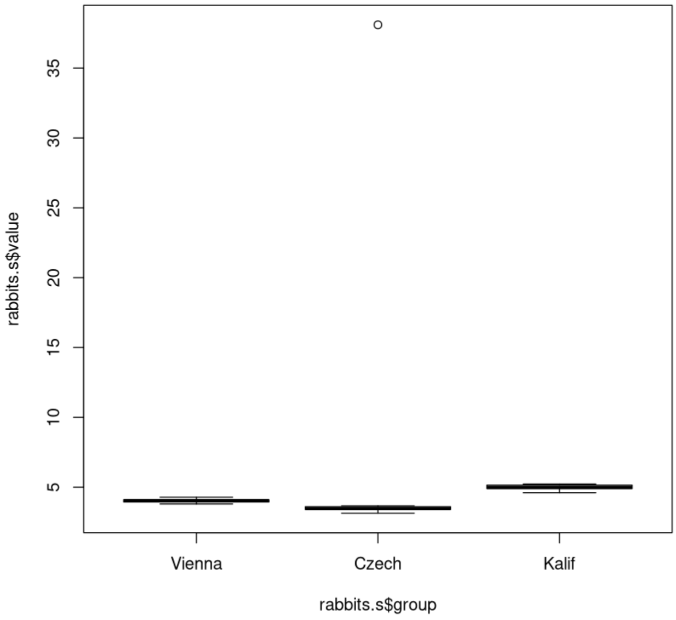
In [27]:

```
# conversion to standard data format
rabbits.s = stack(rabbits)
colnames(rabbits.s) = c("value","group")
rabbits.s = na.omit(rabbits.s)
head(rabbits.s)
```

```
A data.frame: 6 × 2
     value   group
     <dbl>   <fct>
1    4.125   Vienna
2    3.923   Vienna
3    4.046   Vienna
4    4.247   Vienna
5    3.869   Vienna
6    4.094   Vienna
```

In [28]:

```
boxplot(rabbits.s$value ~ rabbits.s$group)
# data contains an outliar
```



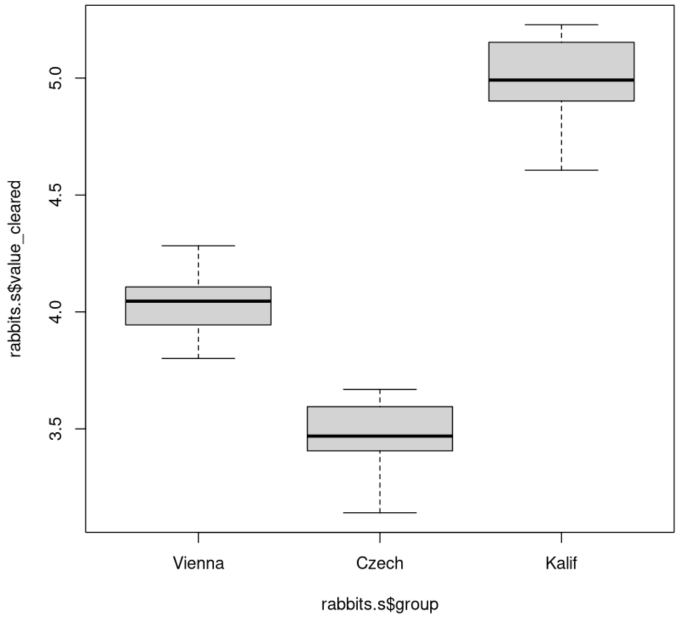
In [29]:

```
# Eliminate outliar

rabbits.s$id = seq(1,length(rabbits.s$value))
```

```
outliars = rabbits.s %>% group_by(group) %>% rstatix::identify_outliers(value)
outliars

rabbits.s$value_cleared = ifelse(rabbits.s$id %in% outliars$id, NA, rabbits.s$value)

# Box chart
boxplot(rabbits.s$value_cleared ~ rabbits.s$group)
```

A tibble: 1 × 5

group	value	id	is.outlier	is.extreme
<fct>	<dbl>	<int>	<lgl>	<lgl>
Czech	38.1	32	TRUE	TRUE



In [30]:

```
library(dplyr)

rabbits.s %>% group_by(group) %>%
    summarise(norm.pval = shapiro.test(value_cleared)$p.value)

# At the significance level of 0.05, we do not reject the assumption of normality.
```

