Podrobný dokument aj s teóriou: Kópia súboru SMVE_final_exam_doc

Základné príkazy

```
setwd("")
library(readxl)
library(latexpdf)
library(latex2exp)
```

Prvotná štatistická analýza, popisná a grafická (cv 3)

```
library(psych)
library(moments)
library(vioplot)
library(car) # qqPlot

#' ## Popisna statisticka analyza
#' Pocet zaznamov
length(x)

#' Ukazka dat
head(x)

#' Tabulka pocetnosti
table(x)
#' Relativne pocetnosti - prop.table(x)
table(x) / length(x)

#' ### Miery polohy
#' Min, Q1, median, mean, Q3, max
summary(x)


#' Modus
modus <- function(x) {
  uniqx <- unique(x)
  uniqx[which.max(tabulate(match(x, uniqx)))]
}
modus(x)


#' ### Miery variability
#' Rozptyl a smerodajna odchylka
var(x)
sd(x)
```

```r
#' ### Medzikvartilove rozpatie
#' Horny a dolny kvartil
quantile(x, probs = 0.75)
quantile(x, probs = 0.25)
IQR(x)

#' Variacny koeficient
sd(x)/mean(x)*100


#' ### Miery asymetrie
#' Sikmost a spicatost
skewness(x)
kurtosis(x)


#' ### Analyza outlierov
Q1<-quantile(x, probs=0.25)   # dolny kvartil
Q3<-quantile(x, probs=0.75)  # horny kvartil
IQR_ <- IQR(x)

k1<-1.5 # pre vybocujuce hodnoty
k2<-3 # pre extremalne hodnoty
x[x<(Q1-k1*IQR_)] # pre najdenie vybocujucich hodnot podla Q1
x[x>(Q3+k1*IQR_)] # pre najdenie vybocujucich hodnot podla Q3


#' ## Graficka statisticka analyza
plot(x)
barplot(table(x))
hist(x)
dotchart(table(x))
pie(table(x))
boxplot(x)
vioplot(x)
qqPlot(x)
```

Intervaly spoľahlivosti (cv 4)

Odhad strednej hodnoty z N - známa smerodajná odchýlka

```
#' Obojstranny odhad strednej hodnoty pri znamej s. odchylke
alpha <-
mean(x) + c(-1, 1)*qnorm(1 - alpha/ 2) * sd/sqrt(length(x))

#' Jednostranny lavy (dolny) odhad strednej hodnoty pri znamej s. odchylke
alpha <-
c(mean(x)-qnorm(1-alpha) * sd/sqrt(length(x)), Inf)

#' Jednostranny pravy (horny) odhad strednej hodnoty pri znamej s.
odchylke
alpha <-
c(-Inf, mean(x)+qnorm(1-alpha) * sd/sqrt(length(x)))
```

Odhad strednej hodnoty z N - neznáma smerodajná odchýlka

```
#' Obojstranny odhad strednej hodnoty pri neznamej s. odchylke
alpha <-
t.test(x, conf.level = 1 - alpha)$conf.int

#' Jednostranny lavy (dolny) odhad rozptylu strednej hodnoty pri neznamej
s. odchylke
alpha <-
t.test(x, conf.level = 1 - alpha, alternative = "g")$conf.int

#' Jednostranny pravy (horny) odhad strednej hodnoty pri neznamej s.
odchylke
alpha <-
t.test(x, conf.level = 1 - alpha, alternative = "l")$conf.int
```

Odhad rozptylu

```
#' Obojstranny odhad rozptylu
library(EnvStats)
alpha <-
varTest(x, conf.level = 1 - alpha)$conf.int

#' Jednostranny lavy (dolny) odhad rozptylu
library(EnvStats)
alpha <-
varTest(x, conf.level =1 - alpha, alternative = "g")$conf.int

#' Jednostranny pravy(horny) odhad rozptylu
```

```
library(EnvStats)
alpha <-
varTest(x, conf.level = 1 - alpha, alternative = "l")$conf.int
```

Odhad mediánu

```
wilcox.test(x, conf.int=TRUE)
```

Odhad proporcie

- m- počet pozorovaní s danou vlastnosťou
- n- počet všetkých pozorovaní

```
m <-
n <-
alpha <-
prop.test(m, n=n, conf.level = 1 - alpha)$conf.int
```

Testy, či dáta sú z normálneho rozdelenia

```
#' Test normality dat
#' $H_0$: data su z normalneho rozdelenia $\qquad  H_1$: data nie su z
normalneho rozdelenia

shapiro.test(x)$p.value

library(nortest)
lillie.test(x)$p.value


#' Test normality dat s danymi parametrami
ks.test(x,"pnorm",mu0,sd)$p.value
```

