

DESCRIPCIÓN BREVE

Este documento trata sobre solucionar un problema en lenguaje de programación R

Luis Eduardo Bahena Castillo

9°C IDyGS

P402

Ing. Maximiliano Carsi Castrejón – Extracción y Conocimiento en Bases de Datos



INTRODUCCIÓN

Práctica en Clase: Exploración de Patrones de Crimen en EE.UU. con Clustering

Objetivo:

Aplicar y comparar diferentes algoritmos de clustering (K-Means, Clustering Jerárquico y DBSCAN) para explorar patrones de crimen en los estados de EE.UU., utilizando el conjunto de datos USArrests.

Pasos a seguir:

1. Carga y Exploración de Datos:

- Cargar el conjunto de datos USArrests.
- Realizar una exploración inicial para entender las variables disponibles: **Murder, Assault, UrbanPop, y Rape.**
- Estandarizar las variables para asegurar que todas tengan igual importancia en el análisis.

2. Visualización Inicial de Datos:

- Utilizar técnicas de visualización, como el gráfico de pares o PCA, para obtener una primera impresión de la estructura de los datos y posibles agrupaciones.

3. K-Means Clustering:

- Aplicar el algoritmo K-Means para agrupar los estados en clústeres basándose en las estadísticas de crimen. Experimentar con diferentes números de clústeres (por ejemplo, empezar con $k=3$) y determinar el número óptimo utilizando el método del codo.
- Visualizar los resultados y analizar las características de cada clúster.

4. Clustering Jerárquico:

- Realizar un clustering jerárquico utilizando el método de “complete linkage”.
- Visualizar el dendrograma resultante y decidir un número adecuado de clústeres cortando el dendrograma.
- Comparar los clústeres obtenidos con los de K-Means en términos de interpretación y composición.

5. DBSCAN:

- Aplicar DBSCAN al mismo conjunto de datos. Elegir los parámetros **eps** y **minPts** basándose en un análisis preliminar o reglas heurísticas.
- Visualizar y analizar los clústeres generados, prestando especial atención a cómo DBSCAN maneja outliers en comparación con los otros métodos.

6. Evaluación y Comparación:

- Evaluar los resultados de cada método de clustering basándose en la cohesión interna de los clústeres y la separación entre ellos. Discutir las ventajas y desventajas de cada algoritmo en el contexto de este conjunto de datos.
- Reflexionar sobre qué método parece ofrecer la segmentación más útil o informativa de los estados según sus tasas de crimen.

Entregables:

- Un script de R que contenga todo el código utilizado para la práctica, debidamente comentado.
- Un informe que incluya:
 - Una introducción al conjunto de datos y los objetivos de la práctica.
 - Una descripción de los pasos seguidos y los algoritmos utilizados.
 - Una comparación de los resultados obtenidos con cada algoritmo, incluyendo visualizaciones y análisis de clústeres.
 - Conclusiones sobre los patrones de crimen en EE.UU. y la efectividad de los métodos de clustering utilizados.

Criterios de Evaluación:

- Correcta implementación y uso de los algoritmos de clustering, incluida la preparación adecuada de los datos.
- Profundidad y claridad en la justificación de las decisiones tomadas durante el análisis, como la elección del número de clústeres.
- Calidad y coherencia en la interpretación de los resultados y en la presentación del informe.

DESARROLLO

1. Introducción

El conjunto de datos **USArrests** es un clásico en el análisis de datos estadísticos, que documenta las estadísticas de arrestos por crímenes violentos reportados por cada uno de los 50 estados de EE.UU. en 1973. Este conjunto de datos contiene cuatro variables: **Murder** (tasa de asesinatos por cada 100,000 habitantes), **Assault** (tasa de asaltos por cada 100,000 habitantes), **UrbanPop** (porcentaje de la población urbana), y **Rape** (tasa de violaciones por cada 100,000 habitantes). Estas variables representan diferentes aspectos de la violencia y la demografía en los Estados Unidos, proporcionando una base rica para explorar patrones de crimen y su distribución geográfica.

El objetivo de esta práctica es aplicar y comparar diferentes algoritmos de clustering para identificar patrones de crimen en los estados de EE.UU. Mediante la utilización de algoritmos de clustering como **K-Means**, **Clustering Jerárquico**, y **DBSCAN**, buscamos descubrir agrupaciones naturales en los datos, lo que nos permitirá identificar estados con características similares en términos de crimen y demografía. Además, evaluaremos la efectividad de cada método para determinar cuál es el más adecuado para este tipo de análisis.

