1 Jednofaktorová analýza rozptylu (ANOVA)

Příklad 1. Datový soubor clavicle.txt obsahuje měření největší délky pravé klíční kosti (proměnná cla.L, v mm) na lidských kostrách pocházejících ze tří populací (proměnná population): indické z Amritsaru (ind1), indické z Varansi (ind2) a řecké z Atén (gre). Na hladině významnosti $\alpha = 0.05$ testujte hypotézu, že se délka klíční kosti v populacích neliší.

Formulujte nulovou a alternativní hypotézu:

- H_0 : .....
- H_1 : .....

Předpoklady analýzy rozptylu:

1. nezávislost náhodných výběrů
2. .....
3. .....

Načteme data a podíváme se na ně. Je potřeba si pohlídat, že nezávislá proměnná je načtená jako kategoriální proměnná factor. To ověříme funkcí is.factor(). Pokud by tomu tak nebylo, změníme proměnou na kategoriální pomocí funkce factor(). Vyřadíme případná pozorování s chybějícími hodnotami a vypočítáme průměry a směrodatné odchylky pro každou populaci zvlášť i pro celý datový soubor společně. Podíváme se také na rozsahy jednotlivých výběrů.

```
data <- read.table("DATA/clavicle.txt", header=TRUE)
summary(data)

##  population sex         cla.L
##  gre :110   m:314   Min.   :123
##  ind1:120           1st Qu.:140
##  ind2: 84           Median :147
##                     Mean   :147
##                     3rd Qu.:153
##                     Max.   :174
##                     NA's   :19

is.factor(data$population)

## [1] TRUE

data <- na.omit(data)
(t <- table(data$population))

##
##  gre ind1 ind2
##   94  120   81

(n <- sum(table(data$population)))

## [1] 295

(m1 <- mean(data[data$population=='gre','cla.L']))

## [1] 153.5213
```

```
(m2 <- mean(data[data$population=='ind1','cla.L']))
```

```
## [1] 145.5667
```

```
(m3 <- mean(data[data$population=='ind2','cla.L']))
```

```
## [1] 141.4938
```

```
(m <- mean(data[,'cla.L']))
```

```
## [1] 146.9831
```

```
(s1 <- sd(data[data$population=='gre','cla.L']))
```

```
## [1] 9.118961
```

```
(s2 <- sd(data[data$population=='ind1','cla.L']))
```

```
## [1] 8.733432
```

```
(s3 <- sd(data[data$population=='ind2','cla.L']))
```

```
## [1] 8.220209
```

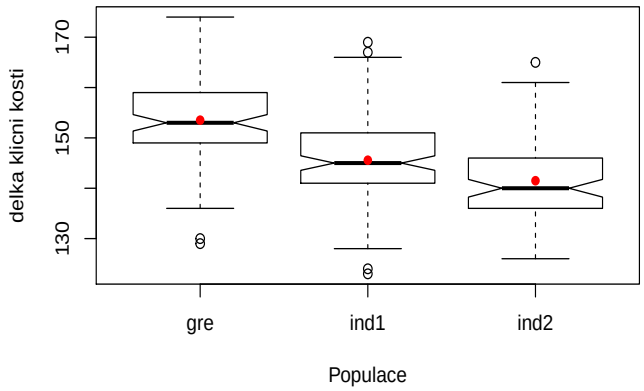
```
(s <- sd(data[,'cla.L']))
```

```
## [1] 9.917245
```

	rozsah	průměr	směrodatná odchylka
řecká populace			
indická populace z Amritsaru			
indická populace z Varansi			
celkový			

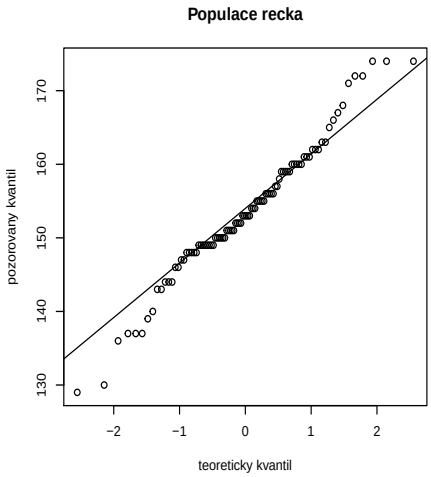
Vykreslíme si krabicové grafy.

