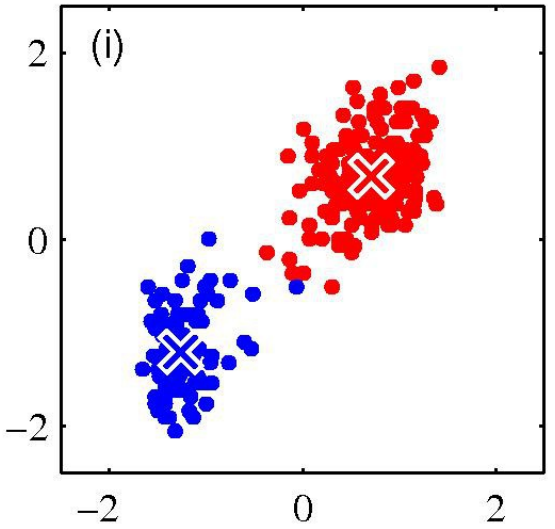
Clustering

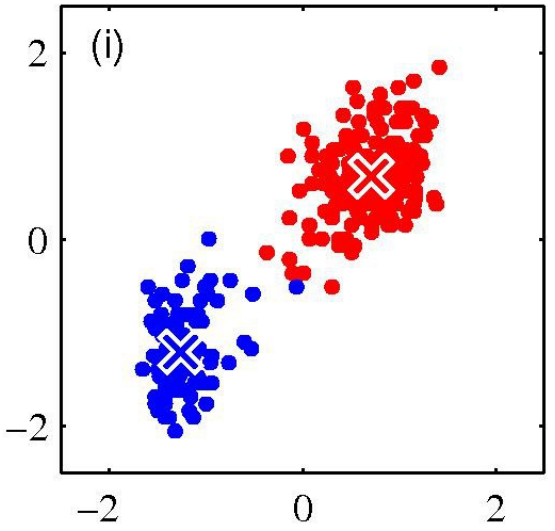
K-means clustering



Clustering

Motivation: Identify clusters of data points in a multidimensional space, i.e. partition the data set $\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$ into K clusters.

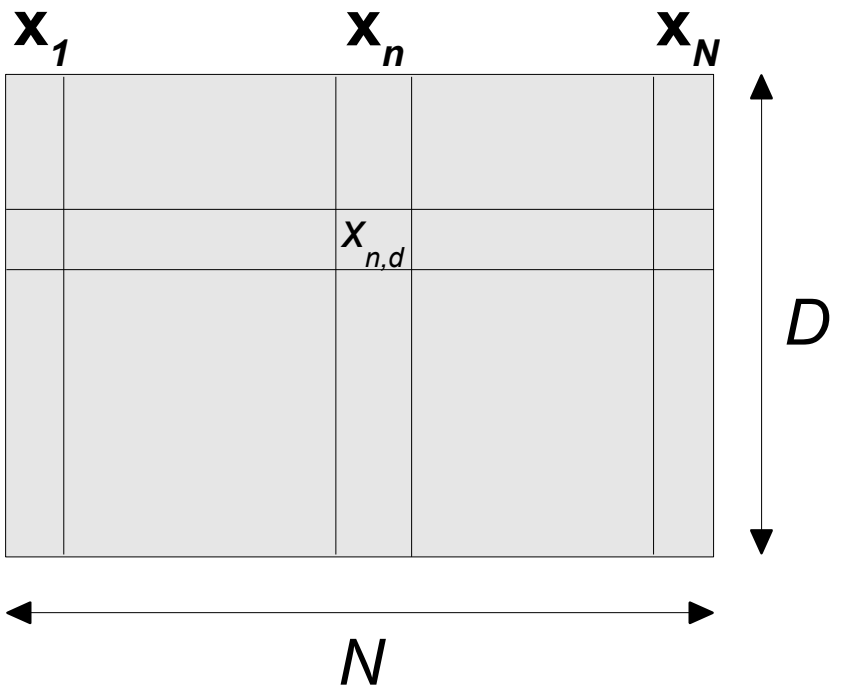
Intuition: A cluster is a group of data points with small inter-point distances compared with the distances to points not in the cluster.