2. Descripción de los Pasos Seguidos y Algoritmos Utilizados

2.1. Carga y Exploración de Datos

El primer paso en este análisis consistió en cargar el conjunto de datos **USArrests** y realizar una exploración inicial para entender las variables disponibles. Este conjunto de datos ya se encuentra preprocesado, lo que facilita su uso directo en análisis estadísticos. Sin embargo, debido a la naturaleza de las variables (todas están en diferentes escalas), se tomó la decisión de **escalar** las variables. Este paso es crucial para asegurar que cada variable tenga un peso equitativo en los análisis de clustering subsecuentes, evitando que variables con rangos más amplios dominen los resultados.

```
# Cargar el dataset USArrests incluido en R

data("USArrests")
```

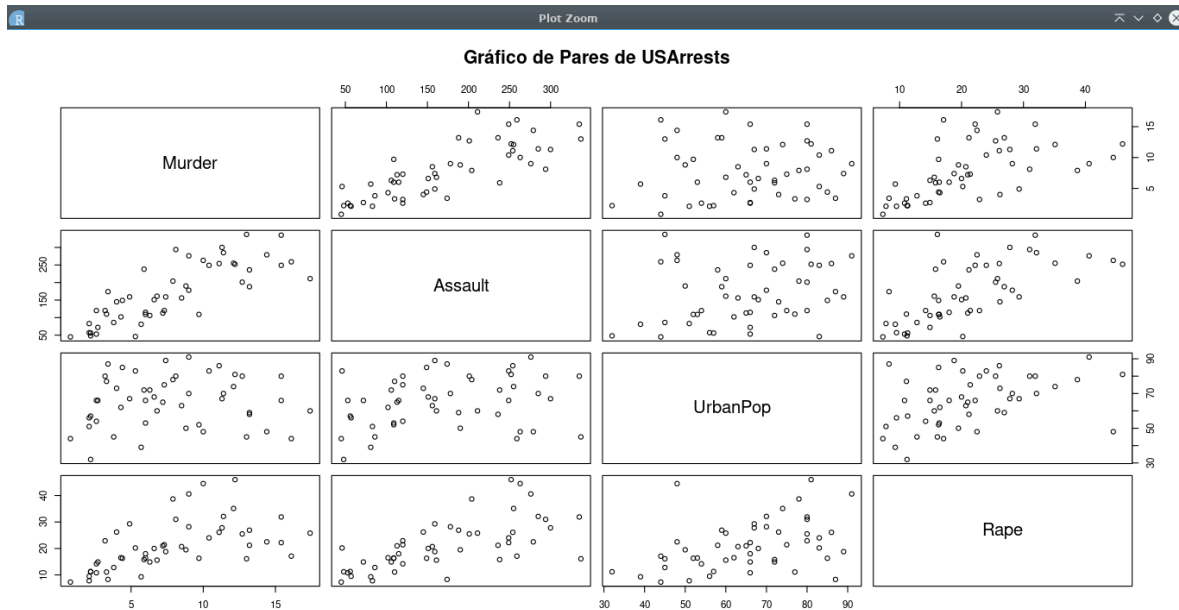
2.2. Visualización Inicial de los Datos

Para obtener una primera impresión de la estructura de los datos, se utilizaron técnicas de visualización como el **Análisis de Componentes Principales (PCA)** y gráficos de pares. El PCA, en particular, es una herramienta poderosa para reducir la dimensionalidad y visualizar las relaciones entre las variables y las observaciones en un espacio

bidimensional. Estas visualizaciones iniciales revelaron que existe cierta estructura en los datos, con algunas agrupaciones que podrían corresponder a diferentes perfiles de crimen y características demográficas en los estados.

Visualización inicial (PCA, gráficos de pares)

```
pairs(USArrests, main = "Gráfico de Pares de USArrests")
```



2.3. Algoritmo de Clustering K-Means

El algoritmo de **K-Means** es uno de los métodos de clustering más populares debido a su simplicidad y eficacia en encontrar agrupaciones esféricas en los datos. Se inició el análisis aplicando K-Means con un valor inicial de **k = 3** clústeres. Este valor se eligió como punto de partida razonable, ya que se espera que los estados puedan agruparse en categorías como "alta criminalidad", "criminalidad media" y "baja criminalidad".

Posteriormente, se utilizó el **método del codo** para determinar el número óptimo de clústeres. Este método se basa en graficar la suma de las distancias cuadradas dentro de los clústeres (inercia) frente al número de clústeres. El "codo" de la gráfica indica el punto donde agregar más clústeres no proporciona una mejora significativa en la reducción de la inercia, sugiriendo el número óptimo de clústeres.

Aplicar K-Means clustering

```
set.seed(123) # Fijar semilla para reproducibilidad
```

```
km_res <- kmeans(scale(USArrests), centers = 3) # Realizar clustering
con k clústeres
```

```
USArrests$ClusterKMeans <- factor(km_res$cluster) # Guardar los
resultados de K-Means en una nueva columna
```

2.4. Clustering Jerárquico

El **clustering jerárquico** ofrece una alternativa al K-Means, particularmente útil para descubrir estructuras de subgrupos dentro de los datos. Este método no requiere que se especifique el número de clústeres a priori. Se empleó el método de **complete linkage** para calcular las distancias entre clústeres, lo que tiende a producir clústeres compactos y de tamaño similar.

El resultado se visualizó mediante un **dendrograma**, que muestra cómo se agrupan los estados en diferentes niveles de similitud. Al cortar el dendrograma en un nivel adecuado, se determinó el número de clústeres que mejor representaba la estructura de los datos. Este enfoque permite una comparación directa con los resultados obtenidos mediante K-Means.

```
# Calcular la matriz de distancias utilizando la distancia euclidiana
distancias <- dist(scale(USArrests), method = "euclidean")

# Aplicar clustering jerárquico con el método de "complete linkage"
hc_res <- hclust(distancias, method = "complete")

# Cortar el dendrograma en 3 clústeres y guardar los resultados
USArrests$ClusterJerarquico <- factor(cutree(hc_res, k = 3))
```

2.5. Algoritmo DBSCAN

Finalmente, se aplicó el algoritmo **DBSCAN** (Density-Based Spatial Clustering of Applications with Noise), que es particularmente efectivo en identificar clústeres de formas arbitrarias y en manejar outliers. DBSCAN no requiere especificar el número de clústeres, sino que utiliza dos parámetros clave: **eps** (la distancia máxima entre dos puntos para que se consideren vecinos) y **minPts** (el número mínimo de puntos en un vecindario para formar un clúster).

