NIO

Exercise 11: RBF-Nets

Duc Duy Pham, M.Sc.

Raum: BC 410
Tel.: 0203-379-3734
Email: duc.duy.pham@uni-due.de

Contents

- Revision CNN
- Revision of Lecture (RBF Nets)

Contents

- **Revision CNN**
- Revision of Lecture (RBF Nets)

Revision CNN

- Which layers does a MLP have?

Revision CNN

- Which layers does a MLP have?
  1) (Input layer)
  2) hidden layers (fully connected)
  3) Backprop layer
  4) output layer
  5) Gradient layer

Revision CNN

- Which layers does a MLP have?
  1) (Input layer)
  2) hidden layers (fully connected)
  3) Backprop layer
  4) output layer
  5) Gradient layer

A: 1,2,4

C: 1,2,4,5

B: 1,2,3,5

D: all

Revision CNN

- Which layers does a MLP have?

1) (Input layer)
2) hidden layers (fully connected)
3) Backprop layer
4) output layer
5) Gradient layer

A: 1,2,4

C: 1,2,4,5

B: 1,2,3,5

D: all

Revision CNN

- Which layers does a CNN not have?

Revision CNN

- Which layers does a CNN not have?
  1) (Input Layer)
  2) Radial Layer
  3) Convolutional layers
  4) Activation layers
  5) (Max-)Pooling layers
  6) Fully connected layers
  7) One output layer

Revision CNN

- Which layers does a CNN not have?
  1) (Input Layer)
  2) Radial Layer
  3) Convolutional layers
  4) Activation layers
  5) (Max-)Pooling layers
  6) Fully connected layers
  7) One output layer

A: 2

C: 5

B: 3

D: 6

Revision CNN

- Which layers does a CNN not have?
  1) (Input Layer)
  2) Radial Layer
  3) Convolutional layers
  4) Activation layers
  5) (Max-)Pooling layers
  6) Fully connected layers
  7) One output layer

A: 2

C: 5

B: 3

D: 6

Revision CNN

- Purpose of Convolutional Layer?

Revision CNN

- Purpose of Convolutional Layer?
  - Feature extraction (by means of convolutions with kernels)

Revision CNN

- Purpose of Convolutional Layer?
  - Feature extraction (by means of convolutions with kernels)
  - Kernels are learned!!! (not handcrafted)

Revision CNN

- Learnable params of Convolutional layer?

Revision CNN

- Learnable params of Convolutional layer?
  1) Kernels
  2) Number of kernels
  3) Kernel size
  4) Stride
  5) Padding
  6) (activation)

Revision CNN

- Learnable params of Convolutional layer?
  1) Kernels
  2) Number of kernels
  3) Kernel size
  4) Stride
  5) Padding
  6) (activation)

A: 1,2,3

C: 1

B: 1,4,5,6

D: all

Revision CNN

- Learnable params of Convolutional layer?
  1) Kernels
  2) Number of kernels
  3) Kernel size
  4) Stride
  5) Padding
  6) (activation)

A: 1,2,3

C: 1

B: 1,4,5,6

D: all

Revision CNN

- Purpose of Pooling Layer?

Revision CNN

- Purpose of Pooling Layer?
  - Translation invariance!

Revision CNN

- Purpose of Pooling Layer?
  - Translation invariance!
  - (Memory reduction)

Revision CNN

- Learnable params of Pooling Layer?

Revision CNN

- Learnable params of Pooling Layer?
  1) None
  2) Pool size
  3) (stride)

Revision CNN

- Learnable params of Pooling Layer?
  1) None
  2) Pool size
  3) (stride)

A: 1

C: 3

B: 2

D: all

Revision CNN

- Learnable params of Pooling Layer?
  1) None
  2) Pool size
  3) (stride)

A: 1

C: 3

B: 2

D: all

Revision CNN

- Purpose of Fully Connected Layer?

Revision CNN

- Purpose of Fully Connected Layer?
  1) Classification
  2) Regression
  3) Convolution
  4) Translation

Revision CNN

- Purpose of Fully Connected Layer?
  1) Classification
  2) Regression
  3) Convolution
  4) Translation

A: 1,2	C: 1,4
B: 1,3	D: all

Revision CNN

- Purpose of Fully Connected Layer?

1) Classification
2) Regression
3) Convolution
4) Translation

A: 1,2

C: 1,4

B: 1,3

D: all

Revision CNN

- Learnable Params of Fully Connected Layer?

Revision CNN

- Learnable Params of Fully Connected Layer?
  1) Weights between layers (same as MLP)
  2) Number of nodes in layer
  3) Activation function

Revision CNN

- Learnable Params of Fully Connected Layer?
  1) Weights between layers (same as MLP)
  2) Number of nodes in layer
  3) Activation function

A: 1

C: 3

B: 2

D: all

Revision CNN

- Learnable Params of Fully Connected Layer?
  1) Weights between layers (same as MLP)
  2) Number of nodes in layer
  3) Activation function

A: 1

C: 3

B: 2

D: all

Contents

- Revision CNN
- Revision of Lecture (RBF Nets)

Revision of Lecture

- Which layers does a RBF-Net have?

Revision of Lecture

- Which layers does a RBF-Net have?
  - (1 Input Layer)
  - 1 Hidden (RBF) Layer
  - 1 Output Layer

Revision of Lecture

- What are differences between MLPs and RBF-Nets?

Revision of Lecture

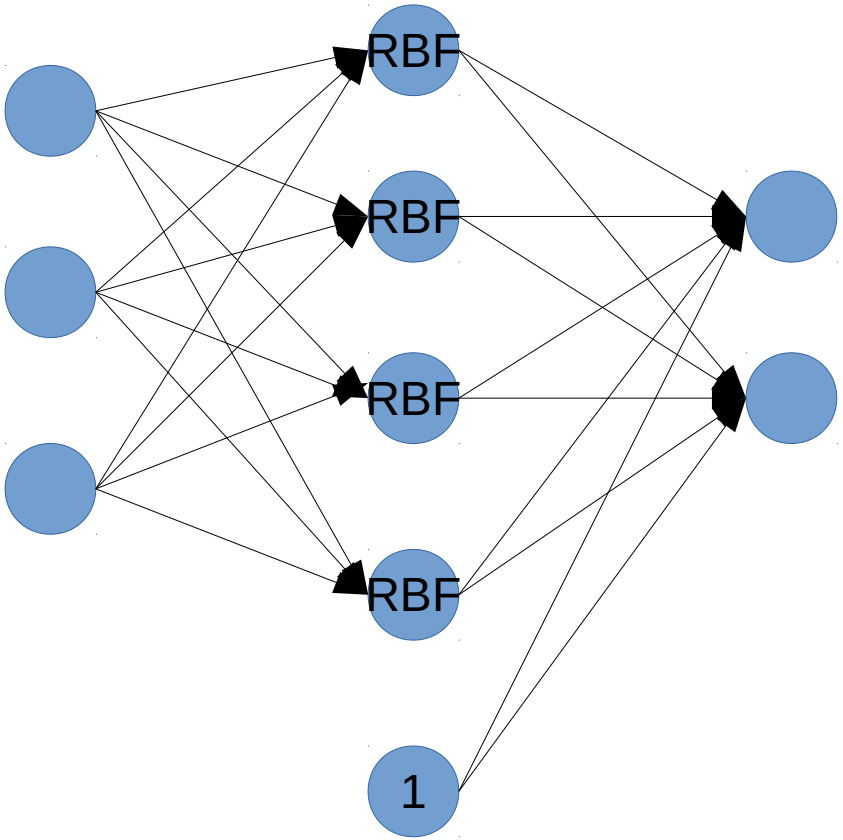
- What are differences between MLPs and RBF-Nets?
  - RBF-Net is a 2 Layer Network, MLPs can have an arbitrary number of layers

Revision of Lecture

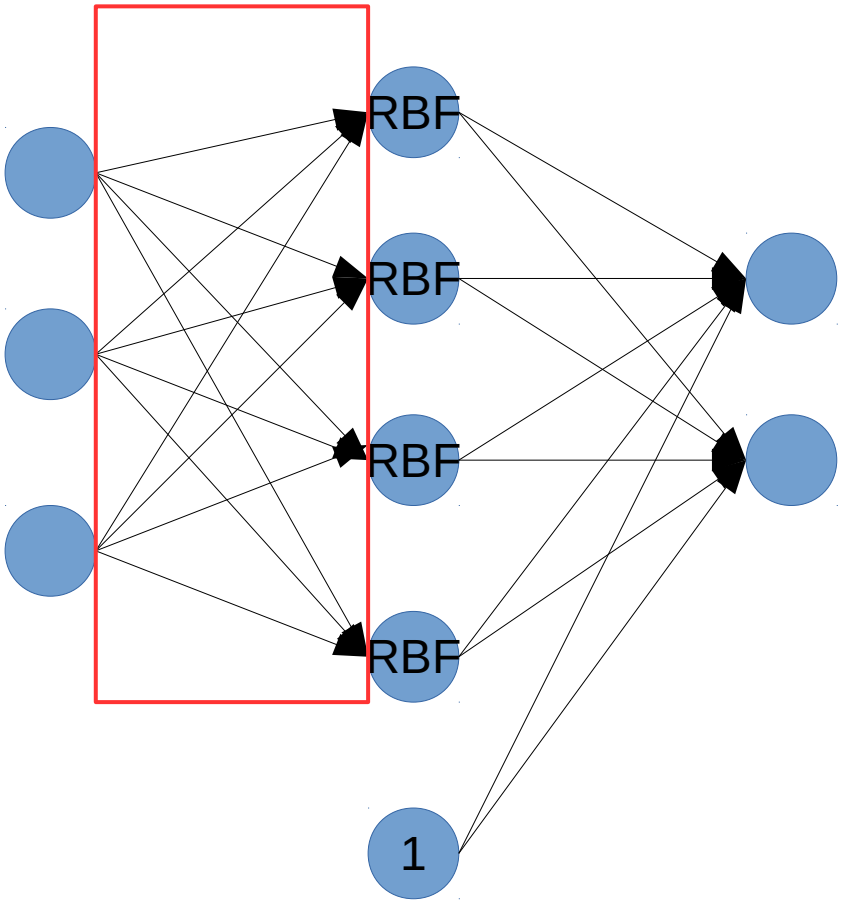
- What are differences between MLPs and RBF-Nets?
  - RBF-Net is a 2 Layer Network, MLPs can have an arbitrary number of layers
  - There aren't any edge weights from input to hidden layer in RBF-Nets

