Clustering

비슷한 문서/문장을 label 없이 임베딩 벡터를 사용하여 묶는 방법

Clustering

- 군집화는 데이터에서 비슷한 객체들을 하나의 그룹으로 묶습니다.
  - 각 객체들이 어떤 군집으로 할당되어야 하는지에 대한 정답 정보 (y) 가 없기 때문에 unsupervised 알고리즘으로 분류됩니다.
  - 군집화 방법들은 각 객체들의 유사도(거리) 정보를 이용합니다.
    유사한 객체를 하나의 군집으로 묶습니다.

Clustering

- 좋은 군집에 대한 기준은 다양하지만 공통적으로
  "군집 내 객체들은 비슷하며, 군집 간 객체들은 이질적"임을 추구합니다.
- 객체의 표현과 거리를 이용하는 방식에 따라 다양한 방법이 있습니다.
  - centroids models
  - connectivity models
  - density models
  - ...
- 그 전에 근본적으로 데이터의 representation 과 dissimilarity measure 가 잘 정의되어야 합니다.

k-means

k-means clustering

- 유사도
  - n개의 데이터 X에 대하여 두 데이터 x_i, x_j간에 정의되는 임의의 거리 (x_i, x_j)
    - 유클리디언, 코싸인 등 벡터에서 정의되는 모든 거리 척도

- 그룹화의 방식
  - 그룹의 개수는 k 개라고 가정
  - 각 그룹을 centroid vector (평균 벡터)로 표현한 뒤, 이를 업데이트

k-means

- k-means 는 빠릅니다.
  - 계산복잡도가 작습니다. $O(n * i * k)$
  - pairwise distance 를 요구하지 않기 때문에 대량의 데이터에 적합하며,
  - row 단위로 학습하기 때문에 mini-batch / 분산환경의 구현이 쉽습니다.

- 대량의 문서 집합의 군집화에는 가장 현실적인 방법입니다.

[1] David Sculley. Web-scale k-means clustering. In Proceedings of the 19th international conference on World wide web, pages 1177–1178. ACM, 2010

k-means

- 문서 군집화는 거리 척도가 중요합니다.
  - 일반적으로 k-means 는 Euclidean distance 를 이용합니다.
  - 문서 간 유사도는 두 문서의 공통된 단어 유무가 제일 중요한 정보입니다.
    - Euclidean distance 는 이를 고려하지 않습니다.
  - Sparse vector 형식으로 문서를 표현할 경우에는 Jaccard, Pearson, Cosine 을 쓸 수 있지만, Euclidean 만은 쓰지 말아야 합니다 [1].
  - Jaccard, Pearson, Cosine 모두 문서 벡터의 방향성에 관련된 척도입니다.

[1] Anna Huang. Similarity measures for text document clustering. In Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand, pages 49–56, 2008

Spherical k-means

- Cosine distance 를 이용하는 k-means 를 Spherical k-means 라 합니다 [1].
  - Distance measure 외의 학습 방법은 Lloyd 와 같습니다만,
  - 이 차이로 결과는 확연히 다릅니다.

- scikit-learn 에는 metric 이 Euclidean 으로 고정되어 있습니다.

[1] Inderjit S Dhillon and Dharmendra S Modha. Concept decompositions for large sparse text data 120 using clustering. Machine learning, 42(1):143–175, 2001.

Difficulty. Defining k

- k-means 는 군집의 개수를 사용자가 직접 정의하여야 합니다.
- Silhouette 은 군집화 품질의 척도로, 적절한 k 를 정하는데 이용됩니다.