Testy pre parametre normálneho rozdelenia (cv 5, 6)

Jednovýberové testy (cv 5)

Test o strednej hodnote pri známom rozptyle

```
#' Test pre strednu hodnotu pri znamom rozptyle
library(DescTools)
#' $$H_0: \mu = \mu_0 \qquad  H_1: \mu \neq \mu_0$$
mu0 <-
sd0 <-
alpha <-
ZTest(x, mu = mu0, sd_pop = sd0, conf.level = 1 - alpha)$p.value

#' Test pre strednu hodnotu pri znamom rozptyle - alternativa G
library(DescTools)
#' $$H_0: \mu = (\leq) \mu_0 \qquad  H_1: \mu > \mu_0$$
mu0 <-
sd0 <-
alpha <-
ZTest(x, mu = mu0, sd_pop = sd0, conf.level = 1 - alpha, alternative =
"g")$p.value

#' Test pre strednu hodnotu pri znamom rozptyle - alternativa L
library(DescTools)
#' $$H_0: \mu = (\geq) \mu_0 \qquad  H_1: \mu < \mu_0$$
mu0 <-
sd0 <-
alpha <-
ZTest(x, mu = mu0, sd_pop = sd0, conf.level = 1 - alpha, alternative =
"l")$p.value
```

Test o strednej hodnote pri neznámom rozptyle

```
#' Test pre strednu hodnotu pri neznamom rozptyle
#' $$H_0: \mu = \mu_0 \qquad  H_1: \mu \neq \mu_0$$
mu0 <-
alpha <-
t.test(x, mu = mu0, conf.level = 1 - alpha)$p.value

#' Test pre strednu hodnotu pri neznamom rozptyle - alternativa G
#' $$H_0: \mu = (\leq) \mu_0 \qquad  H_1: \mu > \mu_0$$
mu0 <-
alpha <-
t.test(x, mu = mu0, conf.level = 1 - alpha, alternative = "g")$p.value
```

```
#' Test pre strednu hodnotu pri neznamom rozptyle - alternativa L
#' $$H_0: \mu = (\geq) \mu_0 \qquad H_1: \mu < \mu_0$$
mu0 <-
alpha <-
t.test(x, mu = mu0, conf.level = 1 - alpha, alternative = "l")$p.value
```

Test pre rozptyl

```
#' Test pre rozptyl
library(EnvStats)
#' $$H_0: \sigma^2 =\ sigma_0^2 \qquad \sigma^2: \neq \sigma_0^2$$
sigma0 <-
alpha <-
varTest(x, sigma.squared = sigma0^2, conf.level = 1 - alpha)$p.value

#' Test pre rozptyl - alternativa G
library(EnvStats)
#' $$H_0: \sigma^2 = (\leq) \sigma_0^2 \qquad  H_1: \sigma^2 (>
\sigma_0^2$$
sigma0 <-
alpha <-
varTest(x, sigma.squared = sigma0^2, conf.level = 1 - alpha, alternative =
"g")$p.value

#' Test pre rozptyl - alternativa L
library(EnvStats)
#' $$H_0: \sigma^2u = (\geq) \sigma_0^2 \qquad  H_1: \sigma^2 <
\sigma_0^2$$
sigma0 <-
alpha <-
varTest(x, sigma.squared = sigma0^2, conf.level = 1 - alpha, alternative =
"l")$p.value
```

Dvojvýberové testy (cv 6)

Párový t-test

```
#' Test pre vyhodnotenie strednych hodnot
#' $$H_0: \mu = \mu_0 \qquad  H_1: \mu \neq \mu_0$$
mu0 <-
alpha <-
t.test(x1, x2, paired = T, conf.level = 1 - alpha)$p.value

#' Test pre vyhodnotenie strednych hodnot - alternativa G
#' $$H_0: \mu = (\leq) \mu_0 \qquad  H_1: \mu \neq (>) \mu_0$$
mu0 <-
alpha <-
t.test(x1, x2, paired = T, conf.level = 1 - alpha, alternative =
"g")$p.value

#' Test pre vyhodnotenie strednych hodnot - alternativa L
#' $$H_0: \mu = (\geq) \mu_0 \qquad  H_1: \mu \neq (<) \mu_0$$
mu0 <-
alpha <-
t.test(x1, x2, paired = T, conf.level = 1 - alpha, alternative =
"l")$p.value
```

