Análisis de Componentes Principales

Víctor Morales Oñate

31/07/2017

Planteamiento[1]

Se aplica a tablas de datos donde las filas son considerados como individuos y las columnas como datos cuantitativos.

Más formalmente, se dispone de los valores de p variables y n elementos dispuestos en una matriz $\mathbf{X}$ de dimensión $n \times p$.

Siempre (casi) se usa la matriz centrada y/o estandarizada, los paquetes suelen hacer este trabajo por nosostros. Supongamos que $\mathbf{X}$ ha sido centrada, su matriz de varianza covarianza viene dada por $\frac{1}{n}\mathbf{X}'\mathbf{X}$.

¿Cómo encontrar un espacio de dimensión más reducida que represente adeucadamente los datos?

Notación

Se desea encontrar un subespacio de dimensión menor que p tal que al proyectar sobre él los puntos conserven su estructura con la menor distorsión posible.

Consideremos primero un subespacio de dimensión uno (una recta) obtenida por un conjunto de $p = 2$ variables.

La siguiente figura indica el diagrama de dispersión y una recta que, intuitivamente, proporciona un buen resumen de los datos, ya que las proyecciones de los puntos sobre ella indican aproximadamente la situación de los puntos en el plano.

Si consideramos un punto $\mathbf{x_i}$ y una dirección $\mathbf{a_1} = (a_{11}, \ldots, a_{1p})'$, definida por un vector $\mathbf{a_1}$ de norma unidad, la proyección del punto $\mathbf{x_i}$ sobre esta dirección es el escalar:

$$z_i = a_{11}x_{i1} + \ldots + a_{1p}x_{ip} = \mathbf{a_1}'\mathbf{x_i}$$

y el vector que representa esta proyección será $z_i\mathbf{a_1}$. Llamando r_i a la distancia entre el punto x_i, y su proyección sobre la dirección $\mathbf{a_1}$, este criterio implica:

$$min \sum_{i=1}^{n} r_i^2 = \sum_{i=1}^{n} |\mathbf{x_i} - z_i\mathbf{a_1}|^2$$

donde $|\cdot|$ es la norma euclideana o módulo del vector.

Notemos que al proyectar cada punto sobre la recta se forma un triángulo rectángulo donde la hipotenusa es la distancia al origen del punto al origen, $(\mathbf{x_i}'\mathbf{x_i})^{1/2}$, y los catetos la proyeccion del punto sobre la recta (z_i) y la distancia entre el punto y su proyección (r_i). Por el teorema de Pitágoras, podemos escribir:

$$(\mathbf{x_i}'\mathbf{x_i}) = z_i^2 + r_i^2$$

y sumando esta expresión para todos los puntos, se obtiene:

[1] Teoría obtenida de Peña, D. *Análisis de datos multivariantes* (2002). Referencias de `FactoMineR` vienen de Husson, F. *Exploratory multivariate analysis by example using R* (2017)

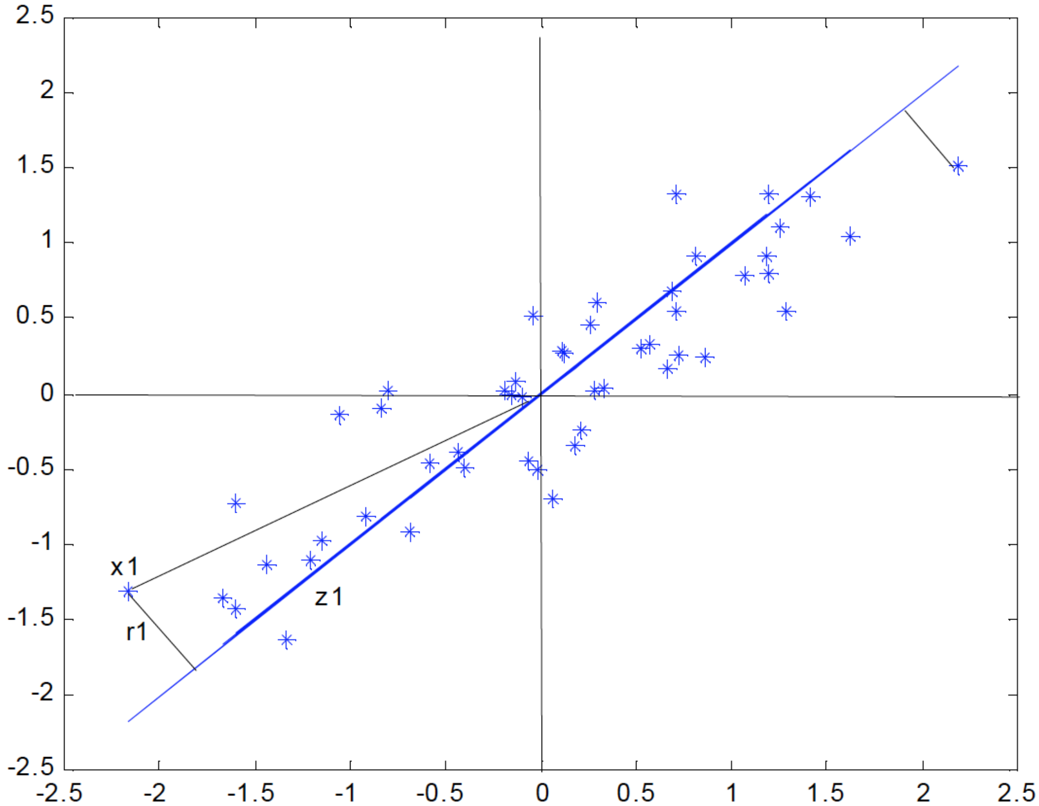


Figure 1: Ejemplo de la recta que minimiza las distancias ortogonales de los puntos a ella.

