Kmeans-Algorithm

Algorithm Process:

- Given data input $\{x^1, x^2, \ldots, x^n\}$;
- Initial k cluster centers C = $\{C_1, C_2, \ldots, C_k\}$;
- Let μ^1, μ^2,…, μ^k denote the centroid vectors for cluster centers C;
- Decide cluster assignment for each data point by assigning to nearest center such that the averaged distance from each data point to centroid is minimized;

To find the nearest center, we can perform different distance method:

(1) Euclidean distance:

$$\mathrm{d}(x, \mu) = \sqrt{\sum_{i=1}^{n} (x^i - \mu^j)^2}$$

(2) Minkowski distance:

$$\mathrm{d}(x, \mu) = \sqrt[p]{\sum_{i=1}^{n} (x^i - \mu^j)^p}$$

(3) Manhattan distance:

$$\mathrm{d}(x, \mu) = \sum_{i=1}^{n} |x^i - \mu^j|$$

(4) Inf distance:

$$\mathrm{d}(x, \mu) = \max_i(|\, x^i - \mu^j|)$$

- Update centroid vectors $\mu^j = \underset{\mu^j}{\operatorname{argmin}} \sum_{x^i \text{ in } C_j} \mathrm{distance}(x^i, \ \mu^j)$
- Repeat the steps until we find a local optimal.

Comment:

The different choices of initial partition can greatly affect results.

For n data points, there are k^m possibilities. It is easy to find a local optimal but hard to find global optimal – NP Hard problem.

Difficult to interpret the quality of the clusters produced.

The algorithm is interpretable.

Hard to determine k.

Hard to converge if dataset is not convex.

Model is noise.