Dvojvýberový test pre disperziu (F test)

```
#' Overenie rovnosti rozptylov
#' $$H_0: \sigma_1 = \sigma_2 \qquad  H_1: \sigma_1 \neq \sigma_2$$
alpha <-
var.test(x0, x1, alternative = "two.sided", conf.level = 1 -
alpha)$p.value
```

Dvojvýberový t-test

```
#' Vyhodnotenie rovnosti strednych hodnot
#' $$H_0: \mu_1 = \mu_2 \qquad  H_1: \mu_1 \neq \mu_2$$
alpha <-
t.test(x1, x2, alternative = "two.sided", var.equal =T, conf.level = 1 -
alpha)$p.value

#' Vyhodnotenie strednych hodnot - alternativa G
#' $$H_0: \mu_1 = (\leq) \mu_2 \qquad  H_1: \mu_1 \neq (>) \mu_2$$
alpha <-
t.test(x1, x2, alternative = "g", var.equal =T, conf.level = 1 -
alpha)$p.value

#' Vyhodnotenie strednych hodnot - alternativa L
```

```
#' $$H_0: \mu_1 = (\geq) \mu_2 \qquad H_1: \mu_1 \neq (<) \mu_2$$
alpha <-
t.test(x1, x2, alternative = "l", var.equal =T, conf.level = 1 -
alpha)$p.value
```

Testy dobrej zhody (cv 7)

Pearsonov chi kvadrat dobrej zhody

```
#' Overenie zhody s diskretnym a spojitym rozdelenim,
#' parametre poznam alebo dohadnem. Sluzi aj na test nezavislosti znakov
#' v kontingencnej tabulke.

tab <- table(x)
test <- chisq.test(x, p=prob)
test
```

Kolmogorov Smirnov test dobrej zhody

```
#' Overujeme zhodu s teoretickym spojitym rozdelenim
#' pre spojite radsej tento ako chi^2, test vyzaduje znalost parametrov.
test <- ks.test(data, "pnorm", mean=20.1, sd=0.9); test
```

Dvojvyberovy KS test

```
ks2 <- ks.test(x1,x2); ks2

# Vizualizacia
library(psych)
data <- data.frame(A,B)
multi.hist(data, breaks = 5, dcol=c('green', 'blue'), bcol='green')
```

Testovanie extremalnych hodnot - vyzaduje N rozdelenie!

Grubbsov test

```
#' $H_0: min(X) (max(X)) nie je outlier \qquad H_1: min(X) (max(X)) je
outlier$
library(outliers)
# Test pre maximalnu hodnotu
grubbs.test(x)
# Test pre minimalnu hodnotu
grubbs.test(x, opposite = T)
```

Dixonov test

```
#' $H_0: min(X) (max(X)) nie je outlier \qquad H_1: min(X) (max(X)) je
outlier$
library(outliers)
# Test pre maximalnu hodnotu
dixon.test(x)
```

```
# Test pre minimalnu hodnotu
dixon.test(x, opposite = T)
```

Testovanie extermalnych hodnot - nie N rozdelenie

```
Q1<-quantile(x, probs=0.25)   # dolny kvartil
Q3<-quantile(x, probs=0.75)  # horny kvartil
IQR_ <- IQR(x)
k1<-1.5 # pre vybocujuce hodnoty
k2<-3 # pre extremalne hodnoty
x[x<(Q1-k1*IQR_)] # pre najdenie vybocujucich hodnot podla Q1
x[x>(Q3+k1*IQR_)] # pre najdenie vybocujucich hodnot podla Q3
x[x<(Q1-k2*IQR_)] # pre najdenie extremalnych hodnot podla Q1
x[x>(Q3+k2*IQR_)] # pre najdenie extremalnych hodnot podla Q3

# ziskanie len outlierov (vsetkych, sortnutych)
d <- x
sort(c(d[which(d < (Q1 - k1 * IQR_))], d[which(d > (Q3 + k1 * IQR_))]))
```

Testy náhodnosti (cv 8)

Testy nahodnosti Wald-Wolfovitz test (test serii) - spojite rozdelenie

```
#' Testovanie nahodnosti
#' $H_0$: vyber je nahodny oproti alternativnej hypoteze \qquad $H_1$: vyber nie je nahodny
library(randtests)
runs.test(x, plot=T)
```

Test kritickych bodov - Turning point test - citlivy na periodicitu

```
#' Testovanie nahodnosti
#' $H0: vyber je nahodny oproti alternativnej hypoteze \qquad H1: vyber nie je nahodny$
library(randtests)
turning.point.test(x)
```