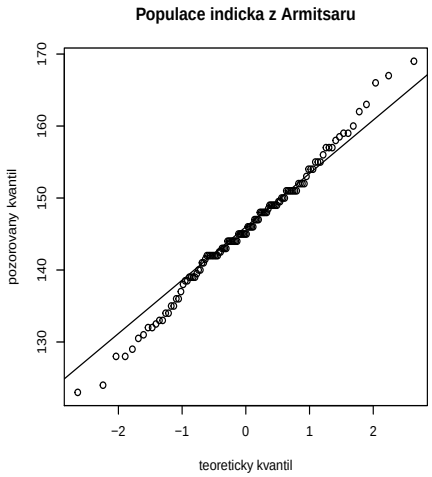
```
boxplot(cla.L ~ population, data=data, var.width=T, notch=T, xlab="Populace",
        ylab="delka klicni kosti")
points(1:3, c(m1,m2,m3), col="red", pch=16)
```

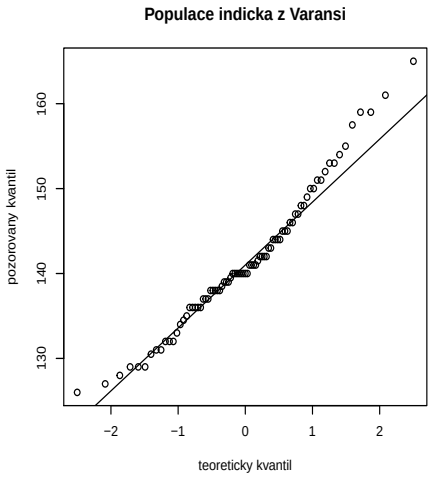


Pro ověření předpokladu ..... použijeme kvantil-kvantilový graf a Shapiro-Wilkův test.

```
par(mfrow=c(1,3))
qqnorm(data[data$population=='gre','cla.L'],main="Populace recka", xlab="teoreticky kvantil",
       ylab='pozorovany kvantil')
qqline(data[data$population=='gre','cla.L'])
shapiro.test(data[data$population=='gre','cla.L'])

##
##  Shapiro-Wilk normality test
##
## data:  data[data$population == "gre", "cla.L"]
## W = 0.97859, p-value = 0.1259

qqnorm(data[data$population=='ind1','cla.L'],main="Populace indicka z Armitsaru",
       xlab="teoreticky kvantil", ylab='pozorovany kvantil')
qqline(data[data$population=='ind1','cla.L'])
shapiro.test(data[data$population=='ind1','cla.L'])

##
##  Shapiro-Wilk normality test
##
## data:  data[data$population == "ind1", "cla.L"]
## W = 0.9915, p-value = 0.6741

qqnorm(data[data$population=='ind2','cla.L'],main="Populace indicka z Varansi",
       xlab="teoreticky kvantil", ylab='pozorovany kvantil')
qqline(data[data$population=='ind2','cla.L'])
```








```
shapiro.test(data[data$population=='ind2','cla.L'])

##
##  Shapiro-Wilk normality test
##
## data:  data[data$population == "ind2", "cla.L"]
## W = 0.97216, p-value = 0.07461
```

	hodnota testovací statistiky	p-hodnota	závěr
řecká populace			
indická populace z Amritsaru			
indická populace z Varansi			

Pro oveření předpokladu ..... použijeme Bartlettův test nebo Levenův test (z knihovny car).

```
bartlett.test(cla.L ~ population, data=data)

##
##  Bartlett test of homogeneity of variances
##
## data:  cla.L by population
## Bartlett's K-squared = 0.91592, df = 2, p-value = 0.6326

library("car")
leveneTest(cla.L ~ population, data=data)

## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value Pr(>F)
## group   2  0.3569 0.7002
##       292
```

Bartlettův test nabývá hodnoty ..... s p-hodnotou ....., hypotézu o ..... tedy ..... .
Levenův test nabývá hodnoty ..... s p-hodnotou ....., hypotézu o ..... tedy ..... .

