Fokomponenselemzes es exploratoros faktorelemzes

Zoltan Kekecs

November 24 2020

Contents

1 Absztrakt

Ebben a gyakorlatban a “dimenzionalitás átkaval” birkozunk meg. Ezt a valtozok szamanak csokkentesevel oldjuk meg a fokomponenselemzes (PCA) es az exploratoros faktorelemzes (EFA) segitsegevel.

2 Adat kezeles es leiro statisztikak

2.1 Csomagok betoltese

Ennek a gyakorlatnak a soran az alabbi csomagokat fogjuk hasznalni:

```
library(GGally) # for ggcorr
library(corrr) # network_plot
library(ggcorrplot) # for ggcorrplot
library(FactoMineR) # multiple PCA functions
library(factoextra) # visualisation functions for PCA (e.g. fviz_pca_var)
library(paran) # for paran

library(psych) # for the mixedCor, cortest.bartlett, KMO, fa functions
library(car) # for vif
library(GPArotation) # for the psych fa function to have the required rotation functionalities
library(MVN) # for mvn function
library(ICS) # for multivariate skew and kurtosis test
library(tidyverse) # for tidy code
```

2.2 Sajat funkciok betoltese

Az alabbi fviz_loadnings_with_cor() a PCA es a faktorelemzes eredmenyeinke vizualizalasara szolgal.

```
fviz_loadnings_with_cor <- function(mod, axes = 1, loadings_above = 0.4){
  require(factoextra)
  require(dplyr)
  require(ggplot2)



if(!is.na(as.character(mod$call$call)[1])){
  if(as.character(mod$call$call)[1] == "PCA"){
  contrib_and_cov = as.data.frame(rbind(mod[["var"]][["contrib"]], mod[["var"]][["cor"]]))

vars = rownames(mod[["var"]][["contrib"]])
attribute_type = rep(c("contribution","correlation"), each = length(vars))
contrib_and_cov = cbind(contrib_and_cov, attribute_type)
contrib_and_cov

plot_data = cbind(as.data.frame(cbind(contrib_and_cov[contrib_and_cov[,"attribute_type"] == "contributi
names(plot_data) = c("contribution", "correlation", "vars")

plot_data = plot_data %>%
  mutate(correlation = round(correlation, 2))

plot = plot_data %>%
  ggplot() +
  aes(x = reorder(vars, contribution), y = contribution, gradient = correlation, label = correlation)+
  geom_col(aes(fill = correlation)) +
```

```
  geom_hline(yintercept = mean(plot_data$contribution), col = "red", lty = "dashed") + scale_fill_gradi
  xlab("variable") +
  coord_flip() +
  geom_label(color = "black", fontface = "bold", position = position_dodge(0.5))


}
} else if(!is.na(as.character(mod$Call)[1])){

  if(as.character(mod$Call)[1] == "fa"){
    loadings_table = mod$loadings %>%
      matrix(ncol = ncol(mod$loadings)) %>%
      as_tibble() %>%
      mutate(variable = mod$loadings %>% rownames()) %>%
      gather(factor, loading, -variable) %>%
      mutate(sign = if_else(loading >= 0, "positive", "negative"))

  if(!is.null(loadings_above)){
    loadings_table[abs(loadings_table[,"loading"]) < loadings_above,"loading"] = NA
    loadings_table = loadings_table[!is.na(loadings_table[,"loading"]),]
  }

  if(!is.null(axes)){

  loadings_table = loadings_table %>%
     filter(factor == paste0("V",axes))
  }


  plot = loadings_table %>%
      ggplot() +
      aes(y = loading %>% abs(), x = reorder(variable, abs(loading)), fill = loading, label =       rou
      geom_col(position = "dodge") +
      scale_fill_gradient2() +
      coord_flip() +
      geom_label(color = "black", fill = "white", fontface = "bold", position = position_dodge(0.5)) +
      facet_wrap(~factor) +
      labs(y = "Loading strength", x = "Variable")
  }
}




return(plot)

}
```

2.3 A Humor Styles Questionnaire betoltese

Alabb betoltjuk a “Humor Styles Questionnaire” adatbazist, ami a Martin et. al. (2003). kutatasabol szarmazik, akik a HSQ kerdoivet vettek fel 1071 szemellyel.

Az adatbazis elso 32 oszlopa Q1-Q32 a kerdoiv egyes teteleire adott valaszokat tartalmazza minden szemelytol. A valaszadoknak mind a 32 allitasrol ertekelnie kellett, hogy mennyire igaz az ra nezve. A valaszok ordinalis skalan mozognak, 1-tol 5-ig: 1=“soha vagy nagyon ritkan igaz”“, 5=”nagyon gyakran vagy soha nem igaz“. Ilyen allitasok szerepelnek a kerdoivben mint:”Q1: Altalaban nem nevetek vagy viccelodok masokkal." (Q1: “I usually don’t laugh or joke around much with other people.”)

A kerdoiv itemei a kovetkezoek:

- Q1. I usually don’t laugh or joke around much with other people.
- Q2. If I am feeling depressed, I can usually cheer myself up with humor.
- Q3. If someone makes a mistake, I will often tease them about it.
- Q4. I let people laugh at me or make fun at my expense more than I should.
- Q5. I don’t have to work very hard at making other people laugh—I seem to be a naturally humorous person.
- Q6. Even when I’m by myself, I’m often amused by the absurdities of life.
- Q7. People are never offended or hurt by my sense of humor.
- Q8. I will often get carried away in putting myself down if it makes my family or friends laugh.
- Q9. I rarely make other people laugh by telling funny stories about myself.
- Q10. If I am feeling upset or unhappy I usually try to think of something funny about the situation to make myself feel better.
- Q11. When telling jokes or saying funny things, I am usually not very concerned about how other people are taking it.
- Q12. I often try to make people like or accept me more by saying something funny about my own weaknesses, blunders, or faults.
- Q13. I laugh and joke a lot with my closest friends.
- Q14. My humorous outlook on life keeps me from getting overly upset or depressed about things.
- Q15. I do not like it when people use humor as a way of criticizing or putting someone down.
- Q16. I don’t often say funny things to put myself down.
- Q17. I usually don’t like to tell jokes or amuse people.
- Q18. If I’m by myself and I’m feeling unhappy, I make an effort to think of something funny to cheer myself up.
- Q19. Sometimes I think of something that is so funny that I can’t stop myself from saying it, even if it is not appropriate for the situation.
- Q20. I often go overboard in putting myself down when I am making jokes or trying to be funny.
- Q21. I enjoy making people laugh.
- Q22. If I am feeling sad or upset, I usually lose my sense of humor.
- Q23. I never participate in laughing at others even if all my friends are doing it.
- Q24. When I am with friends or family, I often seem to be the one that other people make fun of or joke about.
- Q25. I don’t often joke around with my friends.
- Q26. It is my experience that thinking about some amusing aspect of a situation is often a very effective way of coping with problems.
- Q27. If I don’t like someone, I often use humor or teasing to put them down.
- Q28. If I am having problems or feeling unhappy, I often cover it up by joking around, so that even my closest friends don’t know how I really feel.
- Q29. I usually can’t think of witty things to say when I’m with other people.
- Q30. I don’t need to be with other people to feel amused – I can usually find things to laugh about even when I’m by myself.
- Q31. Even if something is really funny to me, I will not laugh or joke about it if someone will be offended.
- Q32. Letting others laugh at me is my way of keeping my friends and family in good spirits.

A HSQ-n kivul az adatbazis tartalmaz mas valtozokat is:

- gender - Faktor valtozo: 1=male, 2=female, 3=other)
- accuracy - Mennyire ereztek pontosnak a valasazikat a HSQ keredeskre. 0-100-as skalan ertekelve. (A vizsgalati szemelyeknek azt mondtak hogy az irjon 0-t aki nem szeretne hogy a kiserletben felhasznaljak az adatat).
- life_stress - Ez egy uj valtozo ami em volt benne az eredeti kutatasban (szimulalt adat). Arra utal, hogy altalanossagban mennyire stresszesnek erzi az eletet a valaszado 0-9-es skalan, ahol a 0 azt jelenti hogy semennyire, 9 azt jelenti, hogy extremen stresszes.

Az adatbazis tartalmaz hianyzo adatokat. Az egyszeruseg kedveert most zarjuk ki azokat a valaszadokat akiknek akar egy adatpont is hianyzik a sorabol.

```
hsq <- read_csv("https://raw.githubusercontent.com/kekecsz/PSZB17-210-Data-analysis-seminar/master/semi
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double()
## )
## i Use `spec()` for the full column specifications.
```

```
hsq <- hsq %>%
  drop_na()


hsq %>%
describe()
```

2.4 Az adatsor megtekintese

Vizsgaljuk meg az adatok strukturajat es az alapveto leiro statisztikakat.

```
str(hsq)

summary(hsq)
```

Mondjuk hogy szeretnenk meghatarozni melyek az emerekek humor stilusanak legfobb jellegzetessegei, melyek meghatarozzak a szemely stresszel valo viszonyat. Masneven hogy a humor stilus mely aspektusaival segithetnek bejosolni a szemely altalanos stressz-szintjet. Egy modja hogy ezt meghatarozzuk, hogy epitunk egy regresszios modellt, amiben a life_stress valtozot josoljuk be a Humor Style Questionnaire itemeivel (Q1-Q32).

Amikor lefuttatjuk ezt a modellt, a regresszios egyutthatok szignifikanciaja alapjan ugy tunik hogy a modell egeszeben szignifikansan jobb mint a null modell, es tobb olyan item is van aminek van sziginfikans hozzaadott erteke a modellhez, vagyis erdemes a humor stilust figyelembe venni a stressz meghatarozasanal.

De ha arra vagyunk kivancsiak hogy a humor stilus mely aspektusai fontosak, ezt nehez ebbol a regresszios elemzesbol megallapitani. Egyreszt nem bizhatunk meg teljesen a p-ertekekben, mert 32 statistzikai tesztet hajtottunk verge, a 32 item hozzaadott megmagyarazo ertekenek tesztelesekor, ami nagyban noveli az elsofaju hiba (fals pozitiv) valoszinuseget. Vagyis nagy a valoszinusege hogy a szignifikanskent megjelol prediktorok kozul egy, vagy tobb nincs valodi osszefuggesben a stresszel a populacioban.

Masreszt figyelembe kell venni hogy a prediktorok korrelalnak egymassal, vagyis bizonyos prediktorok a stressz varianciajanak hasonlo rezset magyarazzak, es ebben a modellben lehet hogy az egyes prediktorok mas korrelalo prediktorok hatasat elmaszkoljak.

```
mod_allitems = lm(life_stress ~ Q1 + Q2 + Q3 + Q4 + Q5 + Q6 + Q7 + Q8 + Q9 + Q10 +
                    Q11 + Q12 + Q13 + Q14 + Q15 + Q16 + Q17 + Q18 + Q19 + Q20 +
```

```
                    Q21 + Q22 + Q23 + Q24 + Q25 + Q26 + Q27 + Q28 + Q29 + Q30 +
                    Q31 + Q32,
                    data = hsq)

summary(mod_allitems)
```

```
##
## Call:
## lm(formula = life_stress ~ Q1 + Q2 + Q3 + Q4 + Q5 + Q6 + Q7 +
##     Q8 + Q9 + Q10 + Q11 + Q12 + Q13 + Q14 + Q15 + Q16 + Q17 +
##     Q18 + Q19 + Q20 + Q21 + Q22 + Q23 + Q24 + Q25 + Q26 + Q27 +
##     Q28 + Q29 + Q30 + Q31 + Q32, data = hsq)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3528 -0.8639  0.0307  0.8807  3.8981
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.084097   0.623579   0.135 0.892750
## Q1          -0.097889   0.050921  -1.922 0.054856 .
## Q2           0.007704   0.047341   0.163 0.870764
## Q3           0.030141   0.043687   0.690 0.490402
## Q4           0.170860   0.045619   3.745 0.000191 ***
## Q5          -0.066691   0.053180  -1.254 0.210121
## Q6           0.026375   0.055614   0.474 0.635433
## Q7          -0.023993   0.046146  -0.520 0.603235
## Q8           0.105826   0.050916   2.078 0.037935 *
## Q9          -0.052866   0.039836  -1.327 0.184797
## Q10          0.009067   0.050208   0.181 0.856732
## Q11         -0.015873   0.039779  -0.399 0.689967
## Q12          0.202653   0.044006   4.605 4.68e-06 ***
## Q13          0.026218   0.067533   0.388 0.697937
## Q14          0.052088   0.045885   1.135 0.256577
## Q15          0.015961   0.040418   0.395 0.693010
## Q16          0.163045   0.043615   3.738 0.000196 ***
## Q17          0.027425   0.054744   0.501 0.616506
## Q18          0.010161   0.050137   0.203 0.839442
## Q19          0.034051   0.041819   0.814 0.415709
## Q20          0.208081   0.055749   3.732 0.000201 ***
## Q21         -0.103708   0.066913  -1.550 0.121496
## Q22          0.062369   0.039423   1.582 0.113967
## Q23          0.044103   0.041631   1.059 0.289695
## Q24          0.222733   0.043525   5.117 3.74e-07 ***
## Q25          0.031822   0.070080   0.454 0.649871
## Q26         -0.072567   0.049911  -1.454 0.146287
## Q27         -0.020883   0.038454  -0.543 0.587211
## Q28          0.201723   0.036348   5.550 3.70e-08 ***
## Q29         -0.031136   0.044079  -0.706 0.480130
## Q30          0.006414   0.046809   0.137 0.891043
## Q31         -0.011109   0.042425  -0.262 0.793497
## Q32          0.148716   0.045016   3.304 0.000990 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 1.302 on 960 degrees of freedom
## Multiple R-squared:  0.3438, Adjusted R-squared:  0.3219
## F-statistic: 15.72 on 32 and 960 DF,  p-value: < 2.2e-16
```

A prediktorok kozotti korrelaciorol megbizonyosodhatunk ha lekerdezzuk a prediktorok korrelacios matrixat a cor() fugvennyel. A vif() fugvennyel megnezhetjuk hogy ez az interkorrelacio problemas multikollinearitashoz vezet-e a modellunkben (ebben az esetben a vif 3 alatt marad minden esetben, vagyis nincs szamottevo multikollinearitas, de mas hasonlo esetben amikor kerdoivek minden itemet a modellbe epitjuk ez is konnyen elofordulhat.)

```
hsq_items_only = hsq %>%
                   select(Q1:Q32)

cor = hsq_items_only %>%
  cor()

cor

vif(mod_allitems)
```

2.5 A dimenzionalitas atka

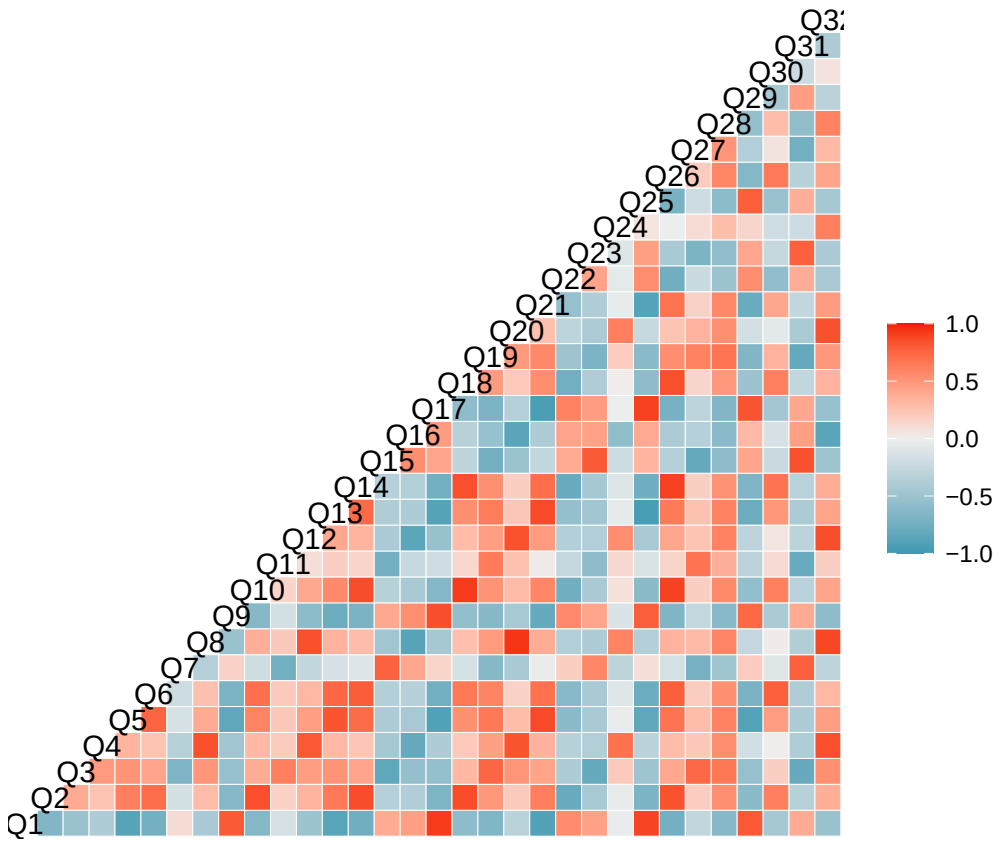
Vagyis 32 egymassal korrelalo prediktor bevonasa a modellbe nem idealis annak a megertesere, hogy a humor stilus mely aspektusai fontosak a stressz meghatarozasaban. Sot, ez kifejezetten problemas minel kisebb a minta-elemszam a prediktorok szamahoz kepest. Ezt a statisztikusok gyakran a "**dimenzionalitas atka**"-kent emlegetik. A "dimenzionalitas" arra utal, hogy minel tobb valtozo van a modellunkben, az adatokat annal tobb-dimenzios terben lehet modellezni. Peldaul az egyszeru regresszional amikor csak egy kimeneti valtozonk es egy prediktorunk van, az adatok egy ketdimenzios terben abrazolhatok: a kimeneti valtozo az y tengelyen (dimenzion), a prediktor pedig az x tengelyen (dimenzion) abrazolva. A regresszios egyenes ebben az esetben valoban egy egyenes. Amikor mar ket prediktorunk van, az adatok egy harom dimenzios terben abrazolhatok, es a regresszios modell egy regresszios sik, nem csak egy egyenes. Amikor 32 prediktorunk van a fenti peldaban, a modellunk egy 33 dimenzios felulet. Minel tobb dimenzioban mozoghat a modell, annal flexibilisebb, vagyis annal nagyobb a tulillesztes valoszinusege. Ezert torekednunk kell a dimenziok (prediktorok) szamanak minimalizalasara hogy elkerulhessuk a tulillesztest.

A tulillesztesnek annal nagyobb a veszelye minel inkabb kozelit a prediktorok szama a megfigyelesek szamahoz. Amikor egy prediktorunk van, vagyis az adat ketdimenzios terben irhato le. Ha csak ket megfigyelesunk lenne, akkor a regresszios egyenes tokeletes illeszkedest erhetne el, mert mindig van egy olyan egyenes ami osszekot ket pontot, es ezt a regresszio megtalalja. Belathato hogy ez a regresszios egyenes amit csak ket megfigyeles alapjan ilélesztettunk nagyon serulekeny lesz az adatokban levo hibara, es nem fogja megbizhatoan megranagni a populacioban talalhato valos osszefuggest a prediktor es a kimeneti valtozo kozott. Szoval ahogy a tulillesztesrol szolo oran is lathattuk, a tokeletes illeszkedes a mintankhoz valojaban nem jo, mert a populacio helyett csak a mintaban levo latszolagos mintazatokra illeszkedik a modell amit nagyon befolyasolhat a mintaveteli hiba. Ugyan ez a helyzet amikor egy 2 prediktoros modellt illesztunk 3 megfigyelesre. Mivel a regresszios modell itt egy regresszios sik, egy sikot mindig lehet ugy forgatni hogy pontosan osszekosson 3 pontot, igy az illeszkedes tokelehetes lesz, ami eros tulilleszteshez vezet. Ebbol extrapolalva konnyen lathato hogy ahogy a prediktorok szama kozelit a megfigyelesek szamahoz, a modellunk egyre kevesbe lesz megbizhato a populacio megismeresehez, es egyre serulekenyebb lesz a tulilessztesre.

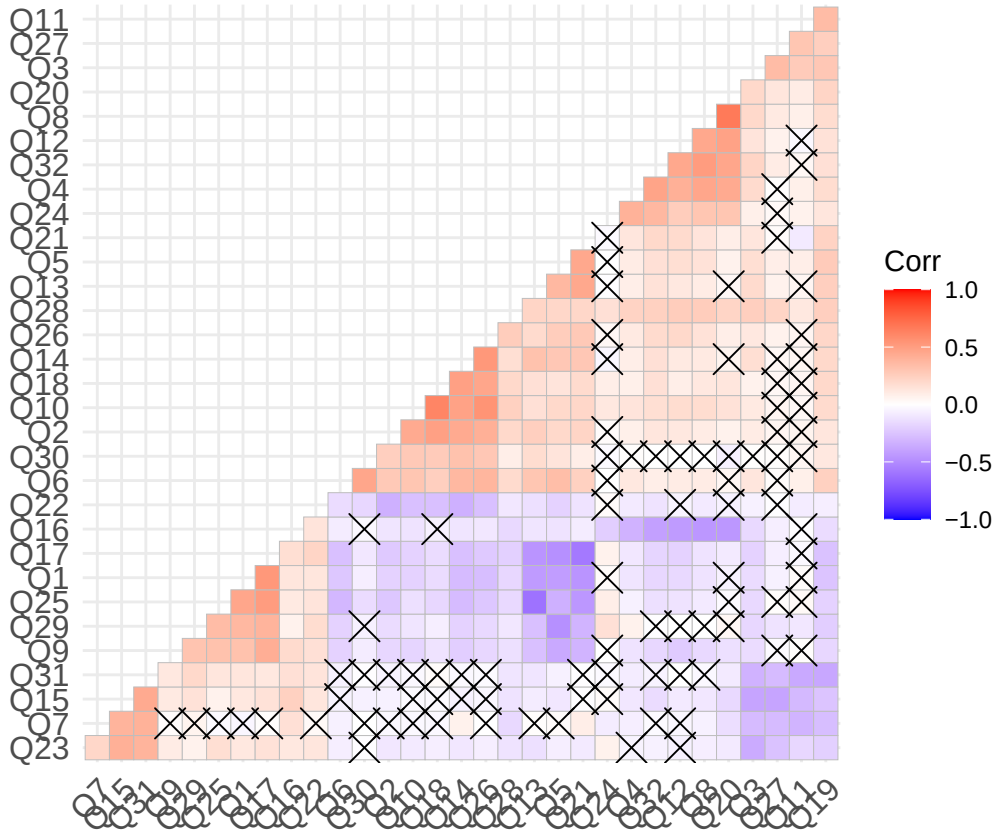
2.6 A korrelacios struktura vizualizacioja

Nehany valtozo osszefuggeset konnyen atlathatova tehetjuk vizualizacion keresztul. Azonban ha **sok valtozoval van dolgunk**, a vizualizacio es egyeb korabban tanult feltaro adatelemzesi technikak kudarcot vallhatnak egyszeruen azert mert tul sok az informacio amit nehez atlatni es vizualizalni. Ez jol latszik az alabbi abrakon.

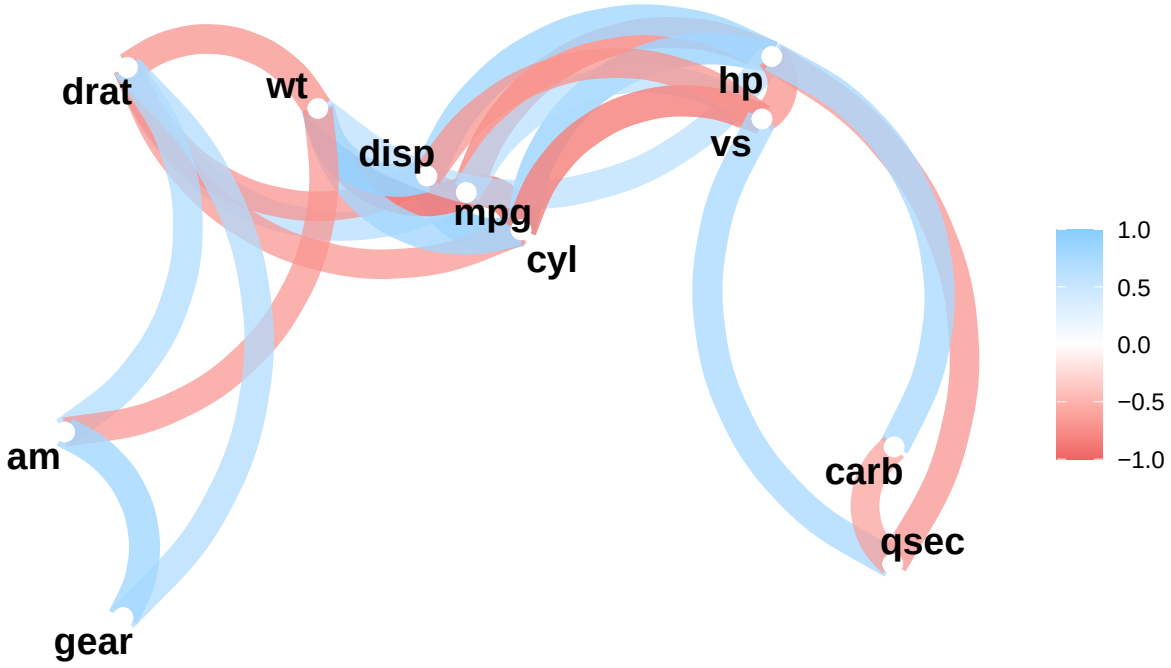
```
ggcorr(cor)
```



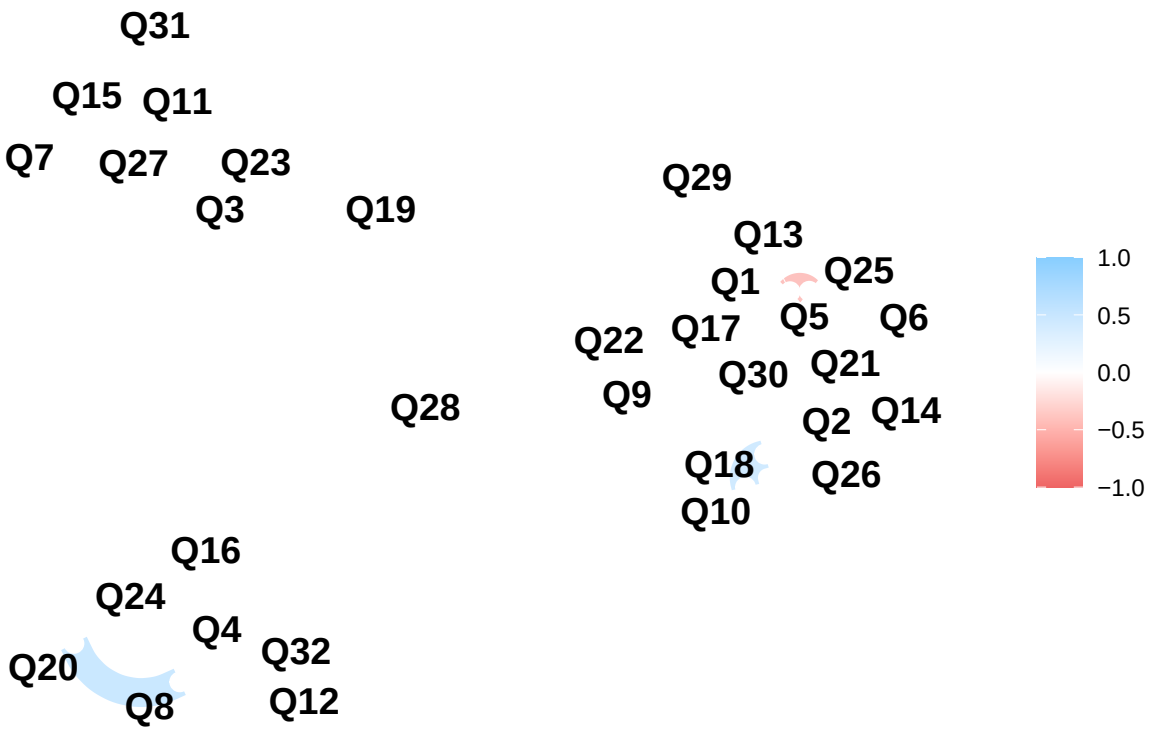

```
ggcorrplot(cor(hsq_items_only), p.mat = cor_pmat(hsq_items_only), hc.order=TRUE, type='lower')
```




```
# from a different dataset, mtcars, because there are too many variables in hsq to be displayed here
cor(mtcars) %>% network_plot(min_cor=0.6)
```




```
cor(hsq_items_only) %>% network_plot(min_cor=0.6)
```



3 Fokomponenselemzes

3.1 A fokomponenselemzes modell megepitese

Amikor a dimenzionalitas atkaval szembesulunk, az egyik megoldasi lehetoseg a valtozok (dimenziok) szamanak csokkentese. Ha szimplan kizarnak nehany valtozot, mondjuk a valtozok modellhez hozzaadott bejoslo erteke alapjan, az szinten hozzajarulna a tulilleszteshez (ezt korabbi orakon lathattuk).

Ehelyett egy masik lehetoseg, hogy osszevonjuk a modellben levo valtozokat valamilyen szempontrendszer szerint. Ha ranezunk a korrelacios matrixra es a korrelacios abrakra, lathatjuk hogy vanak klaszterek a valtozok kozott, es a klasztereken belul a valtozok jobban korrelalnak, mint klaszterek kozott. Idealis lenne, ha azokat a valtozokat vonnank ossze amelyek ugyanazon klaszteren belul vannak. A **fokomponenselemzes** egy matematikai megoldast jelent arra, hogy ezt hogyan tehetjuk meg.

A **fokomponenselemzest** (principal component analysis, roviden PCA), hasznalhatjuk arra, hogy lecsokkentsuk a valtozok szamat amivel dolgoznunk kell, ugy, hogy kozben **a leheto legtobb informaciot tartunk meg az adatok variabilitasarol**.

3.2 Hogyan mukodik a PCA?

A fokomponenselemzes (PCA) soran az a celunk altalaban hogy csokkentsuk a dimenziok szamat, amivel az adataink leirhatok. Ezt ugy erjuk el, hogy eloszor uj dimenziokat keresunk, amelyek minel nagyobb reszet magyarazzak az adatok valtozekonysaganak, majd eldobjuk azokat a dimenziokat, amik a variancia megertesenek viszonylag kis resezert felnek. A PCA soran eloszor azonositunk egy elsodleges dimenziot, ami menten az adatok a legnagyobb varianciat mutatjak. Ez utan azonositunk egy erre meroleges uj dimenziot, ami a **fennmarado** variancia legnagyobb reszet kepes magyarazni, es igy tovabb, addig amig el nem erjuk az eredetileg a fokomponenselemzesbe rakott valtozok szamat. (Mivel a dimenziok merolegesek egymasra,

ezert ket dimenzio a variancia mindig valamilyen uj reszet irja le, nincs redundancia a dimenziok altal megmagyarazott varianciaban.) Igy a fokomponenselemzes vegere egy uj koordinatarendszert kapunk, amiben az adatokat abrazolhatjuk.

Fontos, hogy az uj koordinatarendszer dimenzioi nagyban elternak abban hogy az adatok mennyire variabilisek az adott dimenzioban. Az elso nehany "fokomponens" (dimenzio) amit kivalasztottunk a varianica nagyon nagy reszet lefedi, es a fennmarado dimenziokban az adatok alig mutatnak variabilitast. Egy adott dimenzioban a variancia merteket **eigenvalue**-nak nevezzuk. Ahogy az elsotol az utolsokent azonositott dimenzio fele haladunk az eigenvalue egyre csokken (vagyis a dimenzioban megfigyelheto variancia). Ez engedi meg hogy csokkentsuk a dimenziok szamat, mert a PCA vegen azonositott dimenziokon elhanyagolhato lesz az adatok kulonbozosege egymastol, igy ezeket a dimenziokat elvethetjuk, es csak a **leghasznosabb dimenziokat tartjuk meg**.

Kepzeljuk el peldaul hogy egy kutatasban arra vagyunk kivancsiak hogy mennyire erettek az iskolakezdo elso osztalyosok. A kutatasban merjuk a gyerekek verbalis keszsegeit, szociabilitasat, es eletkorat evekben. Kiderul hogy a vizsgalt mintaban szinte minden elso osztalyos 6 eves, vagyis szinte semmi variabilitas nincs az eletkorban (ha csak evekben merjuk). Ettol a valtozotol akar el is tekinthetunk a kutatasunkban, hiszen annyira nem kulonboznek benne a vizsgalt szemlyeink. Ezzel szemben a szociabilitasban es a verbalis keszsegekben nagyobb szemlyeke kozotti kulonbseget merunk, ezert ezek fontos indikatorok a kutatasunkban abbol a szempontbol, hogy a gyerekeket megkulonboztessuk erettseguk szintjeben. A fokomponenselemzes soran mestersegesen generalunk uj valtozokat ugy hogy azok direkt nagyon nagy vagy nagyon keves varianciat magyarazzanak, hogy a keves varianciat magyarazo valtozokat elvethessuk ugty, hogy kozben minel tobb informaciot tartsunk meg az adatok kulonbozosegerol.

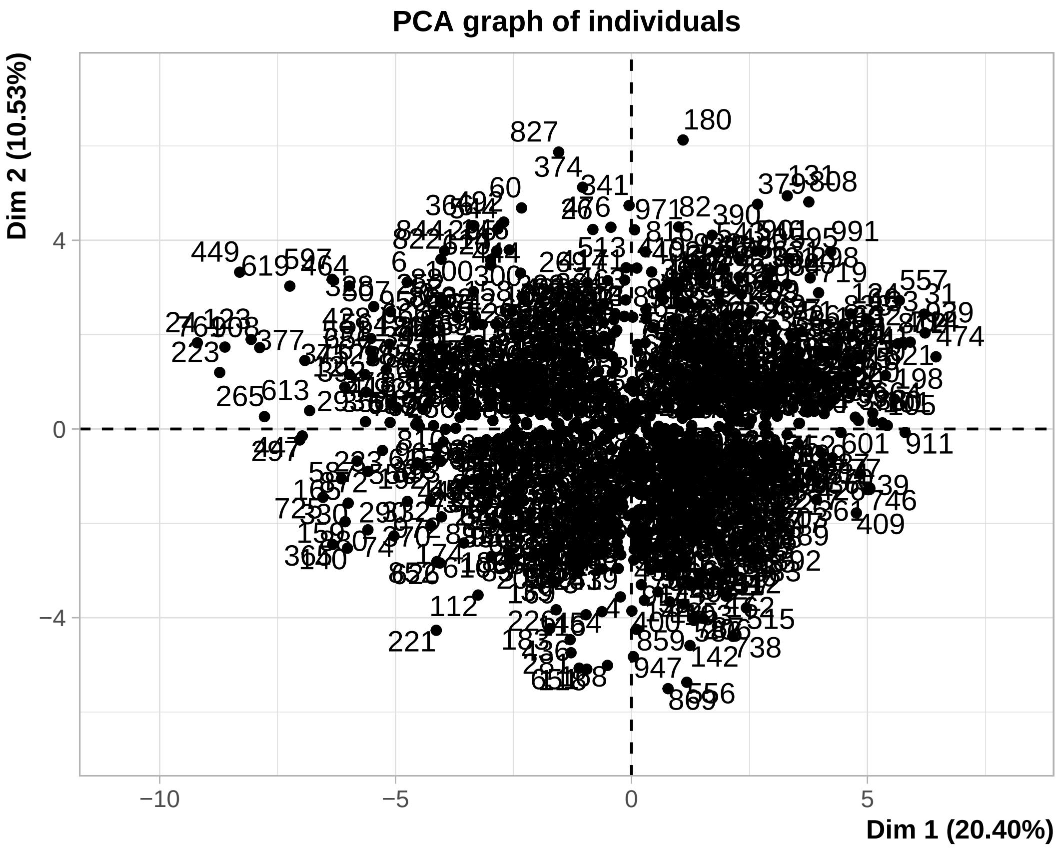
3.3 PCA hasznalata R-ben

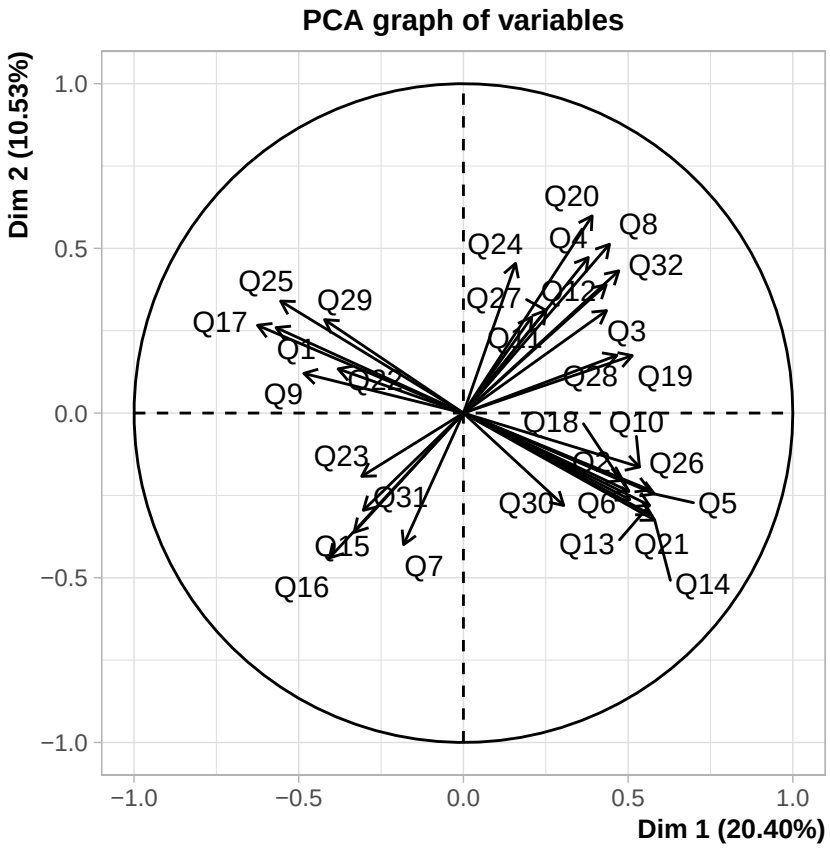
A fokomponenselemzest a **PCA() (principal component analysis)** funkcioval tudjuk elvegezni a FactoMineR package-bol. Az elemzes eredmenyet egy pca_mod modell objektumba mentettem el. Ami egybol kiad ket **abrat a ket legfontosabb fokomponensrol (dimenziorol)** amik a leghatekoonyabban irjak le az adatokat.

Az egyik abran az latszik, hogy **az egyes megfigyelesek** (ebben az esetben az egyes auto-modellek) **hol helyezkednek el a ket dimenzio menten**. A masodik abra pedig arrol szol, hogy a **dimenziok milyen korrelaciot mutatnak az eredeti valtozokkal**. A szaggatott vonalak mutatjak a fokomponenseket. A nyilak minel kozelebb fekszenek a szaggatott vonalhoz, a valtozo annal inkabb egyuttjar az adott dimenzioval a masik dimenzioval szemben.

Peladaul a Q9 es a Q10 itemeket sokkal jobban leirja a Dim1 mint a Dim2. (a nyil iranya alapjan megallapithato hogy a Q9 negativan, a Q10 pozitivan korrelal a Dim1-el.) Ezzel szemben a Q8 valtozo nyila a ket dimenzio kozott helyezkedik el, ami azt jelenti hogy vagy midkettovel nagyjabol azonos mertekben korrelal (ez lehet nagyon kicsi, vagy akar nagyon nagy korrelacio is).

```
pca_mod <- PCA(hsq_items_only)
```


PCA graph of individuals
Dim 2 (10.53%)
Dim 1 (20.40%)
4
0
−4
−10
−5
0
5



Arra oda kell figyelnunk hogy kategorikus valtozok ne keruljenek a fokomponenselemzes valtozoi koze.

A PCA() funkcioban lehetosegunk van arra hogy meghatarozzunk olyan valtozokat az adatbazisban, amiket nem szeretnenk beepiteni a PCA modellbe.

Azokat a folytonos valtozokat, amiket nem szeretnenk figyelembevenni a PCA soran, a quanti.sup parameterben kell megadnunk, azokat pedig amik kategorikusak a quali.sup parameterben. Itt az addott valtozo oszlopszamat kell megadnunk, nem pedig a nevet, igy ezt eloszor ki kell keresnunk. Ezt megtehetjuk a which(names(hsq) == “valtozo neve”) funkcioval.

Az alabbi peldaban a drat folytonos, es a cyl, vs, es az am kategorikus valtozokat kiemeljuk a modellbol, igy azok nincsenek figyelembe veve a pca_mod3 fokomponenseinek meghatarozasa soran, de az abrakon ettol meg szerepelnek. Ebben az esetben termeszetesen a modell ujra illesztesre kerul, es a szamszeru ertekek megvaltoznak a korabbi futtatashoz kepest amikor ezek a valtozok meg szerepeltek a modellben.

```
which(names(hsq) == "gender")
```

```
## [1] 38
```

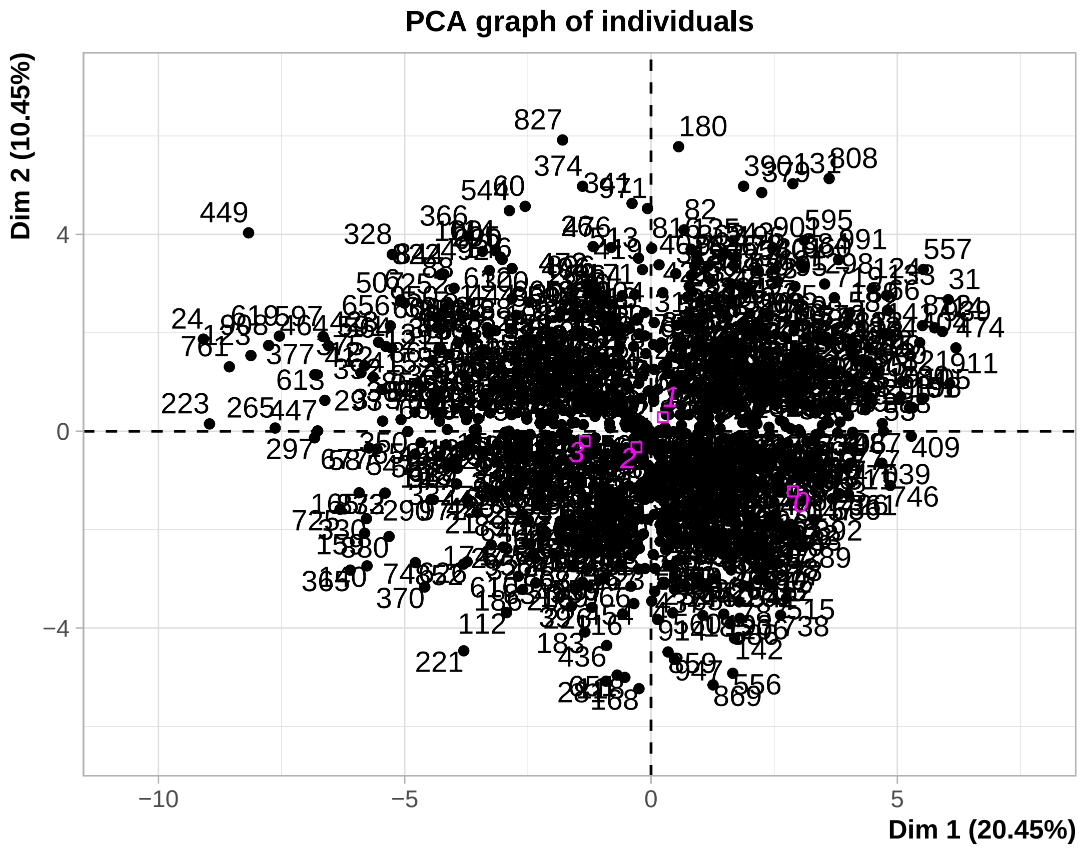
```
which(names(hsq) == "affiliative")
```

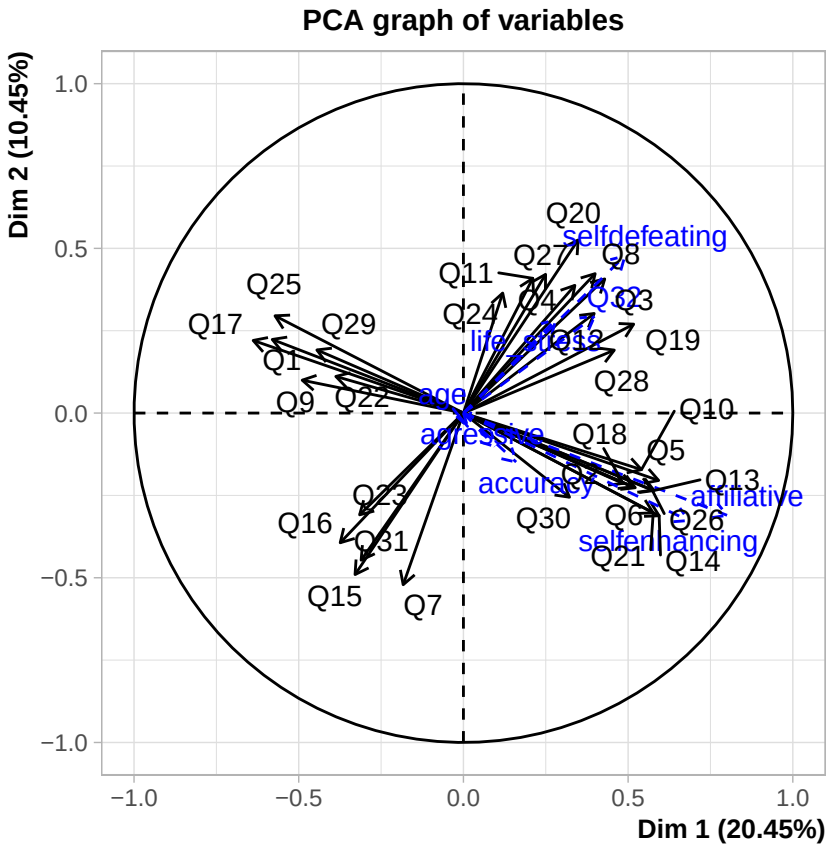
```
## [1] 33
```

```
which(names(hsq) == "life_stress")
```

```
## [1] 40
```

```
pca_mod2 <- PCA(hsq, quanti.sup = c(32, 33, 34, 35, 36, 37, 39, 40), quali.sup = 38)
```


PCA graph of individuals
Dim 2 (10.45%)
Dim 1 (20.45%)
−10
−5
0
5
4
0
−4
1
2
3
0




```
summary(pca_mod2)
```

```
##
## Call:
## PCA(X = hsq, quanti.sup = c(32, 33, 34, 35, 36, 37, 39, 40),
##      quali.sup = 38)
##
##
## Eigenvalues
##                        Dim.1   Dim.2   Dim.3   Dim.4   Dim.5   Dim.6   Dim.7
## Variance               6.338   3.238   2.656   2.236   1.198   1.060   1.036
## % of var.             20.446  10.447   8.568   7.214   3.863   3.421   3.342
## Cumulative % of var.  20.446  30.892  39.461  46.675  50.538  53.959  57.301
##                        Dim.8   Dim.9  Dim.10  Dim.11  Dim.12  Dim.13  Dim.14
## Variance               0.926   0.808   0.789   0.770   0.702   0.690   0.662
## % of var.              2.986   2.607   2.545   2.484   2.265   2.227   2.135
## Cumulative % of var.  60.286  62.893  65.438  67.921  70.187  72.413  74.549
##                       Dim.15  Dim.16  Dim.17  Dim.18  Dim.19  Dim.20  Dim.21
## Variance               0.623   0.611   0.576   0.555   0.542   0.530   0.499
## % of var.              2.010   1.970   1.859   1.792   1.748   1.711   1.610
## Cumulative % of var.  76.559  78.529  80.388  82.180  83.928  85.639  87.248
##                       Dim.22  Dim.23  Dim.24  Dim.25  Dim.26  Dim.27  Dim.28
## Variance               0.487   0.473   0.444   0.424   0.415   0.399   0.364
## % of var.              1.572   1.527   1.432   1.369   1.340   1.287   1.174
## Cumulative % of var.  88.821  90.348  91.780  93.149  94.488  95.775  96.949
##                       Dim.29  Dim.30  Dim.31
```

```
## Variance               0.349   0.309   0.288
## % of var.              1.127   0.996   0.929
## Cumulative % of var.  98.075  99.071 100.000
##
## Individuals (the 10 first)
##                   Dist    Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3
## 1              |  3.825 | -0.679  0.007  0.032 | -1.301  0.053  0.116 | -1.436
## 2              |  4.483 | -1.969  0.062  0.193 | -0.748  0.017  0.028 |  1.133
## 3              |  3.708 | -0.137  0.000  0.001 | -1.154  0.041  0.097 |  0.223
## 4              |  4.336 |  0.015  0.000  0.000 | -3.455  0.371  0.635 | -0.377
## 5              |  3.696 | -1.985  0.063  0.289 |  0.294  0.003  0.006 | -0.176
## 6              |  7.524 | -5.089  0.412  0.458 |  2.670  0.222  0.126 |  0.235
## 7              |  4.351 |  2.136  0.073  0.241 | -0.737  0.017  0.029 | -0.686
## 8              |  4.578 | -1.164  0.022  0.065 | -1.101  0.038  0.058 | -2.182
## 9              |  9.066 | -3.920  0.244  0.187 |  1.809  0.102  0.040 | -1.511
## 10             |  5.932 |  2.837  0.128  0.229 | -2.112  0.139  0.127 |  1.446
##                   ctr   cos2
## 1              0.078  0.141 |
## 2              0.049  0.064 |
## 3              0.002  0.004 |
## 4              0.005  0.008 |
## 5              0.001  0.002 |
## 6              0.002  0.001 |
## 7              0.018  0.025 |
## 8              0.181  0.227 |
## 9              0.087  0.028 |
## 10             0.079  0.059 |
##
## Variables (the 10 first)
##                   Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr
## Q1             | -0.580  5.313  0.337 |  0.223  1.541  0.050 |  0.139  0.725
## Q2             |  0.514  4.162  0.264 | -0.224  1.547  0.050 |  0.131  0.650
## Q3             |  0.429  2.904  0.184 |  0.407  5.118  0.166 | -0.266  2.657
## Q4             |  0.338  1.806  0.114 |  0.388  4.639  0.150 |  0.461  7.985
## Q5             |  0.593  5.542  0.351 | -0.205  1.292  0.042 | -0.118  0.525
## Q6             |  0.522  4.295  0.272 | -0.227  1.587  0.051 |  0.014  0.007
## Q7             | -0.183  0.527  0.033 | -0.521  8.386  0.272 |  0.309  3.599
## Q8             |  0.400  2.524  0.160 |  0.423  5.530  0.179 |  0.503  9.515
## Q9             | -0.489  3.779  0.240 |  0.100  0.308  0.010 |  0.042  0.067
## Q10            |  0.541  4.610  0.292 | -0.172  0.910  0.029 |  0.285  3.048
##                  cos2
## Q1             0.019 |
## Q2             0.017 |
## Q3             0.071 |
## Q4             0.212 |
## Q5             0.014 |
## Q6             0.000 |
## Q7             0.096 |
## Q8             0.253 |
## Q9             0.002 |
## Q10            0.081 |
##
## Supplementary continuous variables
##                   Dim.1   cos2    Dim.2   cos2    Dim.3   cos2
```

```
## Q32           |  0.398  0.158 |  0.294  0.086 |  0.377  0.142 |
## affiliative   |  0.800  0.641 | -0.310  0.096 | -0.221  0.049 |
## selfenhancing |  0.683  0.467 | -0.326  0.106 |  0.225  0.051 |
## agressive     |  0.095  0.009 | -0.101  0.010 |  0.180  0.032 |
## selfdefeating |  0.490  0.240 |  0.477  0.228 |  0.579  0.335 |
## age           |  0.012  0.000 | -0.040  0.002 |  0.024  0.001 |
## accuracy      |  0.160  0.026 | -0.149  0.022 | -0.084  0.007 |
## life_stress   |  0.280  0.079 |  0.283  0.080 |  0.329  0.108 |
##
## Supplementary categories
##                    Dist    Dim.1   cos2 v.test    Dim.2   cos2 v.test    Dim.3
## 0             |  3.604 |  2.879  0.638  2.562 | -1.222  0.115 -1.521 | -0.595
## 1             |  0.428 |  0.237  0.307  3.220 |  0.284  0.439  5.385 | -0.123
## 2             |  0.506 | -0.296  0.341 -3.320 | -0.326  0.416 -5.125 |  0.147
## 3             |  2.811 | -1.347  0.230 -1.519 | -0.202  0.005 -0.318 |  0.507
##                  cos2 v.test
## 0             0.027 -0.818 |
## 1             0.083 -2.583 |
## 2             0.084  2.547 |
## 3             0.032  0.882 |
```

3.4 Hany fokemponenst nyerjunk ki?

A fokomponenselemzes egy dimenzioredukcios technika, vagyis **celunk hogy kevesebb dimenzionk legyen** az elemzes vegere, mint ahany valtozoval kezdtuk az elemzest. Viszont hogyha ranezunk a model summary-ra, lathatjuk hogy a PCA funkcio alapertelmezett modon **pontosan annyi dimenziot generalt mint amennyi valtozonk volt**.

Meg kell adnunk a PCA funkcionak, hany dimenaziot akarunk kinyerni. De hogyan tudjuk eldonteni, mennyi az idealis szamu dimenzio?

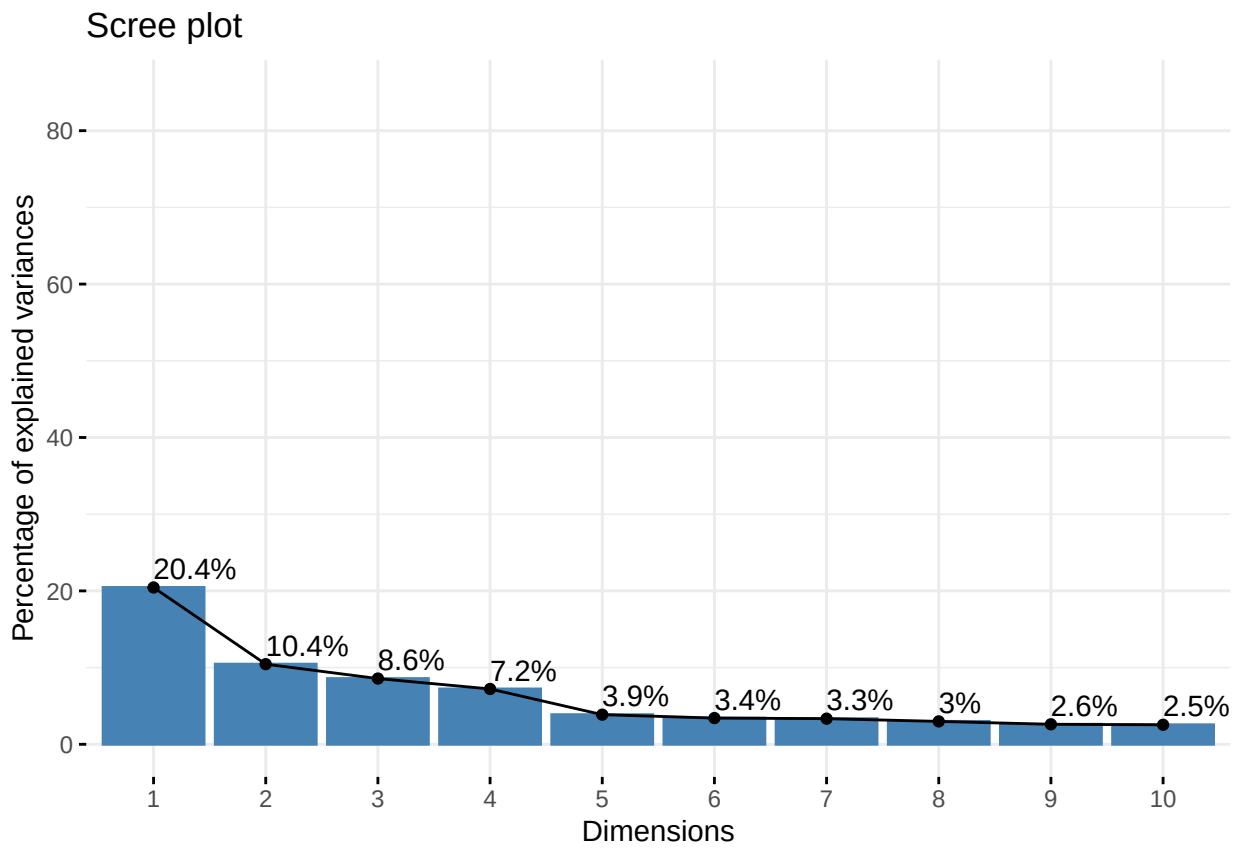
Erre szamos modszer letezik.

1. Scree test

A legismertebb talan a scree-test, ami a megmagyarazott varianciaaranyt abrazolo abra alapjan vegezheto el. Ehhez eloszor a fviz_screeplot() funkcioval abrazolnunk kell az egyes fokomponensek altal megmagyarazott variancia merteket, majd az abra alapjan meg kell allapitanunk, hol van a **"tores" a scree-plotban**, vagyis hol talalhato az a pont, ami utan mar ellaposodik a megmagyarazott varianciaaranyt abrazolo gorbe. **A torespont elotti dimenzional** kell hogy megalljon a dimenzio-extrakcio, vagyis annal a dimenzional, ami meg szignifikansan tobb varianciat kepes megmagyarazni, mint a kesobb kinyerd dimenziok. Ezt a megallasi szabalyt ugy is nevezik hogy a **"konyok kriterium"**, mivel a scree plot egy konyokre emlekeztet, es mi a konyokpontot keressuk a gorbeben.

Ezen az abran ugy tunik, hogy a **negyedik dimenzio** utan mar nem erdemes tovabbmennunk, hiszen az otodik dimenzional megtorik a gorbe es onnantol mar nagyon alacsony a megmagyarazott varianciaarany. Vagyis az idealis dimenzioszam ezen az adaton 4.

```
fviz_screeplot(pca_mod2, addlabels = TRUE, ylim = c(0, 85))
```



The Kaiser-Guttman szabaly

Egy masik jol ismert kriterium, hogy azokat a dimenziokat kell megtartanunk, amelyeknek az **eigenvalue erteke 1-nel magasabb**. Ez azert van, mert **az 1-nel alacsonyabb** eigenvalue azt jelenti, hogy **a dimenzio kevesebb varianciat magyaraz meg mint az eredeti valtozok atlagosan**. A fokomponens elemzes lenyege hogy **hasznos osszefoglalo valtozokat** generaljunk amik **tobb valtozo informaciojat tartalmazzak osszevonva**. Azt pedig nem szeretnenk hogy meg az eredeti valtozoinknal is haszontalanabb valtozokat generaljunk, ezert az atlagos valtozoinknal kisebb varianciat megmagyarazo dimenziokat elutasitjuk.

A peldankban ez az elemzes is azt sugallja, hogy het dimenziot tartsunk meg, hiszen a nyolcadik dimenziohoz tartozo eigenvalue mar 1-nel kisebb.

```
get_eigenvalue(pca_mod2)
```

```
##        eigenvalue variance.percent cumulative.variance.percent
## Dim.1   6.3381690       20.4457065                    20.44571
## Dim.2   3.2384908       10.4467446                    30.89245
## Dim.3   2.6561704        8.5682915                    39.46074
## Dim.4   2.2364863        7.2144721                    46.67521
## Dim.5   1.1975779        3.8631545                    50.53837
## Dim.6   1.0604390        3.4207709                    53.95914
## Dim.7   1.0359658        3.3418250                    57.30097
## Dim.8   0.9255154        2.9855336                    60.28650
## Dim.9   0.8080793        2.6067073                    62.89321
## Dim.10  0.7888049        2.5445319                    65.43774
## Dim.11  0.7699272        2.4836360                    67.92137
## Dim.12  0.7022608        2.2653575                    70.18673
## Dim.13  0.6902394        2.2265788                    72.41331
```

```
## Dim.14  0.6619274        2.1352498                  74.54856
## Dim.15  0.6231722        2.0102328                  76.55879
## Dim.16  0.6107458        1.9701478                  78.52894
## Dim.17  0.5763014        1.8590369                  80.38798
## Dim.18  0.5554613        1.7918106                  82.17979
## Dim.19  0.5418445        1.7478855                  83.92767
## Dim.20  0.5304575        1.7111531                  85.63883
## Dim.21  0.4989770        1.6096031                  87.24843
## Dim.22  0.4874399        1.5723868                  88.82082
## Dim.23  0.4733018        1.5267800                  90.34760
## Dim.24  0.4440585        1.4324469                  91.78004
## Dim.25  0.4242522        1.3685556                  93.14860
## Dim.26  0.4153240        1.3397550                  94.48835
## Dim.27  0.3988724        1.2866853                  95.77504
## Dim.28  0.3638795        1.1738049                  96.94884
## Dim.29  0.3492450        1.1265969                  98.07544
## Dim.30  0.3086315        0.9955854                  99.07103
## Dim.31  0.2879818        0.9289734                 100.00000
```

Parallel elemzes

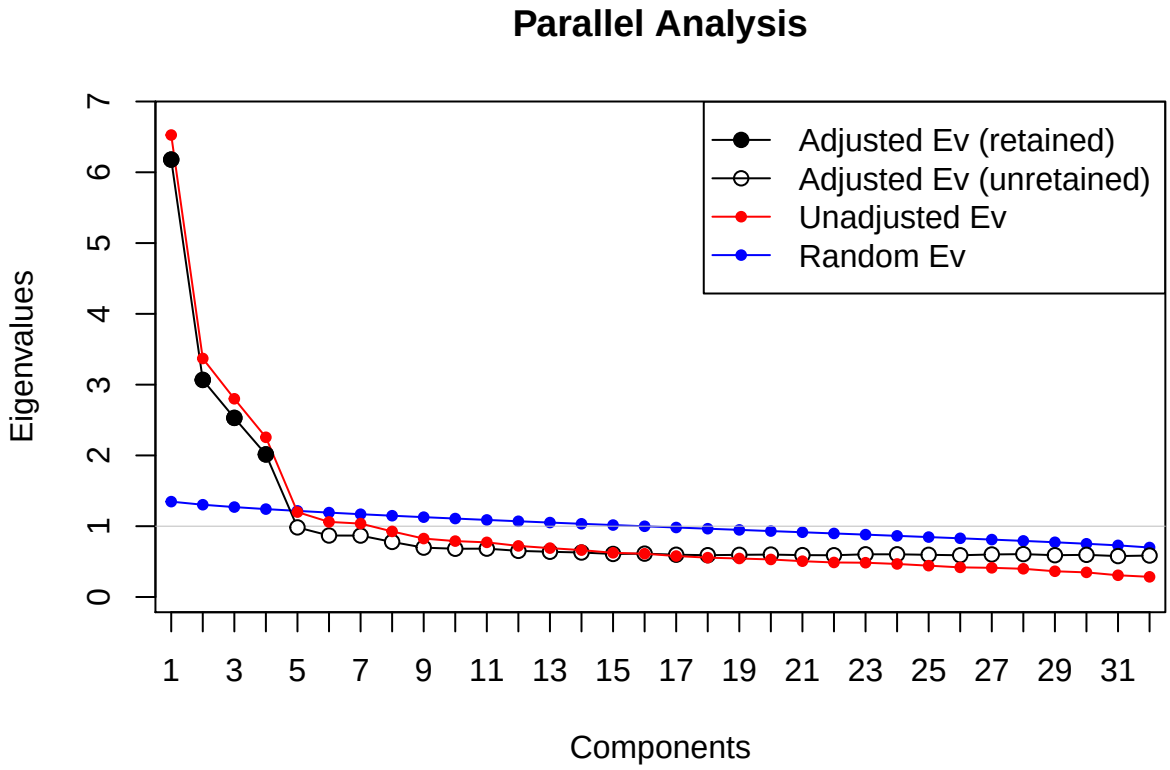
A harmadik (es egyben jelenleg a legelfogadottabb) technika a **parallel elemzes** technika. Ennek a lenyege hogy az eredeti adattablankhoz hasonlo karakterisztikakkal rendelkezo **adatokat generalunk veletlenszeruen**, de ugy, hogy abban a **valtozok ne korrelaljanak egymassal**. Ezt nagyon sokszor megismeteljuk, es ez alapjan a nagy mennyisegu random minta alapjan kiszamoljuk, **mi a veletlenszeruen varhato eigenvalue mintazat**. Ez egyfajta **"null modelkent"** funkcional, amihez hasonlithatjuk a sajat adatainkon kapott eignevalue-kat. Azokat a dimenziokat tartjuk meg, amiknek az **eigenvalue-ja magasabb mint a random mintakban az adott dimenziohoz tartozo null-eigenvalue**.

Ezt a parallel elemzest vegezhetjuk el a **paran()** funkcioval a paran package-bol. Ez a funkcio a null eigenvalue gorbe vizualizalasara is kepes a graph = TRUE prameter beallitasaval, melyet osszehasonlithatunk az adatainkban kapott eigenvalue-val. Az output objektum $Retained komponense megmutatja, az elemzes hany dimenzio megtartasat javasolja.

```
pca_ret = paran(hsq_items_only,
                    graph = TRUE)
```

```
##
## Using eigendecomposition of correlation matrix.
## Computing: 10%  20%  30%  40%  50%  60%  70%  80%  90%  100%
##
##
## Results of Horn's Parallel Analysis for component retention
## 960 iterations, using the mean estimate
##
## ----------------------------------------------------
## Component   Adjusted    Unadjusted    Estimated
##             Eigenvalue  Eigenvalue    Bias
## ----------------------------------------------------
## 1           6.179605    6.526619      0.347013
## 2           3.066565    3.369944      0.303378
## 3           2.529977    2.800201      0.270223
## 4           2.014375    2.256052      0.241677
## ----------------------------------------------------
##
## Adjusted eigenvalues > 1 indicate dimensions to retain.
```

```
## (4 components retained)
```

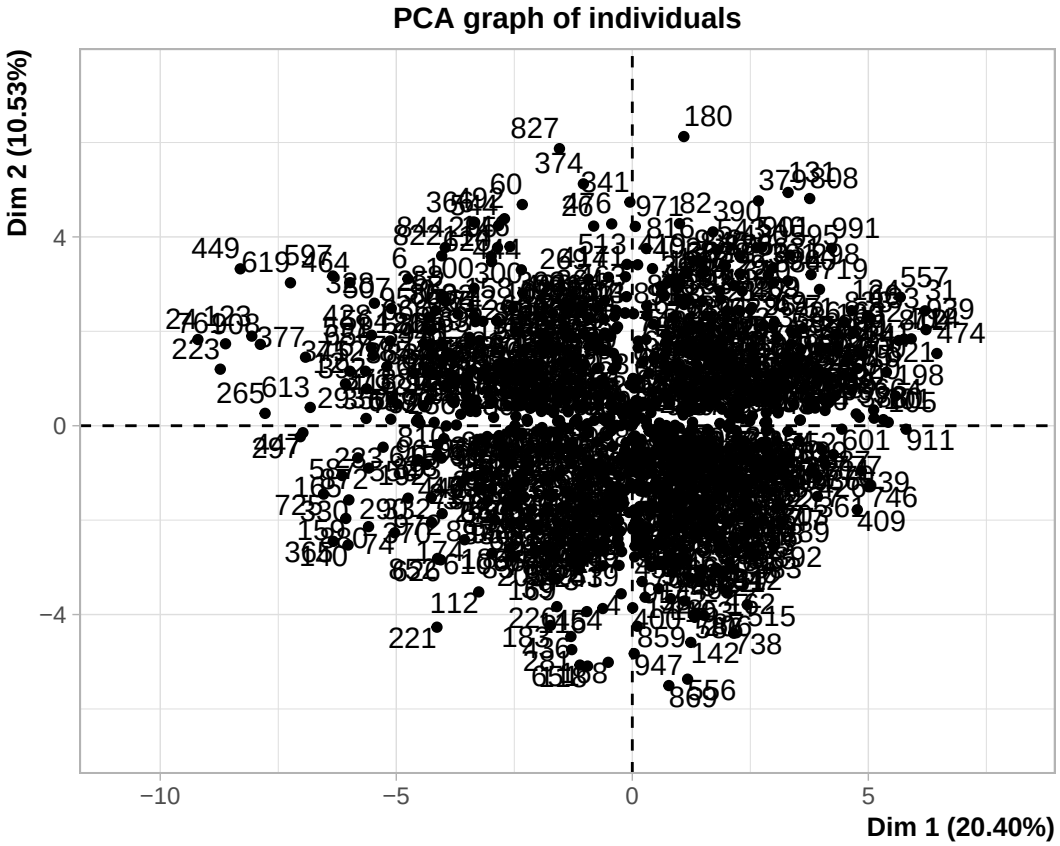


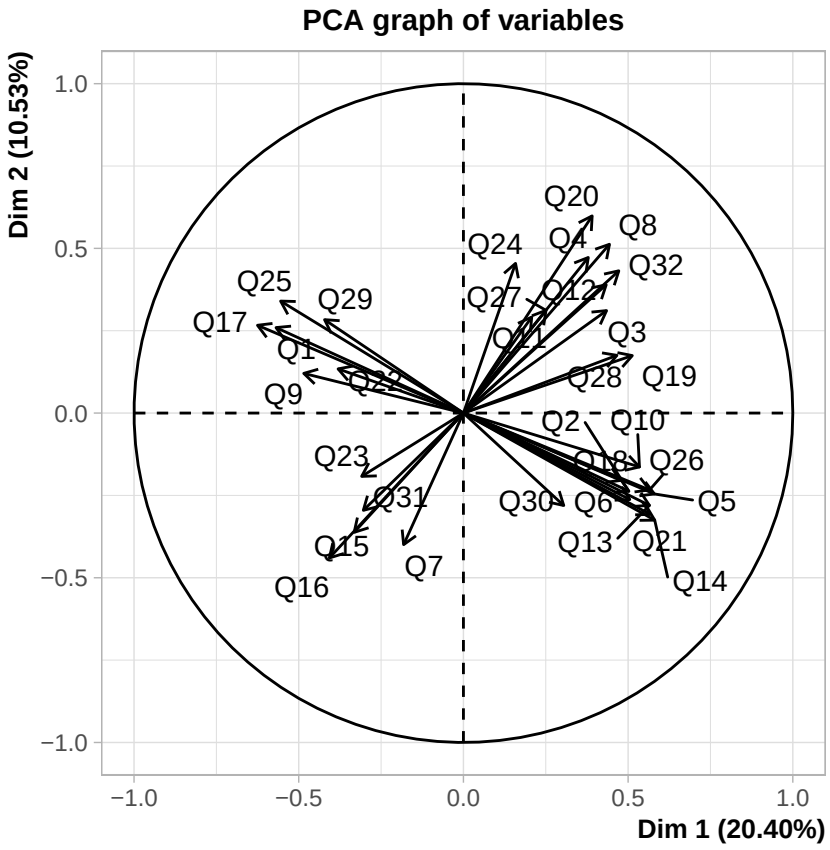

```
pca_ret$Retained
```

```
## [1] 4
```

Amint meghataroztuk az idealis dimenziok szamat, ujra lefuttathatjuk az elemzesunket, de ezuttal mar specifikalmva, mennyi dimenziot szeretnenk, a npc parameter beallitasaval.

```
pca_mod3 <- PCA(hsq_items_only, ncp = 4)
```


PCA graph of individuals
Dim 2 (10.53%)
Dim 1 (20.40%)
4
0
−4
−10
−5
0
5




```
summary(pca_mod3)
```

```
##
## Call:
## PCA(X = hsq_items_only, ncp = 4)
##
##
## Eigenvalues
##                        Dim.1   Dim.2   Dim.3   Dim.4   Dim.5   Dim.6   Dim.7
## Variance               6.527   3.370   2.800   2.256   1.200   1.061   1.037
## % of var.             20.396  10.531   8.751   7.050   3.749   3.315   3.241
## Cumulative % of var.  20.396  30.927  39.677  46.728  50.476  53.791  57.032
##                        Dim.8   Dim.9  Dim.10  Dim.11  Dim.12  Dim.13  Dim.14
## Variance               0.926   0.825   0.789   0.772   0.722   0.690   0.662
## % of var.              2.895   2.579   2.467   2.412   2.255   2.157   2.070
## Cumulative % of var.  59.927  62.506  64.973  67.385  69.640  71.797  73.867
##                       Dim.15  Dim.16  Dim.17  Dim.18  Dim.19  Dim.20  Dim.21
## Variance               0.624   0.611   0.579   0.555   0.544   0.530   0.506
## % of var.              1.951   1.911   1.808   1.736   1.699   1.658   1.581
## Cumulative % of var.  75.818  77.728  79.537  81.273  82.972  84.630  86.211
##                       Dim.22  Dim.23  Dim.24  Dim.25  Dim.26  Dim.27  Dim.28
## Variance               0.488   0.484   0.467   0.442   0.419   0.412   0.398
## % of var.              1.524   1.512   1.459   1.383   1.308   1.287   1.245
## Cumulative % of var.  87.735  89.247  90.706  92.089  93.397  94.684  95.929
##                       Dim.29  Dim.30  Dim.31  Dim.32
## Variance               0.363   0.348   0.307   0.285
```

```
## % of var.              1.135   1.086   0.959   0.890
## Cumulative % of var.  97.064  98.151  99.110 100.000
##
## Individuals (the 10 first)
##         Dist    Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr
## 1    |  3.886 | -0.878  0.012  0.051 | -1.645  0.081  0.179 | -1.133  0.046
## 2    |  4.727 | -2.213  0.076  0.219 | -0.739  0.016  0.024 |  0.932  0.031
## 3    |  3.771 | -0.306  0.001  0.007 | -1.287  0.049  0.116 |  0.197  0.001
## 4    |  4.390 | -0.236  0.001  0.003 | -3.560  0.379  0.658 |  0.309  0.003
## 5    |  3.698 | -1.915  0.057  0.268 |  0.464  0.006  0.016 | -0.026  0.000
## 6    |  7.583 | -4.753  0.349  0.393 |  3.112  0.289  0.168 | -0.131  0.001
## 7    |  4.404 |  1.913  0.056  0.189 | -1.233  0.045  0.078 | -0.842  0.026
## 8    |  4.817 | -1.521  0.036  0.100 | -1.757  0.092  0.133 | -2.019  0.147
## 9    |  9.092 | -3.998  0.247  0.193 |  1.392  0.058  0.023 | -2.109  0.160
## 10   |  5.934 |  2.772  0.119  0.218 | -1.989  0.118  0.112 |  1.588  0.091
##       cos2
## 1    0.085 |
## 2    0.039 |
## 3    0.003 |
## 4    0.005 |
## 5    0.000 |
## 6    0.000 |
## 7    0.037 |
## 8    0.176 |
## 9    0.054 |
## 10   0.072 |
##
## Variables (the 10 first)
##        Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr   cos2
## Q1   | -0.569  4.959  0.324 |  0.259  1.997  0.067 |  0.067  0.158  0.004 |
## Q2   |  0.502  3.863  0.252 | -0.237  1.671  0.056 |  0.123  0.539  0.015 |
## Q3   |  0.435  2.893  0.189 |  0.311  2.870  0.097 | -0.370  4.876  0.137 |
## Q4   |  0.379  2.202  0.144 |  0.472  6.620  0.223 |  0.352  4.418  0.124 |
## Q5   |  0.580  5.151  0.336 | -0.246  1.789  0.060 | -0.064  0.144  0.004 |
## Q6   |  0.505  3.908  0.255 | -0.263  2.045  0.069 |  0.022  0.018  0.000 |
## Q7   | -0.181  0.505  0.033 | -0.399  4.735  0.160 |  0.464  7.677  0.215 |
## Q8   |  0.443  3.012  0.197 |  0.512  7.791  0.263 |  0.382  5.201  0.146 |
## Q9   | -0.485  3.601  0.235 |  0.120  0.431  0.015 |  0.010  0.004  0.000 |
## Q10  |  0.534  4.376  0.286 | -0.163  0.793  0.027 |  0.242  2.084  0.058 |
```

3.5 Fokomponenselemzes eredmenyeinek ertelmezese

3.5.1 a PCA modell opjektum reszei

A modell osszefoglalobol **(model summary)** tovabbi hasznos informaciok olvashatok ki.

Az **Eigenvalues** reszben megtudhatjuk hogy az egyes dimenziok az **adatok teljes varianciajanak hany szazalekat magyarazzak meg (% of var)**, es hogy a legfontosabbtol a legalacsonyabbig egyesevel osszevonva mekkora a tobb dimenzio altal megmagyarazott **osszesitett varianciaarany (Cumulative % of var)**. Vagyis a Dim.3-hoz tartozo % of variance ertek (8.75) azt mutatja, hogy a harmadikkent kinyert dimenzio az adatok varianciajanak 875%-at tudja megmagyarazni onmagaban. Az Dim.3-hoz tartozo dim Cumulative % of var ertek (39.68 pedig azt mutatja, hogy a Dimenzio 3 a Dim.1 es Dim.2-vel egyutt kozosen az adatok varianciajanak (3968%-at kepesek megmagyarazni. Ha csak az eigenvalue-t es a megmagyarazaott varianaciaaranyokat tartalmazo tablazat erdekel minket, ezt kinyerhetjuk ugy hogy csak a pca_mod3$eig komponenst listazzuk ki.

A model summary arrol is tartalmaz informaciot a Variables reszeben, hogy az egyes **valtozok hogyan korrelalnak az egyes uj dimenziokkal** (a Dim.1, Dim.2, Dim.3 ... oszlopokaban), es hogy mekkora a hozzajarulasuk az adott valtozo altal megmagyarazott varianciahoz (a ctr oszlopban). Ez egy nagyon fontos tablazat, mert innen tudjuk leolvasni (az abrak mellett) hogy az egyes valtozokat mely dimenziok (faktorok) irjak le leginkabb. Bovebb informaciot talalunk ha kilistazzuk a pca_mod3$var komponenst.

```
# Get the summary the outputs.
summary(pca_mod3)
```

```
##
## Call:
## PCA(X = hsq_items_only, ncp = 4)
##
##
## Eigenvalues
##                        Dim.1   Dim.2   Dim.3   Dim.4   Dim.5   Dim.6   Dim.7
## Variance               6.527   3.370   2.800   2.256   1.200   1.061   1.037
## % of var.             20.396  10.531   8.751   7.050   3.749   3.315   3.241
## Cumulative % of var.  20.396  30.927  39.677  46.728  50.476  53.791  57.032
##                        Dim.8   Dim.9  Dim.10  Dim.11  Dim.12  Dim.13  Dim.14
## Variance               0.926   0.825   0.789   0.772   0.722   0.690   0.662
## % of var.              2.895   2.579   2.467   2.412   2.255   2.157   2.070
## Cumulative % of var.  59.927  62.506  64.973  67.385  69.640  71.797  73.867
##                       Dim.15  Dim.16  Dim.17  Dim.18  Dim.19  Dim.20  Dim.21
## Variance               0.624   0.611   0.579   0.555   0.544   0.530   0.506
## % of var.              1.951   1.911   1.808   1.736   1.699   1.658   1.581
## Cumulative % of var.  75.818  77.728  79.537  81.273  82.972  84.630  86.211
##                       Dim.22  Dim.23  Dim.24  Dim.25  Dim.26  Dim.27  Dim.28
## Variance               0.488   0.484   0.467   0.442   0.419   0.412   0.398
## % of var.              1.524   1.512   1.459   1.383   1.308   1.287   1.245
## Cumulative % of var.  87.735  89.247  90.706  92.089  93.397  94.684  95.929
##                       Dim.29  Dim.30  Dim.31  Dim.32
## Variance               0.363   0.348   0.307   0.285
## % of var.              1.135   1.086   0.959   0.890
## Cumulative % of var.  97.064  98.151  99.110 100.000
##
## Individuals (the 10 first)
##         Dist    Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr
## 1    |  3.886 | -0.878  0.012  0.051 | -1.645  0.081  0.179 | -1.133  0.046
## 2    |  4.727 | -2.213  0.076  0.219 | -0.739  0.016  0.024 |  0.932  0.031
## 3    |  3.771 | -0.306  0.001  0.007 | -1.287  0.049  0.116 |  0.197  0.001
## 4    |  4.390 | -0.236  0.001  0.003 | -3.560  0.379  0.658 |  0.309  0.003
## 5    |  3.698 | -1.915  0.057  0.268 |  0.464  0.006  0.016 | -0.026  0.000
## 6    |  7.583 | -4.753  0.349  0.393 |  3.112  0.289  0.168 | -0.131  0.001
## 7    |  4.404 |  1.913  0.056  0.189 | -1.233  0.045  0.078 | -0.842  0.026
## 8    |  4.817 | -1.521  0.036  0.100 | -1.757  0.092  0.133 | -2.019  0.147
## 9    |  9.092 | -3.998  0.247  0.193 |  1.392  0.058  0.023 | -2.109  0.160
## 10   |  5.934 |  2.772  0.119  0.218 | -1.989  0.118  0.112 |  1.588  0.091
##       cos2
## 1    0.085 |
## 2    0.039 |
## 3    0.003 |
## 4    0.005 |
## 5    0.000 |
## 6    0.000 |
```

```
## 7    0.037 |
## 8    0.176 |
## 9    0.054 |
## 10   0.072 |
##
## Variables (the 10 first)
##        Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr   cos2
## Q1   | -0.569  4.959  0.324 |  0.259  1.997  0.067 |  0.067  0.158  0.004 |
## Q2   |  0.502  3.863  0.252 | -0.237  1.671  0.056 |  0.123  0.539  0.015 |
## Q3   |  0.435  2.893  0.189 |  0.311  2.870  0.097 | -0.370  4.876  0.137 |
## Q4   |  0.379  2.202  0.144 |  0.472  6.620  0.223 |  0.352  4.418  0.124 |
## Q5   |  0.580  5.151  0.336 | -0.246  1.789  0.060 | -0.064  0.144  0.004 |
## Q6   |  0.505  3.908  0.255 | -0.263  2.045  0.069 |  0.022  0.018  0.000 |
## Q7   | -0.181  0.505  0.033 | -0.399  4.735  0.160 |  0.464  7.677  0.215 |
## Q8   |  0.443  3.012  0.197 |  0.512  7.791  0.263 |  0.382  5.201  0.146 |
## Q9   | -0.485  3.601  0.235 |  0.120  0.431  0.015 |  0.010  0.004  0.000 |
## Q10  |  0.534  4.376  0.286 | -0.163  0.793  0.027 |  0.242  2.084  0.058 |
```

```
pca_mod3$eig
```

```
##         eigenvalue percentage of variance cumulative percentage of variance
## comp 1   6.5266194             20.3956857                           20.39569
## comp 2   3.3699443             10.5310758                           30.92676
## comp 3   2.8002011              8.7506285                           39.67739
## comp 4   2.2560527              7.0501646                           46.72755
## comp 5   1.1995880              3.7487124                           50.47627
## comp 6   1.0608149              3.3150464                           53.79131
## comp 7   1.0370806              3.2408769                           57.03219
## comp 8   0.9263495              2.8948422                           59.92703
## comp 9   0.8252450              2.5788906                           62.50592
## comp 10  0.7894123              2.4669134                           64.97284
## comp 11  0.7718522              2.4120381                           67.38487
## comp 12  0.7216551              2.2551722                           69.64005
## comp 13  0.6903207              2.1572520                           71.79730
## comp 14  0.6622550              2.0695470                           73.86685
## comp 15  0.6242740              1.9508561                           75.81770
## comp 16  0.6114501              1.9107815                           77.72848
## comp 17  0.5786557              1.8082991                           79.53678
## comp 18  0.5554699              1.7358435                           81.27263
## comp 19  0.5437363              1.6991759                           82.97180
## comp 20  0.5304791              1.6577472                           84.62955
## comp 21  0.5059257              1.5810178                           86.21057
## comp 22  0.4877984              1.5243699                           87.73494
## comp 23  0.4839852              1.5124538                           89.24739
## comp 24  0.4668567              1.4589273                           90.70632
## comp 25  0.4424163              1.3825510                           92.08887
## comp 26  0.4185422              1.3079445                           93.39681
## comp 27  0.4118812              1.2871287                           94.68394
## comp 28  0.3984002              1.2450006                           95.92894
## comp 29  0.3633398              1.1354370                           97.06438
## comp 30  0.3476741              1.0864816                           98.15086
## comp 31  0.3069190              0.9591218                           99.10998
## comp 32  0.2848053              0.8900167                          100.00000
```

```
pca_mod3$var
```

```
## $coord
##          Dim.1      Dim.2       Dim.3       Dim.4
## Q1  -0.5688915  0.2594476  0.06654987  0.34960522
## Q2   0.5021178 -0.2373143  0.12284157  0.38417814
## Q3   0.4345612  0.3110200 -0.36950854  0.02163271
## Q4   0.3790587  0.4723304  0.35171209 -0.08316700
## Q5   0.5798081 -0.2455561 -0.06358979 -0.27455182
## Q6   0.5050051 -0.2625465  0.02225282  0.18236620
## Q7  -0.1814692 -0.3994541  0.46364367 -0.20124909
## Q8   0.4433948  0.5123844  0.38163200 -0.10676883
## Q9  -0.4847889  0.1204558  0.01028084  0.29329403
## Q10  0.5344008 -0.1634825  0.24158276  0.49608475
## Q11  0.2073377  0.2914197 -0.45068030  0.25594805
## Q12  0.4334654  0.3894721  0.38434674 -0.17975417
## Q13  0.5646683 -0.2802503 -0.09285473 -0.27570231
## Q14  0.5806946 -0.3250714  0.12846727  0.33887960
## Q15 -0.3316783 -0.3620736  0.52834991 -0.08033977
## Q16 -0.4069220 -0.4398788 -0.20043300  0.13094951
## Q17 -0.6251763  0.2665194  0.09778869  0.39147217
## Q18  0.4781092 -0.2045136  0.22455628  0.53160685
## Q19  0.5127333  0.1744786 -0.29166246  0.06097239
## Q20  0.3907534  0.5985140  0.31822294 -0.07475584
## Q21  0.5672222 -0.3077202  0.06856781 -0.36662308
## Q22 -0.3804674  0.1341635 -0.01415713 -0.25322693
## Q23 -0.3088811 -0.1922108  0.44485076 -0.07375981
## Q24  0.1581873  0.4544262  0.34650080  0.05169798
## Q25 -0.5549809  0.3399577  0.09318884  0.33462739
## Q26  0.5650082 -0.2333713  0.19231956  0.38389427
## Q27  0.2506611  0.3119157 -0.44034085  0.13298736
## Q28  0.4660091  0.1752175  0.03172224  0.06895569
## Q29 -0.4215349  0.2835606  0.30889584  0.26818215
## Q30  0.3045698 -0.2806700  0.04587713  0.38421190
## Q31 -0.3038096 -0.2951999  0.60015283 -0.07033962
## Q32  0.4717226  0.4318453  0.36617067 -0.12580811
##
## $cor
##          Dim.1      Dim.2       Dim.3       Dim.4
## Q1  -0.5688915  0.2594476  0.06654987  0.34960522
## Q2   0.5021178 -0.2373143  0.12284157  0.38417814
## Q3   0.4345612  0.3110200 -0.36950854  0.02163271
## Q4   0.3790587  0.4723304  0.35171209 -0.08316700
## Q5   0.5798081 -0.2455561 -0.06358979 -0.27455182
## Q6   0.5050051 -0.2625465  0.02225282  0.18236620
## Q7  -0.1814692 -0.3994541  0.46364367 -0.20124909
## Q8   0.4433948  0.5123844  0.38163200 -0.10676883
## Q9  -0.4847889  0.1204558  0.01028084  0.29329403
## Q10  0.5344008 -0.1634825  0.24158276  0.49608475
## Q11  0.2073377  0.2914197 -0.45068030  0.25594805
## Q12  0.4334654  0.3894721  0.38434674 -0.17975417
## Q13  0.5646683 -0.2802503 -0.09285473 -0.27570231
## Q14  0.5806946 -0.3250714  0.12846727  0.33887960
## Q15 -0.3316783 -0.3620736  0.52834991 -0.08033977
```

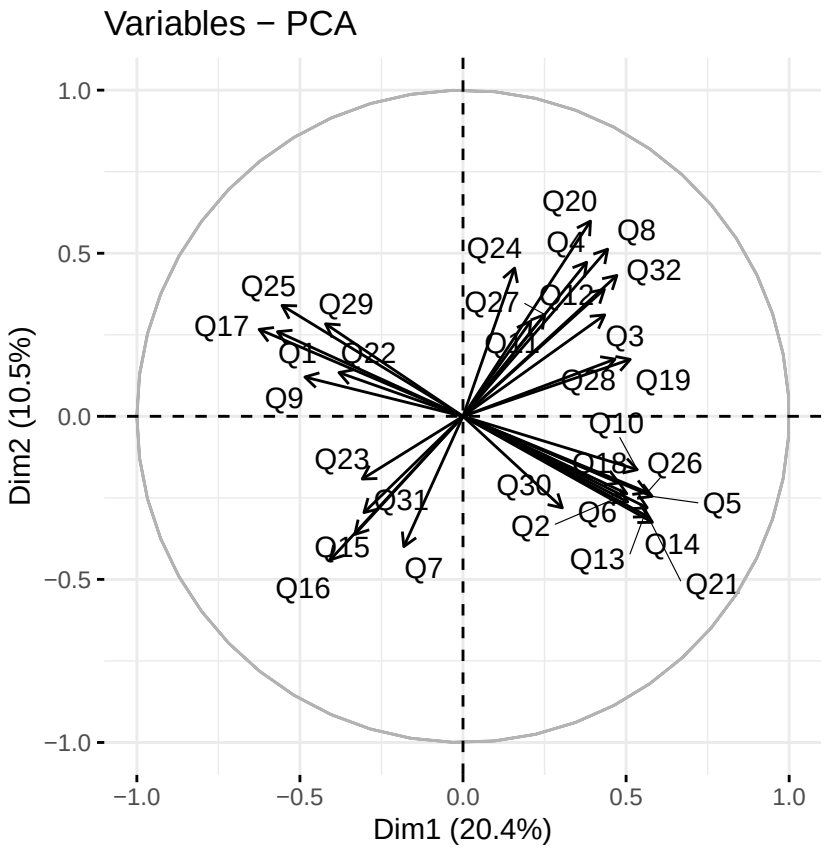
```
## Q16 -0.4069220 -0.4398788 -0.20043300  0.13094951
## Q17 -0.6251763  0.2665194  0.09778869  0.39147217
## Q18  0.4781092 -0.2045136  0.22455628  0.53160685
## Q19  0.5127333  0.1744786 -0.29166246  0.06097239
## Q20  0.3907534  0.5985140  0.31822294 -0.07475584
## Q21  0.5672222 -0.3077202  0.06856781 -0.36662308
## Q22 -0.3804674  0.1341635 -0.01415713 -0.25322693
## Q23 -0.3088811 -0.1922108  0.44485076 -0.07375981
## Q24  0.1581873  0.4544262  0.34650080  0.05169798
## Q25 -0.5549809  0.3399577  0.09318884  0.33462739
## Q26  0.5650082 -0.2333713  0.19231956  0.38389427
## Q27  0.2506611  0.3119157 -0.44034085  0.13298736
## Q28  0.4660091  0.1752175  0.03172224  0.06895569
## Q29 -0.4215349  0.2835606  0.30889584  0.26818215
## Q30  0.3045698 -0.2806700  0.04587713  0.38421190
## Q31 -0.3038096 -0.2951999  0.60015283 -0.07033962
## Q32  0.4717226  0.4318453  0.36617067 -0.12580811
##
## $cos2
##          Dim.1      Dim.2        Dim.3        Dim.4
## Q1  0.32363753 0.06731304 0.0044288853 0.1222238112
## Q2  0.25212231 0.05631808 0.0150900513 0.1475928452
## Q3  0.18884342 0.09673342 0.1365365582 0.0004679743
## Q4  0.14368548 0.22309599 0.1237013940 0.0069167499
## Q5  0.33617747 0.06029780 0.0040436615 0.0753787019
## Q6  0.25503012 0.06893065 0.0004951881 0.0332574304
## Q7  0.03293105 0.15956359 0.2149654501 0.0405011948
## Q8  0.19659894 0.26253776 0.1456429798 0.0113995829
## Q9  0.23502023 0.01450960 0.0001056956 0.0860213884
## Q10 0.28558417 0.02672654 0.0583622301 0.2461000811
## Q11 0.04298891 0.08492543 0.2031127326 0.0655094020
## Q12 0.18789226 0.15168850 0.1477224158 0.0323115623
## Q13 0.31885026 0.07854023 0.0086220015 0.0760117628
## Q14 0.33720617 0.10567144 0.0165038392 0.1148393858
## Q15 0.11001049 0.13109731 0.2791536322 0.0064544782
## Q16 0.16558554 0.19349334 0.0401733860 0.0171477733
## Q17 0.39084539 0.07103259 0.0095626284 0.1532504575
## Q18 0.22858840 0.04182580 0.0504255208 0.2826058383
## Q19 0.26289547 0.03044277 0.0850669913 0.0037176325
## Q20 0.15268822 0.35821901 0.1012658418 0.0055884355
## Q21 0.32174097 0.09469174 0.0047015447 0.1344124851
## Q22 0.14475541 0.01799985 0.0002004244 0.0641238788
## Q23 0.09540756 0.03694500 0.1978921964 0.0054405091
## Q24 0.02502321 0.20650317 0.1200628061 0.0026726810
## Q25 0.30800382 0.11557126 0.0086841601 0.1119754922
## Q26 0.31923424 0.05446216 0.0369868149 0.1473748141
## Q27 0.06283097 0.09729141 0.1939000666 0.0176856369
## Q28 0.21716450 0.03070119 0.0010063002 0.0047548877
## Q29 0.17769165 0.08040662 0.0954166400 0.0719216657
## Q30 0.09276277 0.07877568 0.0021047113 0.1476187847
## Q31 0.09230027 0.08714298 0.3601834227 0.0049476626
## Q32 0.22252219 0.18649034 0.1340809568 0.0158276803
##
## $contrib
```

```
##         Dim.1      Dim.2        Dim.3       Dim.4
## Q1  4.9587314  1.9974527  0.158163113  5.41759565
## Q2  3.8629848  1.6711872  0.538891696  6.54208332
## Q3  2.8934339  2.8704752  4.875955405  0.02074306
## Q4  2.2015299  6.6201684  4.417589606  0.30658637
## Q5  5.1508667  1.7892817  0.144406109  3.34117652
## Q6  3.9075378  2.0454536  0.017684020  1.47414246
## Q7  0.5045652  4.7349029  7.676786071  1.79522382
## Q8  3.0122629  7.7905668  5.201161386  0.50528887
## Q9  3.6009489  0.4305590  0.003774572  3.81291579
## Q10 4.3756829  0.7930854  2.084215649 10.90843688
## Q11 0.6586704  2.5200841  7.253505134  2.90371776
## Q12 2.8788603  4.5012169  5.275421624  1.43221667
## Q13 4.8853816  2.3306092  0.307906509  3.36923707
## Q14 5.1666284  3.1357028  0.589380493  5.09027947
## Q15 1.6855662  3.8901922  9.969056487  0.28609608
## Q16 2.5370798  5.7417372  1.434660733  0.76007859
## Q17 5.9884814  2.1078270  0.341497912  6.79285815
## Q18 3.5024013  1.2411422  1.800782103 12.52656210
## Q19 4.0280496  0.9033613  3.037888616  0.16478483
## Q20 2.3394687 10.6298201  3.616377438  0.24770856
## Q21 4.9296726  2.8098905  0.167900250  5.95786114
## Q22 2.2179233  0.5341289  0.007157501  2.84230416
## Q23 1.4618221  1.0963089  7.067070807  0.24115169
## Q24 0.3834023  6.1277920  4.287649373  0.11846714
## Q25 4.7191938  3.4294709  0.310126298  4.96333680
## Q26 4.8912649  1.6161145  1.320862795  6.53241905
## Q27 0.9626878  2.8870333  6.924504981  0.78391951
## Q28 3.3273657  0.9110296  0.035936712  0.21076138
## Q29 2.7225680  2.3859925  3.407492377  3.18794267
## Q30 1.4212989  2.3375958  0.075162861  6.54323309
## Q31 1.4142126  2.5858878 12.862769718  0.21930617
## Q32 3.4094556  5.5339294  4.788261651  0.70156519
```

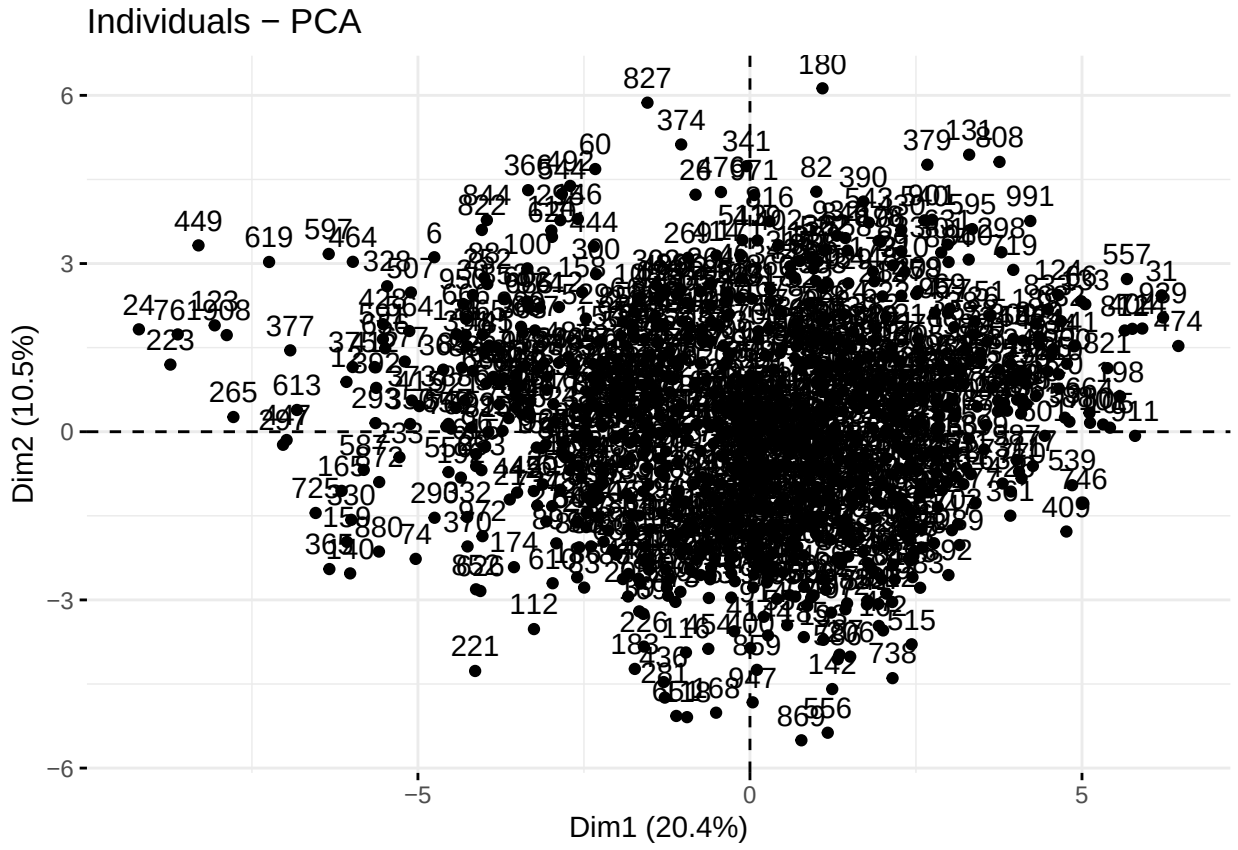
3.5.2 Az eredmenyek vizualizalasa

Az eredmenyek vizualizalasa segithet a komponensek ertelmezeseben. A fviz_pca_var() es a fviz_pca_ind() segitsegevel reprodukalhatjuk a PCA funcio altal eredetileg general abrakat. Sot, a kettot ossze is vonhatjuk a fviz_pca_biplot() funkcioval. Igy egyszerre lathatjuk hogy a ket legfontosabb dimenzio menten hol helyezkednek el az egyes megfigyelesek (az autok), es hogy a dimenziok foleg mely valtozokat reprezentaljak. (A repel = T parameterbeallitas arra jo hogy a feliratok ne fedjek egymast hanem elcsusztatva szerepeljenek az abran ha tul kozel lennenek egymashoz)

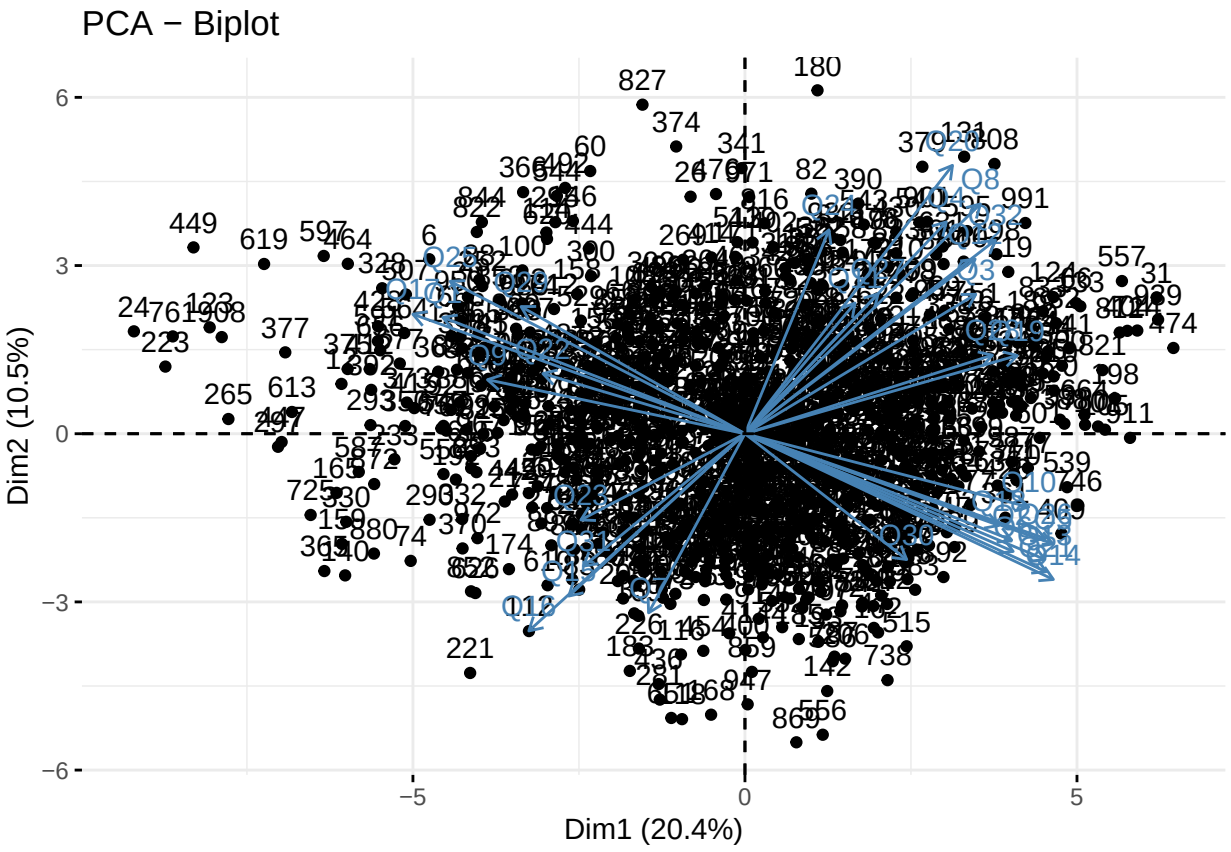
```
fviz_pca_var(pca_mod3, repel = T)
```




```
fviz_pca_ind(pca_mod3)
```



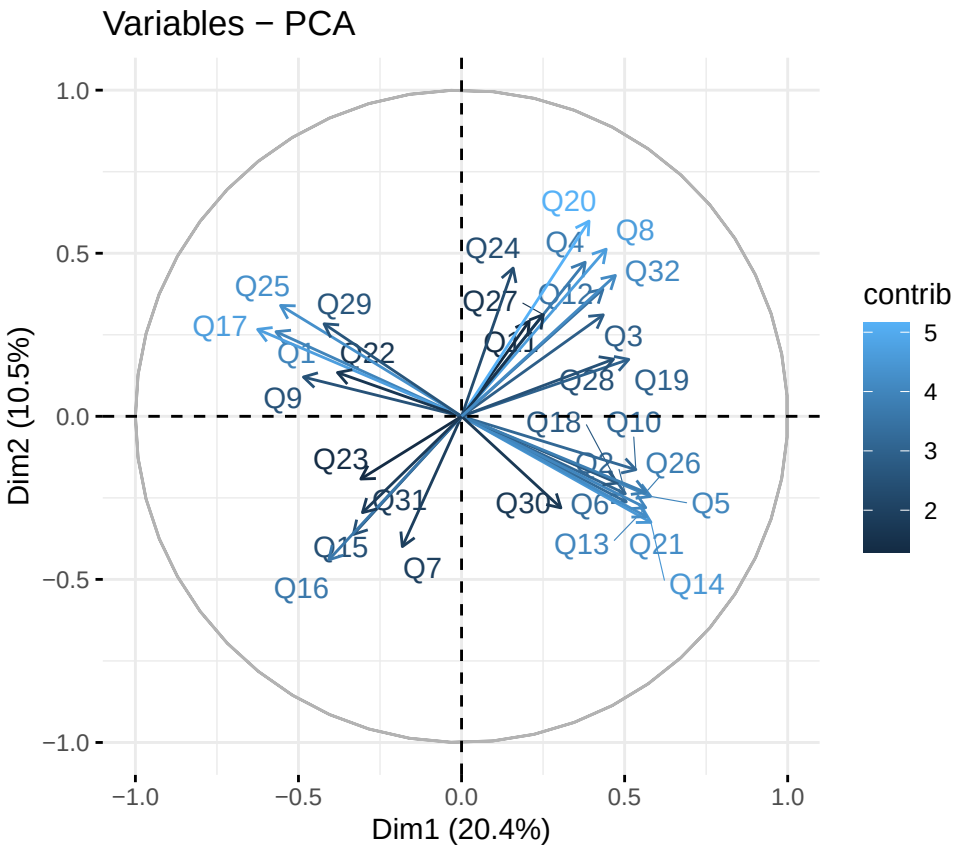

```
fviz_pca_biplot(pca_mod3)
```



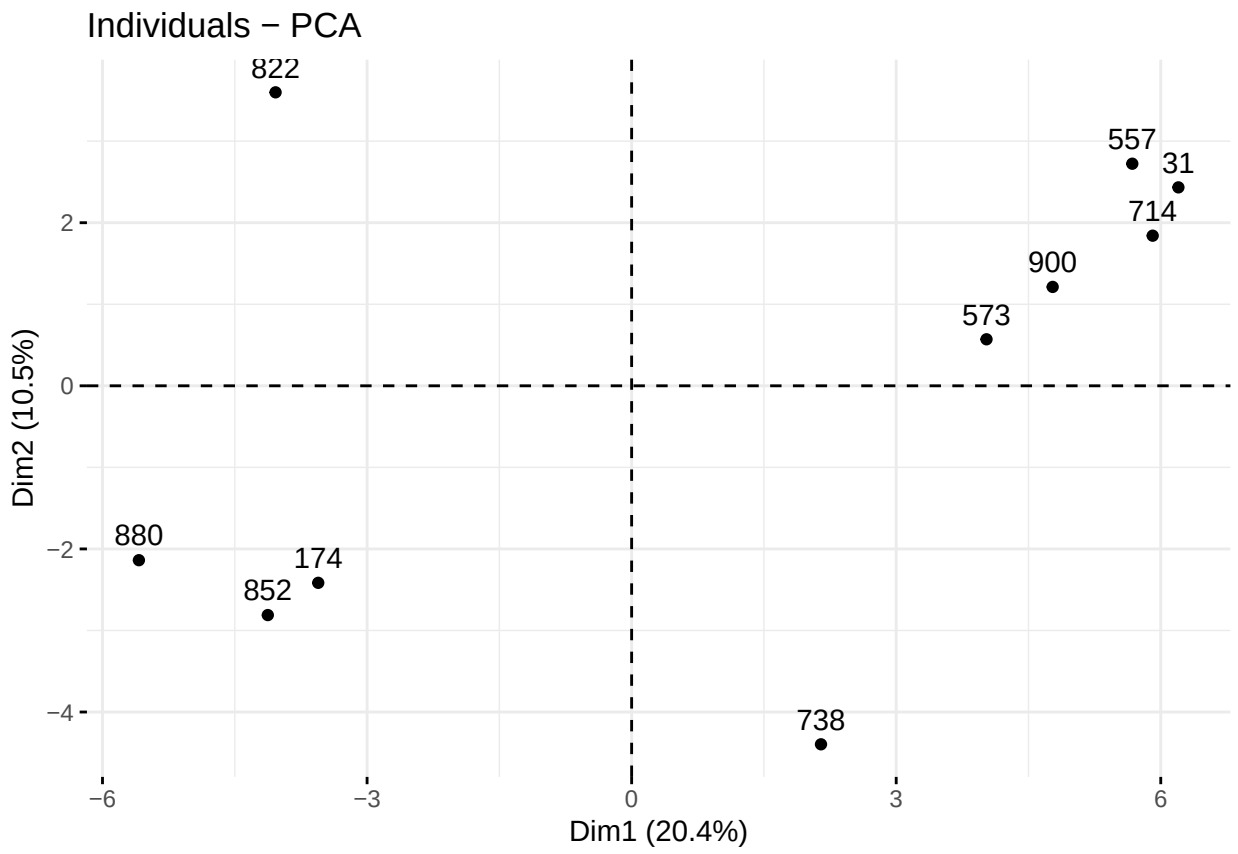
Az abrakat tovabb tunningolhatjuk azzal, hogy abrazoljuk rajtuk az egyes valtozok vagy megfigyelesek hozzajarulasat (contribution) az abrazolt dimenziohoz a , col.var = “contrib” es , col.ind = “contrib” parametereken keresztul.

Azt is megtehetjuk, hogy a select.ind = parameteren keresztul hogy csak bizonyos megfigyeleseket teszunk az abrara.Pl. cos^2 ertek azt mutatja, hogy az adott megfigyeles vagy valtozo menyire jol reprezentalt az adott dimenzio altal. A select.ind = list(cos2 = 10) parameter beallitasaval meghatarozhatjuk, hogy csak az a 10 megfigyeles szerepeljen az abran, akiknek a ket diemnziora vonatkozo cos^2 osszege a legmagasabb. Vagyis a ket dimenzio altal leirt dimenzioter 10 legreprezentativabb megfigyelese.

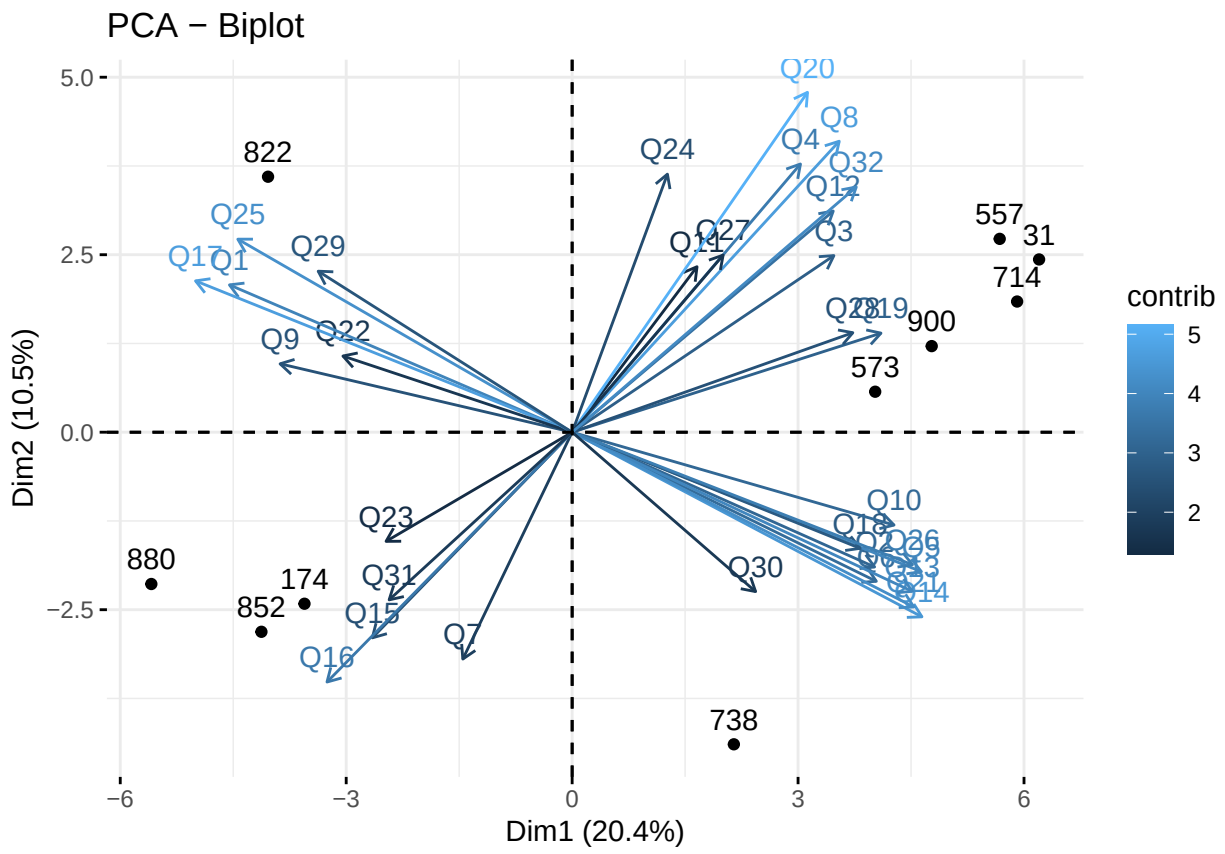
```
fviz_pca_var(pca_mod3, repel = T, col.var = "contrib")
```




```
fviz_pca_ind(pca_mod3, select.ind = list(cos2 = 10))
```

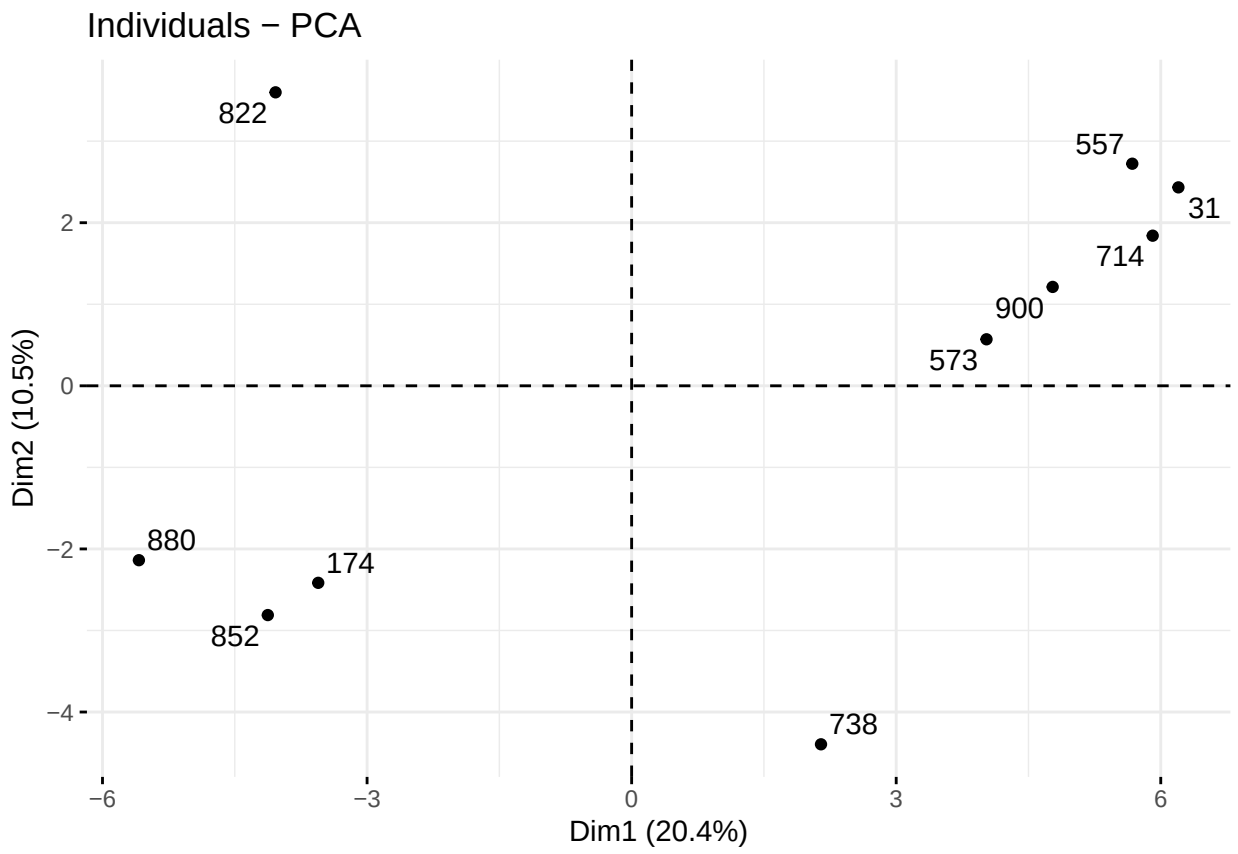



```
fviz_pca_biplot(pca_mod3, select.ind = list(cos2 = 10), col.var = "contrib")
```



Ez az abra azt mutatja hogy a magas Dim.1, kozepes Dim.2 legtipikusabb tagjai pl. az 573-as es a 900-as megfigyelesek, az alacsony Dim.2. kozepes Dim.1 legtipikusabb tagja talan a 738-as szemely, es az alacsony Dim.1. es magas Dim.2. legtipikusabb tagja a 822-es megfigyeles. Ez fontos lehet a dimenziok ertemezeseben.

```
fviz_pca_ind(pca_mod3, select.ind = list(cos2 = 10), repel = T)
```



Egy masik fontos abratipus az egyes dimenziok ertelmezesenek elosegitesehez a **fviz_contrib() altal generalt barchart**, ami az egyes valtozok egyes dimenziokhoz valo hozzajarulasat mutatja meg.

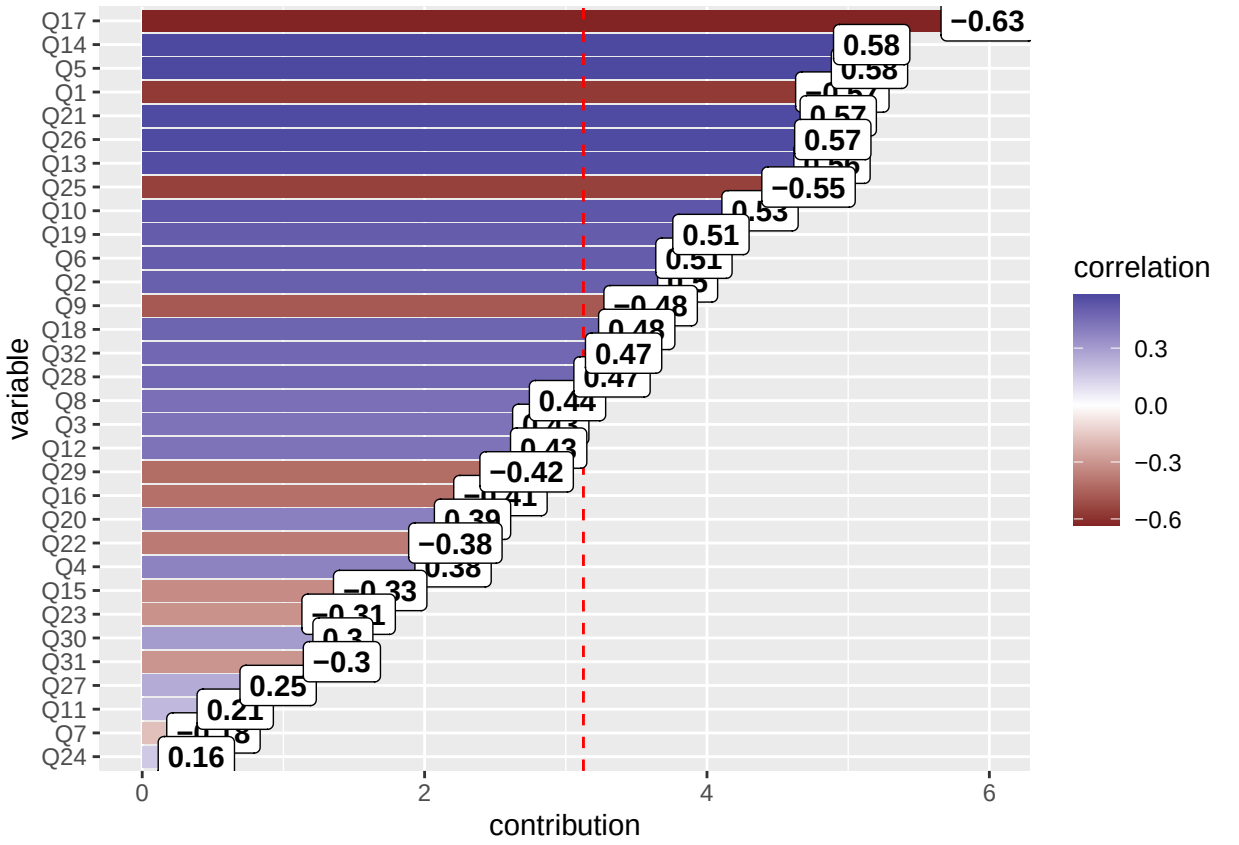
Az **axes = parameterrel allithatjuk be, melyik dimenziora** vagyunk kivancsiak. **A piros szaggatott vonal** azt mutatja, hogy mi lenne az **elvart hozzajarulas szazaleka** abban az esetben ha minden valtozo azonos mertekben jarulna hozza a dimenzio megmagyarazasahoz.

Ez az abra akkor lenne igazan informativ ha a korrelacio merteke es iranya is egyertelmu lenne rola. Ezt onmagaban nem tartalmazza a fviz_contrib() funkcio, ezert a **fviz_loadnings_with_cor() sajat funkcio** hasznalataval helyettesitjuk, melyen az oszlopok a korrelacio szerint vannak szinezve es a korrelacio feliratkent is szerepel az abran.

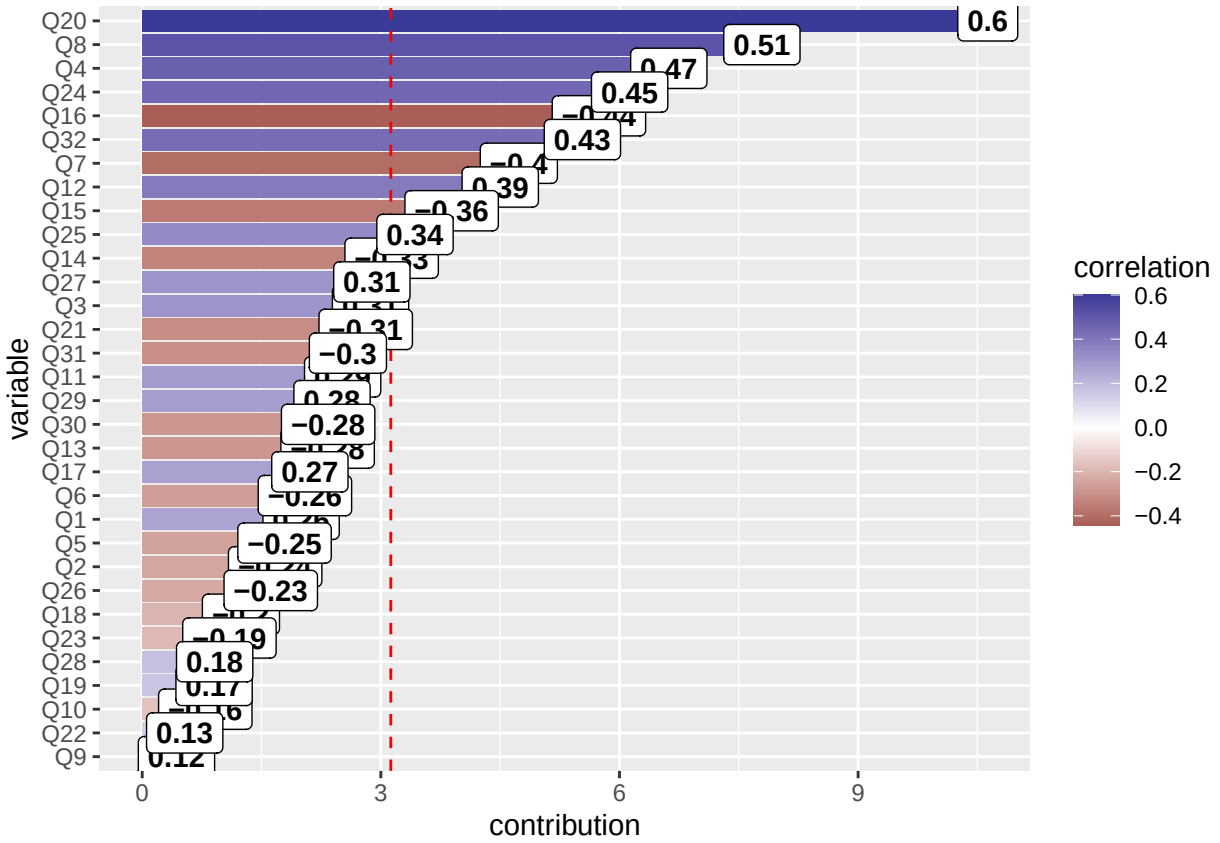
Ezek az abra azt mutatjak, hogy a Dim.1-hez elsosorban Q14, Q17, Q5, Q1 valtozok jarulnak hozza, mig a Dim.2-hoz elsosorban a Q20, Q8, Q4 es Q24 valtozok jarulnak hozza, mig a Dim.3-hoz a Q31, Q15, Q7.

Ezek alapjan az abrak alapjan, es a reprezentativ esetek abraja alapjan mit gondolsz, hogyan nevezhetnek el az egyes dimenziokat?

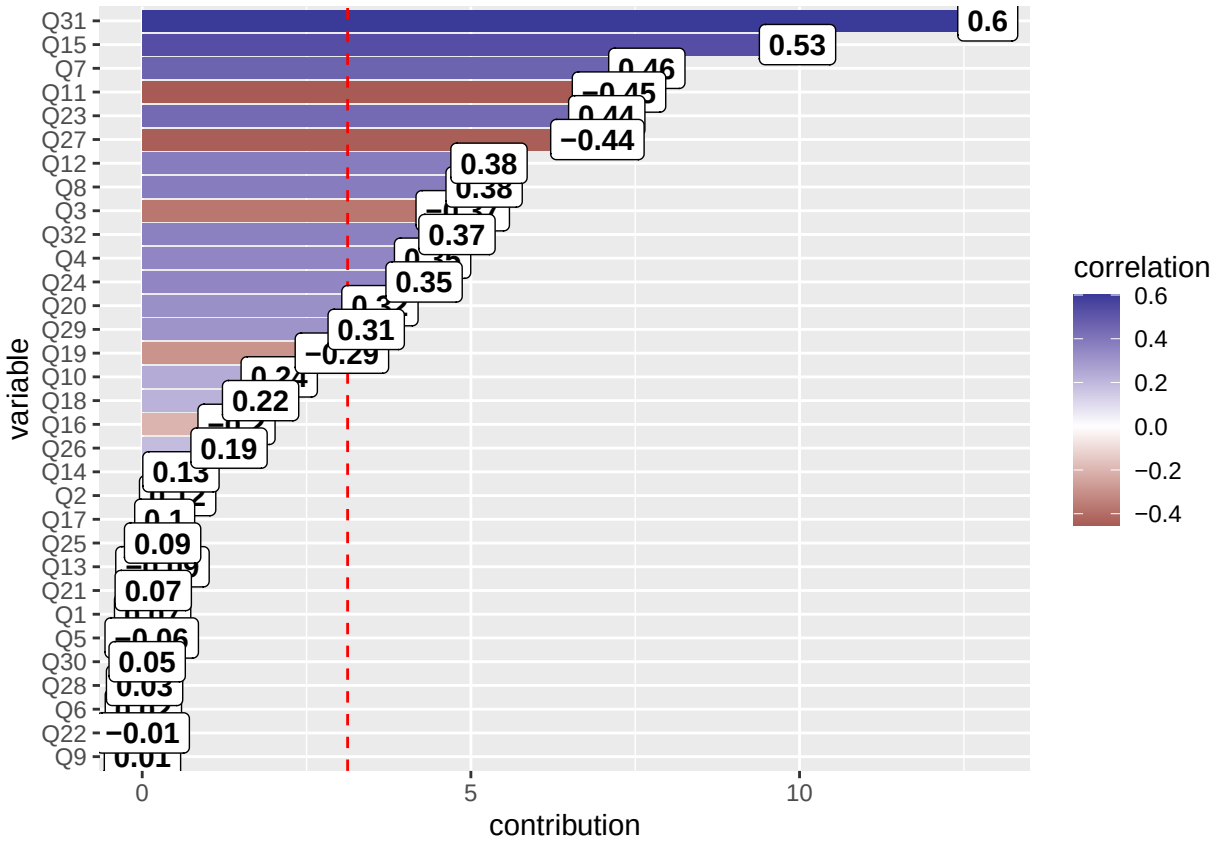
```
# original functions in factoextra
# fviz_contrib(pca_mod3, choice = "var", axes = 1)
# fviz_contrib(pca_mod3, choice = "var", axes = 2)
# fviz_contrib(pca_mod3, choice = "var", axes = 3)

# using custom function for correlation color gradient
fviz_loadnings_with_cor(mod = pca_mod3, axes = 1)
```




```
fviz_loadnings_with_cor(mod = pca_mod3, axes = 2)
```



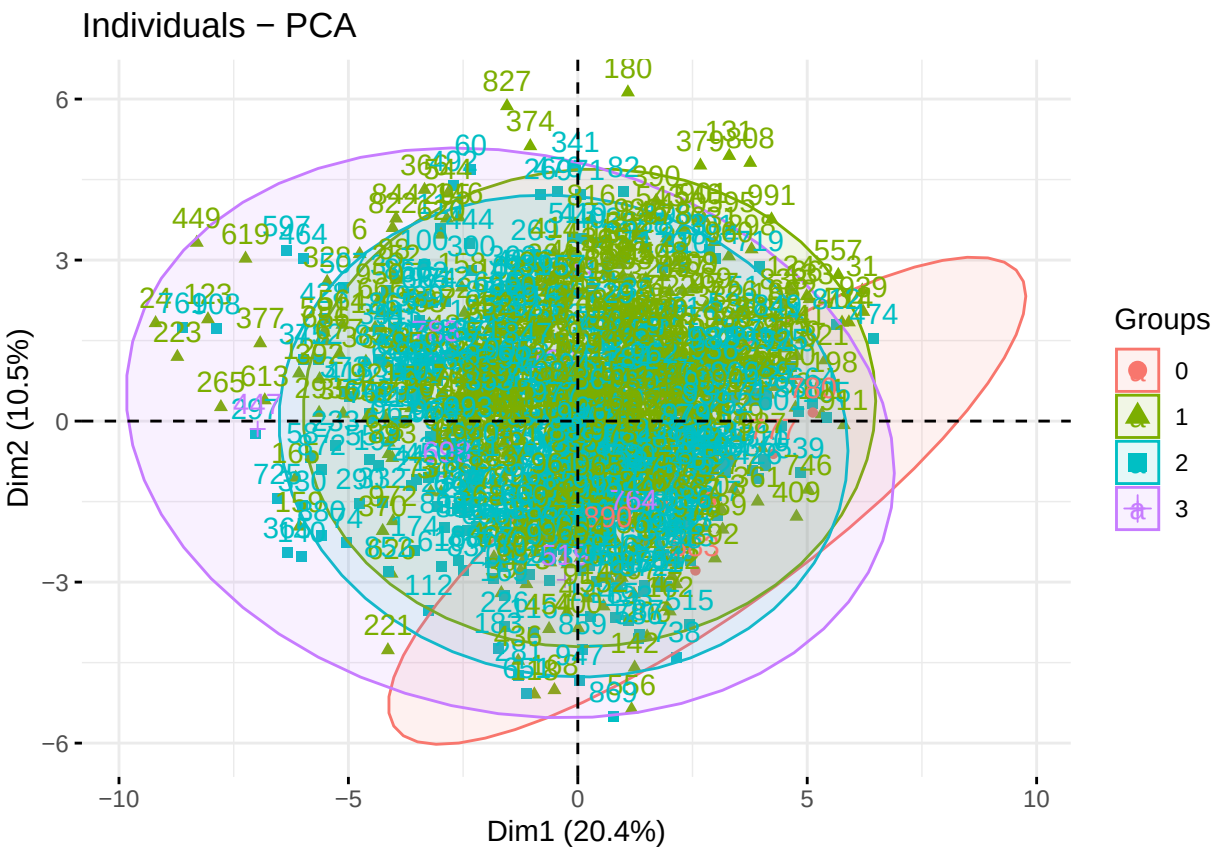

```
fviz_loadnings_with_cor(mod = pca_mod3, axes = 3)
```



A vizualizaciot arra is hasznalhatjuk, hogy csoportositsuk a megfigyeleseket a dimenziokon mutatott ertekuk alapjan. Ezt az addEllipses = T parameterrel adhatjuk meg.

Hogyan jellemezned az egyes elipszisekben talalhato embereket az alapjan, hogy az 1. es 2. dimenzion milyen ertekekt vesznek fel?

```
fviz_pca_ind(pca_mod3,
             label = "ind",
    habillage=factor(hsq$gender),
    addEllipses = T)
```



4 Bevezetes a feltaro faktorelemzesbe (Exploratory Factor Analysis - EFA)

A faktorelemzes egy masik dimenzioredukcios technika ami hasonlit a fokomponenselemzeshez. A ketto kozott fontos kulonbseg hogy a faktorelemzest akkor hasznaljuk, ha feltetelezzuk hogy a sok valtozonk hattereben kozos okok, ugynevezett latens faktorok allnak, es ez okozza, hogy a megfigyelt valtozoink korrelalnak egymassal.

Amikor egy feltaro faktorelemzesi (EFA) modellt epitunk, nem probaljuk megmagyarazni a teljes varianciat az adatokban, mert megengedjuk hogy a latens faktorok csak reszben magyarazzak a megfigyelt valtozok varianciajat. A **fennmarado varianciat vagy meresi hiba, vagy olyan faktorok befolyasoljak, amik egyediak a megfigyelt valtozora**. Ezert az EFA-ban minden egyes megfigyelt valtozohoz tartozik egy "kommunalitas" **(communality)** ertek. Ez az ertek azt mutatja meg, hogy **az adott valtozoban megfigyelheto variancia mekkora hanyadat magyarazzak a latens faktorok**. A fennmarado varianciat a valtozora egyedi faktor vagy meresi hiba magyarazza (ezt egyedisegnek, vagy uniqueness-nek is nevezzuk).

A faktorelemzes legfontsabb lepesei:

- Faktoralhatosag ellenorzese
- Faktorkinyeres
- Idealis faktorszam kivalasztasa
- Faktorforgatas
- Faktorok ertelmezese

4.1 Adatok faktoralhatosaga

Az adatfaktoralhatosag tesztelesekor azt a kerdest valaszoljuk meg, hogy van-e elegendo egyuttjaras (korrelacio) a megfigyelheto valtozok kozott, ami lehetove teszi az EFA elvegzeset. Ennek tesztelesere ket modszert is alkalmazunk: a Bartlett sphericity tesztet es a Kaiser-Meyer-Olkin tesztet.

Mindenek elott azonban **az adatok korrelacios matrixara van szuksegunk**, amin ezeket a teszteket lefuttathatjuk. Ezt megkaphatnank a cor() funkcioval ha folytonos valtozokkal dolgoznank, de ebben az adatbazisban **ordinalis adatokkal van dolgunk**, igy egy masik funkciot hasznalunk aminek a neve **mixedCor() a psych package-bol**. Ez a funkcio kepes az ordinalis adatok eseten hasznalatos **"Polychoric Correlation"** meghatarozasara. A mixedCor() funkcioban meghatarozzuk hogy melyek a folytonos valtozok, es melyek az ordinalis valtozok. A Q1-Q32 mind ordinalis, ezert csak a p = 1:32-t hatarozzuk meg, a c=-t pedig NULL-ra allitjuk, mert nincs folytonos skalan mozgo (continuous) valtozo.

Fontos, hogy a korrelacios matrixot a mixedCor() **a $rho komponenseben tarolja**, ezert ezt kell elmentenunk egy uj adatobjektumba. Mentsuk el ezt a hsq_correl nevu objektumba.

```
hsq_mixedCor <- mixedCor(hsq, c=NULL, p=1:32)
hsq_correl = hsq_mixedCor$rho
```

_____Gyakorlas (opcionalis) _____

A fentebb tanultak alapjan vizualizald a valtozok kozotti korrelaciot. Hasznalj tobb modszert is, pl. ggcorr(), ggcorrplot() hc.order=TRUE-val kombinalva, vagy network_plot().

Bartlett sphericity teszt

A bartlett teszt lenyege hogy a **valos korrelacios matrixot** osszehasonlitjuk egy **hipotetikus null-korrelacios matrix-al**, amiben minden korrelacio 0 erteket vesze fel (identity matrix). A null hipotezis amit itt tesztelunk az, hogy a ket korrelacios matrix nem kulonbozik egymastol. Ha a teszt szignifikans, az azt jelenti hogy **az adattabla valtozoi korrelalnak egymassal**.

Azonban fontos megjegyezni, hogy a Bartlett tesztnek van egy hatulutoje, megpedig hogy **nagy elemszamoknal szinte biztosan szignifikans** eredmenyt ad. Csak olyankor erdemes erre a mutatora hagyatkozni a faktoralhatosag megallapitasakor amikor amikor a megfigyelesek szama es a megfigyelt valtozok szamanak **aranya kisebb mint 5**. A mi esetunkben ez az arany 993/32 = 31.03, vagyis a Bartlett teszt eredmenye nem megbizhato.

```
bfi_factorability <- cortest.bartlett(hsq_correl)
```

```
## Warning in cortest.bartlett(hsq_correl): n not specified, 100 used
```

```
bfi_factorability
```

```
## $chisq
## [1] 1280.585
##
## $p.value
## [1] 9.770759e-71
##
## $df
## [1] 496
```

Kaiser-Meyer-Olkin (KMO) teszt

A KMO teszt a **parcialis korrelacios matrixot** hasonlitja ossze a szokasos korrelacios matrixal. A parcialis korrelacio soran meghatarozzuk hogy **mekkora ket valtozo kozotti korrelacio, ha kivonjuk a tobbi valtozo hatasat a korrelaciobol**. A KMO ertek azt mutatja, hogy mekkora a kulonbseg a parcialis korrelaciok es a szokasos korrelaciok kozott. A KMO egy kulonbseg ertek, a parcialis korrelaciok es a szokasos

korrelaciok kozotti kulonbseget jelzi. Azokban az esetekben ahol a valtozok sok kozos varianciat hordoznak (vagyis valoszinu hogy mogottuk egy kozos latens faktor all), a parcialis korrelaciok alacsonyak, vagyis a KMO index magas. A KMO tesztben az 1-hez kozeli ertekek jok jo faktoralhatosagot mutatnak. A KMO index-nek legalabb 0.6-nak kell lennie hogy ugy iteljuk hogy a valtozok faktoralhatoak.

A mi peldankban a **KMO minden valtozo eseten magasabb 0.6-nal, es az osszesitett KMO is magasabb 0.6-nal**, igy faktoralhatonak tekinthetok a valtozok.

```
KMO(hsq_correl)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = hsq_correl)
## Overall MSA =  0.88
## MSA for each item =
##   Q1   Q2   Q3   Q4   Q5   Q6   Q7   Q8   Q9  Q10  Q11  Q12  Q13  Q14  Q15  Q16
## 0.94 0.93 0.91 0.90 0.91 0.86 0.83 0.85 0.95 0.86 0.79 0.90 0.87 0.92 0.83 0.88
##  Q17  Q18  Q19  Q20  Q21  Q22  Q23  Q24  Q25  Q26  Q27  Q28  Q29  Q30  Q31  Q32
## 0.90 0.82 0.90 0.83 0.89 0.87 0.83 0.83 0.86 0.90 0.84 0.94 0.88 0.79 0.82 0.92
```

4.2 Faktorextrakcio

A faktorokat az fa() funkcioval fogjuk kinyerni. Ez a funkcio tobb faktorextrakcios modszert is kinal. A leggyakrabban hasznalt modszer a **Maximum Likelihood Estimation** (mle) akkor ha a megfigyelt valtozok megfelelnek a tobbvaltozos normalitas feltetelenek, mig a **Principal Axis Factoring** (paf) a preferalt modszer akkor, ha a valtozok nem mutatnak tobbvaltozos normalis eloszlast.

Az mvn() funkcio az MVN package-bol es a mvnorm.kur.test() es a mvnorm.skew.test() funkciok az ICS package-bol segithet eldonteni, hogy tobbvaltozos normalis eloszlast mutatnak-e az adatok. Ha ezeknek a teszteknek a **p-erteke alacsonyabb 0.05-nel, akkor az a tobbvaltozos normalitas serulesere utal**.

```
result <- mvn(hsq[,1:32], mvnTest = "hz")
result$multivariateNormality
```

```
##            Test       HZ p value MVN
## 1 Henze-Zirkler 1.001426       0  NO
```

```
mvnorm.kur.test(na.omit(hsq[,1:32]))
```

```
## Warning in pchisqsum(n * W.stat, df = dfs, a = chi.fac, method = "integration"):
## Package 'CompQuadForm' not found, using saddlepoint approximation
```

```
##
##  Multivariate Normality Test Based on Kurtosis
##
## data:  na.omit(hsq[, 1:32])
## W = 1707.3, w1 = 0.12457, df1 = 527.00000, w2 = 0.23529, df2 = 1.00000,
## p-value < 2.2e-16
```

```
mvnorm.skew.test(na.omit(hsq[,1:32]))
```

```
##
##  Multivariate Normality Test Based on Skewness
##
## data:  na.omit(hsq[, 1:32])
## U = 441.69, df = 32, p-value < 2.2e-16
```

Fent lathato hogy mind a Henze-Zirkler teszt mind a tobbvaltozos ferdeseg es csucsossag tesztek a normalitas feltetelenek serulesere utal. Igy a **paf extrakcios modszert** hasznaljuk majd.

A faktorextrakciora a psych package fa() funkciojat hasznaljuk. Ezen belul megadhatjuk a faktoreztrakcios modszert az fm = parameteren belul. Itt fm = "pa"-t hatarozunk meg, mert a paf modszert szeretnenk hasznalni, de ha a tobbvaltozos normalitas nem serult volna, akkor ehelzett "mle"-t hasznaltunk volna. Az alabbi peldaban meg nem akartam faktorforgatast alkalmazni, hogy lepesrol lepesre tudjam bemutatni a faktorelemzes modszeret, igy a rotate = erteket "none"-ra allittam, de altalaban a faktorokat egybol el is forgatjuk valamelyik modszerrel (lasd alabb). Az nfactors = parameterrel adhatjuk meg, hany faktort szeretnenk kinyerni. Egyelore allitsuk ezt 5-re, lentebb tartgyaljuk majd, hogyan valasztjuk ki az idealis faktormennyiseget.

A modell objektum $communality komponenseben talaljuk a valtozokhoz tartozo kommunalitas ertekekt. Ezt legmagasabbtol legalacsonyabbig sorbarendezzuk es kilistazzuk. Ahogy fentebb emlitettuk a kommunalitas azt jelzi, hogy az egyes megifigyelt valtozokban tapasztalhato variancia mekkora hanyadat magyarazzak a kinyert faktorok. Az output azt mutatja, hogy a Q17 "Altalaban nem szeretek viccelodni, vagy masokat szorakoztatni" ("I usually don't like to tell jokes or amuse people.") a legjobban reprezentalt item az 5 faktoros strukturaban, aminek 68%-at kepesek megmagyarazni az uj faktorok. Ezzel szemben a Q22 "Amikor szomoru vagy ideges vagyok altalaban elvesztem a humorerzekemet" ("If I am feeling sad or upset, I usually lose my sense of humor.") a legkevesbe reprezentalt item, varianciajanak csak 25%-at magyarazza a jelenlegi faktorstruktura.

Neha ahhoz hogy a faktorstruktura jol mukodjon, erdemes a rosszul reprezentalt itemeket kizarni. Ez foleg akkor fontos, ha kicsi a mintaelemszam. **Ha a megfigyelesek szama 250 alatti, akkor MacCallum et al. szerint elvarhato hogy az itemek atlagos kommunalitasa legalabb 0.6 legyen.** A mi esetunkben ennel megengedobbek is lehetunk, mert az elemszamunk nagyobb, de egy melyebb faktorelemzes eseten igy is erdemes lehet elgondolkodni a rosszul reprezentalt itemek kizarasan.

MacCallum, R. C., Widaman, K. F., Zhang, S., & Hong, S. (1999). Sample size in factor analysis. Psychological methods, 4(1), 84.

```
EFA_mod1 <- fa(hsq_correl, nfactors = 5, fm="pa")

# Sorted communality
EFA_mod1_common <- as.data.frame(sort(EFA_mod1$communality, decreasing = TRUE))
EFA_mod1_common
```

```
##     sort(EFA_mod1$communality, decreasing = TRUE)
## Q17                                     0.6754689
## Q20                                     0.6701555
## Q25                                     0.6528782
## Q8                                      0.6265007
## Q21                                     0.6240345
## Q10                                     0.6140024
## Q15                                     0.6047847
## Q18                                     0.5909926
## Q14                                     0.5796418
## Q13                                     0.5557827
## Q32                                     0.5458051
## Q26                                     0.5424384
## Q1                                      0.5311824
## Q31                                     0.5176279
## Q12                                     0.4858949
## Q19                                     0.4806134
## Q5                                      0.4741704
## Q4                                      0.4663425
## Q2                                      0.4442162
## Q16                                     0.4360995
## Q6                                      0.4257846
```

```
## Q29                                    0.4025312
## Q7                                     0.3978300
## Q11                                    0.3930955
## Q3                                     0.3908073
## Q24                                    0.3558342
## Q23                                    0.3419339
## Q27                                    0.3300415
## Q9                                     0.3241293
## Q30                                    0.2867962
## Q28                                    0.2663399
## Q22                                    0.2543910
```

```
mean(EFA_mod1$communality)
```

```
## [1] 0.4777546
```

4.3 Idealis faktorszam kivalasztasa

A fokomponenselemzeshez hasonloan meg kell hataroznunk, hany faktort szeretnenk kinyerni az adatokbol. Ahogy azt a fokomponenselemzesnel is lattuk, ennek az eldontesere hasznalhatjuk a scree-tesztet, a Kaiser-Guttman kriteriumot, vagy a parallel tesztet. Ezen felul a psych pacakge ket ujabb modszert is felkinal a donteshozas elosegitesere: a very simple structure (VSS) kriteriumot, es a Wayne Velicer's Minimum Average Partial (MAP) kriteriumot. (A vss() funkcio a psych package-ben)

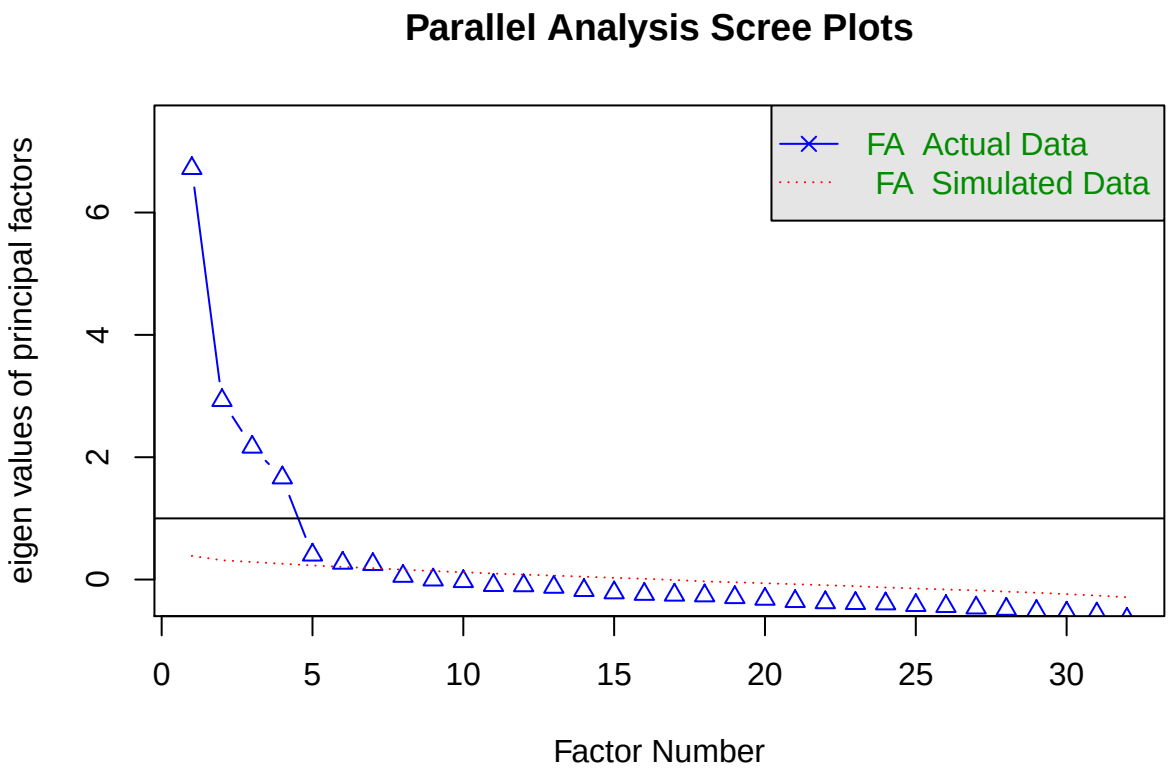
Az alabbi peldaban a psych package fa.parallel funciojat es az nfactors funkciot hasznaljuk arra, hogy a kulonbozo kriteriumok szerint eldonthessuk, hany faktor megtartasa lenne idealis.

A kulonbozo technikak altal javasolt idealis faktorszamok a kovetkezok:

- scree-tesztet: 4
- Kaiser-Guttman kriterium: 4
- Parallel tesztet: 7
- VSS: 3-4
- MAP: 4

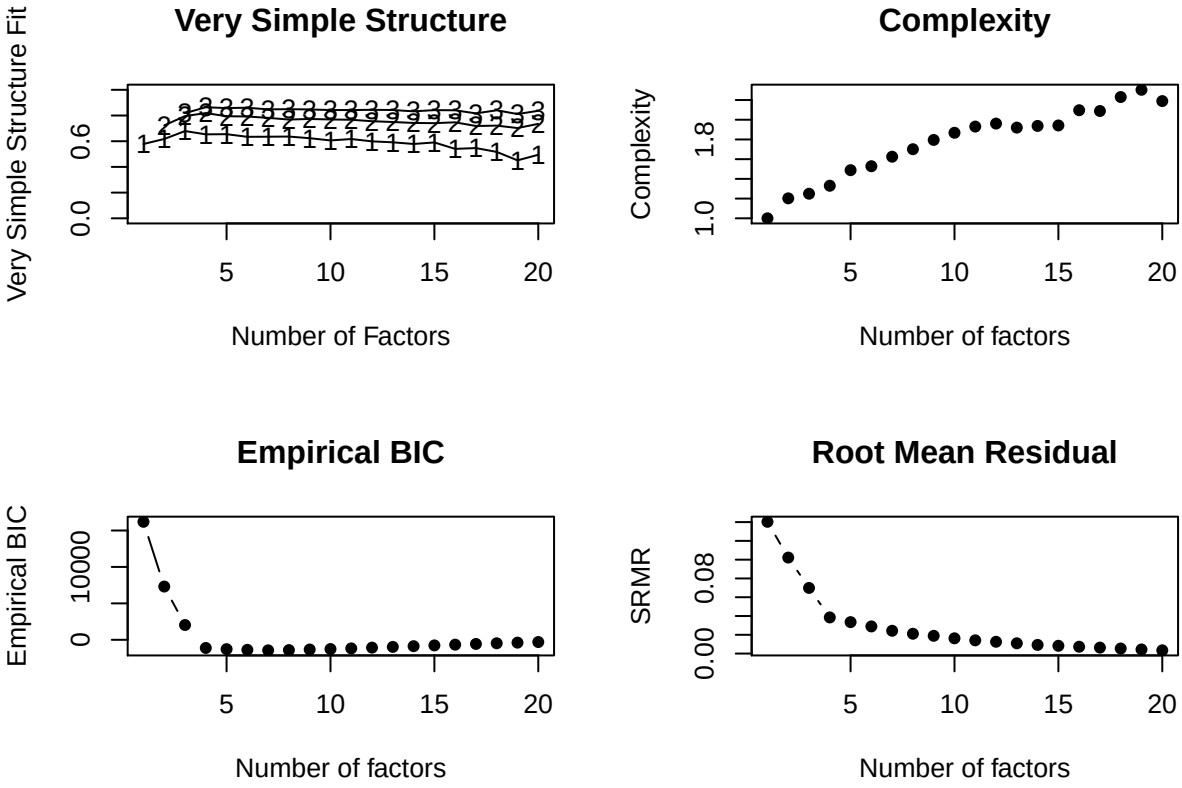
Ezek alapjan ugy tunik, a legtobb technika szerint 4 latens faktor irja le az adatok variabilitasat a legjobban. Alabb meg is epitjuk ezt a 4-faktoros modellt, es megvizsgaljuk a kommunalitas-tablazatot. A faktorelemzes soran nagyon gyakori, hogy a folyamatot ujra es ujra megismeteljuk kulonbozo bemeneti valtozokkal es kulonozo faktorszamokkal es rotacios modszerekkel, amig elerjuk a veglegesnek tekintheto faktorstrukturat. A vegleges faktorstruktura idealis esetben jol ertelmezheto a faktorok es a hozzajuk tartozo valtozo-toltesek alapjan.

```
fa.parallel(hsq_correl, n.obs = nrow(hsq),
        fa = "fa", fm = "pa")
```




```
## Parallel analysis suggests that the number of factors =  7  and the number of components =  NA
```

```
nfactors(hsq_correl, n.obs = nrow(hsq))
```




```
##
## Number of factors
## Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,
##     n.obs = n.obs, plot = FALSE, title = title, use = use, cor = cor)
## VSS complexity 1 achieves a maximimum of 0.68  with  3  factors
## VSS complexity 2 achieves a maximimum of 0.82  with  4  factors
## The Velicer MAP achieves a minimum of 0.01  with  4  factors
## Empirical BIC achieves a minimum of  -1440.89  with  7  factors
## Sample Size adjusted BIC achieves a minimum of  -204.78  with  14  factors
##
## Statistics by number of factors
##    vss1 vss2   map dof chisq     prob sqresid  fit  RMSEA  BIC SABIC complex
## 1  0.58 0.00 0.034 464  8804  0.0e+00    40.0 0.58 0.1345 5602  7075     1.0
## 2  0.62 0.72 0.027 433  6069  0.0e+00    26.3 0.72 0.1145 3081  4457     1.2
## 3  0.68 0.80 0.019 403  4167  0.0e+00    17.2 0.82 0.0970 1386  2666     1.2
## 4  0.65 0.82 0.011 374  2283 1.5e-270    11.5 0.88 0.0717 -298   890     1.3
## 5  0.65 0.79 0.012 346  1980 9.3e-227    10.3 0.89 0.0690 -407   692     1.5
## 6  0.63 0.79 0.013 319  1632 8.3e-175     9.3 0.90 0.0644 -570   443     1.5
## 7  0.63 0.78 0.014 293  1297 3.6e-126     8.4 0.91 0.0587 -725   206     1.6
## 8  0.64 0.77 0.016 268  1099 4.6e-101     7.8 0.92 0.0559 -750   101     1.7
## 9  0.62 0.77 0.017 244   946  3.2e-83     7.3 0.92 0.0538 -738    37     1.8
## 10 0.61 0.77 0.020 221   708  2.5e-52     6.9 0.93 0.0471 -817  -115     1.9
## 11 0.62 0.77 0.022 199   584  1.5e-39     6.4 0.93 0.0441 -789  -157     1.9
## 12 0.60 0.76 0.025 178   500  2.7e-32     6.1 0.94 0.0426 -729  -163     2.0
## 13 0.59 0.75 0.028 158   399  7.8e-23     5.8 0.94 0.0392 -691  -189     1.9
## 14 0.58 0.74 0.032 139   313  2.0e-15     5.5 0.94 0.0355 -646  -205     1.9
```

```
## 15 0.59 0.74 0.036 121   269  3.4e-13     5.2 0.94 0.0350 -566  -182     1.9
## 16 0.54 0.75 0.040 104   223  1.3e-10     4.9 0.95 0.0339 -495  -165     2.1
## 17 0.55 0.72 0.046  88   170  4.1e-07     4.8 0.95 0.0305 -438  -158     2.1
## 18 0.52 0.72 0.052  73   129  6.4e-05     4.5 0.95 0.0277 -375  -143     2.2
## 19 0.45 0.70 0.058  59    73  1.0e-01     4.4 0.95 0.0155 -334  -147     2.3
## 20 0.49 0.74 0.066  46    49  3.6e-01     4.2 0.96 0.0076 -269  -123     2.2
##    eChisq   SRMR eCRMS  eBIC
## 1   19394 0.1403 0.145 16192
## 2   10295 0.1022 0.109  7307
## 3    4812 0.0699 0.078  2031
## 4    1460 0.0385 0.044 -1120
## 5    1105 0.0335 0.040 -1282
## 6     824 0.0289 0.036 -1377
## 7     581 0.0243 0.032 -1441
## 8     439 0.0211 0.029 -1410
## 9     353 0.0189 0.027 -1331
## 10    260 0.0162 0.024 -1265
## 11    196 0.0141 0.022 -1177
## 12    156 0.0126 0.021 -1073
## 13    118 0.0109 0.019  -972
## 14     84 0.0092 0.017  -875
## 15     68 0.0083 0.017  -767
## 16     52 0.0073 0.016  -665
## 17     40 0.0064 0.015  -567
## 18     29 0.0054 0.014  -475
## 19     19 0.0044 0.013  -388
## 20     12 0.0035 0.011  -305
```

```
EFA_mod2 <- fa(hsq_correl, nfactors = 4, fm="pa")

EFA_mod2_common <- as.data.frame(sort(EFA_mod2$communality, decreasing = TRUE))
EFA_mod2_common
```

```
##     sort(EFA_mod2$communality, decreasing = TRUE)
## Q20                                     0.6700371
## Q17                                     0.6697004
## Q25                                     0.6530751
## Q8                                      0.6301917
## Q21                                     0.6195670
## Q10                                     0.6145714
## Q18                                     0.5907715
## Q14                                     0.5678837
## Q32                                     0.5457790
## Q26                                     0.5449964
## Q13                                     0.5427269
## Q1                                      0.5316347
## Q31                                     0.5224554
## Q15                                     0.5181019
## Q12                                     0.4849208
## Q5                                      0.4602084
## Q4                                      0.4523891
## Q2                                      0.4241464
## Q29                                     0.4033447
## Q7                                      0.3973521
## Q3                                      0.3950130
```

```
## Q16                                    0.3775542
## Q19                                    0.3656149
## Q6                                     0.3630614
## Q11                                    0.3313680
## Q27                                    0.3311449
## Q9                                     0.3143381
## Q24                                    0.2969414
## Q23                                    0.2792367
## Q30                                    0.2700595
## Q28                                    0.2369152
## Q22                                    0.1856682
```

```
mean(EFA_mod2$communality)
```

```
## [1] 0.4559615
```

4.4 Faktorforgatas

A faktorforgatas celja hogy megkonnyitse a faktorok ertelmezeset. Igy elkerulheto hogy az egesz faktorstruktura 1 vagy ket nagyon dominans faktorbol alljon, amire angyon erosek a toltesek, mig a tobbi faktor ertelmezese kodos. A faktorforgatas soran az eredeti valtozok ugyan ott maradnak a "faktorterben", viszont a faktorok dimenzio tengelyeit elforgatjuk, hogy jobban railleszkedjenek egyes valtozocsoportokra.

A faktorforgatasnak szamos modszere ismert, de ezek ket fo csoportba sorolhatok: ortogonalis es oblique modszerek koze. Az ortogonalis modszerek (mint pl. Quartimax, Equimax, vagy a pszichologiaban leggyakrabban hasznalt **Varimax** modszer) soran a faktor dimenziok egymasra merolegesek maradnak (ez azt jelenti hogy egymassal nem korrelalnak majd a vegso faktorok). Az oblique modszerek (mint pl. **Direct Oblimin** vagy a Promax) eseten viszont megengedett hogy a vegso faktorok valamellyest korrelaljanak egymassal. Az exploratoros faktorelemzes soran tobb modszert is kiprobalhatunk, de itt fontos az elmeleti megalapozottsag is. Elkepzelheto hogy a faktorok korrelaljanak egymassal? Ha igen, akkor az oblique modszerekre erdemes hagyatkozni. (Altalaban a korrelalatlan faktorokat konnyebb ertelmezni).

Az alapertelmezett faktorforgatasi modszer a Direct Oblimin ("oblimin"). Probaljuk ki a Promax ("promax") es a Varimax ("varimax") modszereket is.

```
EFA_mod2$rotation
```

```
## [1] "oblimin"
```

```
EFA_mod_promax <- fa(hsq_correl, nfactors = 4, fm="pa", rotate = "promax")

EFA_mod_varimax <- fa(hsq_correl, nfactors = 4, fm="pa", rotate = "varimax")
```

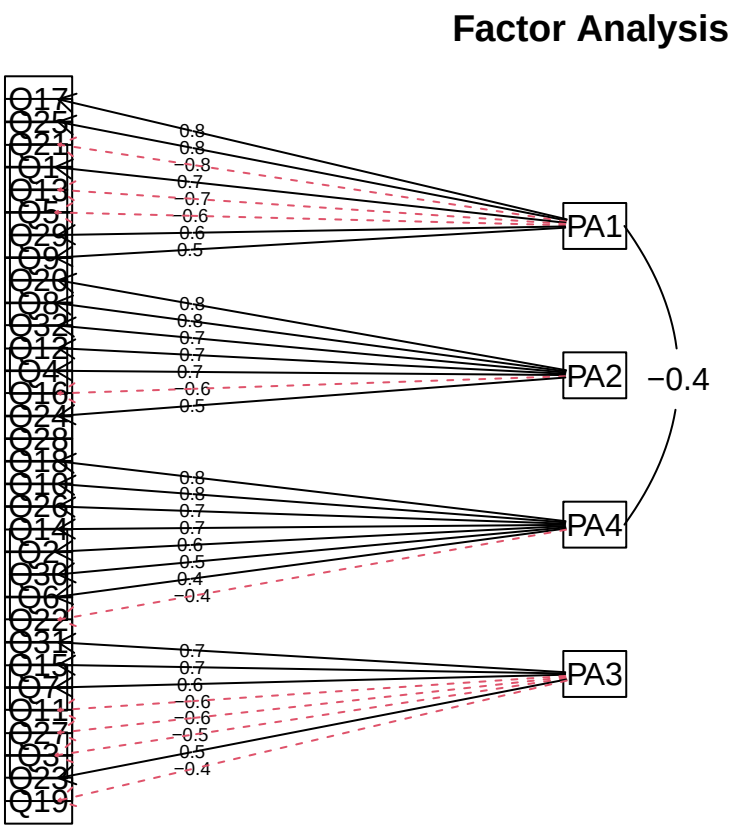
4.5 Faktorok interpretacioja

A faktorok ertelmezese nem konnyu feladat. Sok teruletspecifikus tudasra van szuseg a helyes faktrostruktura kivalasztasahoz es a helyes faktorertelmezeshez. Itt ezert csak a kulonbozo vizualizacios modszereket mutatjuk be amik segithetnek a faktorok ertelmezeseben.

Az fa.diagram() funkcio kirajzolja a model objektum alapjan a faktorstrukturat, es azt, hogy melyik valtozo melyik faktorra mutatja a legnagyobb faktortoltest (melyik faktorral a legnagyobb a korrelacioja). Az abran lathatoak az egyes korrelacios egyutthatok is. A fekete nyilak pozitiv, mig a piros nyilak negativ korrelaciokat jeleznek.

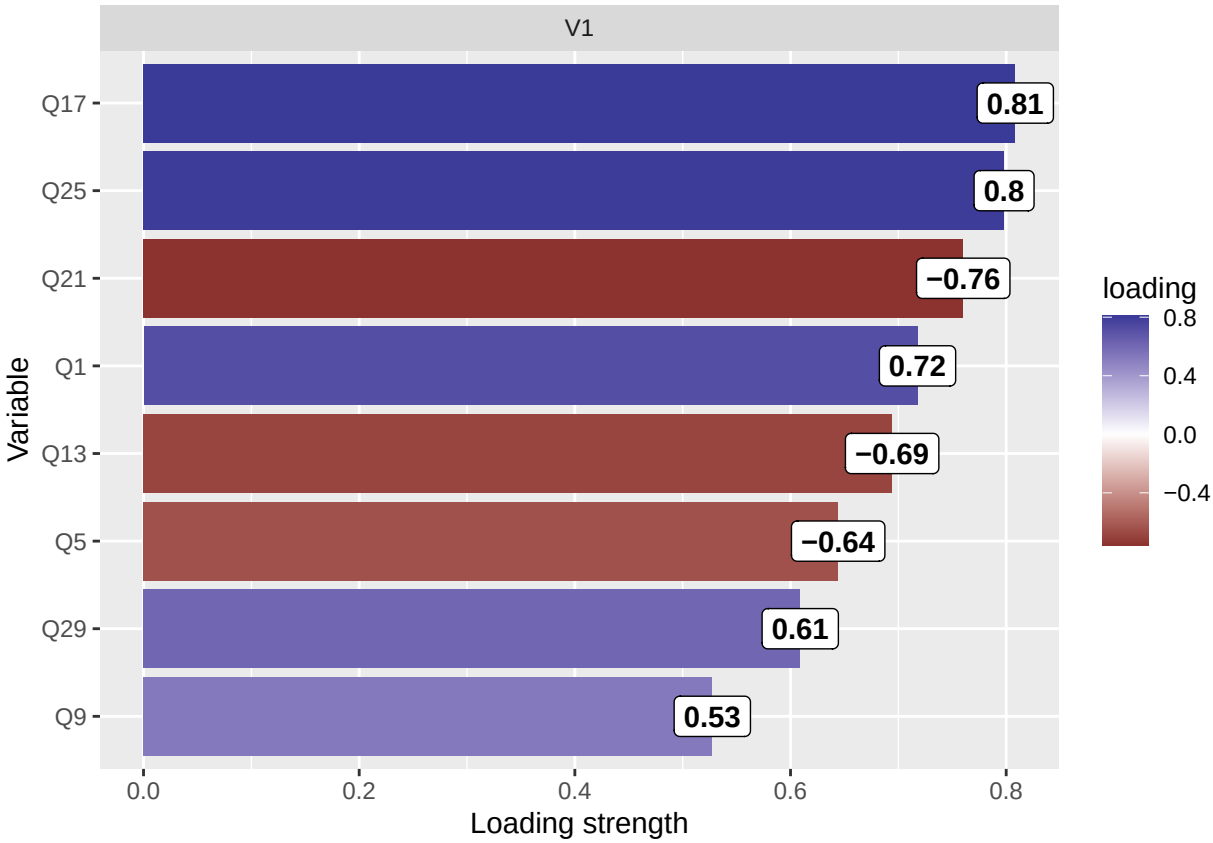
Tovabbi segitseget nyujthat a sajat funkcio amit a fokomponsneselemzesnel is hasznaltunk: fviz_loadnings_with_cor(). Itt a fa() modelek eseten megadhatjuk a loading_above = parameterst is, ahol specifikalhatjuk, hogy csak a bizonyos abszolut faktortoltes (korrelacio) feletti megfigyelt valtozokat abrazoljuk. Ez megkonnyitheti az abra atlathatosagat.

```
fa.diagram(EFA_mod2)
```

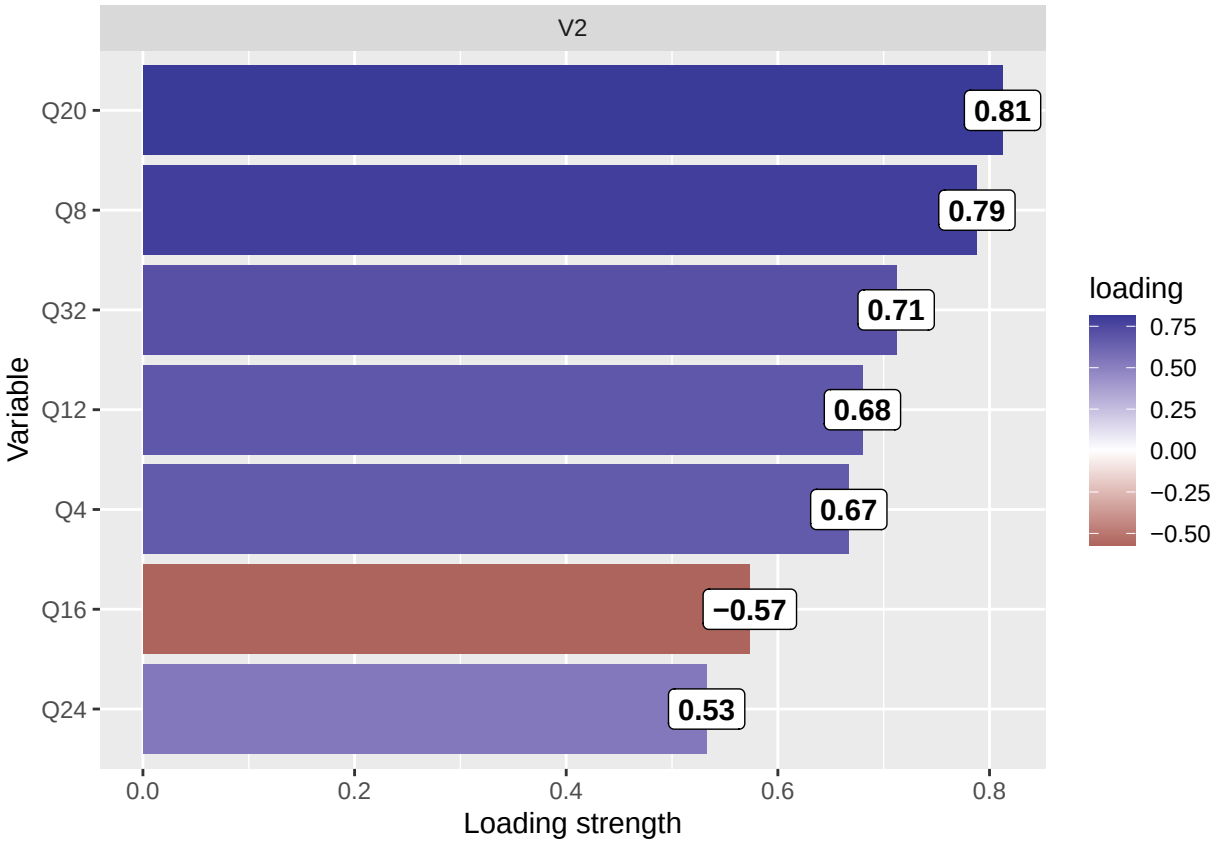



```
fviz_loadnings_with_cor(EFA_mod2, axes = 1, loadings_above = 0.4)
```

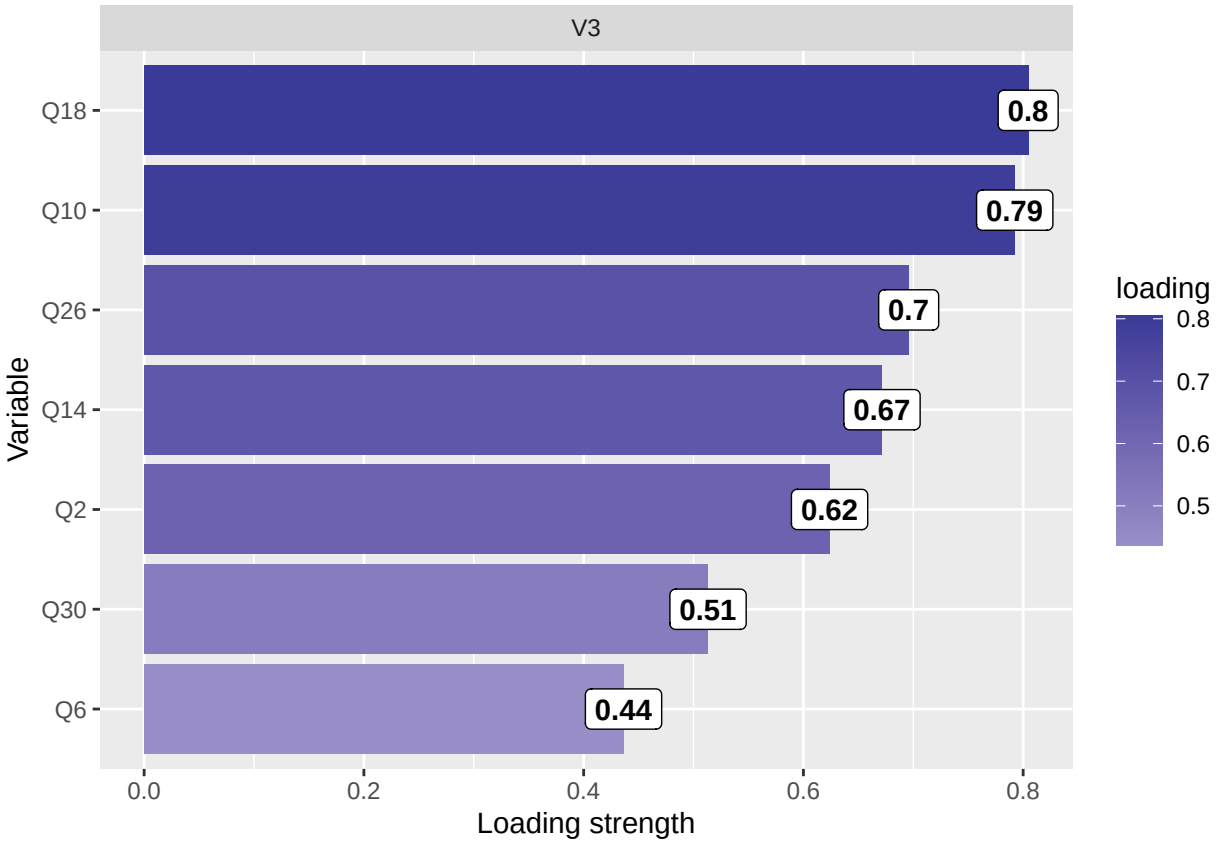
```
## Warning: The `x` argument of `as_tibble.matrix()` must have unique column names if `.name_repair` is
## Using compatibility `.name_repair`.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.

## Warning: The `i` argument of ``[<-`()` can't be a matrix as of tibble 3.0.0.
## Convert to a vector.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.

## Warning: The `i` argument of ``[`()` can't be a matrix as of tibble 3.0.0.
## Convert to a vector.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```



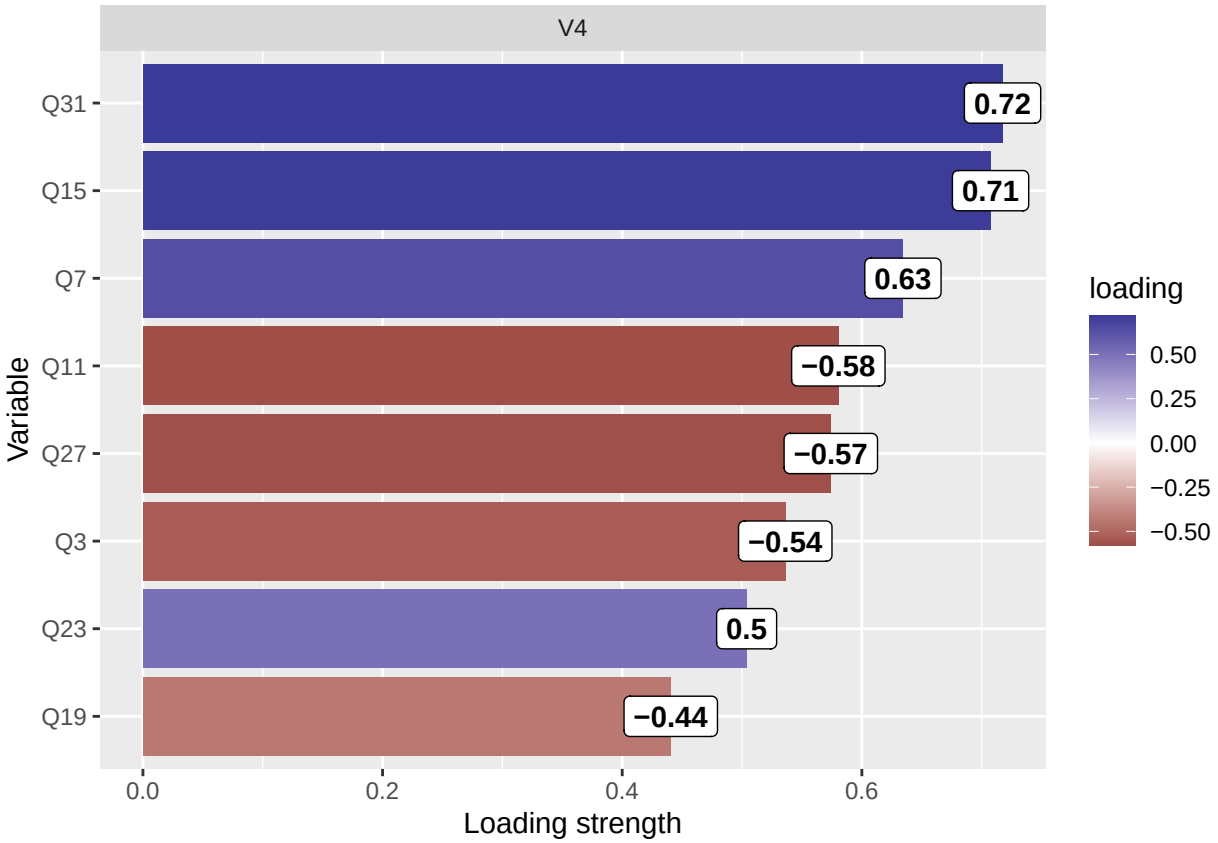

```
fviz_loadnings_with_cor(EFA_mod2, axes = 2, loadings_above = 0.4)
```




```
fviz_loadnings_with_cor(EFA_mod2, axes = 3, loadings_above = 0.4)
```




```
fviz_loadnings_with_cor(EFA_mod2, axes = 4, loadings_above = 0.4)
```



Gyakorlas (opcionalis)

A fent tanult technikakat a Big Five Inventory (bfi) adatbazison gyakorolhatod. Ez a psych package-be beepitett adatbazis, ami 2800 szemely valaszait tartalmazza a Big Five szemelyisegkerdoiv kerdeseira. Az elso 25 oszlop a kerdoiv kerdeseire adott valaszokat tartalmazza, az utolso harom oszlop (gender, education, es age) pedig demografiai kerdeseket tartalmaz. A reszleteket az egyes itemekhez tartozo kerdesekrol es a valaszok kodolasarol elolvashatod ha lefuttatod a ?bfi parancsot.

Az adatbazist betoltheted a kovetkezo parancsokkal.

```
?bfi

data(bfi)
my_data_bfi = bfi[,1:25]
```

Ebben a feladatban csak az elso 25 oszlopot hasznald, az eredeti kerdoiv kerdeseit. Vegezz el feltaro faktorelemzest, es ez alapjan hatarozd meg, hany faktor megtartasa az idealis, mely faktorokra mely itemek toltenek leginkabb, es ez alapjan hogyan nevezned el a faktorokat. Melyek a faktorstruktura altal leginkabb es a legkevesbe reprezentalt itemek?