

Análisis Factorial

Ignacio Aguilera Martos

9 de enero de 2019

Estadística Multivariante

Link: [Documentos LaTeX en GitHub](#)

1. Explicación teórica del modelo
2. Explicación del ejemplo en R

Explicación teórica del modelo

Idea de AF

El objetivo de este modelo es, dada una matriz de covarianzas o de correlación, ser capaz de explicar esta matriz a partir de factores no observados llamados factores comunes, de forma que se pueda explicar la matriz con un número menor de variables que en un punto inicial.

De esta forma matricialmente si tenemos n observaciones de dimensión p

tendríamos una matriz $F = \begin{pmatrix} F_{11} & \dots & F_{1n} \\ \dots & \dots & \dots \\ F_{k1} & \dots & F_{kn} \end{pmatrix}$ de factores y

$L = \begin{pmatrix} l_{11} & \dots & l_{1k} \\ \dots & \dots & \dots \\ l_{p1} & \dots & l_{pk} \end{pmatrix}$ una matriz de coeficientes de forma que

$x - \mu = LF + \epsilon$ donde ϵ es un vector de errores.

Tipos

- EFA (Exploratory Factor Analysis): se usa para identificar relaciones complejas entre conceptos o grupos de conceptos.
- CFA (Confirmatory Factor Analysis): está dirigida a la confirmación de factores que ya se presuponen importantes para explicar la matriz de correlación o covarianza.

Procesos de ajuste

- Máxima verosimilitud: es una buena opción cuando los datos se distribuyen según una normal. Se intenta que el modelo de los factores obtenidos tenga máxima verosimilitud.
- Factorización en el eje principal: la intención es ir obteniendo factores de forma que el primero tenga la varianza lo más próxima al objetivo, el segundo factor la segunda varianza más próxima a la varianza objetivo, etc. Maximiza la fórmula del modelo.

La rotación de factores se emplea para obtener la estructura de factores más simple escogiendo una orientación de los mismos.

Rotación de factores

- Ortogonal: implica que los factores estén incorrelados y busca la estructura más simple.
- Oblicua: permite que los factores estén correlados y busca la estructura más simple.

No sólo tenemos que estudiar la generación de los factores, si no también cuántos factores debemos escoger.

Métodos para escoger el número de factores

- Regla de Kaiser: tomamos los valores propios de la matriz de entrada y comprobamos cuántos de ellos son mayores que 1. Este es el número de factores a tomar. En caso de no haberlo se toma un factor.
- Criterio de la gráfica de Cattell's: obtenemos los valores propios de la matriz de entrada y los pintamos de mayor a menor. Analizamos el cambio entre los valores propios y donde se produzca el último cambio brusco contamos el número de valores propios hasta él. Este es el número de factores. Es un método subjetivo y ampliamente criticado.

Métodos para escoger el número de factores

- VSS (Very Simple Structure): este procedimiento toma un modelo simplificado del problema e intenta ver para qué número de factores los valores obtenidos se acercan más a los que deberían ser.
- Comparación de modelos: intentan obtener una medida de cómo de bueno y complejo es el modelo creado con un número de factores dado, de forma que se intenta maximizar el resultado y minimizar la complejidad.
- OC (Optimal Coordinate): intenta eliminar la subjetividad del método de Cattell. Se calculan los gradientes en la misma gráfica de dicho método y se comprueba dónde hay un cambio más abrupto. Esto delimita el número de factores.
- AF (Acceleration Factor): persigue el mismo objetivo que el método anterior, salvo que en este caso se realiza el cálculo con la pendiente de la curva asociada.

Métodos para escoger el número de factores

- MAP (Minimum Average Partial) test: se realiza, desde $k=1$ hasta el número de variables menos 1 un análisis PCA del modelo con k número de factores. Se estudia cómo se comporta dicho modelo con k factores y se toma el valor de k para el cual se ha obtenido el mejor resultado.
- PA (Parallel Analysis): tomamos la misma gráfica que en el método de Cattell y generamos aleatoriamente un conjunto de valores. Estos valores representan los valores medios de los valores propios de matrices aleatorias con el mismo número de variables y datos que la original. Hallamos la media para estos valores y tomamos como número de factores el de los que superen este valor medio. Se puede ver como un refinamiento de la regla de Kaiser ajustando la cota.