$$\sum_{i=1}^{n}(\mathbf{x_i'x_i}) = \sum_{i=1}^{n} z_i^2 + \sum_{i=1}^{n} r_i^2$$

Como el primer miembro es constante, minimizar $\sum_{i=1}^{n} r_i^2$, la suma de las distancias a la recta de todos los puntos, es equivalente a maximizar $\sum_{i=1}^{n} z_i^2$, la suma al cuadrado de los valores de las proyecciones. Como las proyecciones z_i son variables de media cero, **maximizar la suma de sus cuadrados equivale a mazimizar su varianza**.

¿Cómo es eso posible?

Cálculo del primer componente

El primer componente principal será la combinación lineal de las variables originales que tenga varianza máxima. Los valores de este primer componente en los n individuos se representarán por un vector $\mathbf{z_1}$, dado por

$$\mathbf{z_1} = \mathbf{X a_1}$$

Como las variables originales tienen media cero también $\mathbf{z_1}$ tendrá media nula. Su varianza será:

$$Var(\mathbf{z_1}) = \frac{1}{n}\mathbf{z_1^{'} z_1} = \frac{1}{n}\mathbf{a_1^{'} X' X a_1} = \mathbf{a_1^{'} S a_1}$$

donde S es la matriz de varianzas y covarianzas de las observaciones. Para que la maximización de la ecuación anterior tenga solución debemos imponer una restricción al módulo del vector $\mathbf{a_1}$, y, sin pérdida de generalidad, impondremos que $\mathbf{a_1^{'} a_1} = 1$. Usamos para ello el multiplicador de Lagrange

$$M = \mathbf{a_1^{'} S a_1} - \lambda(\mathbf{a_1^{'} a_1} - 1)$$

Se maximiza derivando respecto a los componentes de $\mathbf{a_1}$ e igualando a cero. Entonces

$$\frac{\partial M}{\partial \mathbf{a_1}} = 2\mathbf{S a_1} - 2\lambda \mathbf{a_1} = 0$$

cuya solución es:

$$\mathbf{S a_1} = \lambda \mathbf{a_1}$$

que implica que **a1** es un vector propio de la matriz **S**, y λ su correspondiente valor propio. Para determinar qué valor propio de **S** es la solución de la ecuación tendremos en cuenta que, multiplicando por la izquierda por $\mathbf{a_1'}$ esta ecuación,

$$\mathbf{a_1^{'} S a_1} = \lambda \mathbf{a_1^{'} a_1} = \lambda$$

y concluimos, que λ es la varianza de $\mathbf{z_1}$. Como esta es la cantidad que queremos maximizar, λ será el mayor valor propio de la matriz **S**. Su vector asociado, **a1**, define los coeficientes de cada variable en el primer componente principal.

En R

El siguiente conjunto de datos corresponde a calificaciones de 20 estudiantes en 5 materias Ciencias Natuales (CNa), Matemáticas (Mat), Francés (Fra), Latín (Lat) y Literatura (Lit)

```
CNa <- c(7,5,5,6,7,4,5,5,6,6,6,5,6,8,6,4,6,6,6,7)
Mat <- c(7,5,6,8,6,4,5,6,5,5,7,5,6,7,7,3,4,6,5,7)
Fra <- c(5,6,5,5,6,6,5,5,7,6,5,4,6,8,5,4,7,7,4,6)
Lat <- c(5,6,7,6,7,7,5,5,6,6,6,5,6,8,6,4,8,7,4,7)
Lit <- c(6,5,5,6,6,6,6,5,6,6,5,4,5,8,6,4,7,7,4,6)
Notas <- cbind(CNa,Mat,Fra,Lat,Lit)
Notas
```

```
##       CNa Mat Fra Lat Lit
##  [1,]   7   7   5   5   6
##  [2,]   5   5   6   6   5
##  [3,]   5   6   5   7   5
##  [4,]   6   8   5   6   6
##  [5,]   7   6   6   7   6
##  [6,]   4   4   6   7   6
##  [7,]   5   5   5   5   6
##  [8,]   5   6   5   5   5
##  [9,]   6   5   7   6   6
## [10,]   6   5   6   6   6
## [11,]   6   7   5   6   5
## [12,]   5   5   4   5   4
## [13,]   6   6   6   6   5
## [14,]   8   7   8   8   8
## [15,]   6   7   5   6   6
## [16,]   4   3   4   4   4
## [17,]   6   4   7   8   7
## [18,]   6   6   7   7   7
## [19,]   6   5   4   4   4
## [20,]   7   7   6   7   6
```