Many approaches: *K*-means clustering, hierarchical clustering, self-organizing maps, …

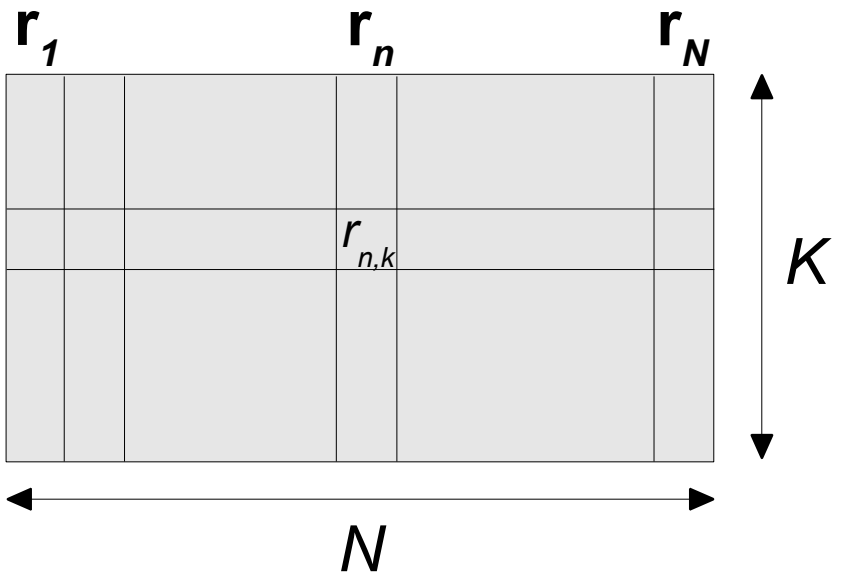
K-means clustering (1)

Data points: *N* observations of a random *D*-dimensional Euclidian variable **x**, i.e. $\{\mathbf{x}_1,...,\mathbf{x}_N\}$, where $\mathbf{x}_n = (x_{n,1}, x_{n,2},..., x_{n,D})$.



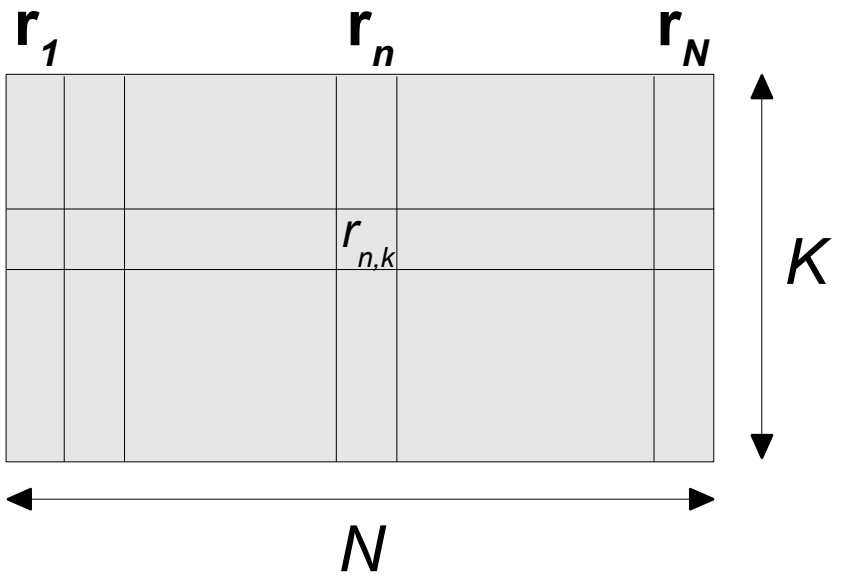
K-means clustering (2)