Métodos para escoger el número de factores

- Comparación de datos: se realiza una comparación entre modelos con una separación en factores ya conocida como correcta comparando los valores propios de la matriz y los factores escogidos en cada caso. Se toma el número de factores del modelo cuyos valores propios se parezcan más a los del caso que estamos analizando.
- Convergencia de múltiples tests: esta estrategia busca de forma empírica el mejor número de factores analizando la convergencia del modelo.

El objetivo de CFA es, dada una hipótesis sobre el número de factores, si los factores están o no correlados y qué variables se relacionan con cada factor. Es por esto que EFA y CFA se usan en contextos muy diferentes aún siendo los dos tipos del mismo método.

Métodos para evaluar la hipótesis

- Ajuste absoluto: consiste en obtener una medida absoluta de cómo de ajusta el modelo hipotético a los valores proporcionados.
- Ajuste relativo: obtiene una medida comparativa de cómo funciona con respecto a un modelo llamado base. Este modelo es fijo y nos sirve para obtener una valoración absoluta del mismo muy mala, de forma que con el ajuste relativo obtenemos una medida de cuánto se acerca la medida de nuestra hipótesis a la de este modelo con mal ajuste.

Medidas de ajuste absoluto

- Test chi-cuadrado: nos da la diferencia entre la matriz de covarianza observada y predicha con el modelo hipotético. Cuanto más cercano sea el valor a cero mejor es el ajuste.
- Aproximación según la raíz cuadrada del error: se obtiene un valor entre 0 y 1 que mide el error del modelo con respecto a la observación. Cuanto más cercano sea el valor a 0 mejor es el ajuste.
- Raíz cuadrada de la media de los residuos al cuadrado y raíz cuadrada de la media de los residuos estandarizados al cuadrado: no sólo estudiamos los residuos si no que además estandarizamos las variables para que los valores obtenidos sean comparables. El rango de valores está entre 0 y 1 siendo 0 la mejor medida. Normalmente cuando tenemos una puntuación de menos de 0.08 decimos que el modelo es robusto

Medidas de ajuste absoluto

- Índice de bondad del ajuste e índice de bondad del ajuste equilibrado: son medidas del ajuste del modelo y la matriz de covarianza observada. El índice equilibrado pondera el valor de ajuste en la matriz de covarianza para cada factor en función de cuántas variables explique. Los valores que arroja están entre 0 y 1 siendo a partir de 0.9 un buen modelo.

Medidas de ajuste relativo

- Índice de ajuste normado e índice de ajuste no normado: el índice normado analiza las discrepancias entre el valor de la chi-cuadrado con los del modelo y los del modelo base. El problema de este índice es que no es insesgado, lo cual es corregido por el índice de ajuste no normado el cual se mueve entre 0 y 1, siendo valores mayores o iguales a 0.95 indicativos de un buen modelo.
- Índice de ajuste comparativo: el anterior índice normado tiene una alta sensibilidad a la dimensión de la muestra tomada, de forma que este índice se plante extender la idea sin este problema- Compara el ajuste del modelo con un modelo base independiente. Los valores de este índice están ente 0 y 1 siendo necesarios valores mayores o iguales a 0.9 para considerar a la hipótesis un buen modelo.

¿Cuál es la diferencia entonces con PCA?

Diferencia entre PCA y AF

- PCA no es más que un posible método de extracción en la primera fase de EFA en la que extraemos los factores. Tras esto viene la fase de rotación de los mismos para escoger una orientación.
- Precisamente, por la carencia de la rotación, PCA no obtiene unos factores únicos, mientras que EFA sí.
- AF es en general un método mucho más genérico y abierto que PCA.

