TP4: Aprendizaje no supervisado

Dey, Patrick
Lombardi, Matías
Vázquez, Ignacio

Tecnologías utilizadas

Ejercicio 1: Análisis del dataset

Uso del dataset

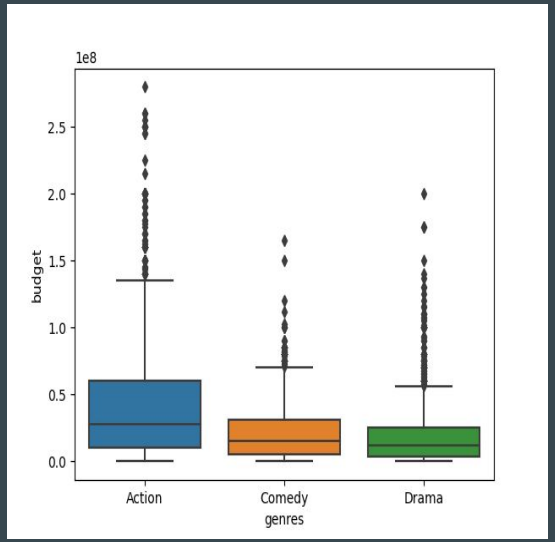
- Eliminamos entradas repetidas en función del valor imdb_id
- Eliminamos entradas incompletas
- Dado que los datos se encuentran en escalas muy variadas, estandarizamos los datos (con el uso de StandardScaler)

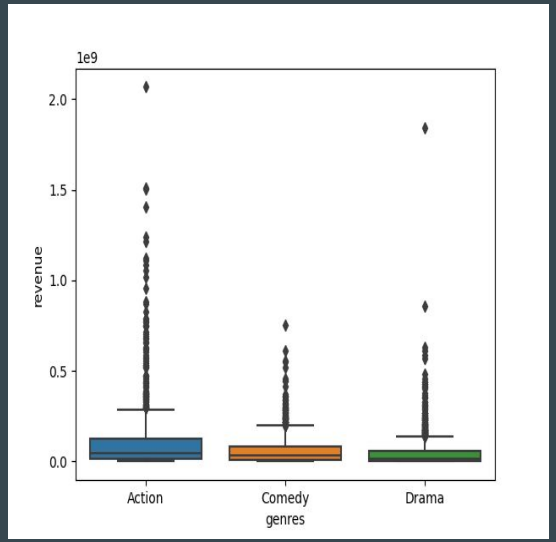
Análisis desvío estándar

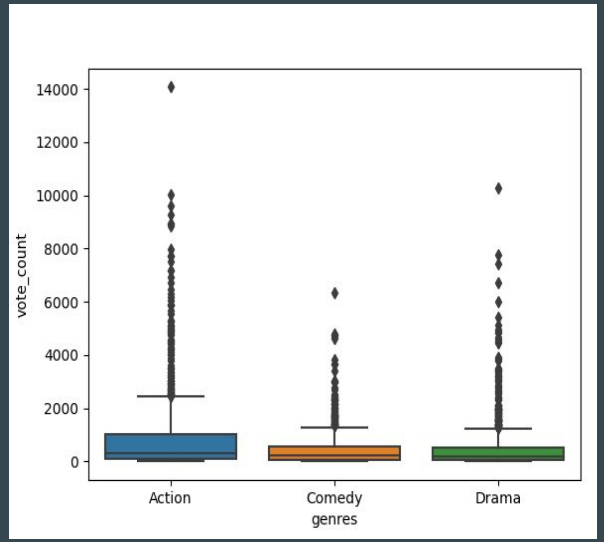
- Vamos a considerar las 3 variables con mayor desvío

revenue	1.409100e+08
budget	3.584321e+07
vote_count	1.109161e+03
runtime	2.182979e+01
popularity	5.833123e+00
production_companies	2.241016e+00
spoken_languages	9.401014e-01
vote_average	9.077369e-01
production_countries	7.708791e-01