A tibble: 3 × 2

group	norm.pval
<fct>	<dbl>
Vienna	0.8247350
Czech	0.2775194
Kalif	0.1685629

In [31]:

```
# Information needed for correct rounding
rabbits.s %>% group_by(group) %>%
    summarize(len = sum(!is.nan(value_cleared)),
              sd = sd(value_cleared, na.rm = TRUE))

# sd is rounded to 2 valid digits
# sd and position measurements round to hundredths
```

A tibble: 3 × 3

group	len	sd
<fct>	<int>	<dbl>
Vienna	23	0.1270971
Czech	23	0.1393983
Kalif	18	0.1859894

In [32]:

```
# The assumption of normality was not rejected ->Bartlett's test
bartlett.test(rabbits.s$value_cleared ~ rabbits.s$group)

# At the significance level of 0.05, the equality of variances cannot be rejected
```

```
	Bartlett test of homogeneity of variances

data:  rabbits.s$value_cleared by rabbits.s$group
Bartlett's K-squared = 3.0553, df = 2, p-value = 0.217
```

In [33]:

```
# We want to compare the mean values of independent samples from normal
# distributions with the same variances -> ANOVA
# The aov() command requires data in the standard data format

results = aov(rabbits.s$value_cleared ~ rabbits.s$group)
summary(results)

# At the significance level of 0.05, we reject the hypothesis of equality of the mean values
```

```
                Df Sum Sq Mean Sq F value Pr(>F)    
rabbits.s$group  2 22.943  11.472   509.3 <2e-16 ***
Residuals       60  1.352   0.023                   
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
1 observation deleted due to missingness
```

In [34]:

```
# post-hoc analysis
TukeyHSD(results)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = rabbits.s$value_cleared ~ rabbits.s$group)

$`rabbits.s$group`
                   diff        lwr        upr p adj
Czech-Vienna -0.5613577 -0.6689197 -0.4537957     0
Kalif-Vienna  0.9539251  0.8404189  1.0674313     0
Kalif-Czech   1.5152828  1.4006497  1.6299160     0
```

In [36]:

```
# effect computation

# overall average
mean_overall = mean(rabbits.s$value_cleared, na.rm = TRUE)
mean_overall

# averages in groups
effects = rabbits.s %>% group_by(group) %>%
    summarize(mean_group = mean(value_cleared, na.rm = TRUE))

# effects
effects$effect = effects$mean_group - mean_overall

# list them sorted
effects %>% arrange(desc(effect))
```

4.11465079365079

A tibble: 3 × 3

group	mean_group	effect
<fct>	<dbl>	<dbl>
Kalif	4.992056	0.87740476
Vienna	4.038130	-0.07652036
Czech	3.476773	-0.63787807

Example 3.

Four manufacturers A, B, C, D sent a total of 66 products to the competition for the best product quality. The jury compiled the ranking (only the position of the product in the list of 66 from best to worst), which is listed in the sheet 3 of the file testy_vicevyberove.xlsx. On the basis of the above data, assess whether the origin of the products affects its quality.

In [37]:

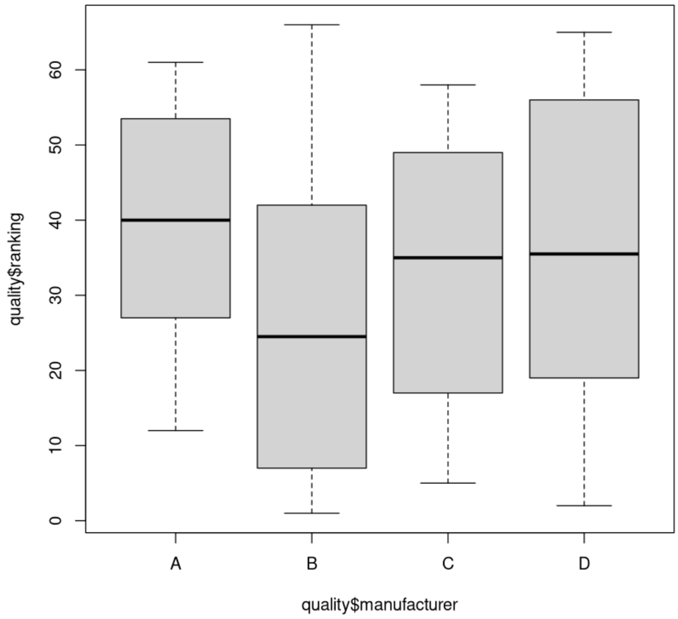
```
quality = readxl::read_excel("data/testy_vicevyberove.xlsx", sheet = 3)
colnames(quality) = c("ranking", "manufacturer")    # rename columns
head(quality)
# data is already in standard format
```

