



UNIVERSIDAD DE CHILE

Minería de Datos

Welcome to the Machine Learning class

Valentin Barriere

Universidad de Chile – DCC

CC5205, Otoño 2024

Clustering

Aprendizaje supervisado y no supervisado

- Aprendizaje supervisado:
 - Objetivo: Aprender una función f que predice una variable Y a partir de un individuo \mathbf{X} .
 - Datos: Conjunto de entrenamiento (\mathbf{X}_i, Y_i)
- Aprendizaje no supervisado:
 - Objetivo: Descubrir una estructura dentro de un conjunto de individuos (\mathbf{X}_i).
 - Datos: Conjunto de entrenamiento (\mathbf{X}_i)
- El primer caso es el más simple:



Richard Socher @RichardSocher

Rather than spending a month figuring out an unsupervised machine learning problem, just label some data for a week and train a classifier.

7:47 PM · Mar 10, 2017

Clustering

Outline : Clustering

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

Otros métodos

Use-case: BERTopic

Outline : Principio

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

Otros métodos

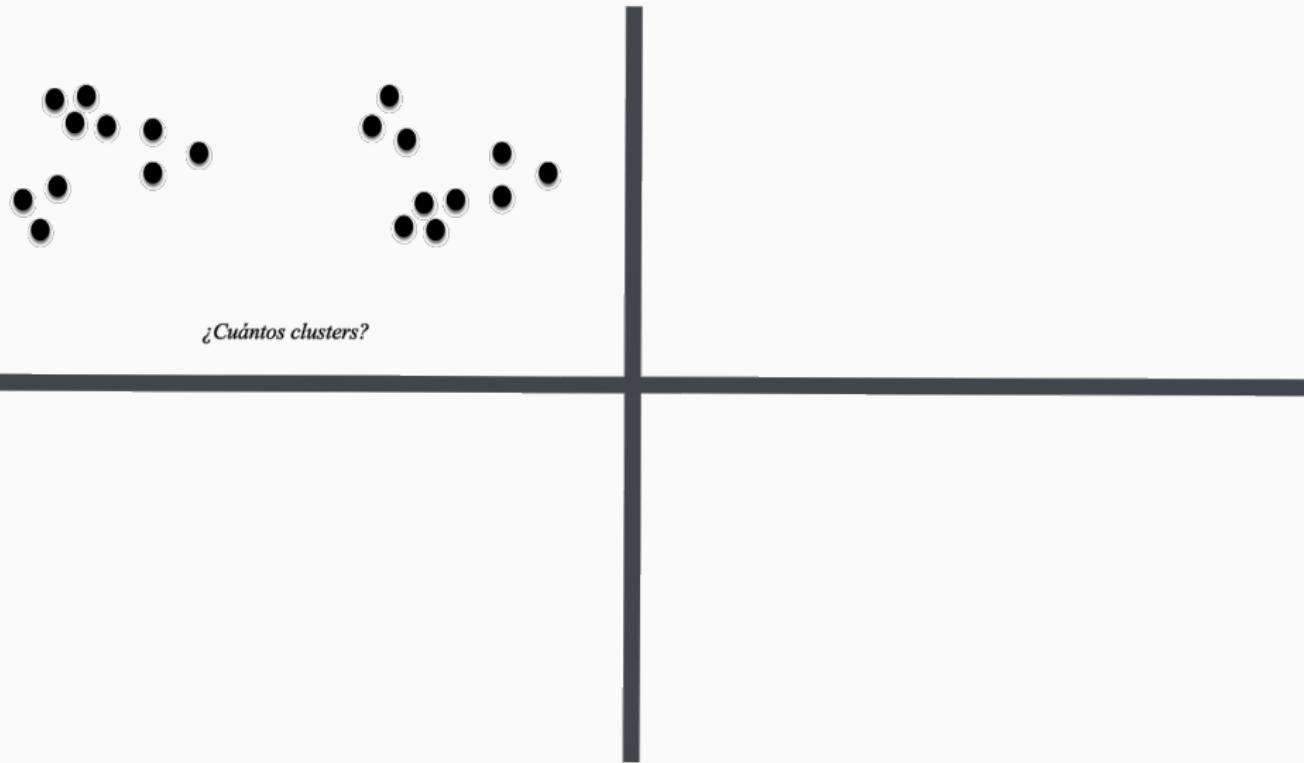
Use-case: BERTopic

Porque?

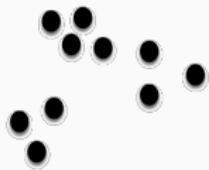
Porque puede ser interesante de crear grupos de documentos coherentes entre ellos:

- **Topic modeling**: to know the different topics of interest on documents or comments
- **Praise/critics discovery**: clustering user's comments with respect to the sentiment
- **Grouping** customers together
- **Efficiency**: train submodels on different clusters

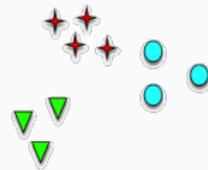
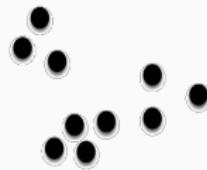
Porque no es tan simple? Ambigüedad



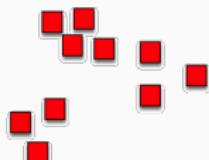
Porque no es tan simple? Ambigüedad



¿Cuántos clusters?



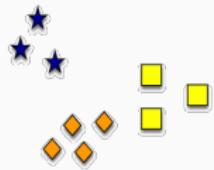
Seis Clusters



Dos Clusters



Cuatro Clusters



Como? Minimizar y Maximizar Distancias

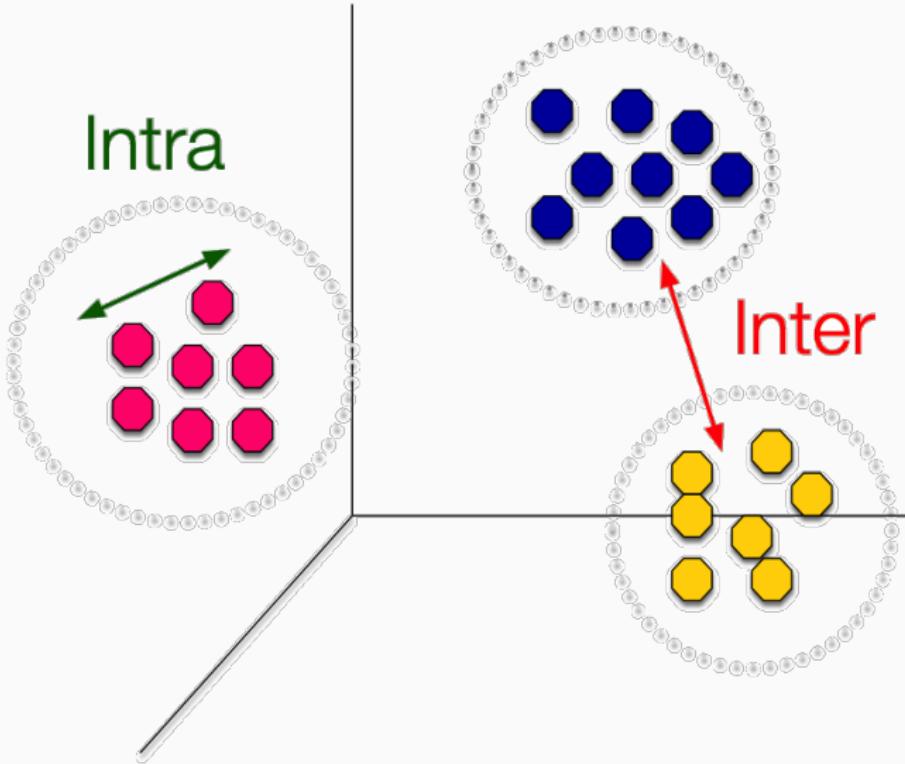


Figure 2: Reducir la distancia intra-clusters y aumentar la inter-clusters

Outline : K-means

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

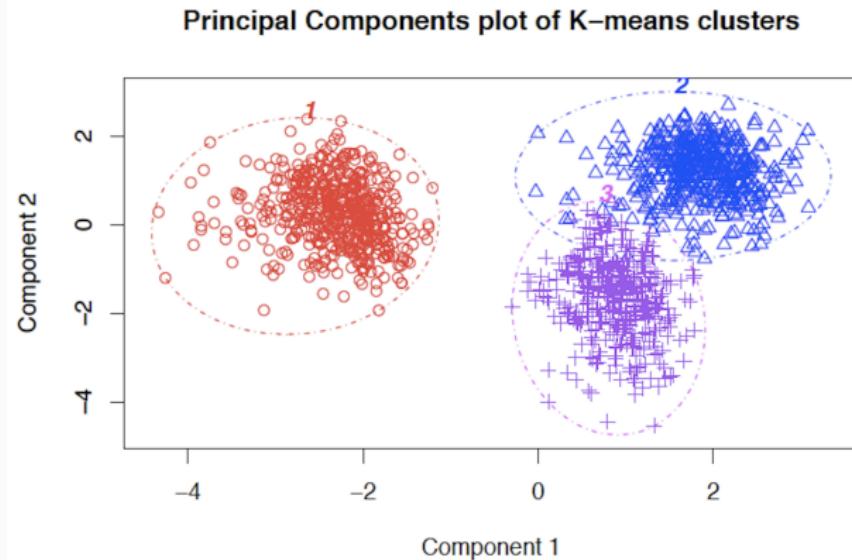
Otros métodos

Use-case: BERTopic

K-means

Principio

Separar los datos en K clusters utilizando los centroides de estos clusters.



K-means: teoría; minimizando la SSE

El algoritmo de K -Means divide los puntos en K grupos disjuntos $\{\mathcal{C}_1, \dots, \mathcal{C}_K\}$ minimizando la varianza intra-cluster. El criterio minimizado G se llama la **inercia** o SSE (Sum of Squared Errors):

$$G_X(\mathcal{C}_1, \dots, \mathcal{C}_K) = \sum_{k=1}^K \sum_{i \in \mathcal{C}_k} \|\mathbf{x}_i - \mu_k\|_2^2$$

donde los $\mu_k \in \mathbb{R}^d$ son los centroides de las K clases ($|\mathcal{C}_k|$ es el cardinal de cada clase):

$$\mu_k = \frac{1}{|\mathcal{C}_k|} \sum_{i \in \mathcal{C}_k} \mathbf{x}_i, \quad \forall k \in [K] ,$$

y donde la i^{th} observación tiene clase igual a la del centroide más cercano, es decir, $i \in \mathcal{C}_j$ si y solo si:

$$j = \arg \min_{k \in [K]} \|\mathbf{x}_i - \mu_k\|_2 = \arg \min_{k \in [K]} \|\mathbf{x}_i - \mu_k\|_2^2 .$$

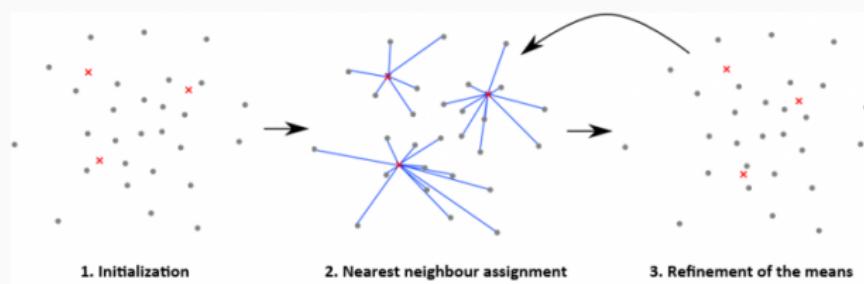
Nota: se resuelven los empates al azar si es necesario.

K-means: aprendizaje

El aprendizaje de los *clusters* se realiza alternando iterativamente los siguientes dos pasos:

- I) un paso de asignación donde, conociendo los $(\mu_k)_{k=1,\dots,K}$, se determinan las etiquetas de cada punto.
- II) un paso de actualización de centroides donde, conociendo las etiquetas, se determinan los centros de clase.

Se detiene el algoritmo cuando la inercia ya no disminuye significativamente.



K-means: inicialización

La inercia es un criterio no convexo, **por lo que la solución encontrada depende de la inicialización**, como suele ocurrir con tales problemas.

Solución

A menudo se lanza el algoritmo varias veces con diferentes inicializaciones, y luego se conserva la solución con la inercia más baja.

K-means: inicialización



(a) Initial points.

(b) Iteration 1.



(c) Iteration 2.

(d) Iteration 3.



Figure 3: La inicialización impacta la clusterización final

K-means: inicialización

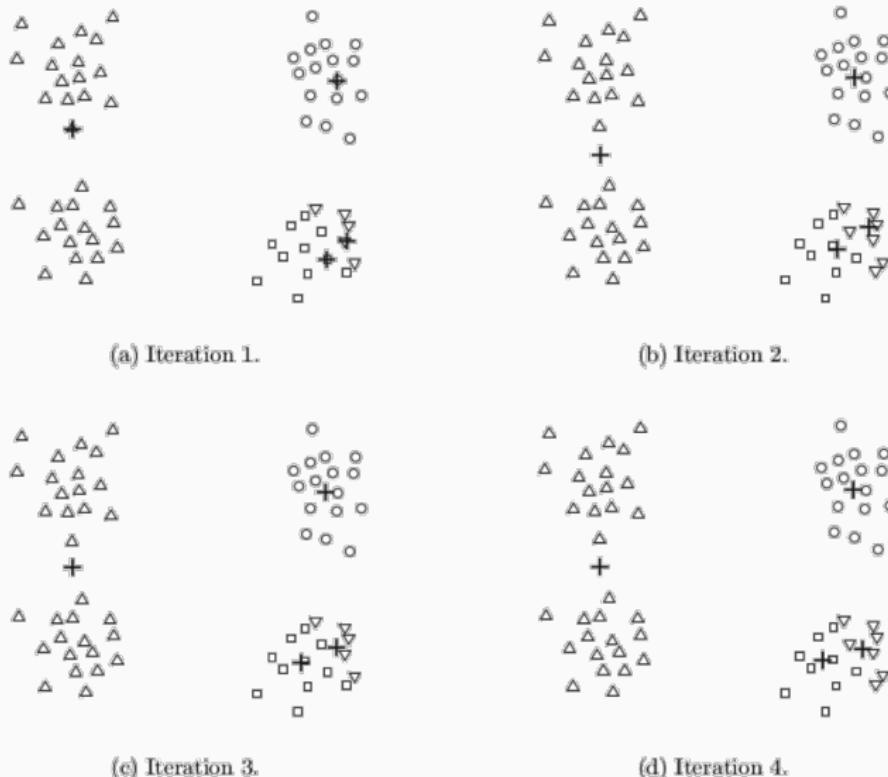
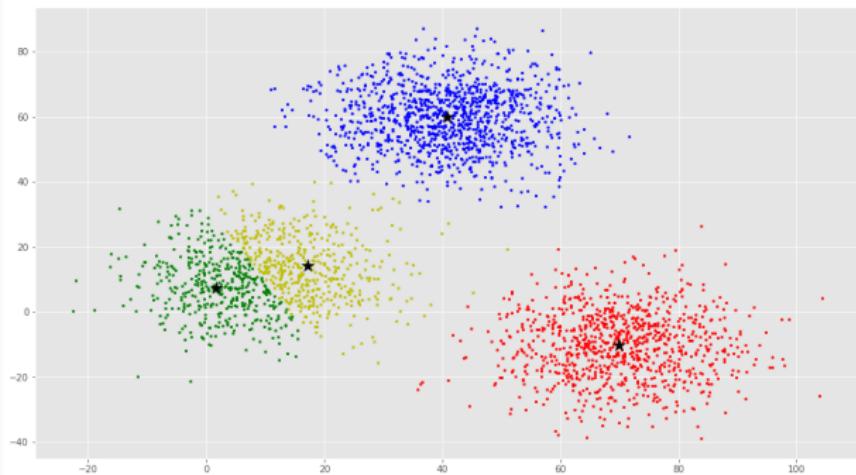


Figure 3: La inicialización impacta la clusterización final

K-means: elección de K

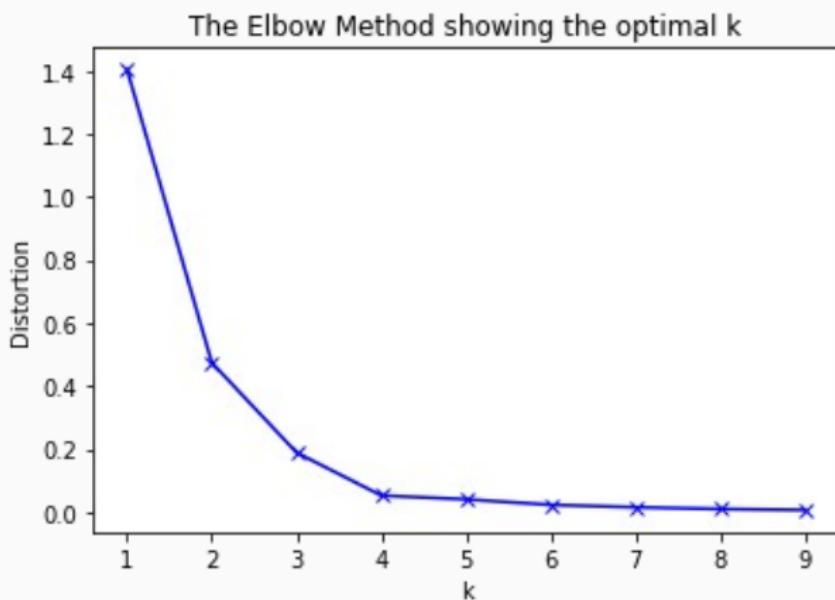
Los resultados obtenidos varían mucho según el K elegido inicialmente.



K-means: Método del codo

Método más común para encontrar el K óptimo

Ejecutar K-Means con diferentes valores de K y calcular la varianza de los diferentes clusters. Cuanto mayor es K , menor es la varianza. Se detiene cuando la varianza deja de disminuir fuertemente.



K-means: otro método

Todavía estamos interesados en un conjunto de datos X con n puntos. Sea $X_t, t \in [T]$, T muestras independientes de n puntos aleatorios seleccionados según una distribución uniforme en la caja delimitadora de X . A partir de las inercias G_X y G_{X_t} de los K -medias aplicados a X y X_t , nosotros definimos la diferencia:

$$\delta(k) = \text{Gap}(k) - (\text{Gap}(k+1) - \sigma(k+1))$$

donde $\text{Gap}(k) \in \mathbb{R}$ corresponde a la diferencia entre la esperanza del logaritmo de las inercias de G_{X_t} y el logaritmo de la inercia G_X :

$$\text{Gap}(k) = \mathbb{E}[\log(G_{X_t})] - \log(G_X)$$

y donde $\sigma(k)$ se define como:

$$\sigma(k) = \sqrt{\frac{T+1}{T} \mathbb{E}[(\log(G_{X_t})) - \mathbb{E}[\log(G_{X_t})]]^2}$$

Outline : DBSCAN

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

Otros métodos

Use-case: BERTopic

DBSCAN: Ejemplo

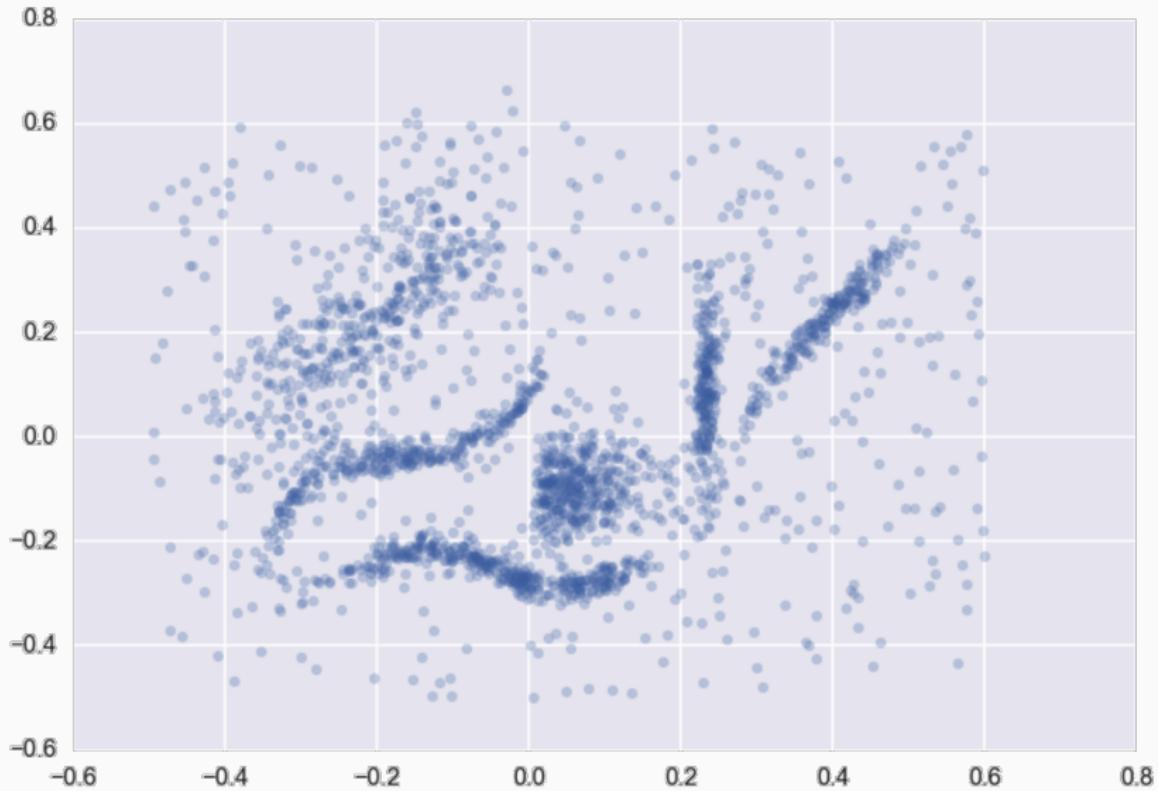


Figure 4: DBSCAN allows to get rid of the noise

DBSCAN: Ejemplo

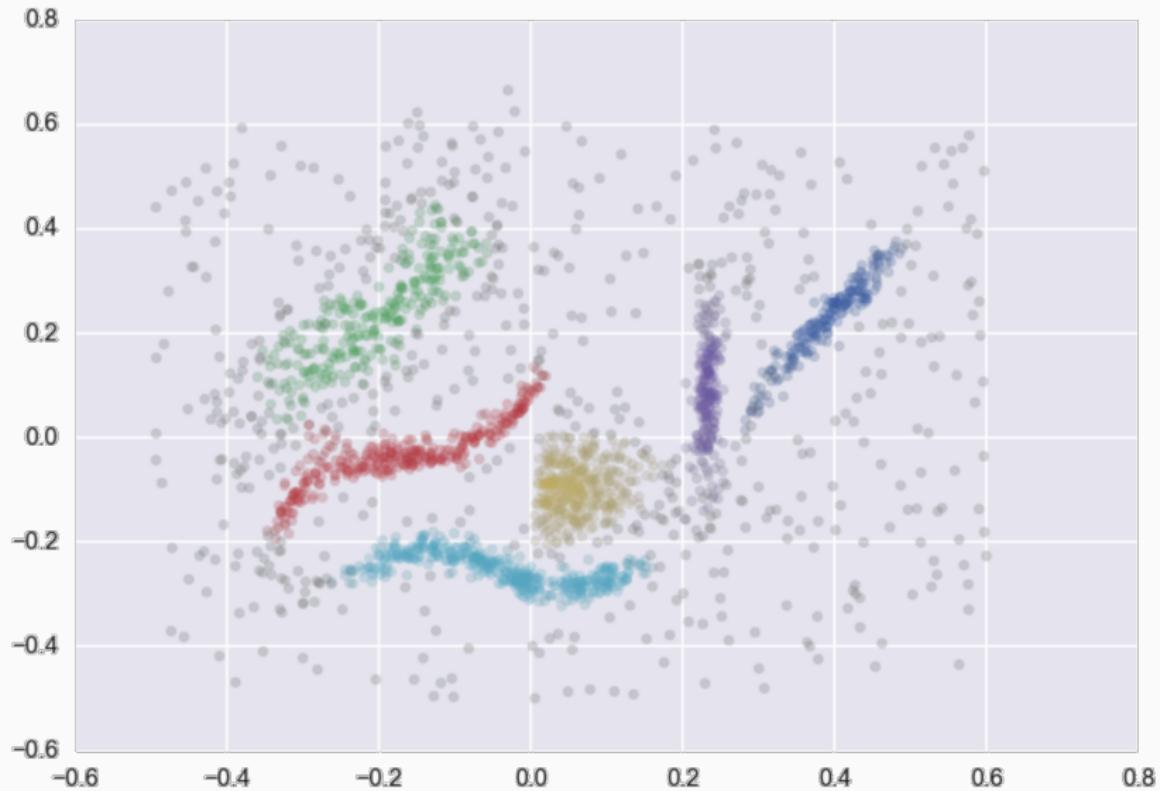


Figure 4: DBSCAN allows to get rid of the noise

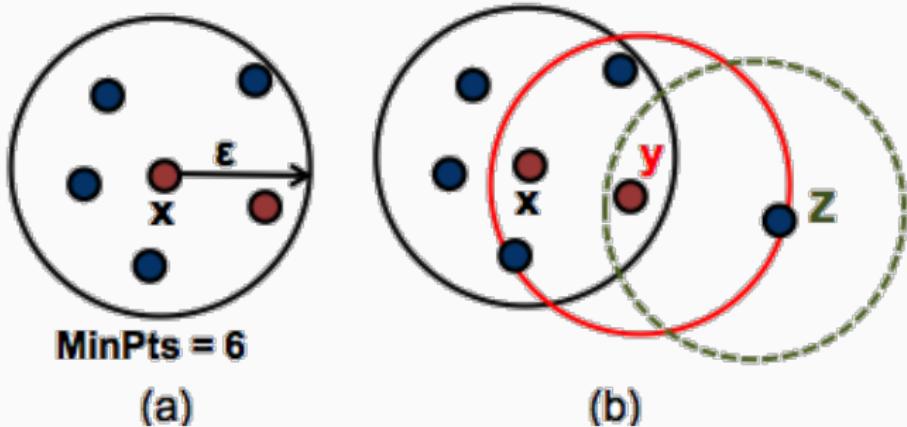
DBSCAN: Principios I

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise) forma clusters en regiones densas de puntos.
- Dos parámetros importantes:
 - eps: radio de vecindad.
 - min_samples: número mínimo de muestras requeridas dentro de esa vecindad para que un punto sea núcleo.

Clasifica en 3 tipos de puntos: core, border, noise

- Puntos núcleo/core: tienen al menos min_samples vecinos en un radio eps.
- Puntos alcanzables/border: pertenecen al mismo cluster si están conectados a un punto núcleo.
- Ruido/Noise: puntos que no cumplen ninguna de las condiciones anteriores.

DBSCAN: Principios II



- Cada punto tiene vecinos, en un circulo de rayo ϵ (ε)
- El punto **x** es un punto **core** porque tiene mas de MinPts vecinos
- El punto **y** es un punto **border** porque no tiene MinPts vecinos
- El punto **z** esta un punto **noise** porque no es vecino con un punto core
- Los puntos border se afectan al cluster vecino

Tambien se puede hacer *soft clustering* con DBSCAN.

