



Trabajo Práctico 4

Machine Learning

Salvador Castagnino	-	60590
Mauro Sambartolomeo	-	61279
Milagros Cornidez	-	61432

Analisis del Dataset

- Datos de películas con 14 variables, utilizamos todos menos imdb, title y overview, total de 5505 datos
- Cuantizar las fechas midiendo la distancia a 01/01/2000 en días
- Convertimos los géneros a enteros para poder comparar igualdad utilizando la misma métrica sobre todas las variables
- Para los valores na completamos con: la media si el campo es numérico, "" si el campo es string y la mediana para la cantidad de días
- Normalizamos con $(x/\text{mean})/\text{std}$ sobre cada campo
- Para medir distancias utilizamos la norma euclídea

Restricción de género

- Segundo análisis en dónde solo se utilizarán los datos de películas que sean de género "Action", "Comedy" y "Drama"
- Se busca reducir las opciones de agrupamiento para intentar predecir de qué género es un dato
- Reduce la cantidad de datos a analizar a 3420





K means



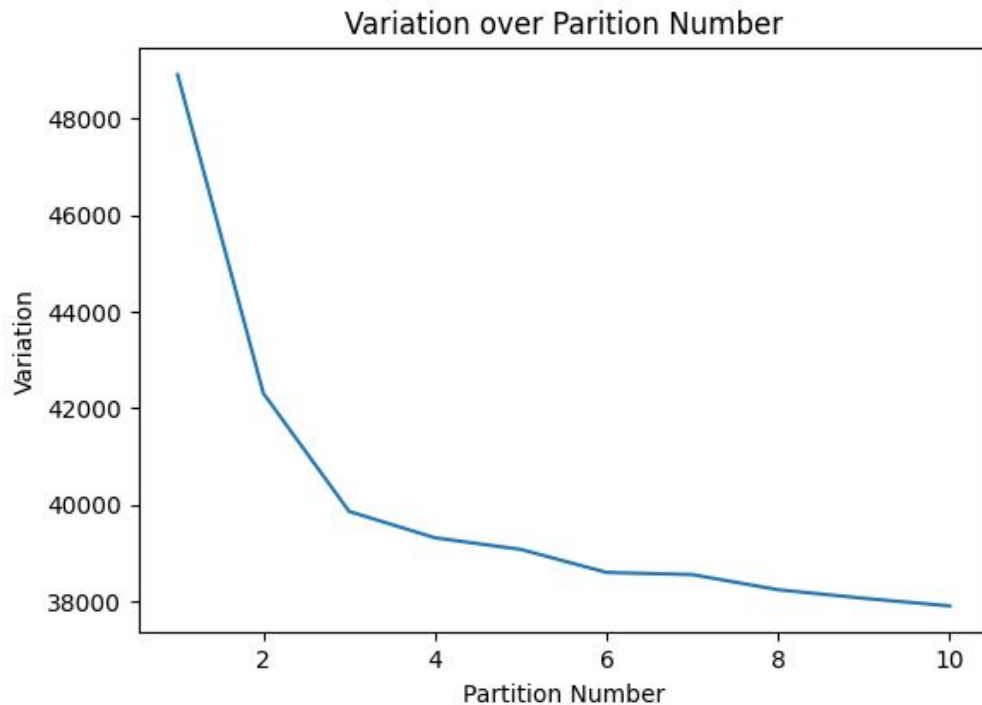
K means

- Busca el **agrupamiento óptimo** con k clusters
- Se busca minimizar la suma de las **varianzas** en los clusters
- Se inician los clusters como un **sample uniforme** sobre el dataset, cada inicialización puede dar resultados diferentes
- Utilizamos el método del codo para buscar el k óptimo



