Fundamentos de Análisis de Datos

Unidad 6

Análisis multivariante de datos

Referencias

- Menoyo Ros, D. García López, E. & García Cabot, A. (2021). *Fundamentos de la ciencia de datos:* (ed.). Editorial Universidad de Alcalá. https://elibro.net/es/ereader/bibliotecautpl/177631?page=11 (apartado 11.5, pág 309)

Agenda

- Análisis multivariante de datos
  - Concepto
  - Técnicas
  - Pasos
- Regresión múltiple
- Análisis de componentes principales (PCA)
- Análisis de correspondencias (CA)

Análisis Multivariante de Datos

- El análisis multivariante es una rama de la estadística que examina múltiples variables simultáneamente para comprender sus interrelaciones y cómo influyen en los resultados.
- Se trabaja con conjuntos de datos que contienen múltiples variables estadísticamente relevantes. Por ejemplo, en el contexto de datos de salud: edad, peso, presión arterial, tipo de sangre, imc, etc.
- Se busca identificar y entender relaciones entre múltiples variables que pueden ser: correlaciones, dependencias, agrupaciones o patrones comunes.
- También, ayudan a reducir la dimensionalidad de los datos facilitando su visualización y análisis.

Análisis Multivariante de Datos

- Algunas de las técnicas más utilizadas en análisis multivariante, son:
  - **Análisis de Regresión Múltiple**: modela la relación entre una variable dependiente y múltiples variables independientes
  - **Análisis de Componentes Principales (PCA)**: reduce la dimensionalidad mientras conserva la variabilidad de los datos
  - **Análisis de Correspondencias**: visualiza la relación entre variables categóricas
  - **Análisis de Clústeres**: agrupa observaciones según la similaridad
  - **Análisis Discriminante**: clasifica las observaciones en grupos predefinidos
  - **Análisis Factorial**: identifica las variables subyacentes (factores) que explican las correlaciones entre variables observadas

Análisis Multivariante de Datos

- Estas técnicas multivariantes son fundamentales para el análisis de datos complejos en diversas disciplinas. Ya que proporcionan métodos robustos para:
  - Estudiar y comprender relaciones complejas
  - Reducir la dimensionalidad
  - Identificar patrones
  - Mejorar la precisión de las predicciones
  - Controlar variables confusoras
  - Evaluar el impacto de múltiples factores simultáneamente
- La elección de la técnica adecuada depende del tipo de datos y los objetivos específicos del análisis.

Análisis Multivariante de Datos

PASOS

Recopilación de datos	• Recopilación de datos • Limpieza de datos • Escalado de datos
Análisis univariante (EDA)	• Análisis univariante • Análisis bivariante • Visualización de datos
Aplicación de técnicas multivariantes	• Seleccionar la técnica adecuada • Aplicar la técnica • Interpretar los resultados
Validación y evaluación	• Validar los resultados • Evaluar la calidad y relevancia de los resultados

Regresión lineal múltiple

- **Propósito**: Modelar la relación entre una variable dependiente y múltiples variables independientes.
- **Descripción**: Extiende el modelo de regresión lineal simple para incluir múltiples predictores.
- **Aplicaciones**: Predicción de valores, análisis de la relación entre variables.
- **Condiciones**:
  - Aplica para la predicción de **variables numéricas**
  - Si los predictores (variables independientes) son variables categóricas, se las debe **factorizar**.

Regresión lineal múltiple

- **Factorizar** variables categóricas se refiere a convertirlas a un formato que pueda ser utilizado en modelos estadísticos y de aprendizaje automático. Permite:
  - Transformar categorías a una equivalencia numérica que algoritmos puedan procesar.
  - Incluir variables categóricas en modelos estadísticos y de aprendizaje automático
  - Mejorar la comprensión y visualización de datos
- Factorizar en R:

```
data$Sex <- as.factor(data$Sex)
```

Categoría	Factor asociado
female	1
male	2

Regresión lineal múltiple

Ejemplo sobre data set **`mtcars`**

Variable	Tipo	Descripción
mpg	Numérica	Miles per gallon: rendimiento de combustible (millas por galón)
cyl	Entera	Número de cilindros del motor (4, 6, 8)
disp	Numérica	Desplazamiento del motor (pulgadas cúbicas)
hp	Entera	Caballos de fuerza
drat	Numérica	Relación del eje trasero (diferencial)
wt	Numérica	Peso del vehículo (en miles de libras)
qsec	Numérica	Tiempo en 1/4 de milla (segundos para recorrer un cuarto de milla)
vs	Entera	Tipo de motor: 0 = V-shaped, 1 = en línea
am	Entera	Tipo de transmisión: 0 = automática, 1 = manual
gear	Entera	Número de marchas (engranajes) de la caja de cambios
carb	Entera	Número de carburadores

Construir un modelo de regresión para predecir el rendimiento de combustible (**mpg**, millas por galón) en función de tres variables:

- **wt**: peso del vehículo (en miles de libras)
- **hp**: caballos de fuerza
- **drat**: diferencial del eje trasero