Revision of Lecture

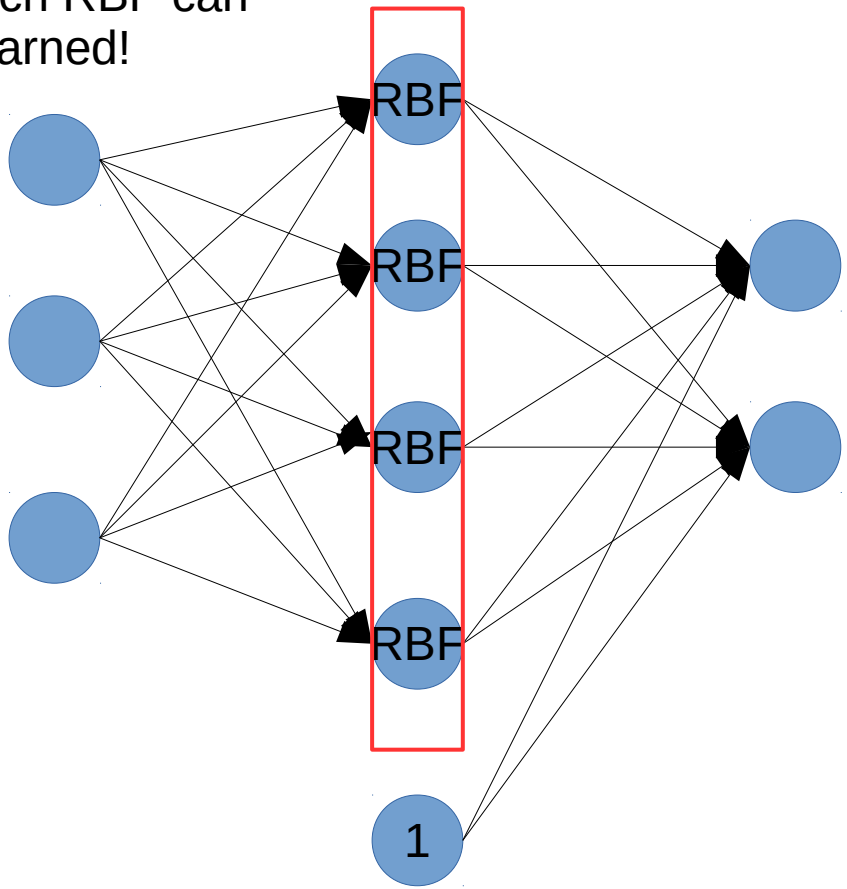
- What are differences between MLPs and RBF-Nets?
  - RBF-Net is a 2 Layer Network, MLPs can have an arbitrary number of layers
  - There aren't any edge weights from input to hidden layer in RBF-Nets
  - Hidden RBF Neurons are not (Rosenblatt) Perceptrons!


RBF
RBF
RBF
RBF
1

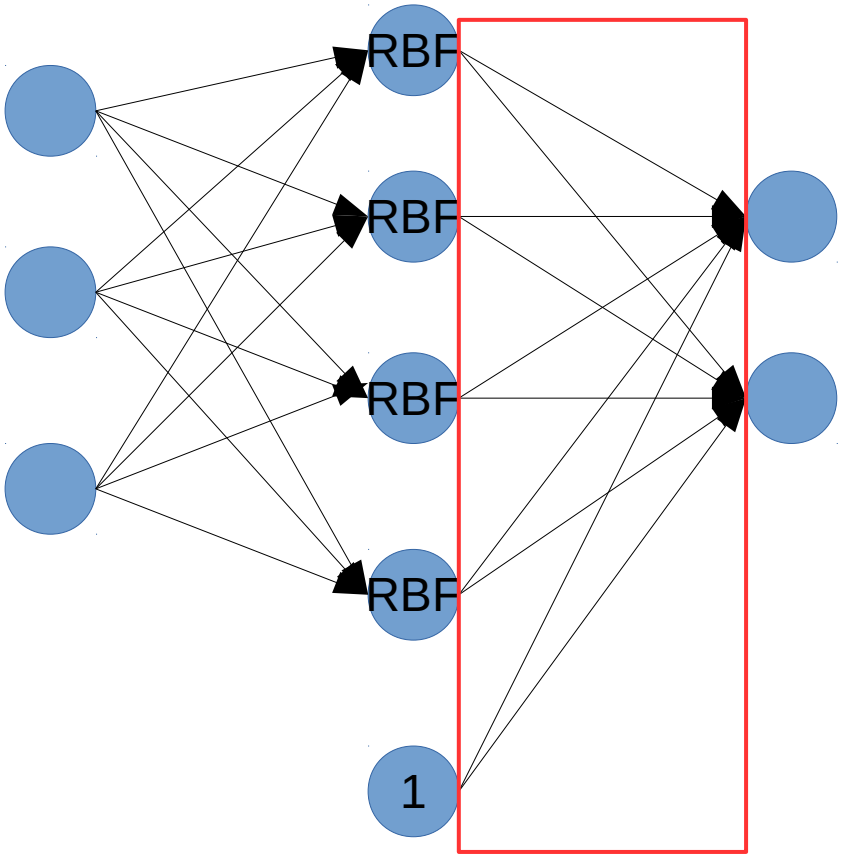
No edge weights!



Centers and areas of influence of each RBF can be learned!



Learnable edge weights (+ bias weight!) like in MLP



Revision of Lecture

- What are the non-adaptable (fixed) parameters in a RBF-Net (aka hyperparameters)?

Revision of Lecture

- What are the non-adaptable (fixed) parameters in a RBF-Net (aka hyperparameters)?
  - (Number of hidden RBF neurons)

Revision of Lecture

- What are the non-adaptable (fixed) parameters in a RBF-Net (aka hyperparameters)?
  - (Number of hidden RBF neurons)
  - Type of Radial Basis Function

Revision of Lecture

- What are the adaptable (learnable) parameters in a RBF-Net?

Revision of Lecture

- What are the adaptable (learnable) parameters in a RBF-Net?
  - Centers of hidden RBF-Neurons

Revision of Lecture

- What are the adaptable (learnable) parameters in a RBF-Net?
  - Centers of hidden RBF-Neurons
  - Area of influence (standard deviation) of Radial Basis Function

Revision of Lecture

- What are the adaptable (learnable) parameters in a RBF-Net?
  - Centers of hidden RBF-Neurons
  - Area of influence (standard deviation) of Radial Basis Function
  - Edge weights between hidden and output layer

Revision of Lecture

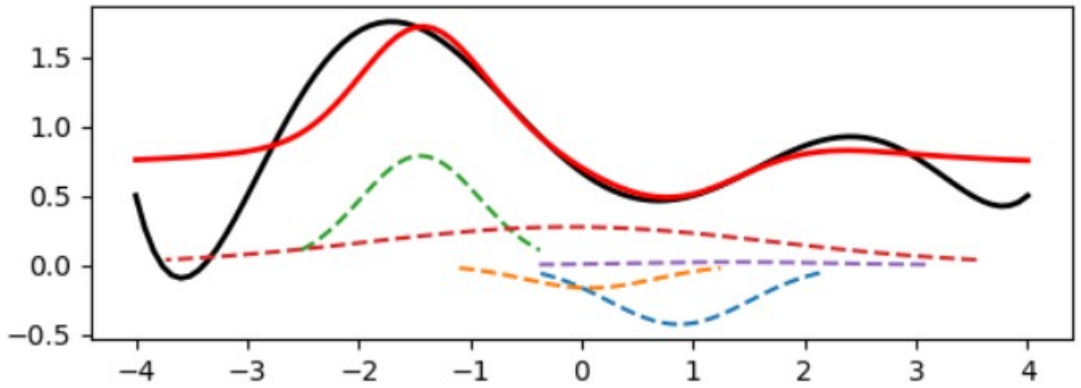
- What are the adaptable (learnable) parameters in a RBF-Net?
  - Centers of hidden RBF-Neurons
  - Area of influence (standard deviation) of Radial Basis Function
  - Edge weights between hidden and output layer
  - (Number of hidden RBF neurons)

Revision of Lecture

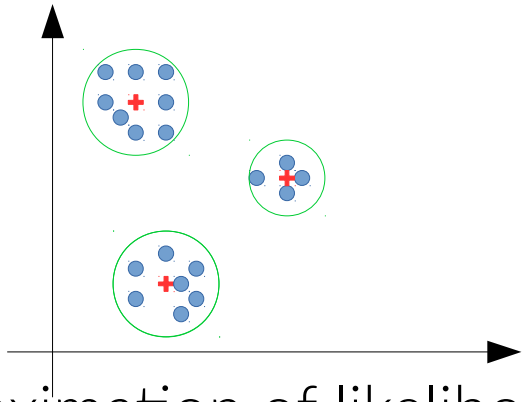
- Which tasks can a RBF-Net solve?

Revision of Lecture

- Which tasks can a RBF-Net solve?
  - Regression tasks:
    - Function approximation by linear combination of RBFs (e.g. Gaussians)



Revision of Lecture

- Which tasks can a RBF-Net solve?
  - Classification tasks:
    - Linear combination of RBFs models likelihood of training data!
    - 
    - Approximation of likelihood to infere posterior probability!

Revision of Lecture

- Learning paradigms for RBF-Net?

Revision of Lecture

- Learning paradigms for RBF-Net?
  - Two-layered, sequential learning

Revision of Lecture

- Learning paradigms for RBF-Net?
  - Two-layered, sequential learning
  - Joint learning paradigm
    - Gradient descent (number of centers fixed!)

Revision of Lecture

- Learning paradigms for RBF-Net?
  - Two-layered, sequential learning
  - Joint learning paradigm
    - Gradient descent (number of centers fixed!)
  - Combination of both paradigms

Revision of Lecture

- Two-layered, sequential learning?

Revision of Lecture

- Two-layered, sequential learning?
  - 1. step: adapt params of 1st layer (unsupervised)
    - Centers
    - Reach (standard deviation)
    - (number of centers)

Revision of Lecture

- Two-layered, sequential learning?
  - 1. step: adapt params of 1st layer (unsupervised)
    - Centers
    - Reach (standard deviation)
    - (number of centers)
  - 2. step: adapt params of 2nd layer (supervised)
    - Edge weights

