Ciclo de vida de los datos. Práctica 2

2025-01-04

```
# Instalar y cargar tinytex
install.packages("tinytex")
```

```
## Installing package into 'C:/Users/riosl/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)
```

```
## package 'tinytex' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\riosl\AppData\Local\Temp\RtmpwRz86I\downloaded_packages
```

```
library(tinytex)

# Instalar TinyTeX
tinytex::install_tinytex(force = TRUE)
```

```
## tlmgr install tlgpg

## tlmgr update --self

## tlmgr install tlgpg

## tlmgr --repository http://www.preining.info/tlgpg/ install tlgpg

## tlmgr option repository "https://ctan.javinator9889.com/systems/texlive/tlnet"

## tlmgr update --list
```

Ciclo de vida de los datos. Práctica 2

Autores: Francisco Javier González Ontañón y Laureano Rios Oriol

Cargar bibliotecas y datos

En este apartado cargo las librerías necesarias y cargado de los datos.

```r
required_libraries <- c('dplyr', 'caret', 'rpart', 'cluster', 'doParallel', 'foreach')
for (lib in required_libraries) {
  if (!require(lib, character.only = TRUE)) {
    install.packages(lib, dependencies = TRUE)
    library(lib, character.only = TRUE)
  }
}
```

```
## Cargando paquete requerido: dplyr
```

```
##
## Adjuntando el paquete: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
## Cargando paquete requerido: caret
```

```
## Cargando paquete requerido: ggplot2
```

```
## Cargando paquete requerido: lattice
```

```
## Cargando paquete requerido: rpart
```

```
## Cargando paquete requerido: cluster
```

```
## Cargando paquete requerido: doParallel
```

```
## Cargando paquete requerido: foreach
```

```
## Cargando paquete requerido: iterators
```

```
## Cargando paquete requerido: parallel
```

```r
# Definir la ruta del archivo
adult_data_path <- "C:\\Users\\riosl\\Desktop\\adult.data"

# Nombres de las columnas
column_names <- c('age', 'workclass', 'fnlwgt', 'education', 'education_num', 'marital_status', 'occupa

# Leer el archivo adult.data como un DataFrame
adult_data <- read.csv(adult_data_path, header = FALSE, sep = ',', strip.white = TRUE, col.names = colu

# Mostramos las primeras líneas del archivo para inspección
print(head(adult_data))
```

```
##   age        workclass fnlwgt education education_num     marital_status
## 1  39        State-gov  77516 Bachelors            13      Never-married
## 2  50 Self-emp-not-inc  83311 Bachelors            13 Married-civ-spouse
## 3  38          Private 215646   HS-grad             9           Divorced
## 4  53          Private 234721      11th             7 Married-civ-spouse
## 5  28          Private 338409 Bachelors            13 Married-civ-spouse
## 6  37          Private 284582   Masters            14 Married-civ-spouse
##          occupation  relationship  race    sex capital_gain capital_loss
## 1      Adm-clerical Not-in-family White   Male         2174            0
## 2   Exec-managerial       Husband White   Male            0            0
## 3 Handlers-cleaners Not-in-family White   Male            0            0
## 4 Handlers-cleaners       Husband Black   Male            0            0
## 5    Prof-specialty          Wife Black Female            0            0
## 6   Exec-managerial          Wife White Female            0            0
##   hours_per_week native_country income
## 1             40  United-States  <=50K
## 2             13  United-States  <=50K
## 3             40  United-States  <=50K
## 4             40  United-States  <=50K
## 5             40           Cuba  <=50K
## 6             40  United-States  <=50K
```

```r
# Mostramos las dimensiones del DataFrame
print(dim(adult_data))
```

```
## [1] 32561    15
```

1. Descripción del dataset

¿Por qué es importante y qué pregunta/problema pretende responder? Resume brevemente las variables que lo forman y su tamaño.

El conjunto de datos adult.data contiene un conjunto Conjunto de Datos del Censo de Ingresos que proviene de la base de datos del Censo de los Estados Unidos.

El objetivo principal del conjunto de datos es predecir si una persona gana más de 50,000 dólares al año en función de una serie de características demográficas y laborales. Este problema es un clásico en tareas de clasificación supervisada, y su resolución puede tener aplicaciones en:

Marketing dirigido. Análisis de políticas laborales. Detección de sesgos en ingresos.

La pregunta principal del conjunto de datos es la siguiente: ¿Qué factores demográficos y profesionales determinan si una persona gana más de $50,000 al año?

El dataset contiene 15 variables, que se pueden agrupar en:

Demográficas:

- age: Edad.
- sex: Género.
- race: Raza.
- native_country: País de origen.

- marital_status: Estado civil.

Educativas:

- education: Nivel educativo.
- education_num: Número asociado al nivel educativo.

Laborales:

- workclass: Tipo de empleo.
- occupation: Ocupación.
- hours_per_week: Horas trabajadas por semana.
- capital_gain: Ganancia de capital.
- capital_loss: Pérdida de capital.

Socioeconómicas:

- fnlwgt: Ponderación final de la muestra.
- relationship: Relación familiar.

Variable Objetivo:

- income: Nivel de ingresos (<=50K o >50K).

2. Integración y selección

Se va a usar el dataset completo para la práctica, a continuación una tabla con todas las variables del dataset.

