# The Data Science Cycle

## Unsupervised Learning - Clustering

DataLab

September 21, 2016

# Outline



DataLab
Data Science Community

# Outline



DataLab
Data Science Community

# Supervised Learning vs. Unsupervised Learning

**Supervised learning:**

- Discover patterns in the data that relate data attributes with a target (class) attribute.

**Unsupervised learning:**

The data have no target attribute.

# Supervised Learning vs. Unsupervised Learning

**Supervised learning:**

- Discover patterns in the data that relate data attributes with a target (class) attribute.

**Unsupervised learning:**

The data have no target attribute.

DataLab
Data Science Community

# Outline



DataLab
Data Science Community

# Clustering

## Clustering

It is a technique for finding similarity groups in data, called clusters.

# Clustering

## Clustering

It is a technique for finding similarity groups in data, called clusters.

## Called

An unsupervised learning task as no class values denoting an a priori grouping of the data instances are given, which is the case in supervised learning.

- In fact, association rule mining is also unsupervised

DataLab
Data Science Community

# Clustering

### Clustering

It is a technique for finding similarity groups in data, called clusters.

### Called

An unsupervised learning task as no class values denoting an a priori grouping of the data instances are given, which is the case in supervised learning.

### Due to historical reasons

Clustering is often considered synonymous with unsupervised learning.

- In fact, association rule mining is also unsupervised.

DataLab
Data Science Community

# Clustering

## Clustering

It is a technique for finding similarity groups in data, called clusters.

## Called

An unsupervised learning task as no class values denoting an a priori grouping of the data instances are given, which is the case in supervised learning.

## Due to historical reasons

Clustering is often considered synonymous with unsupervised learning.

- In fact, association rule mining is also unsupervised.

DataLab
Data Science Community

# Outline



DataLab
Data Science Community

# Pattern Recognition

**Definition**

Search for structure in data

# Pattern Recognition

## Definition

Search for structure in data

## Elements of Numerical Pattern Recognition

1. Process Description
   - Feature Nomination, Test Data, Design Data
2. Feature Analysis
   - Preprocessing, Extraction, Selection, ...
3. **Cluster Analysis**
   - Labeling, Validity, ...
4. Classifier Design
   - Classification, Estimation, Prediction, Control, ...

# Pattern Recognition

**Definition**

Search for structure in data

**Elements of Numerical Pattern Recognition**

1. Process Description
   - Feature Nomination, Test Data, Design Data
2. Feature Analysis
   - Preprocessing, Extraction, Selection, ...
3. **Cluster Analysis**
   - Labeling, Validity, ...
4. Classifier Design
   - Classification, Estimation, Prediction, Control, ...

# Pattern Recognition

## Definition

Search for structure in data

## Elements of Numerical Pattern Recognition

1. Process Description
   - Feature Nomination, Test Data, Design Data
2. Feature Analysis
   - Preprocessing, Extraction, Selection, ...
3. **Cluster Analysis**
   - Labeling, Validity, ...
4. Classifier Design
   - Classification, Estimation, Prediction, Control, ...

# Pattern Recognition

## Definition

Search for structure in data

## Elements of Numerical Pattern Recognition

1. Process Description
   - Feature Nomination, Test Data, Design Data
2. Feature Analysis
   - Preprocessing, Extraction, Selection, ...
3. Cluster Analysis
   - Labeling, Validity, ...
4. Classifier Design
   - Classification, Estimation, Prediction, Control, ...

# Pattern Recognition

## Definition

Search for structure in data

## Elements of Numerical Pattern Recognition

1. Process Description
    - Feature Nomination, Test Data, Design Data
2. Feature Analysis
    - Preprocessing, Extraction, Selection, . . .
3. **Cluster Analysis**
    - Labeling, Validity, . . .
4. Classifier Design
    - Classification, Estimation, Prediction, Control, . . .

# Pattern Recognition

## Definition

Search for structure in data

## Elements of Numerical Pattern Recognition

1. Process Description
   - Feature Nomination, Test Data, Design Data
2. Feature Analysis
   - Preprocessing, Extraction, Selection, . . .
3. **Cluster Analysis**
   - Labeling, Validity, . . .
4. Classifier Design
   - Classification, Estimation, Prediction, Control, . . .

# Pattern Recognition

**Definition**

Search for structure in data

**Elements of Numerical Pattern Recognition**

1. Process Description
   - Feature Nomination, Test Data, Design Data
2. Feature Analysis
   - Preprocessing, Extraction, Selection, . . .
3. **Cluster Analysis**
   - Labeling, Validity, . . .
4. Classifier Design
   - Classification, Estimation, Prediction, Control, . . .