Es pertiente empezar por un análisis explotario para tener una mejor perspectiva de los datos:

```
summary(Notas)
```

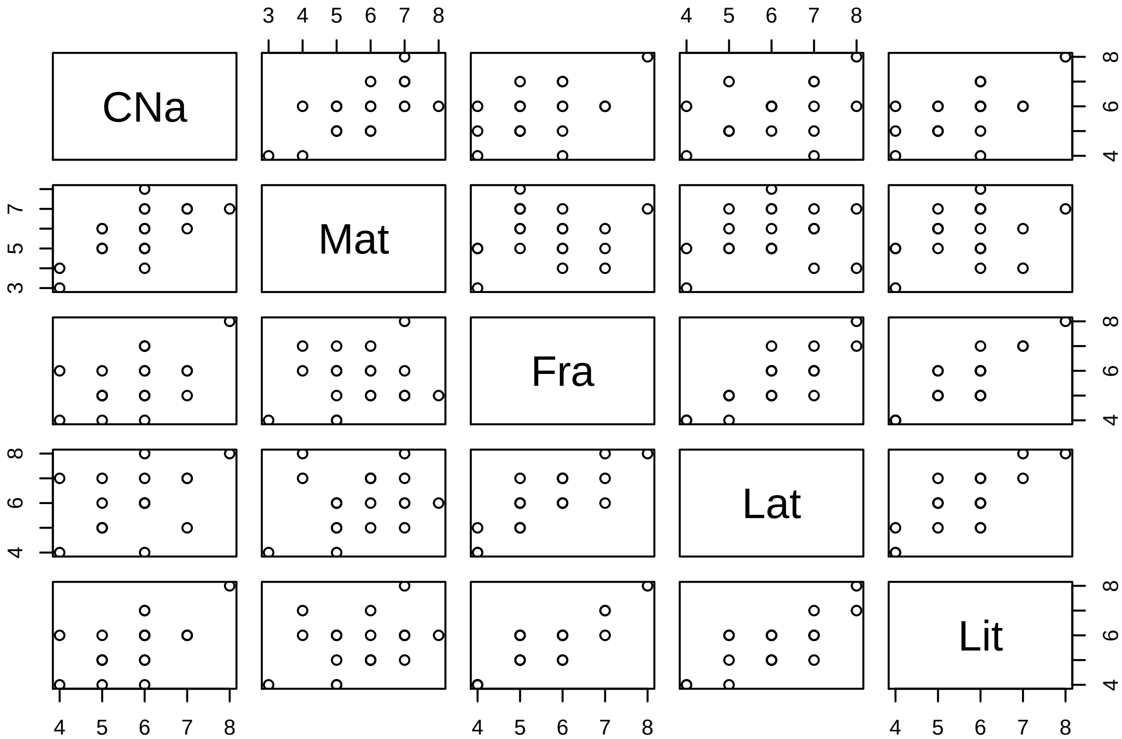
```
##       CNa           Mat           Fra           Lat            Lit
##  Min.   :4.0   Min.   :3.0   Min.   :4.0   Min.   :4.00   Min.   :4.00
##  1st Qu.:5.0   1st Qu.:5.0   1st Qu.:5.0   1st Qu.:5.00   1st Qu.:5.00
##  Median :6.0   Median :6.0   Median :5.5   Median :6.00   Median :6.00
##  Mean   :5.8   Mean   :5.7   Mean   :5.6   Mean   :6.05   Mean   :5.65
##  3rd Qu.:6.0   3rd Qu.:7.0   3rd Qu.:6.0   3rd Qu.:7.00   3rd Qu.:6.00
##  Max.   :8.0   Max.   :8.0   Max.   :8.0   Max.   :8.00   Max.   :8.00
```

Ahora algo gráfico:

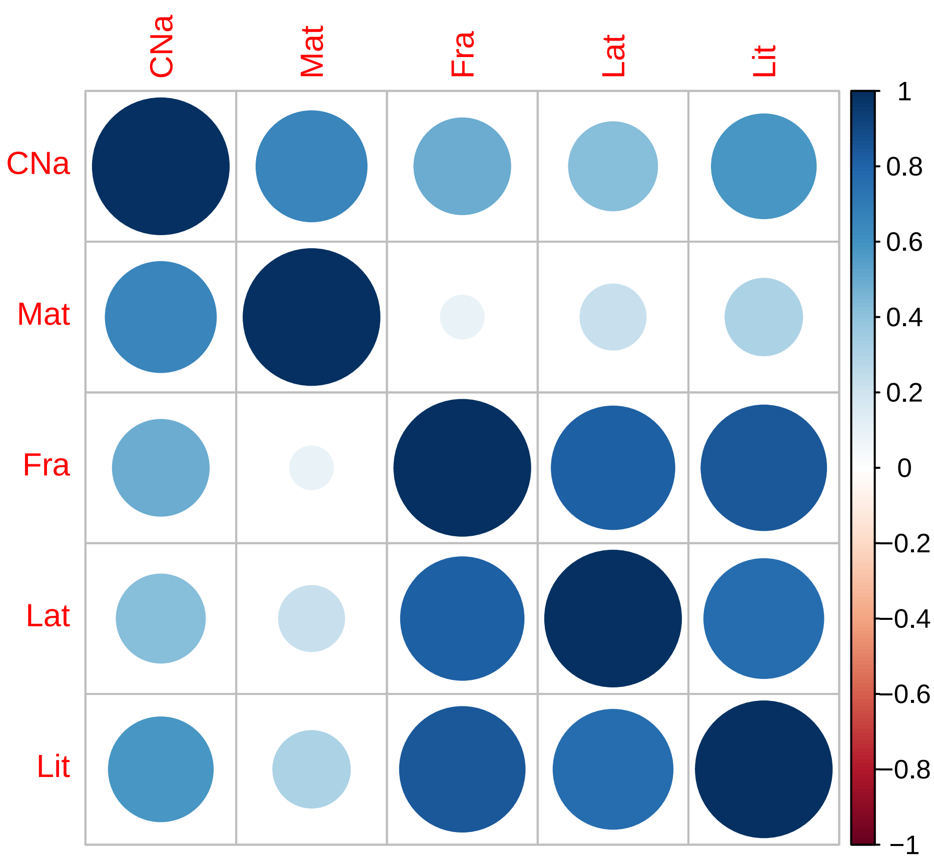
```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
plot(as.data.frame(Notas))
```




```r
corrplot(cor(Notas))
```



Como habíamos visto, los valores propios corresponden la varianzas explicadas de cada componente y los vectores propios son sus direcciones o pesos (*loadings*). Es decir:

```r
fc <- function(x) return((x-mean(x)))
Notasc <- apply(Notas,2,fc) #Datos centrados
S <- cov(Notas*19/20) # Matriz de covarianza
VarLoad <- eigen(S) # valores y vectores propios
VarLoad
```

```
## eigen() decomposition
## $values
## [1] 3.4101493 1.4993717 0.3696656 0.1987624 0.1128010
##
## $vectors
##            [,1]       [,2]       [,3]        [,4]       [,5]
## [1,] -0.3953452  0.3310292  0.6615512 -0.47595392  0.2644611
## [2,] -0.3488288  0.7977236 -0.3708461  0.17998543 -0.2683914
## [3,] -0.4822572 -0.3715412  0.2152088  0.01712248 -0.7633984
## [4,] -0.5040057 -0.2987146 -0.5998378 -0.46491466  0.2842478
## [5,] -0.4852081 -0.1636565  0.1367586  0.72431643  0.4409676
```

Ahora podemos calcular los puntajes de los componetes por individuo:

```r
Notasc%*%VarLoad$vectors #scores
```

```
##              [,1]        [,2]        [,3]        [,4]         [,5]
##  [1,] -0.27915411  1.91357104  0.86033137  0.39423402  0.282362039
##  [2,]  0.70813894 -0.85053394 -0.24246638 -0.18593762 -0.629895638
##  [3,]  0.33756163  0.02001625 -1.42835911 -0.48798933  0.149359151
##  [4,] -0.73664347  2.08155078 -0.77190377  0.58525870  0.033757315
##  [5,] -1.42059398  0.14687700  0.24671081 -0.69845826  0.355850690
##  [6,]  0.46309908 -2.44165783 -0.99625060  0.36943263  0.099250083
##  [7,]  1.20919379 -0.34393456  0.27892119  0.98617099  0.290222569
##  [8,]  1.34557309  0.61744548 -0.22868359  0.44183999 -0.419136475
##  [9,] -0.65467152 -1.05470241  0.77105231  0.07954737 -0.687865235
## [10,] -0.17241431 -0.68316118  0.55584349  0.06242489  0.075533142
## [11,]  0.09739341  1.44748367 -0.53781626 -0.31904316 -0.138818927
## [12,]  2.66186718  0.35491959 -0.20980490 -0.47958435  0.171685659
## [13,] -0.03603501  0.27821886  0.04823871 -0.48190611 -0.633825902
## [14,] -4.60370426 -0.09348019  0.64151305  0.02353642 -0.008693228
## [15,] -0.38781468  1.28382720 -0.40105762  0.40527327  0.302148716
## [16,]  4.25887565 -1.27284218  0.47017392  0.10131336  0.159759512
## [17,] -1.79906227 -2.61351169  0.07898156 -0.30595095  0.589989436
## [18,] -1.99271412 -0.71934991 -0.06287296  0.51893457 -0.231041180
## [19,]  2.77052775  0.98466343  1.05158410 -0.49062361  0.151898983
## [20,] -1.76942278  0.94460058 -0.12413533 -0.51847283  0.087459289
```

El porcentaje de la varianza explicada por cada componente es:

```r
VarLoad$values/(sum(VarLoad$values))
```

```
## [1] 0.60996276 0.26818793 0.06612094 0.03555201 0.02017636
```