Neparametrické testy (cv 8)

Znamienkovy test (parametricky -Test pre strednu hodnotu pri neznamom rozptyle) (t.test)

```
#' $$H_0: md = md_0 \qquad  H_1: md \neq md_0$$
library(BSDA)
md0 <-
alpha <-
SIGN.test(x, md=md0)

#' Parovy znamienkovy test
SIGN.test(pred, po, conf.level= 1-alpha)
```

Jednovyberovy Wilcoxonov test

```
#' Overenie symetrie
library(moments)
skewness(x)
#' H_0 = data nie su zosikmene, H_1 = data su zosikmene
agostino.test(x)

#' $H_0$: median = m0 \qquad $H_1$: median \neq m0
mu0 <-
alpha <-
wilcox.test(x, mu=mu0, conf.level= 1-alpha)

#' Parovy Wilcox test
wilcox.test(x1, x2, paired = T)
```

Dvojvyberovy Wilcoxonov test/ Mann-whitney test (dvojvyberovy t.test)

```
#' Vizualizacia
boxplot(x1, x2)

#' Testovanie rovnosti rozptylov
fligner.test(x1 ~ x2, data = df)

alebo

library(lawstat)
levene.test(x1 ~ x2, data = df)

#' Ak rozdelenia su vyrazne nepodobne, maju rozne disperzie, tak sa
pouzije ks test.
```

```
#' Testovanie
#' $H_0: F_x = F_y \qquad H_1: F_x \neq F_y$
wilcox.test(x, y)

alebo

wilcox.test(x1 ~ x2, data = df)
```

Kruskal Wallisov test (neparametricka ANOVA)

testujeme rovnost medianov viac ako 2 suborov ale hypoteza je v tvare, ze rozdelenia tych suborov sa rovnaju

```
#' Faktorizacia
data$x1 <- factor(data$x1)

#' Vizualizacia
boxplot(x1, x2)

#' Testovanie
#' $H_0: F_1= F_2= ... = F_I$ median k oproti H1: \neg $H_0$
kruskal.test(data$x1, data$x2)

#' Zistenie, ktore triedy sa lisia
library(dunn.test)
dunn.test(data$x1, data$x2)
dunn.test(data$x1, data$x2, altp = TRUE,  list = TRUE)
```

ANOVA (cv 6)

```
#' Faktorizacia
data$x1 <- factor(data$x1)

#' Overenie normality dat
#' $H_0: X_i$ -ty vyber je z normalneho rozdelenia, i=1,2,...,k, kde k-pocet urovni faktora $\qquad  H_1: \exists X_i$, ktory nepochadza z normalneho rozdelenia
tapply(x1, x2, shapiro.test)$p.value

#' Testovanie rovnosti disperzii
#' $H_0: \sigma_1 = \sigma_2 = ... = \sigma_k \qquad  H_1: \exists i,j \sigma_i \neq \sigma_j$
bartlett.test(x1, x2)$p.value

#' Vizualizacia
library(RColorBrewer)
library(vioplot)
library(ggplot2)
library(ggpubr)
boxplot(x$mprij ~ x$vzdelanie, col = brewer.pal(3, "Blues"), xlab = "Vzdelanie", ylab = "Prijem", main = "Vzdelanie vs Prijem")

vioplot(x$mprij ~ x$vzdelanie, col = brewer.pal(3, "Blues"), xlab = "Vzdelanie", ylab = "Prijem", main = "Vzdelanie vs Prijem")

(df <- data.frame(x$vzdelanie, x$prijem))
ggline(df, x="Vzdelanie", y="Prijem", add=c("mean_se", "jitter", "violin"), col="steelblue", main="Prijem podla vzdelania")

#' Anova
#' $$H_0:: \mu_1 = \mu_2 = \mu_3 (faktor nema vplyv) \qquad H_1: \exists i,j: \mu_i \neq \mu_j (faktor ma vplyv)$$
anova <- aov(data$f1~data$f2);anova
summary(anova)

#' Anova s interakciami
anova < -aov(f1 ~ f2+ f1+ f2*f2, x)
summary(anova)

# Ktore triedy sa odlisuju (triedy su rovnake ak p > alpha)
tps <- TukeyHSD(anova);tps
plot(tps)

#' alebo
library(DescTools)
```

```
(sch <- ScheffeTest(anova))
plot(sch)
```