# Pattern Recognition

**Definition**

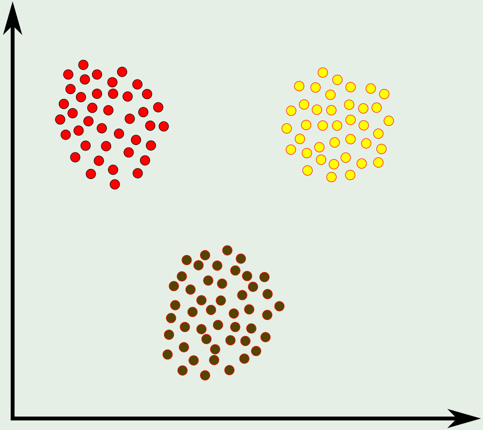
Search for structure in data

**Elements of Numerical Pattern Recognition**

1. Process Description
   - Feature Nomination, Test Data, Design Data
2. Feature Analysis
   - Preprocessing, Extraction, Selection, ...
3. **Cluster Analysis**
   - Labeling, Validity, ...
4. Classifier Design
   - Classification, Estimation, Prediction, Control, ...

# An illustration

The data set has three natural groups of data points, i.e., 3 natural clusters.



# Examples

**Example 1**

Groups people of similar sizes together to make “small”, “medium” and “large” T-Shirts.

**Example 2**

In marketing, segment customers according to their similarities.

# Examples

**Example 1**

Groups people of similar sizes together to make "small", "medium" and "large" T-Shirts.

**Example 2**

In marketing, segment customers according to their similarities.

# How we create this classes?

**For this, we use the following concept**

Clustering!!!

**Basically**

We want to "reveal" the organization of patterns into "sensible" clusters (groups).

**Actually**

Clustering is one of the most primitive mental activities of humans, used to handle the huge amount of information they receive every day.

# How we create this classes?

**For this, we use the following concept**

Clustering!!!

**Basically**

We want to "reveal" the organization of patterns into "sensible" clusters (groups).

**Actually**

Clustering is one of the most primitive mental activities of humans, used to handle the huge amount of information they receive every day.

# How we create this classes?

**For this, we use the following concept**

Clustering!!!

**Basically**

We want to "reveal" the organization of patterns into "sensible" clusters (groups).

**Actually**

Clustering is one of the most primitive mental activities of humans, used to handle the huge amount of information they receive every day.

# Outline



DataLab
Data Science Community

# Aspects of clustering

## A clustering algorithm - They are Many!!!

- Partition clustering.
- Hierarchical clustering.
- etc.

DataLab
Data Science Community

# Aspects of clustering

## A clustering algorithm - They are Many!!!

- Partition clustering.
- Hierarchical clustering.
- etc.

## Based in a function

A distance (similarity, or dissimilarity) function.

- Intra-clusters distance → minimized.

# Aspects of clustering

**A clustering algorithm - They are Many!!!**

- Partition clustering.
- Hierarchical clustering.
- etc.

**Based in a function**

A distance (similarity, or dissimilarity) function.

**Clustering quality**

- Inter-clusters distance → maximized.
- Intra-clusters distance → minimized.

DataLab
Data Science Community

# Aspects of clustering

## A clustering algorithm - They are Many!!!

- Partition clustering.
- Hierarchical clustering.
- etc.

## Based in a function

A distance (similarity, or dissimilarity) function.

## Clustering quality

- Inter-clusters distance $\rightarrow$ maximized.
- Intra-clusters distance $\rightarrow$ minimized.

DataLab
Data Science Community

# Aspects of clustering

## A clustering algorithm - They are Many!!!

- Partition clustering.
- Hierarchical clustering.
- etc.

## Based in a function

A distance (similarity, or dissimilarity) function.

## Clustering quality

- Inter-clusters distance $\rightarrow$ maximized.
- Intra-clusters distance $\rightarrow$ minimized.

# Aspects of clustering

### A clustering algorithm - They are Many!!!

- Partition clustering.
- Hierarchical clustering.
- etc.

### Based in a function

A distance (similarity, or dissimilarity) function.

### Clustering quality

- Inter-clusters distance $\rightarrow$ maximized.
- Intra-clusters distance $\rightarrow$ minimized.

# Outline



DataLab
Data Science Community

# K-Means - Stuart Lloyd(Circa 1957)

**History**

Invented by Stuart Loyd in Bell Labs to obtain the best quantization in a signal data set.