A tibble: 6 × 2

ranking	manufacturer
<dbl>	<chr>
1	B
2	D
3	B
4	B
5	C
6	B

In [38]:

```
boxplot(quality$ranking ~ quality$manufacturer)

# the data are not independent by nature, also they are not continuous!
# outliars should not be present by the nature of the dataset
# the assumptions are corrupted for all tests -> our best bet is the most robust test in our arsenal
# we skip directly into KW test
```



In [39]:

```
# Symmetry verification

quality %>% group_by(manufacturer) %>%
    summarize(skewness = moments::skewness(ranking))
```

A tibble: 4 × 2

manufacturer	skewness
<chr>	<dbl>
A	-0.211238463
B	0.459593402
C	-0.147647782
D	0.009223978

In [40]:

```
# We want to compare the medians of "independent" samples -> Kruskal-Wallis test
kruskal.test(quality$ranking ~ quality$manufacturer)

# At the significance level of 0.05, there are no statistically significant differences in medians
```

```
	Kruskal-Wallis rank sum test

data:  quality$ranking by quality$manufacturer
Kruskal-Wallis chi-squared = 3.7032, df = 3, p-value = 0.2953
```

Example 4.

The effect of three types of medicaments on blood clotting was studied (so called thrombin time). Data of 42 monitored persons are recorded in the sheet 4 of the file testy_vicevyberove.xlsx. Does the thrombin time depend on which preparation was used?

In [41]:

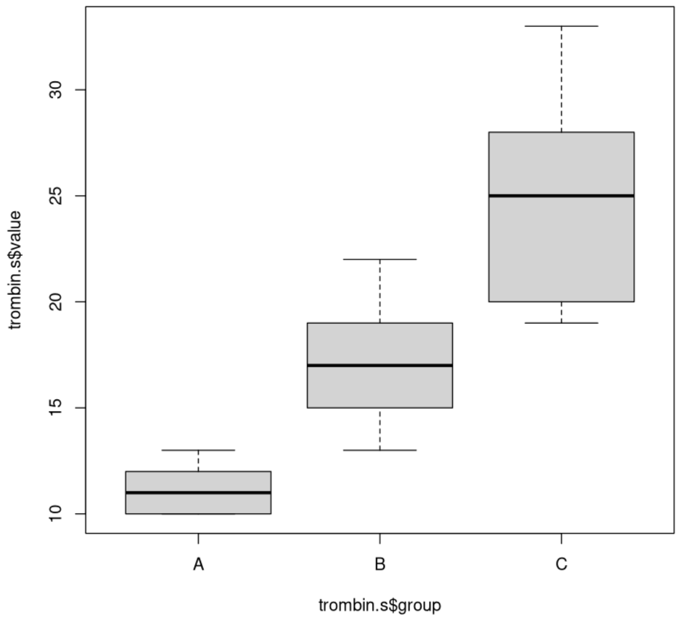
```
trombin.s = readxl::read_excel("data/testy_vicevyberove.xlsx",
                               sheet=4, skip = 1)
colnames(trombin.s) = c("value","group")    # rename columns

head(trombin.s)
# data is already in standard format
```

A tibble: 6 × 2

value	group
<dbl>	<chr>
12	A
10	A
10	A
12	A
10	A
12	A

In [42]:

```
# exploratory analysis
boxplot(trombin.s$value ~ trombin.s$group)
# no outliars
```



In [43]:

```
# verification of normality
library(dplyr)

trombin.s %>% group_by(group) %>%
    summarize(norm.pval = shapiro.test(value)$p.value)

# At the significance level of 0.05 we reject the assumption of normality(for A)
```

A tibble: 3 × 2

group	norm.pval
<chr>	<dbl>
A	0.03179805
B	0.94597139
C	0.27138568

In [44]:

```
# we can at least test the equality of variances -> same variances
# means better KW test result in terms of type II error

# The assumption of normality was rejected -> Levene's test

car::leveneTest(trombin.s$value ~ trombin.s$group)

# the assumption of homoskedasticity was rejected (at significance 0.05)
```

```
Warning message in leveneTest.default(y = y, group = group, ...):
“group coerced to factor.”
```

A anova: 2 × 3

	Df	F value	Pr(>F)
	<int>	<dbl>	<dbl>
group	2	9.390456	0.0004687749
	39	NA	NA

In [45]:

```
# Symmetry verification
trombin.s %>% group_by(group) %>%
  summarize(skewness = moments::skewness(value))
# we do not reject the assumption of data symmetry
```