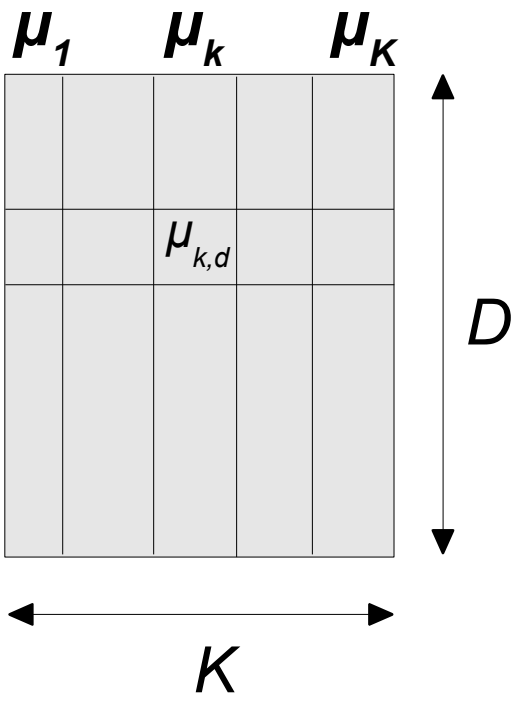
Cluster assignment: Each data point $\mathbf{x}_n$ is assigned to precisely one of *K* clusters, where *K* is given. The clustering is given by $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$, where $r_{n,k}=1$ if $\mathbf{x}_n$ is assigned to cluster *k* and 0 otherwise.



K-means clustering (2)

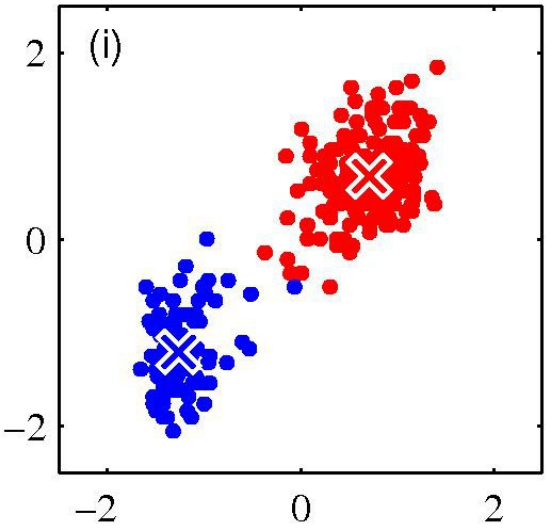
Cluster assignment: Each data point $\mathbf{x}_n$ is assigned to precisely one of K clusters, where K is given. The clustering is given by $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$, where $r_{n,k}=1$ if $\mathbf{x}_n$ is assigned to cluster k and 0 otherwise.