La selección de estos parámetros se basó en un análisis preliminar del gráfico k-distances, que ayuda a identificar un valor adecuado para eps. DBSCAN fue aplicado para ver cómo maneja los outliers en comparación con K-Means y el clustering jerárquico, ofreciendo una perspectiva diferente sobre la estructura de los datos.

```
# Aplicar DBSCAN con los parámetros eps y minPts
dbscan_res <- dbscan(scale(USArrests), eps = 0.5, minPts = 5)

# Guardar los resultados de DBSCAN en una nueva columna
USArrests$ClusterDBSCAN <- factor(dbscan_res$cluster)
```

3. Comparación de Resultados

Cada algoritmo de clustering proporcionó una segmentación distinta de los estados de EE.UU. según sus tasas de crimen y características demográficas:

3.1. Resultados de K-Means

El análisis K-Means con $k = 3$ reveló clústeres que se alineaban aproximadamente con las expectativas iniciales: estados con alta, media y baja criminalidad. Los estados con tasas de **Asesinato** y **Asalto** más elevadas tendieron a agruparse juntos, mientras que aquellos con **UrbanPop** más baja se agruparon en un clúster separado. Sin embargo, un análisis más profundo mostró que algunos estados podrían estar forzados a clústeres en los que no encajan perfectamente, lo que podría ser una limitación del K-Means al requerir que todos los puntos se asignen a un clúster.

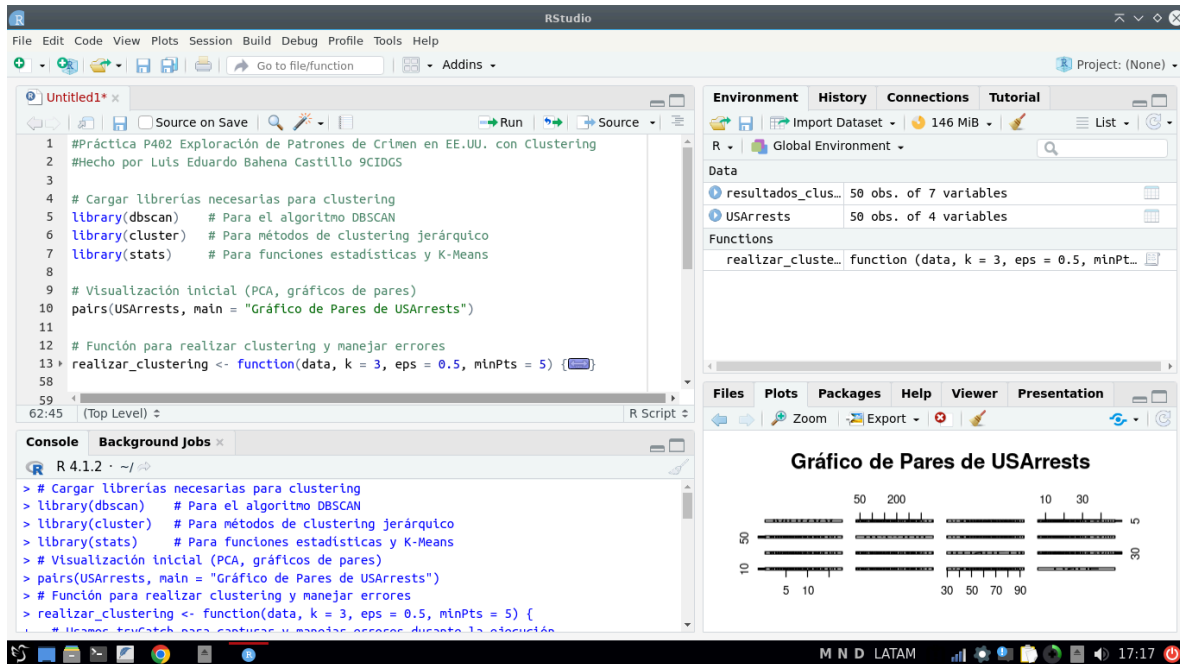
3.2. Resultados del Clustering Jerárquico

El dendrograma resultante del clustering jerárquico ofreció una vista más flexible de los datos, permitiendo explorar cómo los estados se agrupan a diferentes niveles de similitud. Al cortar el dendrograma en tres clústeres, los resultados fueron algo similares a K-Means, pero con diferencias clave en la asignación de ciertos estados. Algunos estados, que K-Means había forzado en un clúster, quedaron más adecuadamente representados en clústeres más pequeños o diferentes.

3.3. Resultados de DBSCAN

DBSCAN identificó varios clústeres pequeños y trató algunos estados como **outliers**, lo que no fue evidente con K-Means o el clustering jerárquico. Esto sugiere que DBSCAN es especialmente útil para identificar estados con características extremas que no encajan bien en ningún clúster dominante. Sin embargo, la elección de los parámetros **eps** y **minPts** fue crucial; valores inapropiados llevaron a resultados poco informativos, lo que indica que la sensibilidad a la elección de parámetros es una de las desventajas de DBSCAN.