Elección de k - runs = 5

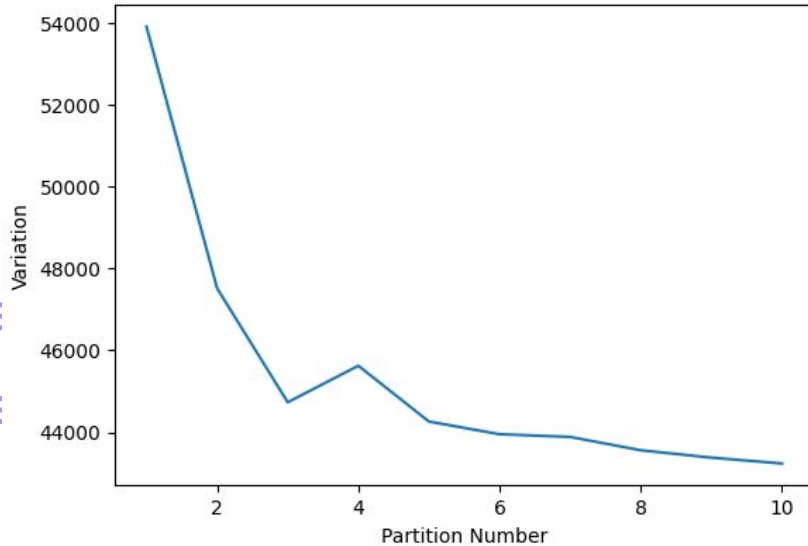
Sin date ni genre



Elección de k - runs = 5

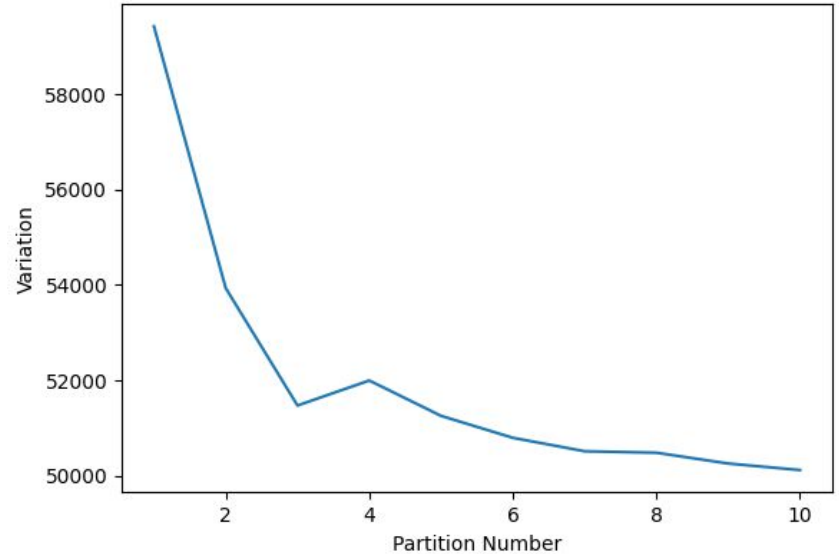
Con date sin genre

Variation over Partition Number



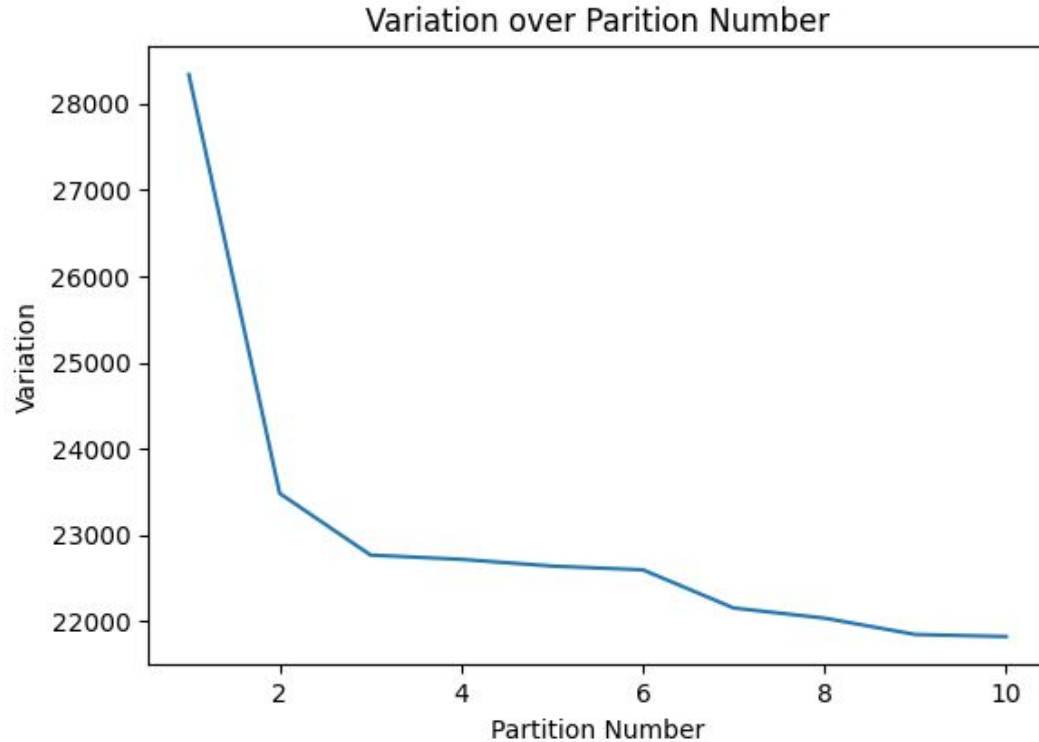
Con date y genre

Variation over Partition Number



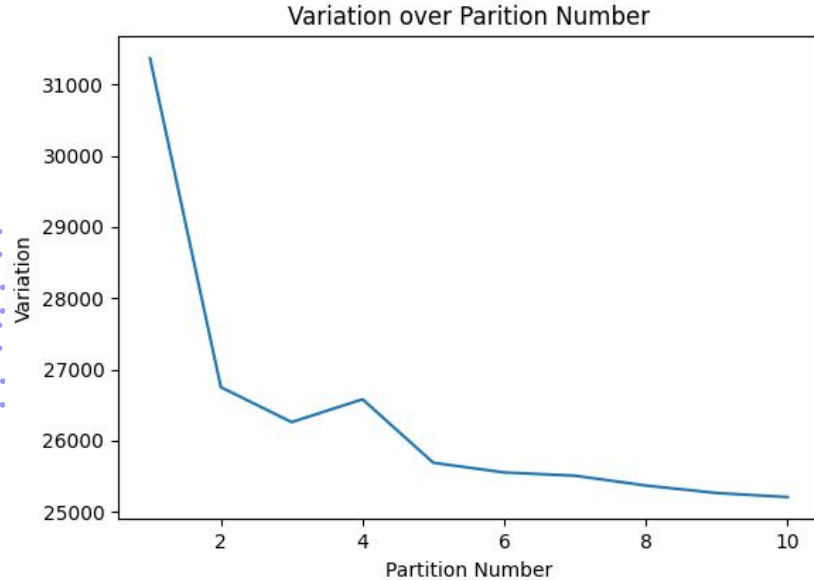
Elección de k - runs = 5 - subset

Sin date ni genre

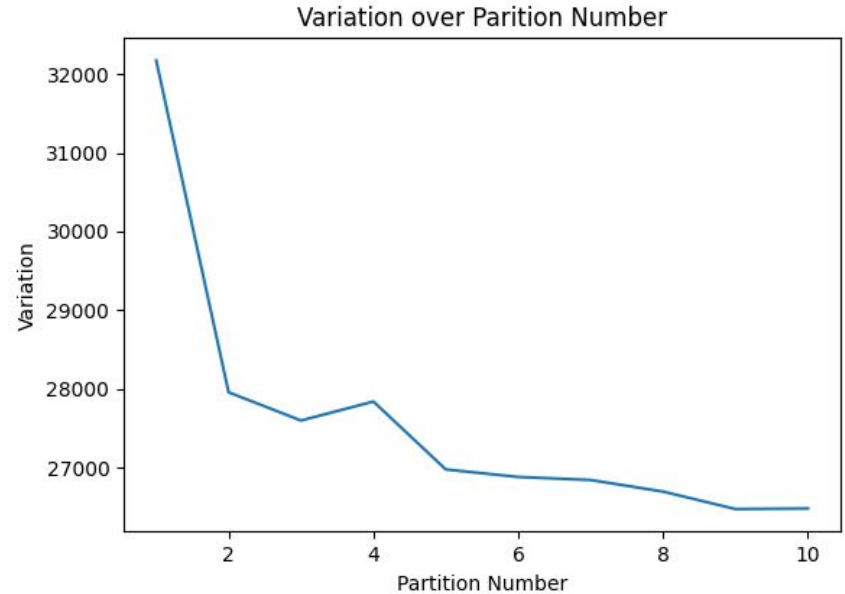


Elección de k - runs = 5 - subset

Con date sin genre



Con date y genre




Genres

- Se clusteriza omitiendo el campo genre para que la elección se dé puramente por los demas campos
- Se observa cual es la cantidad de puntos pertenecientes a cada género en cada cluster
- El análisis se realiza conociendo el total de datos por cada clase
 - Comedy: 1095
 - Drama: 1328
 - Action: 997
- La idea es asignar a cada cluster un género en base a la cantidad de elementos de cada género, luego el centroide que esté más cerca al punto decidirá el género predecido
- Nuestros resultados son sobre el subdataset pero se obtienen resultados similares tomando todo el dataset

Centroides $k = 3$

Mirando el primer ejemplo, el único cluster que da cierta certeza es el 1 ya que uno de los valores supera ampliamente a los demás




[11 206 878]
[32 505 791]
[100 364 533]
25590.170072768575

[112 22 961]
[303 62 963]
[223 144 630]
26395.434943669792

[178 59 858]
[415 168 745]
[180 242 575]
26966.774809911272

Centroides $k = 4$

Vemos algo análogo al anterior en la columna 1 del primer ejemplo, la columna 4 puede ser interesante también



[12 826 123 134]
[35 644 294 355]
[101 492 239 165]
26232.707867076984

[144 79 17 855]
[372 230 53 673]
[168 171 123 535]
26602.99838392047

[66 142 19 868]
[212 375 63 678]
[156 166 135 540]
26717.399949541024

Centroides $k = 5$

