Proyecto1

Juan Luis Solórzano (carnet: 201598)   Micaela Yataz (carnet: 18960)

2025-01-20

#(20 puntos) Análisis de componentes Principales

# • Estudia la matriz de correlación, la agrega y explica lo que observa en ella

En el analisis se realiza a patir de una muestra de datos de 10000 peliculas obtenidos de la plataforma The movie DB. Se evalua la correlacion entre variables. Se presentan las varibales que se incuyen en el analisis:

Indice de popularidad de la película Presupuesto de la película Ingreso de la película Duración de la película Cantidad de géneros que representan la película Cantidad de companias productoras que participaron en la película Cantidad de paises que se llevó a cabo la pelicula Número de votos en la platadorma de la película Promedio de votos en la plataforma de la película Índice de popularidad del elenco de la película Cantidad de personas que actúan en la película Cantidad de actrices en el elenco de la película Cantidad de actores en el elenco de la película.

```r
datos$castWomenAmount<- as.numeric(datos$castWomenAmount)
```

```
## Warning: NAs introduced by coercion
```

```r
datos$castMenAmount<- as.numeric(datos$castMenAmount)
```

```
## Warning: NAs introduced by coercion
```

```r
datos$actorsPopularity <-as.character(datos$actorsPopularity)
datos$actorsPopularity<- strsplit(datos$actorsPopularity, "\\|")
```

```
## Warning in strsplit(datos$actorsPopularity, "\\|"): unable to translate 'Self -
## President, Marvel Studios (archive footage)|Self - Director (archive
## footage)|Self - Executive Producer (archive footage)|Self - Supervising
## Producer, The Falcon and the Winter Soldier|Self - Supervising Producer,
## WandaVision|Self (archive foota...' to a wide string
```

```
## Warning in strsplit(datos$actorsPopularity, "\\|"): input string 9121 is
## invalid
```

```r
datos$actorsPopularity<-lapply(datos$actorsPopularity, as.numeric, use = "pairwise.complete.obs")
```

```
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
```

```
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
## Warning in lapply(datos$actorsPopularity, as.numeric, use =
## "pairwise.complete.obs"): NAs introduced by coercion
```

```r
datos$actorsPopularity<- sapply(datos$actorsPopularity, function(x) if (all(is.na(x))) NA else mean(x,
datos$actorsPopularity<- sapply(datos$actorsPopularity, function(x) {
  if (all(is.na(x))) {
    return(NA)  # Si todos son NA, el promedio es NA
  } else {
    return(mean(x, na.rm = TRUE))  # Calcular la media sin contar los NA
  }
})

sub_datos<-datos[, c("popularity", "budget", "revenue", "runtime", "genresAmount", "productionCoAmount"
matriz_cor <- cor(sub_datos, use = "pairwise.complete.obs" )
determinante<-det(matriz_cor)
```

```r
print(matriz_cor)
```

```