A tibble: 3 × 2

group	skewness
<chr>	<dbl>
A	0.5400617
B	0.2886751
C	0.2975920

In [46]:

```
# We want to compare medians (data not from normal dist.)-> Kruskal - Wallis test

kruskal.test(trombin.s$value,trombin.s$group)

# At the significance level of 0.05, we found statistically significant differences in medians
```

```
	Kruskal-Wallis rank sum test

data:  trombin.s$value and trombin.s$group
Kruskal-Wallis chi-squared = 34.535, df = 2, p-value = 3.169e-08
```

In [47]:

```
FSA::dunnTest(trombin.s$value~trombin.s$group,method = "bonferroni")
```

```
Warning message:
“trombin.s$group was coerced to a factor.”
Dunn (1964) Kruskal-Wallis multiple comparison

  p-values adjusted with the Bonferroni method.
```

```
  Comparison         Z      P.unadj        P.adj
1      A - B -3.189477 1.425303e-03 4.275909e-03
```

```
2      A - C -5.869256 4.377545e-09 1.313264e-08
3      B - C -2.679779 7.367082e-03 2.210125e-02
```

In [48]:

```
# effect counting
library(dplyr)

# overall average
median_overall = median(trombin.s$value)
median_overall

# averages in groups
effects = trombin.s %>% group_by(group) %>%
    summarize(median_group = median(value))

# effects
effects$effect = effects$median_group - median_overall

# List sorted
effects %>% arrange(desc(effect))
```

17

A tibble: 3 × 3

group	median_group	effect
<chr>	<dbl>	<dbl>
C	25	8
B	17	0
A	11	-6

Example 5.(multiple groups)

When Snow White got to the seven dwarves, she sensed an opportunity to make a lot of money. The Dwarves basically fell in love with the Snow White and immediately handed over all of their mined gold. However, even this is not enough for Snow White and she feels that she could benefit more from the dwarves. Therefore, she began to record how many kilograms of gold a day she received from each of the dwarves(snehurka.xlsx). Verify that the dwarves differ in the amount of gold mined.

In [49]:

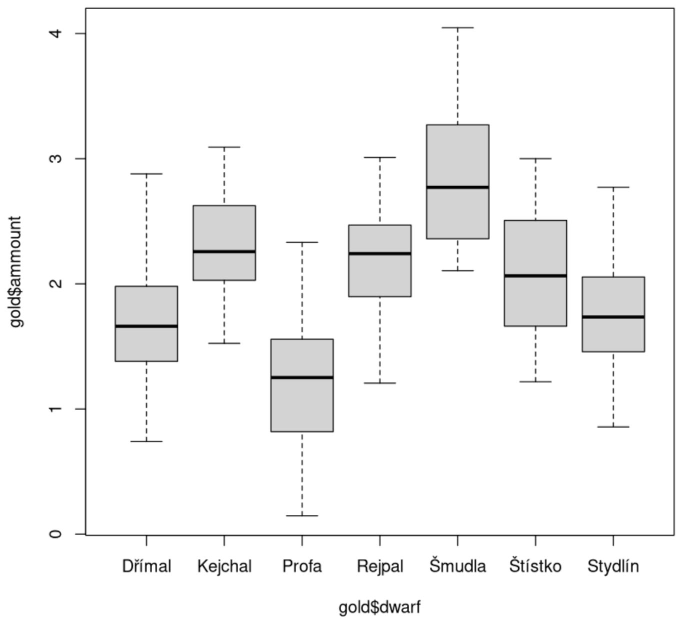
```
gold = readxl::read_excel("data/snehurka.xlsx")
colnames(gold) = c("ammount","dwarf")
head(gold)
# data is in the standard data format
```

A tibble: 6 × 2

ammount	dwarf
<dbl>	<chr>
0.8892697	Profa
1.5882393	Profa
2.0176732	Profa
1.2511435	Profa
2.1443305	Profa
1.2500689	Profa

In [50]:

```
boxplot(gold$ammount ~ gold$dwarf)
# data does not outliars
```



In [51]:

```
# verification of normality
library(dplyr)

gold %>% group_by(dwarf) %>%
    summarize(p.val = shapiro.test(ammount)$p.value)
```

A tibble: 7 × 2

dwarf	p.val
<chr>	<dbl>
Dřímal	0.8295308
Kejchal	0.8162545
Profa	0.8265843
Rejpal	0.6555333
Šmudla	0.1177217
Štístko	0.1866139
Stydlín	0.9177484