Regresión lineal múltiple

- Ejemplo regresión línea múltiple en R:

```
model <- lm(mpg ~ wt + hp + drat, data = mtcars)
summary(model)
```

```
Call:
lm(formula = mpg ~ wt + hp + drat, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-3.3598 -1.8374 -0.5099  0.9681  5.7078

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 29.394934   6.156303   4.775 5.13e-05 ***
wt          -3.227954   0.796398  -4.053 0.000364 ***
hp          -0.032230   0.008925  -3.611 0.001178 **
drat         1.615049   1.226983   1.316 0.198755
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.561 on 28 degrees of freedom
Multiple R-squared:  0.8369,    Adjusted R-squared:  0.8194
F-statistic: 47.88 on 3 and 28 DF,  p-value: 3.768e-11
```

Regresión lineal múltiple

```
Call:
lm(formula = mpg ~ wt + hp + drat, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-3.3598 -1.8374 -0.5099  0.9681  5.7078

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 29.394934   6.156303   4.775 5.13e-05 ***
wt          -3.227954   0.796398  -4.053 0.000364 ***
hp          -0.032230   0.008925  -3.611 0.001178 **
drat         1.615049   1.226983   1.316 0.198755
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.561 on 28 degrees of freedom
Multiple R-squared: 0.8369,   Adjusted R-squared: 0.8194
F-statistic: 47.88 on 3 and 28 DF,  p-value: 3.768e-11
```

Valor esperado de mpg cuando wt, hp y drat son cero

Por cada aumento de 1000 libras de peso, se espera que mpg disminuya 5.05

p-valor < 0.05, estadísticamente significativo

En promedio el modelo se equivoca en +/- 2.56 mpg

Coeficiente de determinación: El modelo explica aprox el 83.69 % de la variabilidad de mpg

Coeficiente de determinación ajustado

p-valor < 0.05, el modelo en su conjunto de estadísticamente significativo

Regresión lineal múltiple

- Los parámetros más importantes a analizar en el modelo resultante de una regresión lineal múltiple, son:
  - **Residual Standard Error (Error Estándar de los Residuos)**: Indica qué tan lejos están los valores observados de los valores predichos. Valores más bajos indican un mejor ajuste del modelo.
  - **R-cuadrado (R-squared) (coeficiente de determinación)**: Indica en que proporción la variabilidad en la variable dependiente se explica por las variables independientes en el modelo.
  - **Adjusted R-squared (coedificente de determinación ajustado)**: Ajusta el R-cuadrado por el número de predictores en el modelo. Un valor ajustado más alto indica un mejor ajuste del modelo, con base en la cantidad de predictores. Evalúa si una nueva variable mejora el modelo más alla del azar.
  - **p-value**: ayuda a decidir si una variable del modelo tiene un efecto real o si su efecto puede deberse al azar. Si es un valor pequeño (< 0.05) es poco probable que el resultado de la predicción sea por casualidad.

Regresión lineal múltiple

Evaluación del modelo:

```
predicciones <- predict(model, newdata = mtcars)
comparacion <- data.frame(Real = mtcars$mpg,
                          Predicho = predicciones,
                          Residuo = mtcars$mpg - predicciones)

print(comparacion)
```

Real	Predicho	Residuo
21.0	23.691040	-2.69103977
21.0	22.867911	-1.86791148
22.8	25.126590	-2.32659041
21.4	20.446067	0.95393269
18.7	17.737855	0.96214517
18.1	19.299555	-1.19955505
14.3	15.158995	-0.85899549
24.4	23.059005	1.34099488
22.8	22.495981	0.30401886
19.2	20.657423	-1.45742317

Regresión lineal múltiple

```
Call:
lm(formula = Age ~ Pclass + Sex + SibSp + Parch + Fare, data = titanic_clean)

Residuals:
    Min      1Q  Median      3Q     Max
-36.341  -8.314  -0.720   7.224  45.277

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  40.65972    1.53443  26.498  < 2e-16 ***
Pclass2     -10.13353    1.55326  -6.524 1.31e-10 ***
Pclass3     -14.89020    1.46354 -10.174  < 2e-16 ***
Sexmale       3.15374    1.03882   3.036  0.00249 **
SibSp        -3.79680    0.55876  -6.795 2.31e-11 ***
Parch        -0.68086    0.63108  -1.079  0.28101
Fare         -0.02579    0.01178  -2.189  0.02891 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.68 on 705 degrees of freedom
Multiple R-squared:  0.2406,    Adjusted R-squared:  0.2342
F-statistic: 37.24 on 6 and 705 DF,  p-value: < 2.2e-16
```

Ejercicio 1

- Desarrollar tarea en clase sobre regresión lineal múltiple

Reducción de dimensionalidad

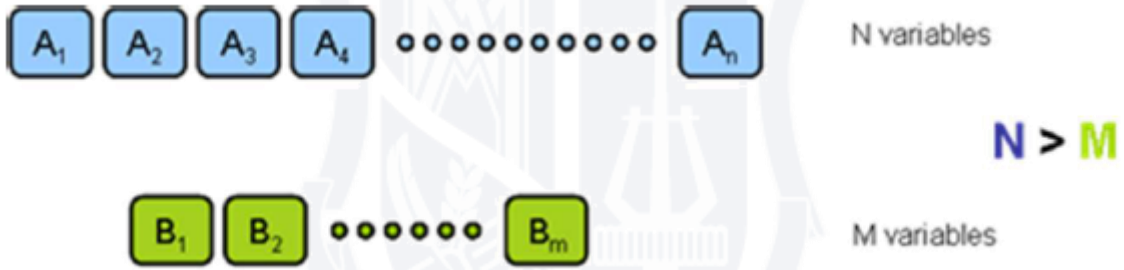
- En analítica de datos es muy importante reducir la dimensionalidad a efectos de centrar nuestro análisis en las variables más relevantes, pero sin perder la variabilidad de los datos.
- Métodos
  - Eliminar variables no relevantes estadísticamente
  - Eliminar variables derivadas
  - Eliminar variables correlacionadas al 100% con otras
  - Eliminar variables con alto porcentaje (>40%) de datos faltantes
  - Análisis de componentes principales (máxima varianza)
  - Análisis de componentes independientes (máxima independencia)
  - Feature Selection (elegir variables más relevantes)