Explicación del ejemplo en R

Paquetes a instalar

- `install.packages("psych")`: implementa Análisis Factorial Exploratorio
- `install.packages("GPArotation")`: implementa la Rotación de Factores
- `install.packages("cfa")`: implementa Análisis Factorial Confirmatorio
- `install.packages("lavaan")`: implementa la sintaxis de especificación del modelo para cfa

Importa los paquetes

- `library(psych)`
- `library(GPArotation)`
- `library(cfa)`
- `library(lavaan)`

Funciones a usar

- `fa`: función del paquete `psych` que implementa Análisis Factorial Exploratorio.
- `VSS`: función que devuelve el número de factores a obtener usando el método VSS
- `factanal`: función del paquete `stats` que implementa Análisis Factorial Exploratorio con máxima verosimilitud.
- `cfa`: función del paquete `cfa` que implementa el análisis confirmatorio de factores.

Función fa

La función tiene un gran número de parámetros, pero los esenciales son los siguientes:

```
fa(r,nfactors=1,n.obs = NA,n.iter=1, rotate="oblimin",  
scores="regression", residuals=FALSE, SMC=TRUE,  
covar=FALSE,missing=FALSE,impute="median", min.err = 0.001,  
max.iter = 50,symmetric=TRUE, warnings=TRUE, fm="minres", alp-  
ha=.1,p=.05,oblique.scores=FALSE,np.obs=NULL,use="pairwise",cor="cor",  
correct=.5,weight=NULL,...)
```

Parámetros esenciales de la función fa

- r: matriz de covarianza o correlación.
- nfactors: número de factores a obtener.
- fm: método de ajuste empleado.
- rotate: método de rotación empleado.

Función VSS

La función tiene un gran número de parámetros, pero los esenciales son los siguientes:

```
VSS(x, n = 8, rotate = "varimax", diagonal = FALSE, fm = "minres",  
n.obs=NULL,plot=TRUE,title="Very Simple  
Structure",use="pairwise",cor="cor",...)
```

Parámetros esenciales de la función VSS

- x: matriz de correlación o datos.
- n: número de factores máximos.
- fm: método de ajuste empleado.

Función factanal

La función tiene un gran número de parámetros, pero los esenciales son los siguientes:

```
factanal(x, factors, data = NULL, covmat = NULL, n.obs = NA, subset,  
na.action, start = NULL, scores = c("none", "regression", "Bartlett"),  
rotation = "varimax", control = NULL, ...)
```

Parámetros esenciales de la función factanal

- x: matriz de correlación o datos.
- factors: número de factores máximos.

Función cfa

La función tiene un gran número de parámetros, pero los esenciales son los siguientes:

```
cfa(model = NULL, data = NULL, ordered = NULL, sampling.weights =  
NULL, sample.cov = NULL, sample.mean = NULL, sample.th = NULL,  
sample.nobs = NULL, group = NULL, cluster = NULL, constraints =  
"", WLS.V = NULL, NACOV = NULL, ...)
```

Parámetros esenciales de la función cfa

- model: modelo especificado en sintaxis lavaan
- data: variables observadas experimentalmente

A continuación se muestra el script de ejemplo elaborado.

```

1 cov_matrix <- Harman74.cor$cov
2 library(psych)
3 library(GPArotation)
4 library(cfa)
5 library(lavaan)
6 ##                                EFA                                ##
7
8 ##                                Numero de factores                                ##
9 VSS(sim.item(nvar=24),n=8,fm="minres",title="VSS of 24 simple structure variables")
10
11 ##                                Con rotacion                                ##
12 # Principal Axis
13 pa_rotated <- fa(cov_matrix, 4, fm="pa", rotate="varimax")
14 # Unweighted Least squares: minres
15 uls_rotated <- fa(cov_matrix, 4, rotate = "varimax")
16 # Weighted least squares
17 wls_rotated <- fa(cov_matrix, 4, fm = "wls")
18
19 ##                                Máxima verosimilitud                                ##
20 mle_rotated <- factanal(covmat = cov_matrix, factors = 4)
21
22 ##                                Sin rotacion                                ##
23 # Weighted Least squares
24 wls_nonrotated <- fa(cov_matrix, 4, rotate = "none", fm="wls")
25 # Principal Axis
26 pa_nonrotated <- fa(cov_matrix, 4, rotate = "none", fm="pa")
27 # Minres
28 minres_nonrotated <- factanal(factors=4,covmat=cov_matrix,rotation="none")
29 # Maximum Likelihood
30 mle_nonrotated <- fa(cov_matrix, 4, rotate = "none", fm="mle")
31 # Unweighted Least squares
32 uls_nonrotated <- fa(cov_matrix, 4, rotate = "none", fm="uls")
33
34 #----- / CON ROTACION / -----
35 #Resultados de principal axis con rotacion:
36 summary(pa_rotated)
37
38 #Resultados de unweighted least squares con rotacion:
39 summary(uls_rotated)
40
41 #Resultados de weighted least squares con rotacion:
42 summary(wls_rotated)
43
44 #Resultados de maximum likelihood con rotacion:
45 summary(mle_rotated)
46
47 #----- / SIN ROTACION / -----
48 #Resultados de weighted least squares sin rotacion:
49 summary(wls_nonrotated)
50
51 #Resultados de principal axis sin rotacion:
52 summary(pa_nonrotated)
53
54 #Resultados de minres sin rotacion:
55 summary(minres_nonrotated)
56
57 #Resultados de maximum likelihood sin rotacion:
58 summary(mle_nonrotated)
59
60 #Resultados de unweighted least squares sin rotacion:
61 summary(uls_nonrotated)
62
63 ##                                CFA                                ##
64
65 # Hacemos nuestra hipotesis del modelo
66 HS.model <- ' visual ~ x1 + x2 + x3
67             textual ~ x4 + x5 + x6
68             speed ~ x7 + x8 + x9 '
69
70 # Comprobamos la hipotesis
71 fit <- cfa(HS.model, data=HolzingerSwineford1939)
72
73 # Resultados de CFA
74 summary(fit, fit.measures=TRUE)
75

```

Figura 1: Código de ejemplo en R

Análisis de los resultados de fa

Resultados

Factor analysis with Call: `fa(r = cov_matrix, nfactors = 4, rotate = "varimax", fm = "pa")`

Test of the hypothesis that 4 factors are sufficient. The degrees of freedom for the model is 186 and the objective function was 1.72

The root mean square of the residuals (RMSA) is 0.04 The df corrected root mean square of the residuals is 0.05

- La primera línea nos dice la llamada a la función `fa`.
- La segunda nos da la información sobre si los 4 factores son suficientes.
- La penúltima línea es la raíz cuadrada de la media de los residuos RMSA.
- La última línea es la raíz cuadrada de la media de los residuos estandarizados al cuadrado.

Resultados

Si hacemos una llamada con el objeto obtenido usando la función `fa`, por ejemplo `pa_rotate`, obtenemos entre otras cosas una matriz de pesos que nos explica con qué pesos los factores explican cada variable de forma que podríamos poner cada variable menos su media como una combinación lineal de los factores con los pesos obtenidos.