**Something Notable**

The paper was published until 1982

**Basically given $N$ vectors $x_1, \ldots, x_N \in \mathbb{R}^d$**

It tries to find $k$ points $\mu_1, \ldots, \mu_k \in \mathbb{R}^d$ that minimize the expression (i.e. a partition $S$ of the vector points):

$$\sum_{k=1}^{N} \sum_{i: x_i \in C_k} \|x_i - \mu_k\|^2 = \sum_{k=1}^{N} \sum_{i: x_i \in C_k} (x_i - \mu_k)^T (x_i - \mu_k)$$

# K-Means - Stuart Lloyd(Circa 1957)

## History

Invented by Stuart Loyd in Bell Labs to obtain the best quantization in a signal data set.

## Something Notable

The paper was published until 1982

## Basically given $N$ vectors $x_1, ..., x_N \in \mathbb{R}^d$

It tries to find $k$ points $\mu_1, ..., \mu_k \in \mathbb{R}^d$ that minimize the expression (i.e. a partition $S$ of the vector points):

$$\sum_{k=1}^{N} \sum_{i:x_i \in C_k} \|x_i - \mu_k\|^2 = \sum_{k=1}^{N} \sum_{i:x_i \in C_k} (x_i - \mu_k)^T (x_i - \mu_k)$$

# K-Means - Stuart Lloyd(Circa 1957)

## History

Invented by Stuart Loyd in Bell Labs to obtain the best quantization in a signal data set.

## Something Notable

The paper was published until 1982

## Basically given $N$ vectors $\boldsymbol{x}_1, ..., \boldsymbol{x}_N \in \mathbb{R}^d$

It tries to find $k$ points $\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_k \in \mathbb{R}^d$ that minimize the expression (i.e. a partition $S$ of the vector points):

$$\sum_{k=1}^{N} \sum_{i:\boldsymbol{x}_i \in C_k} \left\| \boldsymbol{x}_i - \boldsymbol{\mu}_k \right\|^2 = \sum_{k=1}^{N} \sum_{i:\boldsymbol{x}_i \in C_k} \left( \boldsymbol{x}_i - \boldsymbol{\mu}_k \right)^T \left( \boldsymbol{x}_i - \boldsymbol{\mu}_k \right)$$

# $K$-means clustering

## $K$-means

It is a partitional clustering algorithm.

# $K$-means clustering

## $K$-means

It is a partitional clustering algorithm.

## Definition

Let the set of data points (or instances) $D$ be $\{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$ where $\mathbf{x}_i = (x_{i1}, \cdots, x_{ir})^T$:

- The $K$-means algorithm partitions the given data into $K$ clusters.
- Each cluster has a cluster center, called centroid.
- $K$ is specified by the user.

# $K$-means clustering

## $K$-means

It is a partitional clustering algorithm.

## Definition

Let the set of data points (or instances) $D$ be $\{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$ where $\mathbf{x}_i = (x_{i1}, \cdots, x_{ir})^T$:

- The $K$-means algorithm partitions the given data into $K$ clusters.
- Each cluster has a cluster center, called centroid.
- $K$ is specified by the user.

DataLab
Data Science Community

# $K$-means clustering

**$K$-means**

It is a partitional clustering algorithm.

**Definition**

Let the set of data points (or instances) $D$ be $\{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$ where $\mathbf{x}_i = (x_{i1}, \cdots, x_{ir})^T$:

- The $K$-means algorithm partitions the given data into $K$ clusters.
- Each cluster has a cluster center, called centroid.
- $K$ is specified by the user.

DataLab
Data Science Community

# $K$-means clustering

**$K$-means**

It is a partitional clustering algorithm.

**Definition**

Let the set of data points (or instances) $D$ be $\{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$ where $\mathbf{x}_i = (x_{i1}, \cdots, x_{ir})^T$:

- The $K$-means algorithm partitions the given data into $K$ clusters.
- Each cluster has a cluster center, called centroid.
- $K$ is specified by the user.

# $K$-means algorithm

## The $K$-means algorithm works as follows

Given $k$ as the possible number of cluster:

1. Randomly choose $K$ data points (seeds) to be the initial centroids, cluster centers,
   - $\{\mathbf{v}_1, \cdots, \mathbf{v}_k\}$
2. Assign each data point to the closest centroid
   - $c_i = \arg\min_j \{dist(\mathbf{x}_i - \mathbf{v}_j)\}$
3. Re-compute the centroids using the current cluster memberships.
   - $\mathbf{v}_j = \frac{\sum_{i=1}^{n} I(c_i = j)\mathbf{x}_i}{\sum_{i=1}^{n} I(c_i = j)}$
4. If a convergence criterion is not met, go to 2.