Análisis de Componentes Principales (PCA)

- **Propósito**: Reducir la dimensionalidad de un conjunto de variables numéricas mientras se conserva la mayor cantidad de variabilidad posible.
- **Descripción**: Transforma las variables originales en un conjunto de variables no correlacionadas llamadas componentes principales.
- **Para que se usa**:
  - **Resumir** variables muy correlacionadas.
  - **Visualizar** datos de muchas dimensiones en 2 o 3.
  - **Mejorar** modelos de predicción al eliminar redundancia..
- **Condiciones**:
  - Aplica a la reducción de la dimensionalidad de **variables continuas**

Análisis de Componentes Principales (PCA)

- Es uno de los métodos de transformación de atributos más eficiente. Consiste en transformar las n variables originales en otro conjunto de atributos más reducido.



- Busca reducir la dimensión encontrando unas componentes principales (PC) resultantes de la combinación lineal ortogonal de las variables originales, intentando obtener la mayor cantidad de varianza posible.
- La primera PC (PC1) es la combinación lineal que contiene mayor varianza, la segunda PC (PC2) es la combinación lineal que contiene mayor varianza después de la primera, y así sucesivamente

Análisis de Componentes Principales (PCA)

- **Ejemplo** (dataset "**Country-data.csv**"):
  - **Country**: Pais
  - **Child_mort**: Índice mortalidad infantil
  - **Exports**: Índice exportación de bienes y servicios (% del PIB)
  - **Health**: Gasto en salud como % del PIB
  - **Imports**: Índice importación de bienes y servicios (% del PIB)
  - **Income**: Ingreso neto por persona
  - **Inflation**: Índice de inflación
  - **life_expec**: Expectativa de vida
  - **total_fer**: Índice de Niños nacidos por cada mujer
  - **Gdpp**: Producto interno bruto percapita

Análisis de Componentes Principales (PCA)

```r
library(dplyr)
library(ggplot2)

# Cargar y explorar datos
df <- read.csv("Country-data.csv")

head(df)
summary(df)
names(df)

# Calcular PCA
df_pca <- df %>% select(-1)

pca <- prcomp(df_pca, center = TRUE, scale. = TRUE)

summary(pca)

plot(pca)

# Ver la matriz de rotación
print(pca$rotation)
```

Análisis de Componentes Principales (PCA)

```r
library(dplyr)
library(ggplot2)

# Cargar y explorar datos
df <- read.csv("Country-data.csv")

head(df)
summary(df)
names(df)

# Calcular PCA
df_pca <- df %>% select(-1)

pca <- prcomp(df_pca, center = TRUE, scale. = TRUE)

summary(pca)

plot(pca)

# Ver la matriz de rotación
print(pca$rotation)
```

center : resta la media de cada variable, para que la variable tenga media cero. Centrar los datos. Ayuda a que se calcule correctamente la varianza explicada por cada componente

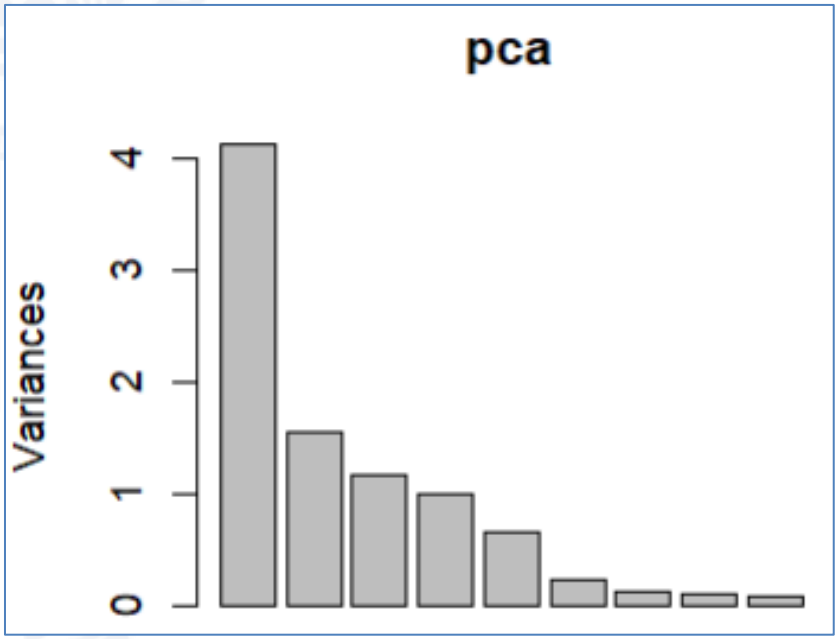
scale. : divide cada variables por su desviación estándar (estandariza) , con ello todas las variables tendrán varianza 1. Misma escala.