Variable Name	Role	Type	Demographic	Description	Missing Values
age	Feature	Integer	Age	N/A	no
workclass	Feature	Categorical	Income	Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.	yes
fnlwgt	Feature	Integer			no
education	Feature	Categorical	Education Level	Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.	no
education-num	Feature	Integer	Education Level		no
marital status	Feature	Categorical	Other	Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.	no
occupation	Feature	Categorical	Other	Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.	yes

Variable Name	Role	Type	Demographic	Description	Missing Values
relationship	Feature	Categorical	Other	Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.	no
race	Feature	Categorical	Race	White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.	no
sex	Feature	Binary	Sex	Female, Male.	no
capital-gain	Feature	Integer			no
capital-loss	Feature	Integer			no
hours-per-week	Feature	Integer			no
native-country	Feature	Categorical	Other	United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands.	yes
income	Target	Binary	Income	>50K, <=50K.	no

3. Limpieza de datos.

3.1 Gestión de ceros, elementos vacíos y valores perdidos

El dataset ya viene con esta limpieza hecha por los mismos autores del dataset colocando un símbolo "?" en los valores faltantes. Sin embargo la forma de proceder sería la que seguimos a continuación:

Las variables que tienen datos perdidos son categóricas, con la intención de no generar sesgos, se colocará el valor "unknown". En el apartado anterior ya habíamos marcado como NA todas los valores vacíos y los convertimos

```
# Identificar valores perdidos
colSums(is.na(adult_data))
```

```
##            age      workclass         fnlwgt      education  education_num
##              0           1836              0              0              0
## marital_status     occupation   relationship           race            sex
##              0           1843              0              0              0
##   capital_gain   capital_loss hours_per_week native_country         income
##              0              0              0            583              0
```

```
adult_data <- adult_data %>% mutate(
  workclass = ifelse(is.na(workclass), "Unknown", workclass),
  occupation = ifelse(is.na(occupation), "Unknown", occupation),
  native_country = ifelse(is.na(native_country), "Unknown", native_country)
)
```

```r
colSums(is.na(adult_data))
```

```
##            age      workclass         fnlwgt      education  education_num
##              0              0              0              0              0
## marital_status     occupation   relationship           race            sex
##              0              0              0              0              0
##   capital_gain   capital_loss hours_per_week native_country         income
##              0              0              0              0              0
```

```r
# Ver los tipos de datos de cada columna
sapply(adult_data, class)
```

```
##            age      workclass         fnlwgt      education  education_num
##      "integer"    "character"      "integer"    "character"      "integer"
## marital_status     occupation   relationship           race            sex
##    "character"    "character"    "character"    "character"    "character"
##   capital_gain   capital_loss hours_per_week native_country         income
##      "integer"      "integer"      "integer"    "character"    "character"
```

3.2 Conversión de tipos de datos:

Se convierten todos las variables categoricas a "factor", esto se traduce en la conversión de las variables categóricas a números para poder ser procesados correctamente con modelos estadísticos y de machine learning.

```r
#Transformamos las variables categoricas a factor
adult_data$workclass <- as.factor(adult_data$workclass)
adult_data$education <- as.factor(adult_data$education)
adult_data$marital_status <- as.factor(adult_data$marital_status)
adult_data$occupation <- as.factor(adult_data$occupation)
adult_data$relationship <- as.factor(adult_data$relationship)
adult_data$race <- as.factor(adult_data$race)
adult_data$sex <- as.factor(adult_data$sex)
adult_data$native_country <- as.factor(adult_data$native_country)
adult_data$income <- as.factor(adult_data$income)
```

3.3. Identificación y gestión de valores extremos

```r
initial_rows <- nrow(adult_data)
```

```r
# 3.3.1. Edad (age)
# Mantener valores realistas y eliminar valores imposibles
adult_data <- adult_data %>% filter(age <= 100)

# 3.3.2. Ponderación muestral (fnlwgt)
# Mantener todos los valores ya que representan ponderaciones válidas
cat("Nota: No se eliminan outliers en fnlwgt, ya que son ponderaciones válidas.\n")
```

```
## Nota: No se eliminan outliers en fnlwgt, ya que son ponderaciones válidas.
```

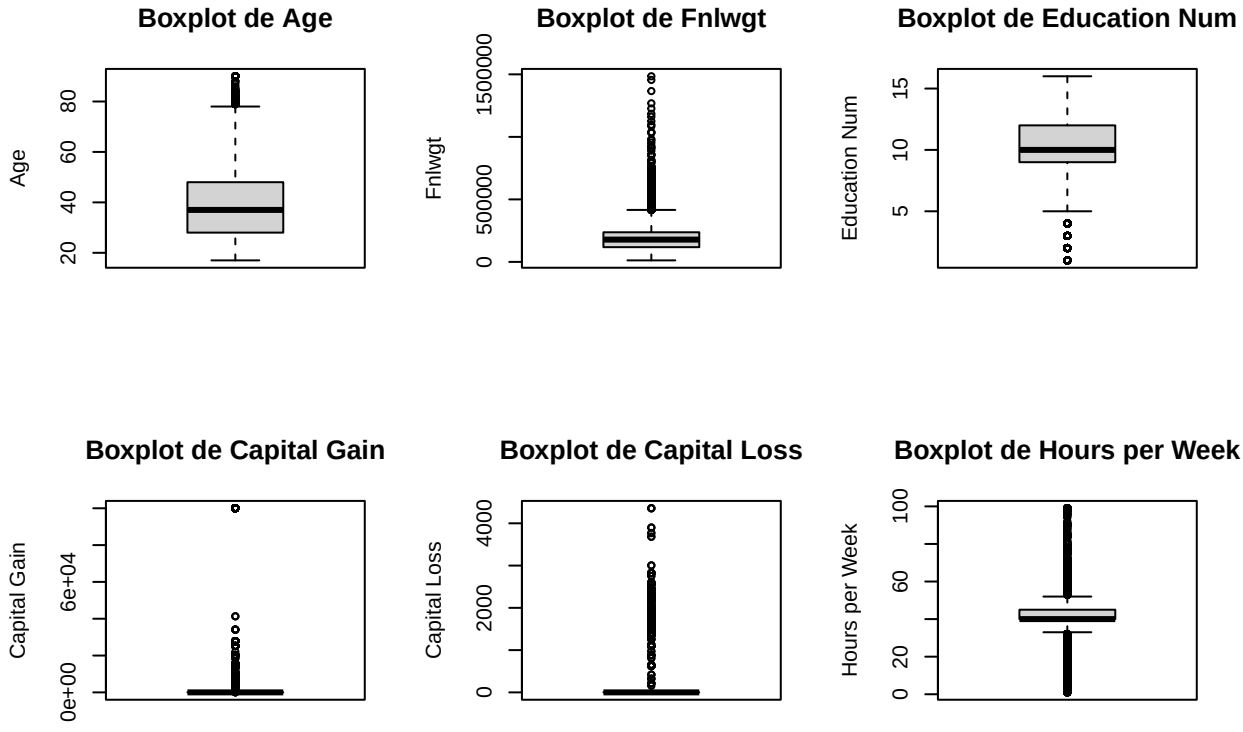
```
# 3.3.3. Años de educación (education_num)
# Mantener valores entre 1 y 20
adult_data <- adult_data %>% filter(education_num <= 20)

# 3.3.4. Ganancia de capital (capital_gain) y Pérdida de capital (capital_loss)
# Mantener todos los valores ya que son válidos en su contexto económico
cat("Nota: No se eliminan outliers en capital_gain ni capital_loss, ya que representan valores económic
```

```
## Nota: No se eliminan outliers en capital_gain ni capital_loss, ya que representan valores económicos
```

```
# 3.3.5. Horas trabajadas por semana (hours_per_week)
# Eliminar valores imposibles  > 100 horas)
adult_data <- adult_data %>% filter(hours_per_week <= 100)
```

```
# Mostrar los boxplots después de la limpieza
par(mfrow = c(2, 3)) # Configurar para mostrar múltiples gráficos
numeric_columns <- c('age', 'fnlwgt', 'education_num', 'capital_gain', 'capital_loss', 'hours_per_week'
labels <- c('Age', 'Fnlwgt', 'Education Num', 'Capital Gain', 'Capital Loss', 'Hours per Week')
for (i in 1:length(numeric_columns)) {
  boxplot(adult_data[[numeric_columns[i]]], main = paste('Boxplot de', labels[i]), ylab = labels[i])
}
```



```r
par(mfrow = c(1, 1)) # Restablecer la configuración

# Mostrar el número de filas eliminadas
deleted_rows <- initial_rows - nrow(adult_data)
cat("Número de filas eliminadas debido a valores extremos:", deleted_rows, "\n")
```

```
## Número de filas eliminadas debido a valores extremos: 0
```