Empiezan a aparecer más clusters interesantes, en varios existe un género que supera por el doble a los otros géneros



[243 711 55 11 75]

[491 477 179 25 156]

[293 415 119 96 74]

25555.531

[72 11 128 713 171]

[123 26 309 541 329]

[136 88 131 407 235]

25977.732

[77 136 16 812 54]

[128 326 49 647 178]

[145 140 108 492 112]

26539.794

Centroides $k \geq 6$

[46 11 116 237 63 622]
[150 23 197 383 112 463]
[90 87 88 221 130 381]

$k = 6$

Cuantos más clusters mayor la P de que
alguno de una clasificación interesante.
Ahora, que tan relevante sera ese cluster?

[384 269 42 184 30 61 54 30 2 39]
[364 137 24 187 124 103 106 71 5 207]
[166 197 106 138 63 109 46 36 51 85]

$k = 10$

[13 55 40 164 9 5 198 72 135 34 2 206 4 32 126]
[50 92 187 114 15 56 78 100 78 76 4 248 31 95 104]
[25 102 67 105 69 20 152 64 102 21 40 65 9 57 99]

$k = 15$



K-means - Conclusiones

- El k óptimo varía dependiendo las columnas que se incluyan, siendo los posible valores $k = 3, 5$ y 7
- Al reducirse los datos el gráfico del codo mantiene la misma tendencia en términos generales
- No pareciera haber una clara correlación entre los 3 géneros estudiados y los clusters que generado por el método. Se ven clusters que se mantienen entre diferentes k .
- Al incrementar el valor de k los clusters se vuelven más granulares encontrando clusters que corresponden principalmente a un solo género





Agrupamiento Jerarquico



Agrupamiento Jerárquico Aglomerativo

- Algoritmo **no supervisado** para el agrupamiento de datos en clusters
- Se va agrupando en clusters de a **pasos**, en el paso n hay $\text{len}(\text{data}) - n$ clusters
- Los clusters a agrupar son aquellos dos con **menor distancia**
- La distancia la medimos con los métodos **min**, **max**, **avg** y **cent** entre clusters
- Utilizamos la matriz de distancias para encontrar la mínima y dendrograma para entender la estructura de los clusters