Boxplots por género de budget, revenue y vote_count

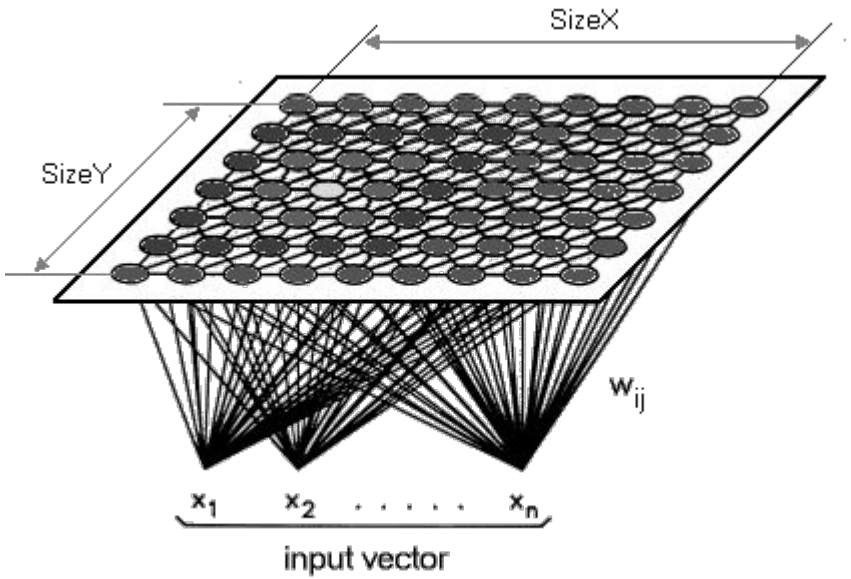






Drama 1328	39%
Comedia 1095	32%
Acción 997	29%

Kohonen



Implementación

- Se crea una grilla bidimensional (rectangular) de KxK y se recibe una tasa de aprendizaje y un radio (para los vecinos) iniciales y finales
- Cada una de las neuronas de la grilla se la inicializa con los pesos iguales a algunos de los ejemplos de entrada de manera aleatoria
- Se realiza una cantidad de iteraciones:
  - Se elige un ejemplo de manera aleatoria
  - Se busca la neurona ganadora (menor distancia euclídea al ejemplo presentado)
  - Se actualiza la neurona ganadora y sus vecinos (según el radio)
  - Se reduce el radio y la tasa de aprendizaje

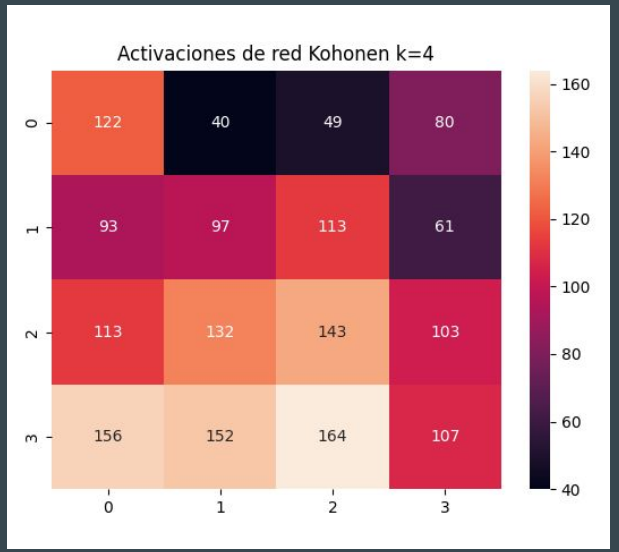
Resultados

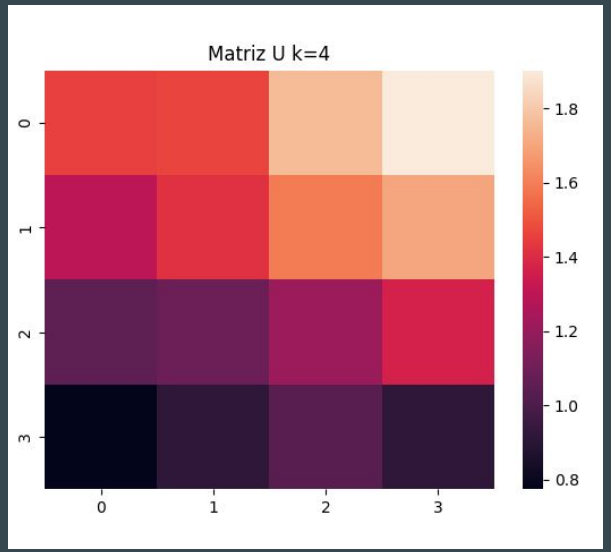
- La cantidad de iteraciones la tomamos en base a la cantidad de componentes de cada observación = #componentes * 2500
- El radio final será 1, para que actualice únicamente los vecinos adyacentes
- Radio inicial = tamaño grilla
- Tasa de aprendizaje inicial = 0.1
- Tasa de aprendizaje final = 0.0001
- k = [4, 5, 6]
- Para ver como cambia la cantidad de neuronas, vamos a utilizar las 3 variables mencionadas anteriormente