Proporción total de la varianza explicada

```
Importance of components:
                          PC1    PC2    PC3    PC4    PC5     PC6    PC7     PC8     PC9
Standard deviation     2.0336 1.2435 1.0818 0.9974 0.8128 0.47284 0.3368 0.29718 0.25860
Proportion of Variance 0.4595 0.1718 0.1300 0.1105 0.0734 0.02484 0.0126 0.00981 0.00743
Cumulative Proportion  0.4595 0.6313 0.7614 0.8719 0.9453 0.97015 0.9828 0.99257 1.00000
```

Análisis de Componentes Principales (PCA)

Como se ve, estas componentes principales (combinaciones lineales de las variables originales) aparecen ordenadas de manera que la primera componente tiene la mayor varianza, la segunda tiene la segunda mayor varianza, y así sucesivamente.




```
Importance of components:
                          PC1    PC2    PC3    PC4    PC5     PC6    PC7     PC8     PC9
Standard deviation     2.0336 1.2435 1.0818 0.9974 0.8128 0.47284 0.3368 0.29718 0.25860
Proportion of Variance 0.4595 0.1718 0.1300 0.1105 0.0734 0.02484 0.0126 0.00981 0.00743
Cumulative Proportion  0.4595 0.6313 0.7614 0.8719 0.9453 0.97015 0.9828 0.99257 1.00000
```

Análisis de Componentes Principales (PCA)

- Matriz de rotación o matriz de carga

	PC1	PC2	PC3	PC4	PC5	PC6
child_mort	-0.4195194	-0.192883937	0.02954353	0.370653262	-0.16896968	-0.200628153
exports	0.2838970	-0.613163494	-0.14476069	0.003091019	0.05761584	0.059332832
health	0.1508378	0.243086779	0.59663237	0.461897497	0.51800037	-0.007276456
imports	0.1614824	-0.671820644	0.29992674	-0.071907461	0.25537642	0.030031537
income	0.3984411	-0.022535530	-0.30154750	0.392159039	-0.24714960	-0.160346990
inflation	-0.1931729	0.008404473	-0.64251951	0.150441762	0.71486910	-0.066285372
life_expec	0.4258394	0.222706743	-0.11391854	-0.203797235	0.10821980	0.601126516
total_fer	-0.4037290	-0.155233106	-0.01954925	0.378303645	-0.13526221	0.750688748
gdpp	0.3926448	0.046022396	-0.12297749	0.531994575	-0.18016662	-0.016778761

	PC7	PC8	PC9
child_mort	-0.07948854	0.68274306	-0.32754180
exports	-0.70730269	0.01419742	0.12308207
health	-0.24983051	-0.07249683	-0.11308797
imports	0.59218953	0.02894642	-0.09903717
income	0.09556237	-0.35262369	-0.61298247
inflation	0.10463252	0.01153775	0.02523614
life_expec	0.01848639	0.50466425	-0.29403981
total_fer	0.02882643	-0.29335267	0.02633585
gdpp	0.24299776	0.24969636	0.62564572

Muestra cómo cada componente principal (PC) se relaciona con las variables originales. Ayuda a entender qué variables tienen mayor influencia en cada componente

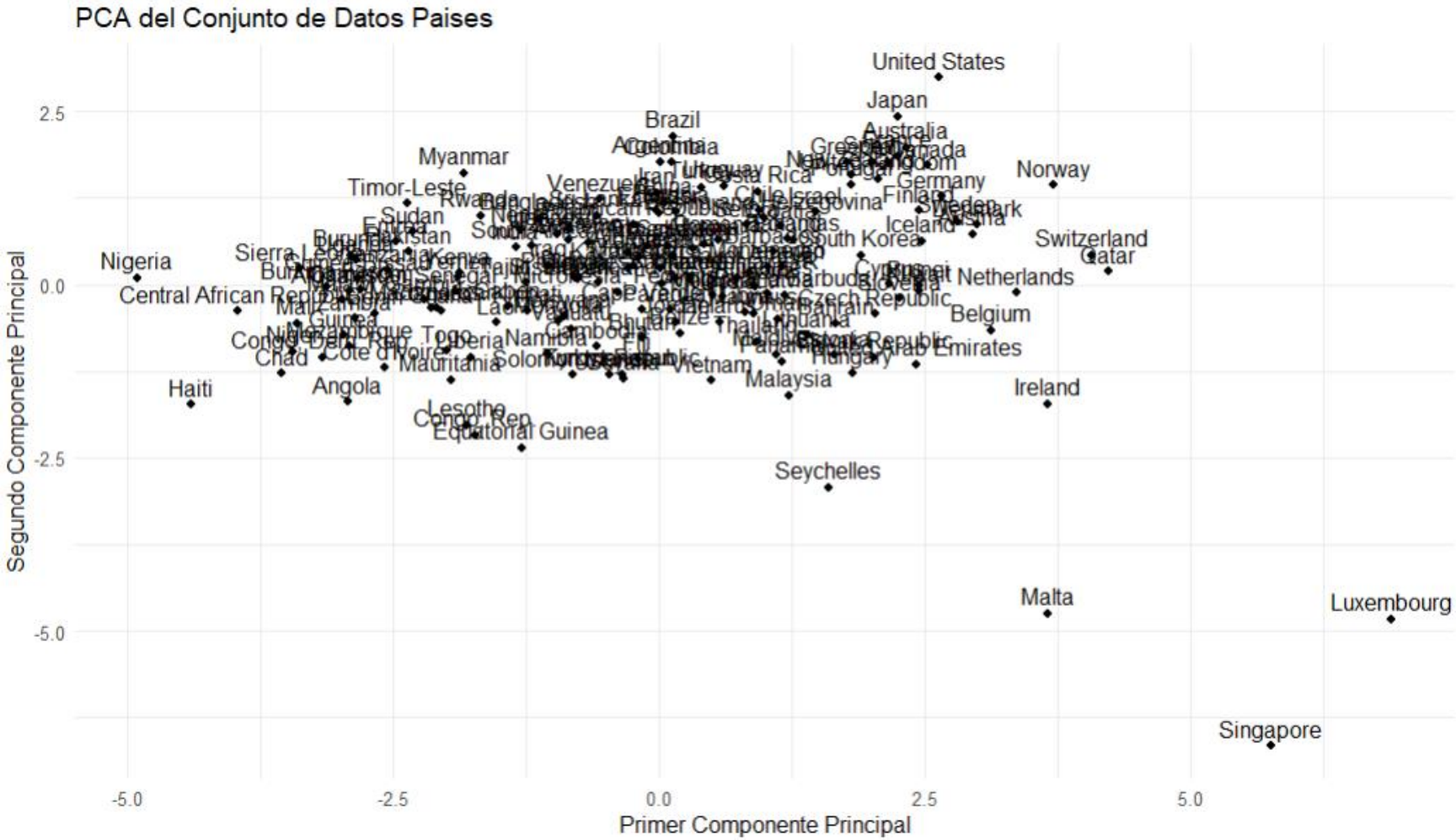
Cargas o pesos (En términos absolutos: > 0.3 mínimamente relacionada, > 0.5 moderadamente relacionada, > 0.7 altamente relacionada)