DBSCAN: Ventajas y Desventajas

Ventajas

- Identifica **clusters de forma arbitraria**; no asume formas esféricas.
- Distingue **ruido** (outliers) del resto de datos.
- Normalmente requiere especificar menos parámetros que métodos jerárquicos.

DBSCAN: Ventajas y Desventajas

Ventajas

- Identifica **clusters de forma arbitraria**; no asume formas esféricas.
- Distingue **ruido** (outliers) del resto de datos.
- Normalmente requiere especificar menos parámetros que métodos jerárquicos.

Desventajas

- **Sensibilidad** a la elección de `eps` y `min_samples`.
- Si la densidad varía mucho dentro del dataset, un solo valor de `eps` puede no resultar apropiado para todos los clusters.

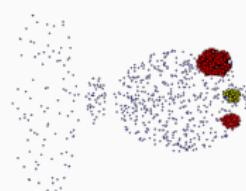


Figure 5: $\varepsilon = 9.92$

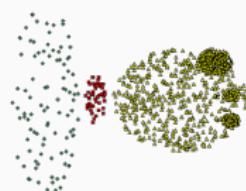
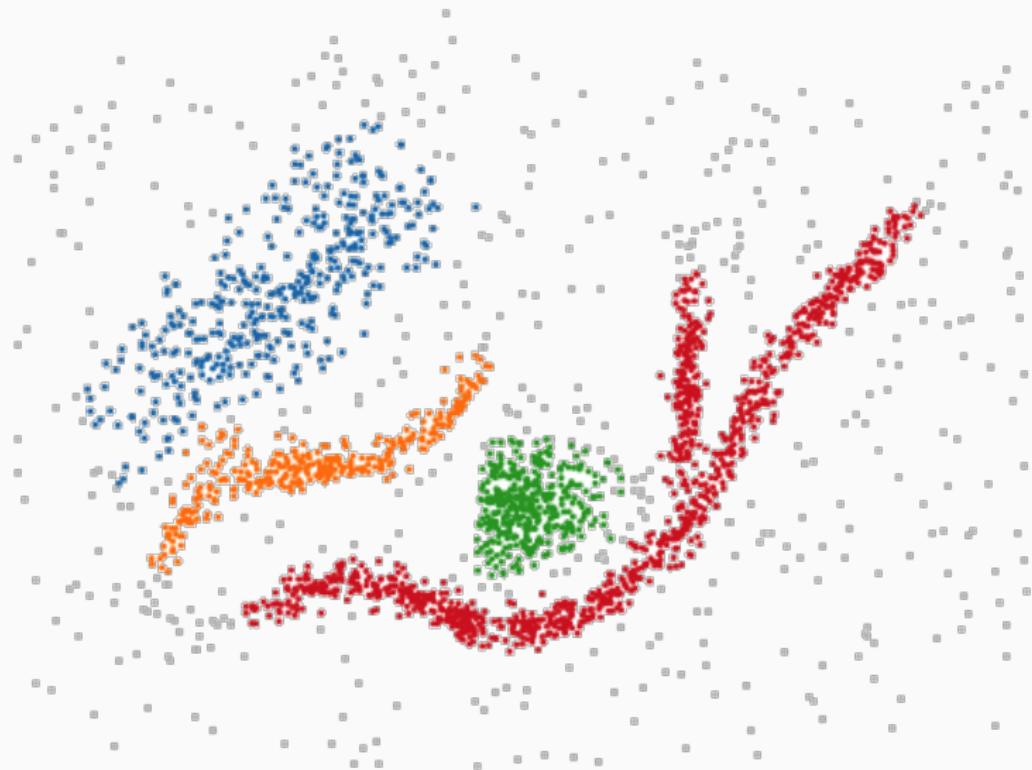
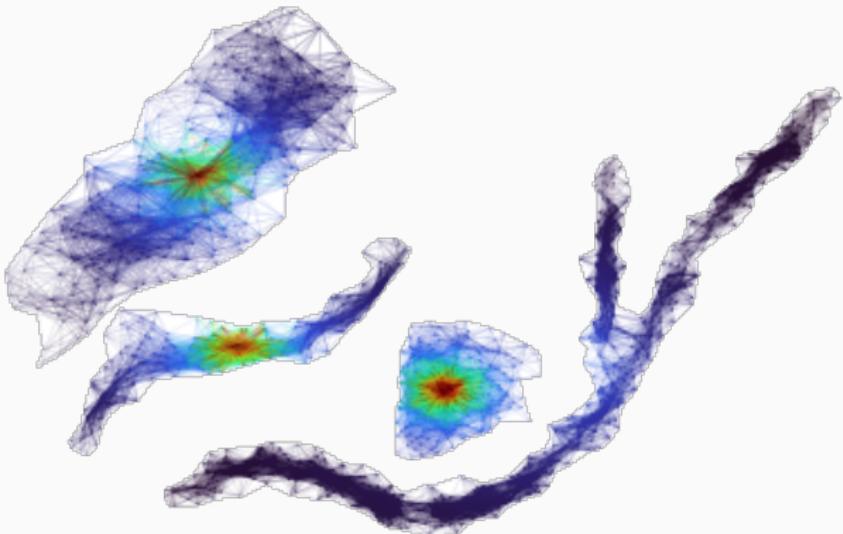


Figure 6: $\varepsilon = 9.75$

DBSCAN: Graph-like visualization



DBSCAN: Graph-like visualization



DBSCAN: Elección de minPts y ε

Rule of thumbs para cantidad de vecinos

Como regla general, se puede obtener un minPts mínimo a partir del número de dimensiones D del conjunto de datos, como $\text{minPts} \geq D + 1$.

ε : k-distance graph y “codo”

- Para seleccionar ε , se traza el gráfico de **k-distancia** (con $k = \text{minPts} - 1$), ordenando las distancias de mayor a menor.
- Un ε “bueno” suele aparecer en el “**codo**” de esta curva.
- ε muy pequeño: muchos puntos quedan como “ruido” sin asignar.
- ε muy grande: se fusionan muchos clusters, produciendo uno muy grande.
- **Regla práctica:** elegir un valor ε tal que solo una fracción pequeña de puntos esté dentro de esa distancia.

Se puede tambien usar el algoritmo **OPTICS** para elegir ε .

DBSCAN: Elbow Method

- Calcular k -dist para todos los puntos con un valor fijo de k .
- Ordenar los valores de k -dist de manera creciente.
- Graficar k -dist vs. la cantidad de puntos.
- Identificar el "salto" en la gráfica de k -dist (10 en la figura)
- minPts puede ser el valor de k asociado.

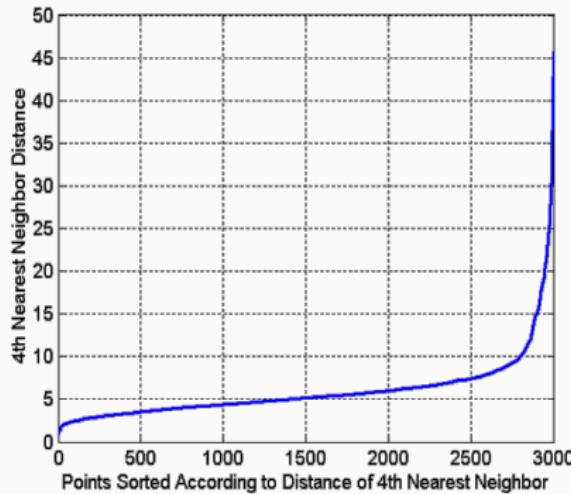


Figure 7: Un grafico de k -distance

DBSCAN: Desventajas

- No funciona con densidades variables → se confunden con el ruido
- Alta dimensionalidad: es difícil de estimar una densidad en alta dimensión

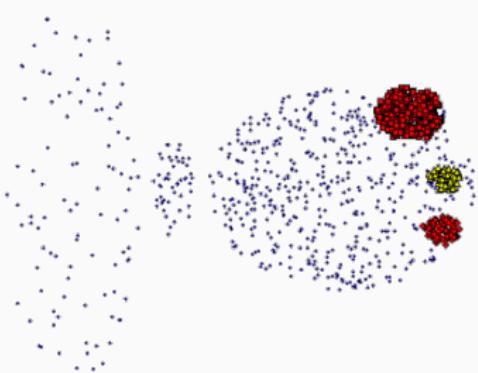


Figure 8: $\varepsilon = 9.92$

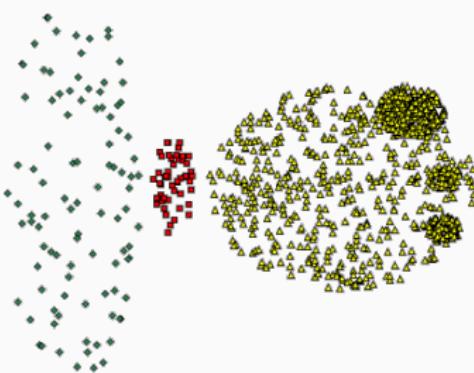
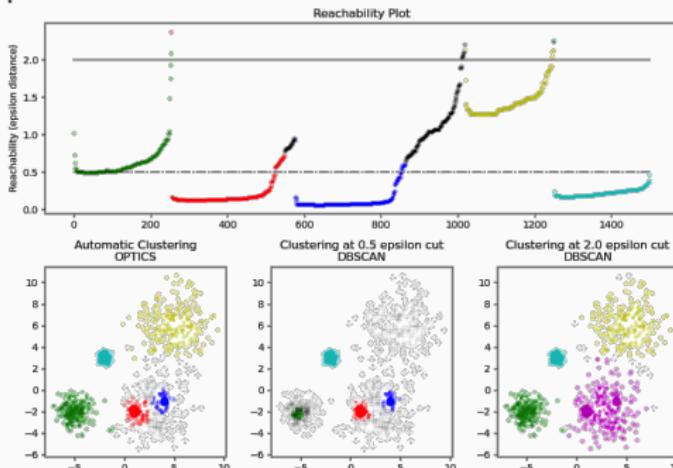


Figure 9: $\varepsilon = 9.75$

OPTICS: Una generalización de DBSCAN

- Permite **densidades variables** ajustando un rango de ε , en lugar de un valor fijo.
- Construye un **grafo de alcanzabilidad**, asignando a cada muestra una `reachability_distance` y un `ordering_`.
- El **reachability plot** ayuda a identificar clusters “cortando” la gráfica y detectando transiciones bruscas (parámetro ξ).
- Ofrece una **estructura jerárquica** de clusters y puede emular DBSCAN para distintos ε sin recalcular todo.



Hierarchical Clustering

Outline : Hierarchical Clustering

Clustering	Clustering Jerárquico Aglomerativo
Principio	Hierarchical DBSCAN
K-means	Validation Metrics
DBSCAN	Ejemplos
Hierarchical Clustering	Otros métodos
Bisecting K-means	Use-case: BERTopic

Outline : Bisecting K-means

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

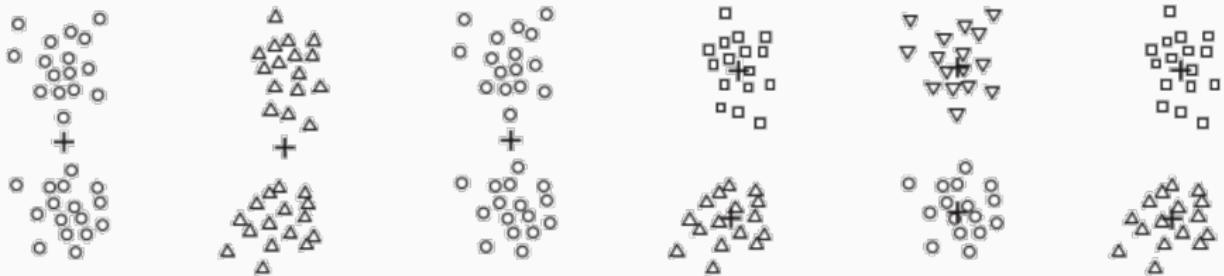
Ejemplos

Otros métodos

Use-case: BERTopic

Bisecting K-means: Idea general

- **Bisecting K-means** es una variante jerárquica de K-means:
 - Empieza tratando todos los datos como un único cluster.
 - Aplica K-means (con $K = 2$) para “dividir” ese cluster en dos subgrupos.
 - Elige el subgrupo con mayor inercia (SSE) y lo vuelve a dividir.
 - Repite hasta obtener el número deseado de clusters.
- Combina **ventajas** de K-means con un enfoque jerárquico, y puede dar mejores resultados que K-means simple cuando K es grande.
- Tiende a crear agrupaciones que tienen una **estructura a gran escala más regular**.



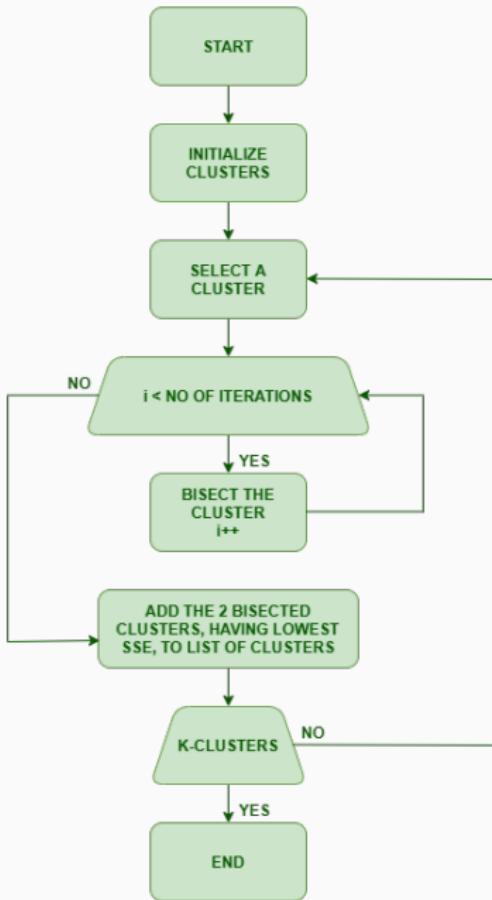
(a) Iteration 1.

(b) Iteration 2.

(c) Iteration 3.

Bisecting K-means: Algoritmo

- Extensión simple de K-means
- Dividir el conjunto de todos los puntos en dos clusters, escoger uno de los dos para ser dividido, e iterar hasta producir K clusters.
- Cada división se obtiene ejecutando K-means (con k=2)



Outline : Clustering Jerárquico Aglomerativo

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico
Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

Otros métodos

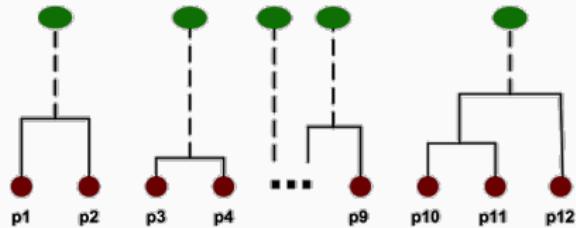
Use-case: BERTopic

Clustering Jerárquico Aglomerativo: Principio



	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Matriz de distancias



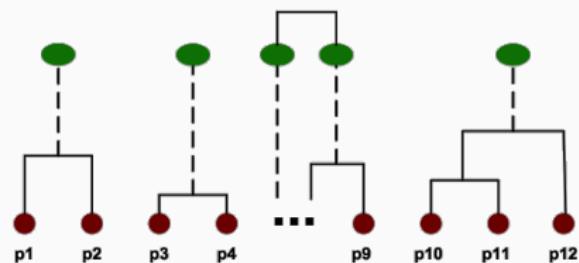
- Calcular las distancias entre clusters
- Fusionar los mas cercanos
- Re-calcular la matriz
- Se representa como un arbol (o dendogram)

Clustering Jerárquico Aglomerativo: Principio



	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Matriz de distancias



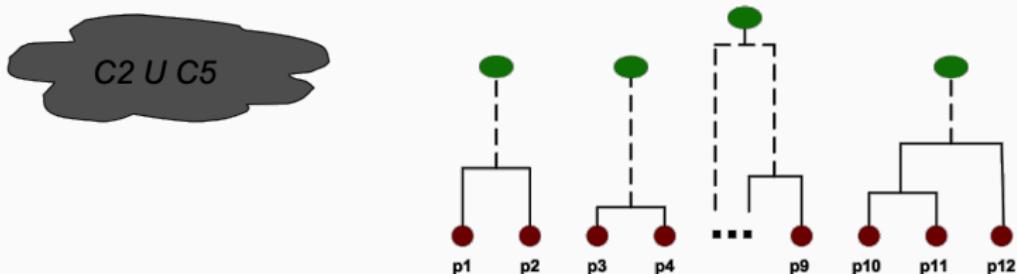
- Calcular las distancias entre clusters
- Fusionar los mas cercanos
- Re-calcular la matriz
- Se representa como un arbol (o dendogram)

Clustering Jerárquico Aglomerativo: Principio



	C1	C25	C3	C4
C1		?		
C25	?	?	?	?
C3		?		
C4		?		

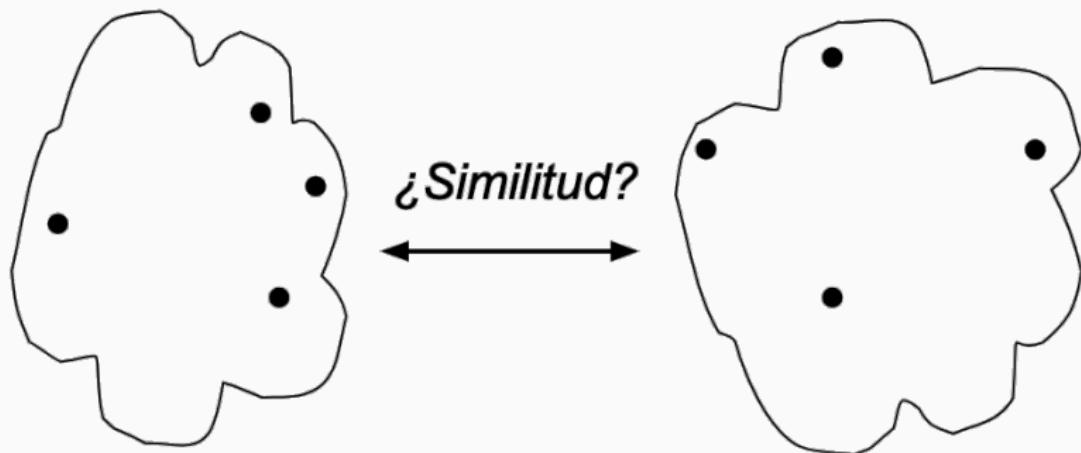
Matriz de distancias



- Calcular las distancias entre clusters
- Fusionar los mas cercanos
- Re-calcular la matriz
- Se representa como un arbol (o dendogram)

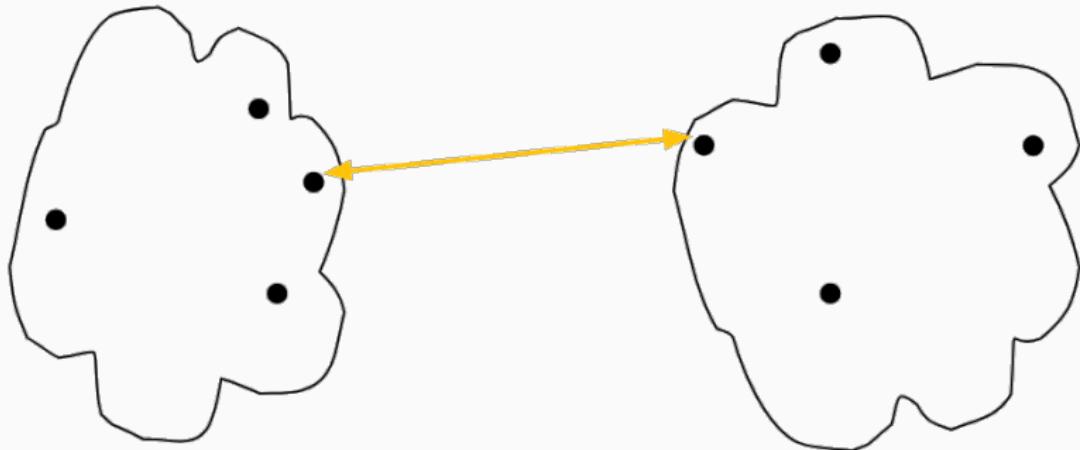
Distancias entre clusters: Linkage Criteria

- Como estimar la distancia/similitud entre clusters?



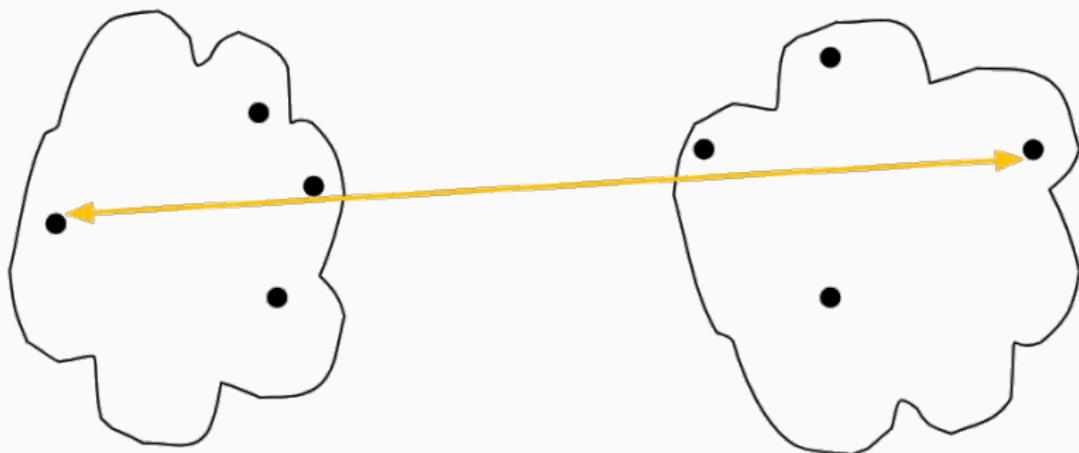
Distancias entre clusters: Linkage Criteria

- Como estimar la distancia/similitud entre clusters?
- Los puntos los mas cercanos: sensible al ruido y outliers



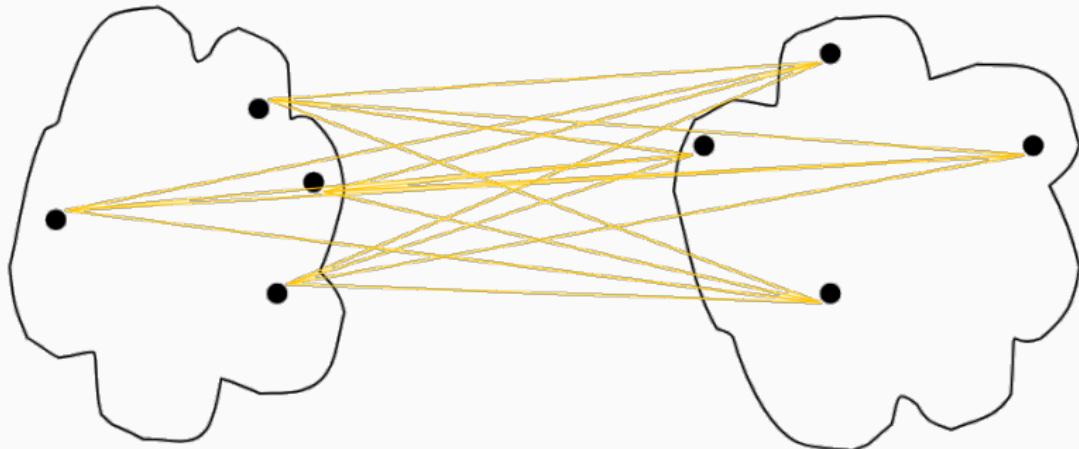
Distancias entre clusters: Linkage Criteria

- **Como estimar la distancia/similitud entre clusters?**
- Los puntos los mas cercanos: sensible al ruido y outliers
- Los puntos los mas alejados: funciona mal por cluster grandes y circulares



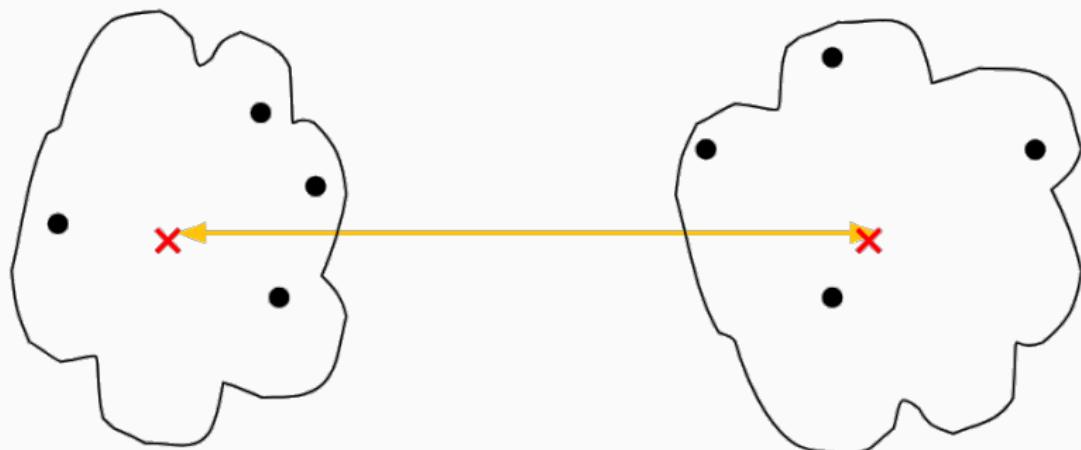
Distancias entre clusters: Linkage Criteria

- **Como estimar la distancia/similitud entre clusters?**
- Los puntos los mas cercanos: sensible al ruido y outliers
- Los puntos los mas alejados: funciona mal por cluster grandes y circulares
- En promedio: compromiso entre los dos



Distancias entre clusters: Linkage Criteria

- **Como estimar la distancia/similitud entre clusters?**
- Los puntos los mas cercanos: sensible al ruido y outliers
- Los puntos los mas alejados: funciona mal por cluster grandes y circulares
- En promedio: compromiso entre los dos
- Usando los centroides



Distancias entre clusters: Linkage Criteria

- **Como estimar la distancia/similitud entre clusters?**
- Los puntos los mas cercanos: sensible al ruido y outliers
- Los puntos los mas alejados: funciona mal por cluster grandes y circulares
- En promedio: compromiso entre los dos
- Usando los centroides
- Basado en el incremento del SSE cuando se mezclan dos clusters.
Similar a la función objetivo de k-means pero abordada con un enfoque jerárquico aglomerativo.

Outline : Hierarchical DBSCAN

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

Otros métodos

Use-case: BERTopic

- **HDBSCAN** es una extensión jerárquica de DBSCAN:
 - Explora múltiples valores de densidad, construyendo una jerarquía de clusters donde cada nivel corresponde a un criterio de densidad diferente.
 - No requiere elegir `eps` de forma fija; en su lugar, utiliza una métrica de estabilidad de clusters para decidir qué subconjuntos de la jerarquía son más relevantes.
 - Produce clusters con **forma arbitraria** y maneja bien los outliers.
- **Ventaja clave:** evita la sensibilidad a un único valor `eps` y puede adaptarse a distribuciones de densidad variables.