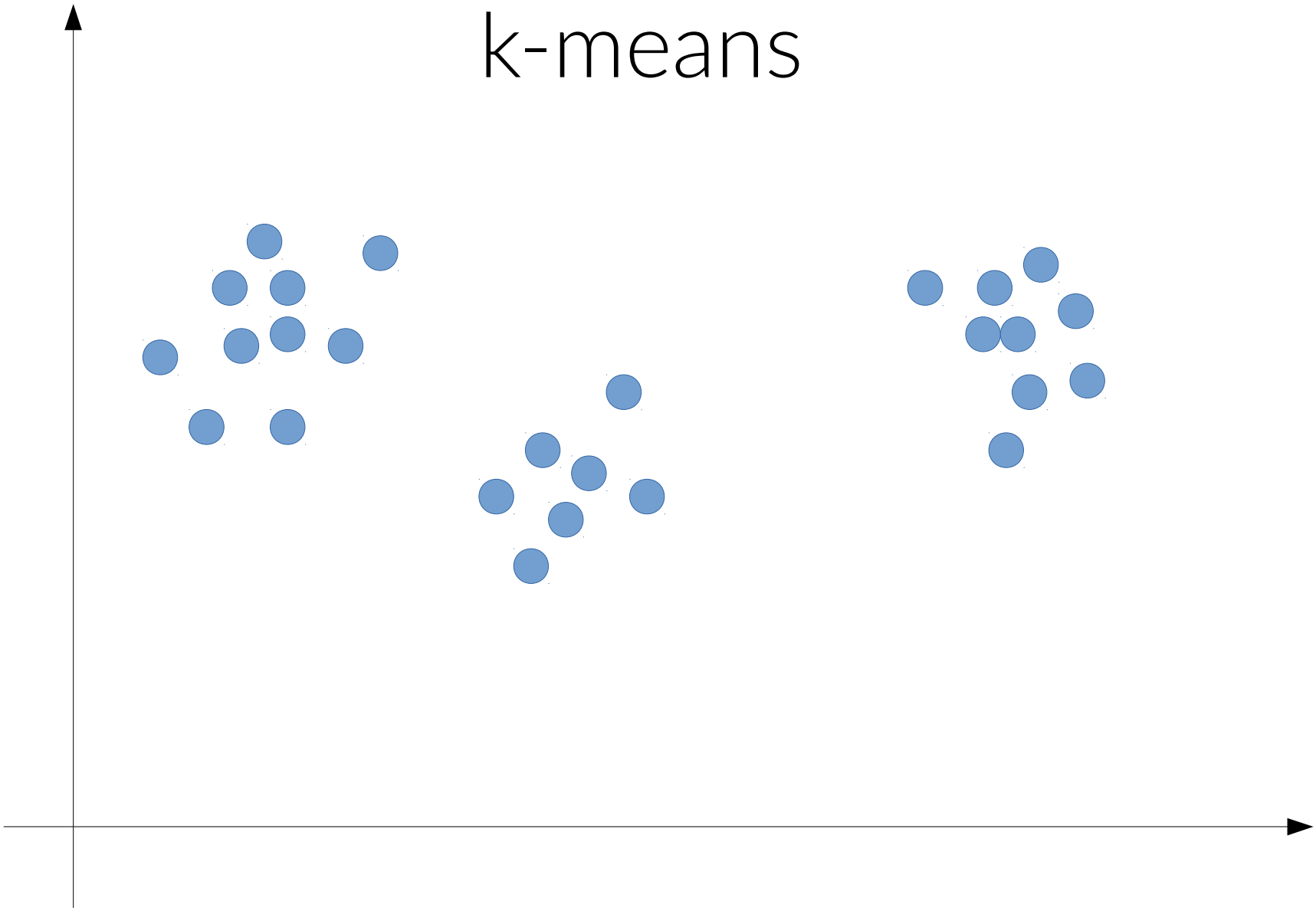
Revision of Lecture

- How to adapt params of 1st layer?

