¿Cómo utilizar R para realizar un análisis de componentes principales?

Reducción de dimensión en R

Alondra E. Matos Mendoza

6 de diciembre de 2022

|| Introducción

La reducción de dimensión facilita el análisis estadístico multivariado, pues permite manejar una representación reducida de los datos en cuanto a volumen,sin perder la integridad de los datos originales.

El análisis de componentes principales se centra en proyectar los datos en otras direcciones que definen un nuevo espacio coordenado, revelendo relaciones que no se observan en la escala original. Dichas direcciones vienen dadas por la estructura de covarianzas de los datos y son combinaciones lineales de las variables originales que rotan el sistema de coordenadas original en menos dimensiones.

El objetivo del presente y trabajo es realizar un análisis de componentes principales en el software estadístico R.

La base de datos a analizar se obtuvo de la evaluación de catadores a distintos vinos. Cada catador calificó los siguientes aspectos: Costo, Tamaño, Alcohol, Reputación, Color, Aroma y Sabor. La calificación mínima es 0, mientras que la máxima es 100.Por ejemplo, para la variable Alcohol, el valor 0 corresponde al vino con menos grados de alcohol y el valor 100 corresponde al de más grados de alcohol (de acuerdo con el criterio del catador). Para Color, 0 es casi transparente y 100 es rojo intenso. Para Aroma, 0 es muy suave y 100 es muy fuerte. Para Sabor, 0 es muy suave y 100 es muy amargo. La variable Sexo señala si el catador es hombre (1) o mujer (2).

|| Marco teórico

Consideremos que tenemos un vector $X = (X_1, X_2, ..., X_r)$ tal que su vector de medias es μ_X y su matriz de covarianza es $\Sigma_X X$. Este vector representaría los atributos de un registro. La intención es reemplazar las variables (no ordenadas y correlacionadas) $X_1, X_2, ..., X_r$ por un conjunto de t proyecciones (ordenadas y no correlacionadas) $Y_1, ..., Y_t$ con $t \leq r$, es decir, para $j = 1, 2, ..., t$

$$Y_j = b_j' X = b_{j_1} X_1 + \ + b_{j_r} X_r$$

donde queremos minimizar la pérdida de información por el reemplazo de las variables.

La *información* es interpretada como el *total de variación* de las variables originales, es decir

$$\sum_{j=1}^{r} Var(X_j) = traza(\Sigma_{XX})$$

Existen vectores U y una matriz diagonal Λ tal que

$$\Sigma_{XX} = U \Lambda U^t \ con \ U^t U = I_r$$

donde la diagonal de la matriz Λ son los eigenvalores $_j$ de la matriz Σ_{XX} y las columnas de U son eigenvectores de Σ_{XX}. Luego

$$traza(\Sigma_{XX}) = traza(\Lambda) = \sum_{j=1}^{r} \lambda_j$$

El $j - ésimo$ vector $b_j = (b_{j1}, ..., b_{jr})^t$ es seleccionado de tal manera que:

1. las primeras t proyecciones Y_j, $j = 1, ..., t$ de X están rankeadas en orden decreciente de sus varianzas, es decir, $Var(Y_1)Var(Y_2)...Var(Y_t)$
2. Y_j está no correlacionado con Y_k para $k < j$

Estas proyecciones Y_j son conocidas como las componentes principales de X.

Una medida del ajuste se obtiene a través de que tanto las primeras t componentes representan de las r variables originales, es decir,

$$\frac{\lambda_{t+1} + ... + \lambda_r}{\lambda_1 + ... + \lambda_r}$$

Si las primeras componentes explican mucho, entonces esta medida debe ser pequeña.

Si la varianza de un componente es prácticmente cero, tal componente representa una combinación lineal de las variables y sugiere presencia de colinearidad.

|| Metodología

Primero, se realiza la lectura de datos en R, eliminando la columna de Sexo (al ser categórica), como se muestra a continuación.