$$s(x) = \frac{b - a}{\max(a, b)}, \qquad s(X) = mean(s(x))$$

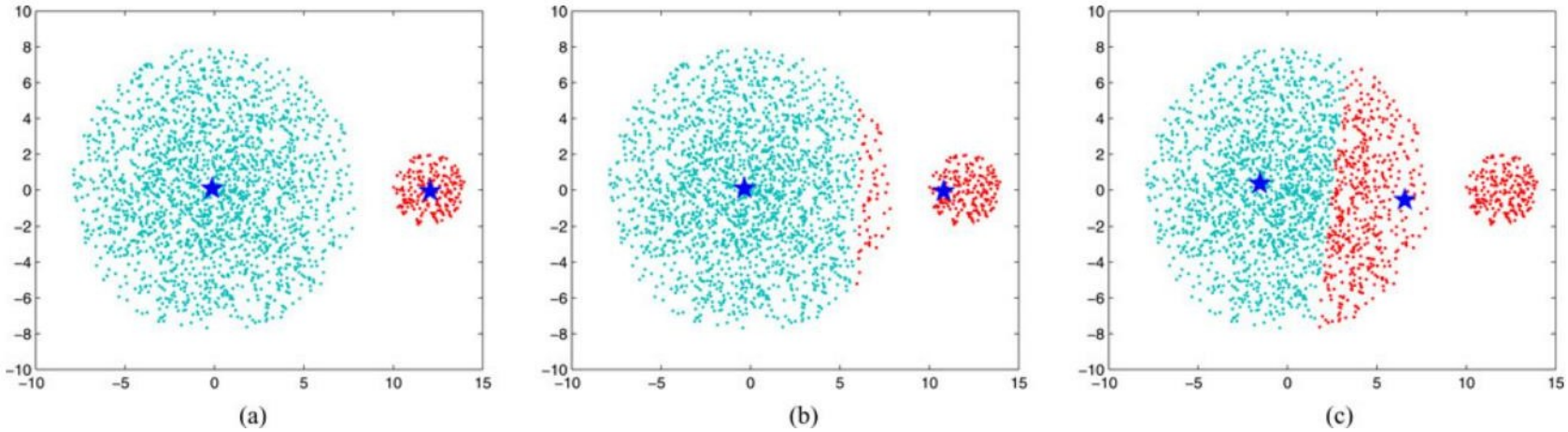
a: mean distance between a sample and all other points in the same class

b: mean distance between a sample and all other points in the next nearest class

- Silhouette 은 사후 평가 방법입니다.
  - 모든 k 에 대하여 테스트 할 비용도 만만치 않습니다. 특히, 고차원 데이터
- Silhouette 은 model fitness measure 입니다.
  - 높은 값이 "우리가 예상하는" 좋은 군집화 결과를 의미하지는 않습니다.

Difficulty. Uniform effect

- Imbalanced class data 일 때, class distribution 이 잘 반영되지 않고, 모든 군집의 크기가 균일해지는 현상 [1]



- k-means type 은 몇 번의 반복으로 거의 수렴합니다.
  - "repeat until converged" → Uniform effect 가 일어날 가능성 높음

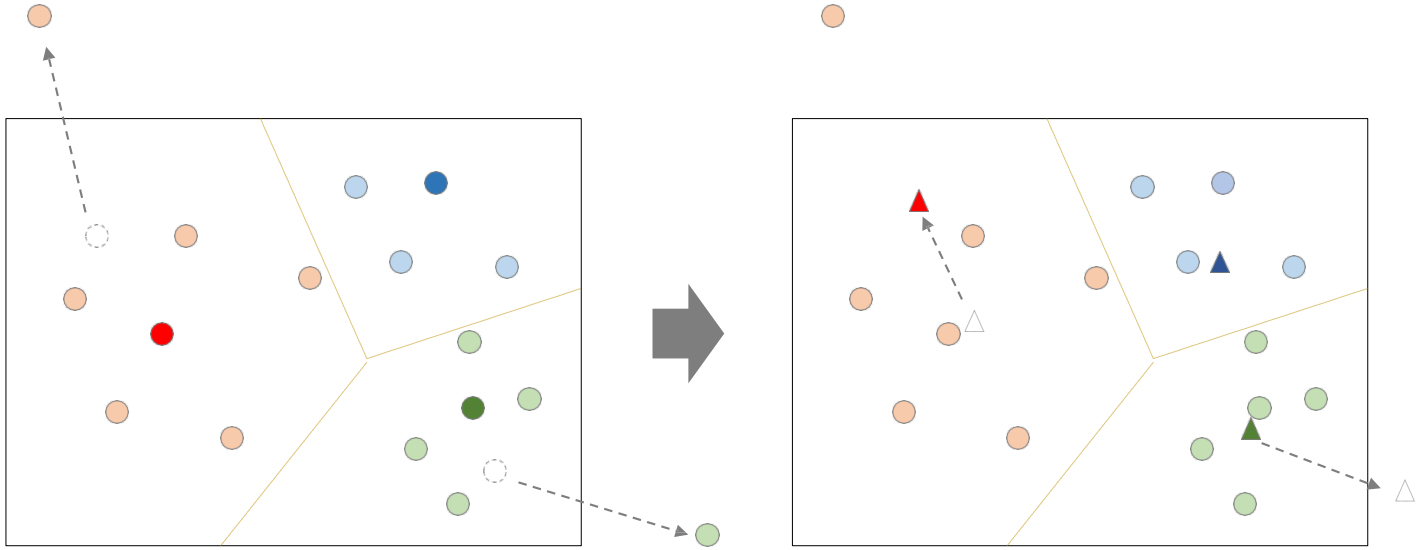
[1] Hui Xiong, Junjie Wu, and Jian Chen. K-means clustering versus validation measures: a data distribution perspective. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39(2):318–331, 2009.

Solution. Defining k

- 현실적인 방법은 예상하는 군집의 개수보다 크게 k 설정한 뒤,
  후처리로 비슷한 군집을 병합합니다. (AutoML)
  - 학습 후, 하나의 군집이 여러개로 나뉘어 졌는지를 확인하기는 쉽습니다.
  - 하나의 군집에서 잘못된 점을 찾는 것이 더 어려우며,
    그 점들의 후처리도 어렵습니다.
  - 실제 군집의 개수보다 k 가 크면 major 군집들이 여러 개로 나뉘어집니다.
  - k 가 작으면 minor 군집이 찢어질 가능성이 높습니다.
    - 이 때 centroids 는 major 편입니다.

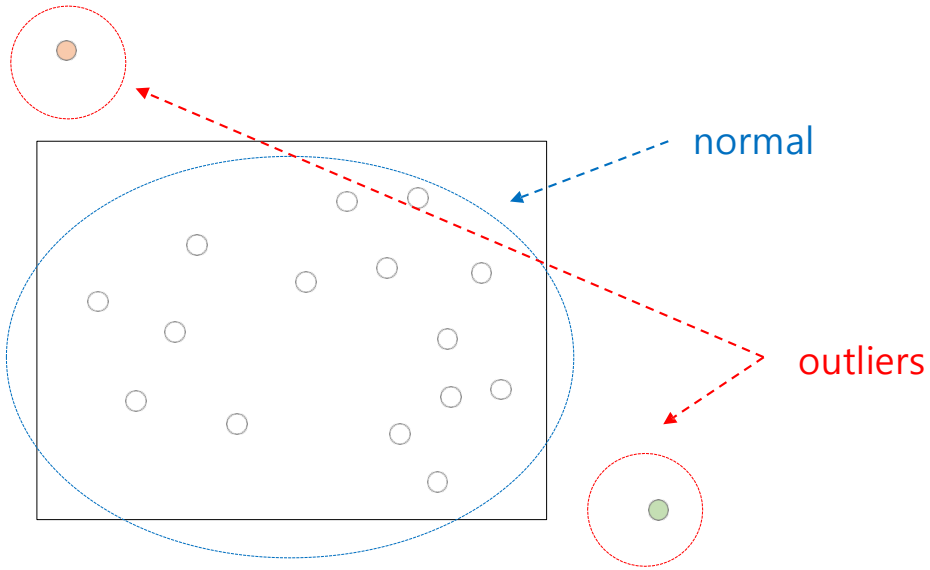
Difficulty. Sensitive to noise points

- 모든 점을 반드시 한 개 이상의 군집으로 assign 하기 때문에, 일단 가장 가까운 군집에 할당되어 centroid 를 크게 움직입니다.