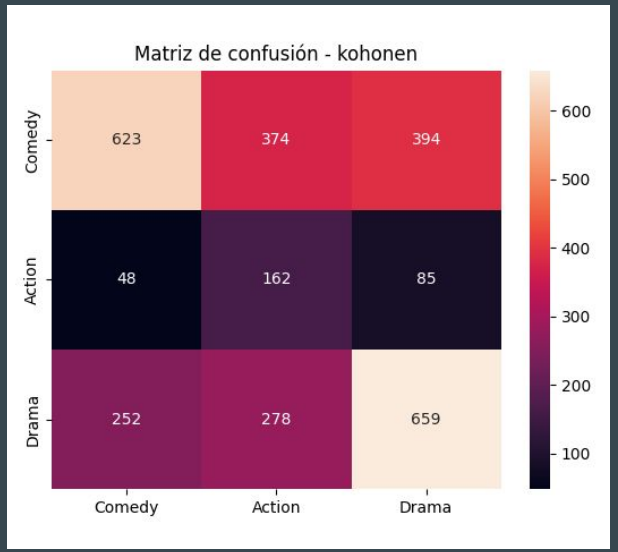
Resultados

- Primero vamos a ver si cambia utilizar todas las variables
- En todos los casos, vamos a usar los resultados para intentar predecir los géneros
  - Separamos un conjunto para testeo (20%)
  - Para cada una de las particiones corremos el algoritmo
  - Para predecir el género, cada neurona va a representar un género dependiendo de cuántas veces salió vencedora para cada una (es decir, si ganó 3 veces comedia, 2 drama y 1 acción, va a representar comedia)
  - Obtenemos una matriz de confusión que es la suma de todos los resultados
  - Obtenemos una precisión promedio
  - Obtenemos la matriz U y la cantidad de neuronas ganadoras para una de las particiones

Todas las variables (k=4)

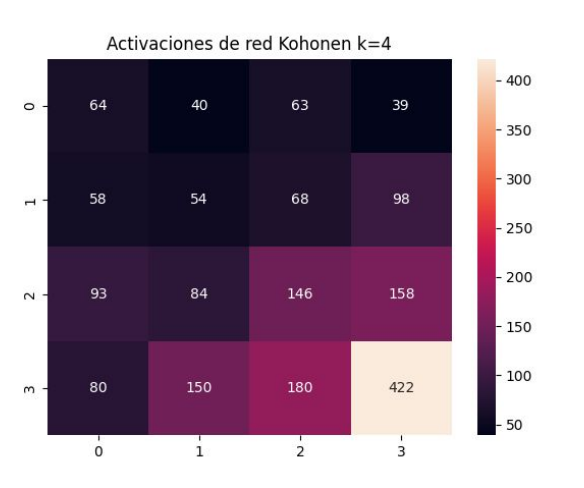


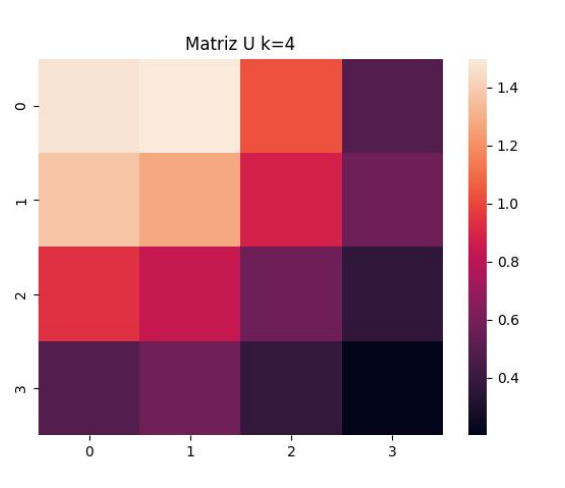


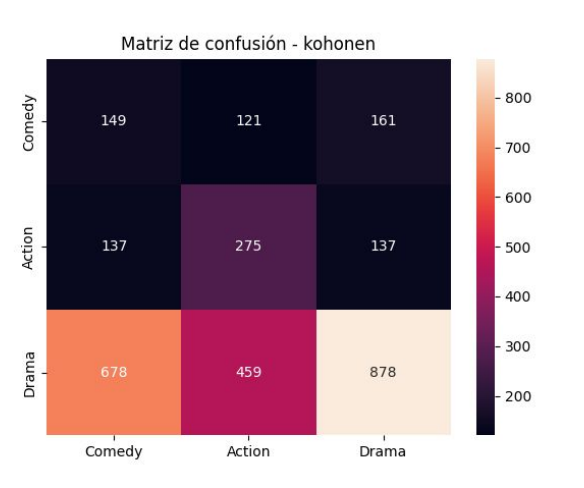


Precisión: 0.51

Variables budget, revenue, vote_count (k=4)

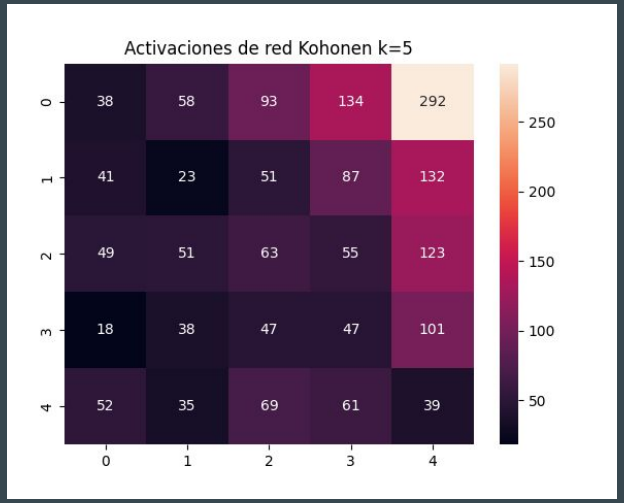


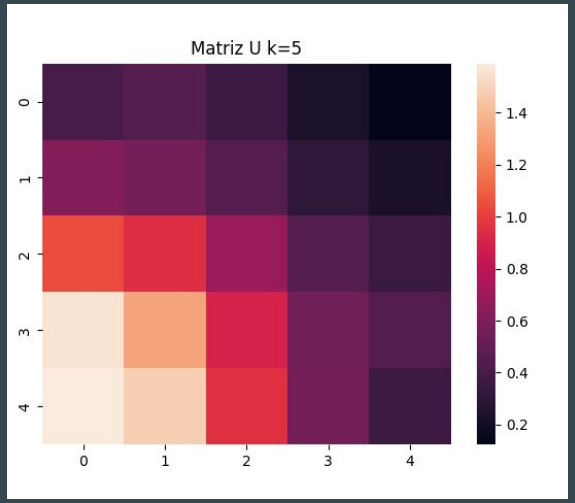


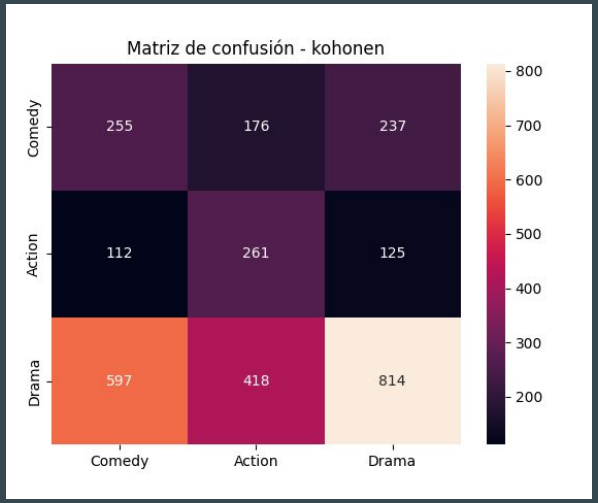


Precisión: 0.43

Variables budget, revenue, vote_count (k=5)