# $K$-means algorithm

## The $K$-means algorithm works as follows

Given $k$ as the possible number of cluster:

1. Randomly choose $K$ data points (seeds) to be the initial centroids, cluster centers,
   - $\{\mathbf{v}_1, \cdots, \mathbf{v}_k\}$
2. Assign each data point to the closest centroid
   - $c_i = \arg\min_j \{dist(\mathbf{x}_i - \mathbf{v}_j)\}$
3. Re-compute the centroids using the current cluster memberships.
   - $\mathbf{v}_j = \dfrac{\sum_{i=1}^{n} I(c_i = j)\mathbf{x}_i}{\sum_{i=1}^{n} I(c_i = j)}$
4. If a convergence criterion is not met, go to 2.

# $K$-means algorithm

## The $K$-means algorithm works as follows

Given $k$ as the possible number of cluster:

1. Randomly choose $K$ data points (seeds) to be the initial centroids, cluster centers,
   - $\{\mathbf{v}_1, \cdots, \mathbf{v}_k\}$
2. Assign each data point to the closest centroid
   - $c_i = \arg\min_j \{dist(\mathbf{x}_i - \mathbf{v}_j)\}$
3. Re-compute the centroids using the current cluster memberships.
   - $$\mathbf{v}_j = \frac{\sum_{i=1}^{n} I(c_i = j)\mathbf{x}_i}{\sum_{i=1}^{n} I(c_i = j)}$$
4. If a convergence criterion is not met, go to 2.

# $K$-means algorithm

## The $K$-means algorithm works as follows

Given $k$ as the possible number of cluster:

1. Randomly choose $K$ data points (seeds) to be the initial centroids, cluster centers,
   - $\{\mathbf{v}_1, \cdots, \mathbf{v}_k\}$
2. Assign each data point to the closest centroid
   - $c_i = \arg\min_j \{dist(\mathbf{x}_i - \mathbf{v}_j)\}$
3. Re-compute the centroids using the current cluster memberships.
   - $\mathbf{v}_j = \dfrac{\sum_{i=1}^{n} I(c_i = j)\mathbf{x}_i}{\sum_{i=1}^{n} I(c_i = j)}$
4. If a convergence criterion is not met, go to 2.

# $K$-means algorithm

## The $K$-means algorithm works as follows

Given $k$ as the possible number of cluster:

1. Randomly choose $K$ data points (seeds) to be the initial centroids, cluster centers,
   - $\{\mathbf{v}_1, \cdots, \mathbf{v}_k\}$
2. Assign each data point to the closest centroid
   - $c_i = \arg\min_j \{dist(\mathbf{x}_i - \mathbf{v}_j)\}$
3. Re-compute the centroids using the current cluster memberships.
   - $\mathbf{v}_j = \dfrac{\displaystyle\sum_{i=1}^{n} I(c_i = j)\mathbf{x}_i}{\displaystyle\sum_{i=1}^{n} I(c_i = j)}$
4. If a convergence criterion is not met, go to 2.

# What is the code trying to do?

## It is trying to find a partition $S$

$K$-means tries to find a partition $S$ such that it minimizes the cost function:

$$\min_{S} \sum_{k=1}^{N} \sum_{i:\boldsymbol{x}_i \in C_k} \left(\boldsymbol{x}_i - \boldsymbol{\mu}_k\right)^T \left(\boldsymbol{x}_i - \boldsymbol{\mu}_k\right) \tag{1}$$

## Where $\mu_k$ is the centroid for cluster $C_k$

$$\mu_k = \frac{1}{N_k} \sum_{i:\boldsymbol{x}_i \in C_k} x_i \tag{2}$$

Where $N_k$ is the number of samples in the cluster $C_k$.

# What is the code trying to do?

## It is trying to find a partition $S$

$K$-means tries to find a partition $S$ such that it minimizes the cost function:

$$\min_{S} \sum_{k=1}^{N} \sum_{i:\boldsymbol{x}_i \in C_k} \left(\boldsymbol{x}_i - \boldsymbol{\mu}_k\right)^T \left(\boldsymbol{x}_i - \boldsymbol{\mu}_k\right) \tag{1}$$

## Where $\mu_k$ is the centroid for cluster $C_k$

$$\mu_k = \frac{1}{N_k} \sum_{i:\boldsymbol{x}_i \in C_k} x_i \tag{2}$$

Where $N_k$ is the number of samples in the cluster $C_k$.