Solutions. Sensitive to noise points

- 데이터의 노이즈를 미리 제거하는 것이 좋습니다.
  - 텍스트 데이터에서는 길이가 극단적으로 짧거나 긴 문서들은 노이즈입니다.
  - Cosine distance 도 길이가 1, 2 처럼 지나치게 짧은 문서 간의 거리는 잘 정의되지 않습니다.



k-means

sklearn.cluster.KMeans¶

class sklearn.cluster. **KMeans** (*n_clusters=8, init='k-means++', n_init=10, max_iter=300, tol=0.0001, precompute_distances='auto', verbose=0, random_state=None, copy_x=True, n_jobs=1, algorithm='auto'*) [source]

- k-means 는 수렴 속도가 매우 빠릅니다. 절대 max_iter=300 으로 설정할 필요가 없습니다.
  - k-means 역시 근사알고리즘입니다. 반복을 많이 한다하여 더 좋은 결과가 나오지 않습니다.
  - 일반적으로 20 ~ 30 반복이면 거의 수렴합니다.

GMM / BGMM

Agglomerative clustering

DBSCAN

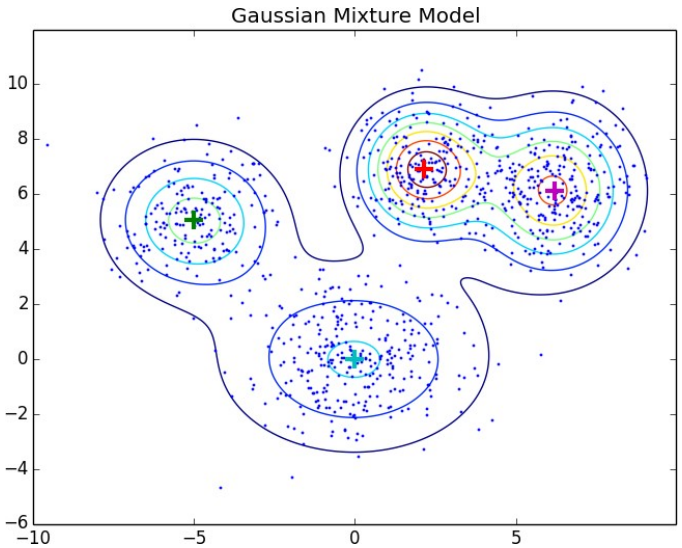
Gaussian Mixture Model (GMM)

- 유사도
  - n개의 데이터 X에 대하여 k개의 Gaussian distribution의 확률값 $P(x_i | G_j)$
  - 데이터의 분포 (밀도)를 고려한 k-means 라 생각할 수 있습니다.
  - Gaussian 을 이용하기 때문에 Euclidean distance 에 대해서만 정의됩니다.

- 그룹화의 방식
  - 데이터의 분포를 가장 잘 설명할 수 있는 k개의 Gaussian 의 parameter 인 (μ, Σ) 를 학습합니다.

Gaussian Mixture Model (GMM)

- 데이터가 Centroids 를 중심으로 Gaussian 을 따른다고 가정합니다.
  - 군집 사이에 밀도 차이가 있을 경우에 적합합니다.



< 같은 확률값을 가지는 점들의 등고선 >

http://yulearning.blogspot.kr/2014/11/einsteins-most-famous-equation-is-emc2.html

Gaussian Mixture Model (GMM)

- scikit-learn 에 GMM 이 구현되어 있습니다.
  - k-means 처럼 n_components 를 사용자가 정의합니다.
  - 원형이 아닌 데이터의 분포를 학습하기 위해 분산행렬의 모양을 선택합니다.
    - covariance_type : {'full', 'tied', 'diag', 'spherical'}

sklearn.mixture.GaussianMixture

class sklearn.mixture. GaussianMixture (*n_components=1, covariance_type='full', tol=0.001, reg_covar=1e-06, max_iter=100, n_init=1, init_params='kmeans', weights_init=None, means_init=None, precisions_init=None, random_state=None, warm_start=False, verbose=0, verbose_interval=10*) [source]