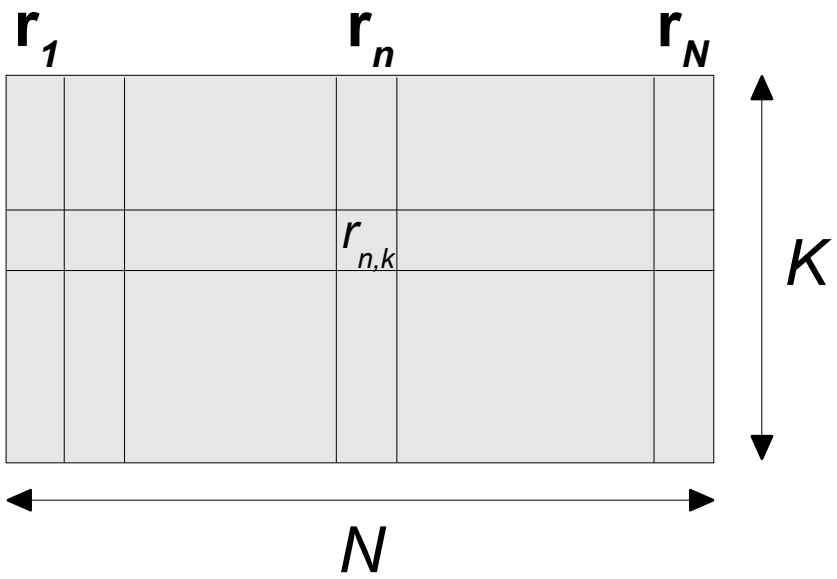


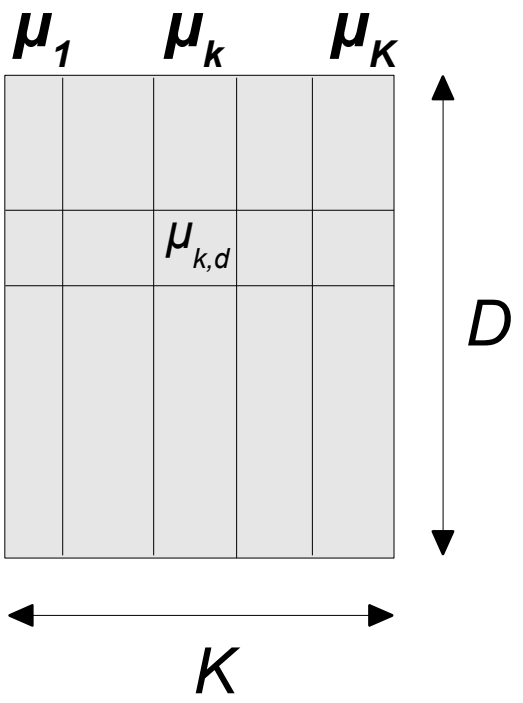
Center points: Each cluster is assigned a center point $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$.

K-means clustering



Cluster assignment: Each data point $\mathbf{x}_n$ is assign one of K clusters, where K is given. The clustering $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$, where $r_{n,k}=1$ if $\mathbf{x}_n$ is assigned to cluster k





Center points: Each cluster is assigned a center point $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$.

K-means clustering (3)

Quality of clustering: The quality of a clustering $\{\mathbf{r}_1,...,\mathbf{r}_N\}$ of data points $\{\mathbf{x}_1,...,\mathbf{x}_N\}$ with center points $\{\boldsymbol{\mu}_1,...,\boldsymbol{\mu}_K\}$ is:

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n,k}\|\mathbf{x}_n - \mu_k\|^2$$

$$\|\mathbf{x}_n - \mu_k\|^2 = \sum_{d=1}^{D}(x_{n,d} - \mu_{k,d})^2$$

The sum of the squares of the distances of each data points to the center point of its assigned cluster.

Objective: Find $\{\mathbf{r}_1,...,\mathbf{r}_N\}$ and $\{\boldsymbol{\mu}_1,...,\boldsymbol{\mu}_K\}$ such that *J* is minimized.

Algorithm

1) **Init:** Select initial center points $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$

2) **Update clustering:** Minimize *J* wrt. clustering $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$ while keeping the center points $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$ fixed.

3) **Update center points:** Minimize *J* wrt. center points $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$ while keeping the clustering $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$ fixed.

   Repeat 2) and 3) until convergence.

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n,k}\|\mathbf{x}_n - \mu_k\|^2$$

The algorithms has similarities with the EM-algorithm.

Algorithm – update clustering

2) Update clustering: Minimize *J* wrt. clustering $\{\mathbf{r}_1,...,\mathbf{r}_N\}$ while keeping the center points $\{\boldsymbol{\mu}_1,...,\boldsymbol{\mu}_K\}$ fixed.

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n,k}\|\mathbf{x}_n - \mu_k\|^2$$

Observe that *J* is a linear function of $\mathbf{r}_n$. We minimize for each *n* independently by setting $r_{n,k} = 1$ for that choice of *k* that minimize the distance $\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$,i.e. we assign data point $\mathbf{x}_n$ to the cluster *k* which has its center point $\boldsymbol{\mu}_k$ nearest to $\mathbf{x}_n$.

$$r_{n,k} = \begin{cases} 1 & \text{if } k = \arg\min_j \|\mathbf{x}_n - \mu_j\|^2 \\ 0 & \text{otherwise} \end{cases}$$

Algorithm – update clustering

2) Update clustering: Minimize J wrt. clustering $\{\mathbf{r}_1,...,\mathbf{r}_N\}$ while keeping the center points $\{\boldsymbol{\mu}_1,...,\boldsymbol{\mu}_K\}$ fixed.