Korelačná analýza (cv 9)

```
#' ## Kovariancia a Pearsonov koeficient korelacie

#' Overenie normality dat
library(nortest)
shapiro.test(x)
shapiro.test(y)
```

alebo

```
library(mvnormtest)
mshapiro.test(t(matrix(c(x,y), ncol = 2)))

#' Vizualizacia
plot(x,y)

#' Kovariancia (ak je rozna od 0, existuje medzi premennymi linearny
vztah)
kov <- cov(x,y)
(varX <- sum( (x-mean(x))^2 ) / (n-1)) # vyberovy rozptyl X
(varY <- sum( (y-mean(y))^2 ) / (n-1)) # vyberovy rozptyl Y
(kor <- kov / sqrt(varX*varY)) # normovanie kovariancie, vyberovy
korelacny koeficient (kedze sa pocita z vyberu a nie pre celu populaciu)
```

alebo

```
#' Pearsonov koeficient korelácie
cor(x,y,use="everything",method="pearson")

#' ## Spearmanov a Kendallov koeficient korelacie

#' Vizualizacia
plot(x,y)
#' Spearmanov koeficient korelacie
cor(x,y,use="everything",method="spearman")

#' Kendallov koeficient korelacie
cor(x,y,use="everything",method="kendall")

#' ## Test nulovosti korelačného koeficienta
```

```
#' $H_0: ρ=0 (neexistuje zavislost) \qquad H_1: ρ \neq 0 (existuje zavislost)$
alpha <-
cor.test(x,y,alternative = "two.sided", method="pearson", conf.level = 1 - alpha)

#' $H_0: ρ<0 (nepriama zavislost) \qquad H_1: ρ>0 (priama zavislost)$
alpha <-
cor.test(x,y,alternative = "less", method="pearson", conf.level = 1 - alpha)

#' $H_0: ρ>0 (priama zavislost) \qquad H_1: ρ<0 (nepriama zavislost)$
alpha <-
cor.test(x,y,alternative = "greater", method="pearson", conf.level = 1 - alpha)

#' ## Korelacna matica (n - rozmerny pripad)
library(corrplot)
(cor<-cor(x, use="complete.obs", method="kendall"))
corrplot.mixed(cor,lower="number",upper="circle",tl.pos = "lt")

#' Vyznamnost korelacii (Vystupom su hladiny vyznamnosti)

library(Hmisc)
rcorr(as.matrix(mtcars), type = "spearman")

#' ## Zavislost medzi kvalitativnymi znakmi
#' $H_0: nezavislost \qquad H_1: zavislost$
prijem$Vek <- as.factor(prijem$Vek)
levels(prijem$Vek) <- c("18-25r", "26-35r", "36-45r", "46-55r", "56+")
prijem$Vzdelanie <- as.factor(prijem$Vzdelanie)
levels(prijem$Vzdelanie) <- c("ZŠ & SŠ", "SŠ+M", "VŠ1", "VŠ2", "VŠ3")
tab <- table(prijem$Vzdelanie, prijem$Vek); tab
chisq.test(tab)
```

Regresná analýza (cv 10, 11)

```
#’ Overenie korelácie
cor(x,y)

#’ Graf závislosti pre 2 premenne
plot(x,y, type='b', xlab='Nazov_x', ylab='Nazov_y')

#’ Vypocet koeficientov lineárnej funkcie
lm(y~x)

library(corrplot)
corrplot.mixed(cors)

#’ parovy diagram
pairs(Diab, col=Diab$diabetes)
```

Linearna regresia

```
model <- lm(y ~ x)
summary(model)
plot(model)

(linAIC<-AIC(model))
(linBIC<-BIC(model))

#’ model s viacerymi parametrami
model <- lm(sales ~ youtube + facebook + newspaper, data = marketing)
summary(model)
plot(model)
```

Logisticka regresia

```
#’ model GLM
model<- glm(diabetes ~., data = Diab, family = binomial)
summary(model)

#’atributy s nizkou p-hodnotou su pre nas dobre a chceme ich v modeli, su
#’ oznaceme hviezdickami

probs <- predict(model, type="response")
#’ rozdelime podla hranicnej hodnoty
pred <- ifelse((probs>0.5), 1, 0)

table(pred, Diab$Outcome) # confusion matrix
```

Zhluková analýza cv(12)

- je prieskumná metóda, nie štatistický test

library(factoextra)

Predpríprava:

- Odľahlé pozorovania je treba dopredu vylúčiť
- Dáta s chýbajúcimi hodnotami vylúčiť alebo chýbajúce dáta nahradiť vhodnou metódou
  data <- na.omit(data)