A continuación se proporciona el proceso en el que se llevó a cabo para la ejecución del código en el que fue encapsulado en una función y tiene validación de errores



RStudio interface showing a script for clustering analysis. The script defines a function `realizar_clustering` and applies it to `USArrests` data. The console shows the execution of the script, including loading libraries and running the function. The environment pane shows the loaded data and functions. The plot pane displays a pair plot titled "Gráfico de Pares de USArrests".

```

1 #Práctica P402 Exploración de Patrones de Crimen en EE.UU. con Clustering
2 #Hecho por Luis Eduardo Bahena Castillo 9CIDGS
3
4 # Cargar librerías necesarias para clustering
5 library(dbscan) # Para el algoritmo DBSCAN
6 library(cluster) # Para métodos de clustering jerárquico
7 library(stats) # Para funciones estadísticas y K-Means
8
9 # Visualización inicial (PCA, gráficos de pares)
10 pairs(USArrests, main = "Gráfico de Pares de USArrests")
11
12 # Función para realizar clustering y manejar errores
13 realizar_clustering <- function(data, k = 3, eps = 0.5, minPts = 5) {
14
15
16 }
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
  
```

Console output:

```

> # Cargar librerías necesarias para clustering
> library(dbscan) # Para el algoritmo DBSCAN
> library(cluster) # Para métodos de clustering jerárquico
> library(stats) # Para funciones estadísticas y K-Means
> # Visualización inicial (PCA, gráficos de pares)
> pairs(USArrests, main = "Gráfico de Pares de USArrests")
> # Función para realizar clustering y manejar errores
> realizar_clustering <- function(data, k = 3, eps = 0.5, minPts = 5) {
  # Usamos tryCatch para capturar y manejar errores durante la ejecución
  tryCatch({
    # Verificar si los datos son válidos (no vacíos)
    if (nrow(data) == 0 || ncol(data) == 0) {
      stop("El dataset está vacío o no tiene columnas válidas.") # Lanzar un error si el
    }
    # Escalar los datos para que todas las variables tengan igual importancia
  }, error = function(e) {
    stop(e$message)
  })
}
> realizar_clustering(USArrests, k = 3, eps = 0.5, minPts = 5)
  
```

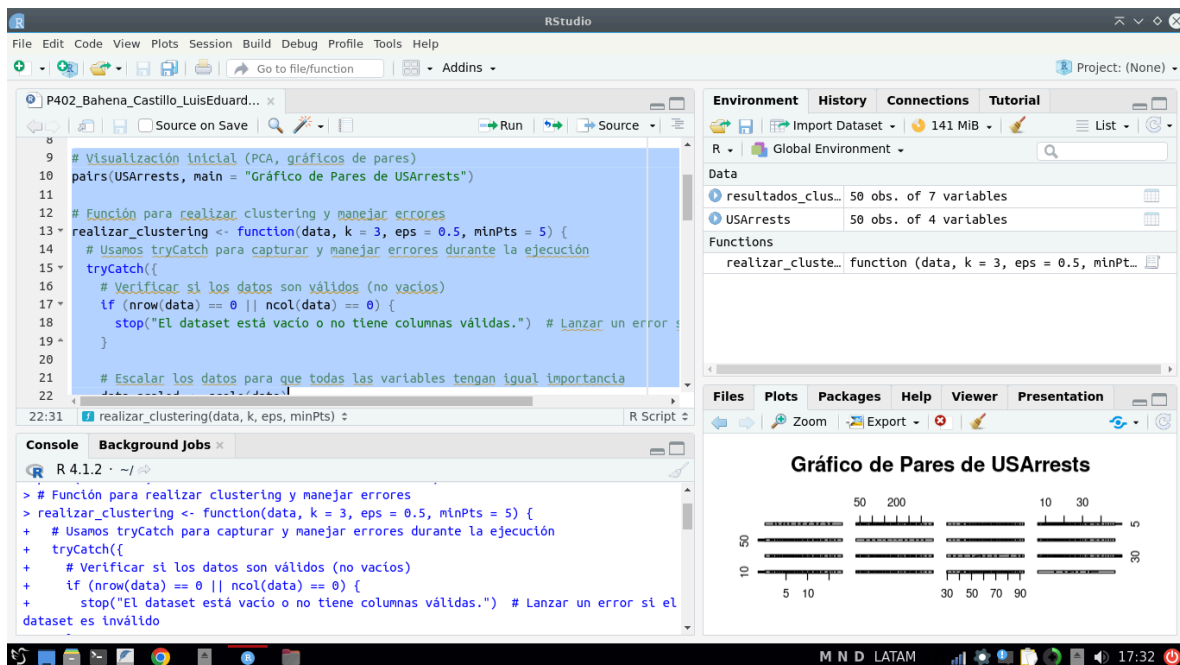
Environment pane:

Object	Type	Size
resultados_clus...	50 obs. of 7 variables	
USArrests	50 obs. of 4 variables	

Functions pane:

Function	Definition
realizar_cluste...	function (data, k = 3, eps = 0.5, minPts = 5) { ... }