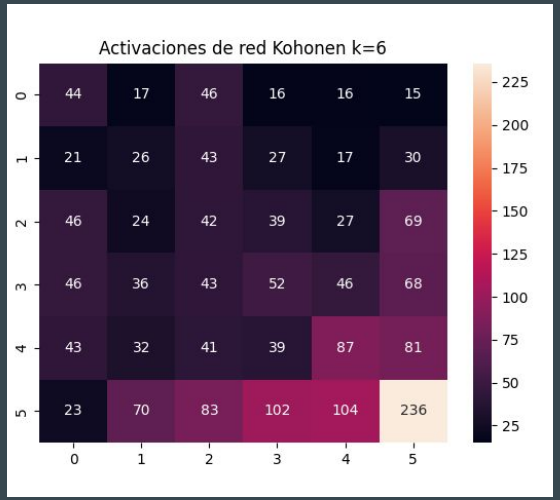


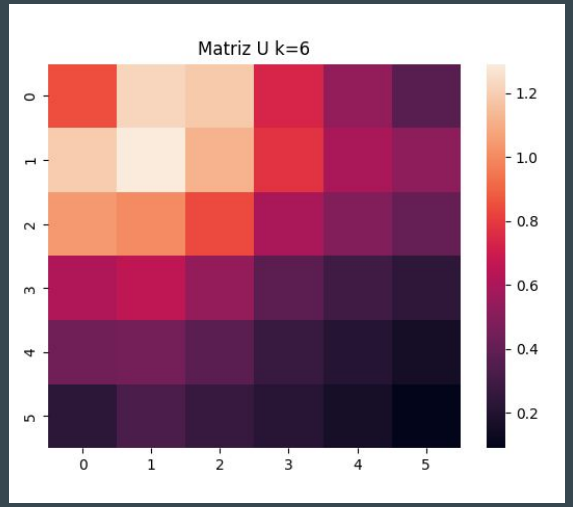


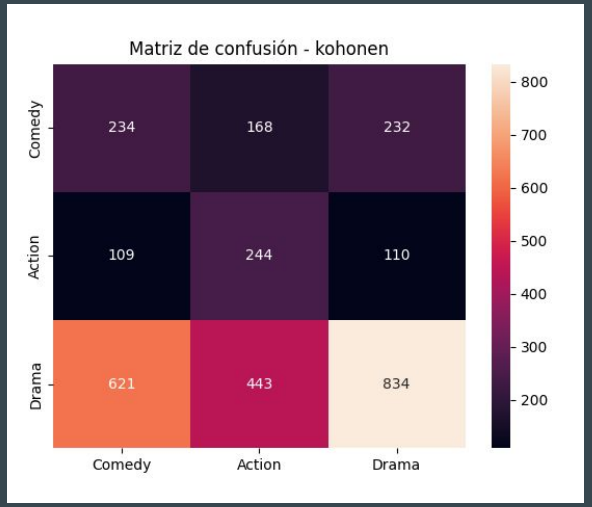


Precisión: 0.45

Variables budget, revenue, vote_count (k=6)

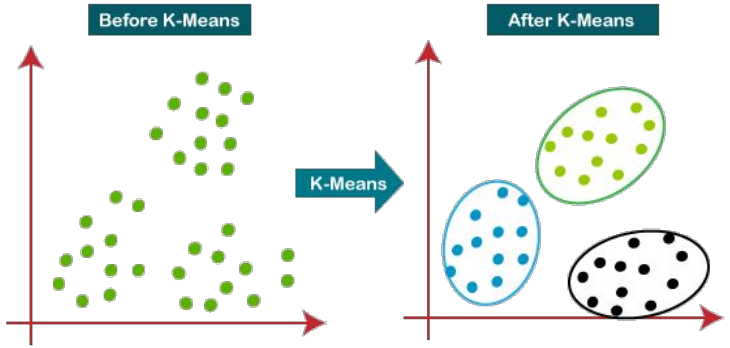






Precisión: 0.45

K-Medias



Implementación

1. Centroides iniciales: seleccionamos K entradas del dataset de manera aleatoria
2. Para todas las entradas del dataset calculamos la distancia euclídea con cada uno de los centroides y lo agregamos al cluster perteneciente al que menor distancia obtuvo
3. Utilizando la siguiente ecuación actualizamos cual debe ser el centroide del cluster