Verifiquemos nuestros resultados usando la función `princomp` de R:

```r
result1 <- princomp(Notas,cor=FALSE)
summary(result1)
```

```
## Importance of components:
##                           Comp.1    Comp.2     Comp.3     Comp.4
## Standard deviation     1.8946321 1.2562985 0.62379622 0.45740967
```

```
## Proportion of Variance 0.6099628 0.2681879 0.06612094 0.03555201
## Cumulative Proportion  0.6099628 0.8781507 0.94427162 0.97982364
##                            Comp.5
## Standard deviation     0.34458364
## Proportion of Variance 0.02017636
## Cumulative Proportion  1.00000000
```

```
result1$loadings
```

```
##
## Loadings:
##     Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## CNa -0.395  0.331  0.662 -0.476  0.264
## Mat -0.349  0.798 -0.371  0.180 -0.268
## Fra -0.482 -0.372  0.215        -0.763
## Lat -0.504 -0.299 -0.600 -0.465  0.284
## Lit -0.485 -0.164  0.137  0.724  0.441
##
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## SS loadings       1.0    1.0    1.0    1.0    1.0
## Proportion Var    0.2    0.2    0.2    0.2    0.2
## Cumulative Var    0.2    0.4    0.6    0.8    1.0
```

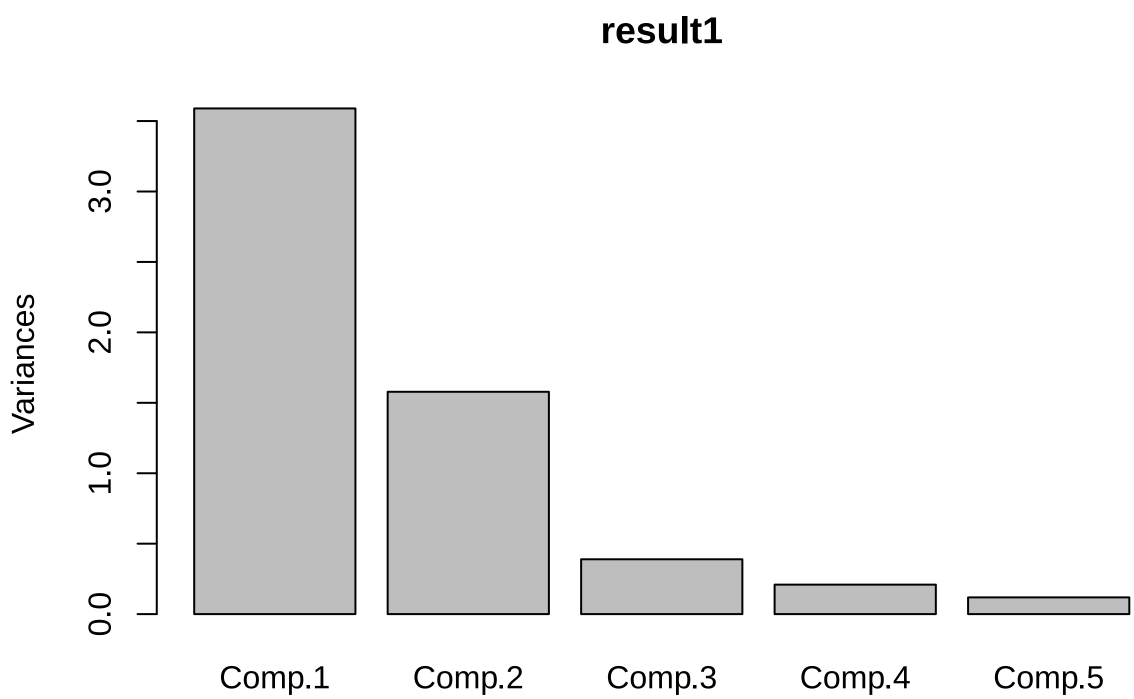
```
result1$sdev
```

```
##    Comp.1    Comp.2    Comp.3    Comp.4    Comp.5
## 1.8946321 1.2562985 0.6237962 0.4574097 0.3445836
```