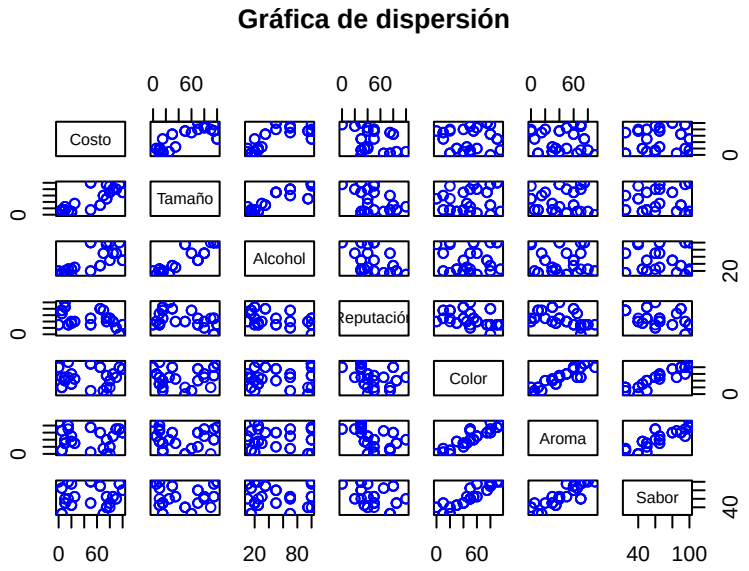
```
#Lectura de los datos
library(readxl)
base <- read_excel("base.xlsx")
head(base)

## # A tibble: 6 x 8
##   Costo Tamaño Alcohol Reputación Color Aroma Sabor  Sexo
##   <dbl>  <dbl>   <dbl>      <dbl> <dbl> <dbl> <dbl> <dbl>
## 1    90     80      70         20    50    70    60     1
## 2    75     95     100         50    55    40    65     1
## 3    10     15      20         85    40    30    50     2
## 4   100     70      50         30    75    60    80     2
## 5    20     10      25         35    30    35    45     1
## 6    50    100     100         30    90    75   100     1

#Elmina la última columna porque no es numérica
base<-base[,-8]
```

Mediante la función *pairs*, se procede a realizar una gráfica de dispersión para visualizar la correlación entre variables.

```
#Matriz de dispersión
pairs(base,col="blue",main="Gráfica de dispersión",cex.main=0.8)
```



Conforme a lo observado en el Gráfico de dispersión, se tiene que la variable Color presenta una relación lineal con la variable Aroma, así como con la variable Sabor. De igual forma, estas dos últimas, Aroma y Sabor, también mantienen una relación lineal entre ellas. Por otro lado, se observa que las variables Tamaño y Alcohol tienen un comportamiento lineal entre sí; mientras que, con el Costo y el Tamaño, se podría insinuar una dependencia lineal, al igual que Costo con Alcohol. Sin embargo, la variable Reputación no aparenta tener una relación con ninguna otra variable en conjunto.

Por lo tanto, se espera que haya una reducción exitosa de dimensionalidad debido a la existencia de asociación lineal entre las variables, como se observó anteriormente.

Se usará la función *eigen* para calcular valores propios y vectores propios de la matriz de correlaciones, obtenida mediante la función *cor*.

```
eig<-eigen(cor(base))   #es una lista que tiene valores y vectores
eig$values              #muestra los  valores propios de la matriz de correlaciones

## [1] 3.31565395 2.54101151 0.62047728 0.26615907 0.14019272 0.07759511 0.03891037

eig$vectors              #muestra los  vectores propios de la matriz de correlaciones

##            [,1]        [,2]       [,3]         [,4]        [,5]        [,6]
## [1,] -0.3146130  0.43846151 0.12052779  0.800109008  0.01508446  0.21437273
## [2,] -0.3750824  0.41252374 0.28130348 -0.225700557 -0.27482915 -0.48371232
## [3,] -0.3390622  0.44641402 0.07384198 -0.549742243  0.25400862  0.43642566
## [4,]  0.3841915 -0.05207974 0.89797543 -0.006526088  0.03082877  0.20257684
## [5,] -0.4234704 -0.35555051 0.28960226  0.035769082 -0.05135256 -0.45020449
## [6,] -0.3924182 -0.40330988 0.03820447 -0.050438074 -0.60591319  0.53075431
## [7,] -0.4055199 -0.38275694 0.09591585  0.052936603  0.69928730  0.06463563
##             [,7]
## [1,]  0.08880089
## [2,] -0.49955610
## [3,]  0.35084816
## [4,] -0.03597714
## [5,]  0.63544203
```

```
## [6,] -0.17468427
## [7,] -0.42879850
```