##                             popularity      budget     revenue     runtime
## popularity                 1.000000000  0.16019093  0.16254614  0.03253848
## budget                     0.160190932  1.00000000  0.75745404  0.28149796
## revenue                    0.162546141  0.75745404  1.00000000  0.24865547
## runtime                    0.032538483  0.28149796  0.24865547  1.00000000
## genresAmount               0.039169328  0.19834670  0.13622820  0.02979840
## productionCoAmount         0.005712254  0.13013398  0.05879918  0.15928174
## productionCountriesAmount -0.007465647 -0.03666957 -0.03848513  0.01532322
## voteCount                  0.107669354  0.63025709  0.76825496  0.28236304
## voteAvg                    0.066436274  0.04437110  0.14126181  0.22047585
## castWomenAmount           -0.001778282 -0.04770909 -0.03739659 -0.17118468
## castMenAmount              0.002778599 -0.08359648 -0.06014310 -0.11032001
##                           genresAmount productionCoAmount
## popularity                  0.03916933        0.005712254
## budget                      0.19834670        0.130133978
## revenue                     0.13622820        0.058799182
## runtime                     0.02979840        0.159281740
## genresAmount                1.00000000        0.058615252
## productionCoAmount          0.05861525        1.000000000
## productionCountriesAmount  -0.04200703        0.039177966
## voteCount                   0.10846115        0.100842759
## voteAvg                     0.07472374        0.009449635
## castWomenAmount            -0.09019272        0.131061742
## castMenAmount              -0.10969890       -0.138791288
##                           productionCountriesAmount   voteCount      voteAvg
## popularity                             -0.007465647  0.10766935  0.066436274
## budget                                 -0.036669569  0.63025709  0.044371099
## revenue                                -0.038485129  0.76825496  0.141261812
## runtime                                 0.015323223  0.28236304  0.220475853
## genresAmount                           -0.042007031  0.10846115  0.074723745
## productionCoAmount                      0.039177966  0.10084276  0.009449635
## productionCountriesAmount               1.000000000 -0.04615046 -0.019255309
## voteCount                              -0.046150460  1.00000000  0.262296826
## voteAvg                                -0.019255309  0.26229683  1.000000000
## castWomenAmount                         0.005399121 -0.05361706 -0.076963984
## castMenAmount                           0.433906048 -0.08295119 -0.039721741
##                           castWomenAmount castMenAmount
## popularity                   -0.001778282   0.002778599
## budget                       -0.047709088  -0.083596480
## revenue                       -0.037396591  -0.060143096
## runtime                       -0.171184682  -0.110320010
## genresAmount                  -0.090192725  -0.109698900
## productionCoAmount             0.131061742  -0.138791288
## productionCountriesAmount      0.005399121   0.433906048
## voteCount                     -0.053617059  -0.082951195
## voteAvg                       -0.076963984  -0.039721741
## castWomenAmount                1.000000000  -0.090474251
## castMenAmount                 -0.090474251   1.000000000
```