Análisis de Componentes Principales (PCA)

Visualización de las componentes principales a través de un gráfico de dispersión

```r
# Convertir los datos de PCA a un data frame
pca_data <- as.data.frame(pca$x)

# Añadir las etiquetas de los paises
pca_data$Pais <- df$country

# Crear un gráfico de dispersión de las dos primeras componentes principales
ggplot(pca_data, aes(x = PC1, y = PC2, label = Pais)) +
  geom_point() +
  geom_text(vjust = -0.5, hjust = 0.5) +
  labs(title = "PCA del Conjunto de Datos Paises",
       x = "Primer Componente Principal",
       y = "Segundo Componente Principal") +
  theme_minimal()
```

Análisis de Componentes Principales (PCA)



Ejercicio 2

- Desarrollar tarea en clase sobre Análisis de Componentes Principales

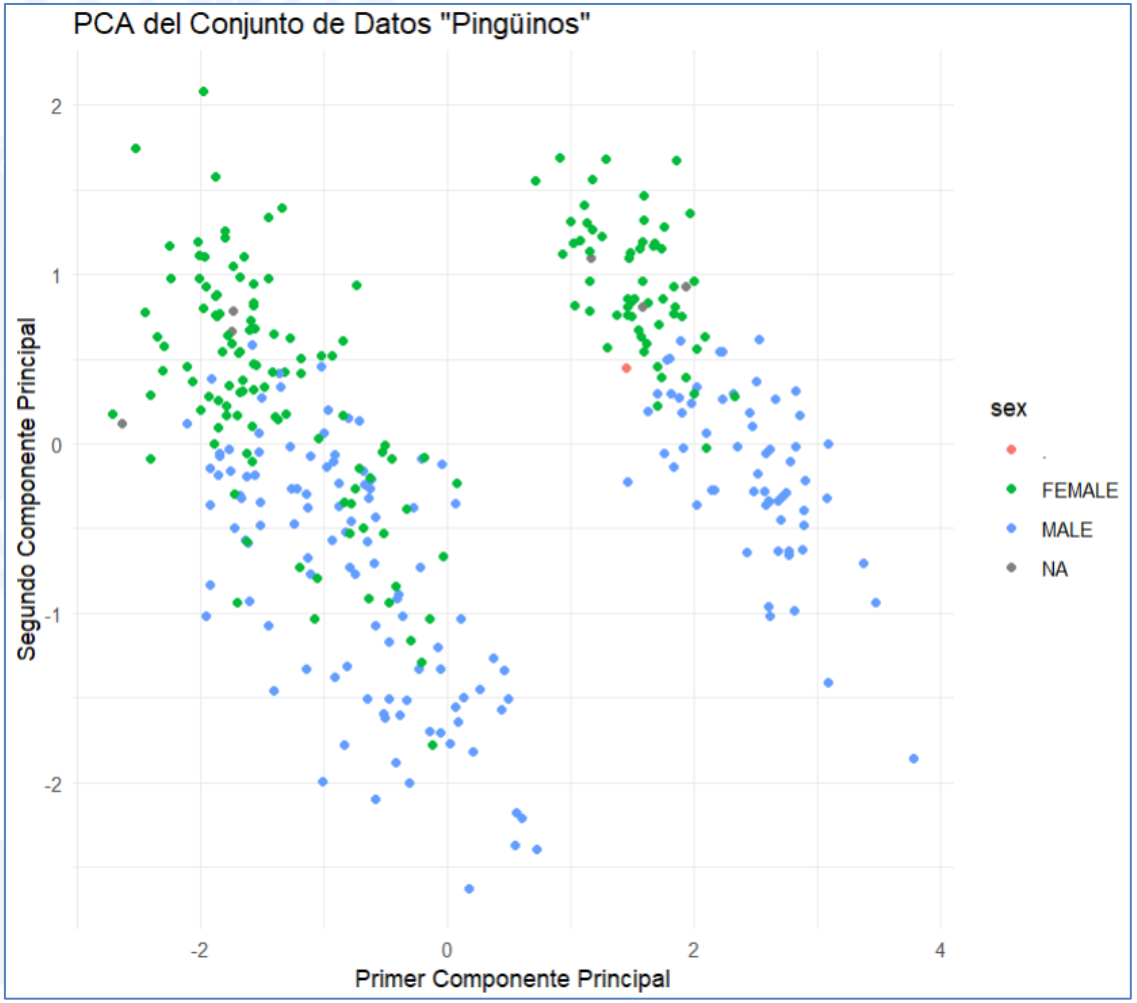
Ejercicio 2

Varianza explicada

```
Importance of components:
                          PC1    PC2
Standard deviation     1.6580 0.8863
Proportion of Variance 0.6872 0.1964
Cumulative Proportion  0.6872 0.8836
                          PC3     PC4
Standard deviation     0.59838 0.32807
Proportion of Variance 0.08951 0.02691
Cumulative Proportion  0.97309 1.00000
```

Matriz de rotación

	PC1	PC2
culmen_length_mm	0.4527180	-0.606745797
culmen_depth_mm	-0.3990109	-0.791665543
flipper_length_mm	0.5768384	-0.003161008
body_mass_g	0.5505399	-0.071520721
	PC3	PC4
culmen_length_mm	-0.6366354	0.1469738
culmen_depth_mm	0.4333322	-0.1621087
flipper_length_mm	0.2324246	-0.7830877
body_mass_g	0.5940517	0.5821454



Análisis de Correspondencias (CA)

- **Objetivo**: es una técnica de análisis multivariante que se utiliza para explorar la relación entre categorías de dos variables cualitativas, a partir de una tabla de contingencia.
- **Descripción**: Utiliza una técnica llamada “descomposición en valores singulares” para reducir la dimensionalidad y representar gráficamente (mediante un biplot o gráfico de correspodencias) las relaciones entre filas y columnas de una **tabla de contingencia**.
- **Aplicaciones**: Análisis de encuestas, estudios de mercado, salud, sociología y política, etc.
- **Condiciones**:
  - Aplica para correlacionar **variables categóricas**

Análisis de Correspondencias (CA)

- Se busca asociar cada categorías a un punto en un espacio bidimensional y mediante una análisis de proximidad determinar la asociación entre diferentes categorías.