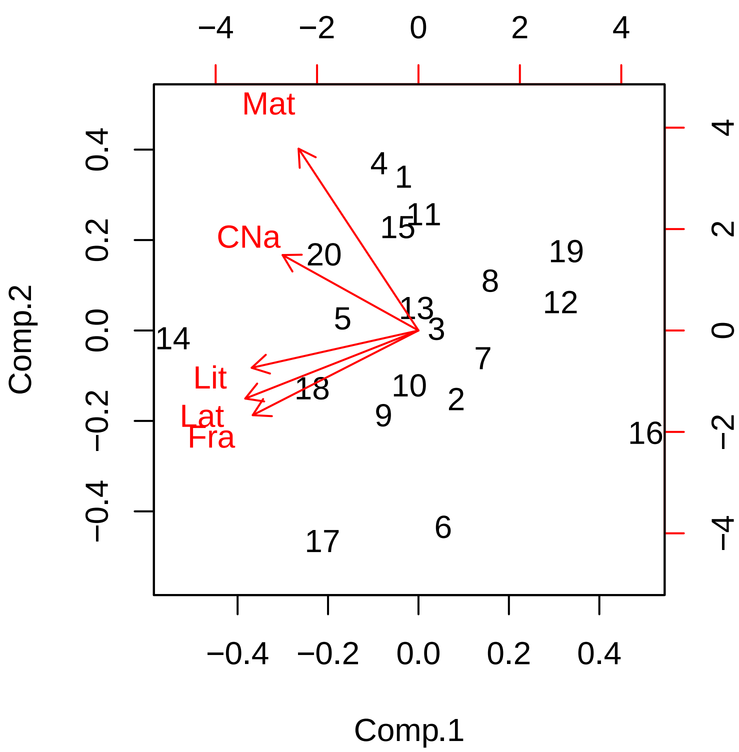
```
str(result1)
```

```
## List of 7
##  $ sdev    : Named num [1:5] 1.895 1.256 0.624 0.457 0.345
##   ..- attr(*, "names")= chr [1:5] "Comp.1" "Comp.2" "Comp.3" "Comp.4" ...
##  $ loadings: loadings [1:5, 1:5] -0.395 -0.349 -0.482 -0.504 -0.485 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:5] "CNa" "Mat" "Fra" "Lat" ...
##   .. ..$ : chr [1:5] "Comp.1" "Comp.2" "Comp.3" "Comp.4" ...
##  $ center  : Named num [1:5] 5.8 5.7 5.6 6.05 5.65
##   ..- attr(*, "names")= chr [1:5] "CNa" "Mat" "Fra" "Lat" ...
##  $ scale   : Named num [1:5] 1 1 1 1 1
##   ..- attr(*, "names")= chr [1:5] "CNa" "Mat" "Fra" "Lat" ...
##  $ n.obs   : int 20
##  $ scores  : num [1:20, 1:5] -0.279 0.708 0.338 -0.737 -1.421 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:5] "Comp.1" "Comp.2" "Comp.3" "Comp.4" ...
##  $ call    : language princomp(x = Notas, cor = FALSE)
##  - attr(*, "class")= chr "princomp"
```

```
plot(result1)
```