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n,k}\|\mathbf{x}_n - \mu_k\|^2$$

Observe that J is a linear function of $\mathbf{r}_n$. We minimize for each n independently by setting $r_{n,k} = 1$ for that choice of k that minimize the distance $\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$,i.e. we assign data point $\mathbf{x}_n$ to the cluster k which has its center point $\boldsymbol{\mu}_k$ nearest to $\mathbf{x}_n$.

$$r_{n,k} = \begin{cases} 1 & \text{if } k = \arg\min_j \|\mathbf{x}_n - \mu_j\|^2 \\ 0 & \text{otherwise} \end{cases}$$

Takes time O(*NKD*).

Algorithm – update center points

3) Update center points: Minimize J wrt. center points $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$ while keeping the clustering $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$ fixed.

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n,k}\|\mathbf{x}_n - \mu_k\|^2$$

Observe that J is a quadratic function of $\boldsymbol{\mu}_k$. We can minimize for each k independently. This yields:

$$\mu_k = \frac{\sum_{n=1}^{N} r_{n,k}\mathbf{x}_n}{\sum_{n=1}^{N} r_{n,k}}$$

Algorithm – update center points

3) Update center points: Minimize J wrt. center points $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$ while keeping the clustering $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$ fixed.

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n,k}\|\mathbf{x}_n - \mu_k\|^2$$

Observe that J is a quadratic function of $\boldsymbol{\mu}_k$. We can minimize for each k independently. This yields:

$$\mu_k = \frac{\sum_{n=1}^{N} r_{n,k}\mathbf{x}_n}{\sum_{n=1}^{N} r_{n,k}} \qquad \mu_{k,d} = \frac{\sum_{n=1}^{N} r_{n,k}x_{n,d}}{\sum_{n=1}^{N} r_{n,k}}$$

Takes time O(*NKD*).

Algorithm – update center points

3) Update center points: Minimize J wrt. center points $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$ while keeping the clustering $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$ fixed.

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n,k}\|\mathbf{x}_n - \mu_k\|^2$$

Observe that J is a quadratic function of $\boldsymbol{\mu}_k$. We can minimize for each k independently. This yields:

The sum of d'th coordinate of the data points in cluster k

$$\mu_k = \frac{\sum_{n=1}^{N} r_{n,k}\mathbf{x}_n}{\sum_{n=1}^{N} r_{n,k}} \qquad \mu_{k,d} = \frac{\sum_{n=1}^{N} r_{n,k}x_{n,d}}{\sum_{n=1}^{N} r_{n,k}}$$

Number of data points in cluster k

Takes time O(*NKD*).

Algorithm – update center points

3) Update center points: Minimize J wrt. center points $\{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K\}$ while keeping the clustering $\{\mathbf{r}_1,\ldots,\mathbf{r}_N\}$ fixed.

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n,k}\|\mathbf{x}_n - \mu_k\|^2$$

Observe that J is a quadratic function of $\boldsymbol{\mu}_k$. We can minimize for each k independently. This yields:

The sum of d'th coordinate of the data points in cluster k

$$\mu_k = \frac{\sum_{n=1}^{N} r_{n,k}\mathbf{x}_n}{\sum_{n=1}^{N} r_{n,k}}$$

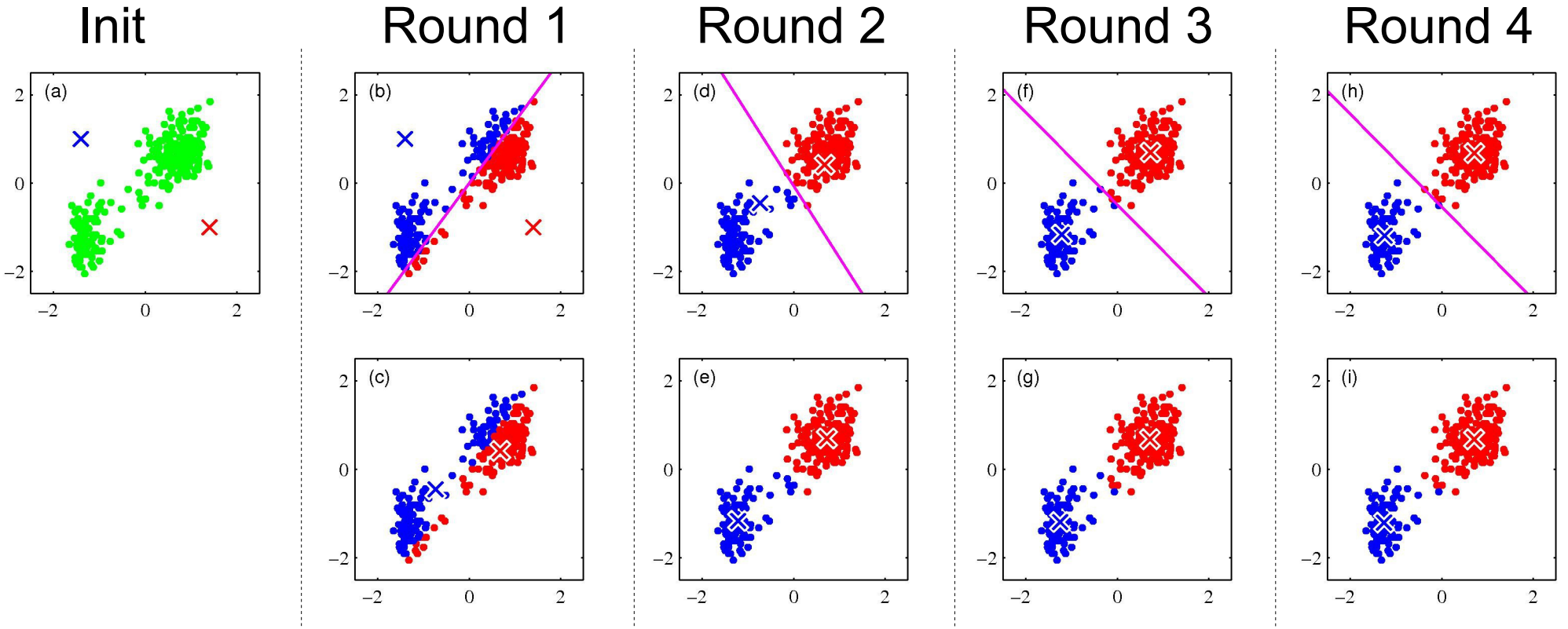
$$\mu_{k,d} = \frac{\sum_{n=1}^{N} r_{n,k}\mathbf{x}_{n,d}}{\sum_{n=1}^{N} r_{n,k}}$$