$$c_j^i = \frac{1}{|C_i|} \sum_{I=1}^{|C_i|} x_I^j$$

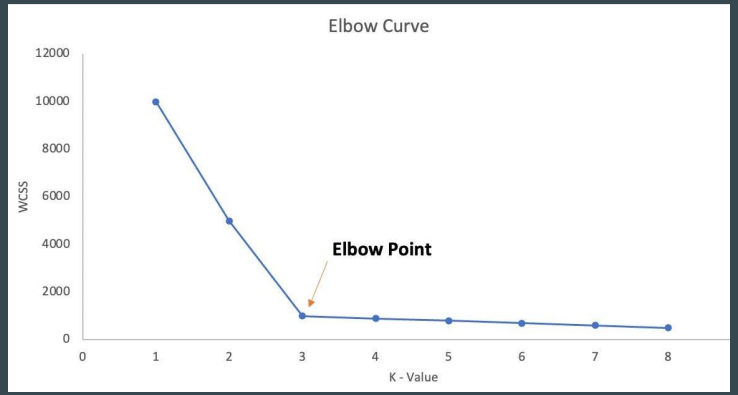
4. Repetimos a partir del paso 2 hasta que los clusters no cambien

Resultados

- Cantidad de clusters = [1, ..., 10]
- Para cada una de estas pruebas, medimos la variación:

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,j \in C_k} \sum_{L=1}^{p} (x_{iL} - x_{jL})^2$$

- Método del codo: para encontrar el k óptimo, se da cuando no hay una diferencia significativa en la variación



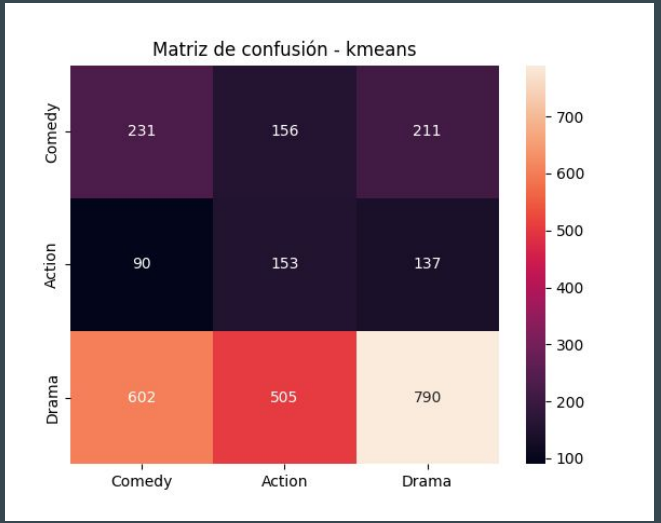
Resultados

- Vamos a ver cómo cambia entre 3 y 4 clusters (en uno de los casos con todas las variables)
- En todos los casos vamos a intentar predecir los géneros de la misma manera que con Kohonen
  - Para predecir el género, cada cluster va a quedar representado por el género que predomina dentro de los puntos que contiene
  - Nos vamos a quedar con sólo una de las particiones para mostrar los clusters
- Vamos a buscar el k óptimos con el método del codo
  - Corremos 5 iteraciones por cada k
  - Nos quedamos con la mínima variación

Resultados: Todas las variables

- K = 3

	Total	Comedy	Action	Drama
Cluster 1	845	43.9%	17.99%	38.11%
Cluster 2	984	29.07%	37.8%	33.13%
Cluster 3	471	19.53%	24.84%	55.63%

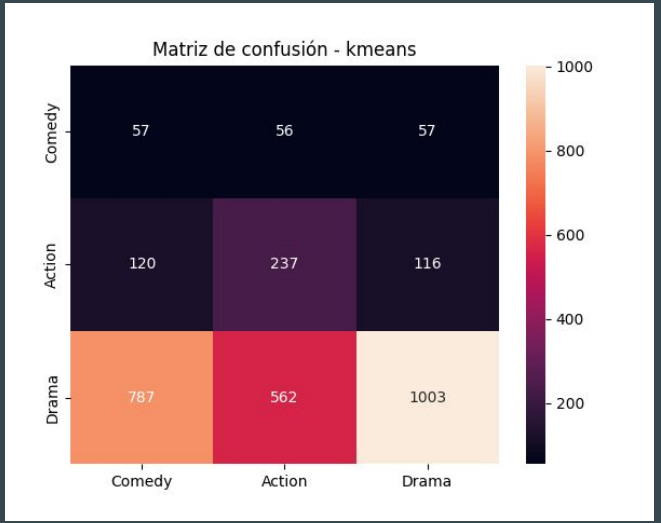


Precisión: 0.41

Resultados: budget, revenue, vote_count

- K = 3

	Total	Comedy	Action	Drama
Cluster 1	169	42.6%	23.67%	33.73%
Cluster 2	658	29.48%	40.43%	30.09%
Cluster 3	970	32.89%	22.99%	44.12%

