Domača naloga 1

Neza Krzan, Tom Rupnik

Podatki

Uporabila bova podatke *Swiss banknotes data*, ki vsebujejo šest meritev, opravljenih na 100 pravih in 100 ponarejenih starih švicarskih bankovcih za 1000 frankov.

Podatki vsebujejo 7 spremenljivk - 6 številskih in eno opisno. Vsebujejo različne izmerjene dolžine in širine bankovca v milimetrih:

- `length`: dolžina bankovca(na sliki x_1),
- `left`: dolžina levega roba(na sliki x_2),
- `right`: dolžina desnega roba(na sliki x_3),
- `bottom`: dolžina spodnjega roba(na sliki x_4) in
- `top`: dolžina zornjega roba(na sliki x_5) ter
- `diag`: dolžina diagonale bankovca(na sliki x_6).

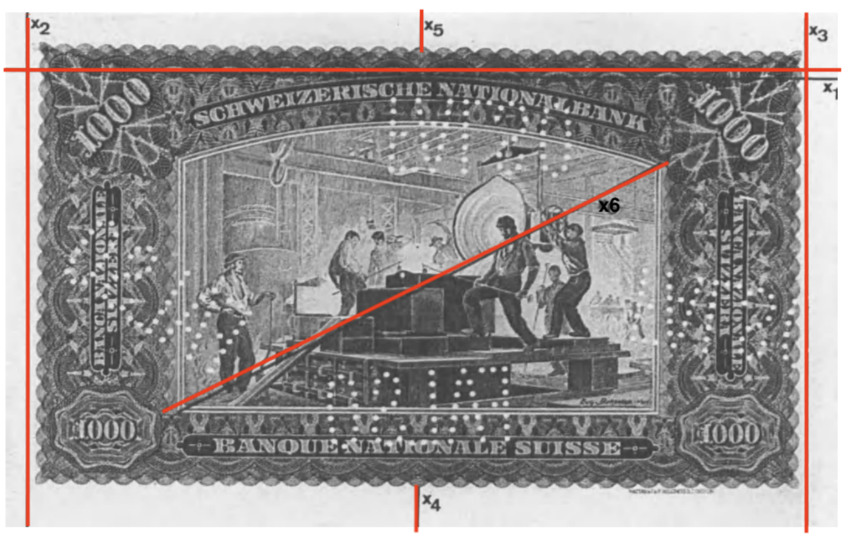


Figure 1: Označene mere na bankovcu.

Opisna spremenlivka `status` pa določa ali je bankovec pravi(`genuine`) ali ponarejen(`counterfeit`). V tabeli imamo torej meritve za 200 različnih bankovcev.

Urejanje podatkov

Imena spremenljivk in vrednosti kategorične spremenljivke sva preimenovala v slovenska imena ter, kot sva že napisala zgoraj, sva podatke skalirala.

Preimenovane spremenljivke:

- `length`: `dolžina`,

Table 1: Opisne statistike za števillske spremenljivke v podatkovnem okviru `Swiss banknotes data`.

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 50	Pctl. 75	Max
dolzina	200	215	0.4	214	215	215	215	216
levi.rob	200	130	0.4	129	130	130	130	131
desni.rob	200	130	0.4	129	130	130	130	131
spodnji.rob	200	9	1	7	8	9	11	13
zgornji.rob	200	11	0.8	8	10	11	11	12
diagonala	200	140	1	138	140	140	142	142

- `left`: `levi.rob`,
- `right`: `desni.rob`,
- `bottom`: `spodnji.rob`,
- `top`: `zgornji.rob`,
- `diag`: `diagonala` in
- `status` : `tip`, kjer je potem `counterfeit:ponarejen bankovec` in `genuine:pravi bankovec`.

Za lažjo predstavo si poglejmo opisne statistike številskih spremenljivk, da bomo vedeli s kakšnimi podatki imamo opravka.

Spremenljivke imajo različen razpon vrednosti, zato jih bova, skalirala; vidimo pa tudi, da nimamo mankajočih vrednosti v podatkih.

Poglejmo si še porazdelitve spremenljivk.

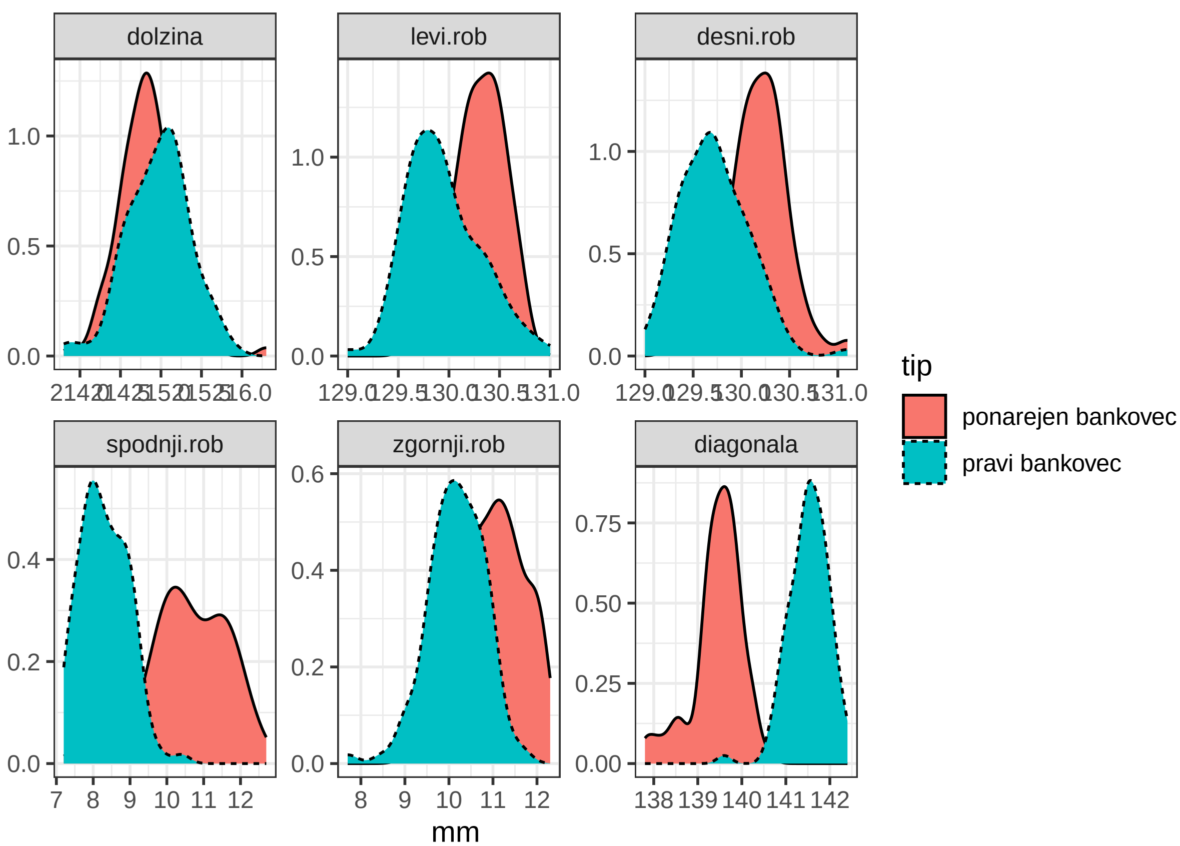


Figure 2: Porazdelitve spremenljivk v podatkovnem okviru `Swiss banknotes data`.

Opazna je razlika med pravimi bankovci in ponarejenimi pri vseh spremenljivkah.

Za razvrščanje bova uporabljala samo številske spremenljivke, in sicer `dolzina`, `levi.rob`, `desni.rob`, `spodnji.rob`, `zgornji.rob`; za analizo pa spremenljivki `tip` in `diagonala`. Ker je `diagonala` edina številska spremenljivka pri analizi, le ta ne bo skalirana.