Pakliže předpoklady jsou ....., můžeme přistoupit k samotné analýze rozptylu. K tomu se používá funkce aov().

```r
an1 <- aov(cla.L ~ population, data=data)
anova(an1)

## Analysis of Variance Table
##
## Response: cla.L
##             Df  Sum Sq Mean Sq F value    Pr(>F)
## population   2  6699.7  3349.9   44.03 < 2.2e-16 ***
## Residuals  292 22215.7    76.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

an1$coefficients

##    (Intercept) populationind1 populationind2
##      153.52128       -7.95461      -12.02745
```

Můžeme využít i funkci lm(), což je obecnější funkce pro lineární regresní model (toto téma bude později během semestru).

```r
an2 <- lm(cla.L ~ population, data=data)
summary(an2)

##
## Call:
## lm(formula = cla.L ~ population, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -24.5213  -4.5440  -0.5667   5.4333  23.5062
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    153.5213     0.8997 170.645  < 2e-16 ***
## populationind1  -7.9546     1.2014  -6.621 1.71e-10 ***
## populationind2 -12.0274     1.3224  -9.095  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.722 on 292 degrees of freedom
## Multiple R-squared:  0.2317,Adjusted R-squared:  0.2264
## F-statistic: 44.03 on 2 and 292 DF,  p-value: < 2.2e-16

anova(an2)

## Analysis of Variance Table
##
## Response: cla.L
##             Df  Sum Sq Mean Sq F value    Pr(>F)
## population   2  6699.7  3349.9   44.03 < 2.2e-16 ***
## Residuals  292 22215.7    76.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Skupinový součet čtverců S_A = ..., počet stupňů volnosti f_A = ..., reziduální součet čtverců S_E = ..., počet stupňů volnosti f_E = ..., testovací statistika F_A = ..., p-hodnota ....., nulovou hypotézu o shodě středních hodnot tedy na hladině významnosti $\alpha = 0.05$ ..... . Protože jsme hypotézu o shodnosti středních hodnot

....., chceme zjistit, které populace se mezi sebou liší. Přistoupíme tedy k mnohonásobnému porovnávání. V případě různých rozsahů můžeme použít Tukeyho HSD metodu, Scheffého metodu nebo Bonferroniho metodu.

Budeme testovat hypotézy:

- H_{01} : ..... vs H_{11} : .....
- H_{02} : ..... vs H_{12} : .....
- H_{03} : ..... vs H_{13} : .....

Podívejme se nejprve na Tukeyho HSD metodu.

```
TukeyHSD(aov(cla.L~population, data=data))
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = cla.L ~ population, data = data)
##
## $population
##                diff        lwr       upr     p adj
## ind1-gre   -7.95461 -10.784801 -5.124419 0.0000000
## ind2-gre  -12.02745 -15.142579 -8.912320 0.0000000
## ind2-ind1  -4.07284  -7.027641 -1.118038 0.0037145
```

Tukeyho HSD metoda ..... rovnost středních hodnot dvojic ....., protože odpovídající p-hodnoty jsou ..... než zvolená hladina významnosti $\alpha = 0.05$.

Scheffého metoda v R implementovaná není, musíme ji vypočítat ručně.

```
K <- 3
alpha <- 0.05
#gre vs ind1
abs(m1-m2)
```

```
## [1] 7.95461
```

```
s*sqrt((K-1)*(1/t[1] + 1/t[2]) * qf(1-alpha,K-1,n-K))
```

```
##      gre
## 3.360791
```

```
#gre vs ind2
abs(m1-m3)
```

```
## [1] 12.02745
```

```
s*sqrt((K-1)*(1/t[1] + 1/t[3]) * qf(1-alpha,K-1,n-K))
```

```
##      gre
## 3.699148

#ind1 vs ind2
abs(m2-m3)

## [1] 4.07284

s*sqrt((K-1)*(1/t[2] + 1/t[3]) * qf(1-alpha,K-1,n-K))

##     ind1
## 3.508763
```

Scheffého metoda ..... rovnost středních hodnot dvojic ....., protože odpovídající pravé strany jsou ..... než levé strany.

Nakonec se podíváme na Bonferroniho metodu.

```
pairwise.t.test(data$cla.L, data$population, p.adjust.method="bonferroni", pool.sd=T)

##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  data$cla.L and data$population
##
##      gre     ind1
## ind1 5.1e-10 -
## ind2 < 2e-16 0.0039
##
## P value adjustment method: bonferroni
```

Bonferroniho metoda ..... rovnost středních hodnot dvojic ....., protože odpovídající p-hodnoty jsou ..... než zvolená hladina významnosti $\alpha = 0.05$.

Závěr: