**Algoritmos no supervisados** Maria hernandez BBVA D&A

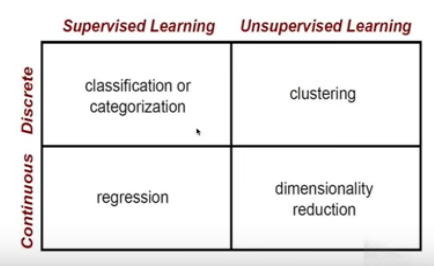
Hemos dado la clase con Google drive. Colaborative



Si tengo una columna con una escala enorme comparado con el resto de columnas



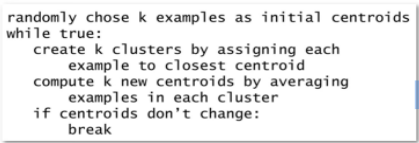
**Clustering**

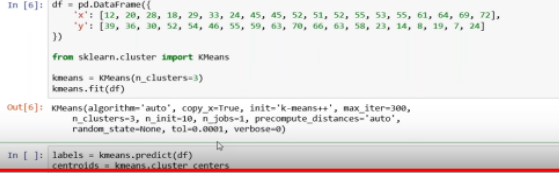


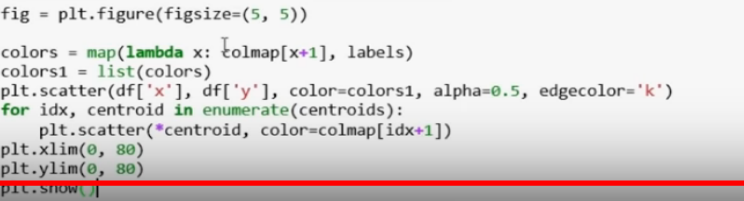
**kmeans**

Si queremos hacer un cluster lo primero es hacer un Kmeans.

El método del codo(Elbow) me ayuda a decidir el numero de clusters.- Cuantos más clusters la distancia el centroide va cayendo hasta cero que seria cuando los clusters son igual a las observaciones. Este punto no serviría para nada por lo que este método dice que cuando la función deja de caer de forma importante y forma ese codo, entonces entendemos que es el cluster ideal.



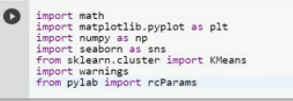


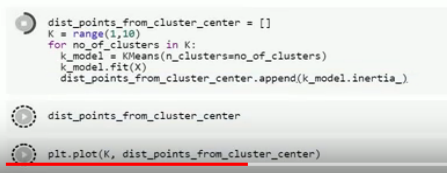


SSE para ver si mas o menos los clusters son correctos.

Si parte de los clientes pertenecen a varios cluster utilizamos Fuzzy o C-Means

**Elbow Method** para saber el numero ideal de clusters





Hay otra forma de hacerlo que es crear una linea entre k = 0 y k= al numero que decida y despues ver la distancia máxima desde la linea al arco que forma el elbow. Así te permite hacer el calculo automatico <https://www.youtube.com/watch?v=IEBsrUQ4eMc>

**Hierarchical clustering**

Con la grafica se ve mejor. Las distancias verticales a los nuevos grupos de clusters no harán elegir el grupo cluster que ya empieza a tener menos distancia vertical vs el anterior nivel.

**DBScan**

Funciona por densidades de numero de observaciones. Cuando cambia esa densidad cambia el cluster. Este es uno de los mejores algoritmos de clustering. El problema es que es mas lento. Primero haríamos un kmeans y después podemos ajustar con este.

**OPTICS**

Evolución de DBScan

**Reducción de dimensionamiento**

**PCA**

Es clave quedarme con las variables que generan mayor dispersión para que sea mas fácil agruparlas. P.e. Entre altura y peso , me quedaría con el peso porque hay mas variabilidad. En altura casi todo el mundo mide entre 1,60 y 1,90-

Un vector v es autovector de una matriz si cumple que al multiplicar la matriz por el vector me da la misma matriz pero multiplicada por un numero.

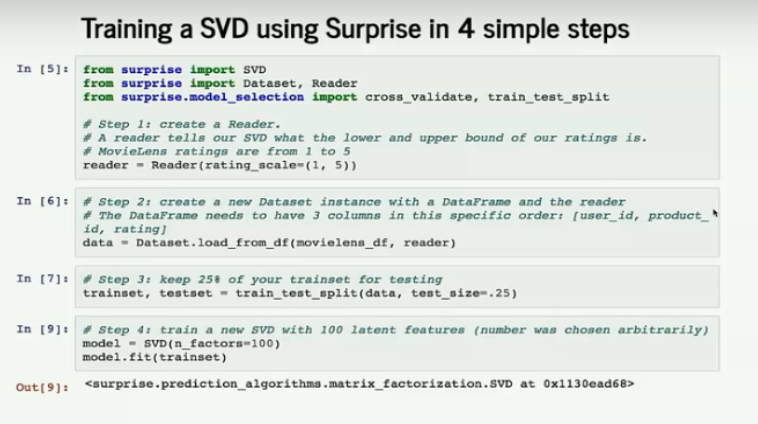
Es muy sensible a la escala de los datos(si no son similares en escala de cada variable).