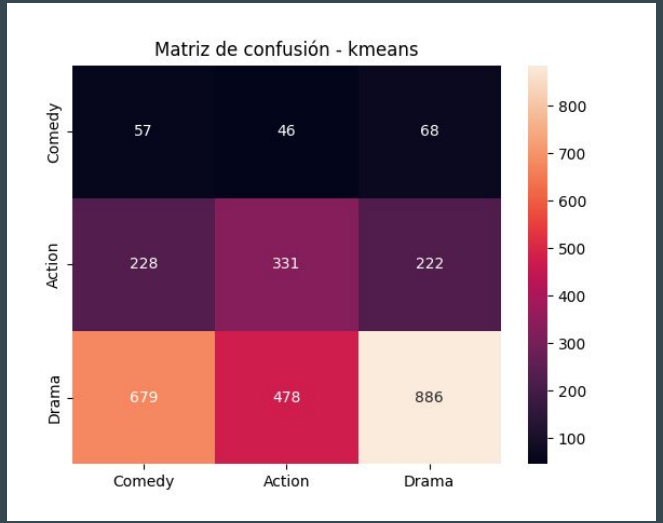


Precisión: 0.43

Resultados: budget, revenue, vote_count

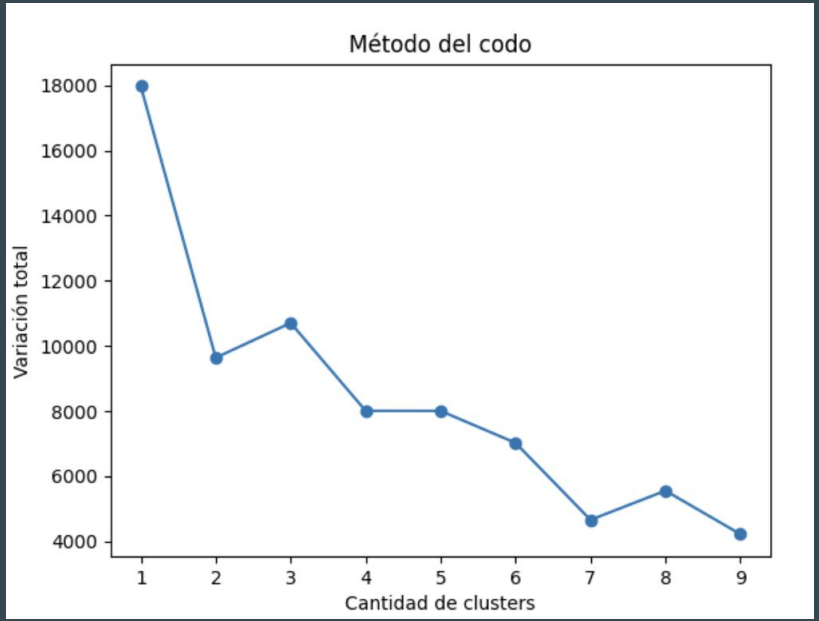
- K = 4

	Total	Comedy	Action	Drama
Cluster 1	349	40.4%	25.5%	34.1%
Cluster 2	687	32%	19.8%	48.2%
Cluster 3	320	27.81%	25%	47.19%
Cluster 4	1040	31.15%	36.54%	32.31%