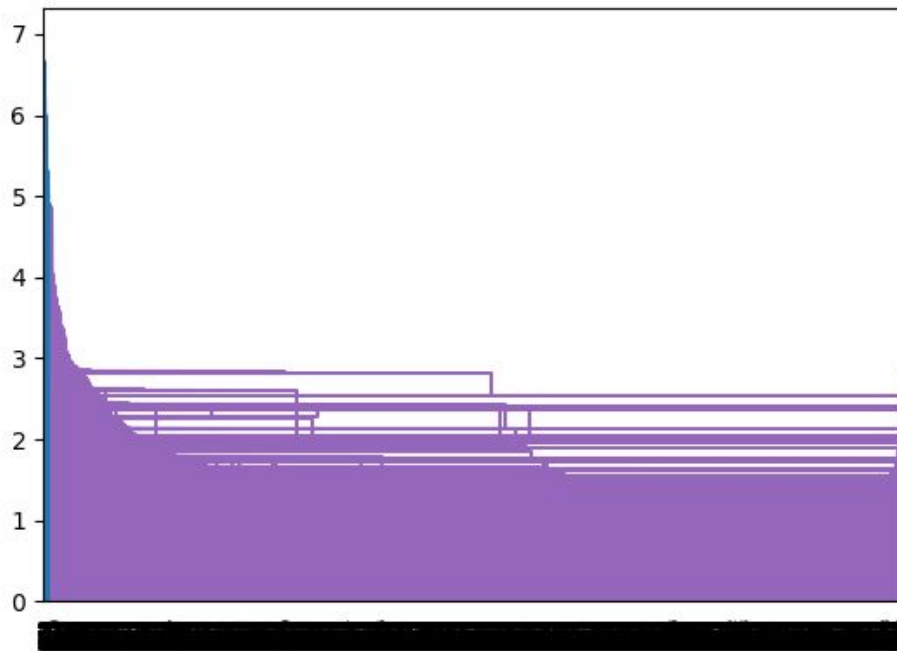


Generación de matrices y visualización

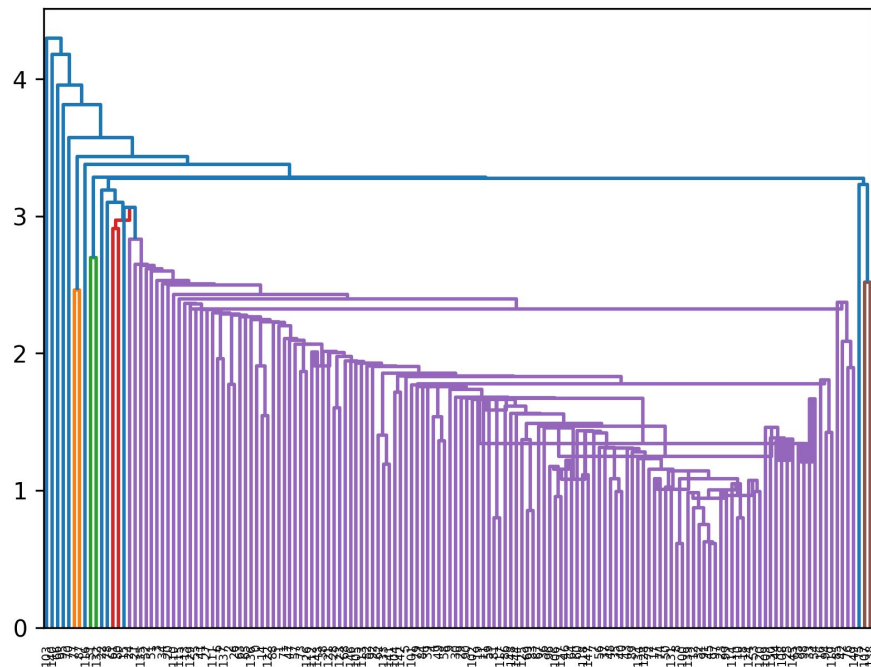
- A todos los puntos del dataset se les agrega un ID para su identificación
- En cada paso del algoritmo generamos una matriz con todas las distancias entre todos los clusters para luego poder obtener la distancia mínima.
- Se remueven del dataset los puntos con mínima distancia, y se agrega uno con un ID nuevo que sea la unión de los dos clusters removidos
- Se crea la matriz $4 \times (N-1)$ de Linkage, utilizada por la librería para hacer el dendograma. Cada fila contiene los IDs de los clusters removidos, la distancia, y la cantidad de puntos originales del dataset que contienen juntos.