La determinante es

```r
print(determinante)
```

```
## [1] 0.08460687
```

indicando que las variabes estan relacionadas entre si

#Determina si es posible usar la técnica de análisis factorial para hallar las componentes principales

```r
KMO(as.matrix(sub_datos))
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = as.matrix(sub_datos))
## Overall MSA =  0.67
## MSA for each item = 
##                popularity                    budget                   revenue 
##                      0.81                      0.75                      0.68 
##                   runtime              genresAmount        productionCoAmount 
##                      0.72                      0.64                      0.52 
## productionCountriesAmount                 voteCount                   voteAvg 
##                      0.48                      0.75                      0.55 
##           castWomenAmount             castMenAmount 
##                      0.46                      0.51
```

El indice es de 0.67 lo cual es un valor regular, es suficiente pero no el ideal.

#• Determina si vale la pena aplicar las componentes principales interpretando la prueba de esfericidad de Bartlett

```r
cortest.bartlett(sub_datos)
```

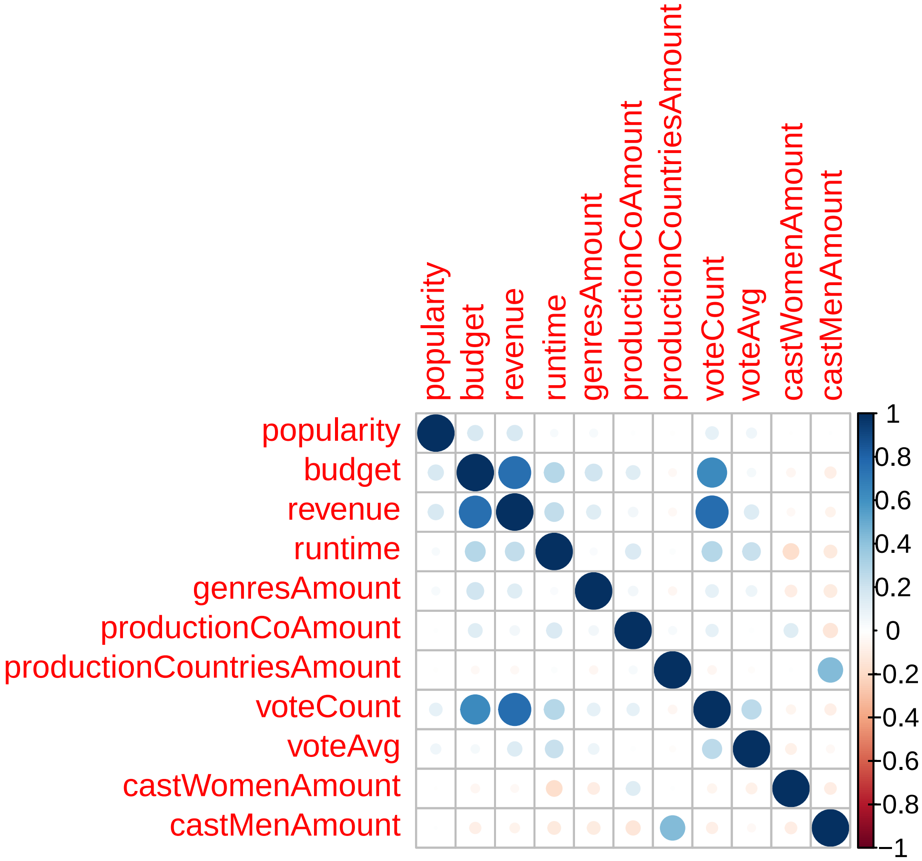
```
## R was not square, finding R from data
```

```
## $chisq
## [1] 24683.81
## 
## $p.value
## [1] 0
## 
## $df
## [1] 55
```

Como P=0 por lo que se rechaza la hipotesis nula, implicando que el analisis factorial es apropiado.

#• Obtiene los componentes principales y explica cuántos seleccionará para explicar la mayor variabilidad posible.

```r
matriz_cor <- cor(sub_datos, use = "pairwise.complete.obs" )
corrplot(matriz_cor)
```



Segun la tabla vemos que la variable del presupuesto de la pelicula se correlaciona con ingreso de pelicula, asi mismo con el numero de votos en la plataforma. El numero de votos en la plataforma con el ingreso de la pelicula tambien estan correlacionados. El numero de actores en el elenco de las peliculas esta relacionado con la cantidad de paises en los que se rodo la pelicula.

#• Interpreta los coeficientes principales.

```
pca_result<-princomp(covmat=matriz_cor,use = "pairwise.complete.obs" )
```

```
## Warning: In princomp.default(covmat = matriz_cor, use = "pairwise.complete.obs") :
##  extra argument 'use' will be disregarded
```

```
#compPrinc<-prcomp(sub_datos, scale = T, use = "pairwise.complete.obs")
#compPrinc
summary(pca_result)
```

```
## Importance of components:
##                           Comp.1    Comp.2    Comp.3    Comp.4     Comp.5
## Standard deviation     1.6670613 1.2035413 1.0882368 1.0497615 0.99313055
## Proportion of Variance 0.2526448 0.1316829 0.1076599 0.1001817 0.08966439
## Cumulative Proportion  0.2526448 0.3843277 0.4919876 0.5921694 0.68183377
##                            Comp.6     Comp.7     Comp.8     Comp.9    Comp.10
## Standard deviation     0.98388184 0.93595208 0.80386875 0.71186565 0.56377525
## Proportion of Variance 0.08800213 0.07963694 0.05874591 0.04606843 0.02889478
## Cumulative Proportion  0.76983590 0.84947284 0.90821875 0.95428717 0.98318195
##                           Comp.11
## Standard deviation     0.43011460
## Proportion of Variance 0.01681805
## Cumulative Proportion  1.00000000
```