- Se usa cuando queremos entender asociaciones entre dos variables cualitativas.
- **Ejemplo** (dataset "**laptop_data_cleaned_ori.csv**"):
  - **Objetivo**: Mediante análisis de correspondencias Visualizar y analizar la relación entre el *tipo de laptop*, y el *fabricante del procesador gráfico*.
  - **Librería a utilizar**: FactoMineR

Análisis de Correspondencias (CA)

```r
install.packages("FactoMineR")
library(FactoMineR)
library(ggplot2)
library(dplyr)

# Cargar y explorar datos
df <- read.csv("laptop_data_cleaned_ori.csv")

head(df)
summary(df)
names(df)

# Verificar que no existan datos faltantes
sapply(df,function(x) mean(is.na(x) | !nzchar(x)))

# Crear la Tabla de Contingencia
# Variables analizar: TypeName, Gpu_brand

tabla_contingencia <- table(df$TypeName,df$Gpu_brand)

print(tabla_contingencia)

# Realizar el Análisis de Correspondencias

resultado_ca <- CA(tabla_contingencia,graph = FALSE)

summary(resultado_ca)
```

Tabla de contingencia (Frecuencias)

	AMD	Intel	Nvidia
2 in 1 Convertible	2	105	9
Gaming	7	0	198
Netbook	0	23	0
Notebook	152	408	146
Ultrabook	10	166	18
Workstation	3	1	25

Análisis de Correspondencias (CA)

Resultados Análisis de Correspondencias

```
The chi square of independence between the two variables is equal to 661.17 (p-value =
1.351919e-135 ).

Eigenvalues
                       Dim.1   Dim.2
Variance               0.458   0.062
% of var.             88.098  11.902
Cumulative % of var.  88.098 100.000

Rows
                  | Iner*1000 |  Dim.1 |    ctr |  cos2 |  Dim.2 |    ctr |  cos2 |
2 in 1 Convertible |    46.035 | -0.577 |  6.625 | 0.658 | -0.415 | 25.432 | 0.342 |
Gaming             |   323.264 |  1.410 | 69.955 | 0.990 | -0.140 |  5.142 | 0.010 |
Netbook            |    14.649 | -0.758 |  2.267 | 0.708 | -0.486 |  6.915 | 0.292 |
Notebook           |    45.122 | -0.195 |  4.608 | 0.467 |  0.208 | 38.888 | 0.533 |
Ultrabook          |    56.834 | -0.527 |  9.237 | 0.744 | -0.309 | 23.565 | 0.256 |
Workstation        |    33.475 |  1.212 |  7.308 | 0.999 |  0.040 |  0.059 | 0.001 |

Columns
                  | Iner*1000 |  Dim.1 |    ctr |  cos2 |  Dim.2 |    ctr |  cos2 |
AMD                |    57.709 | -0.192 |  1.097 | 0.087 |  0.621 | 85.235 | 0.913 |
Intel              |   153.172 | -0.513 | 31.710 | 0.947 | -0.121 | 13.066 | 0.053 |
Nvidia             |   308.499 |  0.994 | 67.193 | 0.997 | -0.058 |  1.700 | 0.003 |
```