Mean of the d'th coordinate of the data points in cluster k

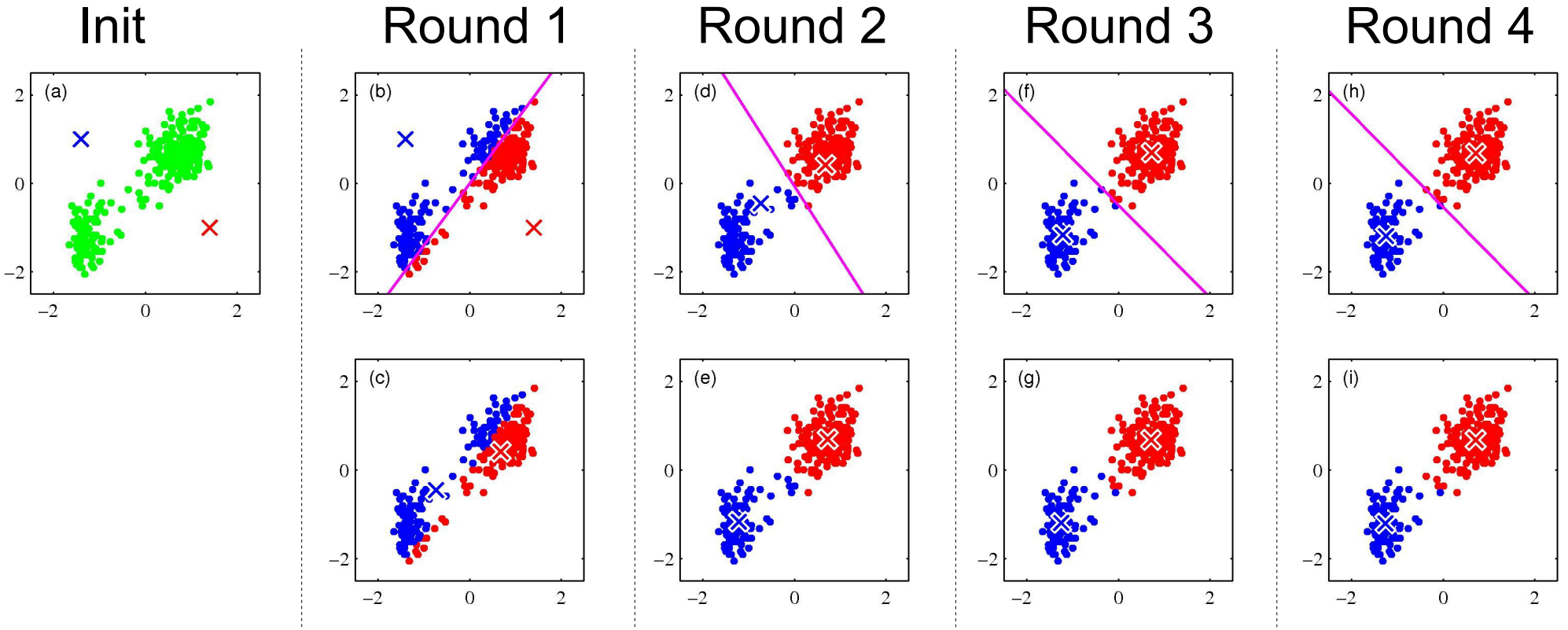
Number of data points in cluster k

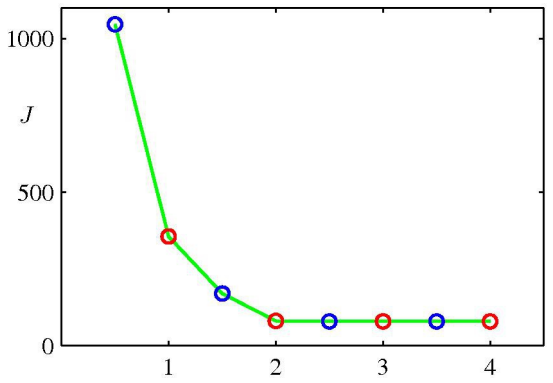
Takes time O(NKD).

Example (N=?, K=2, D=2)



Example (N=?, K=2, D=2)





Extensions

Improve running time: The running time is O(*NKD*) per round. This might be limiting. Use data structures to e.g. speed up the determination of the closest center point (step 3).

Extensions

Improve running time: The running time is O(*NKD*) per round. This might be limiting. Use data structures to e.g. speed up the determination of the closest center point (step 3).

Other dissimilarity measures: Euclidian distance is not applicable to all types of data, so one might want to use another dissimilarity measure V(**x**,**x'**) between data points.

$$J' = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{n,k} \mathcal{V}(\mathbf{x}_n, \mu_k)$$

The algorithm remains the same, but the complexity of step 3 (minimizing *J'* wrt. the center points) might change depending on the dissimilarity measure. To avoid this "problem" one might say that the center point must be one of the data points.

Choosing initial center points

The quality of clustering depends on the choice of initial center pointers. Can you think of examples of 'bad' initial center points?

Choosing initial center points

The quality of clustering depends on the choice of initial center pointers. Can you think of examples of 'bad' initial center points?

- **Simple approach:**
  - Choose *K* random data points as the initial centers
- **Approach from the paper "k-means++: The Advantages of Careful Seeding":**
  1. Choose one center point uniformly at random from among the data points.
  2. For each data point **x**, compute d(**x**), the euclidian distance between **x** and the nearest center point that has already been chosen.
  3. Add one new data point at random as a new center point, using a weighted probability distribution where a data point **x** is chosen with probability proportional to $d(\mathbf{x})^2$.
  4. Repeat Steps 2 and 3 until *K* centers have been chosen.