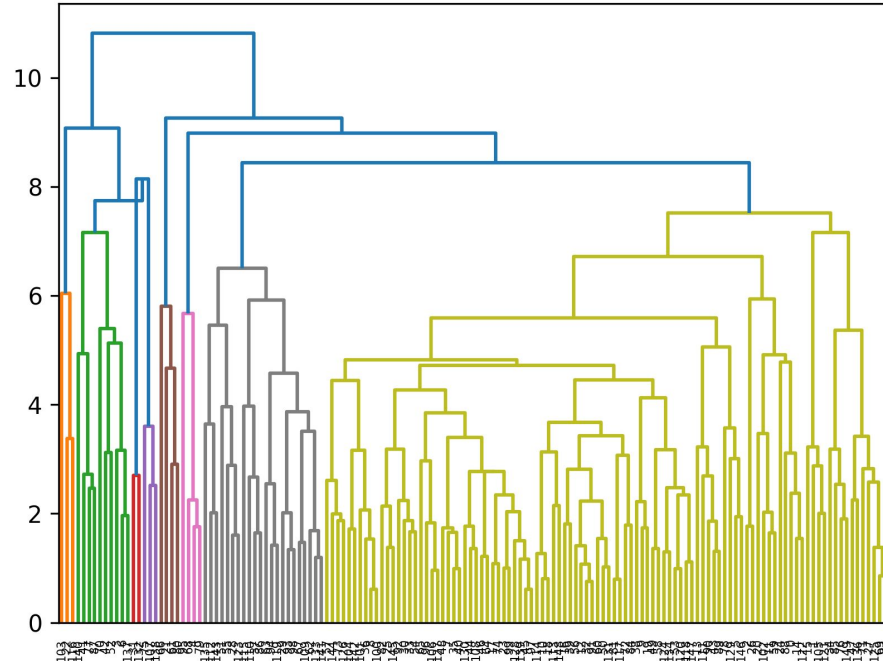
Too much data... data_len = 5505



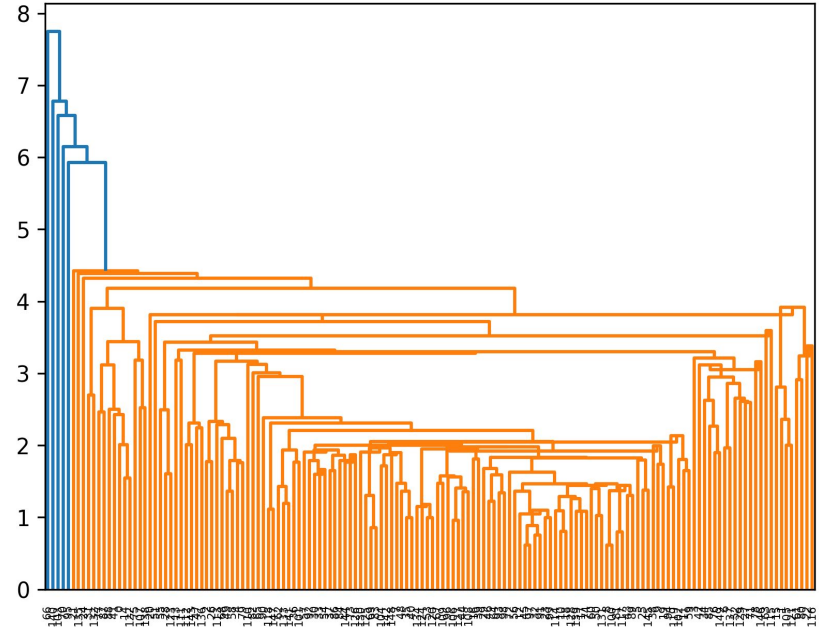
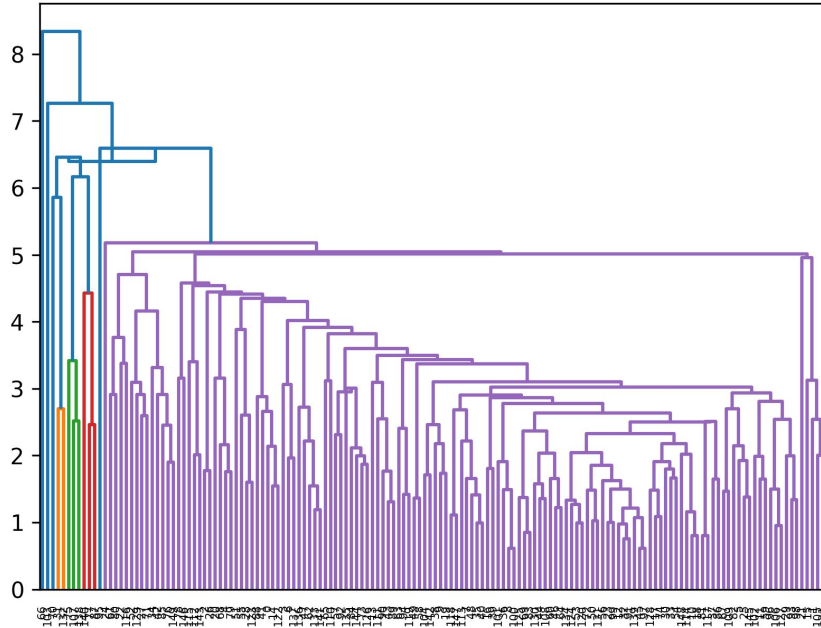
Dendrogram - Min - data_len = 150



Dendrogram - Max - data_len = 150




Dendrogram - Avg & Cent - data_len = 150



Genres - min - k = 30

[0. 0. 1.] [0. 0. 1.] [1. 0. 0.] [0. 0. 1.] [0. 1. 0.] [0. 1. 0.]
[0. 1. 0.] [0. 1. 0.] [0. 1. 0.] [0. 1. 0.] [0. 1. 0.] [0. 1. 0.]
[0. 1. 0.] [0. 1. 0.] [0. 1. 0.] [0. 1. 0.] [0. 1. 0.] [0. 1. 0.]
[0. 1. 0.] [0. 1. 0.] [0. 1. 0.] [0. 1. 0.] [0. 0. 2.] [2. 0. 0.]
[0. 1. 1.] [0. 3. 0.] [0. 0. 4.] [0. 0. 2.] [0. 3. 0.] [1092. 972. 1316.]



Se ve 1 cluster con la mayoría de los elementos (se mantiene la tendencia para $< k$)
Para las distancias avg y cent los resultados son bastante parecidos al min, menos concentrados pero no al punto de ser útiles

Genres - max - $k = 30$

[0. 0. 2.] [0. 2. 0.] [0. 3. 0.] [2. 1. 0.] [0. 0. 2.] [0. 4. 2.] [0. 2. 0.] [0. 2. 1.] [0. 3. 0.] [0. 2. 0.] [0. 3. 0.]
[1. 1. 3.] [0. 2. 1.] [5. 2. 5.] [0. 2. 3.] [0. 6. 0.] [0. 3. 1.] [0. 2. 1.] [1. 0. 5.] [0. 0. 4.] [0. 3. 3.] [2. 6. 9.]
[9. 12. 26.] [0. 1. 8.] [35. 89. 116.] [2. 9. 14.] [1. 21. 9.] [472. 283. 340.] [565. 527. 772.]
[0. 6. 1.]

Se ven 3 clusters con la mayoría de elementos, se mantiene la tendencia para clusters con $< k$



Jerárquico - Conclusiones

- Diferentes distancias generan estructuras diferentes dentro de la clusterización
- Max da las agrupaciones más balanceadas, se generan clusters independientes los cuales más tarde se agruparan
- Min genera las agrupaciones menos balanceadas, es un cluster grande que va sumando elementos uno por uno
- No es un buen método para hacer la clasificación por género por la dificultad de encontrar diferentes clusters que tengan una cantidad relevante de elementos, pocos clusters con muchos elementos