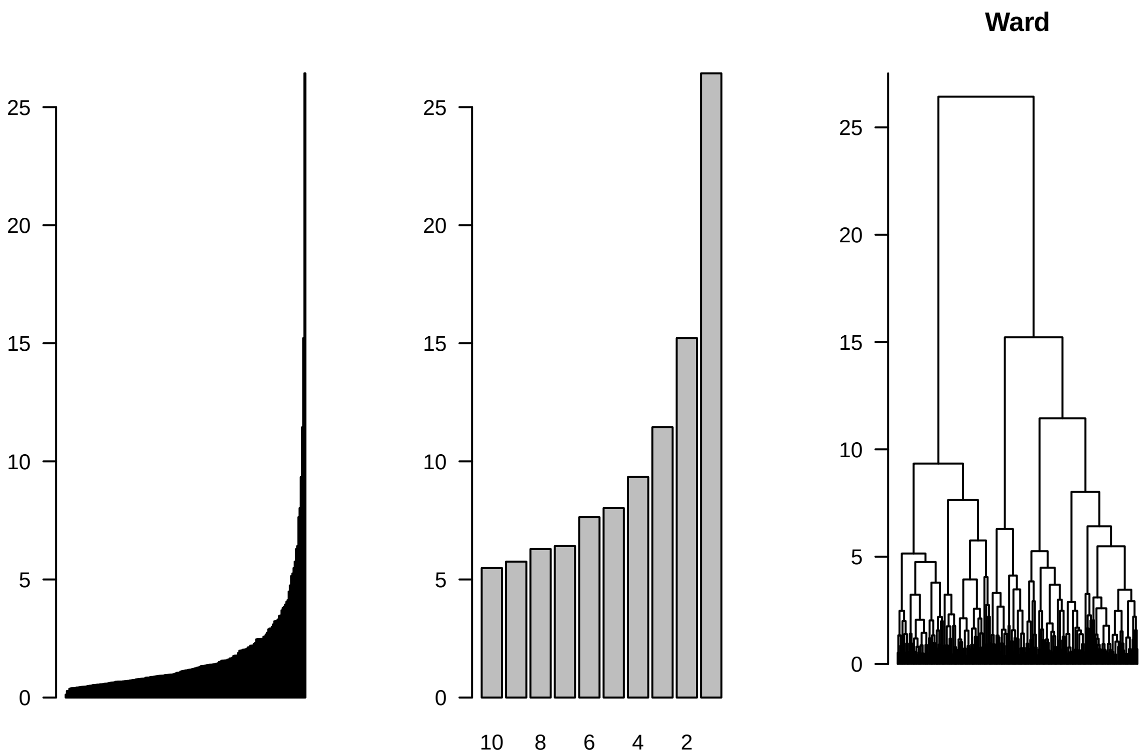
Hierarhično razvršanje

Torej za razvrščanje uporabljava spremenljivke `dolzina`, `levi.rob`, `desni.rob`, `spodnji.rob` in `zgornji.rob` ter primerjala bova tri različne metode in sicer, Wardovo metodo, minimalno metoda (single linkage) in maksimalno metoda (complete linkage).

Wardowa metoda

```
# matrika razliznosti na standardiziranih podatkih (Evklidska razdalja)
dz <- dist(x=dfz, method="euclidean")

hc.ward <- hclust(d=dz, method="ward.D2")
oldpar <- par(las=1, mfrow=c(1, 3))
barplot(hc.ward$height)
barplot(tail(x=hc.ward$height, n=10), names.arg=rev(seq_len(10)))
plot(hc.ward, labels=F, hang=-1, main="Ward", sub="", xlab="", ylab="")
```

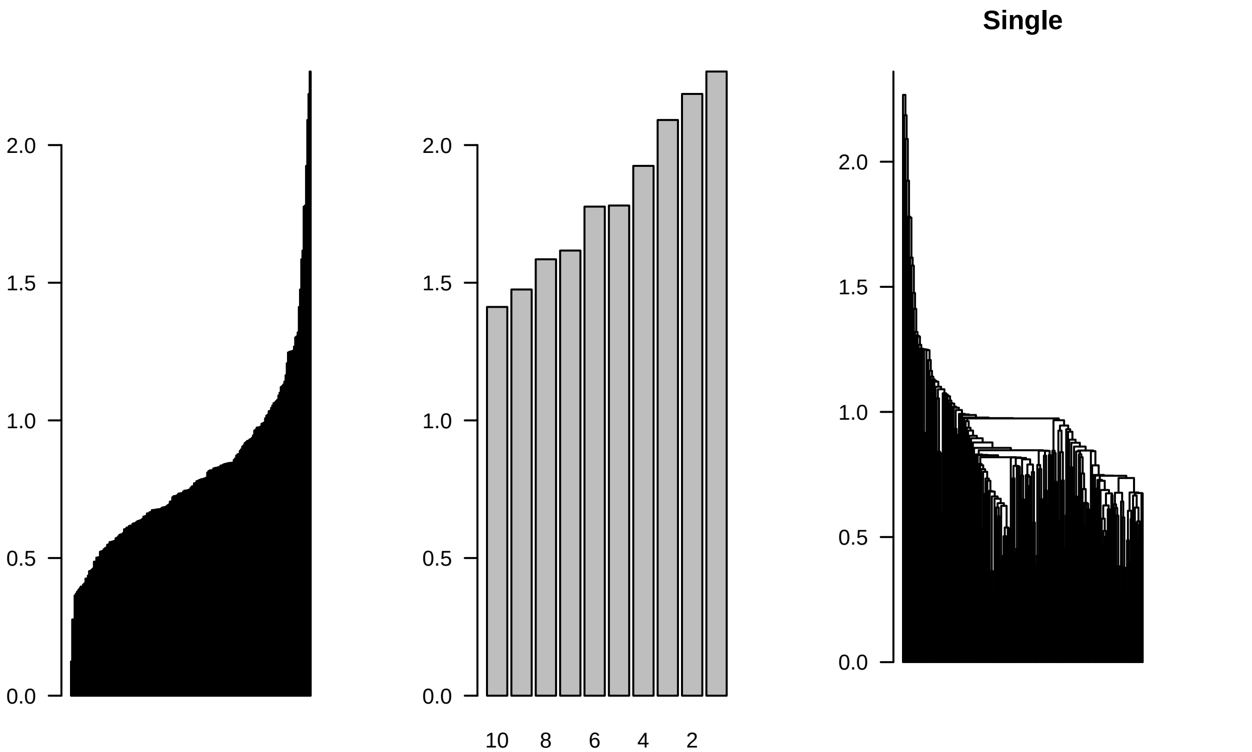



```
par(oldpar)
```

Minimalna metoda

```
# uporabili smo Evklidsko razdaljo
hc.sin <- hclust(d=dz, method="single")
```