# What is the code trying to do?

## It is trying to find a partition $S$

$K$-means tries to find a partition $S$ such that it minimizes the cost function:

$$\min_{S} \sum_{k=1}^{N} \sum_{i:\boldsymbol{x}_i \in C_k} (\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T (\boldsymbol{x}_i - \boldsymbol{\mu}_k) \tag{1}$$

## Where $\mu_k$ is the centroid for cluster $C_k$

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{i:\boldsymbol{x}_i \in C_k} \boldsymbol{x}_i \tag{2}$$

Where $N_k$ is the number of samples in the cluster $C_k$.

# Outline

# What Stopping/convergence criterion should we use?

**First**

No (or minimum) re-assignments of data points to different clusters.

# What Stopping/convergence criterion should we use?

## First

No (or minimum) re-assignments of data points to different clusters.

## Second

No (or minimum) change of centroids.

- $C_j$ is cluster $j$.
- $m_j$ is the centroid of cluster $C_j$.

$$SSE = \sum_{j=1}^{k} \sum_{x \in C_j} dist(x, m_j)^2$$

# What Stopping/convergence criterion should we use?

**First**

No (or minimum) re-assignments of data points to different clusters.

**Second**

No (or minimum) change of centroids.

**Third**

Minimum decrease in the sum of squared error (SSE),

- $C_k$ is cluster $k$.
- $\mathbf{v}_k$ is the centroid of cluster $C_k$.

$$SSE = \sum_{k=1}^{K} \sum_{x \in C_k} dist\left(\mathbf{x}, \mathbf{v}_k\right)^2$$

# What Stopping/convergence criterion should we use?

**First**

No (or minimum) re-assignments of data points to different clusters.

**Second**

No (or minimum) change of centroids.

**Third**

Minimum decrease in the sum of squared error (SSE),

- $C_k$ is cluster $k$.
- $\mathbf{v}_k$ is the centroid of cluster $C_k$.

$$SSE = \sum_{k=1}^{K} \sum_{x \in C_k} dist\left(\mathbf{x}, \mathbf{v}_k\right)^2$$

# What Stopping/convergence criterion should we use?

## First

No (or minimum) re-assignments of data points to different clusters.

## Second

No (or minimum) change of centroids.

## Third

Minimum decrease in the sum of squared error (SSE),

- $C_k$ is cluster $k$.
- $\mathbf{v}_k$ is the centroid of cluster $C_k$.

$$SSE = \sum_{k=1}^{K} \sum_{x \in c_k} dist\left(\mathbf{x}, \mathbf{v}_k\right)^2$$

# Outline

1. Supervised Learning vs. Unsupervised Learning
   - Supervised vs Unsupervised
   - Clustering
   - Pattern Recognition
   - Aspect of Clustering

2. K-Means Clustering
   - K-Means Clustering
   - Convergence Criterion
   - The Distance Function
   - Example
   - Properties of K-Means

DataLab
Data Science Community

# The distance function

Actually, we have the following distance functions:

**Euclidean**

$$dixt(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}|| = \sqrt{(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y})}$$

**Manhattan**

$$dixt(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}||_1 = \sum_{i=1}^{n} |x_i - y_i|$$

**Mahalanobis**

$$dixt(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}||_A = \sqrt{(\mathbf{x}-\mathbf{y})^T A(\mathbf{x}-\mathbf{y})}$$

# The distance function

Actually, we have the following distance functions:

**Euclidean**

$$dixt(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}|| = \sqrt{(\mathbf{x} - \mathbf{y})^T (\mathbf{x} - \mathbf{y})}$$

**Manhattan**

$$dixt(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_1 = \sum_{i=1}^{n} |x_i - y_i|$$

**Mahalanobis**

$$dixt(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_A = \sqrt{(\mathbf{x} - \mathbf{y})^T A (\mathbf{x} - \mathbf{y})}$$

# The distance function

Actually, we have the following distance functions:

**Euclidean**

$$dixt(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}|| = \sqrt{(\mathbf{x} - \mathbf{y})^T (\mathbf{x} - \mathbf{y})}$$

**Manhattan**

$$dixt(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_1 = \sum_{i=1}^{n} |x_i - y_i|$$

**Mahalanobis**

$$dixt(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_A = \sqrt{(\mathbf{x} - \mathbf{y})^T A (\mathbf{x} - \mathbf{y})}$$

