

# Análisis Factorial - Psicología

Arleth Michell Morales García

2022-04-21

## Descarga de paquetes y librería

```
install.packages("psych")
```

```
library(psych)
```

```
install.packages("polycor")
```

```
library(polycor)
```

```
install.packages("ggcorrplot")
```

```
library(ggcorrplot)
```

## Extracción de datos

```
x<-bfi
```

## Exploración de la matriz

```
dim(x)
```

```
## [1] 2800 28
```

Nombre de las variables

```
colnames(x)
```

```
## [1] "A1" "A2" "A3" "A4" "A5" "C1"
## [7] "C2" "C3" "C4" "C5" "E1" "E2"
## [13] "E3" "E4" "E5" "N1" "N2" "N3"
## [19] "N4" "N5" "O1" "O2" "O3" "O4"
## [25] "O5" "gender" "education" "age"
```

Creación de una matriz de datos en donde se incluye las variables 1 a la 25 y las primeras 200 observaciones

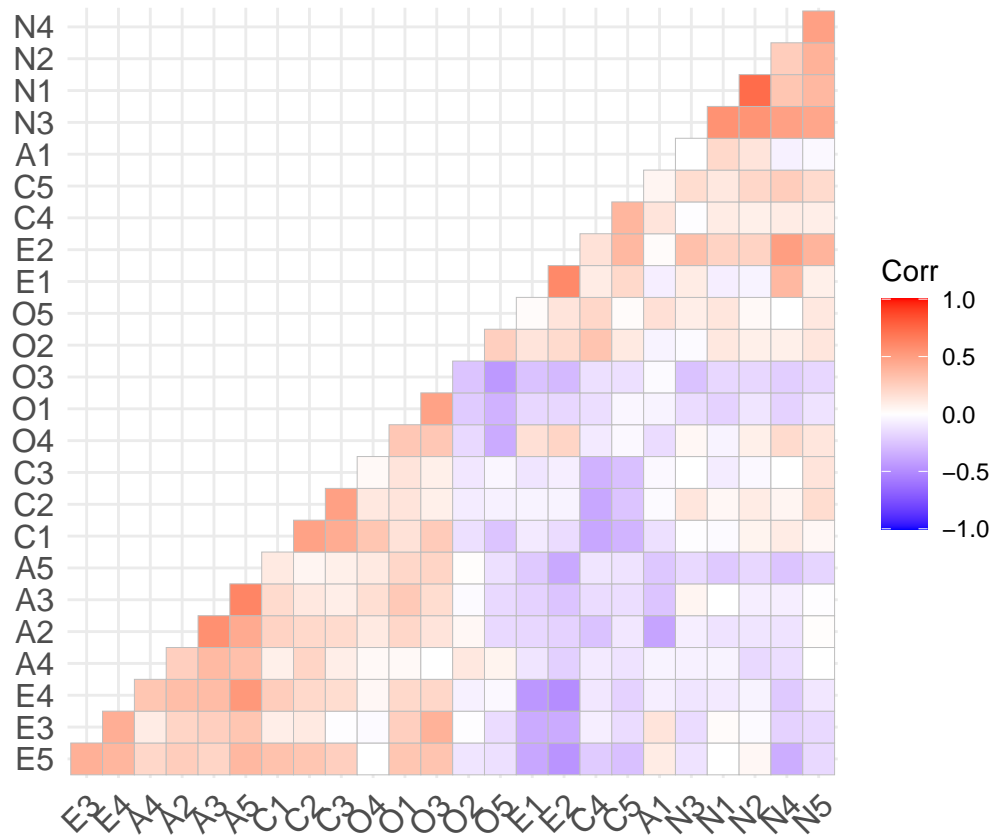
```
x1<-bfi[1:200, 1:25]
```

## Matriz de correlaciones

```
R<-hetcor(x1)$correlations
```

Gráfico de correlaciones

```
ggcorrplot(R, type="lower", hc.order=TRUE)
```



## Factorización de la matriz de correlaciones

Se utiliza la prueba de esfericidad de Bartlett.

```
p_Bartlett<-cortest.bartlett(R)
```

Visualización del p-valor

```
p_Bartlett$p.value
```

```
## [1] 5.931663e-60
```

Ho: Las variables están correlacionadas Ha: Las variables no están correlacionadas

No rechazo Ho, ya que las variables están correlacionadas.