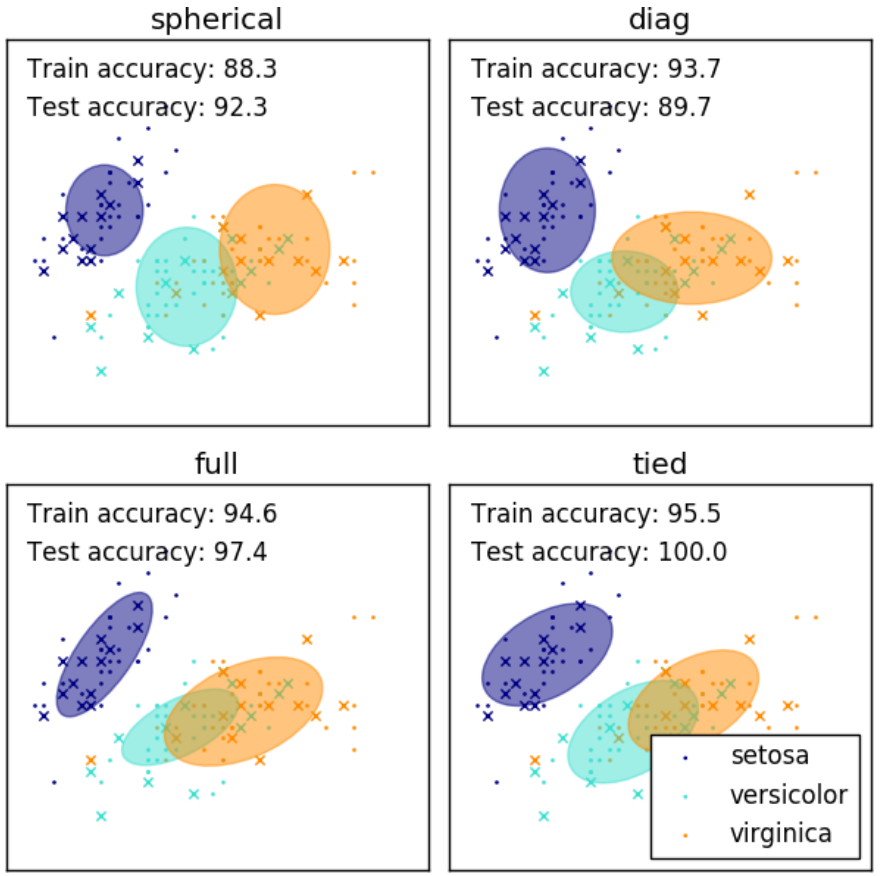
Gaussian Mixture Model (GMM)

'full': its own general covariance matrix,

'tied': share the same general covariance matrix,

'diag': its own diagonal covariance matrix,

'spherical': its own single variance



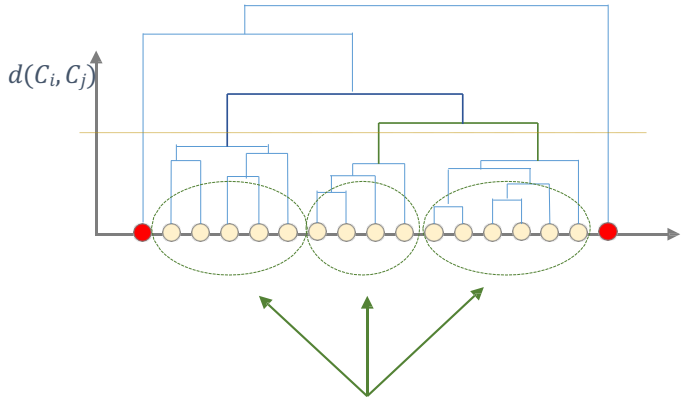
http://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html#sklearn.mixture.GaussianMixture

Hierarchical clustering

- 유사도
  - 두 데이터 x_i, x_j간에 정의되는 임의의 거리 $d(x_i, x_j)$
    - 그룹 간의 거리는 $d(C_i, C_j)$를 기반으로 정의 (min, max, average 등)
    - single linkage / complete linkage
- 그룹화의 방식
  - 거리가 가장 가까운 두 집합을 하나의 집합으로 묶으며, 모든 집합이 하나의 집합이 될 때 까지 반복합니다.

Hierarchical clustering

- Outliers 의 영향을 덜받습니다.
  - Single linkage 는 가장 가까운 점들을 하나씩 이어나갑니다.
  - 마지막까지 다른 점들과 큰 군집으로 묶이지 않는 점들이 outliers 입니다.