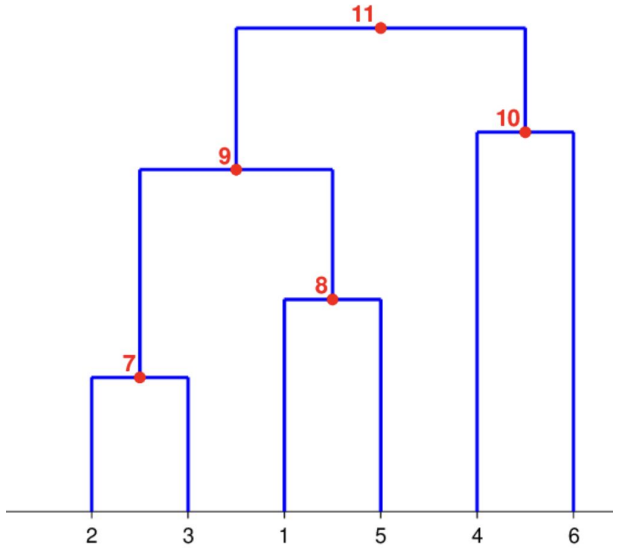
Precisión: 0.44

Método del codo: budget, revenue, vote_count



5 iteraciones por #clusters

Agrupamiento jerárquico



Implementación

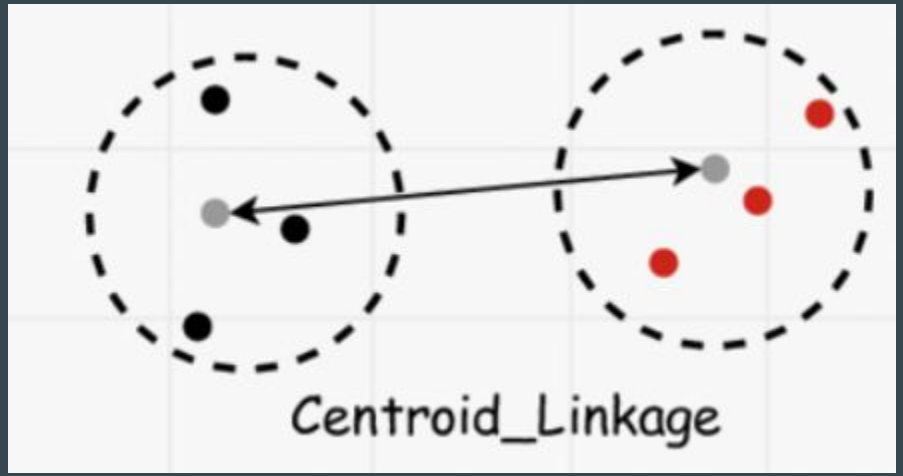
- Inicializamos cada una de las observaciones como un grupo individual
- Calculamos las distancias entre todos los grupos (en este caso son elementos individuales) en una matriz simétrica, donde las diagonales tienen valor infinito
- Mientras que la cantidad de grupos no sea la deseada:
  - Calculamos los grupos cuya distancia es la mínima
  - Al grupo de menor índice dentro de la matriz se le anexa el de mayor índice
  - El grupo de mayor índice es removido (y también se remueve su fila y su columna de la matriz de distancias)
  - Calculamos todas las distancias a este nuevo grupo (en realidad no es nuevo, es el anterior, sumándole los elementos del grupo removido)
  - En este momento tenemos un grupo menos

Implementación: distancia entre grupos

- En la primer iteración en cada grupo hay un solo elemento, calcular la distancia es trivial
- Entre un único elemento y un grupo
  - Centroides: se calcula el centroide del grupo y ese punto se toma como referencia para calcular la distancia
  - Máx: se toma la distancia máxima a todos los puntos del grupo
  - Mín: se toma la distancia mínima a todos los puntos del grupo
  - Avg: se toma la distancia promedio a todos los puntos del grupo
- Entre grupos con varios elementos es de forma análoga al punto anterior (ej: calcular 2 centroides, distancia máxima entre todos los puntos de los 2 grupos, etc)

Resultados

- Cantidad de grupos = [3, 4]
- No vamos a predecir el género ya que los grupos quedan muy desbalanceados
- Método de distancia: centroide



Resultados: todas las variables

Grupos: 3

	Total	Comedy	Action	Drama
Cluster 1	2860	32.27%	28.11%	39.62%
Cluster 2	11	0%	90.91%	9.09%
Cluster 3	4	0%	0%	100%

Resultados: budget, revenue, vote_count

Grupos: 3

	Total	Comedy	Action	Drama
Cluster 1	2935	32.74%	27.56%	39.69%
Cluster 2	59	5.08%	76.27%	18.64%
Cluster 3	1	0%	100%	0%

Resultados: budget, revenue, vote_count

Grupos: 4

	Total	Comedy	Action	Drama
Cluster 1	2935	32.74%	27.57%	39.69%
Cluster 2	51	5.88%	74.51%	19.61%
Cluster 3	8	0%	87.5%	12.5%
Cluster 4	1	0%	100%	0%

¡Muchas gracias!

Dey, Patrick
Lombardi, Matías
Vázquez, Ignacio