3.4. Otros métodos de limpieza

Eliminar las filas duplicadas.

```r
# Identificar filas duplicadas
duplicated_rows <- nrow(adult_data) - nrow(adult_data %>% distinct())
cat("Número de filas duplicadas detectadas:", duplicated_rows, "\n")
```

```
## Número de filas duplicadas detectadas: 24
```

```r
# Eliminar filas duplicadas
adult_data <- adult_data %>% distinct()

# Validar la limpieza
dim(adult_data)
```

```
## [1] 32537    15
```

```r
summary(adult_data)
```

```
##       age                    workclass         fnlwgt
##  Min.   :17.00   Private         :22673   Min.   :  12285
##  1st Qu.:28.00   Self-emp-not-inc: 2540   1st Qu.: 117827
##  Median :37.00   Local-gov       : 2093   Median : 178356
##  Mean   :38.59   Unknown         : 1836   Mean   : 189781
##  3rd Qu.:48.00   State-gov       : 1298   3rd Qu.: 236993
##  Max.   :90.00   Self-emp-inc    : 1116   Max.   :1484705
##                  (Other)         :  981
##         education     education_num                  marital_status
##  HS-grad     :10494   Min.   : 1.00   Divorced             : 4441
##  Some-college: 7282   1st Qu.: 9.00   Married-AF-spouse    :   23
##  Bachelors   : 5353   Median :10.00   Married-civ-spouse   :14970
##  Masters     : 1722   Mean   :10.08   Married-spouse-absent:  418
##  Assoc-voc   : 1382   3rd Qu.:12.00   Never-married        :10667
##  11th        : 1175   Max.   :16.00   Separated            : 1025
##  (Other)     : 5129                   Widowed              :  993
##            occupation           relationship                    race
##  Prof-specialty :4136   Husband       :13187   Amer-Indian-Eskimo:  311
##  Craft-repair   :4094   Not-in-family : 8292   Asian-Pac-Islander: 1038
##  Exec-managerial:4065   Other-relative:  981   Black              : 3122
##  Adm-clerical   :3768   Own-child     : 5064   Other              :  271
```

```
##  Sales          :3650   Unmarried     : 3445   White             :27795
##  Other-service  :3291   Wife          : 1568
##  (Other)        :9533
##      sex         capital_gain    capital_loss     hours_per_week
##  Female:10762   Min.   :    0   Min.   :   0.00   Min.   : 1.00
##  Male  :21775   1st Qu.:    0   1st Qu.:   0.00   1st Qu.:40.00
##                 Median :    0   Median :   0.00   Median :40.00
##                 Mean   : 1078   Mean   :  87.37   Mean   :40.44
##                 3rd Qu.:    0   3rd Qu.:   0.00   3rd Qu.:45.00
##                 Max.   :99999   Max.   :4356.00   Max.   :99.00
##
##        native_country    income
##  United-States:29153   <=50K:24698
##  Mexico       :  639   >50K : 7839
##  Unknown      :  582
##  Philippines  :  198
##  Germany      :  137
##  Canada       :  121
##  (Other)      : 1707
```

Análisis de los datos.

Clasificación con Árboles de Decisión

Variable Objetivo Claramente Definida: La columna income es una variable categórica con dos clases: <=50K y >50K, lo que hace que el problema sea adecuado para clasificación binaria. Datos Mixtos: El dataset contiene tanto variables categóricas (workclass, marital_status) como numéricas (age, hours_per_week), lo cual es manejable para algoritmos de árboles de decisión. Interpretabilidad: Los árboles de decisión permiten interpretar fácilmente los factores más importantes que determinan si una persona gana más de $50K. Robustez ante Datos Faltantes y Outliers: Los árboles pueden manejar valores perdidos (aunque ya los gestionamos) y son menos sensibles a valores extremos que otros métodos, como la regresión logística.

```
# --- 4. Método Supervisado: Árbol de Decisión ---
set.seed(123)
train_index <- createDataPartition(adult_data$income, p = 0.7, list = FALSE)
train_data <- adult_data[train_index, ]
test_data <- adult_data[-train_index, ]

# Entrenar el modelo
income_model <- rpart(income ~ age + workclass + education + marital_status + occupation + relationship
                      data = train_data, method = "class")

# Evaluar el modelo
predictions <- predict(income_model, test_data, type = "class")
conf_matrix <- confusionMatrix(predictions, test_data$income)
print(conf_matrix)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction <=50K >50K
##      <=50K  7037 1148
```

```
##      >50K    372 1203
##
##                Accuracy : 0.8443
##                  95% CI : (0.8369, 0.8514)
##     No Information Rate : 0.7591
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5201
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9498
##             Specificity : 0.5117
##          Pos Pred Value : 0.8597
##          Neg Pred Value : 0.7638
##              Prevalence : 0.7591
##          Detection Rate : 0.7210
##    Detection Prevalence : 0.8386
##       Balanced Accuracy : 0.7307
##
##        'Positive' Class : <=50K
##
```

```r
# Calcular precisión
accuracy <- sum(diag(conf_matrix$table)) / sum(conf_matrix$table)
cat("Precisión del modelo:", accuracy, "\n")
```

```
## Precisión del modelo: 0.8442623
```

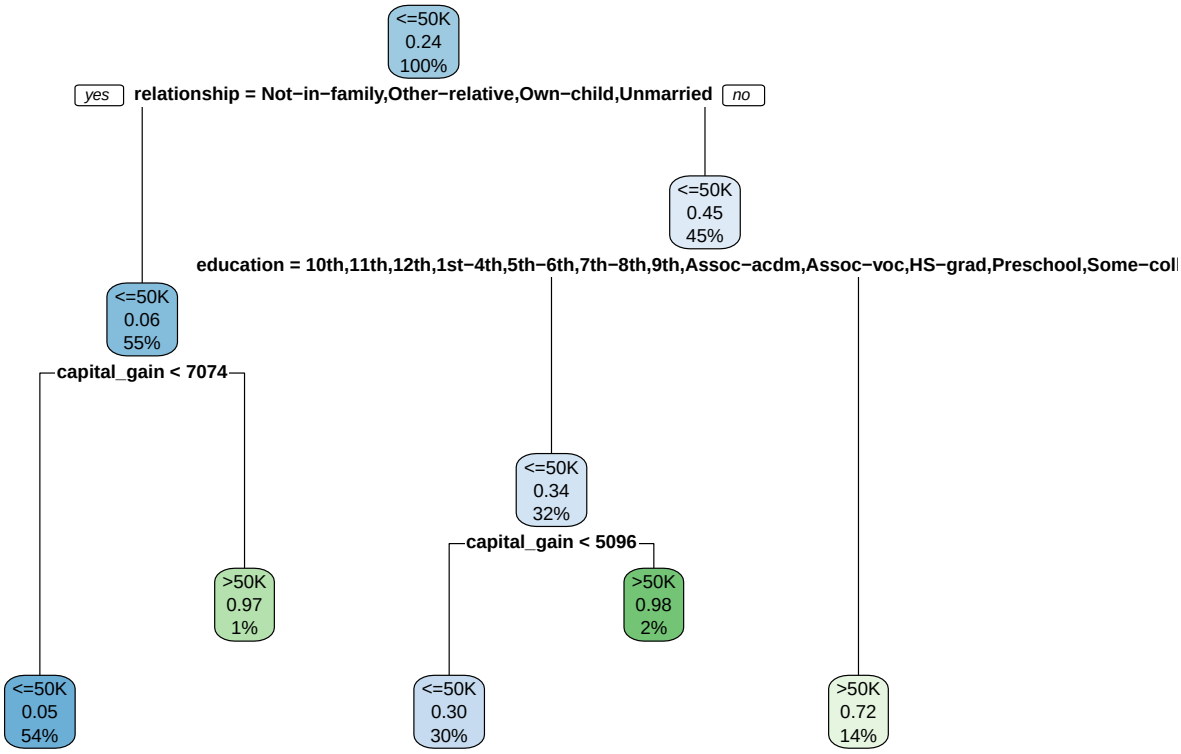
```r
install.packages("rpart.plot")
```

```
## Installing package into 'C:/Users/riosl/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)
```

```
## package 'rpart.plot' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\riosl\AppData\Local\Temp\RtmpwRz86I\downloaded_packages
```

```r
library(rpart.plot)

rpart.plot(income_model)
```



Conclusiones:

El modelo es bastante preciso (84.44%) y supera significativamente el No Information Rate.Logra identificar bien los verdaderos positivos.Lo que podemos observar de este modelo es que aquellos que tienen un nivel de estudio mas alto son lo que en general tienen un salario por encima de 50K.

Clustering con K-Means

Identificación de Patrones Ocultos: Permite descubrir grupos de individuos con características similares (por ejemplo, patrones en ocupaciones, horas trabajadas y nivel de educación).

Reducción de la Complejidad: Ayuda a simplificar la estructura de los datos, especialmente si se busca segmentar la población para análisis adicionales.

Datos Mixtos: Las variables numéricas (age, hours_per_week) se pueden utilizar para el clustering, aunque puede ser necesario estandarizar los datos para evitar que las variables con mayor rango dominen el análisis.

```
# Instalar y cargar los paquetes necesarios
install.packages("cluster")
```

```
## Warning: package 'cluster' is in use and will not be installed
```

```
install.packages("ggplot2")
```

```
## Warning: package 'ggplot2' is in use and will not be installed
```

```r
library(cluster)
library(ggplot2)
```

```r
install.packages("fastDummies")
```

```
## Installing package into 'C:/Users/riosl/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)
```

```
## package 'fastDummies' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\riosl\AppData\Local\Temp\RtmpwRz86I\downloaded_packages
```

```r
library(fastDummies)
```

```r
# Codificar las variables categóricas usando fastDummies
adult_data_encoded <- dummy_cols(adult_data, select_columns = c('workclass', 'education', 'marital_statu

# Seleccionar todas las columnas numéricas y codificadas para el clustering
numeric_columns <- c('age', 'fnlwgt', 'education_num', 'capital_gain', 'capital_loss', 'hours_per_week')
dummy_columns <- grep("_", colnames(adult_data_encoded), value = TRUE)
all_columns <- c(numeric_columns, dummy_columns)
adult_data_numeric <- adult_data_encoded[, all_columns]

# Verificar que todas las columnas sean numéricas
str(adult_data_numeric)
```

```
## 'data.frame':    32537 obs. of  104 variables:
##  $ age                                : int  39 50 38 53 28 37 49 52 31 42 ...
##  $ fnlwgt                             : int  77516 83311 215646 234721 338409 284582 160187 209
##  $ education_num                      : int  13 13 9 7 13 14 5 9 14 13 ...
##  $ capital_gain                       : int  2174 0 0 0 0 0 0 0 14084 5178 ...
##  $ capital_loss                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hours_per_week                     : int  40 13 40 40 40 40 16 45 50 40 ...
##  $ education_num.1                    : int  13 13 9 7 13 14 5 9 14 13 ...
##  $ capital_gain.1                     : int  2174 0 0 0 0 0 0 0 14084 5178 ...
##  $ capital_loss.1                     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hours_per_week.1                   : int  40 13 40 40 40 40 16 45 50 40 ...
##  $ workclass_Local-gov                : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ workclass_Never-worked             : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ workclass_Private                  : int  0 0 1 1 1 1 1 0 1 1 ...
##  $ workclass_Self-emp-inc             : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ workclass_Self-emp-not-inc         : int  0 1 0 0 0 0 0 1 0 0 ...
##  $ workclass_State-gov                : int  1 0 0 0 0 0 0 0 0 0 ...
##  $ workclass_Unknown                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ workclass_Without-pay              : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_11th                     : int  0 0 0 1 0 0 0 0 0 0 ...
##  $ education_12th                     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_1st-4th                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_5th-6th                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_7th-8th                  : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
##  $ education_9th                          : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ education_Assoc-acdm                   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_Assoc-voc                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_Bachelors                    : int  1 1 0 0 1 0 0 0 0 1 ...
##  $ education_Doctorate                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_HS-grad                      : int  0 0 1 0 0 0 0 1 0 0 ...
##  $ education_Masters                      : int  0 0 0 0 0 1 0 0 1 0 ...
##  $ education_Preschool                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_Prof-school                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ education_Some-college                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ marital_status_Married-AF-spouse       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ marital_status_Married-civ-spouse      : int  0 1 0 1 1 1 0 1 0 1 ...
##  $ marital_status_Married-spouse-absent   : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ marital_status_Never-married           : int  1 0 0 0 0 0 0 0 1 0 ...
##  $ marital_status_Separated               : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ marital_status_Widowed                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Armed-Forces                : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Craft-repair                : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Exec-managerial             : int  0 1 0 0 0 1 0 1 0 1 ...
##  $ occupation_Farming-fishing             : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Handlers-cleaners           : int  0 0 1 1 0 0 0 0 0 0 ...
##  $ occupation_Machine-op-inspct           : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Other-service               : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ occupation_Priv-house-serv             : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Prof-specialty              : int  0 0 0 0 1 0 0 0 1 0 ...
##  $ occupation_Protective-serv             : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Sales                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Tech-support                : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Transport-moving            : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ occupation_Unknown                     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ relationship_Not-in-family             : int  1 0 1 0 0 0 1 0 1 0 ...
##  $ relationship_Other-relative            : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ relationship_Own-child                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ relationship_Unmarried                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ relationship_Wife                      : int  0 0 0 0 1 1 0 0 0 0 ...
##  $ race_Asian-Pac-Islander                : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ race_Black                             : int  0 0 0 1 1 0 1 0 0 0 ...
##  $ race_Other                             : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ race_White                             : int  1 1 1 0 0 1 0 1 1 1 ...
##  $ sex_Male                               : int  1 1 1 1 0 0 0 1 0 1 ...
##  $ native_country_Canada                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_China                   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Columbia                : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Cuba                    : int  0 0 0 0 1 0 0 0 0 0 ...
##  $ native_country_Dominican-Republic      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Ecuador                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_El-Salvador             : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_England                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_France                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Germany                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Greece                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Guatemala               : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Haiti                   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Holand-Netherlands      : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
##  $ native_country_Honduras                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Hong                      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Hungary                   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_India                     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Iran                      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Ireland                   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Italy                     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Jamaica                   : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ native_country_Japan                     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Laos                      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Mexico                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Nicaragua                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Outlying-US(Guam-USVI-etc): int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Peru                      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Philippines               : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Poland                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Portugal                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Puerto-Rico               : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Scotland                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_South                     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Taiwan                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ native_country_Thailand                  : int  0 0 0 0 0 0 0 0 0 0 ...
##   [list output truncated]
```

```r
# Normalizar los datos
adult_data_scaled <- scale(adult_data_numeric)

# Aplicar K-means clustering
set.seed(123)  # Fijar semilla para reproducibilidad
kmeans_result <- kmeans(adult_data_scaled, centers = 3, nstart = 20)

# Agregar el resultado del clustering al DataFrame original
adult_data$cluster <- as.factor(kmeans_result$cluster)

# Ver los centros de los clusters
print(kmeans_result$centers)
```

```
##           age      fnlwgt education_num capital_gain capital_loss
## 1  0.13361921 -0.03047809     1.2340581   0.24604998   0.15084807
## 2  0.16844507 -0.01063110    -0.3221390  -0.06415206  -0.06658034
## 3 -0.07021337  0.01364904    -0.4885661  -0.09741822  -0.05740471
##   hours_per_week education_num.1 capital_gain.1 capital_loss.1 hours_per_week.1
## 1     0.26941287       1.2340581     0.24604998     0.15084807       0.26941287
## 2    -0.69120230      -0.3221390    -0.06415206    -0.06658034      -0.69120230
## 3    -0.05379323      -0.4885661    -0.09741822    -0.05740471      -0.05379323
##   workclass_Local-gov workclass_Never-worked workclass_Private
## 1          0.18413786            -0.01466899        -0.1553919
## 2         -0.26219662             0.24430281        -1.5160771
## 3         -0.05466731            -0.01466899         0.1940689
##   workclass_Self-emp-inc workclass_Self-emp-not-inc workclass_State-gov
## 1             0.15760451                0.045105365          0.14959934
## 2            -0.18845837               -0.290985449         -0.20383657
## 3            -0.04985178                0.005917697         -0.04519512
##   workclass_Unknown workclass_Without-pay education_11th education_12th
```

```
## 1         -0.244542          -0.015419338    -0.19355767    -0.11613355
## 2          4.072692          -0.020747322     0.15251846     0.07326732
## 3         -0.244542           0.008213931     0.06794518     0.04232013
##   education_1st-4th education_5th-6th education_7th-8th education_9th
## 1       -0.07160928       -0.10153148       -0.14221071   -0.12669045
## 2        0.01977987        0.06043907        0.14193927    0.09523296
## 3        0.02825770        0.03730691        0.04737631    0.04486390
##   education_Assoc-acdm education_Assoc-voc education_Bachelors
## 1           0.41272102         -0.12294756           1.0997753
## 2          -0.04094108         -0.04649371          -0.1905628
## 3          -0.16908477          0.05536700          -0.4436222
##   education_Doctorate education_HS-grad education_Masters education_Preschool
## 1          0.27945535       -0.68287580         0.5898004         -0.03923047
## 2         -0.04068237       -0.07128623        -0.1200606          0.03002845
## 3         -0.11338446        0.29160075        -0.2363898          0.01384649
##   education_Prof-school education_Some-college marital_status_Married-AF-spouse
## 1            0.33331366             -0.5136342                     -0.005809287
## 2           -0.06018165              0.1347685                      0.014233367
## 3           -0.13424385              0.2032901                      0.001217056
##   marital_status_Married-civ-spouse marital_status_Married-spouse-absent
## 1                        0.18005704                         -0.015958624
## 2                       -0.22964810                          0.025647452
## 3                       -0.05573252                          0.004488879
##   marital_status_Never-married marital_status_Separated marital_status_Widowed
## 1                  -0.08684993              -0.08166643           -0.084920726
## 2                   0.19277999               0.02466636            0.305195595
## 3                   0.01989920               0.03204680            0.009520287
##   occupation_Armed-Forces occupation_Craft-repair occupation_Exec-managerial
## 1            -0.003344280             -0.2557799                  0.3640381
## 2            -0.016633586             -0.3793845                 -0.3778458
## 3             0.002814703              0.1392540                 -0.1200414
##   occupation_Farming-fishing occupation_Handlers-cleaners
## 1                 -0.1085629                  -0.17104863
## 2                 -0.1773307                  -0.20957553
## 3                  0.0604932                   0.08936601
##   occupation_Machine-op-inspct occupation_Other-service
## 1                   -0.2062305              -0.2409007
## 2                   -0.2559146              -0.3354471
## 3                    0.1080224               0.1292913
##   occupation_Priv-house-serv occupation_Prof-specialty
## 1                -0.05089065                 0.7399212
## 2                -0.06736689                -0.3816074
## 3                 0.02701523                -0.2768889
##   occupation_Protective-serv occupation_Sales occupation_Tech-support
## 1                -0.02253592       0.04088465              0.06652167
## 2                -0.14265994      -0.35545820             -0.17124620
## 3                 0.02157052       0.01317240             -0.01323297
##   occupation_Transport-moving occupation_Unknown relationship_Not-in-family
## 1                  -0.1755309         -0.2450356                 0.09524305
## 2                  -0.2271881          4.0809136                -0.03696142
## 3                   0.0927399         -0.2450356                -0.03667666
##   relationship_Other-relative relationship_Own-child relationship_Unmarried
## 1                 -0.10329682            -0.22571607            -0.10996614
## 2                  0.02358495             0.38340246            -0.03550863
```