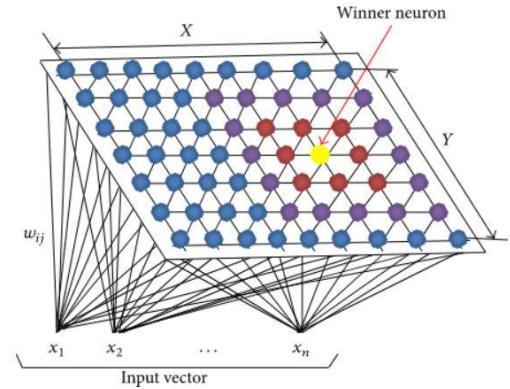




Kohonen

Redes de Kohonen

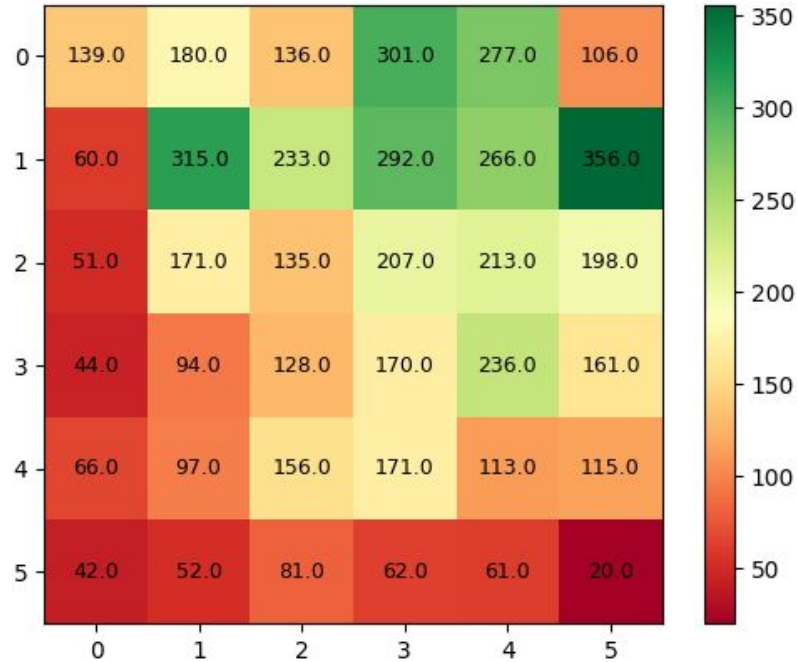
- Red de una sola capa bidimensional ($k \times k$ neuronas), cada una con su respectivo vector de pesos (representados como matriz). La entrada \mathbf{x} es n -dimensional.
- Las neuronas se conectan:
 - Con sí mismas
 - Con sus vecinas (dependiendo de un radio R)
- **Aprendizaje competitivo:** las neuronas compiten unas con otras de forma tal que sólo una de las neuronas de salida se active. Esta es denominada **neurona ganadora**.
- Dado un input \mathbf{x} , la neurona que tenga un vector de pesos \mathbf{w} más parecido a \mathbf{x} será la ganadora. Esto induce una **clasificación** en el conjunto, ya que inputs similares resultan en salidas similares.



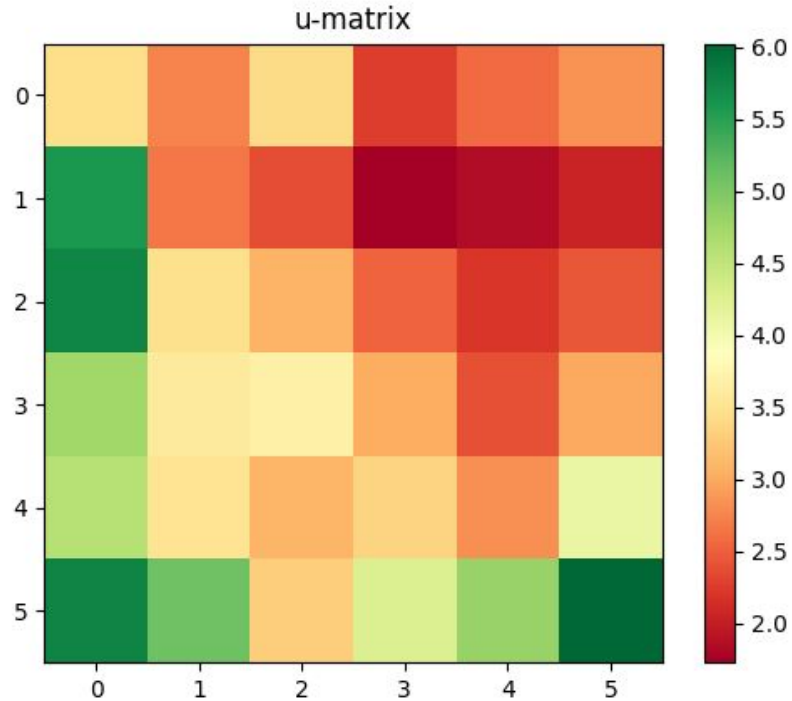
Parámetros utilizados

- $K = 6$
- Radio = k , decrecimiento lineal
- $LR = 1$, decrecimiento lineal
- Epochs = 500
- Pesos inicializados con valores aleatorios del conjunto de datos

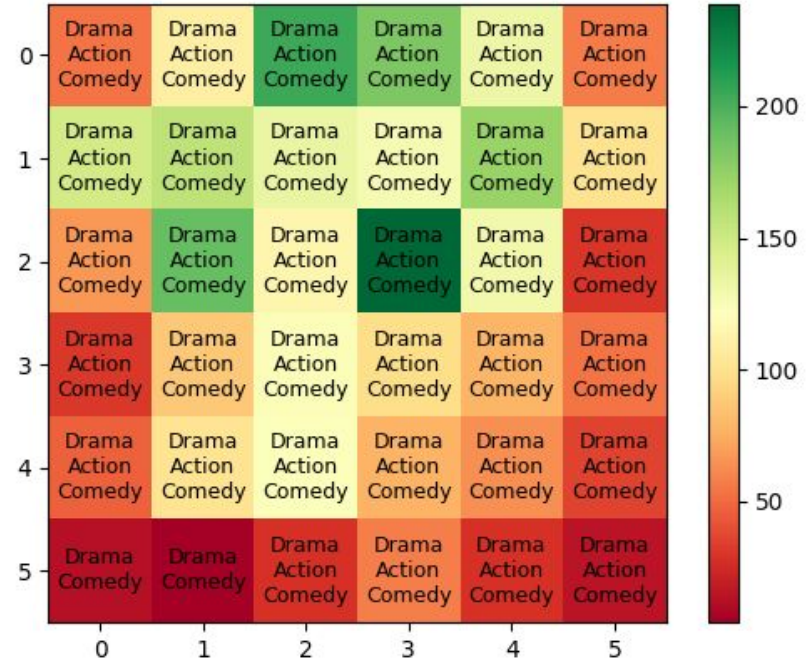
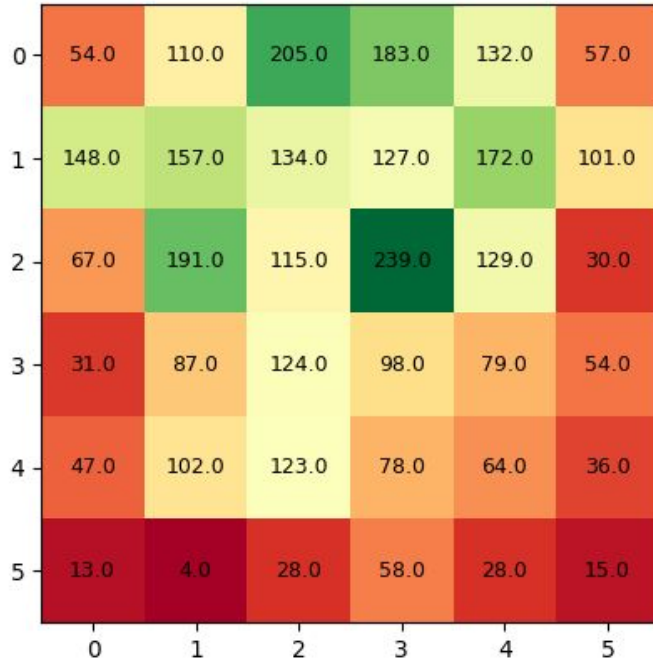
Kohonen - Hit Matrix - All Data



