Technology review: Review K-Means, kernel K-means, K-Medians, K-Medoids – Pros and cons

Introduction:

The clustering is the task of finding groups of similar documents in a collection of documents. The similarity is computed by using a similarity function. There are many clustering algorithms that can be used in the context of text data. Text document can be represented as a binary vector, considering the presence or absence of word in the document. Or we can use more refined representations which involves weighting methods such as TF-IDF. since text data has several distinct characteristics which demands the design of text-specific algorithms for the task. We describe some of these unique properties of text representation.

Unique properties of text:

- Text representation has a very large dimensionality, but the underlying data is sparse. In other words, the size of the vocabulary from which the documents are drawn is massive, but a given document may have only a few hundred words. This problem becomes even more severe when we deal with short data such as tweets.
- Words of the vocabulary of a given collection of documents are commonly correlated with each other. i.e. the number of concepts in the data are much smaller that the feature space. Thus, we need to design algorithms which take the word correlation into consideration in the clustering task.
- Since documents differ from one another in terms of the number of words they contain, normalizing document representations during the clustering process is important.

Text clustering algorithms are split into many different types such as agglomerative clustering algorithms, partitioning algorithms, and probabilistic clustering algorithms. Clustering algorithms have varied tradeoffs in terms of effectiveness and efficiency. Let's discuss Partitioning algorithms, K-menas family algorithms.

K-Means (MacQueen'67, Lloyd'57/'82) - In K-Means each cluster is represented by the center of the cluster. Given K, the number of clusters, the K-Means clustering algorithm is outlined as follows, Select K points as initial centroids, Repeat - Form K clusters by assigning each point to its closest centroid. Re-compute the centroids (i.e., mean point) of each cluster. Up Until convergence criterion is satisfied

Different kinds of measures can be used -Manhattan distance (L1 norm), Euclidean distance (L2 norm), Cosine similarity

Select K points as initial centroids

Repeat

- Form K clusters by assigning each point to its closest centroid
- Re-compute the centroids (i.e., mean point) of each cluster

Until convergence criterion is satisfied

Efficiency of K-Means: O(tKn) where n: # of objects, K: # of clusters, and t: # of iterations, Normally, K, t << n; thus, an efficient method. K-means clustering often terminates at a local optimal. Initialization can be important to find high-quality clusters. we need to specify K, the number of clusters, in advance. There are ways to automatically determine the "best" K. In practice, one often runs a range of values and selects the "best" K value. It is sensitive to noisy data and outliers. K-means is applicable only to objects in a continuous n-dimensional space. K-Means is not suitable to discover clusters with non-convex shapes, for that use density-based clustering, kernel K-means.

Advantages of K-Means –

- Relatively simple to implement.
- Scales to large data sets.
- Guarantees convergence.
- Can warm-start the positions of centroids.
- Easily adapts to new examples.
- Generalizes to clusters of different shapes and sizes, such as elliptical clusters.

Disadvantages of K-Means:

- Choosing K manually.
- Being dependent on initial values - For a low, you can mitigate this dependence by running k-means several times with different initial values and picking the best result. As size increases, you need advanced versions of k-means to pick better values of the initial centroids
- Clustering data of varying sizes and density - k-means has trouble clustering data where clusters are of varying sizes and density. To cluster such data, you need to generalize k-means as described in the Advantages section.
- Clustering outliers - Centroids can be dragged by outliers, or outliers might get their own cluster instead of being ignored. Consider removing or clipping outliers before clustering.

Scaling with number of dimensions - As the number of dimensions increases, a distance-based similarity measure converges to a constant value between any given examples. Reduce dimensionality either by using PCA on the feature data, or by using "spectral clustering" to modify the clustering algorithm as explained below.

K-Medoids: Instead of taking the mean value of the object in a cluster as a reference point, medoids can be used, which is the most centrally located object in a cluster. In K-medoids algorithm, first select k clustering center points randomly from n data objects before computing the distance of other data objects to each clustering center, then choose the one which is closest to clustering center to set up an initial partition, and then use the iteration methods to change the clustering center continuously until the most suitable fixed partition is found.

The K-Medoids clustering algorithm:

- Select K points as the initial representative objects (i.e., as initial K medoids)

- Repeat
    - Assigning each point to the cluster with the closest medoid
    - Randomly select a non-representative object oi
    - Compute the total cost S of swapping the medoid m with oi
    - If S < 0, then swap m with oi to form the new set of medoids
- Until convergence criterion is satisfied

Select initial K medoids randomly Repeat Object re-assignment Swap medoid m with oi if it improves the clustering quality Until convergence criterion is satisfied

PAM (Partitioning Around Medoids: Kaufmann & Rousseeuw 1987) It starts from an initial set of medoids, and Iteratively replaces one of the medoids by one of the non-medoids if it improves the total sum of the squared errors (SSE) of the resulting clustering. PAM works effectively for small data sets but does not scale well for large data sets (due to the computational complexity).

Computational complexity: PAM: O(K(n − K)2) (quite expensive!). Efficiency improvements on PAM. To improve efficiency, CLARA (Kaufmann & Rousseeuw, 1990) is proposed which calculates PAM on samples; O(Ks2 + K(n − K)), s is the sample size. Whereas CLARANS (Ng & Han, 1994): Randomized re-sampling, ensuring efficiency + quality.

Advantages

- K-Medoids method is more robust than k-Means in the presence of noise and outliers

Disadvantages

- K-Medoids is more costly that the k-Means method
- Like k-means, k-medoids requires the user to specify k
- It does not scale well for large data sets

K-Medians: K-Medians is good in handling outliers by Computing Medians, Medians are less sensitive to outliers than means. Think of the median salary vs. mean salary of a large firm when adding a few top executives! Instead of taking the mean value of the object in a cluster as a reference point, medians are used (L1-norm as the distance measure).

K-Medians clustering algorithm:

- Select K points as the initial representative objects (i.e., as initial K medians)
- Repeat
    - Assign every point to its nearest median
    - Re-compute the median using the median of each individual feature
- Until convergence criterion is satisfied

Advantages:

- Less sensitive to outliers.

Conclusion: K- mean is efficient for large data sets but sensitive to outliers. K-means clustering is rather easy to apply to even large data sets, particularly when using heuristics. It has been successfully used in market segmentation, computer vision, and astronomy among many other domains. It often is used as a preprocessing step for other algorithms, for example to find a starting configuration, This use of k-means has been successfully combined with simple, linear classifiers for semi-supervised learning in NLP (specifically for named entity recognition).

Reference:

J. MacQueen. Some Methods for Classification and Analysis of Multivariate Observations. In Proc.

of the 5th Berkeley Symp. on Mathematical Statistics and Probability, 1967

A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. Prentice Hall, 1988

L. Kaufman and P. J. Rousseeuw. Finding Groups in Data: An Introduction to Cluster Analysis. John

Wiley & Sons, 1990