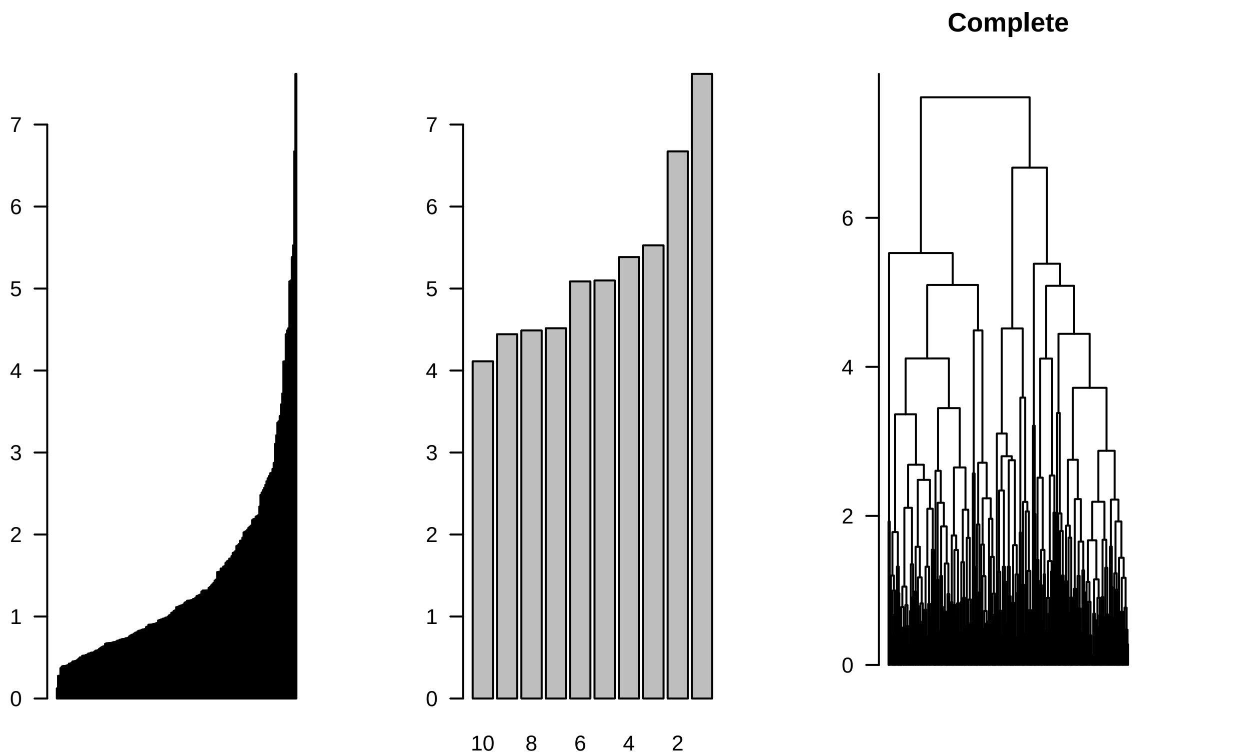
```r
oldpar <- par(las=1, mfrow=c(1, 3))
barplot(hc.sin$height)
barplot(tail(x=hc.sin$height, n=10), names.arg=rev(seq_len(10)))
plot(hc.sin, labels=F, hang=-1, main="Single", sub="", xlab="", ylab="")
```




```r
par(oldpar)
```

Maksimalna metoda

```r
# uporabili smo Evklidsko razdaljo
hc.com <- hclust(d=dz, method="complete")
oldpar <- par(las=1, mfrow=c(1, 3))
barplot(hc.com$height)
barplot(tail(x=hc.com$height, n=10), names.arg=rev(seq_len(10)))
plot(hc.com, labels=F, hang=-1, main="Complete", sub="", xlab="", ylab="")
```




```r
par(oldpar)
```

Analiza

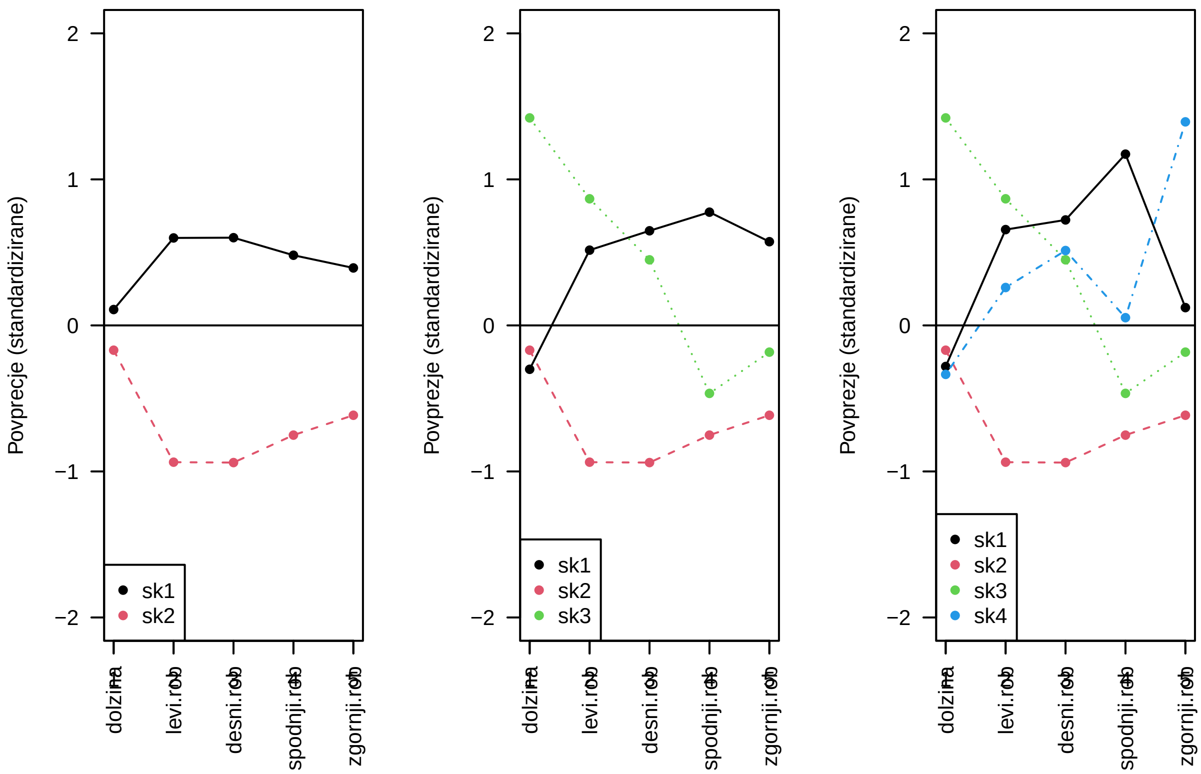
Glede na izgled grafov (razvrstitve) sva se odločila, da je najbolj primerna razvrstitev po Wardowi metodi. Pri ostalih dveh metodah so različnosti dokaj majhne (ni tako izrazitih različnosti (višin)). Grafe bomo narisali za *2*, *3* in *4* skupine, saj so tu razlike med različnostmi bolj izrazite.

```r
# Ward skupine
hc.ward1 <- cutree(tree=hc.ward, k=1)
hc.ward2 <- cutree(tree=hc.ward, k=2)
hc.ward3 <- cutree(tree=hc.ward, k=3)
hc.ward4 <- cutree(tree=hc.ward, k=4)

ime = c('dolzina','levi.rob','desni.rob', 'spodnji.rob', 'zgornji.rob')

oldpar <- par(las=1, mfrow=c(1, 3))
nsk <- 2
sk <- hc.ward2
sk.ime <- paste("sk", 1:nsk, sep="")
agr <- aggregate(x=dfz, by=list(sk), FUN=mean)
y <- t(agr[, -1])
matplot(x=seq_along(ime), y=y, type="o", pch=16, ylim=c(-2, 2), xlab="",
        ylab="Povprecje (standardizirane)")
axis(side=1, at=seq_along(ime), labels=ime, las=2)
legend("bottomleft", legend=sk.ime, col=1:nsk, pch=16)
abline(h=0, v=6.5)
nsk <- 3
sk <- hc.ward3
sk.ime <- paste("sk", 1:nsk, sep="")
```

```r
agr <- aggregate(x=dfz, by=list(sk), FUN=mean)
y <- t(agr[, -1])
matplot(x=seq_along(ime), y=y, type="o", pch=16, ylim=c(-2, 2), xlab="",
        ylab="Povprezje (standardizirane)")
axis(side=1, at=seq_along(ime), labels=ime, las=2)
legend("bottomleft", legend=sk.ime, col=1:nsk, pch=16)
abline(h=0, v=6.5)
nsk <- 4
sk <- hc.ward4
sk.ime <- paste("sk", 1:nsk, sep="")
agr <- aggregate(x=dfz, by=list(sk), FUN=mean)
y <- t(agr[, -1])
matplot(x=seq_along(ime), y=y, type="o", pch=16, ylim=c(-2, 2), xlab="",
        ylab="Povprezje (standardizirane)")
axis(side=1, at=seq_along(ime), labels=ime, las=2)
legend("bottomleft", legend=sk.ime, col=1:nsk, pch=16)
abline(h=0, v=6.5)
```




```r
par(oldpar)
```

Razvrščanje K-means

```r
# WSS
kmax <- 10
wss <- NULL
for (k in 1:kmax) {
  withinss <- kmeans(x=dfz, centers=k, nstart=100)$tot.withinss
  wss <- c(wss, withinss)
```

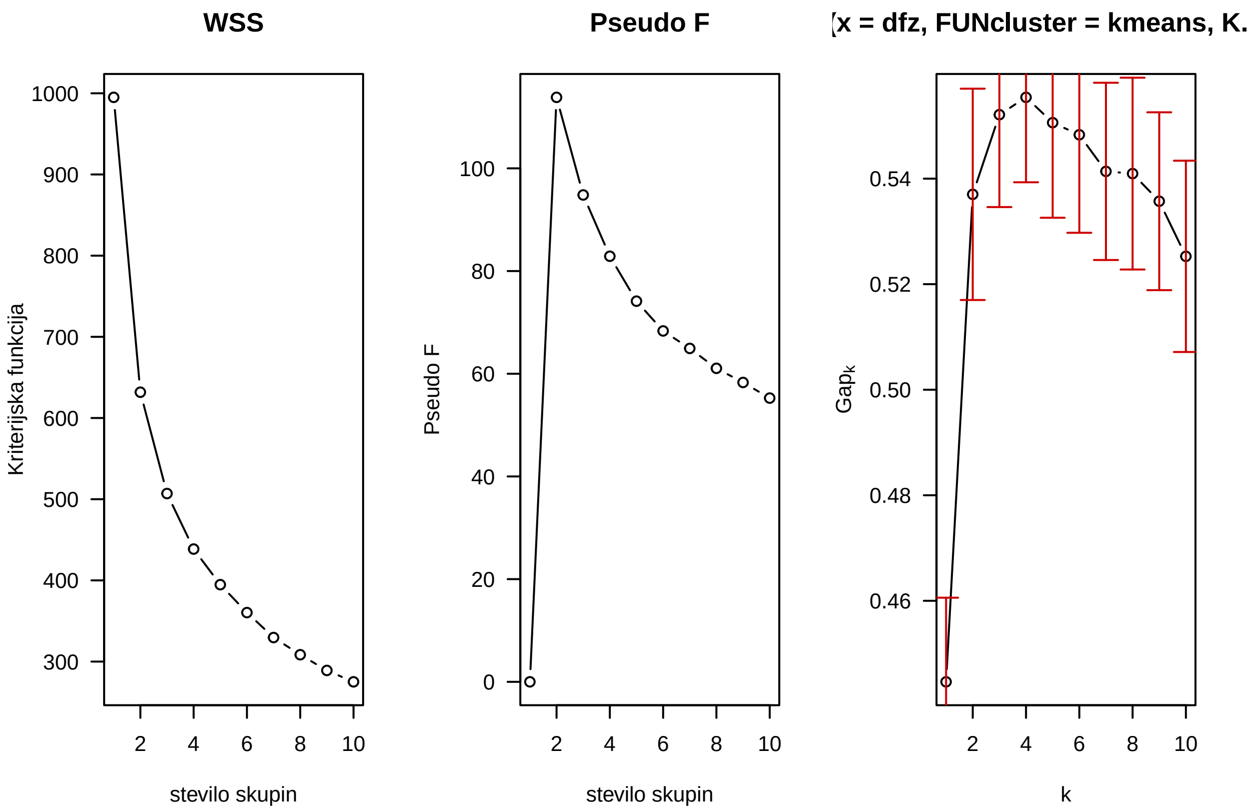
```r
}

# Pseudo F (Calinski and Harabasz index)
pseudoF <- 0
bss <- wss[1] - wss
for (k in 2:kmax) pseudoF <- c(pseudoF, (bss[k]/(k-1)) / (wss[k]/(nrow(dfz)-k)))

# gap statistika
(gap_stat <- clusGap(x=dfz, FUNcluster=kmeans, K.max=kmax))
```

```
## Clustering Gap statistic ["clusGap"] from call:
## clusGap(x = dfz, FUNcluster = kmeans, K.max = kmax)
## B=100 simulated reference sets, k = 1..10; spaceH0="scaledPCA"
##  --> Number of clusters (method 'firstSEmax', SE.factor=1): 3
##           logW   E.logW       gap     SE.sim
##  [1,] 4.994566 5.439212 0.4446460 0.01594209
##  [2,] 4.760047 5.297068 0.5370204 0.02002134
##  [3,] 4.649114 5.201232 0.5521178 0.01750428
##  [4,] 4.570348 5.125754 0.5554062 0.01609846
##  [5,] 4.517241 5.067849 0.5506078 0.01801816
##  [6,] 4.468143 5.016452 0.5483085 0.01857304
##  [7,] 4.433855 4.975227 0.5413717 0.01679319
##  [8,] 4.398703 4.939657 0.5409541 0.01817291
##  [9,] 4.370556 4.906275 0.5357188 0.01685166
## [10,] 4.353401 4.878676 0.5252752 0.01812347
```

```r
# grafizni prikaz (vse tri skupaj)
oldpar <- par(las=1, mfrow=c(1, 3))
plot(x=1:kmax, y=wss, type="b", main="WSS", xlab="stevilo skupin",
     ylab="Kriterijska funkcija")
plot(x=1:kmax, y=pseudoF, type="b", main="Pseudo F", xlab="stevilo skupin",
     ylab="Pseudo F")
plot(gap_stat)
```




```
par(oldpar)

km1 <- kmeans(x=dfz, centers=1, nstart=1)
km2 <- kmeans(x=dfz, centers=2, nstart=100)
km3 <- kmeans(x=dfz, centers=3, nstart=100)
km4 <- kmeans(x=dfz, centers=4, nstart=100)
```

Število skupin glede na posamezni graf:

- WSS: sprememba naklona izgleda največja pri **4** skupinah
- Pseudo F: maksimum doseze pri **2** skupinah
- gap statistika: najvišjo točko preden začne padati doseže pri **4** skupinah

Do podobnih ugotovitev smo prišli s pomočjo rezultatov hierarhičnega razvrščanja. Tam smo se odločali med 2, 3 ali 4 skupinami. Vendar pa težimo k večjemu številu skupin kot le 2.

Torej glede na analizo, ki smo jo naredili do sedaj so **4** najbolj primeren rezultat.

```
# Wardova kriterijska funkcija
WardKF <- function(data, cluster){
  # vsota kvadratov
  # x = ena spremenljivka
  ss <- function(x) sum((x-mean(x))^2)
  # vsota kvadratov znotraj ene skupine po vseh spremenljivkah
  # X = matrika, stolpci so spremenljivke
  withinss <- function(X) sum(apply(X=X, MARGIN=2, FUN=ss))
  # vsota kvadratov vseh skupin
  sum(by(data=data, INDICES=cluster, FUN=withinss))
}
```

```r
wkf <- NULL
wkf <- rbind(wkf, c(WardKF(dfz, hc.ward1), WardKF(dfz, hc.ward2), WardKF(dfz, hc.ward3), WardKF(dfz, hc
wkf <- rbind(wkf, c(WardKF(dfz, km1$cluster), WardKF(dfz, km2$cluster), WardKF(dfz, km3$cluster), WardKF
rownames(wkf) <- c("Ward", "Kmeans")
colnames(wkf) <- c("k=1", "k=2", "k=3", "k=4")
wkf
```

```
##        k=1      k=2      k=3      k=4
## Ward   995 645.6782 529.8618 464.3683
## Kmeans 995 631.7882 506.9825 438.5950
```

Vidimo da ima v vseh primerih (z izjemo prvega kjer sta enaka) K-means manjšo vrednost, kar tudi želimo. Primerjavo razvrtitev bomo naredili na številu skupin **4**.

```r
# kontingenčna tabela
table(km4$cluster, c(3, 2, 4, 1)[hc.ward4])
```

```
##
##      1  2  3  4
##   1 27  0 17  4
##   2  1 69  0  0
##   3  3  3 43  0
##   4  2  6  0 25
```

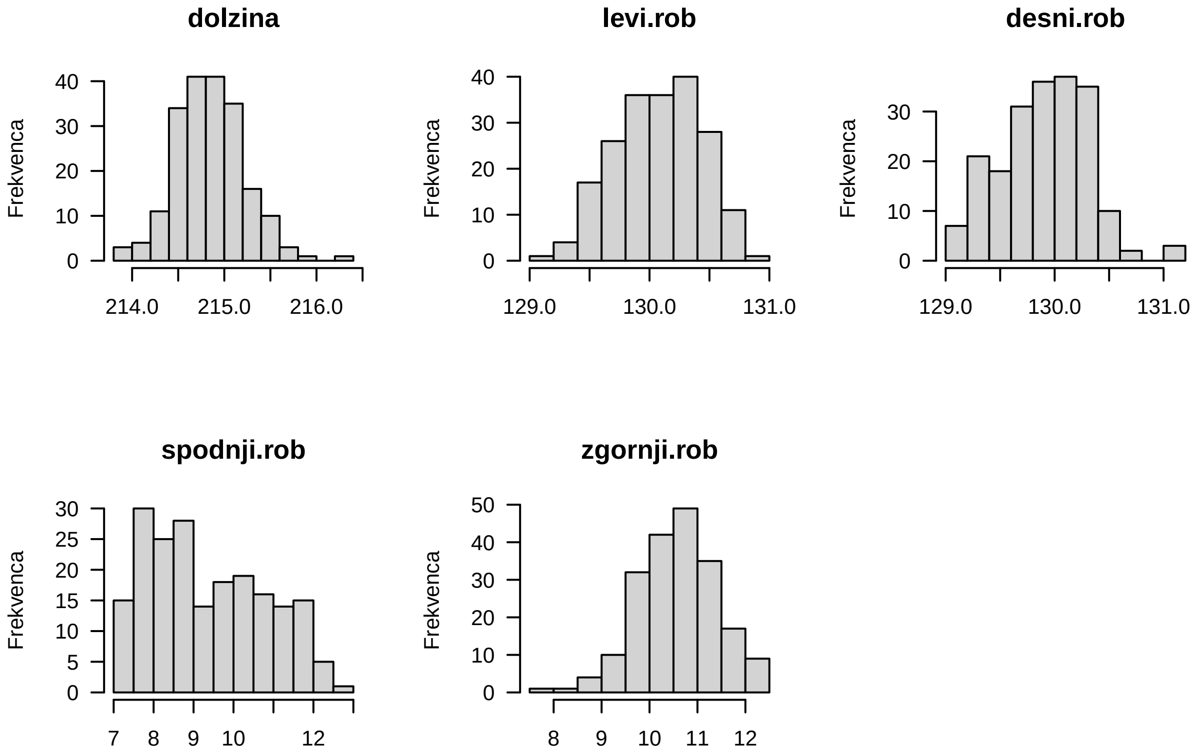
```r
# popravljen Randov indeks
adjustedRandIndex(x=km4$cluster, y=hc.ward4)
```

```
## [1] 0.6442817
```

Razvrščanje na podlagi modelov

Metoda predpostavlja multivariatno normalno porazdelitev zato si poglejmo porazdelitev spremenljivk

```r
oldpar <- par(las=1, mfrow=c(2, 3))
for (i in seq_along(ime)) {
  hist(data[, ime[i]],
       main=ime[i],
       xlab="",
       ylab="Frekvenca")
}
par(oldpar)
```



Nekatere spremenljivke niso porazdeljene po normalni porazdelitvi, zato ne moremo trditi da je zadoščen pogoj.

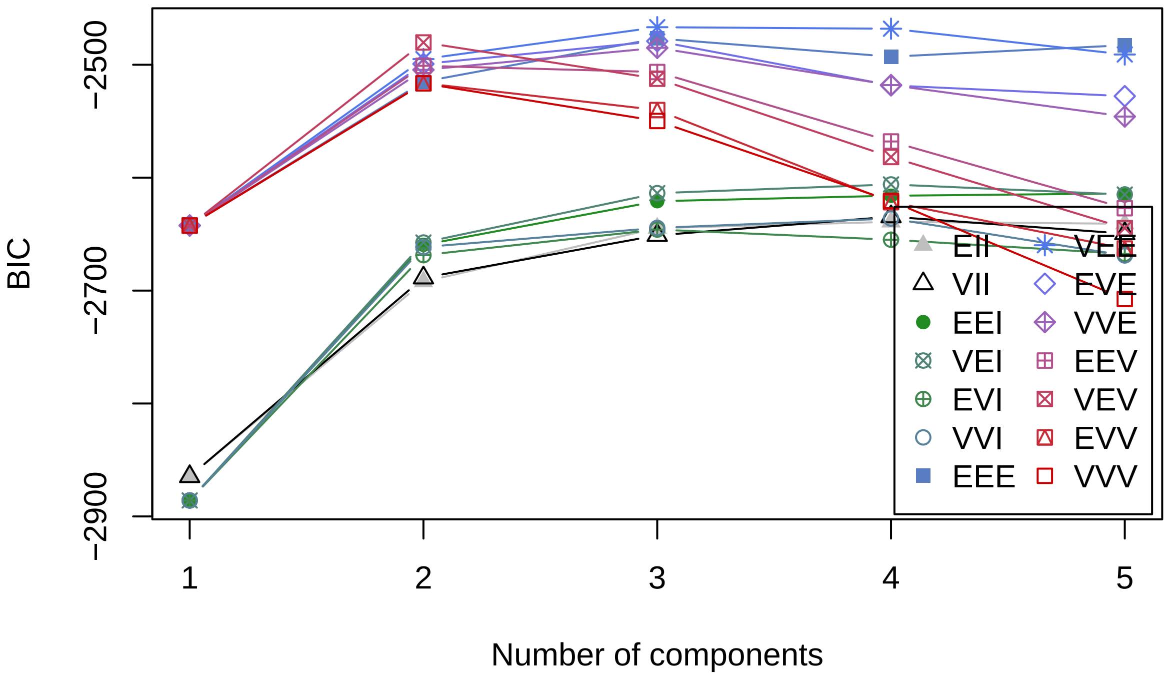
```r
(mc <- Mclust(data=dfz, G=1:5))
```

```
## 'Mclust' model object: (VEE,3)
##
## Available components:
##  [1] "call"           "data"           "modelName"      "n"
##  [5] "d"              "G"              "BIC"            "loglik"
##  [9] "df"             "bic"            "icl"            "hypvol"
## [13] "parameters"     "z"              "classification" "uncertainty"
```

```r
summary(mc)
```

```
## ----------------------------------------------------
## Gaussian finite mixture model fitted by EM algorithm
## ----------------------------------------------------
##
## Mclust VEE (ellipsoidal, equal shape and orientation) model with 3 components:
##
##  log-likelihood   n df       BIC       ICL
##        -1143.31 200 34 -2466.763 -2484.678
##
## Clustering table:
##  1  2  3
## 40 78 82
```

```r
plot(x=mc, what="BIC")
```



Najbolj primeren model je VEE (vsi elipticne oblike, razlicnih velikosti in enake orientacije) in 3 skupine.