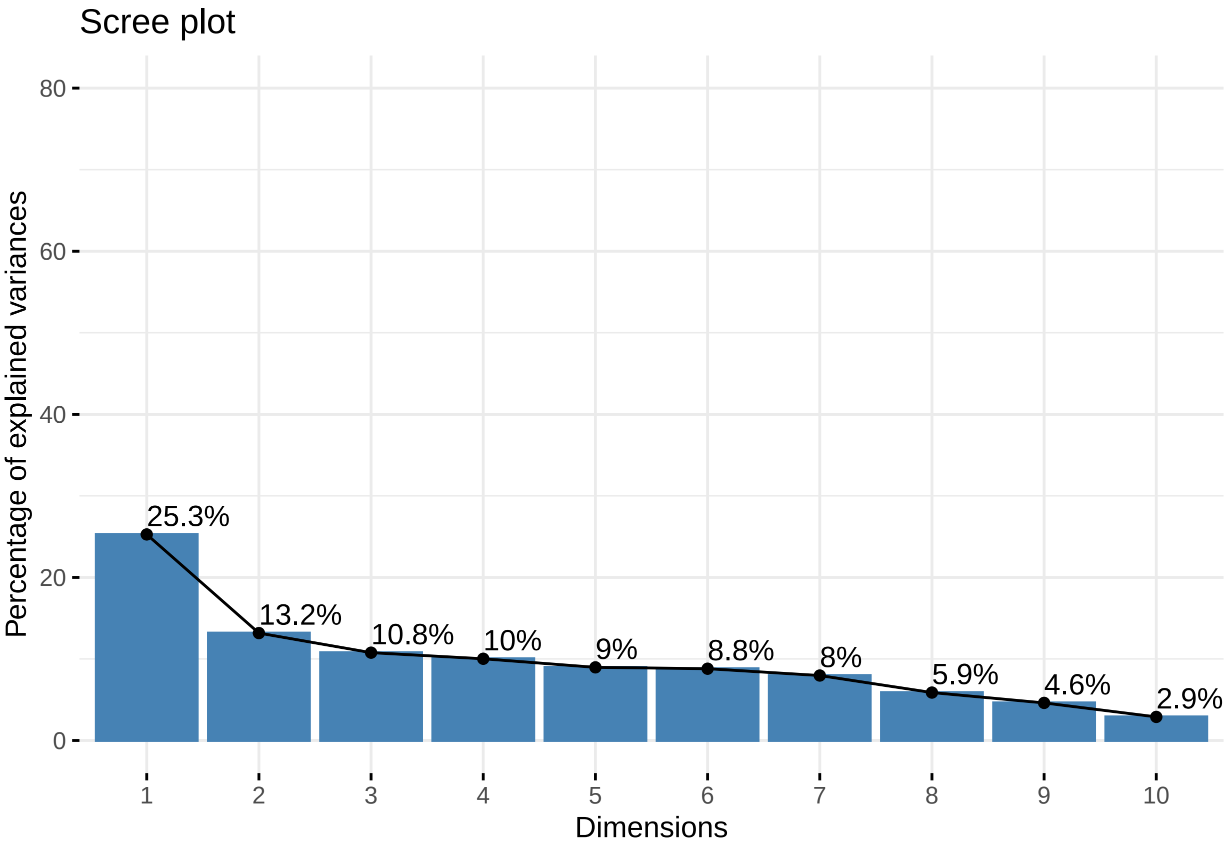
#regla de Kaiser

```
valores_propios<-pca_result$sdev^2
valores_propios
```