< 0.5 => asociación significativa entre las variables

Varianza explicada por cada dimensión

Cantidad de varianza de cada fila o columna

Coordenada de cada categoría en el espacio de la 1era dimensión

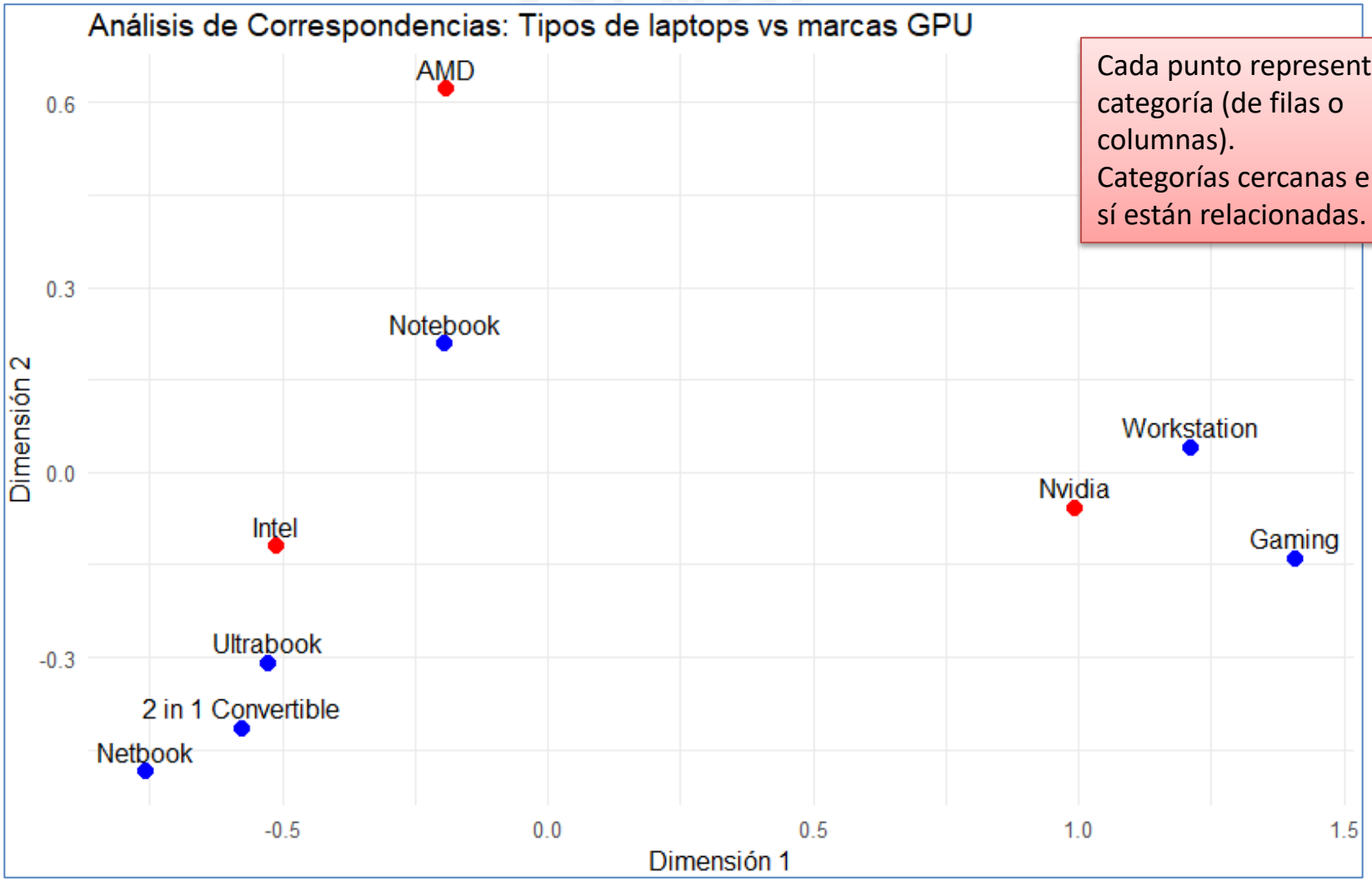
Contribución de cada categoría a la formación de la 1era dimensión