[User-guide here](#)

Validation Metrics

Outline : Validation Metrics

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

Otros métodos

Use-case: BERTopic

Validación de Clustering

- A diferencia del aprendizaje supervisado, la validación de **clustering** suele requerir:
 - Conocer **etiquetas verdaderas** (si existen) para usar métricas externas.
 - O emplear **métricas internas** (silhouette, SSE, etc.) que sólo usan la estructura de los clusters.
- **Ejemplos de métricas externas:**
 - Basadas en información mutua.
 - Homogeneidad, Completeness, V-measure.
 - Índice de Rand, Fowlkes–Mallows.
- **Ejemplos de métricas internas:**
 - Índice Silhouette.
 - SSE / inercia (usado en K-means).

Métricas basadas en MI, Homogeneidad, Completeness y V-measure

Información Mutua (Mutual Information, MI)

- Mide la **dependencia** entre variables aleatorias.
- Cuanto mayor sea el valor, más información comparten la distribución de clústeres y las etiquetas verdaderas.

Homogeneidad y Completeness

- **Homogeneidad**: todos los puntos de un cluster pertenecen a la misma clase real.
- **Completeness**: todos los puntos de una clase real están contenidos en un único cluster.
- **V-measure**: media armónica de Homogeneidad y Completeness.

Fowlkes-Mallows e Índice Silhouette

Fowlkes–Mallows

- Se basa en la comparación de **pares** de puntos.
- Mezcla **precisión** y **recall** de la partición de clusters contra las etiquetas reales.
- Oscila entre 0 (peor) y 1 (mejor).

Silhouette

- Métrica interna: no requiere etiquetas reales.
- Calcula la **cohesión** e **separación** de cada punto y promedia sobre todo el conjunto.
- Varía entre -1 y 1: valores altos indican clusters densos y bien separados.

Ejemplos

Outline : Ejemplos

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

Otros métodos

Use-case: BERTopic

Outline : Otros métodos

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

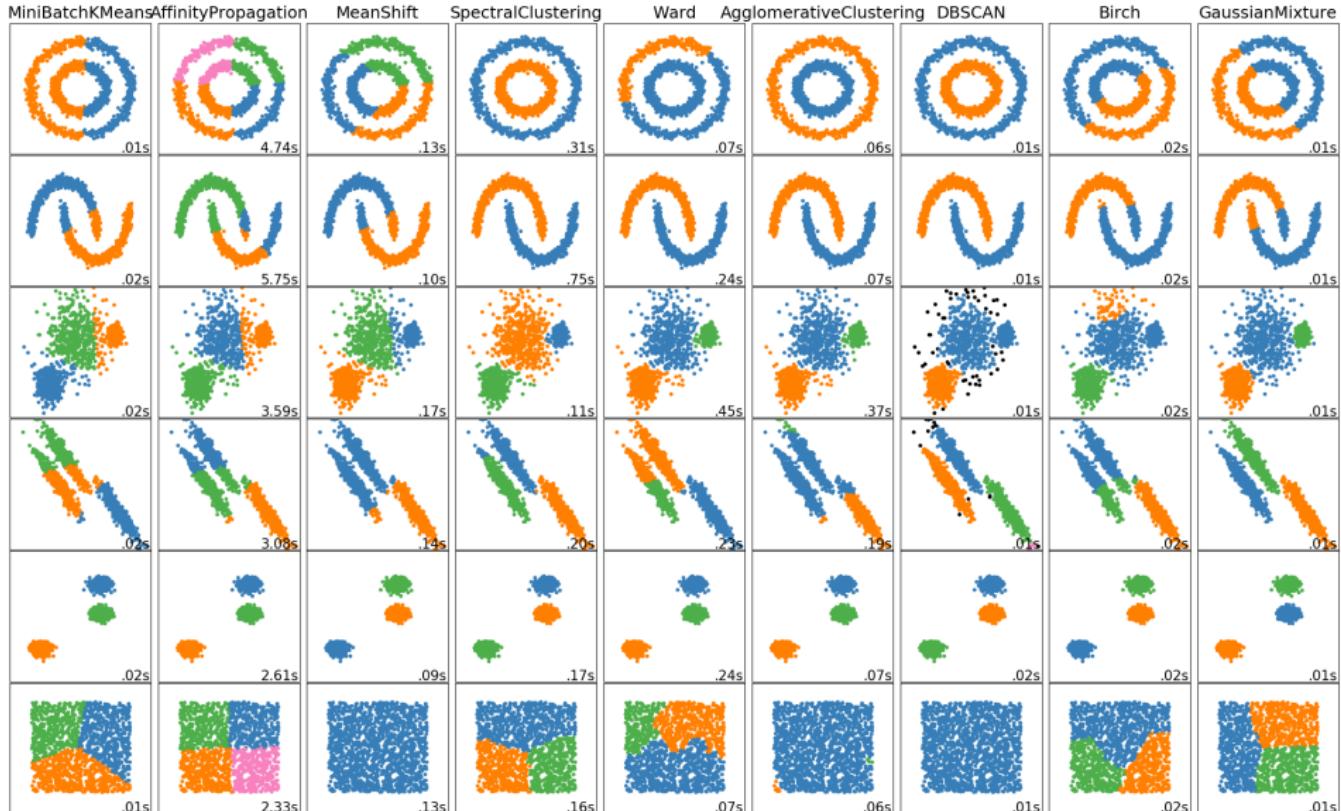
Validation Metrics

Ejemplos

Otros métodos

Use-case: BERTopic

Otros métodos de agrupamiento



Otros métodos de agrupamiento

Method Name	Use Case
K-Means	General-purpose, even cluster size, flat geometry, inductive
Affinity Propagation	Many clusters, uneven cluster size, non-flat geometry, inductive
Mean-Shift	Many clusters, uneven cluster size, non-flat geometry, inductive
Spectral Clustering	Few clusters, even cluster size, non-flat geometry, transductive
Ward Hierarchical Clustering	Many clusters, possibly connectivity constraints, transductive
Agglomerative Clustering	Many clusters, non-Euclidean distances, transductive
DBSCAN	Non-flat geometry, uneven cluster sizes, outlier removal, transductive
OPTICS	Non-flat geometry, uneven cluster sizes, variable density, outlier removal, transductive
Gaussian Mixtures	Flat geometry, good for density estimation, inductive
BIRCH	Large dataset, outlier removal, data reduction, inductive

Mas informaciones [aca](#)

Outline : Use-case: BERTopic

Clustering

Principio

K-means

DBSCAN

Hierarchical Clustering

Bisecting K-means

Clustering Jerárquico

Aglomerativo

Hierarchical DBSCAN

Validation Metrics

Ejemplos

Otros métodos

Use-case: BERTopic

BERTopic

BERTTopic is a topic modeling technique that leverages 🤗 transformers and c-TF-IDF to create dense clusters allowing for easily interpretable topics whilst keeping important words in the topic descriptions.

BERTTopic supports all kinds of topic modeling techniques:

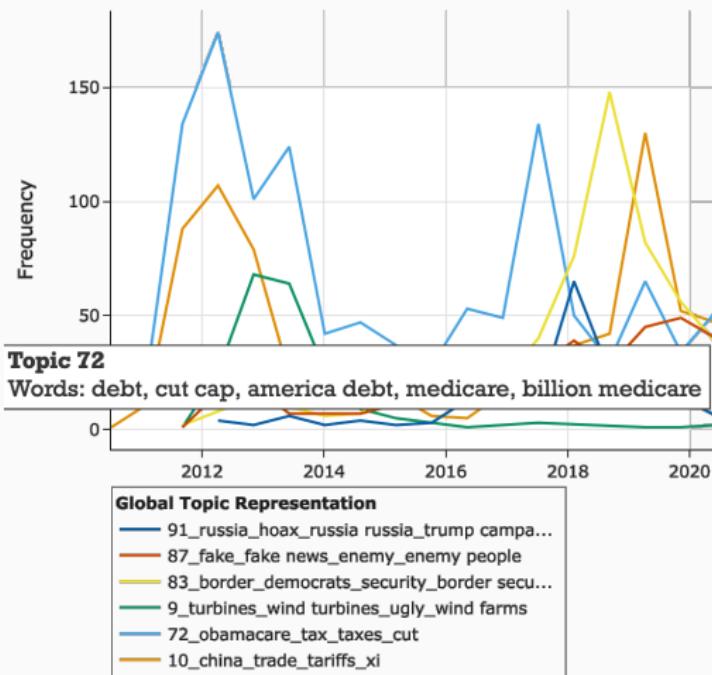
Guided	Supervised	Semi-supervised
Manual	Multi-topic distributions	Hierarchical
Class-based	Dynamic	Online/Incremental
Multimodal	Multi-aspect	Text Generation/LLM
Zero-shot (new!)	Merge Models (new!)	Seed Words (new!)



Corresponding medium posts can be found [here](#), [here](#) and [here](#). For a more detailed overview, you can read the [paper](#) or see a brief overview.