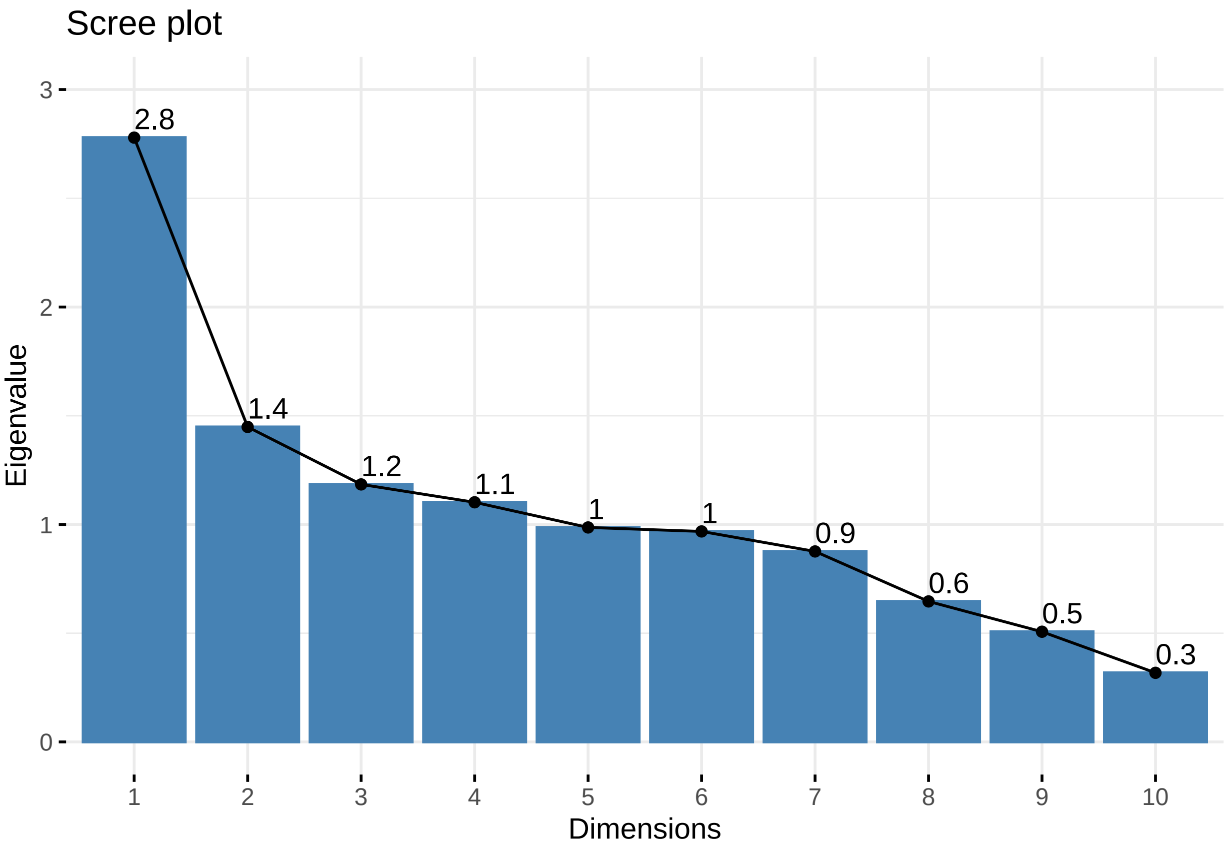
```
##    Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6    Comp.7    Comp.8
## 2.7790932 1.4485116 1.1842592 1.1019991 0.9863083 0.9680235 0.8760063 0.6462050
##    Comp.9   Comp.10   Comp.11
## 0.5067527 0.3178425 0.1849986
```

Tomamos los 4 componentes principales.

```
fviz_eig(pca_result, addlabels = TRUE, ylim = c(0, 80))
```



```
fviz_eig(pca_result, addlabels = TRUE, choice = c("eigenvalue"), ylim = c(0, 3))
```



Interpretacion: EL componenete 1 se relaciona con el exito comercial y popularidad de la pelicula, las peliculas con alto presupuesto alto ingreso, y votaciones tienen valores altos en este componente.

El componente 2, puede indicar la cantidad de actores en el elenco pueden estar asociados con los paises productores.

El componente 3, inidica un mayor numero de mujeres en el elenco tienden a tener menor puntuacion.

El componente 4, Los valores altos podriamos relacionarlo con las peliculas independientes, es decir con menos productoras.