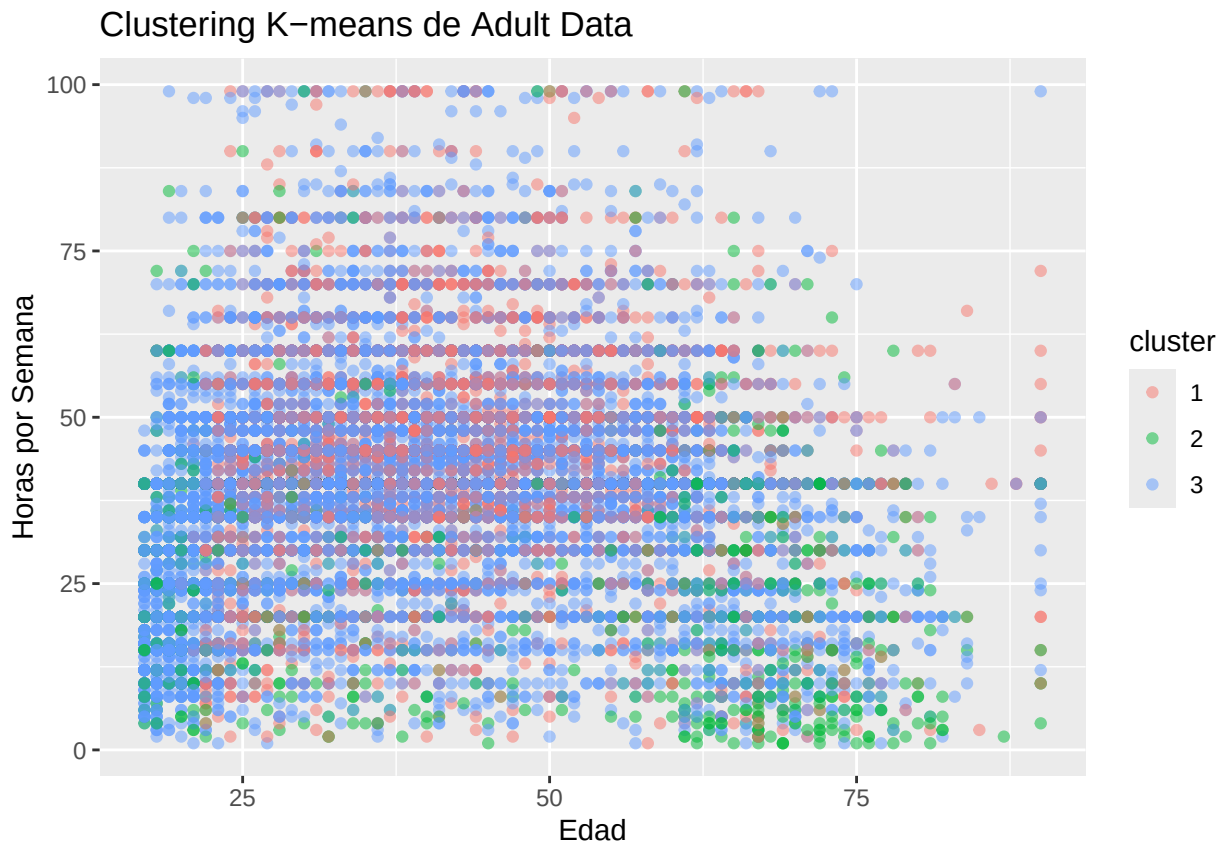
```
## 3                  0.04118320             0.06173165             0.04900369
##   relationship_Wife race_Asian-Pac-Islander  race_Black  race_Other  race_White
## 1        0.05049287              0.11524009 -0.12956983 -0.03450131  0.07545857
## 2        0.11193568              0.01915556  0.07029946  0.04567012 -0.09133823
## 3       -0.03064396             -0.04981641  0.04819095  0.01053718 -0.02377396
##       sex_Male native_country_Canada native_country_China
## 1  0.073132195            0.01333246           0.04640286
## 2 -0.266866580            0.06370105           0.03113420
## 3 -0.007854891           -0.01099888          -0.02205351
##   native_country_Columbia native_country_Cuba native_country_Dominican-Republic
## 1            -0.014052153         0.007323739                       -0.03212361
## 2            -0.004361003        -0.023944643                       -0.01130092
## 3             0.006246965        -0.001023372                        0.01439412
##   native_country_Ecuador native_country_El-Salvador native_country_England
## 1           -0.010506136              -3.390112e-02             0.03779970
## 2           -0.010843501              -3.979915e-05            -0.01134215
## 3            0.005316259               1.417845e-02            -0.01483939
##   native_country_France native_country_Germany native_country_Greece
## 1           0.040485471             0.03394805         -0.0002451299
## 2           0.006497429             0.01038945         -0.0298673849
## 3          -0.017481439            -0.01507936          0.0026457224
##   native_country_Guatemala native_country_Haiti
## 1              -0.04115959         -0.012743842
## 2              -0.03125171         -0.007268671
## 3               0.01987115          0.005947511
##   native_country_Holand-Netherlands native_country_Honduras native_country_Hong
## 1                      -0.005543847            -0.008934261         0.024239270
## 2                      -0.005543847             0.007157814        -0.002908593
## 3                       0.002790109             0.003126188        -0.009887491
##   native_country_Hungary native_country_India native_country_Iran
## 1            0.007652816           0.09219507          0.06095077
## 2           -0.019992312          -0.05552301         -0.02144177
## 3           -0.001497512          -0.03382168         -0.02365955
##   native_country_Ireland native_country_Italy native_country_Jamaica
## 1            0.001320789          0.001624087            -0.02334800
## 2           -0.027168798          0.009920241            -0.03906789
## 3            0.001761179         -0.001523794             0.01308915
##   native_country_Japan native_country_Laos native_country_Mexico
## 1          0.034850818       -0.0094293101           -0.11207040
## 2         -0.006368574       -0.0004517546           -0.01249354
## 3         -0.014029875        0.0039811421            0.04792380
##   native_country_Nicaragua native_country_Outlying-US(Guam-USVI-etc)
## 1             -0.018662470                               0.0005646154
## 2             -0.015548692                              -0.0207473219
## 3              0.009127315                               0.0015305648
##   native_country_Peru native_country_Philippines native_country_Poland
## 1        -0.012973794                0.060990203           0.003377249
## 2        -0.013294434               -0.008479157           0.007605904
## 3         0.006556758               -0.024779812          -0.002059776
##   native_country_Portugal native_country_Puerto-Rico native_country_Scotland
## 1            -0.020626441                -0.03124504           -1.943585e-03
## 2             0.014556874                -0.01338206            9.050615e-03
## 3             0.007385001                 0.01420397            4.200537e-05
##   native_country_South native_country_Taiwan native_country_Thailand
```

```
## 1            0.02398100            0.06651698            0.0187654667
## 2            0.04895672            0.08381824           -0.0004517546
## 3           -0.01419586           -0.03494990           -0.0078079324
##   native_country_Trinadad&Tobago native_country_United-States
## 1                    -0.015024022                 -0.028145501
## 2                    -0.001711910                  0.026093527
## 3                     0.006427761                  0.009546591
##   native_country_Unknown native_country_Vietnam native_country_Yugoslavia
## 1             0.06762660           -0.008862017                0.012707681
## 2            -0.02442435           -0.009516689               -0.022180502
## 3            -0.02619695            0.004515825               -0.003424776
```

```r
# Ver la cantidad de observaciones en cada cluster
table(adult_data$cluster)
```

```
##
##     1     2     3
##  9050  1843 21644
```

```r
# Crear un gráfico de dispersión de dos variables, coloreado por cluster
ggplot(adult_data, aes(x = age, y = hours_per_week, color = cluster)) +
  geom_point(alpha = 0.5) +
  labs(title = "Clustering K-means de Adult Data", x = "Edad", y = "Horas por Semana")
```



```r
install.packages("plotly")
```

```
## Installing package into 'C:/Users/riosl/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)

## package 'plotly' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\riosl\AppData\Local\Temp\RtmpwRz86I\downloaded_packages
```

```r
library(plotly)
```

```
##
## Adjuntando el paquete: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout
```