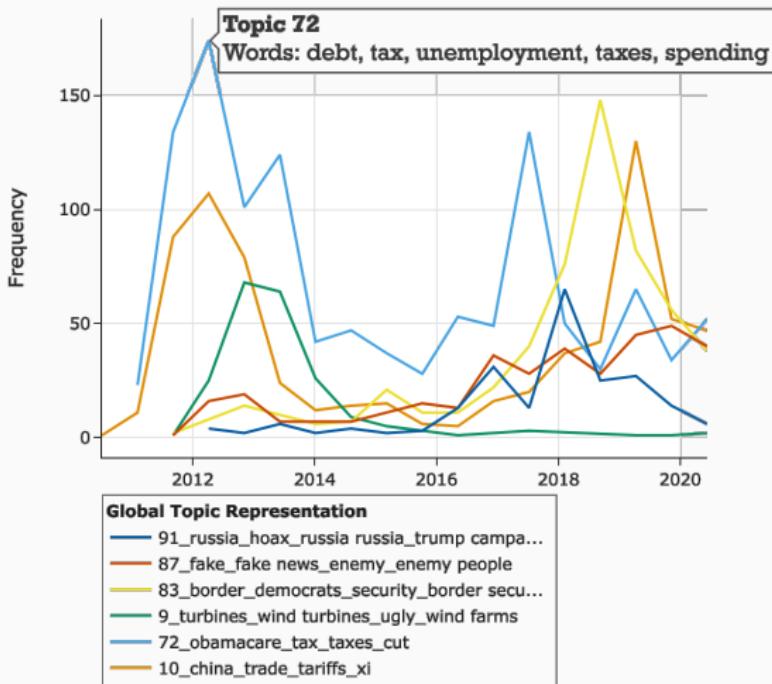
Figure 10: A simple Library based on LM for topic modeling in text corpora

Topics over Time

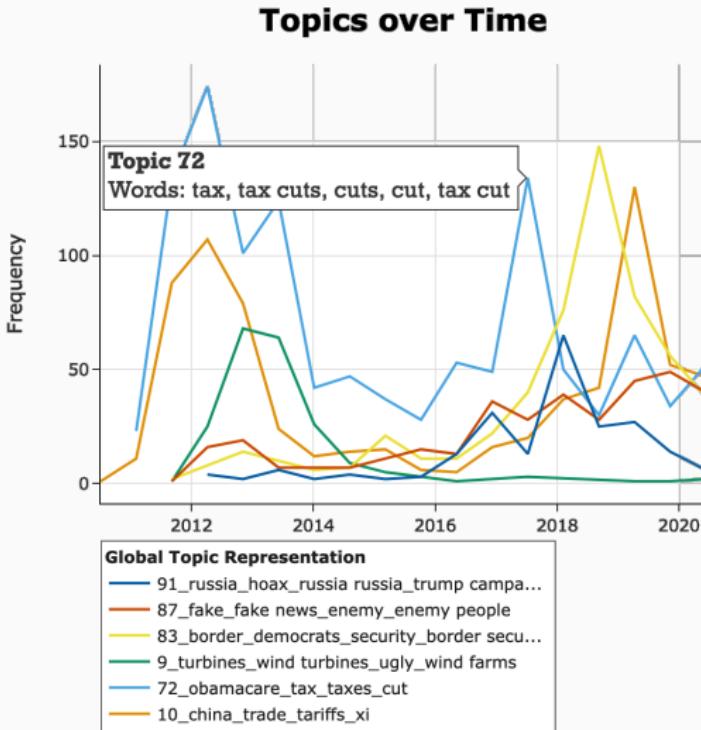


Can be used to cluster and easily visualize dynamic topic modeling or hierarchical clustering!

Topics over Time

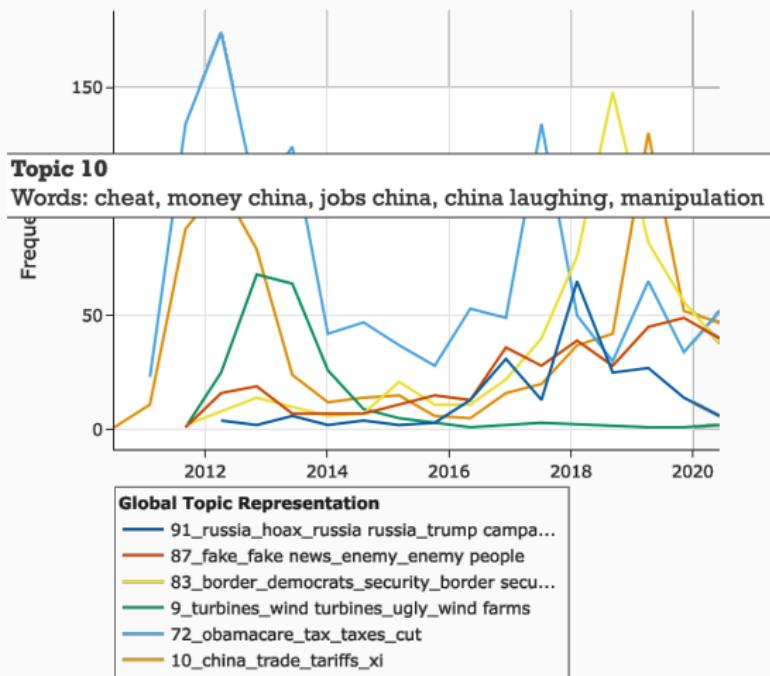


Can be used to cluster and easily visualize dynamic topic modeling or hierarchical clustering!



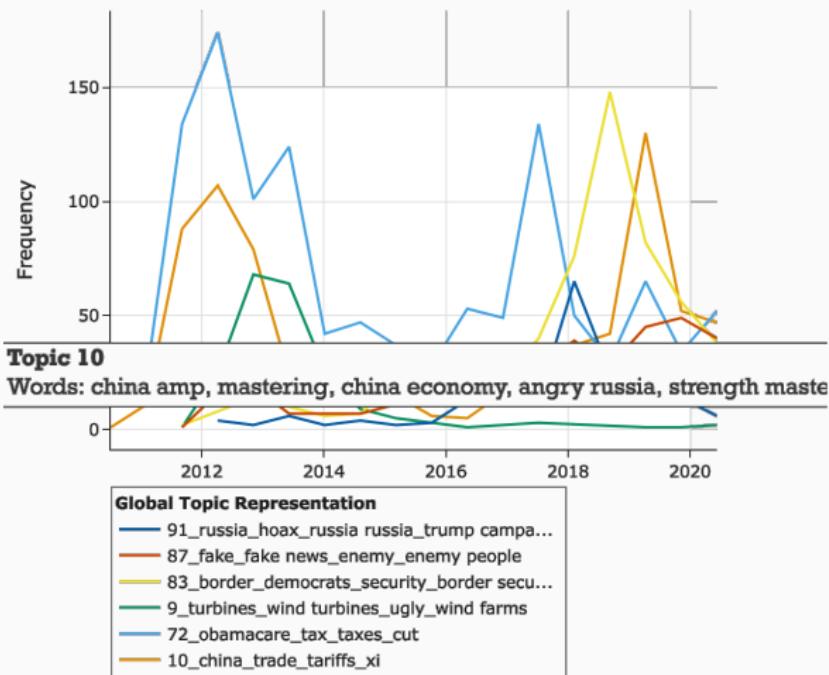
Can be used to cluster and easily visualize dynamic topic modeling or hierarchical clustering!

Topics over Time

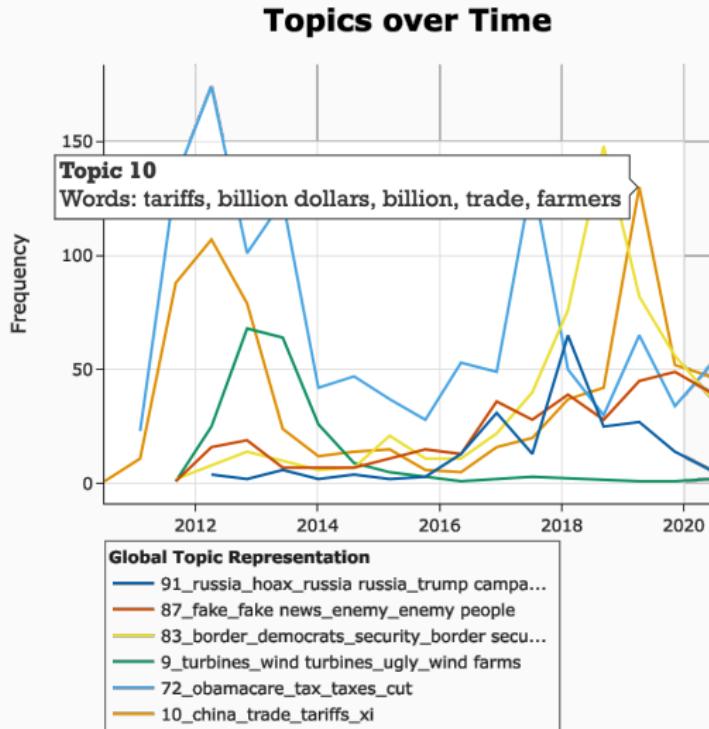


Can be used to cluster and easily visualize dynamic topic modeling or hierarchical clustering!

Topics over Time



Can be used to cluster and easily visualize dynamic topic modeling or hierarchical clustering!



Can be used to cluster and easily visualize dynamic topic modeling or hierarchical clustering!

Questions?

References i