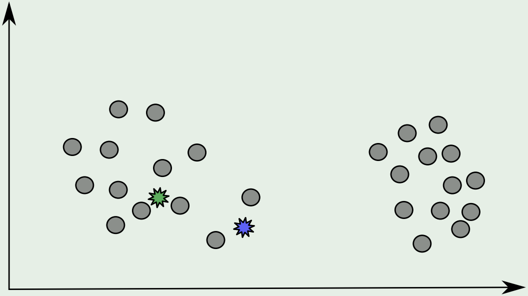
# Outline



DataLab
Data Science Community

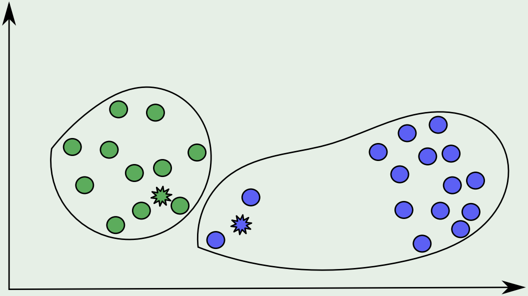
# An example

## Dropping two possible centroids



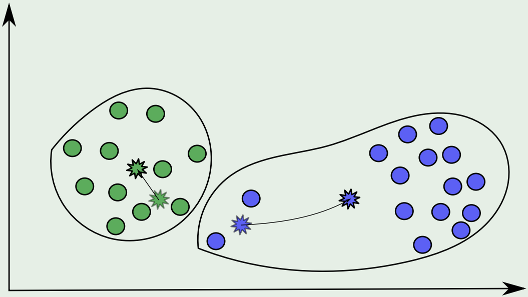
# An example

## Calculate the memberships



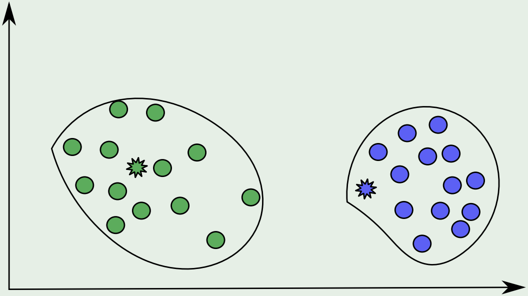
# An example

## We re-calculate centroids



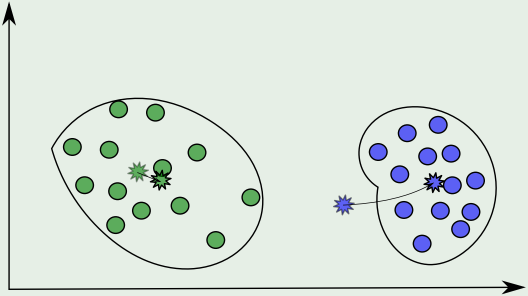
# An example

## We re-calculate memberships



# An example

## We re-calculate centroids and keep going



# Outline



DataLab
Data Science Community

# Strengths of $K$-means

## Strengths

- Simple: easy to understand and to implement
- Efficient: Time complexity: $O(tKN)$, where $N$ is the number of data points, $K$ is the number of clusters, and $t$ is the number of iterations.
- Since both $K$ and $t$ are small. $K$-means is considered a linear algorithm.

# Strengths of $K$-means

## Strengths

- Simple: easy to understand and to implement
- Efficient: Time complexity: $O(tKN)$, where $N$ is the number of data points, $K$ is the number of clusters, and $t$ is the number of iterations.
- Since both $K$ and $t$ are small. $K$-means is considered a linear algorithm.

## Popularity

$K$-means is the most popular clustering algorithm.

# Strengths of $K$-means

## Strengths

- Simple: easy to understand and to implement
- Efficient: Time complexity: $O(tKN)$, where $N$ is the number of data points, $K$ is the number of clusters, and $t$ is the number of iterations.
- Since both $K$ and $t$ are small. $K$-means is considered a linear algorithm.

## Popularity

$K$-means is the most popular clustering algorithm.

## Note that

It terminates at a local optimum if SSE is used. The global optimum is hard to find due to complexity.

# Strengths of $K$-means

## Strengths

- Simple: easy to understand and to implement
- Efficient: Time complexity: $O(tKN)$, where $N$ is the number of data points, $K$ is the number of clusters, and $t$ is the number of iterations.
- Since both $K$ and $t$ are small. $K$-means is considered a linear algorithm.

## Popularity

$K$-means is the most popular clustering algorithm.

## Note that

It terminates at a local optimum if SSE is used. The global optimum is hard to find due to complexity.

# Strengths of $K$-means

## Strengths

- Simple: easy to understand and to implement
- Efficient: Time complexity: $O(tKN)$, where $N$ is the number of data points, $K$ is the number of clusters, and $t$ is the number of iterations.
- Since both $K$ and $t$ are small. $K$-means is considered a linear algorithm.