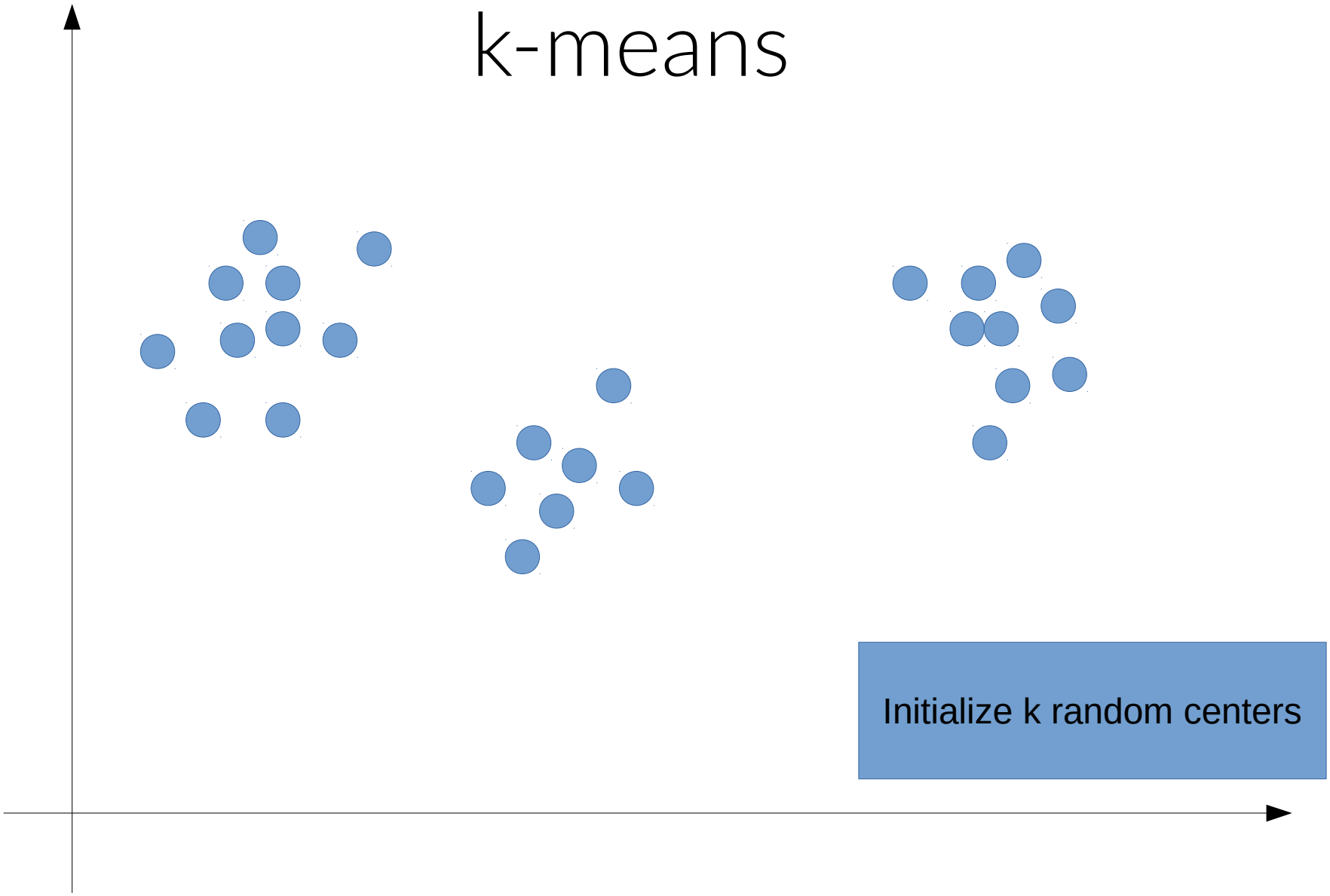
Revision of Lecture

- How to adapt params of 1st layer?
  - K-means (number of centers not adaptable)
  - ISODATA (number of centers adaptable)
  - (DBSCAN)(number of centers adaptable)