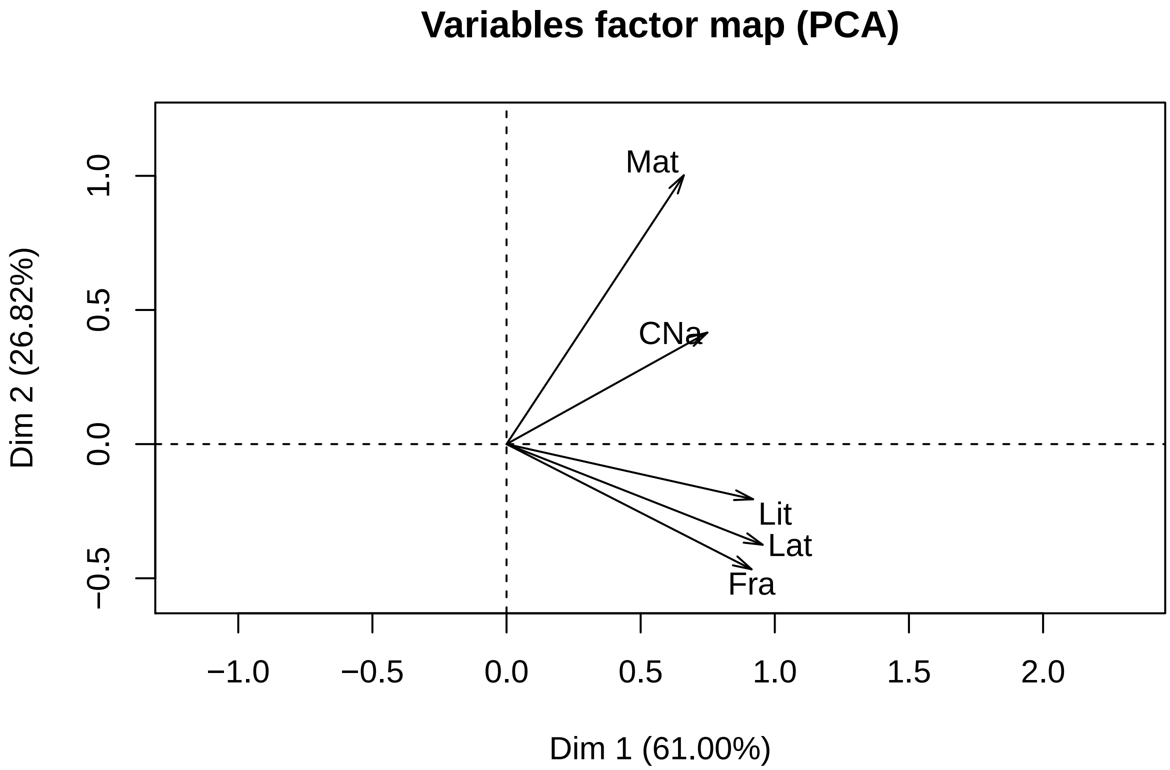

```r
biplot(result1)
```



FactoMineR

En este paquete tenemos la función `PCA` que nos brinda la misma información anterior además de otros temas interesantes:

```r
library(FactoMineR)
result <- PCA(Notas,graph=FALSE,scale.unit = FALSE)
plot(result,choix="var")
```




```r
summary(result)
```

```
##
## Call:
## PCA(X = Notas, scale.unit = FALSE, graph = FALSE)
##
##
## Eigenvalues
##                        Dim.1   Dim.2   Dim.3   Dim.4   Dim.5
## Variance               3.590   1.578   0.389   0.209   0.119
## % of var.             60.996  26.819   6.612   3.555   2.018
## Cumulative % of var.  60.996  87.815  94.427  97.982 100.000
##
## Individuals (the 10 first)
##         Dist    Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr
## 1   |  2.171 |  0.279  0.109  0.017 |  1.914 11.600  0.777 |  0.860  9.511
## 2   |  1.310 | -0.708  0.698  0.292 | -0.851  2.292  0.422 | -0.242  0.755
## 3   |  1.554 | -0.338  0.159  0.047 |  0.020  0.001  0.000 | -1.428 26.216
## 4   |  2.411 |  0.737  0.756  0.093 |  2.082 13.726  0.745 | -0.772  7.656
## 5   |  1.648 |  1.421  2.811  0.743 |  0.147  0.068  0.008 |  0.247  0.782
## 6   |  2.705 | -0.463  0.299  0.029 | -2.442 18.887  0.815 | -0.996 12.753
## 7   |  1.648 | -1.209  2.037  0.539 | -0.344  0.375  0.044 |  0.279  1.000
## 8   |  1.617 | -1.346  2.522  0.692 |  0.617  1.208  0.146 | -0.229  0.672
## 9   |  1.617 |  0.655  0.597  0.164 | -1.055  3.524  0.425 |  0.771  7.639
## 10  |  0.903 |  0.172  0.041  0.036 | -0.683  1.479  0.573 |  0.556  3.970
```

```
##       cos2
## 1    0.157 |
## 2    0.034 |
## 3    0.845 |
## 4    0.102 |
## 5    0.022 |
## 6    0.136 |
## 7    0.029 |
## 8    0.020 |
## 9    0.227 |
## 10   0.379 |
##
## Variables
##         Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr   cos2
## CNa |  0.749 15.630  0.584 |  0.416 10.958  0.180 |  0.413 43.765  0.177 |
## Mat |  0.661 12.168  0.289 |  1.002 63.636  0.665 | -0.231 13.753  0.035 |
## Fra |  0.914 23.257  0.732 | -0.467 13.804  0.191 |  0.134  4.631  0.016 |
## Lat |  0.955 25.402  0.731 | -0.375  8.923  0.113 | -0.374 35.981  0.112 |
## Lit |  0.919 23.543  0.822 | -0.206  2.678  0.041 |  0.085  1.870  0.007 |
```

```
sum(sqrt(result$eig[,1]))
```

```
## [1] 4.57672
```

```
result$var
```

```
## $coord
##         Dim.1      Dim.2       Dim.3        Dim.4       Dim.5
## CNa 0.7490336  0.4158715  0.41267316  0.217705924 -0.09112898
## Mat 0.6609022  1.0021789 -0.23133242 -0.082327077  0.09248328
## Fra 0.9137000 -0.4667667  0.13424645 -0.007831989  0.26305459
## Lat 0.9549054 -0.3752747 -0.37417653  0.212656462 -0.09794715
## Lit 0.9192908 -0.2056014  0.08530952 -0.331309338 -0.15195023
##
## $cor
##         Dim.1      Dim.2       Dim.3       Dim.4       Dim.5
## CNa 0.7644793  0.4244471  0.42118278  0.22219518 -0.09300813
## Mat 0.5378346  0.8155617 -0.18825566 -0.06699682  0.07526183
## Fra 0.8557585 -0.4371671  0.12573332 -0.00733533  0.24637320
## Lat 0.8549488 -0.3359921 -0.33500884  0.19039621 -0.08769433
## Lit 0.9069054 -0.2028314  0.08416016 -0.32684569 -0.14990305
##
## $cos2
##         Dim.1      Dim.2       Dim.3        Dim.4       Dim.5
## CNa 0.5844285 0.18015533 0.177394933 4.937070e-02 0.008650511
## Mat 0.2892661 0.66514083 0.035440192 4.488575e-03 0.005664343
## Fra 0.7323225 0.19111504 0.015808868 5.380707e-05 0.060699751
## Lat 0.7309374 0.11289068 0.112230921 3.625072e-02 0.007690295
## Lit 0.8224775 0.04114056 0.007082933 1.068281e-01 0.022470923
##
## $contrib
##        Dim.1     Dim.2     Dim.3       Dim.4     Dim.5
## CNa 15.62978 10.958035 43.765004 22.65321318  6.993969
## Mat 12.16815 63.636291 13.752686  3.23947557  7.203394
## Fra 23.25720 13.804288  4.631484  0.02931794 58.277708
```

```
## Lat 25.40218  8.923042 35.980534 21.61456435  8.079682
## Lit 23.54269  2.678344  1.870292 52.46342895 19.445246
```

```r
loadings<-sweep(result$var$coord,2,sqrt(result$eig[1:5,1]),FUN="/")