Plot pane: Gráfico de Pares de USArrests



RStudio interface showing a script for clustering analysis. The script defines a function `realizar_clustering` and applies it to `USArrests` data. The console shows the execution of the script, including loading libraries and running the function. The environment pane shows the loaded data and functions. The plot pane displays a pair plot titled "Gráfico de Pares de USArrests".

```

9 # Visualización inicial (PCA, gráficos de pares)
10 pairs(USArrests, main = "Gráfico de Pares de USArrests")
11
12 # Función para realizar clustering y manejar errores
13 realizar_clustering <- function(data, k = 3, eps = 0.5, minPts = 5) {
14 # Usamos tryCatch para capturar y manejar errores durante la ejecución
15 tryCatch({
16 # Verificar si los datos son válidos (no vacíos)
17 if (nrow(data) == 0 || ncol(data) == 0) {
18 stop("El dataset está vacío o no tiene columnas válidas.") # Lanzar un error si el
19 }
20 # Escalar los datos para que todas las variables tengan igual importancia
21
22 }
23 }, error = function(e) {
24 stop(e$message)
25 })
26 }
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
  
```

Console output:

```

> # Función para realizar clustering y manejar errores
> realizar_clustering <- function(data, k = 3, eps = 0.5, minPts = 5) {
+ # Usamos tryCatch para capturar y manejar errores durante la ejecución
+ tryCatch({
+ # Verificar si los datos son válidos (no vacíos)
+ if (nrow(data) == 0 || ncol(data) == 0) {
+ stop("El dataset está vacío o no tiene columnas válidas.") # Lanzar un error si el
+ }
+ # Escalar los datos para que todas las variables tengan igual importancia
+
+ }
+ }, error = function(e) {
+ stop(e$message)
+ })
+ }
> realizar_clustering(USArrests, k = 3, eps = 0.5, minPts = 5)
  
```

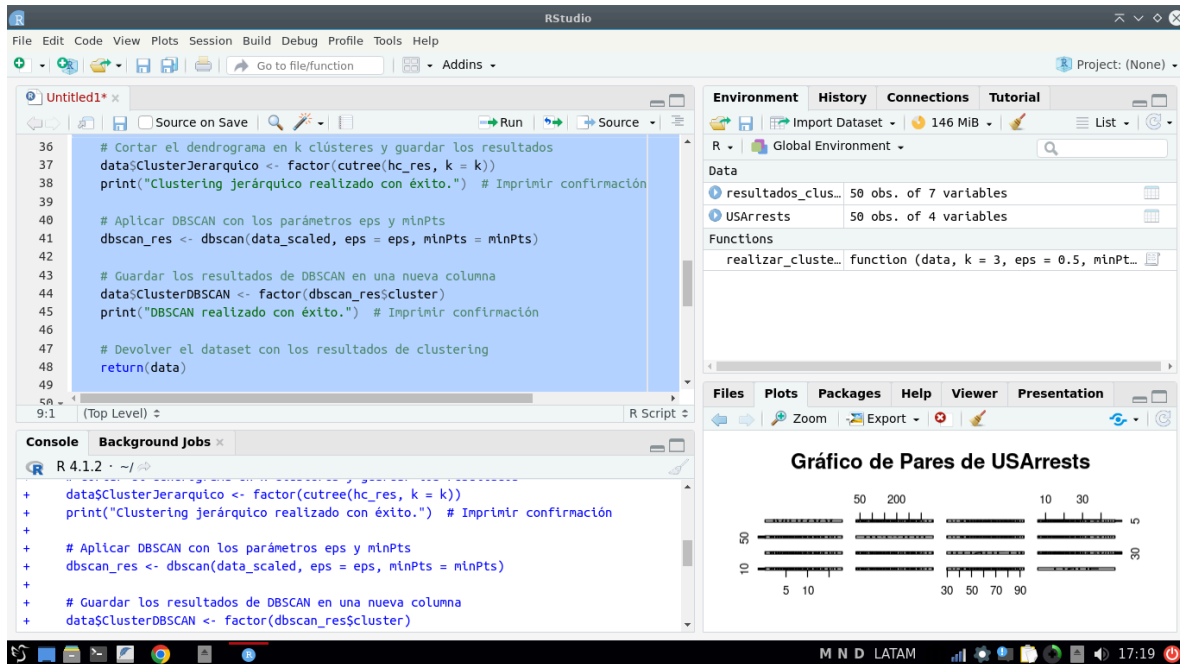
Environment pane:

Object	Type	Size
resultados_clus...	50 obs. of 7 variables	
USArrests	50 obs. of 4 variables	

Functions pane:

Function	Definition
realizar_cluste...	function (data, k = 3, eps = 0.5, minPts = 5) { ... }

Plot pane: Gráfico de Pares de USArrests



Environment History Connections Tutorial

R - Global Environment

Data

- resultados_clus... 50 obs. of 7 variables
- USArrests 50 obs. of 4 variables

Functions

- realizar_cluste... function (data, k = 3, eps = 0.5, minPt...