In [52]:

```
# The assumption of normality was not rejected -> Bartlett's test
bartlett.test(gold$ammount ~ gold$dwarf)

# At the significance level of 0.05, there are no statistically significant differences in variances
```

```
	Bartlett test of homogeneity of variances

data:  gold$ammount by gold$dwarf
Bartlett's K-squared = 5.1736, df = 6, p-value = 0.5217
```

In [53]:

```
# ANOVA
results = aov(gold$ammount ~ gold$dwarf)
summary(results)
```

```
             Df Sum Sq Mean Sq F value Pr(>F)    
gold$dwarf    6  50.71   8.451   35.87 <2e-16 ***
Residuals   210  49.47   0.236                   
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

In [54]:

```
# POST-HOC
res = TukeyHSD(results)[[1]]
res
```

A matrix: 21 × 4 of type dbl

	diff	lwr	upr	p adj
Kejchal-Dřímal	0.61338486	0.24635938	0.98041034	2.769364e-05
Profa-Dřímal	-0.45472230	-0.82174778	-0.08769682	5.265389e-03
Rejpal-Dřímal	0.54761573	0.18059025	0.91464121	2.871456e-04
Šmudla-Dřímal	1.18016854	0.81314306	1.54719402	2.609024e-14
Štístko-Dřímal	0.42198262	0.05495714	0.78900810	1.296949e-02
Stydlín-Dřímal	0.09891331	-0.26811217	0.46593879	9.845413e-01
Profa-Kejchal	-1.06810716	-1.43513263	-0.70108168	5.551115e-14
Rejpal-Kejchal	-0.06576913	-0.43279460	0.30125635	9.983200e-01
Šmudla-Kejchal	0.56678368	0.19975820	0.93380916	1.485215e-04
Štístko-Kejchal	-0.19140224	-0.55842772	0.17562324	7.125148e-01
Stydlín-Kejchal	-0.51447155	-0.88149703	-0.14744607	8.575061e-04
Rejpal-Profa	1.00233803	0.63531255	1.36936351	8.026912e-13
Šmudla-Profa	1.63489084	1.26786536	2.00191632	0.000000e+00
Štístko-Profa	0.87670492	0.50967944	1.24373039	3.719213e-10
Stydlín-Profa	0.55363560	0.18661013	0.92066108	2.339229e-04
Šmudla-Rejpal	0.63255281	0.26552733	0.99957829	1.346718e-05
Štístko-Rejpal	-0.12563312	-0.49265859	0.24139236	9.491834e-01
Stydlín-Rejpal	-0.44870243	-0.81572790	-0.08167695	6.246760e-03
Štístko-Šmudla	-0.75818592	-1.12521140	-0.39116044	8.055135e-08
Stydlín-Šmudla	-1.08125523	-1.44828071	-0.71422975	3.763656e-14
Stydlín-Štístko	-0.32306931	-0.69009479	0.04395617	1.250710e-01

In [55]:

```
# effects computation
library(dplyr)

# overall average
overall = mean(gold$ammount)
overall

# averages in groups
effects = gold %>% group_by(dwarf) %>%
    summarize(mean_dwarf = mean(ammount))

# effects
effects$effect = effects$mean_dwarf - overall

# list sorted
effects %>% arrange(desc(effect))
```

2.01366708938714

A tibble: 7 × 3

dwarf	mean_dwarf	effect
<chr>	<dbl>	<dbl>
Šmudla	2.849930	0.83626243
Kejchal	2.283146	0.26947875
Rejpal	2.217377	0.20370962
Štístko	2.091744	0.07807651
Stydlín	1.768674	-0.24499280
Dřímal	1.669761	-0.34390611
Profa	1.215039	-0.79862841

In [56]:

```
# letter scheme, library rcompanion

# make a dataframe with columns of pairs and pvalues
matrix_posthoc = TukeyHSD(results)[[1]]
```

```
posthoc_DF = data.frame(pairs = rownames(matrix_posthoc),
                        pval = matrix_posthoc[,'p adj'])
# letter scheme
rcompanion::cldList(pval ~ pairs,
        data = posthoc_DF,
        threshold = 0.05)
```

A data.frame: 7 × 3

Group	Letter	MonoLetter
<chr>	<chr>	<chr>
Kejchal	a	a
Profa	b	b
Rejpal	a	a
Šmudla	c	c
Štístko	ad	a d
Stydlín	de	de
Dřímal	e	e

In [ ]: