Unsupervised project: IQ test

Tommaso Locatelli

11/11/2021

Abstract

In this report I will apply unsupervised techniques to argue that intelligence is a multidimensional skill that can hardly be classified with just one number. Analyzing the scores obtained in an IQ test, first with clustering techniques I will identify groups of people with different performances depending on the type of ability tested by the questions. Secondly, I will bring out several concepts of intelligence from principal component analysis.

Research question and dataset

The key idea of an IQ test is that there is some sort of total order relationship between people's intelligence. Thanks to this assumption, it seems sensible to associate a number with each person's performance so that the higher the intelligence, the higher the score. The aim of this analysis is to demonstrate how this approach is at least reductive since intelligence, as a complex concept, cannot be considered a one-dimensional but a multidimensional characteristic. By this I mean that depending on specific tasks, different people can perform better in one but not the other. To do this I intend to analyze a database available on Kaggle (linked phrase) that collects the answers to an IQ test.

The IQ test consisted of: seven vocabulary test questions, six mental rotations and six short-term memory questions. Each question had eight possible answers of which three to five were correct. Each correct answer equals +1 while each incorrect answer -1. Each question is coded according to the type of question (VQ stands for vocabulary question, RQ for rotation question and MQ for memory question) and the number of the question. Furthermore, for each question it is reported: the score (s), the answers (a) and the elapsed time in milliseconds (e). Other values were also recorded:

1. introelapse time spent on the landing page in seconds
2. testelapse time spent on the test page in seconds
3. endelapse time spent on the page where they agreed to donate their data

Here is an example of some values from the first observations

```
##   VQ1s    VQ1a  VQ1e RQ1s MQ1s introelapse testelapse endelapse
## 1    4 2,4,3,1  8382    1    4           9        674        40
## 2    3   4,3,1 42029   -1    4           3        703        46
## 3    4 3,4,2,1 24844    1    3           2       1627        57
## 4    4 4,2,3,1 12188    3    4          30        848        95
## 5    4 1,4,2,3  9023    0    4           4        756       168
## 6    3   2,4,3  4979   -1    4         181        492        85
```

In this project we will only be interested in the scores obtained in the questions, so we get rid of all the remaining columns and calculate the partial sums by type of question and the total sum of the score.

```
##   VQ1s VQ2s VQ3s VQ4s VQ5s VQ6s VQ7s RQ1s RQ2s RQ3s RQ4s RQ5s RQ6s MQ1s MQ2s
## 1    4    3    3    3    3    3    3    1    3    3    4    4    5    4    2
## 2    3    3    3    2    4    4    3   -1    5    4    3    4    4    4    2
```

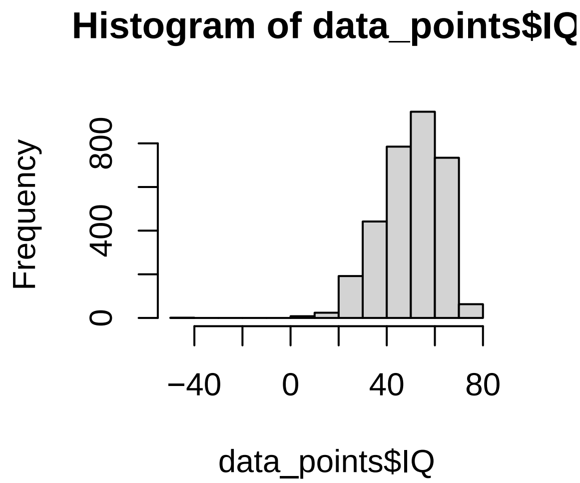
```
## 3    4    2    3    3    3    3    3    1    4    4    4    4    4    3    3
##   MQ3s MQ4s MQ5s MQ6s VQt RQt MQt IQ
## 1    4    4    4    3  22  20  21 63
## 2    3    5    4    4  22  19  22 63
## 3    4    5    4    5  21  21  24 66
```

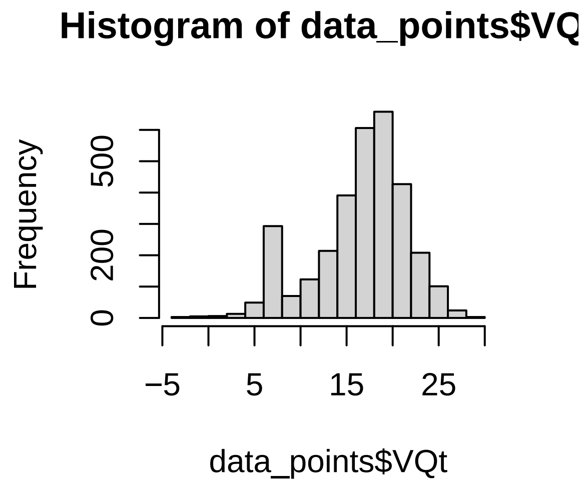
VQt means vocubulary question's total, RQt rotational question's total and MQt memory question's total; the IQ is the total sum.

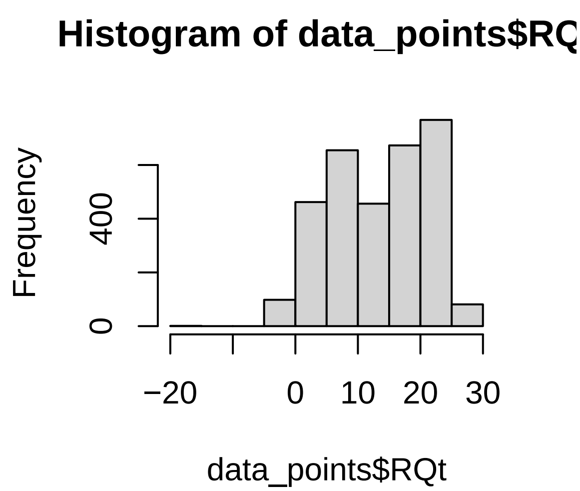
In light of the division into sections of the test we will try to highlight how people can be divided into different types of intelligence rather than one-dimensional degrees of intelligence. For the sake of simplicity we will refer to a one-dimensional view of intelligence as a *traditional paradigm* and to a multidimensional view of intelligence as a *paradigm of heterogeneity*.

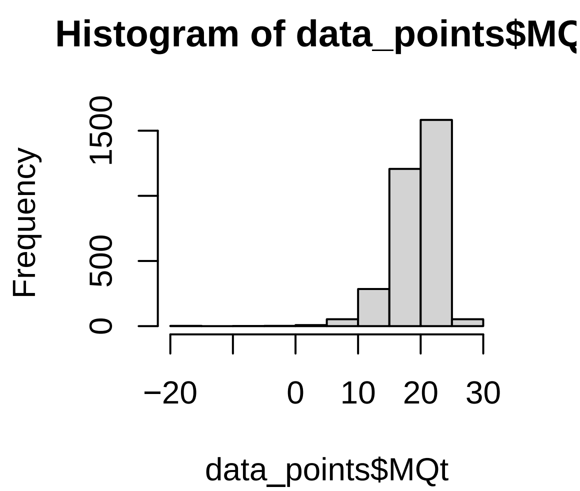
Data desciption

Before starting the analysis let’s look at the distribution of the total scores and in the different sections.









We can make the first observations starting from the histograms and the summary of the different columns:

1. the scores seem more concentrated in the memory questions than the others
2. the average score per question varies between questions, this can be interpreted as having more difficult questions and easier questions
3. there are outliers in the left-hand tail, those who have obtained highly negative final scores
4. if for each question you take the best answer that someone gave you get a score of 81, we don’t know what the real total score was

```
##       VQ1s            VQ2s            VQ3s            VQ4s
##  Min.   :-1.000   Min.   :-2.000   Min.   :-1.000   Min.   :-2.000
##  1st Qu.: 3.000   1st Qu.: 1.000   1st Qu.: 2.000   1st Qu.: 1.000
##  Median : 4.000   Median : 2.000   Median : 3.000   Median : 2.000
```

```
##  Mean   : 3.251   Mean   : 1.605   Mean   : 2.536   Mean   : 1.971
##  3rd Qu.: 4.000   3rd Qu.: 2.000   3rd Qu.: 3.000   3rd Qu.: 3.000
##  Max.   : 4.000   Max.   : 3.000   Max.   : 4.000   Max.   : 4.000
##       VQ5s             VQ6s             VQ7s             RQ1s
##  Min.   :-3.000   Min.   :-2.000   Min.   :-1.000   Min.   :-5.0000
##  1st Qu.: 1.000   1st Qu.: 2.000   1st Qu.: 2.000   1st Qu.: 0.0000
##  Median : 2.000   Median : 3.000   Median : 2.000   Median : 1.0000
##  Mean   : 2.268   Mean   : 2.869   Mean   : 2.481   Mean   : 0.9061
##  3rd Qu.: 3.000   3rd Qu.: 4.000   3rd Qu.: 3.000   3rd Qu.: 2.0000
##  Max.   : 4.000   Max.   : 5.000   Max.   : 5.000   Max.   : 3.0000
##       RQ2s             RQ3s             RQ4s             RQ5s
##  Min.   :-2.000   Min.   :-4.000   Min.   :-3.000   Min.   :-4.00
##  1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.: 2.000   1st Qu.: 1.00
##  Median : 3.000   Median : 3.000   Median : 3.000   Median : 3.00
##  Mean   : 2.815   Mean   : 2.291   Mean   : 2.678   Mean   : 2.32
##  3rd Qu.: 4.000   3rd Qu.: 4.000   3rd Qu.: 4.000   3rd Qu.: 4.00
##  Max.   : 5.000   Max.   : 4.000   Max.   : 5.000   Max.   : 4.00
##       RQ6s            MQ1s             MQ2s            MQ3s
##  Min.   :-3.00   Min.   :-4.000   Min.   :-5.00   Min.   :-4.000
##  1st Qu.: 1.00   1st Qu.: 3.000   1st Qu.: 2.00   1st Qu.: 3.000
##  Median : 3.00   Median : 4.000   Median : 3.00   Median : 4.000
##  Mean   : 2.85   Mean   : 3.555   Mean   : 2.45   Mean   : 3.479
##  3rd Qu.: 5.00   3rd Qu.: 4.000   3rd Qu.: 3.00   3rd Qu.: 4.000
##  Max.   : 5.00   Max.   : 4.000   Max.   : 3.00   Max.   : 4.000
##       MQ4s             MQ5s             MQ6s             VQt
##  Min.   :-3.000   Min.   :-3.000   Min.   :-2.000   Min.   :-4.00
##  1st Qu.: 3.000   1st Qu.: 3.000   1st Qu.: 2.000   1st Qu.:15.00
##  Median : 4.000   Median : 4.000   Median : 3.000   Median :18.00
##  Mean   : 3.841   Mean   : 3.603   Mean   : 2.971   Mean   :16.98
##  3rd Qu.: 5.000   3rd Qu.: 4.000   3rd Qu.: 4.000   3rd Qu.:20.00
##  Max.   : 5.000   Max.   : 5.000   Max.   : 5.000   Max.   :29.00
##       RQt              MQt              IQ
##  Min.   :-18.00   Min.   :-20.0   Min.   :-41.00
##  1st Qu.:  7.00   1st Qu.: 18.0   1st Qu.: 43.00
##  Median : 15.00   Median : 21.0   Median : 52.00
##  Mean   : 13.86   Mean   : 19.9   Mean   : 50.74
##  3rd Qu.: 21.00   3rd Qu.: 22.0   3rd Qu.: 60.00
##  Max.   : 26.00   Max.   : 26.0   Max.   : 79.00
```

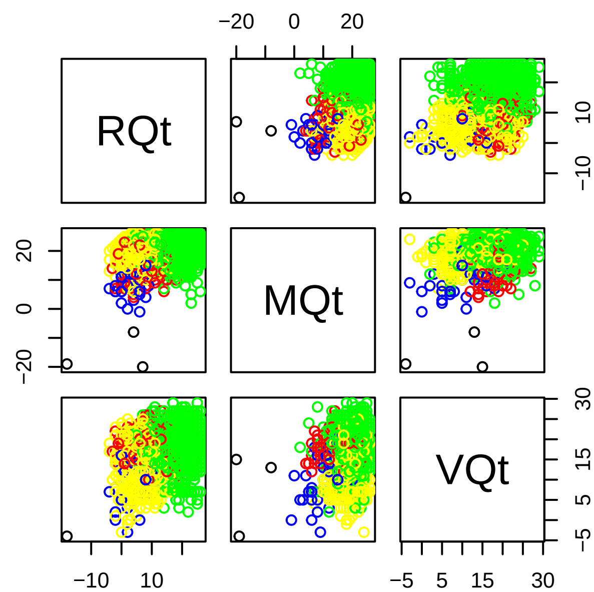
Cluster analysis

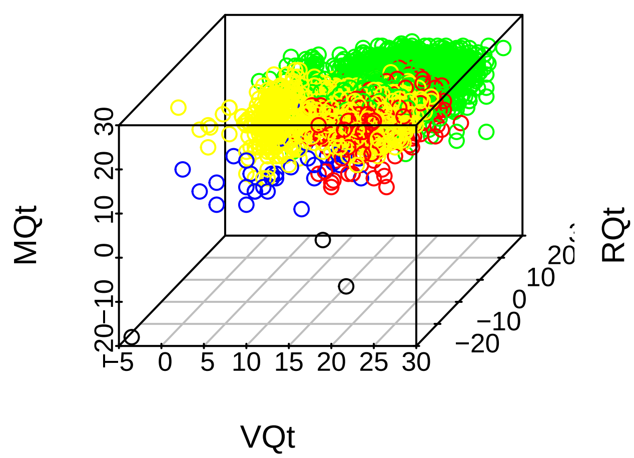
Before proceeding with the cluster analysis it is useful to clarify which results we expect might confirm the *traditional paradigm* or the *heterogeneity paradigm*. So that the data will confirm the *traditional paradigm* if they let identify clusters with a increasing total and partial score in each section, while the data will confirm the *heterogeneity paradigm* if different clusters will perform better or lower than others depending on the section considered.

Complete hierarchical clustering

Let's start with hierarchical clustering using the complete method. We arbitrarily choose to cut the tree to 5 clusters because a larger number would make interpretation difficult for our problem and a smaller number would not bring out enough interest groups.

From the plots it is possible to understand that the clusters obtained are in a certain sense intermediate between the previously hypothesized cases. Although there is a general degree of overall improvement in performance from cluster to cluster it appears that some have areas of strengths and weaknesses.



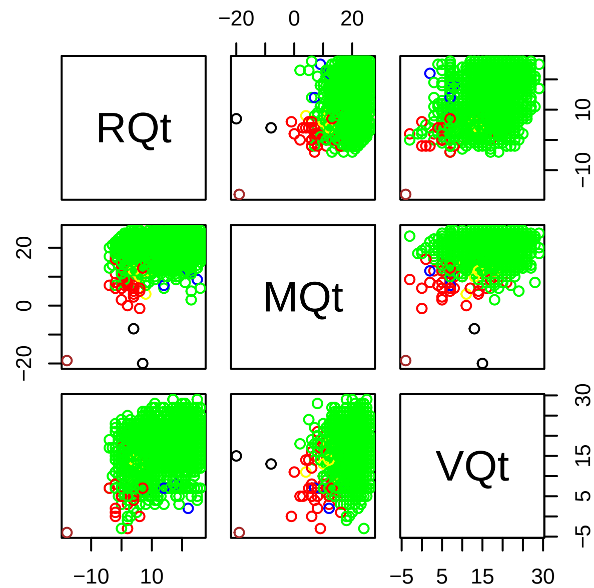


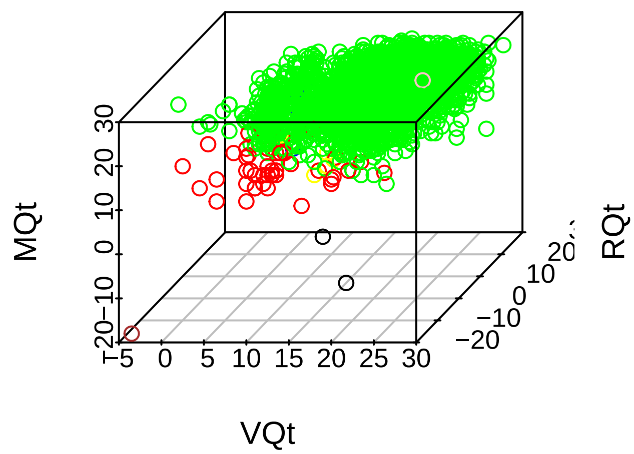

```
##          VQt       RQt        MQt        IQ
## 1 18.638791 18.577330  20.761209  57.97733
## 2 18.217949  7.429487  18.105769  43.75321
## 3 13.089412  5.788235  19.203529  38.08118
## 4  9.181818  3.727273   9.545455  22.45455
## 5  8.000000 -2.333333 -15.666667 -10.00000
```

The first cluster is fine in all but stands out especially for excelling in rotations over the others, the second is distinguished from the first only by a low score in rotations, the third worsens slightly in rotation but drops dramatically in vocabulary, the last two fall in turn in all sectors but represent relatively small clusters and with relatively isolated points with respect to the others.

Average hierarchical clustering

We proceed by repeating the same analysis but using the average method.





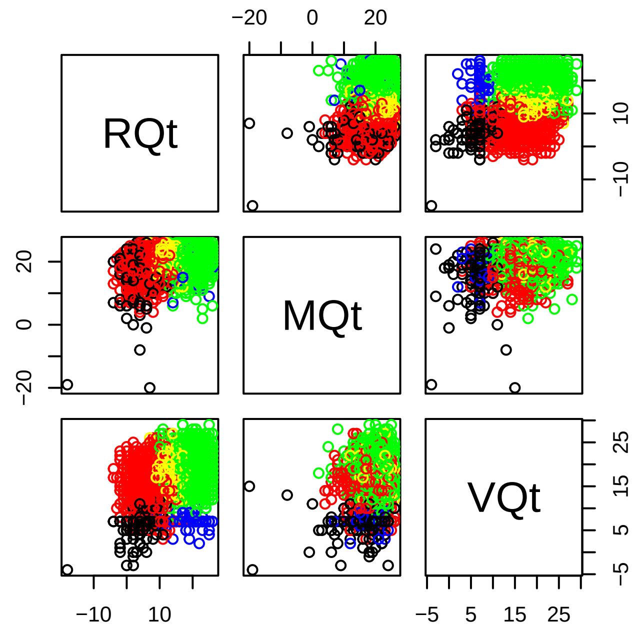

```
##          VQt        RQt        MQt        IQ
## 1 17.171895  14.073624  20.149168  51.39469
## 2  7.843137   2.509804   8.882353  19.23529
## 3 13.142857   5.428571  10.142857  28.71429
## 4  6.375000  17.375000  13.375000  37.12500
## 5 14.000000   5.500000 -14.000000   5.50000
## 6 -4.000000 -18.000000 -19.000000 -41.00000
## 7 24.000000   7.000000  26.000000  57.00000
```

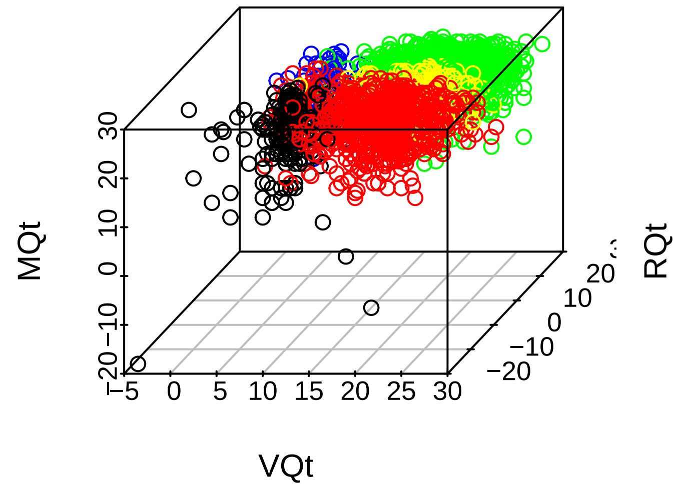
```
##   Count:
## 1   3124
## 2     51
## 3      7
## 4      8
## 5      2
## 6      1
## 7      1
```

This time it seems that most of the points collapse in the same cluster and put small sets of isolated data in the others making the division useless.

Ward hierarchical clustering

The last choice we will evaluate for hierarchical clustering is the Ward method which, trivially speaking, makes the outliers count less.



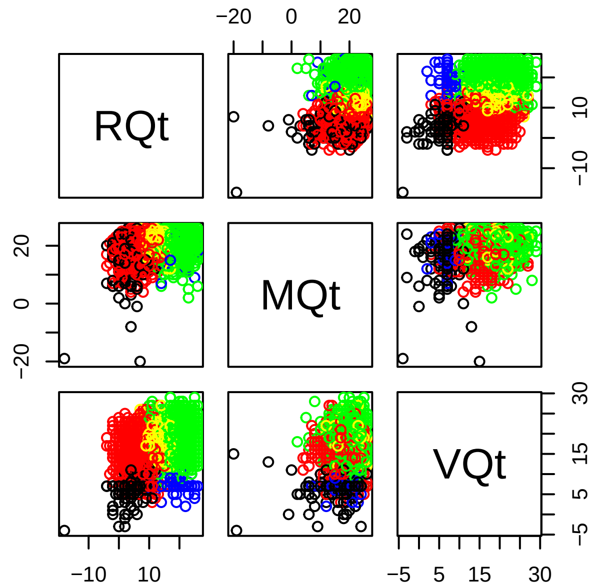


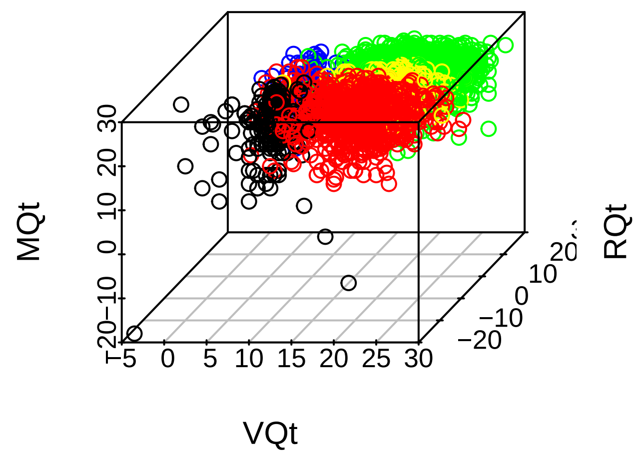

```
##          VQt       RQt      MQt       IQ
## 1 19.250333 20.540613 20.72636 60.51731
## 2 16.512144  6.170011 19.12988 41.81204
## 3 18.784777 12.892388 20.74541 52.42257
## 4  7.181818 19.220779 19.51948 45.92208
## 5  6.881533  4.125436 17.08014 28.08711
```

The results are consistent with what has been found so far: the cluster that achieves the best average score is good in all sections, the second loses points in the rotations, the third in the vocabulary, the fourth very bad in the rotation and thank goodness in the vocabulary and the last in all but memory.

K-means

Finally we try to apply k-means to find the clusters.






```
##          VQt       RQt      MQt       IQ
## 1 18.976387 17.290437 20.51240 56.77922
## 2 17.622912  6.859189 19.89021 44.37232
## 3 12.429043 17.267327 18.35644 48.05281
## 4  9.054348  3.821739 16.33696 29.21304
## 5 20.729223 22.638070 22.03351 65.40080
```

Once again the component with the highest average total score seems to do well in all sections of the test. The second group performs slightly worse in all but in reality it differs especially for the score in the rotation (consistently with the first cluster analysis). The third group seems to recover a little in the rotation but perform much worse in the vocabulary. The fourth group continues not to do badly if it were not for a bad result in the rotations. Finally the last group is worse in everything without lowering too much the score in the memory.

As in the first and third cases, the presence of the best group and the worst group seems to support the *traditional paradigm* while the presence of intermediate groups that are distinguished by preference in the sections is in favor of the *heterogeneity paradigm*.

Which questions are most significant?

We will use the clusters obtained with the ward method because they are the ones that have a greater interpretability in the plots.

```
##        col    R2
##   [1,] "VQ1s" "0.189891844006696"
##   [2,] "VQ2s" "0.0720731341847017"
##   [3,] "VQ3s" "0.10694638913912"
##   [4,] "VQ4s" "0.0654800300681875"
##   [5,] "VQ5s" "0.151680981839776"
##   [6,] "VQ6s" "0.187794547067379"
##   [7,] "VQ7s" "0.0897622633701561"
##   [8,] "RQ1s" "0.18701678044398"
##   [9,] "RQ2s" "0.358856978313244"
```

```
## [10,] "RQ3s" "0.427363266956955"
## [11,] "RQ4s" "0.256286188112877"
## [12,] "RQ5s" "0.388558589723273"
## [13,] "RQ6s" "0.548762611792546"
## [14,] "MQ1s" "0.0640103180504532"
## [15,] "MQ2s" "0.0446826919651256"
## [16,] "MQ3s" "0.0399369984322127"
## [17,] "MQ4s" "0.0176358292168162"
## [18,] "MQ5s" "0.0439505905261409"
## [19,] "MQ6s" "0.0368790433046102"
```

Coherently with the fact that the scores in the memory section are more concentrated and with the fact that it is the characteristic that changes less between clusters, r^2 associated with the related questions turn out to be smaller than 0.1 unlike the values associated with the questions of the other two sections. In general, a question's r^2 can be interpreted as its power to discern between peopleintelligence. Questions with a low value were either too easy or too difficult.

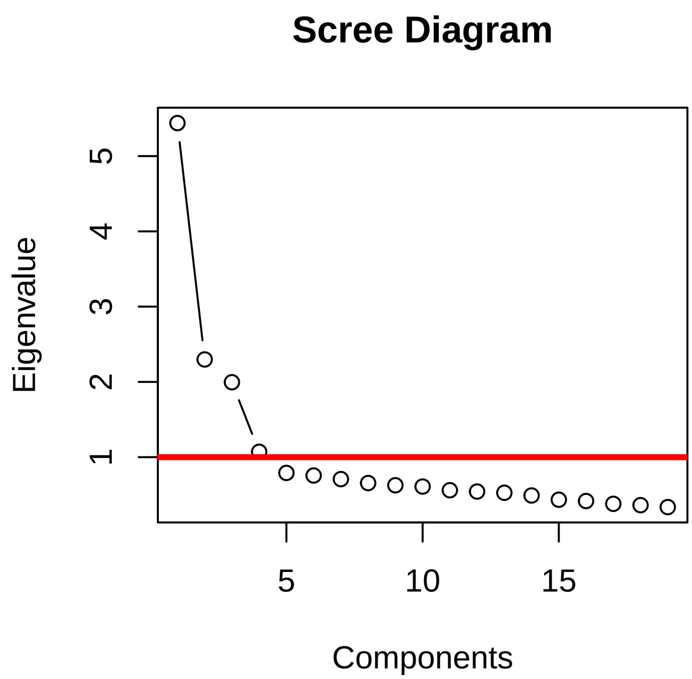
PCA

Up to this point the cluster analysis seems to have confirmed what this project tries to argue, in order to have a deeper point of view on the issue we also proceed with the principal component analysis.

The explain variance of each component is:

```
##  [1] 0.28627341 0.12096436 0.10502547 0.05636385 0.04159870 0.03978615
##  [7] 0.03727912 0.03450778 0.03296934 0.03206879 0.02949568 0.02859120
## [13] 0.02771974 0.02581958 0.02285284 0.02194080 0.01998942 0.01904430
## [19] 0.01770948
```

The result is quite bad, from the mono-dimensional ranking point of view, because there are no components with a high value, this probably depends also on the large number of variables. Despite this, the first three can explain about 50% of the variance.



We can see how the first 4 have an eigenvalue greater than one.

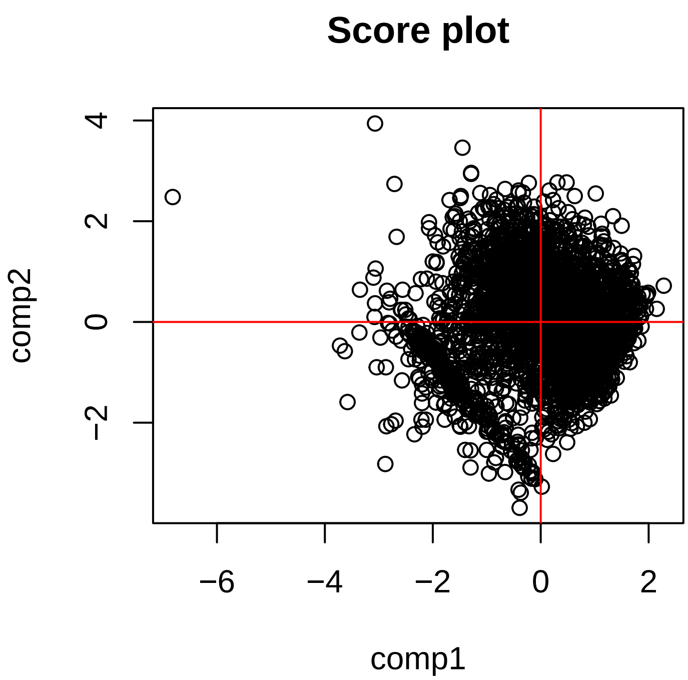
Looking at these components it seems that the analysis is positive all in all for the thesis of this project. Although the first component is positively linked to each question, which makes it interpreted as a sort of degree of general intelligence, but the low variance explained by it show that is not really easy to rank only on one dimension. It is also true that have larger coefficient in the rotation question, so probably is also linked with a sort of mathematical intelligence.

The second and third also have a behavior very favorable to the *paradigm of heterogeneity*. The second component assigns positive values only to questions related to the vocabulary, we could therefore interpret it as a linguistic or verbal intelligence. While the third associates negative values mainly only with questions related to rotation, this could be interpreted as a negation of logical-mathematical intelligence. Having no other information on the content of the questions it is difficult to interpret the fourth component.

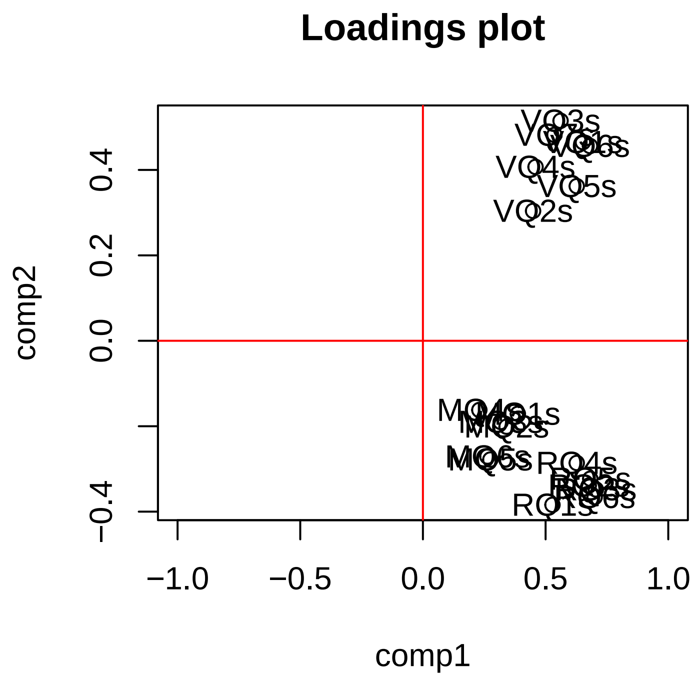
```
##      Comp1  Comp2  Comp3  Comp4 comunalita
## VQ1s 0.652  0.465  0.050  0.032   0.644853
## VQ2s 0.449  0.304 -0.003 -0.080   0.300426
```

```
## VQ3s 0.561  0.515  0.060  0.062   0.587390
## VQ4s 0.459  0.407  0.048  0.063   0.382603
## VQ5s 0.627  0.362 -0.002 -0.010   0.524277
## VQ6s 0.681  0.456  0.002  0.013   0.671870
## VQ7s 0.536  0.482  0.058  0.050   0.525484
## RQ1s 0.528 -0.384 -0.261 -0.066   0.498717
## RQ2s 0.676 -0.339 -0.270  0.046   0.646913
## RQ3s 0.705 -0.348 -0.278 -0.015   0.695638
## RQ4s 0.627 -0.286 -0.241  0.089   0.540927
## RQ5s 0.682 -0.323 -0.266 -0.023   0.640738
## RQ6s 0.700 -0.366 -0.270  0.020   0.697256
## MQ1s 0.387 -0.171  0.464 -0.543   0.689155
## MQ2s 0.341 -0.201  0.508 -0.521   0.686187
## MQ3s 0.317 -0.190  0.592 -0.090   0.495153
## MQ4s 0.230 -0.162  0.531  0.350   0.483605
## MQ5s 0.275 -0.278  0.502  0.335   0.517138
## MQ6s 0.263 -0.271  0.453  0.477   0.575348
```

It seems that low comunalities are associated with some questions, it may be link to low variance which could lead us to consider them also as less significant.



Now we can see, in the score plot, pretty well how the data are far from being crushed just on one dimension.



Furthermore, the loads seem to group according to the type of question. Therefore the observers belonging to the second and fourth quadrant of the score plot, are clearly people who, even if they have similar total scores, simply have different cognitive abilities and without being one better than the other; contradicting the *traditional paradigm*.

Conclusion

Given the high interpretability and consistency of the distribution of scores in each section between the clusters found, the low variance explained by the main component and the match between components and types of questions, this report claims to be largely successful in its goal, that is to highlight how intelligence cannot be considered a one-dimensional characteristic. In particular, the data showed that according to the type of questions different people find themselves answering better or worse than others; as for example, a good verbal intelligence does not also involve a good logical-mathematical intelligence and vice versa. In light of this, the value of the IQ tests should be reconsidered not in the mere final score but in the range of different cognitive areas. For example, it might be valuable to add still different types of questions to allow those with other qualities to demonstrate it. A final consideration is that not all available information has been used in this project. It would be interesting to use the columns relating to the elapsed times to evaluate whether who performs best takes even less time to answer or if speed is a characteristic independent from general intelligence. Or through a study with association rules to understand if there are ways of thinking shared between groups that lead them not only to similar scores but also to similar correct answers and errors.

Appendix

```
library(dplyr)
library(tidyverse)
library("scatterplot3d")
library(wesanderson)

data <- read.csv("C:/Users/tomma/Desktop/Salini projects/Unsupervised/data.csv")
head(data[c(1,2,3,22,40,59,60,61)])
data_points=data[c(seq(1,56,3))]
data_points <-  data_points %>%
  add_column(VQt = rowSums(.[1:7]),
             RQt = rowSums(.[8:13]),
             MQt = rowSums(.[14:19]))
data_points <-  data_points %>%
  add_column(IQ = rowSums(.[20:22]))
head(data_points, n=3L)
hist(data_points$IQ)
hist(data_points$VQt)
hist(data_points$RQt)
hist(data_points$MQt)
summary(data_points)
h1=hclust(dist(data_points[-c(20,21,22,23)]),method="complete") #euclidean distance by default
#plot(h1)
complete=cutree(h1, k=5)
colors <- c('green', 'red','yellow','blue','black')[unclass(complete)]
pairs(~RQt+MQt+VQt,data=data_points, col=colors)
scatterplot3d(data_points[,20:22], angle = 45, color=colors)
medie1<-aggregate(data_points, list(complete), mean)
medie1[c(21,22,23,24)]
count1<-aggregate(data_points, list(complete), FUN = length)
#count1[c(24)]
h2=hclust(dist(data_points[-c(20,21,22,23)]),method="average") #euclidean distance by default
#plot(h1)
average=cutree(h2, k=7)
colors2 <- c('green', 'red','yellow','blue','black','brown','pink')[unclass(average)]
pairs(~RQt+MQt+VQt,data=data_points, col=colors2)
scatterplot3d(data_points[,20:22], angle = 45, color=colors2)
medie2<-aggregate(data_points, list(average), mean)
medie2[c(21,22,23,24)]
count2<-aggregate(data_points, list(average), FUN = length)
names(count2)[c(24)] <- 'Count:'
count2[c(24)]
h4=hclust(dist(data_points[-c(20,21,22,23)]),method="ward.D2") # forcing to include outliers
#plot(h1)
ward=cutree(h4, k=5)
colors <- c('green', 'red','yellow','blue','black')[unclass(ward)]
pairs(~RQt+MQt+VQt,data=data_points, col=colors)
scatterplot3d(data_points[,20:22], angle = 45, color=colors)
medie4<-aggregate(data_points, list(ward), mean)
medie4[c(21,22,23,24)]
count4<-aggregate(data_points, list(ward), FUN = length)
#count4[c(24)]
```

```
fit <- kmeans(data_points, 5) # 4 cluster solution
pairs(~RQt+MQt+VQt,data=data_points, col=colors)
scatterplot3d(data_points[,20:22], angle = 45, color=colors)
medie3=aggregate(data_points,by=list(fit$cluster),FUN=mean)
medie3[c(21,22,23,24)]
##### calculate R^2
mydata<-data_points[-c(20,21,22,23)]
mydata$group<-complete #choose the cluster
R2 <- rep(NA, (ncol(mydata)-1))
for(i in 1:(ncol(mydata)-1))
  R2[i] <- anova(aov(mydata[,i] ~ mydata[,ncol(mydata)]))[1,2]/(anova(aov(mydata[,i] ~ mydata[,ncol(myda
mydata<-mydata[,-ncol(mydata)]
col<-colnames(mydata)
finali<-cbind(col,R2)
finali
n <- nrow(data_points[-c(20,21,22,23)])
p <- ncol(data_points[-c(20,21,22,23)])
rho <- cor(data_points[-c(20,21,22,23)])
autoval <- eigen(rho)$values
autovec <- eigen(rho)$vectors

# Select components
pvarsp = autoval/p
pvarspcum = cumsum(pvarsp)
pvarsp
# Scree Diagram:
plot(autoval, type="b", main="Scree Diagram", xlab="Components", ylab="Eigenvalue")
abline(h=1, lwd=3, col="red")
medie <- colMeans(data_points[-c(20,21,22,23)])
scarto <- apply(data_points[-c(20,21,22,23)], 2, sd)
descrittive<-round(cbind(medie, scarto),2)
comp<-round(cbind(-eigen(rho)$vectors[,1]*sqrt(autoval[1]),-eigen(rho)$vectors[,2]*sqrt(autoval[2]),-eig
rownames(comp)<-row.names(descrittive)
colnames(comp)<-c("Comp1","Comp2","Comp3","Comp4")
comunalita<-comp[,1]^2+comp[,2]^2+comp[,3]^2+comp[,4]^2
comp<-cbind(comp,comunalita)
comp
# scores:
oecd=data_points[-c(20,21,22,23)]
oecd.scale <- scale(oecd, T, T)
punteggi <- oecd.scale%*%autovec[,1:2]
# standardized scores
punteggiz<-round(cbind(-punteggi[,1]/sqrt(autoval[1]),-punteggi[,2]/sqrt(autoval[2])),2)
plot(punteggiz, main="Score plot",
     xlab="comp1",ylab="comp2")
abline(v=0,h=0,col="red")
# loadinngs
plot(comp[,1:2], main="Loadings plot",
     xlab="comp1",ylab="comp2", xlim=range(-1,1))
text(comp, rownames(comp))
abline(v=0,h=0,col="red")
```