- Premenné je niekedy treba štandarizovať, aby sa odstránil vplyv merných jednotiek
- Výsledky môžu byť tiež ovplyvnené závislosťou medzi sledovanými znakmi (eliminujeme metódou hlavných komponentov, Principal Component Analysis - PCA)

Metódy:

- hierarchické - default method complete
  library(ecodist) # ??
  library(magrittr) # ??
  library(dendextend) # ??

  clustering <- hclust(matica_vzdialenosti)
  plot(clustering)
  - Najbližšieho suseda (simple linkage) - najkratšia vzdialenosť ľubovoľného bodu (objektu) v zhluku voči ľubovoľnému bodu v zhluku inom,
  - Najvzdialenejšieho suseda (complete linkage)- vzdialenosť medzi zhlukmi je určená ako vzdialenosť medzi dvomi najvzdialenejšími objektmi v zhlukoch. Jej výhodou je, že vytvára menej početné, dobre odlíšiteľné zhluky
    hclust(matica_vzdialenosti, method = "complete")
  - Priemernej vzdialenosti, priemernej väzby (between groups linkage) – vzdialenosť medzi zhlukmi je určená ako priemerná vzdialenosť medzi všetkými objektami v dvoch zhlukoch
    hclust(matica_vzdialenosti, method = "average")
  - Wardova - je odlišná od ostatných, princípom nie je optimalizácia vzdialenosti ale minimalizácia heterogenity zhluku, nájde sa minimálny rast súčtu štvorcov odchýliek od priemeru zhluku pridaním nového prvku
  - Centroidná - vzdialenosť dvoch zhlukov je určená ako štvorcová euklidovská vzdialenosť ich centroidov
    hclust(matica_vzdialenosti, method = "centroid")
  - Mediánová - vzdialenosť dvoch zhlukov je určená ako štvorcová euklidovská vzdialenosť ich mediánov
    hclust(matica_vzdialenosti, method = "median")

        - aglomeratívne - považuje na začiatku každý prvok za zhluk
        - divízne - na začiatku 1 zhluk, postupne zhluky s 1 objektom
    - nehierarchické - na začiatku známy počet zhlukov
        - hľadanie optimálneho počtu zhlukov
            - skok z klesania na rast na grafe
              fviz_nbclust(data, kmeans, method = "wss")
            - cez štatistiku medzery
              gap_stat <- clusGap(data, FUN = kmeans,
              nstart = 25, K.max = 10, B = 50)
              fviz_gap_stat(gap_stat)

        - K-priemer - utvoríme bod, ktorý reprezentuje priemer hodnôt ukazovateľov v zhlukoch, určíme počiatočný počet k priemerov (k je počet výsledných zhlukov) a objekty sa do zhlukov priraďujú podľa najmenších vzdialeností od tohto reprezentanta, v novovzniknutom zhluku sa priemery prepočítajú
          ckms <- kmeans(data, centers=3)
          library(factoextra)
          fviz_cluster(ckms, data=data)

        - K-medoid (Partitioning Arounds Medoids PAM) - lepšie funguje pri odľahlých pozorovaniach
          kmed <- pam(data, k = 4)
          fviz_cluster(kmed, data = data)

Miery podobnosti:

- miery vzdialenosti:
    - Euklidovská (default)
      matica_vzdialenosti <- dist(A) # dist(A, method = "euclidean")
    - štvorcová Euklidovská (keď je potrebné dať progresívnejšiu váhu vzdialenejším údajom),
    - Manhatanova - Absolute distance between the two vectors,
    - Čebyševova vzdialenosť
      matica_vzdialenosti <- dist(A, method="maximum")
- asociačné koeficienty - Jaccardov koeficient (podobnosti), Jaccardova miera vzdialenosti (nepodobnosti), Yuleova miera vzdialenosti (nepodobnosti,Yule dissimilarity), Russel Rao miera vzdialenosti
- korelačné koeficienty
- pravdepodobnostné miery podobnosti - Hammingova vzdialenosť,

Validácia - informácie o „dobrom“ a „zlom“ zhluku, resp. určenie počtu optimálnych zhlukov:

- metóda siluety
    - Siluetu vyčíslime pre každý objekt, pri rôznej definícii nepodobnosti je $siW \in [-1,1]$

        - ak hodnota siW je blízka 1, objekt Oi je dobre klasifikovaný v zhluku A, jeho vzdialenosť k objektom zhluku A je podstatne kratšia ako k objektom ostatných zhlukov
        - ak je siW blízka 0, objekt Oi možno zaradiť aj do iného zhluku
        - ak je siW blízka −1, objekt Oi je zle klasifikovaný v zhluku A
    - Prehľadnou štatistikou je aj priemerná silueta sW , počítaná cez všetky objekty pri danom zhlukovaní W, správny počet zhlukov je ten, pre ktorý je priemerná silueta sW maximálna,
- Dunnov validačný index
    - založený na predpoklade, že zhluky sú kompaktné a dobre oddelené
    - Nadobúda hodnotu od nula po nekonečno, vysoké hodnoty indikujú optimálny počet zhlukov.