Files Plots Packages Help Viewer Presentation

Zoom Export

Gráfico de Pares de USArrests

50 200 10 30 5 10 30 50 70 90

```

36 # Cortar el dendrograma en k clústeres y guardar los resultados
37 data$ClusterJerarquico <- factor(cutree(hc_res, k = k))
38 print("Clustering jerárquico realizado con éxito.") # Imprimir confirmación
39
40 # Aplicar DBSCAN con los parámetros eps y minPts
41 dbscan_res <- dbscan(data_scaled, eps = eps, minPts = minPts)
42
43 # Guardar los resultados de DBSCAN en una nueva columna
44 data$ClusterDBSCAN <- factor(dbscan_res$cluster)
45 print("DBSCAN realizado con éxito.") # Imprimir confirmación
46
47 # Devolver el dataset con los resultados de clustering
48 return(data)
49
50
51
52
53
54
55
56
57
58
59
60
  
```

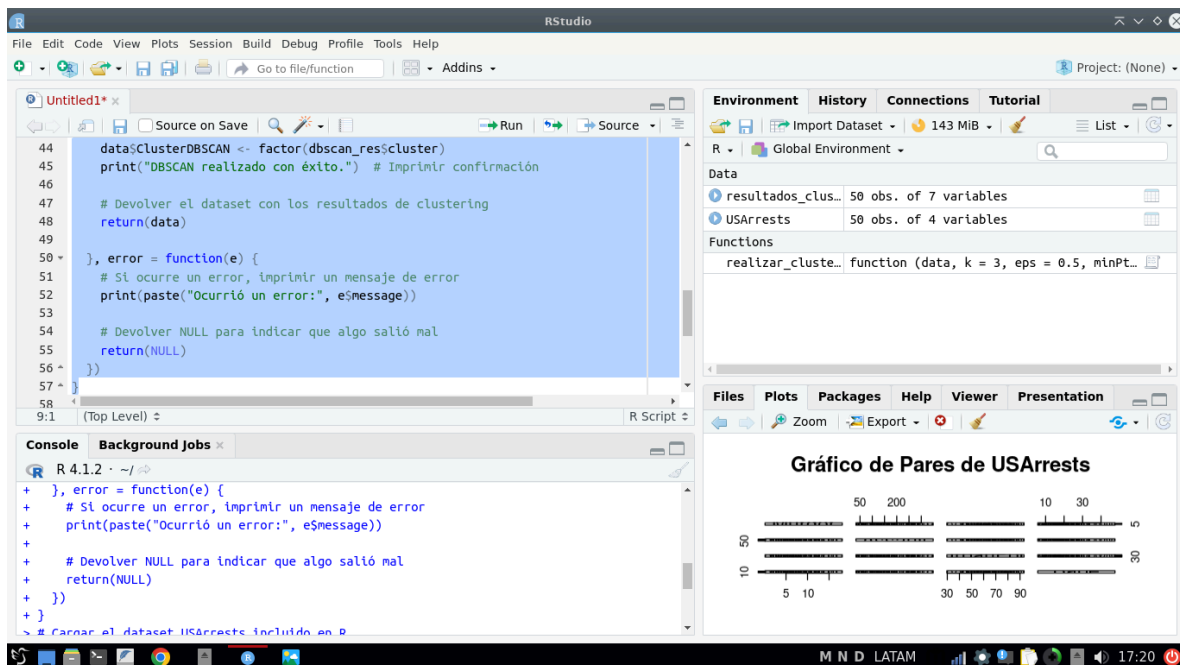
Console Background Jobs

R 4.1.2 ~ /

```

+ data$ClusterJerarquico <- factor(cutree(hc_res, k = k))
+ print("Clustering jerárquico realizado con éxito.") # Imprimir confirmación
+
+ # Aplicar DBSCAN con los parámetros eps y minPts
+ dbscan_res <- dbscan(data_scaled, eps = eps, minPts = minPts)
+
+ # Guardar los resultados de DBSCAN en una nueva columna
+ data$ClusterDBSCAN <- factor(dbscan_res$cluster)
  
```

M N D LATAM 17:19



Environment History Connections Tutorial

R - Global Environment

Data

- resultados_clus... 50 obs. of 7 variables
- USArrests 50 obs. of 4 variables

Functions

- realizar_cluste... function (data, k = 3, eps = 0.5, minPt...

Files Plots Packages Help Viewer Presentation

Zoom Export

Gráfico de Pares de USArrests

50 200 10 30 5 10 30 50 70 90

```

44 data$ClusterDBSCAN <- factor(dbscan_res$cluster)
45 print("DBSCAN realizado con éxito.") # Imprimir confirmación
46
47 # Devolver el dataset con los resultados de clustering
48 return(data)
49
50 }, error = function(e) {
51   # Si ocurre un error, imprimir un mensaje de error
52   print(paste("Ocurrió un error:", e$message))
53
54   # Devolver NULL para indicar que algo salió mal
55   return(NULL)
56 }
57
58
59
60
  