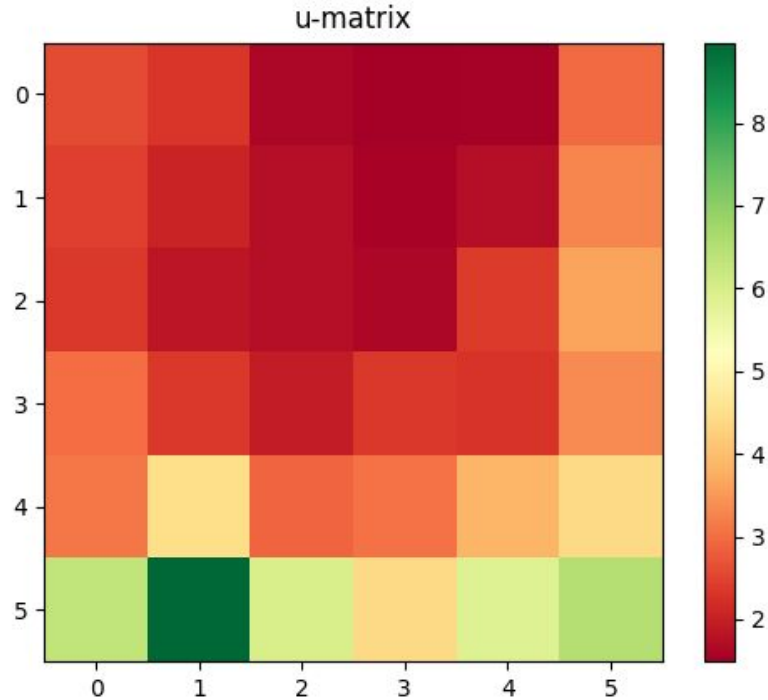
Kohonen - UMatrix - All data



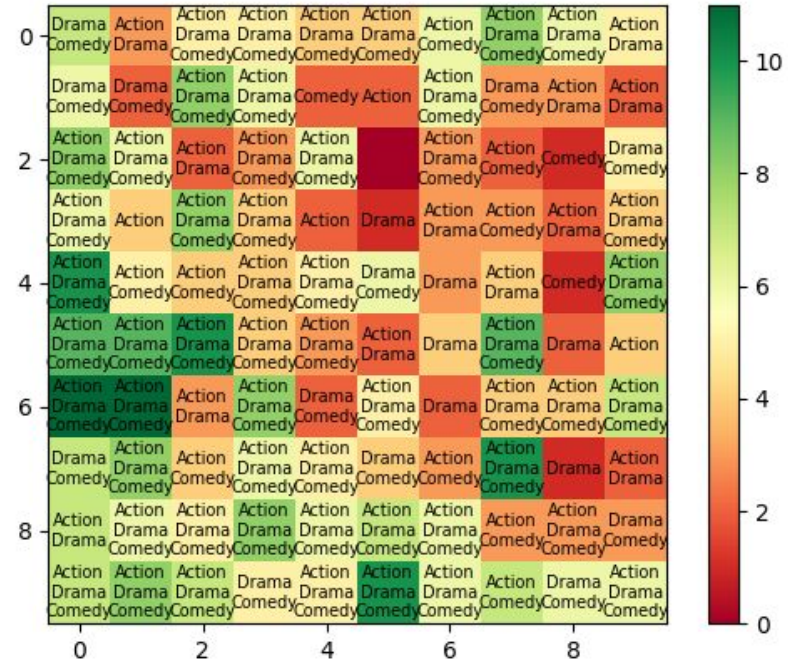
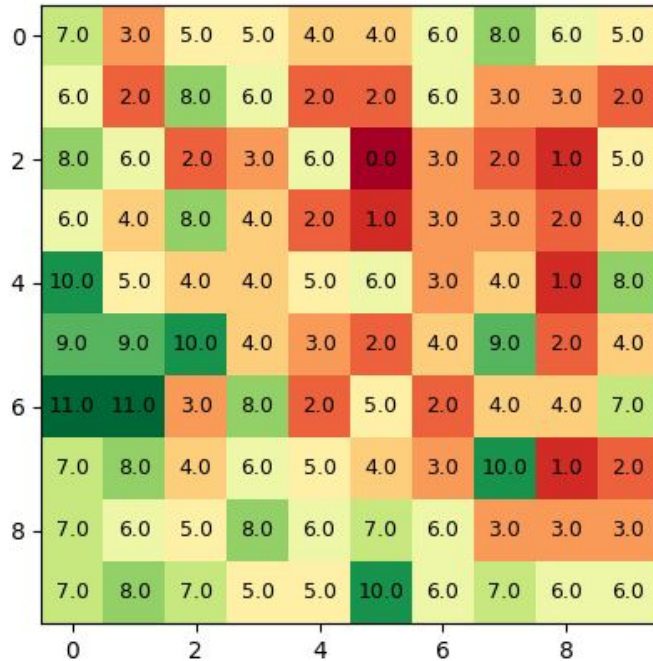
Kohonen - Hit Matrix - Restricción de Género