- treba sa rozhodnúť, ktorá metóda dáva najlepšie výsledky. použijeme kofonetický koeficient korelácie. je to miera zhody matice vzdialeností a konkrétnou zhlukovacou metódou. ako sme pri koreláciách zvyknutí, hľadáme silnú koreláciu, teda vyberieme metódu, kde kof.kor. je najväčšia. existujú samozrejme aj iné kritériá na výber najvhodnejšej zhlukovacej metódy. v praxi je často ale najvhodnejšia metóda priemernej väzby (average) a Wardova metóda (ward), nevytvárajú sa pri nich zacyklené zhluky a wardova metóda predchádza vytváraniu jednoprvkových, resp. príliš malých zhlukov (keď už sa vytvoria menšie zhluky, sú významne odlišné od iných a metóda ich preto nepripojí k žiadnemu zhluku). Wardova pracuje na báze ANOVY, teda minimalizuje rozptyl. (ale nie je to pravidlo, takže treba overiť, ktorá metóda je najvhodnejšia pre konkrétny problém).

    ```
    cophcomp <-cophenetic(ccomp) # kofonetická vzdialenosť
    cophward <- cophenetic(cward)
    cophsg <- cophenetic(csingle)
    cophavg<- cophenetic(cavg)

    c(cor(d,cophcomp), cor(d,cophward), cor(d,cophsg),
      cor(d,cophavg))
    ```

- treba vyhodnotiť optimálny počet zhlukov

    ```
    k<-1.25 # konštantna odvodená simulačne
    (mean<-mean(cavg$height)) # priemer zhlukovacích vzdialeností
    (sd<-sd(cavg$height)) # smerod. odchýlka zhluk. vzdialeností
    alfa<-mean+(k*sd) # dendrogram= grafický výstup ZA, pretneme na úrovni vzdialenosti alfa a určíme tak výsledný počet zhlukov
    plot(cavg)
    abline(h=alfa, col="red")
    ```

- iné vykreslenie keď už poznám počet zhlukov

    ```
    library(dplyr)
    dend <- data %>% dist("euclidean") %>% hclust("average") %>% as.dendrogram
    ```

dend
%>%set("branches_lwd",2)%>%color_branches(k=5)%>%color_labels(k=5)%>%set("labels_cex", 1) %>%plot(horiz=F, main = "Dendrogram- average method")
dend %>% rect.dendrogram(k=5,border=4,lty=5,horiz=F, lwd=1)
abline(h = alfa, lwd = 1, lty = 2, col = "red")

- Heatmap - popísanie vzťahu v zhlukoch - Čím tmavší odtieň v heatmape v danom zhluku, tým viac podobné sú si objekty vzhľadom na danú premennú a tiež tým lepšie prispieva premenná k zhlukovaniu
heatmap(data, Rowv = as.dendrogram(cavg),Colv=NA)