다른 점들은 큰 3개의 그룹으로 묶이지만,
붉은색 점들은 마지막에 큰 군집으로 묶임

Hierarchical clustering

- 계산 비용이 비쌉니다.
  - 데이터의 개수가 N개라고 할 때, 모든 점들간의 거리를 계산해야 하기 때문에 $O(N^2)$ 계산 공간과 비용이 필요합니다.
  - 상대적으로 k-means 보다 큰 계산 공간과 계산 시간을 필요로 합니다.

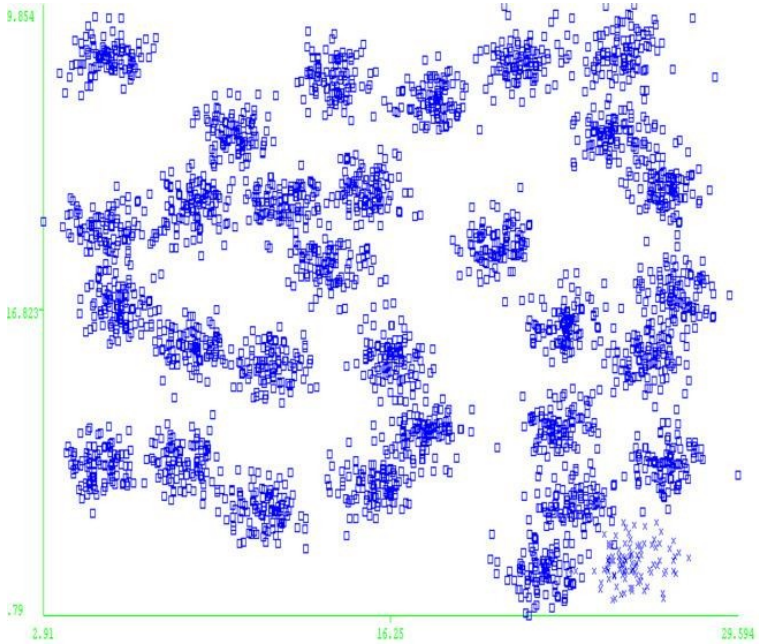
Hierarchical clustering

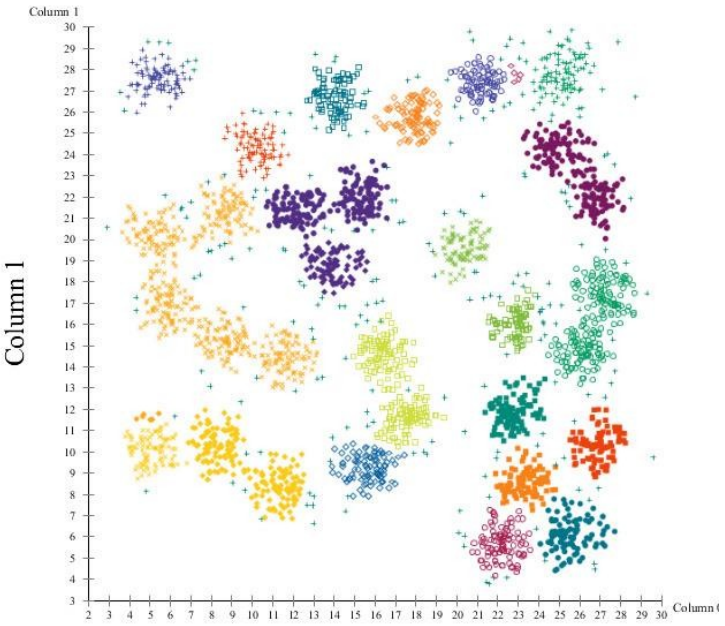
- 고차원 벡터에서 잘 작동하지 않습니다.
  - 고차원에서는 최인접이웃들의 거리 외에는 정보력이 없습니다.
  - average linkage 는 두 군집의 모든 점들 간의 거리의 평균을 군집 간의 거리로 이용하기 때문에 대부분의 군집 간 거리가 비슷합니다.
  - 고차원 데이터에서는 최초의 몇 단계 외에는 의미가 없습니다.

DBSCAN

- Density-Based Spatial Clustering of Applications with Noise
  - 모든 점이 반드시 그룹에 속하지 않는다고 가정합니다 (노이즈)
  - 유사도
    - n 개의 데이터 X 에 대하여 두 데이터 $x_i,\ x_j$ 간에 정의되는 임의의 거리 $d(x_i,\ x_j)$
  - 그룹화의 방식
    - Threshold 이상의 밀도를 지닌 점들을 모두 이어나갑니다.