Se procede a realizar una estructura de información que contenga la desviación estándar, la proporción de la varianza y el acumulado de la proporción de varianza.

```
resumen<-data.frame(componente=(1:length(eig$values)),std=sqrt(eig$values),
                    prop.var=eig$values/sum(eig$values),prop.acum=cumsum(eig$values/sum(eig$values)))
resumen
```

```
##   componente       std    prop.var prop.acum
## 1          1 1.8208937 0.473664850 0.4736648
## 2          2 1.5940551 0.363001644 0.8366665
## 3          3 0.7877038 0.088639611 0.9253061
## 4          4 0.5159061 0.038022724 0.9633288
## 5          5 0.3744232 0.020027531 0.9833564
## 6          6 0.2785590 0.011085016 0.9944414
## 7          7 0.1972571 0.005558624 1.0000000
```

De la tabla se observa que las primeras 3 componentes explican en conjunto un 92.53 % de varianza. Particularmente la primera componente explica un 47.36 %, la segunda un 36.30 % y la tercera un 8.86 %, mientras que las componentes restantes explican individualmente un porcentaje muy pequeño de varianza (menor al 5 %), por lo que sería provechoso prescindir de estas últimas componentes con el fin de reducir de 7 a 3 el número de variables estandarizadas.

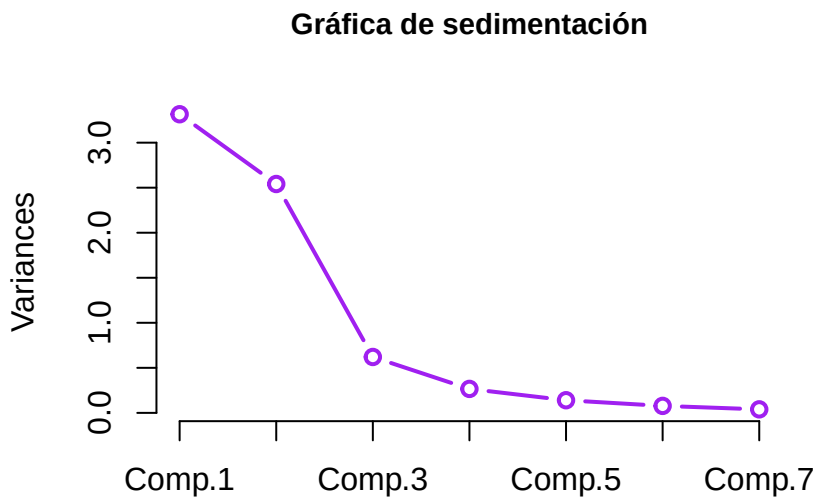
Las componentes principales, también estandarizando las variables, se pueden obtener de manera directa con la función *princomp*. Mediante la función *summary*, se obtiene la desviación estándar, la proporción de la varianza y el acumulado de la proporción de varianza.

```
componentes<-princomp(base,cor=T)
componentes
```

```
## Call:
## princomp(x = base, cor = T)
##
## Standard deviations:
##    Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6    Comp.7
## 1.8208937 1.5940551 0.7877038 0.5159061 0.3744232 0.2785590 0.1972571
##
##  7  variables and  50 observations.
```

A continuación, se presenta la respectiva gráfica de sedimentación, realizada mediante la función *screeplot*.

```
#Gráfica de sedimentación
screeplot(componentes,type="l",main="Gráfica de sedimentación" ,
col="purple",lwd=2,cex.main=0.9)
```



La gráfica de sedimentación es una herramienta visual que se utiliza para la elección del número de componentes principales. De la figura, se observa que los eigenvalores de los primeros tres componentes tienen grandes variaciones, dando así un porcentaje mayor de varianza entre estos y sugiriendo 3 componentes principales, pues a partir del cuarto componente, se comienza a apreciar un comportamiento gradual y menos abrupto en la tendencia descendente de los autovalores.