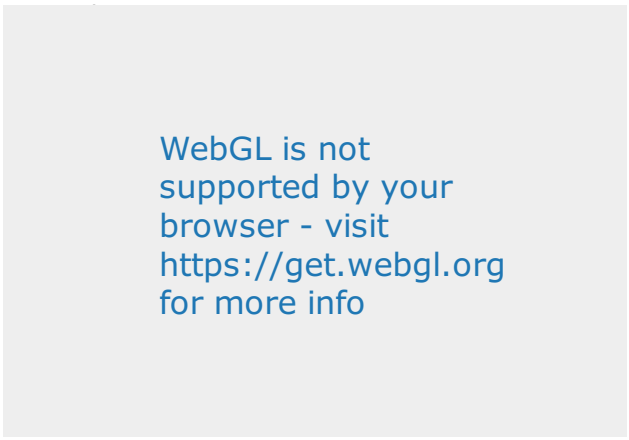
```r
# Filtrar los datos para income <= 50K
data_income_low <- subset(adult_data, income == "<=50K")

# Crear el gráfico de dispersión en 3D para income <= 50K
plot_low_income <- plot_ly(data_income_low, x = ~age, y = ~hours_per_week, z = ~education_num, color =
  add_markers() %>%
  layout(scene = list(xaxis = list(title = 'Edad'),
                      yaxis = list(title = 'Horas por Semana'),
                      zaxis = list(title = 'Nivel Educativo')),
         title = 'Clustering K-means para Income <= 50K')

# Filtrar los datos para income > 50K
data_income_high <- subset(adult_data, income == ">50K")

# Crear el gráfico de dispersión en 3D para income > 50K
plot_high_income <- plot_ly(data_income_high, x = ~age, y = ~hours_per_week, z = ~education_num, color =
  add_markers() %>%
  layout(scene = list(xaxis = list(title = 'Edad'),
                      yaxis = list(title = 'Horas por Semana'),
                      zaxis = list(title = 'Nivel Educativo')),
         title = 'Clustering K-means para Income > 50K')

# Mostrar los gráficos
plot_low_income
```




```
plot_high_income
```