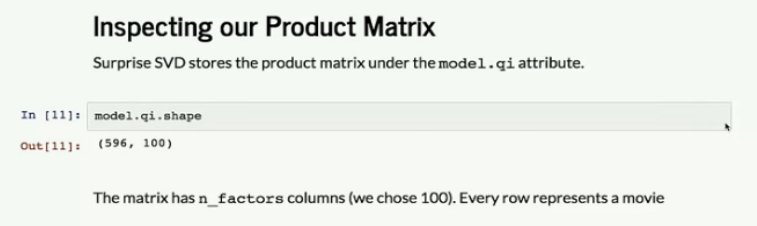
Una vez realizado PCA, se modeliza con esos datos rotados. Después hago el fit y modelizo. Los siguientes datos tendrán que pasar por pca antes de pasar el modelo.

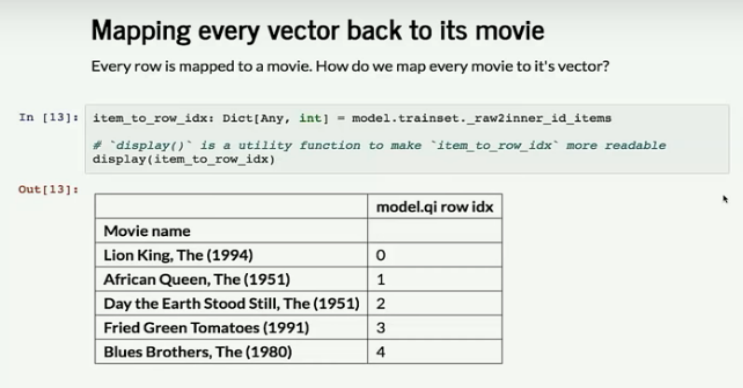
<https://www.youtube.com/watch?v=Lsue2gEM9D0>

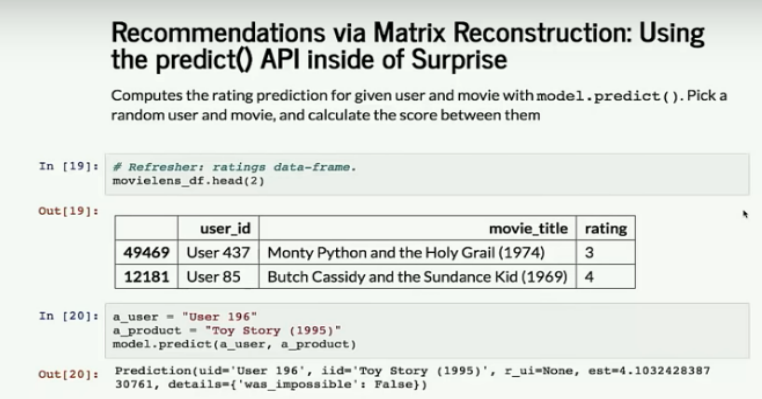
**Factorización de matrices**

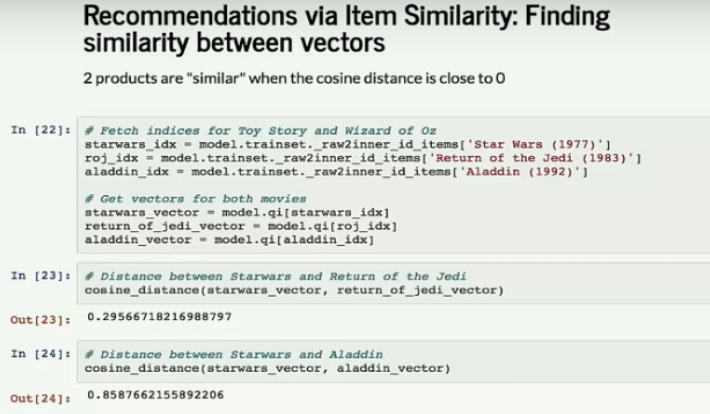
**SVD**- Muy utilizado en recomendación

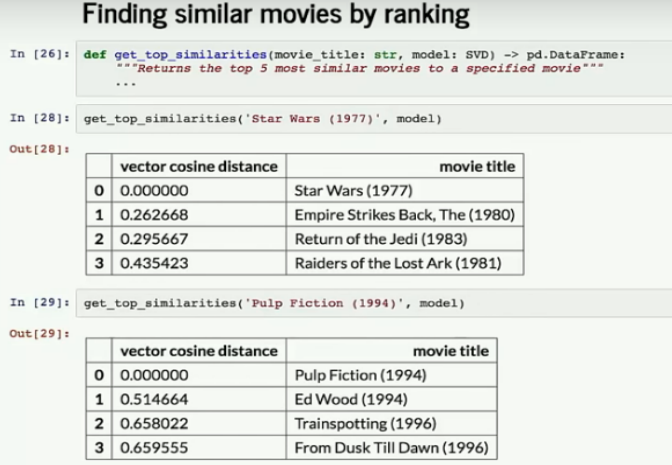












<https://www.youtube.com/watch?v=d7iIb_XVkZs>

NMF-el algoritmo descomponen la matriz original en dos. Se utiliza para generalizar características relevantes con factores latentes nuevos.

Autoencoder-