Los pesos de las componentes de las variables estandarizadas se obtienen de la siguiente manera:

```
componentes$loadings

##
## Loadings:
##            Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7
## Costo       0.315  0.438  0.121  0.800         0.214
## Tamaño      0.375  0.413  0.281 -0.226 -0.275 -0.484 -0.500
## Alcohol     0.339  0.446        -0.550  0.254  0.436  0.351
## Reputación -0.384         0.898                0.203
## Color       0.423 -0.356  0.290               -0.450  0.635
## Aroma       0.392 -0.403               -0.606  0.531 -0.175
## Sabor       0.406 -0.383                0.699        -0.429
##
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7
## SS loadings     1.000  1.000  1.000  1.000  1.000  1.000  1.000
## Proportion Var  0.143  0.143  0.143  0.143  0.143  0.143  0.143
## Cumulative Var  0.143  0.286  0.429  0.571  0.714  0.857  1.000
```

Del resultado anterior se observa que, en cuanto al componente 1, las variables con mayor peso en valor absoluto son Color, Aroma y Sabor. Por otro lado, para la componente 2, las variables con mayor valor absoluto son Costo, Tamaño y Alcohol(la variable Aroma en realidad debería estar en la componente 2, pero por interpretación, se deja en el componente 1). Por último, en la componente 3, la variable que más aporta es Reputación, con un peso bastante alto respecto a los pesos de las demás variables.

La interpretación de cada componente depende de las variables que más la representan, es decir, de aquellas con mayor valor absoluto de peso asociado. Una posible interpretación puede ser *Características Sensitivas*

para la componente 1, *Características comerciales* para la componente 2 y *Características comerciales* para la componente 3, coincidiendo con el nombre de la única variable que se le asoció en el análisis anterior.

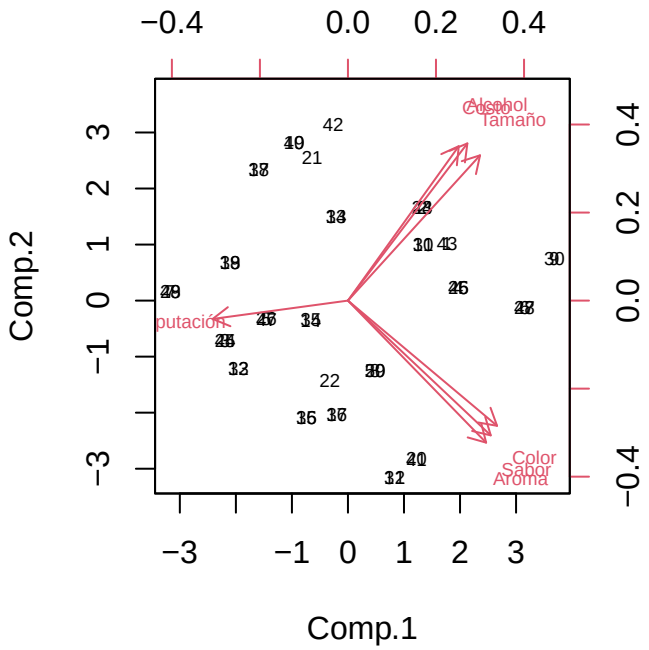
La obtención de la base la base transformada se obtiene de la siguiente manera:

```
base.trasformada<-componentes$scores
head(base.trasformada)

##          Comp.1     Comp.2      Comp.3      Comp.4      Comp.5       Comp.6
## [1,]  1.770053  1.0151257 -0.59285835  0.35046472 -0.92320042  0.332255893
## [2,]  1.325507  1.6574605  0.72085645 -0.53328902  0.05101885 -0.003879955
## [3,] -2.192001 -0.7126463  0.74939388 -0.25479156 -0.07496793 -0.020576862
## [4,]  1.971043  0.2328223  0.03576148  1.07891871 -0.21270078 -0.214001139
## [5,] -1.454869 -0.3292636 -1.27665103 -0.08878611 -0.29962999  0.006819937
## [6,]  3.151179 -0.1289574  0.48258813 -1.09454994  0.11057798 -0.193454145
##           Comp.7
## [1,] -0.22408457
## [2,]  0.01998102
## [3,]  0.09686456
## [4,]  0.03115638
## [5,]  0.14181702
## [6,] -0.10794507
```

Con la función *biplot()* se puede obtener una representación bidimensional de las componentes. Para que las flechas estén en la misma escala que las componentes, se recomienda indicar el argumento $scale = 0$.

```
#biplot
biplot(componentes,cex=0.6,scale=0)
```



Conclusión

R permite realizar un análisis de componentes principales muy completo y con gran rapidez, permite ahorrar tiempo y es fácil de utilizar. Incluso permite mostrar los resultados de manera visual, gracias a sus capacidades avanzadas de gráficos. Además, en R se pueden manejar grandes conjuntos de datos y es gratuito. Así que R es una buena opción para realizar un análisis de componentes principales.