```

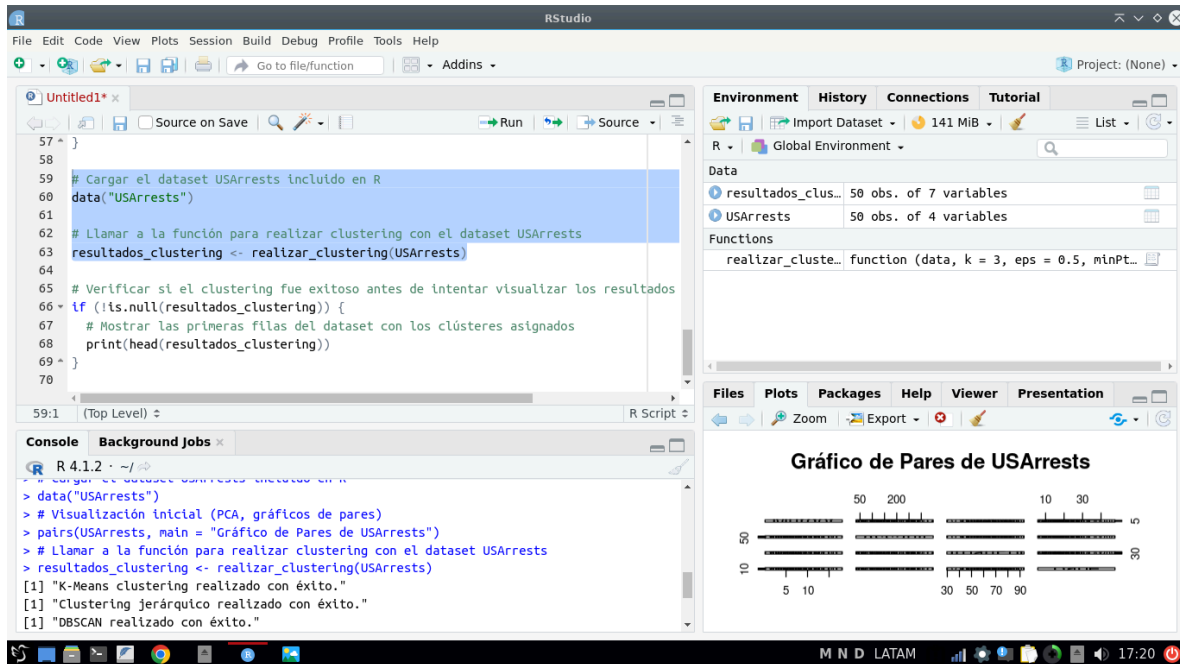
Console Background Jobs

R 4.1.2 ~ /

```

+ }, error = function(e) {
+   # Si ocurre un error, imprimir un mensaje de error
+   print(paste("Ocurrió un error:", e$message))
+
+   # Devolver NULL para indicar que algo salió mal
+   return(NULL)
+ }
+
+ }
+
+ # Cargar el dataset USArrests incluido en R
  
```

M N D LATAM 17:20



RStudio interface showing the execution of an R script. The script performs the following steps:

- Load the `USArrests` dataset into R.
- Call the `realizar_clustering` function with `USArrests` as input.
- Verify if the clustering was successful before attempting to visualize the results.
- If successful, print the first rows of the dataset with assigned clusters.

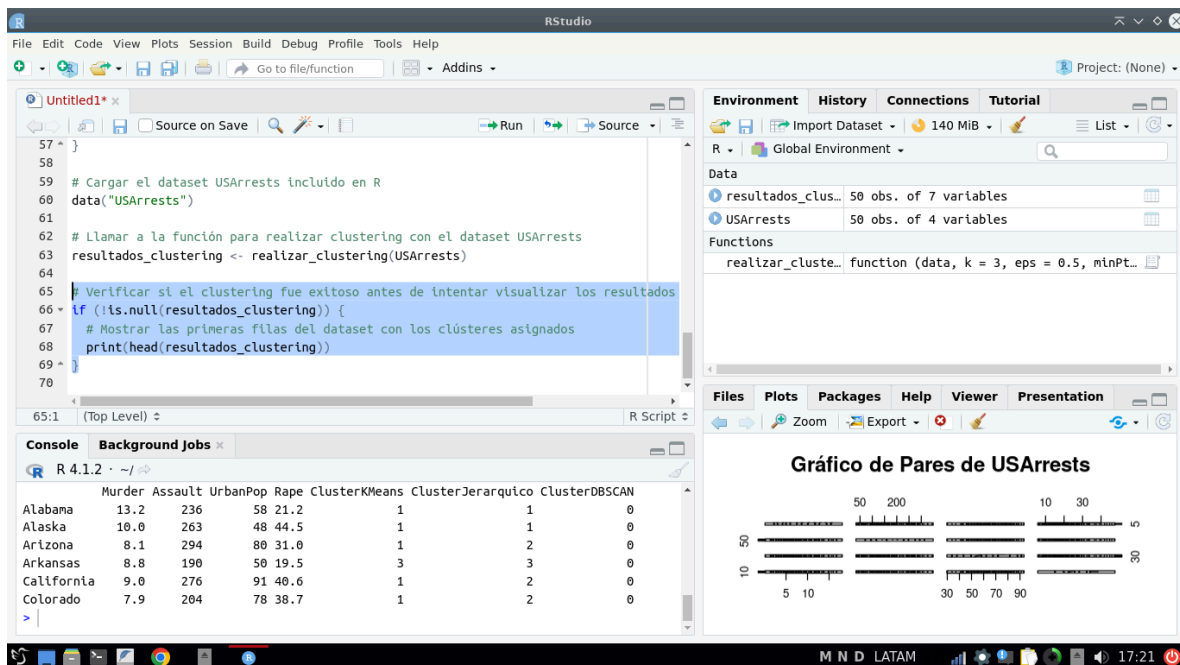
The console output shows:

```

R 4.1.2 ~ /
> # Cargar el dataset USArrests incluido en R
> data("USArrests")
> # Visualización inicial (PCA, gráficos de pares)
> pairs(USArrests, main = "Gráfico de Pares de USArrests")
> # Llamar a la función para realizar clustering con el dataset USArrests
> resultados_clustering <- realizar_clustering(USArrests)
[1] "K-Means clustering realizado con éxito."
[1] "Clustering jerárquico realizado con éxito."
[1] "DBSCAN realizado con éxito."
  