## Popularity

$K$-means is the most popular clustering algorithm.

## Note that

It terminates at a local optimum if SSE is used. The global optimum is hard to find due to complexity.

DataLab
Data Science Community

# Weaknesses of $K$-means

**Important**

The algorithm is only applicable if the mean is defined.

- For categorical data, $K$-mode - the centroid is represented by most frequent values.

# Weaknesses of $K$-means

**Important**

The algorithm is only applicable if the mean is defined.

- For categorical data, $K$-mode - the centroid is represented by most frequent values.

**In addition**

The user needs to specify $K$.

- Outliers are data points that are very far away from other data points.
- Outliers could be errors in the data recording or some special data points with very different values.

# Weaknesses of $K$-means

**Important**

The algorithm is only applicable if the mean is defined.

- For categorical data, $K$-mode - the centroid is represented by most frequent values.

**In addition**

The user needs to specify $K$.

**Outliers**

The algorithm is sensitive to outliers.

- Outliers are data points that are very far away from other data points.
- Outliers could be errors in the data recording or some special data points with very different values.

# Weaknesses of $K$-means

## Important

The algorithm is only applicable if the mean is defined.

- For categorical data, $K$-mode - the centroid is represented by most frequent values.

## In addition

The user needs to specify $K$.

## Outliers

The algorithm is sensitive to outliers.

- Outliers are data points that are very far away from other data points.
- Outliers could be errors in the data recording or some special data points with very different values.

# Weaknesses of $K$-means

## Important

The algorithm is only applicable if the mean is defined.

- For categorical data, $K$-mode - the centroid is represented by most frequent values.

## In addition

The user needs to specify $K$.

## Outliers

The algorithm is sensitive to outliers.

- Outliers are data points that are very far away from other data points.
- Outliers could be errors in the data recording or some special data points with very different values.

# Weaknesses of $K$-means

## Important

The algorithm is only applicable if the mean is defined.

- For categorical data, $K$-mode - the centroid is represented by most frequent values.

## In addition

The user needs to specify $K$.

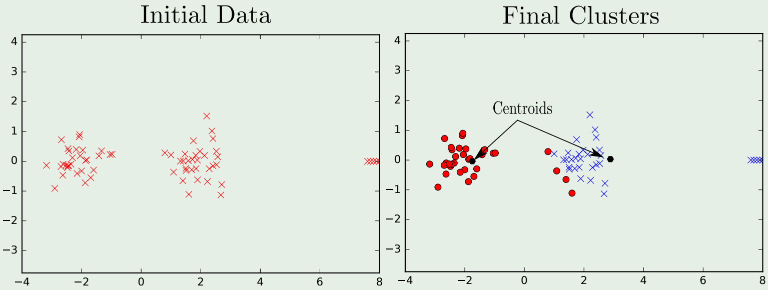
## Outliers

The algorithm is sensitive to outliers.

- Outliers are data points that are very far away from other data points.
- Outliers could be errors in the data recording or some special data points with very different values.

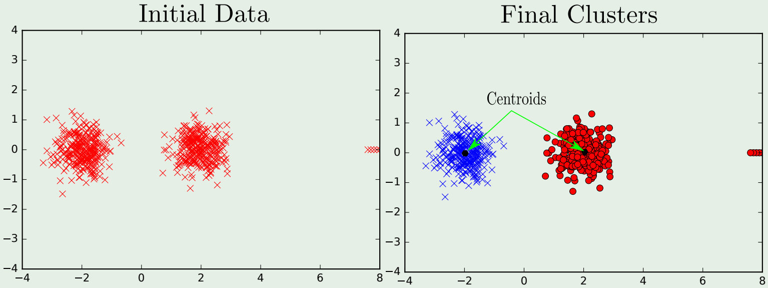
# Weaknesses of $K$-means: Problems with outliers

## A series of outliers



# Weaknesses of $K$-means: Problems with outliers

## Nevertheless, if you have more dense clusters



# Weaknesses of $K$-means: How to deal with outliers

**One method**

To remove some data points in the clustering process that are much further away from the centroids than other data points.

- To be safe, we may want to monitor these possible outliers over a few iterations and then decide to remove them.

DataLab
Data Science Community

# Weaknesses of $K$-means: How to deal with outliers

**One method**

To remove some data points in the clustering process that are much further away from the centroids than other data points.

- To be safe, we may want to monitor these possible outliers over a few iterations and then decide to remove them.