Proporción de la varianza de la categoría, explicada por la 1era dimensión

Análisis de Correspondencias (CA)

```r
# Visualizar los Resultados

# Extraer los resultados del análisis de correspondencias
fila_coords <- as.data.frame(resultado_ca$row$coord)
columna_coords <- as.data.frame(resultado_ca$col$coord)

# Añadir etiquetas
fila_coords$Etiqueta <- rownames(fila_coords)
columna_coords$Etiqueta <- rownames(columna_coords)

# Renombrar las columnas para facilidad de uso
colnames(fila_coords) <- c('Dim1', 'Dim2', 'Etiqueta')
colnames(columna_coords) <- c('Dim1', 'Dim2', 'Etiqueta')

# Crear un gráfico de dispersión de los resultados
ggplot() +
  geom_point(data = fila_coords, aes(x = Dim1, y = Dim2), color = 'blue', size = 3) +
  geom_text(data = fila_coords, aes(x = Dim1, y = Dim2, label = Etiqueta),
            vjust = -0.5, hjust = 0.5) +
  geom_point(data = columna_coords, aes(x = Dim1, y = Dim2), color = 'red', size = 3) +
  geom_text(data = columna_coords, aes(x = Dim1, y = Dim2, label = Etiqueta),
            vjust = -0.5, hjust = 0.5) +
  labs(title = 'Análisis de Correspondencias: Tipos de laptops vs marcas GPU',
       x = 'Dimensión 1',
       y = 'Dimensión 2') +
  theme_minimal()
```

Análisis de Correspondencias (CA)



Cada punto representa una categoría (de filas o columnas).
Categorías cercanas entre sí están relacionadas.

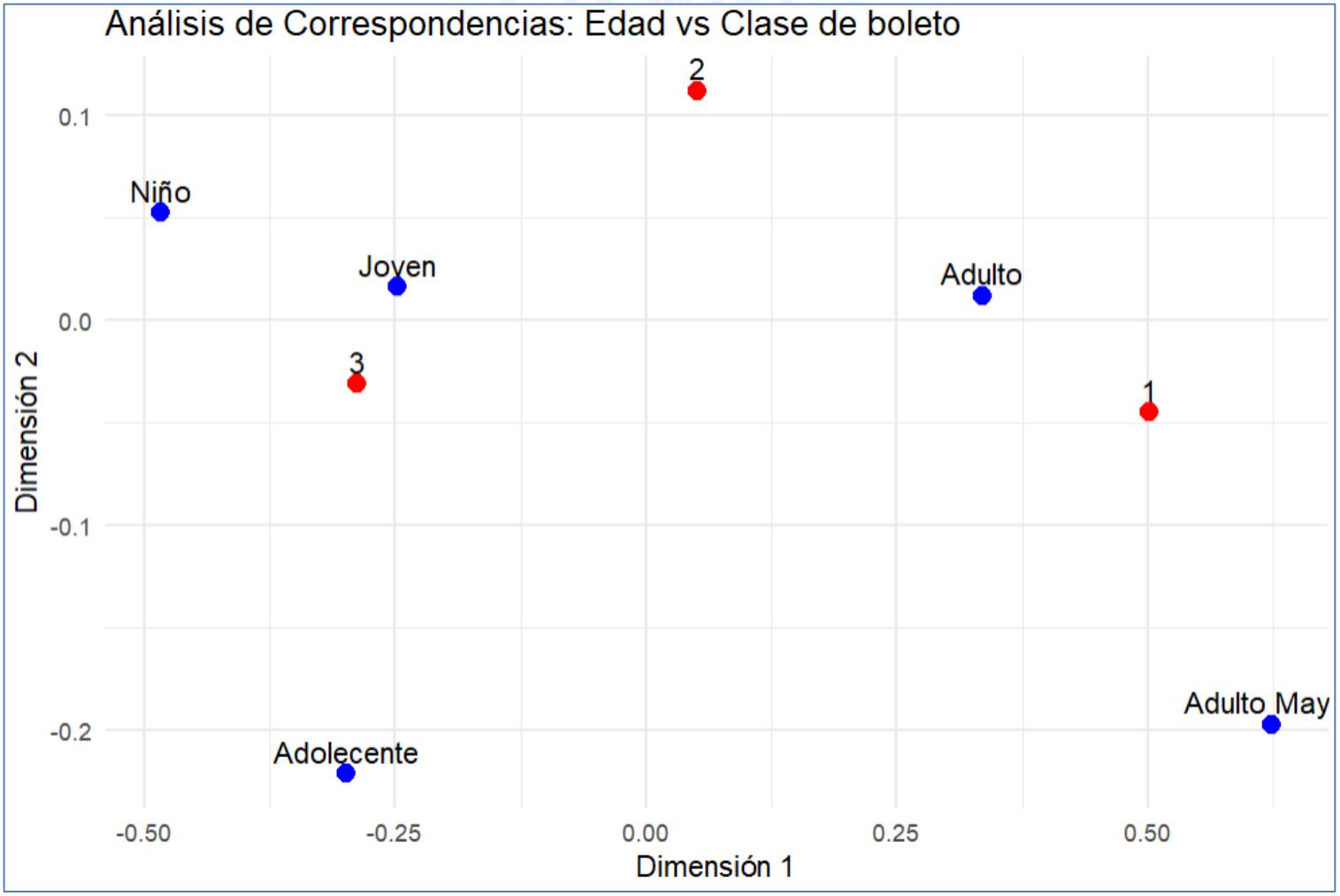
Análisis de Correspondencias (CA)

El análisis de correspondencias es como el PCA para tablas categóricas: transforma una tabla de frecuencias en un gráfico fácil de leer, donde ves qué categorías se relacionan entre sí.

Ejercicio 3

- Desarrollar tarea en clase sobre Análisis de Correspondencias

Ejercicio 3


Análisis de Correspondencias: Edad vs Clase de boleto
2
0.1
Niño
Joven
Adulto
0.0
3
1
Dimensión 2
-0.1
Adulto May
-0.2
Adolecente
-0.50
-0.25
0.00
0.25
0.50
Dimensión 1