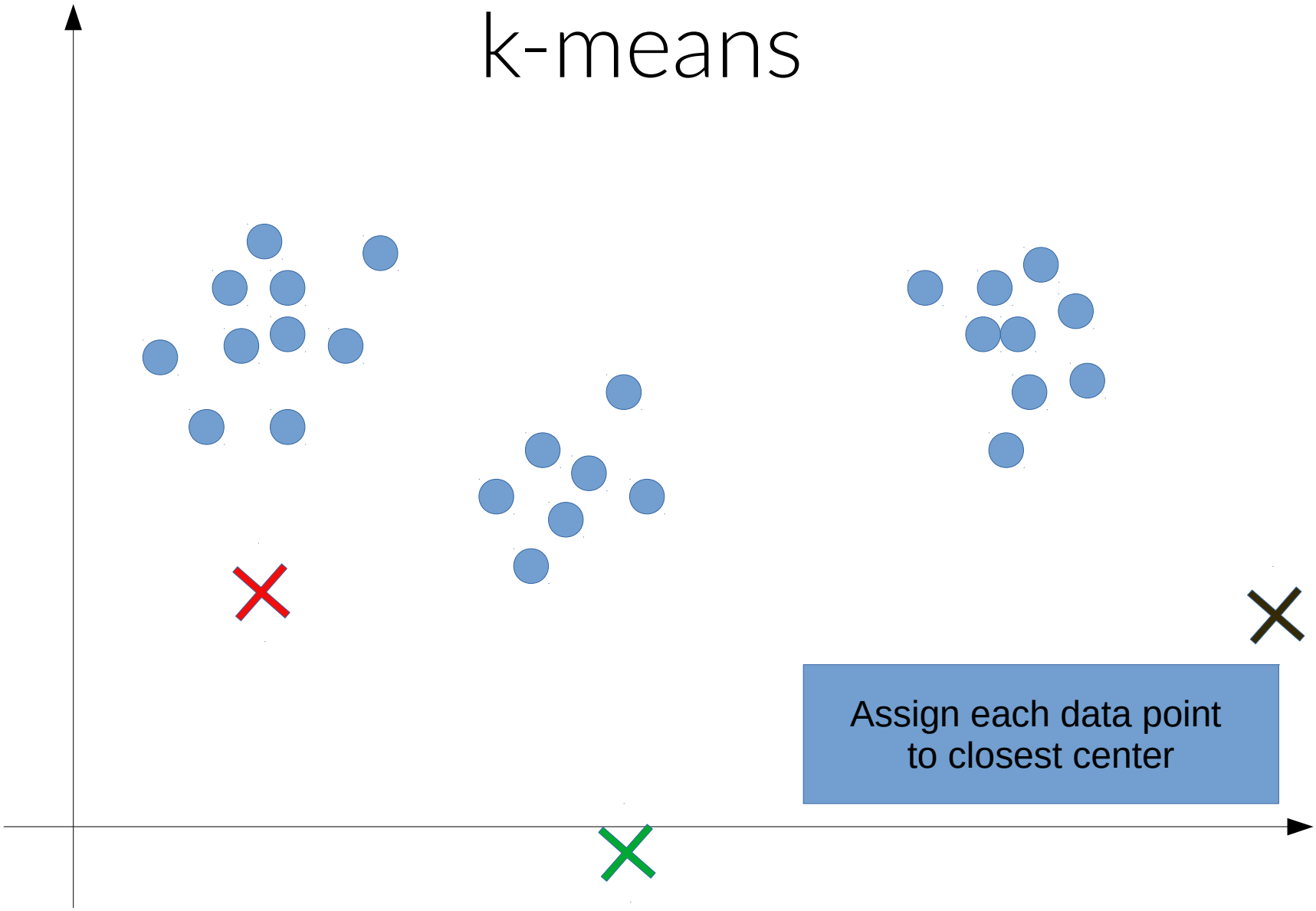
k-means



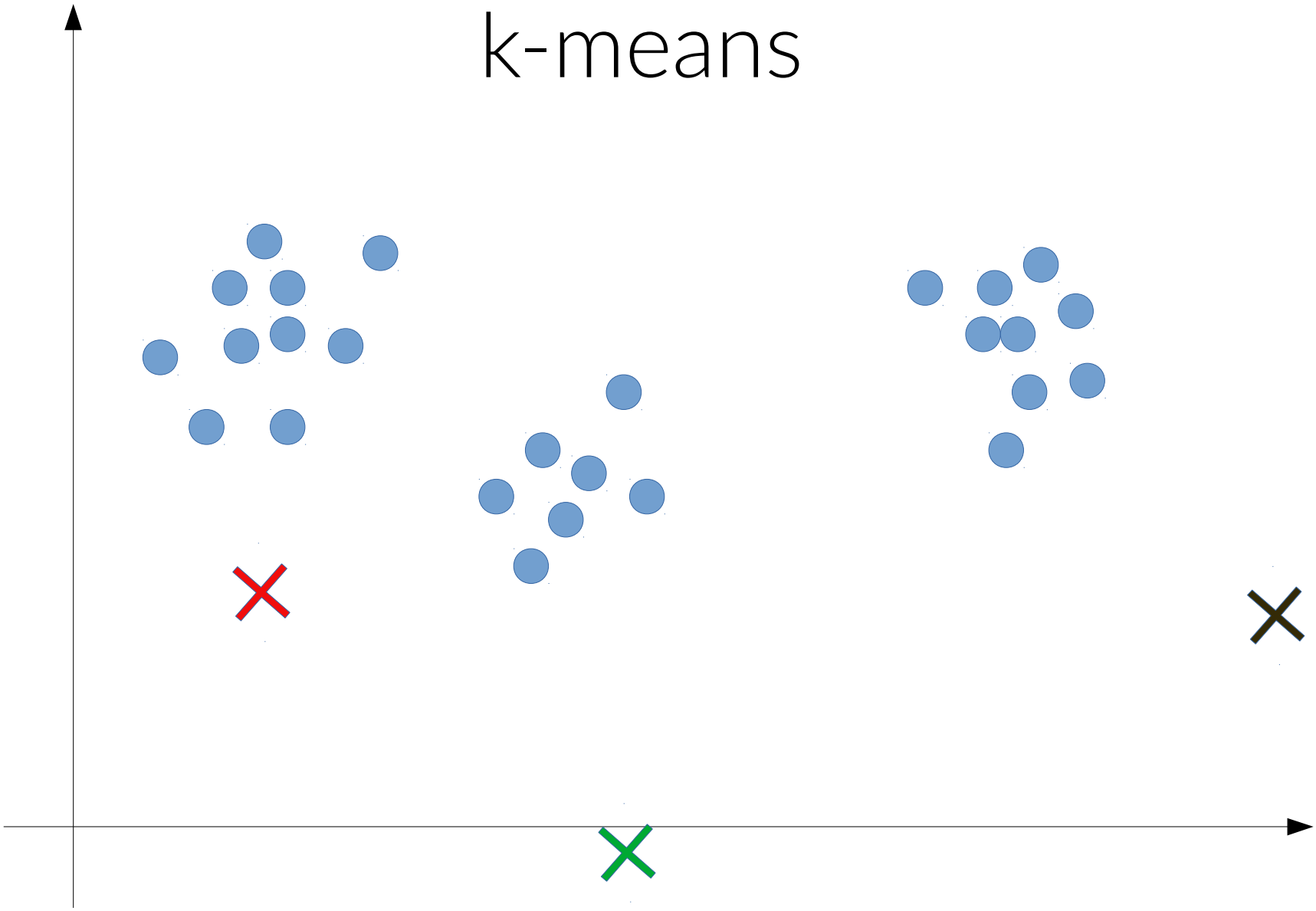
k-means