**Another method**

To perform random sampling.

- Since in sampling we only choose a small subset of the data points, the chance of selecting an outlier is very small.
- Assign the rest of the data points to the clusters by distance or similarity comparison, or classification.

# Weaknesses of $K$-means: How to deal with outliers

## One method

To remove some data points in the clustering process that are much further away from the centroids than other data points.

- To be safe, we may want to monitor these possible outliers over a few iterations and then decide to remove them.

## Another method

To perform random sampling.

- Since in sampling we only choose a small subset of the data points, the chance of selecting an outlier is very small.
- Assign the rest of the data points to the clusters by distance or similarity comparison, or classification.

# Weaknesses of $K$-means: How to deal with outliers

## One method

To remove some data points in the clustering process that are much further away from the centroids than other data points.

- To be safe, we may want to monitor these possible outliers over a few iterations and then decide to remove them.

## Another method

To perform random sampling.

- Since in sampling we only choose a small subset of the data points, the chance of selecting an outlier is very small.
- Assign the rest of the data points to the clusters by distance or similarity comparison, or classification.

# Weaknesses of $K$-means: How to deal with outliers

## One method

To remove some data points in the clustering process that are much further away from the centroids than other data points.

- To be safe, we may want to monitor these possible outliers over a few iterations and then decide to remove them.

## Another method

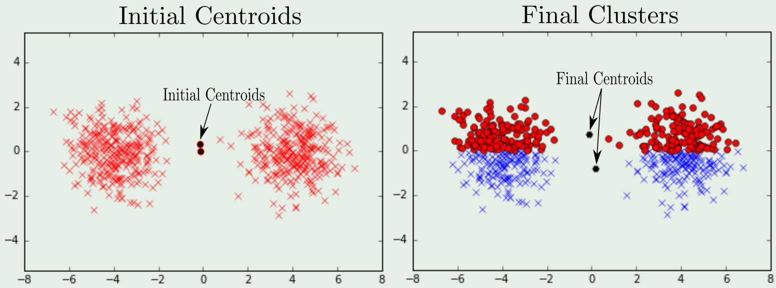
To perform random sampling.

- Since in sampling we only choose a small subset of the data points, the chance of selecting an outlier is very small.
- Assign the rest of the data points to the clusters by distance or similarity comparison, or classification.

DataLab
Data Science Community

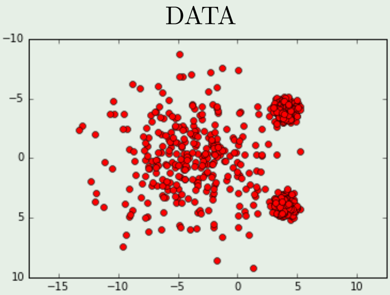
# Weaknesses of $K$-means (cont...)

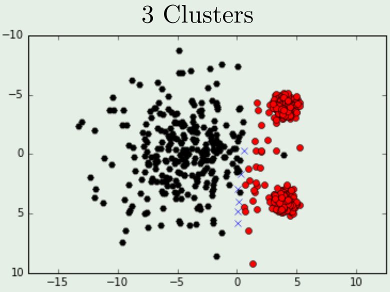
## The algorithm is sensitive to initial seeds



# Weaknesses of $K$-means : Different Densities

## We have three cluster nevertheless

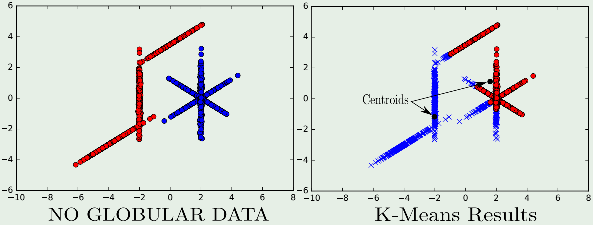



DataLab
Data Science Community

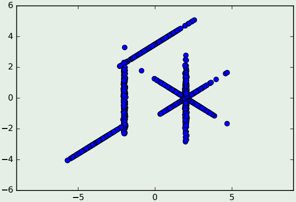
# Weaknesses of $K$-means: Non-globular Shapes

**Here, we notice that $K$-means may only detect globular shapes**

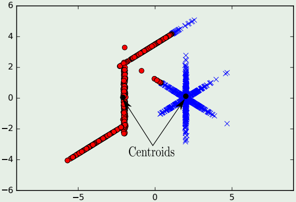


# Weaknesses of $K$-means: Non-globular Shapes

## However, it sometimes work better than expected




NO GLOBULAR DATA




K-Means Results