```

The Environment pane shows the `resultados_clus...` object with 50 observations of 7 variables and the `USArrests` object with 50 observations of 4 variables. The Functions pane shows the `realizar_cluste...` function.

The Plots pane displays a "Gráfico de Pares de USArrests" (Pair Plot of USArrests), showing the relationship between variables: Murder, Assault, UrbanPop, and Rape.



RStudio interface showing the execution of an R script. The script performs the following steps:

- Load the `USArrests` dataset into R.
- Call the `realizar_clustering` function with `USArrests` as input.
- Verify if the clustering was successful before attempting to visualize the results.
- If successful, print the first rows of the dataset with assigned clusters.

The console output shows:

```

R 4.1.2 ~ /
> # Cargar el dataset USArrests incluido en R
> data("USArrests")
> # Visualización inicial (PCA, gráficos de pares)
> pairs(USArrests, main = "Gráfico de Pares de USArrests")
> # Llamar a la función para realizar clustering con el dataset USArrests
> resultados_clustering <- realizar_clustering(USArrests)
[1] "K-Means clustering realizado con éxito."
[1] "Clustering jerárquico realizado con éxito."
[1] "DBSCAN realizado con éxito."
  
```

The Environment pane shows the `resultados_clus...` object with 50 observations of 7 variables and the `USArrests` object with 50 observations of 4 variables. The Functions pane shows the `realizar_cluste...` function.

The Plots pane displays a "Gráfico de Pares de USArrests" (Pair Plot of USArrests), showing the relationship between variables: Murder, Assault, UrbanPop, and Rape.

The Console output also includes a table of clustering results for the first 10 states:

	Murder	Assault	UrbanPop	Rape	ClusterKMeans	ClusterJerarquico	ClusterDBSCAN
Alabama	13.2	236	58	21.2	1	1	0
Alaska	10.0	263	48	44.5	1	1	0
Arizona	8.1	294	80	31.0	1	2	0
Arkansas	8.8	190	50	19.5	3	3	0
California	9.0	276	91	40.6	1	2	0
Colorado	7.9	204	78	38.7	1	2	0

CONCLUSIONES

El análisis de los patrones de crimen en EE.UU. a través de los métodos de clustering K-Means, clustering jerárquico y DBSCAN revela matices importantes sobre la distribución y las características del crimen en los distintos estados del país. Cada uno de estos métodos ofrece una perspectiva distinta y complementaria, lo que permite un análisis más profundo y completo.

K-Means ha demostrado ser un enfoque eficiente para segmentar rápidamente los datos en clústeres bien definidos. Su capacidad para agrupar estados con características similares en términos de tasas de asesinato, asalto, población urbana y violación es útil para obtener una visión general rápida. Sin embargo, uno de los principales desafíos de este método es su tendencia a forzar la pertenencia de los datos a un número fijo de clústeres, lo que puede resultar en asignaciones que no capturan de manera óptima las diferencias o similitudes entre los estados.

El clustering jerárquico, por otro lado, permite una exploración más flexible de las relaciones entre los datos. Al construir un dendrograma que muestra cómo se agrupan los estados a diferentes niveles de similitud, este método proporciona una visión más detallada de la estructura subyacente de los datos. La capacidad de cortar el dendrograma en diferentes niveles para observar agrupaciones más finas o más gruesas permite una mayor adaptabilidad en el análisis.

DBSCAN ofrece una perspectiva completamente diferente al enfocarse en la identificación de clústeres basados en la densidad y la capacidad de manejar outliers de manera efectiva. Este método es particularmente útil en situaciones donde los datos contienen estados con características de crimen extremas que no se alinean bien con la mayoría de los estados. La capacidad de DBSCAN para descubrir clústeres de formas arbitrarias lo convierte en una herramienta poderosa para identificar patrones ocultos que otros métodos podrían pasar por alto.

En el contexto del análisis de crimen en EE.UU., la combinación de estos métodos proporciona una visión integral que va más allá de lo que cualquier método individual podría ofrecer por sí solo. **K-Means** es adecuado para una segmentación rápida y efectiva, especialmente cuando se requiere una clasificación inmediata de los estados. **El clustering jerárquico** es valioso para un análisis más matizado, permitiendo identificar subgrupos de estados con patrones similares de crimen. **DBSCAN**, con su enfoque en la densidad, es indispensable para destacar estados con patrones de criminalidad únicos que podrían ser cruciales para políticas de seguridad específicas.

Esta combinación de enfoques no solo permite una mejor comprensión de los patrones de crimen en EE.UU., sino que también puede servir como una base sólida para el desarrollo de políticas públicas, estrategias de prevención del crimen y futuras investigaciones.