Análisis de los resultados de fa

Resultados

Standardized loadings (pattern matrix) based upon correlation matrix						
	PA1	PA3	PA2	PA4	h2	u2 com
VisualPerception	0.15	0.68	0.20	0.15	0.55	0.45 1.4
Cubes	0.11	0.45	0.08	0.08	0.23	0.77 1.3
PaperFormBoard	0.15	0.55	-0.01	0.11	0.34	0.66 1.2
Flags	0.23	0.53	0.09	0.07	0.35	0.65 1.5
GeneralInformation	0.73	0.19	0.22	0.14	0.64	0.36 1.4
ParagraphComprehension	0.76	0.21	0.07	0.23	0.68	0.32 1.4
SentenceCompletion	0.81	0.19	0.15	0.07	0.73	0.27 1.2
wordClassification	0.57	0.34	0.23	0.14	0.51	0.49 2.2
wordMeaning	0.81	0.20	0.05	0.22	0.74	0.26 1.3
Addition	0.17	-0.10	0.82	0.16	0.74	0.26 1.2
code	0.18	0.10	0.54	0.37	0.47	0.53 2.1
countingDots	0.02	0.20	0.71	0.09	0.55	0.45 1.2
StraightCurvedCapitals	0.18	0.42	0.54	0.08	0.51	0.49 2.2
wordRecognition	0.21	0.05	0.08	0.56	0.36	0.64 1.3
NumberRecognition	0.12	0.12	0.08	0.52	0.31	0.69 1.3
FigureRecognition	0.07	0.42	0.06	0.52	0.45	0.55 2.0
ObjectNumber	0.14	0.06	0.22	0.58	0.41	0.59 1.4
NumberFigure	0.02	0.31	0.34	0.45	0.41	0.59 2.7
Figureword	0.15	0.25	0.18	0.35	0.23	0.77 2.8
Deduction	0.38	0.42	0.10	0.29	0.42	0.58 2.9
NumericalPuzzles	0.18	0.40	0.43	0.21	0.42	0.58 2.8
ProblemReasoning	0.37	0.41	0.13	0.29	0.40	0.60 3.0
SeriesCompletion	0.37	0.52	0.23	0.22	0.51	0.49 2.7
ArithmeticProblems	0.36	0.19	0.49	0.29	0.49	0.51 2.9

Figura 2: Resultados del objeto fa

El valor h2 es la suma de todos los pesos al cuadrado para esa variable, u2 nos da una medida de la unicidad y com una medida de la complejidad para expresar esa variable con los factores obtenidos.

Análisis de los resultados de VSS

En este caso estamos llamando a la función VSS con unos datos simulados aleatorios de 24 variables.

Resultados

```
very simple structure of vss of 24 simple structure variables
call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,
      n.obs = n.obs, plot = plot, title = title, use = use, cor = cor)
VSS complexity 1 achieves a maximum of 0.8 with 5 factors
VSS complexity 2 achieves a maximum of 0.83 with 8 factors

The velicer MAP achieves a minimum of 0.01 with 2 factors
BIC achieves a minimum of -1226.66 with 2 factors
sample size adjusted BIC achieves a minimum of -499.8 with 2 factors

Statistics by number of factors
vss1 vss2 map dof chisq prob sqresid fit RMSEA BIC SABIC complex echisq SRMR eCRMS eBIC
1 0.46 0.00 0.0431 252 1776 4.1e-227 30.3 0.46 0.11 210 1010 1.0 7045 0.160 0.167 5479
2 0.80 0.81 0.0056 229 196 9.4e-01 10.7 0.81 0.00 -1227 -500 1.0 166 0.025 0.027 -1257
3 0.80 0.81 0.0080 207 170 9.7e-01 10.3 0.82 0.00 -1116 -459 1.1 139 0.022 0.026 -1147
4 0.80 0.82 0.0105 186 143 9.9e-01 9.9 0.82 0.00 -1012 -422 1.1 114 0.020 0.025 -1042
5 0.80 0.82 0.0134 166 119 1.0e+00 9.5 0.83 0.00 -913 -386 1.2 91 0.018 0.023 -941
6 0.75 0.83 0.0169 147 99 1.0e+00 9.1 0.84 0.00 -814 -348 1.3 76 0.017 0.023 -837
7 0.80 0.83 0.0209 129 85 1.0e+00 8.8 0.84 0.00 -716 -307 1.3 63 0.015 0.022 -738
8 0.77 0.83 0.0255 112 71 1.0e+00 8.3 0.85 0.00 -625 -270 1.3 51 0.014 0.021 -645
```

Figura 3: Resultados del objeto VSS

Podemos observar que con summary visualizamos todas las medidas de bonanza del modelo como por ejemplo con chisq (chi-cuadrado) tal y como explicamos en la etapa teórica. Además podemos observar que nos indica que el modelo con mejor ajuste y menor complejidad es el que tiene 2 factores.