## Criterio Kaiser-Meyer-Olkin

Me permite identificar si los datos que voy a analizar son adecuados para un análisis factorial.

0.00 a 0.49 No adecuados 0.50 a 0.59 Poco adecuados 0.60 a 0.69 Aceptables 0.70 a 0.89 Buenos 0.90 a 1.00 Excelente

```
KMO(R)
```

```
## Kaiser-Meyer-Olkin factor adequacy
```

```
## Call: KMO(r = R)
```

```
## Overall MSA = 0.76
```

```
## MSA for each item =
##   A1   A2   A3   A4   A5   C1   C2   C3   C4   C5   E1   E2   E3   E4   E5   N1
## 0.66 0.77 0.69 0.73 0.75 0.74 0.79 0.76 0.76 0.74 0.80 0.81 0.79 0.81 0.83 0.70
##   N2   N3   N4   N5   O1   O2   O3   O4   O5
## 0.67 0.82 0.79 0.82 0.79 0.65 0.81 0.62 0.77
```

## Extracción de factores

minres: mínimo residuo mle: max verosimilitud paf: ejes principales alpha: alfa minchi: mínimos cuadrados minrak: rango mínimo

```
modelo1<-fa(R, nfactor = 3, rotate = "none", fm="mle")
```

```
modelo2<-fa(R, nfactor = 3, rotate = "none", fm="minres")
```

Extraer el resultado de las comunialidades, ahí se encuentra la proporción de varianza explicada. Se interpreta de tal forma que número cercanos a 1 el factor explica mejor la variable.

```
C1<-sort(modelo1$communality, decreasing = TRUE)
```

```
C2<-sort(modelo2$communality, decreasing = TRUE)
```

```
head(cbind(C1,C2))
```

```
##           C1           C2
## N1 0.7576920 0.6809294
## E2 0.6802809 0.6564523
## N2 0.6797943 0.5866483
## E1 0.5219674 0.5394762
## N3 0.5198285 0.4942059
## N4 0.4839516 0.4744005
```

Extracción de **Unicidades** La unicidad es el cuadrado del coeficiente del factor único, y se expresa como la porción de la varianza explicada por el factor único. Es decir, no puede ser explicada por otros factores.

```
u1<-sort(modelo1$uniquenesses, decreasing = TRUE)
```

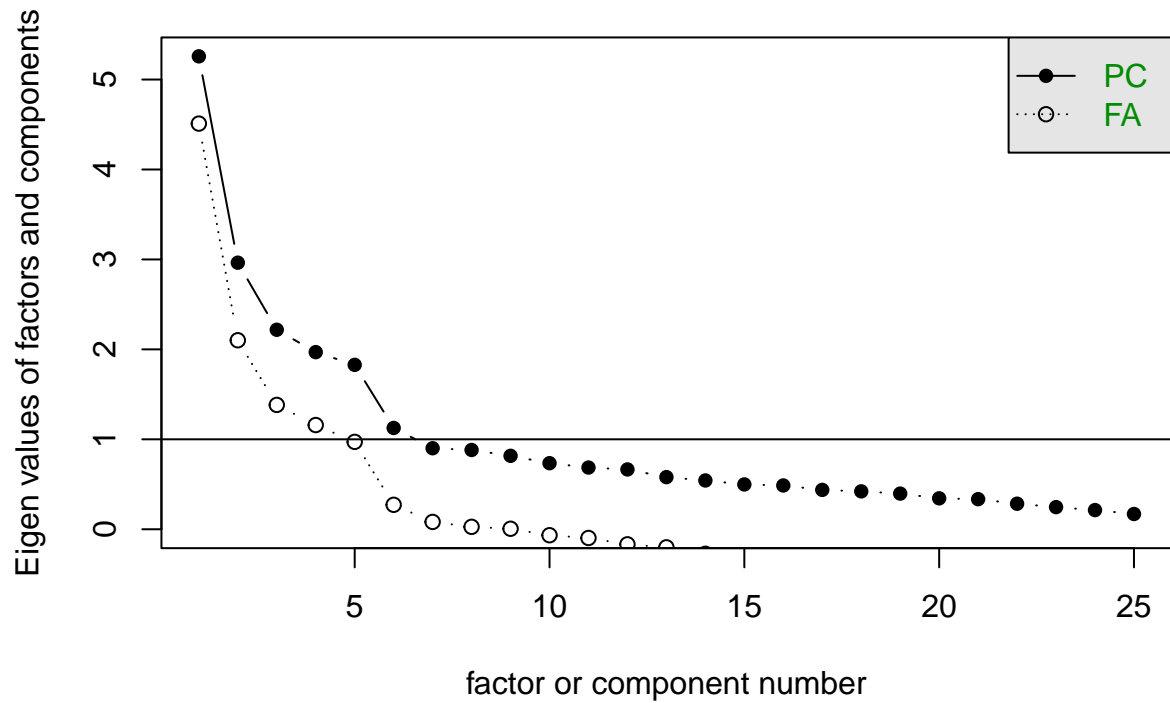
```
u2<-sort(modelo2$uniquenesses, decreasing = TRUE)
```

```
head(cbind(u1,u2))
```

```
##           u1           u2
## O2 0.9460554 0.9293483
## A4 0.8928892 0.8908844
## A1 0.8607240 0.8822080
## O5 0.8533481 0.8272041
## C5 0.8136600 0.7931685
## O1 0.7986908 0.7904667
```

```
scree(R)
```

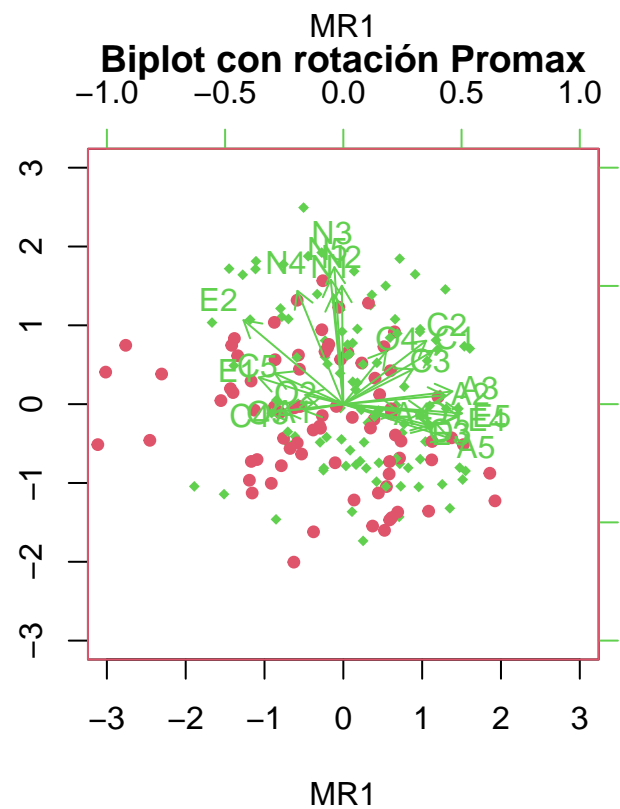
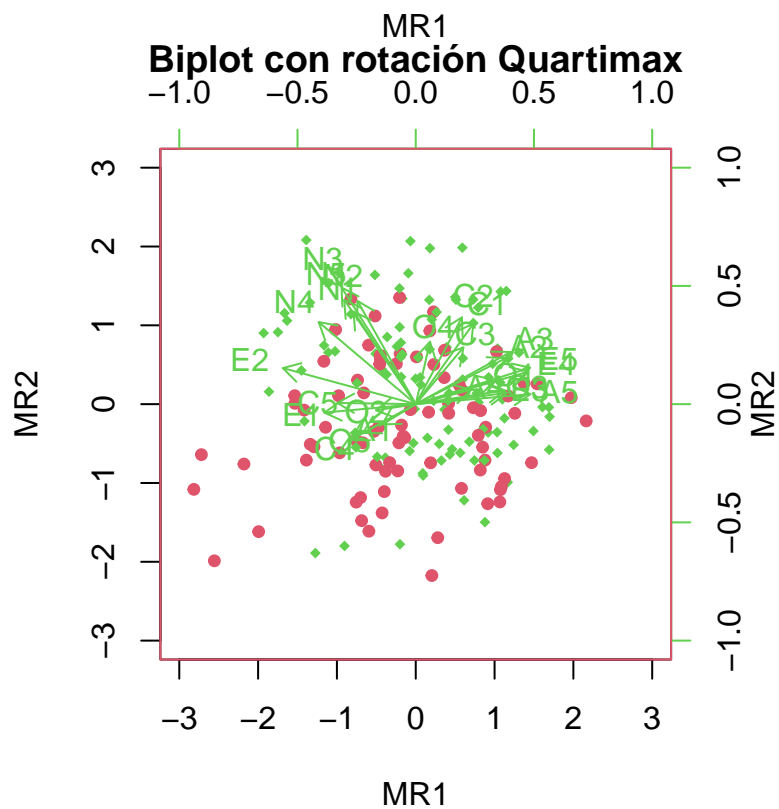
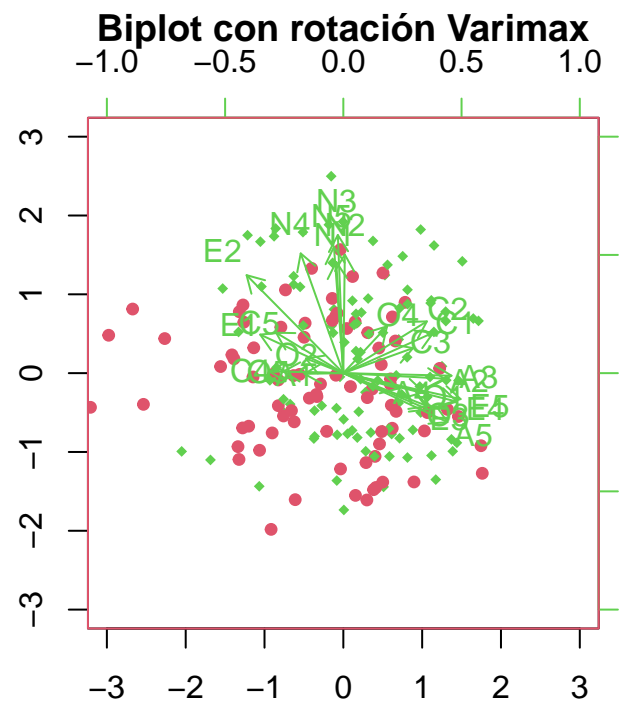
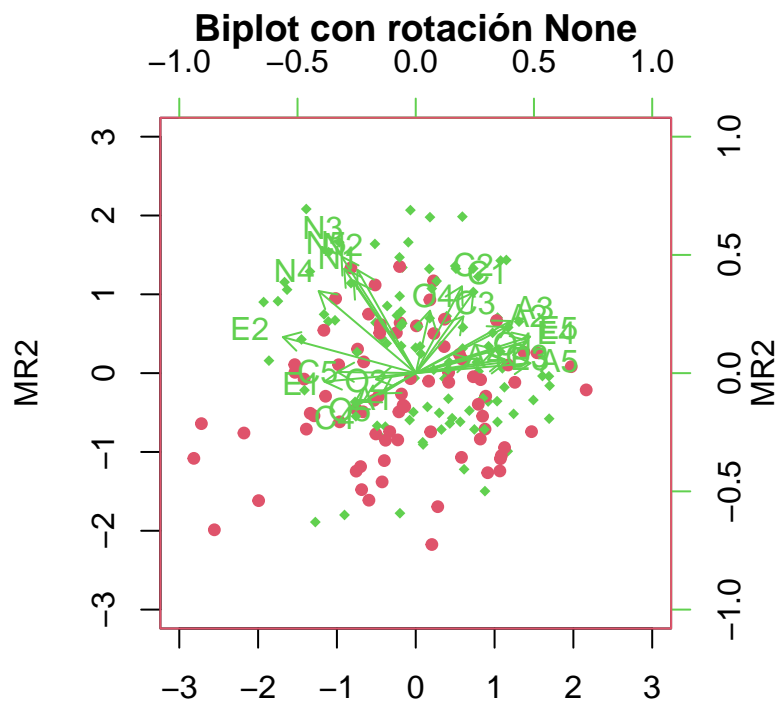
## Scree plot



Rotación de la matriz

```
library(GPArotation)

rot<-c("None", "Varimax", "Quartimax", "Promax")
bi_mod<-function(tipo){
  biplot.psych(fa(x1, nfactors = 2,
    fm="minres", rotate = tipo),
    main=paste("Biplot con rotación", tipo),
    col=c(2,3,4), pch=c(21,18), group=bfi[, "gender"])
}
sapply(rot, bi_mod)
```



```
## $None
## NULL
##
## $Varimax
## NULL
```

```
##
## $Quartimax
## NULL
##
## $Promax
## NULL
```

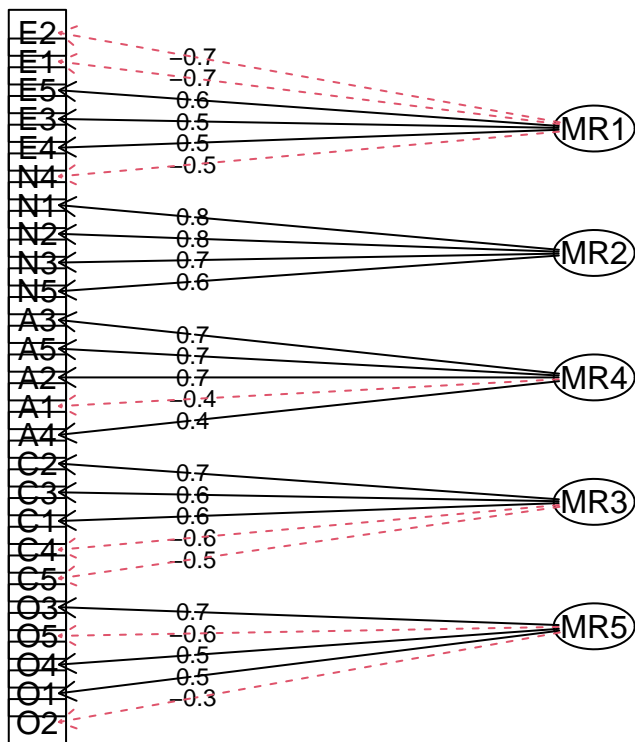
## Interpretación

Para esto se utiliza el gráfico de árbol. Ya que proporciona una mejor visualización de los resultados

```
modelo_varimax<-fa(R, nfactor = 5,
                    rotate = "varimax",
                    fm="minres")
```

```
fa.diagram(modelo_varimax)
```

## Factor Analysis



negras son cargas positivas.

Las líneas rojas son cargas negativas y las

Visualización de la matriz de carga rotada.

```
print(modelo_varimax$loadings, cut=0)
```

```
##
## Loadings:
##      MR1  MR2  MR4  MR3  MR5
## A1  0.234  0.106 -0.422 -0.072 -0.092
## A2  0.112 -0.032  0.653  0.190  0.113
## A3  0.198  0.066  0.744  0.051  0.169
## A4  0.163 -0.048  0.413  0.137 -0.142
```

```

## A5  0.328 -0.154  0.692 -0.009  0.115
## C1  0.054  0.089  0.140  0.634  0.287
## C2  0.052  0.174  0.114  0.690  0.050
## C3  0.032  0.018  0.076  0.642  0.016
## C4 -0.058  0.087 -0.090 -0.559 -0.159
## C5 -0.241  0.228 -0.040 -0.459  0.014
## E1 -0.691 -0.006 -0.066 -0.084 -0.017
## E2 -0.713  0.345 -0.138 -0.133 -0.025
## E3  0.546  0.003  0.157 -0.008  0.221
## E4  0.522 -0.027  0.416  0.167  0.048
## E5  0.588 -0.009  0.148  0.308  0.159
## N1  0.131  0.802 -0.150 -0.074 -0.133
## N2  0.088  0.800 -0.151 -0.038 -0.008
## N3 -0.183  0.701  0.005  0.037 -0.087
## N4 -0.513  0.491 -0.006  0.004  0.034
## N5 -0.274  0.571  0.059  0.096 -0.082
## O1  0.203 -0.107  0.148  0.076  0.535
## O2 -0.099  0.096  0.144 -0.191 -0.330
## O3  0.326 -0.159  0.034  0.062  0.680
## O4 -0.240  0.122  0.169  0.105  0.548
## O5 -0.004  0.061 -0.074 -0.077 -0.636
##
##              MR1   MR2   MR4   MR3   MR5
## SS loadings    2.823 2.667 2.223 2.103 1.867
## Proportion Var 0.113 0.107 0.089 0.084 0.075
## Cumulative Var 0.113 0.220 0.309 0.393 0.467

```