Conclusiones:

El tamaño de cada cluster nos da una idea de la distribución de los datos:

- **Cluster 1**: 9064 observaciones. Agrupa a personas con mayores niveles de educación y roles profesionales más estables o de mayor ingreso.
- **Cluster 2**: 1843 observaciones. Incluye a personas con menor estabilidad laboral o niveles de educación más bajos, posiblemente en situaciones de empleo menos consistentes.Relacionado con la edad ,en este grupo se encuentra gran cantidad de personas mayores a 60 años.
- **Cluster 3**: 21654 observaciones. Representa una gran parte de la población en roles de trabajo más típicos, con educación y ganancias en línea con la media o ligeramente por debajo del promedio.

Prueba por contraste de hipotesis

Mediante una prueba por contraste de hipotesis podemos determinar si hay diferencias significativas entre grupos o si una observación específica es significativa en este análisis.

Dado que estamos comparando más de dos grupos (clusters), una prueba ANOVA (análisis de varianza) sería lo mas apropiado.

Definicion de hipotesis: evaluamos si hay una diferencia significativa en las edades (age) entre los clusters:

- **Hipótesis nula (H0)**: No hay diferencia en las edades entre los clusters.
- **Hipótesis alternativa (H1)**: Hay una diferencia en las edades entre los clusters.

```r
install.packages("car")
```

```
## Installing package into 'C:/Users/riosl/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)

## package 'car' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\riosl\AppData\Local\Temp\RtmpwRz86I\downloaded_packages
```

```r
library(car)
```

```
## Cargando paquete requerido: carData

##
## Adjuntando el paquete: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
# Realizamos la prueba ANOVA
anova_result <- aov(age ~ cluster, data = adult_data)

# Imprimimos de los resultados de ANOVA
summary(anova_result)
```

```
##                Df  Sum Sq Mean Sq F value Pr(>F)
## cluster         2   59625   29813   161.9 <2e-16 ***
## Residuals   32534 5991895     184
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusiones: El valor p (Pr(>F)) es menor que 0.05, por lo que rechazamos la hipótesis nula y concluimos que hay diferencias significativas en las edades entre los clusters.

```r
# Realizamos la prueba de Tukey
tukey_result <- TukeyHSD(anova_result)

# Imprimimos los resultados de la prueba de Tukey
print(tukey_result)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = age ~ cluster, data = adult_data)
##
## $cluster
##           diff        lwr       upr     p adj
## 2-1  0.4749545 -0.3378803  1.287789 0.3570127
## 3-1 -2.7798653 -3.1780175 -2.381713 0.0000000
## 3-2 -3.2548197 -4.0266074 -2.483032 0.0000000
```

Conclusiones: Hay una diferencia significativa en las edades entre el Cluster 3 y el Cluster 1. El intervalo de confianza no incluye 0, y el valor p es menor que 0.05. Esto sugiere que las edades en el Cluster 3 son significativamente menores que en el Cluster1.

También hay una diferencia significativa en las edades entre el Cluster 3 y el Cluster 2. El intervalo de confianza no incluye 0, y el valor p es menor que 0.05. Esto sugiere que las edades en el Cluster 3 son significativamente menores que en el Cluster 2.

Contribuciones	Firma
Investigación previa	Francisco Javier González Ontañón y Laureano Rios Oriol
Redacción de las respuestas	Francisco Javier González Ontañón y Laureano Rios Oriol
Desarrollo del código	Francisco Javier González Ontañón y Laureano Rios Oriol
Participación en el vídeo	Francisco Javier González Ontañón y Laureano Rios Oriol