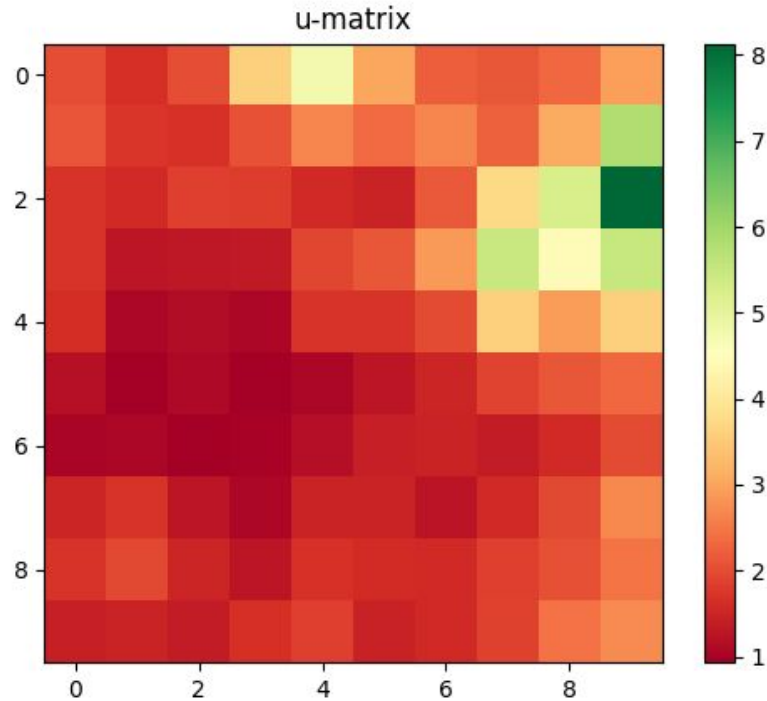
Kohonen - UMatrix - Restricción de Género



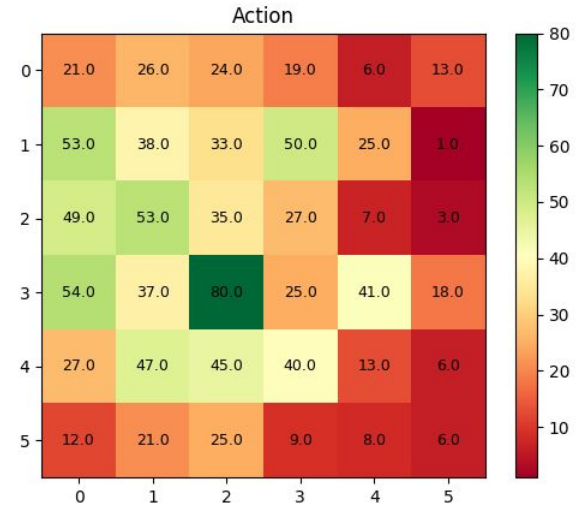
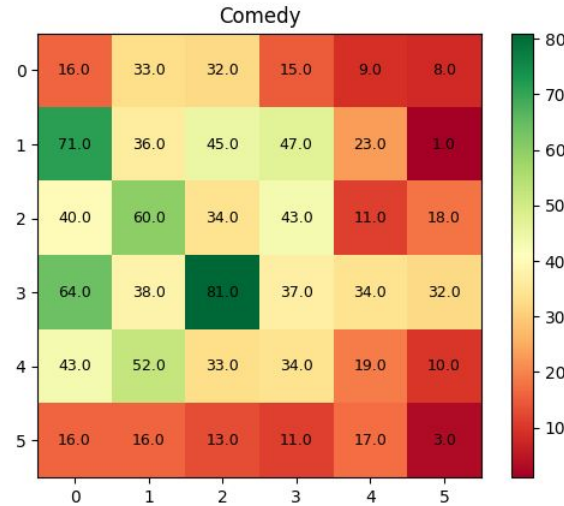
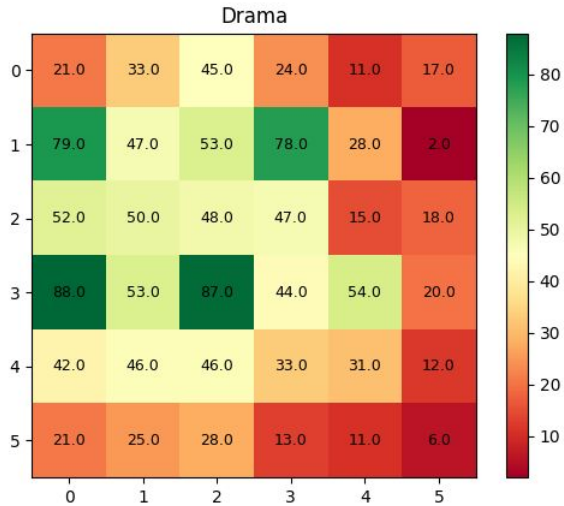
Kohonen - Hit Matrix - Restricción - 500 Datos



Kohonen - UMatrix - Restricción - 500 Datos



Kohonen - Hit Matrix - Hits por Género



Kohonen - Conclusiones

- Agrupación por similitud a neuronas en un plano
- U-matrix nos muestra la similitud entre los puntos que quedaron en las neuronas cercanas.
- Si intenta predecir el género viendo qué tipos de películas quedaron en cada casilla pero no dió resultado, pues la U-Matrix nos muestra una alta similitud entre neuronas cercanas las cuales contienen todos los géneros.
- Se intenta predecir el género viendo si hay zonas en las cuales se agrupan los mismos pero no dió resultado, porque los 3 géneros se agrupan en las mismas zonas.