DBSCAN





http://quipu-strands.blogspot.kr/

DBSCAN

- Parameters 에 민감합니다.
  - 군집을 결정하는 밀도값 threshold 에 의하여 데이터에서의 노이즈 비율이 예민하게 변합니다.
- 계산 비용이 큽니다.
  - DBSCAN 은 모든 점들간의 거리를 한 번 이상 계산해야하기 때문에 $O(N^2)$ 의 계산 비용을 필요로 합니다.

Summary

군집화

- k-means 는 centroids 를 중심으로 구형의 군집을 만듭니다.
  - Euclidean 을 이용할 경우 구 형태의 군집을 만듭니다.
  - Cosine 을 이용할 경우, 벡터의 각도를 기준으로 만들어진 partition 입니다.

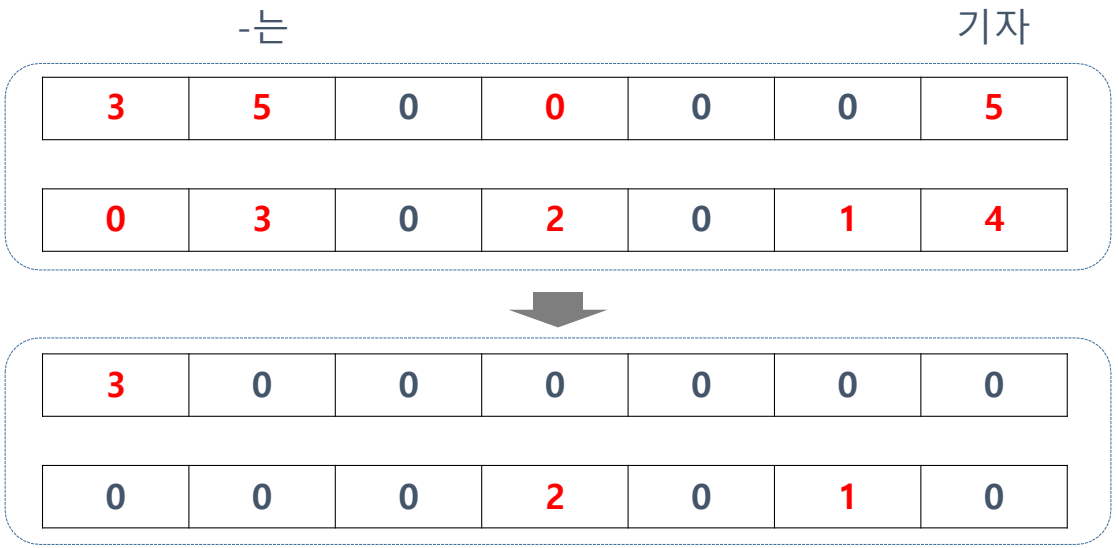
- Hierarchical clustering, DBSCAN 은 복잡한 모양의 데이터용입니다.
  - Sparse vector + Cosine 의 공간은 복잡하지 않습니다.
  - 단순한 알고리즘이 빠르며 안정적입니다.

문서 군집화

- 고차원 벡터에서는 매우 가까운 거리만 의미를 지닙니다.
  - k-means 이용 시 k 가 지나치게 작을 경우 먼 문서들이 하나의 클러스터에 할당될 수 있기 때문에 불안정한(unstable) 학습이 될 수 있습니다.
  - 고차원 벡터의 경우 충분히 큰 k 로 군집화를 수행한 뒤, 동일한 의미를 지니는 군집들을 하나로 묶는 후처리 (post-processing) 방식을 추천합니다.

문서 군집화

- 불필요한 단어들을 제거하는 것은 군집화 알고리즘에 도움이 됩니다.
  - Document frequency (DF)가 지나치게 높거나 낮은 단어
  - 뉴스 문서에서 '기자'와 같은 단어나 '-는'과 같은 단어



문서 군집화

- 단어-문서 행렬이 뚜렷한 블록들로 구분이 된다면 군집화에 최적입니다.