result$var$coord # correlacion entre las variables y los componentes
```

```
##         Dim.1      Dim.2       Dim.3        Dim.4       Dim.5
## CNa 0.7490336  0.4158715  0.41267316  0.217705924 -0.09112898
## Mat 0.6609022  1.0021789 -0.23133242 -0.082327077  0.09248328
## Fra 0.9137000 -0.4667667  0.13424645 -0.007831989  0.26305459
## Lat 0.9549054 -0.3752747 -0.37417653  0.212656462 -0.09794715
## Lit 0.9192908 -0.2056014  0.08530952 -0.331309338 -0.15195023
```

```r
result$eig # Descomposicion de la varianza por componente
```

```
##        eigenvalue percentage of variance cumulative percentage of variance
## comp 1  3.5896308              60.996276                          60.99628
## comp 2  1.5782860              26.818793                          87.81507
## comp 3  0.3891217               6.612094                          94.42716
## comp 4  0.2092236               3.555201                          97.98236
## comp 5  0.1187379               2.017636                         100.00000
```

```r
result$ind$dist # Distancias de los individuos al centro de la nube
```

```
##         1         2         3         4         5         6         7 
## 2.1714051 1.3095801 1.5540270 2.4114311 1.6477257 2.7046257 1.6477257 
##         8         9        10        11        12        13        14 
## 1.6170962 1.6170962 0.9027735 1.5858752 2.7413500 0.8455767 4.6491935 
##        15        16        17        18        19        20 
## 1.4882876 4.4738127 3.2426841 2.1943108 3.1646485 2.0772578
```

```r
result$ind$contrib # Contribucion de los individuos a la construccion de los componentes
```

```
##           Dim.1        Dim.2       Dim.3       Dim.4        Dim.5
## 1   0.108544611 11.600414101  9.51077807  3.71421905  3.357324542
## 2   0.698485136  2.291751954  0.75541845  0.82621650 16.707747628
## 3   0.158718070  0.001269257 26.21557252  5.69088729  0.939386641
## 4   0.755848757 13.726453207  7.65615735  8.18568608  0.047986216
## 5   2.810995563  0.068342664  0.78209751 11.65843449  5.332321556
## 6   0.298722577 18.886605793 12.75327489  3.26159349  0.414803550
## 7   2.036629526  0.374745094  0.99964903 23.24147933  3.546852011
## 8   2.521940300  1.207762516  0.67197716  4.66540517  7.397613193
## 9   0.596990078  3.524067250  7.63927623  0.15122062 19.924499760
## 10  0.041406337  1.478531825  3.96999144  0.09312685  0.240245799
## 11  0.013212328  6.637608850  3.71665612  2.43252994  0.811480465
## 12  9.869450648  0.399065565  0.56560830  5.49653928  1.241219950
## 13  0.001808713  0.245220882  0.02990032  5.54988754 16.916895709
## 14 29.521270989  0.027683660  5.28804968  0.01323854  0.003182313
## 15  0.209492606  5.221526175  2.06679824  3.92514087  3.844343734
## 16 25.264466770  5.132552812  2.84054449  0.24529731  1.074766582
## 17  4.508297917 21.638801643  0.08015599  2.23698436 14.657812964
## 18  5.531083508  1.639323628  0.05079400  6.43553319  2.247809449
## 19 10.691662101  3.071566546 14.20929560  5.75249447  0.971606552
## 20  4.360973464  2.826706580  0.19800462  6.42408564  0.322101386
```

```r
result$var$contrib # Contribucion de las variables a la construccion de los componentes
```

```
##        Dim.1     Dim.2     Dim.3       Dim.4     Dim.5
## CNa 15.62978 10.958035 43.765004 22.65321318  6.993969
## Mat 12.16815 63.636291 13.752686  3.23947557  7.203394
## Fra 23.25720 13.804288  4.631484  0.02931794 58.277708
## Lat 25.40218  8.923042 35.980534 21.61456435  8.079682
## Lit 23.54269  2.678344  1.870292 52.46342895 19.445246
```

```r
result$var$cos2
```

```
##         Dim.1      Dim.2       Dim.3        Dim.4       Dim.5
## CNa 0.5844285 0.18015533 0.177394933 4.937070e-02 0.008650511
## Mat 0.2892661 0.66514083 0.035440192 4.488575e-03 0.005664343
## Fra 0.7323225 0.19111504 0.015808868 5.380707e-05 0.060699751
## Lat 0.7309374 0.11289068 0.112230921 3.625072e-02 0.007690295
## Lit 0.8224775 0.04114056 0.007082933 1.068281e-01 0.022470923
```