ADEI:

**1) Data analysis**

-> Visulaize : Boxplot(colName) or ll<-Boxplot(numCol~factCol) (ll is then outliers)

hist(colName,/freq=F,20,col=rainbow(20)/)

->Analize var: summary()

mean(*colName*)

sd(*colName*) -standart deviration, nummeric, na.rm = TRUE - ignores missing values

dnorm(*x*, *mean*, *sd*), gives density(plutnost) x-refered to previous lien, should be a plot (qnorm -func with normal distribution)

curve(dnorm(...),/add=T,col="red",lwd=2/) - Draws a curve corresponding to a function over the interval [from, to]. curve can plot also an expression in the variable xname

calQ - calculate quartiles and calculate upper and lower threshold (graniza, bounds for severe outliers)

condes(*colNames* ,which(colName==colnames(*colNames*))) ) - Description continuous (take only num in colNames) by quantitative variables and/or by categorical variables,

see correlation between var, if p.value< 0.05 then significant statistical association is found.

cor(*colNames*, y = NULL, method = c(/"pearson", /"spearman")) - pearson is only for normal distribution

kruskal.test(*numCol*~*factCol*) -> look at the p.value -> if we reject there is some average group different than the others reject if <00.05

with(df,pairwise.wilcox.test(*numCol*,*factCol*))

shapiro.test((*colName*)) - Shapiro-Wilk normality test, if p.value < 0.05 then null-hypothesis is rejected, so not a normal distribution.

-> Syntax stuff

setwd("D:/..")

names(df)[c(1:6,8:17,25)]

save(file="Traw.RData",/list=c("kpinet")/)

round(smt, dig=2)

sort(*colName*, decreasing= TRUE)

which(*colName* > *value*)

**2) PCA**

-> Visulaize: plot.PCA(*pca.res*,/ choix="ind",select="cos2 6"/)

barplot(*res.pca*$eig[,1], main="Eigenvalues", names.arg = paste("dim", 1:nrow(*res.pca*$eig)))

plot(*res.pca*$eig[,1], type = "l") # line chart

fviz\_eig(*res.pca*, addlabels = TRUE)

fviz\_eig(*res.pca*, choice = "eigenvalue",addlabels = TRUE)

->Analize: eigen(*corrMatrix* with cor()) - Computes eigenvalues and eigenvectors of numeric (double, integer, logical) or complex matrices

-> Syntax stuff: blob<-PCA(df[,ColNames],quali.sup=1,quanti.sup=8)

plot.PCA(blob,choix="ind",invisible="ind") - display only factors

pca.res$ind$contrib[, 1] contibution of ind to the 1st, 2nd.. PC

pca.res$ind$cos2[,2] - representation

sort(rbind(*res.pca*$ind$cos2[,1],res.pca$ind$cos2[,2],res.pca$ind$cos2[,3],*res.pc*

*a*$ind$cos2[,3]),decreasing=T)[1:3] -> filter

length <-length(which(*res.pca*$eig[,1]>=1));length

kaiser <- *res.pca*$eig[1:length,1] #keep only EV >=1 ->first 7

**3) Clustering hierarchical**

-> Visulaize:

->Analize: hcpc$desc.var - describe each cluster

hcpc$desc.ind$para - most central observations in a cluster(near to the center) (PARAGON)

$dist - the most distant from every class

-> Syntax stuff: HCPC(res.pca,nb.clust=3,order=TRUE)

**calcQ** <- function(x) {

s.x <- summary(x)

iqr<-s.x[5]-s.x[2]

list(souti=s.x[2]-3\*iqr, mouti=s.x[2]-1.5\*iqr, min=s.x[1], q1=s.x[2], q2=s.x[3], q3=s.x[5], max=s.x[6], mouts=s.x[5]+1.5\*iqr, souts=s.x[5]+3\*iqr ) }