Priprava 2020

```
mean(data$"ročný príjem")

modus <- function(x) {
  uniqx <- unique(x)
  uniqx[which.max(tabulate(match(x, uniqx)))]
}

modus(data$"ročný príjem")

var(data$"ročný príjem")

sd(data$"ročný príjem")

quantile(data$"ročný príjem", probs = 0.75)
quantile(data$"ročný príjem", probs = 0.25)

#' Sikmost a spicatost
skewness(data$"ročný príjem")
kurtosis(data$"ročný príjem")

hist(data$"ročný príjem")

boxplot(data$"ročný príjem", horizontal = TRUE)

vioplot(data$"ročný príjem")

#' box a violint grafy podla postu clenov

data$skupina <- ifelse(data$"počet členov domácnosti" <= 2, "do_2",
"3_viac")
```

```
vioplot(data$"ročný príjem" ~ data$skupina)

boxplot(data$"ročný príjem" ~ data$skupina)


#' rozdelime do dvoch subsetov
do_2 <- subset(data, data$"počet členov domácnosti" < 2)
nad_3 <- subset(data, data$"počet členov domácnosti" > 3)


#' pre skupinu do 2
mean(do_2$"ročný príjem")

modus(do_2$"ročný príjem")

var(do_2$"ročný príjem")

sd(do_2$"ročný príjem")

hist(do_2$"ročný príjem", main="Rocny prijem do 2 clenov")

#' pre skupinu nad 3

mean(nad_3$"ročný príjem")

modus(nad_3$"ročný príjem")

var(nad_3$"ročný príjem")

sd(nad_3$"ročný príjem")

hist(nad_3$"ročný príjem", main="Rocny prijem nad 3 clenov")

#' # Uloha 2
#'

shapiro.test(do_2$"ročný príjem")
#' p-hodnota > 0.05 -> data su normalne rozdelene

t.test(do_2$"ročný príjem", conf.level = 0.95)$conf.int

shapiro.test(nad_3$"ročný príjem")
#' p-hodnota < 0.05 -> data nie su normalne rozdelene

quantile(nad_3$"ročný príjem", probs = 0.025)
quantile(nad_3$"ročný príjem", probs = 0.975)
```

```
#' # Uloha 3

pracujuci <- subset(data, data$zamestnanie != 1)
pracujuci_do_2 <- subset(pracujuci, pracujuci$"počet členov domácnosti" <
2)
pracujuci_nad_3 <- subset(pracujuci, pracujuci$"počet členov domácnosti" >
3)

x1 <- nad_3$"ročný príjem"
x2 <- do_2$"ročný príjem"

shapiro.test(x1)
shapiro.test(x2)

df1 <- data.frame("prijem"=c(x1,x2), "skupina"=rep( c("do2","nad3"),
times=c(length(x1),length(x2)) ) )

library(lawstat)
levene.test(df1$prijem, df1$skupina)
# rovnake disperzie mozem pouzit wilcox


#' $$H_0: \mu_1 = \mu_2 \qquad  H_1: \mu_1 \neq \mu_2$$
wilcox.test(x1, x2)
# p-hodnota < 0.05, zamietam zhodnot strednej hodnoty

#' # Uloha 4

kruh <- c(0.995,  0.828,      0.947,      1.161,      0.996,      1.053,
      0.889,      1.056,      1.074,      0.961)
stvorec <- c(1.208,     1.06, 0.924,      1.057,      1.007,      1.047,
      1.153,      1.02)
obdlznik <- c(1.131,    1.161,      1.066,      1.116,      1.056,
1.057,      1.201)

data_dosky <- data.frame(vychylka = c(kruh, stvorec, obdlznik), typ =
rep(c("kruh", "stvorec", "obdlznik"), c(length(kruh), length(stvorec),
length(obdlznik))))
data_dosky

# overime rovnost disperzii
bartlett.test(data_dosky$vychylka, data_dosky$typ)

typ <- factor(data_dosky$typ)
anova <- aov(data_dosky$vychylka~typ);anova
summary(anova)
```

```
#' p-hodnota < 0.05, zamietam $H_0$. Nie su rovnake.


tps <- TukeyHSD(anova);tps

#' Ktorý prierez stĺpa je najbezpečnejší, ak vieme, že experimentálne
#' vypočítaná hodnota musí byť väčšia ako hodnota normy?

#' # Uloha 5

kruskal.test(data_dosky$vychylka, typ)
#' p-hodnota < 0.05, zamietam $H_0$. Nie su rovnake.

library(dunn.test)
dunn.test(data_dosky$vychylka, typ)

#' # Uloha 6

bb <- c(12.189,   55.217,     32.741,     26.634,     22.326,     17.322,
      15.226,     20.058,     12.375,     21.096,     29.955,     17.322)
brehy <- c(2+8.592,     148.655,    67.999,     39.927,     35.96,
26.277,     22.281,     31.305,     18.833,     34.568,     48.604,
26.69)

plot(bb, brehy)

#' korelacny koeficient
cor(bb, brehy)
cor(bb, brehy, method="spearman")
cor(bb, brehy, method="kendall")
#' velmi vysoky 0.97

prietok <- data.frame(bb = bb, brehy=brehy)
model <- lm(brehy ~ bb, data=prietok)
summary(model)

pred <- predict(model, newdata = data.frame(bb = c(40)))
pred

#' tuto to nemam asi usortene a preto to tak zle kresli, ona raz robila
taky pekny graf
#' neviem to najst
plot(bb, brehy)
lines(bb, fitted(model, bb), col="red")
```

```
ggplot(prietok, aes(x = bb, y = brehy )) +
  geom_point() +
  stat_smooth(method = lm) +
  labs(x = "bb", y = "brehy", title = "Regresna priamka")
```