Análisis de los resultados de VSS

Además podemos observar que la función hace un plot de la gráfica que compara la medida obtenida para cada tipo de medida y número de factores.

Resultados

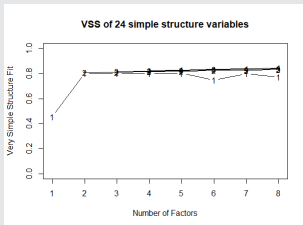


Figura 4: Resultados del plot de VSS

Como podemos observar el modelo obtiene una mejora al usar dos factores y no mejora al incrementar dicho número, por lo que los resultados obtenidos son consistentes.

Análisis de los resultados de factanal

Resultados

Factor analysis with Call: `fa(r = cov_matrix, nfactors = 4, rotate = "none", fm = "uls")`

Test of the hypothesis that 4 factors are sufficient. The degrees of freedom for the model is 186 and the objective function was 1.72

The root mean square of the residuals (RMSA) is 0.04 The df corrected root mean square of the residuals is 0.05

Como podemos comprobar los resultados arrojados son iguales a los de la función `fa`, por lo que la interpretación es la misma.

Análisis de los resultados de cfa

En este caso estamos empleando el dataset HolzingerSwineford1939 compuesto por 9 variables que reflejan los patrones de respuestas de niños a unos tests. Se cree que el modelo puede ser explicado mediante 3 factores:

- $visual = x_1 + x_2 + x_3$
- $textual = x_4 + x_5 + x_6$
- $celeridad = x_7 + x_8 + x_9$

De esta forma podemos expresar en R este modelo con sintaxis lavaan:

```
> HS.model <- ' visual  =~ x1 + x2 + x3  
+          textual =~ x4 + x5 + x6  
+          speed   =~ x7 + x8 + x9 '
```

Figura 5: Código para expresar el modelo

Análisis de los resultados de cfa

Resultados

Si hacemos summary del objeto obtenido nos arroja la siguiente información:

```
lavaan 0.6-3 ended normally after 35 iterations

optimization method                   NLMINB
Number of free parameters             21

Number of observations                 301

Estimator                             ML
Model Fit Test Statistic              85.306
Degrees of freedom                    24
P-value (Chi-square)                  0.000

Parameter Estimates:

Information                           Expected
Information saturated (h1) model      Structured
Standard Errors                       Standard

Latent variables:
      Estimate Std.Err z-value P(>|z|)
visual =~
  x1          1.000
  x2          0.554    0.100   5.554   0.000
  x3          0.729    0.109   6.685   0.000
textual =~
  x4          1.000
  x5          1.113    0.065  17.014   0.000
  x6          0.926    0.055  16.703   0.000
speed =~
  x7          1.000
  x8          1.180    0.165   7.152   0.000
  x9          1.082    0.151   7.155   0.000
```

```
Covariances:
      Estimate Std.Err z-value P(>|z|)
visual =~
  textual    0.408    0.074   5.552   0.000
  speed      0.262    0.056   4.660   0.000
textual =~
  speed      0.173    0.049   3.518   0.000

Variances:
      Estimate Std.Err z-value P(>|z|)
.x1      0.549    0.114   4.833   0.000
.x2      1.134    0.102  11.146   0.000
.x3      0.844    0.091   9.317   0.000
.x4      0.371    0.048   7.779   0.000
.x5      0.446    0.058   7.642   0.000
.x6      0.356    0.043   8.277   0.000
.x7      0.799    0.081   9.823   0.000
.x8      0.488    0.074   6.573   0.000
.x9      0.566    0.071   8.003   0.000
visual   0.809    0.145   5.564   0.000
textual  0.979    0.112   8.737   0.000
speed    0.384    0.086   4.451   0.000
```

Figura 6: Resultados de summary(cfa)

Los resultados obtenidos son:

Resultados

- Si el método ha convergido y en cuantas iteraciones.
- El número de observaciones en los datos.
- Estimador usado, en este caso máxima verosimilitud.
- Valores estadísticos que nos miden entre otras cosas el error al usar los factores para explicar el modelo.

¿Preguntas?