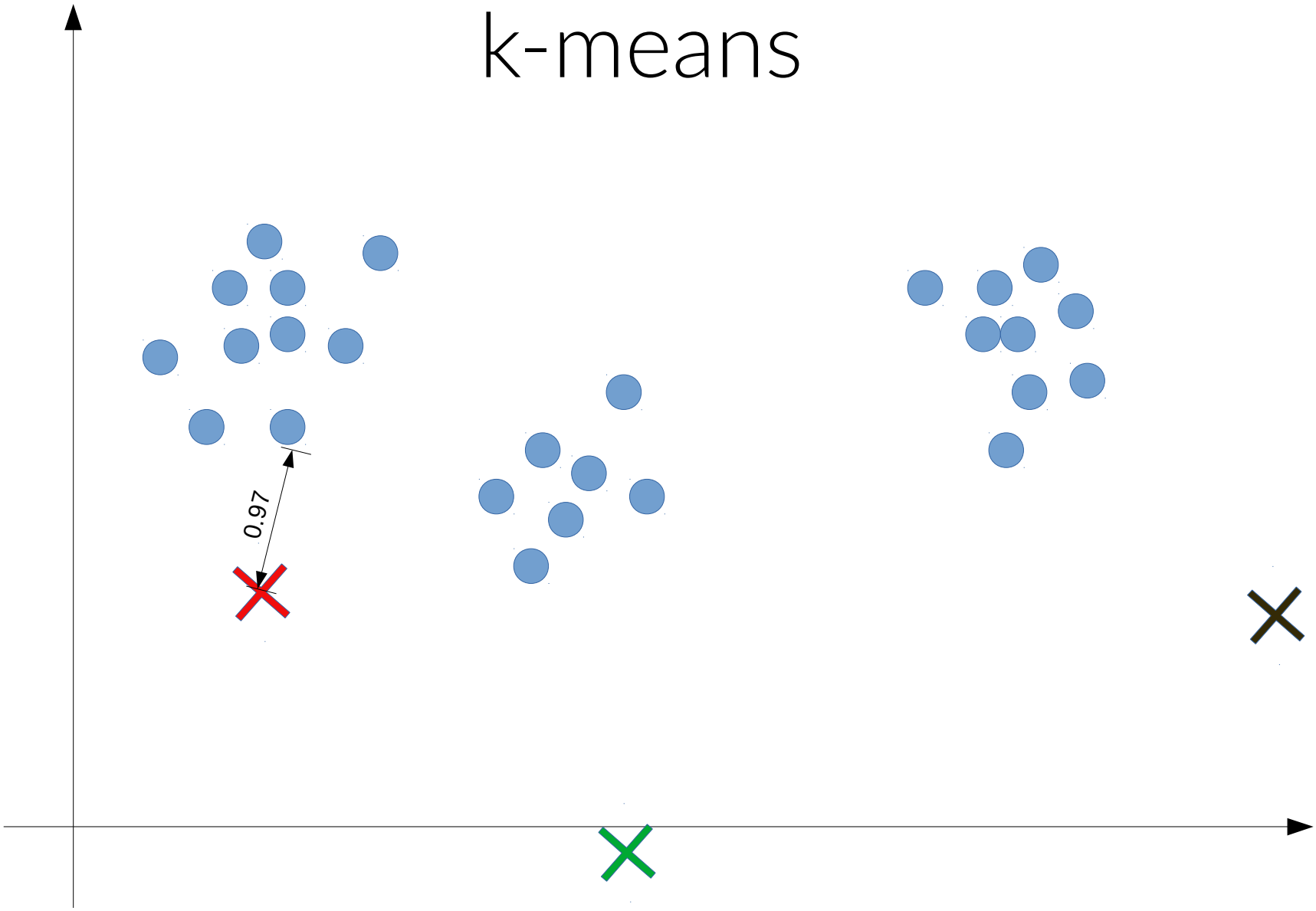
k-means



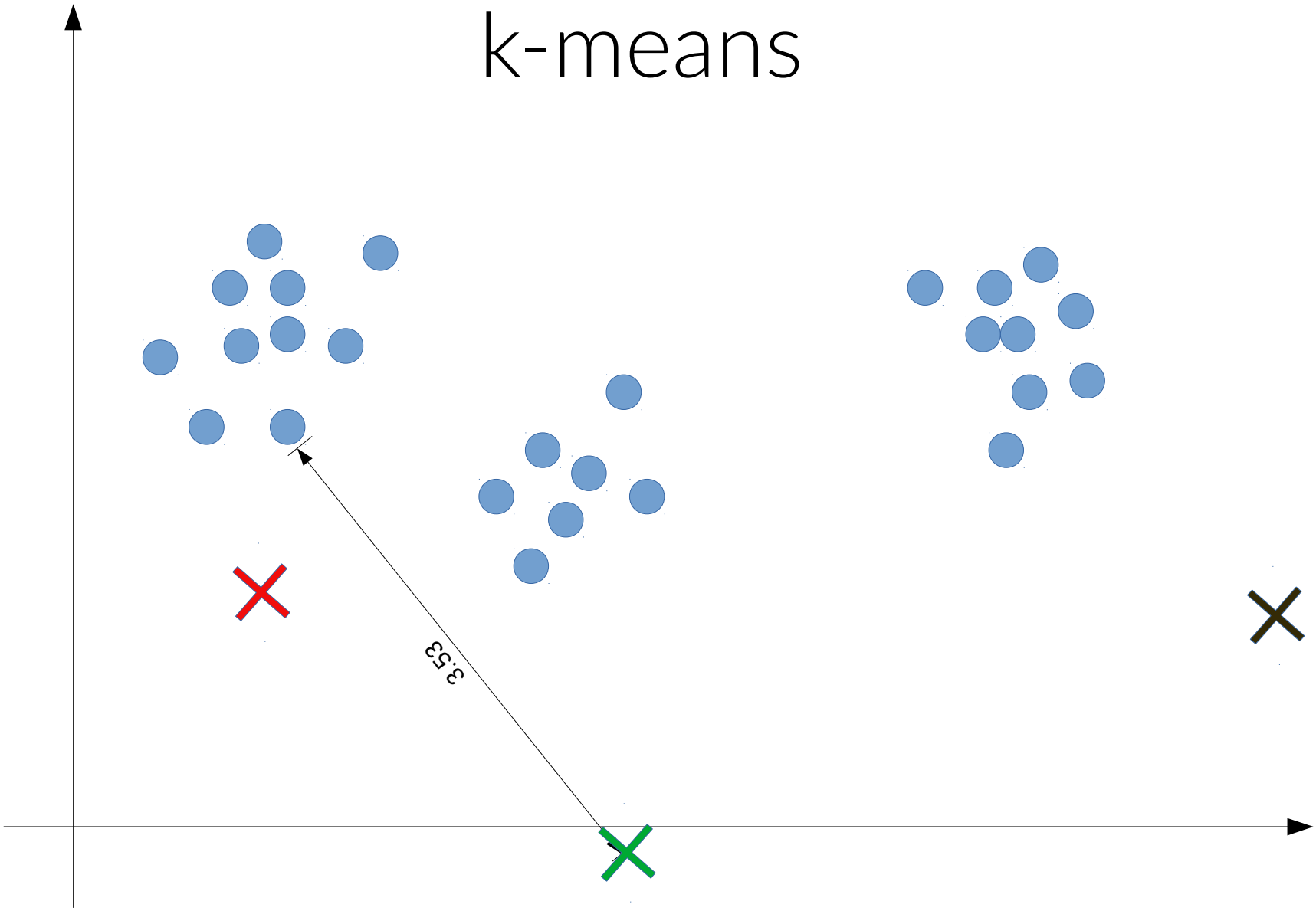
k-means



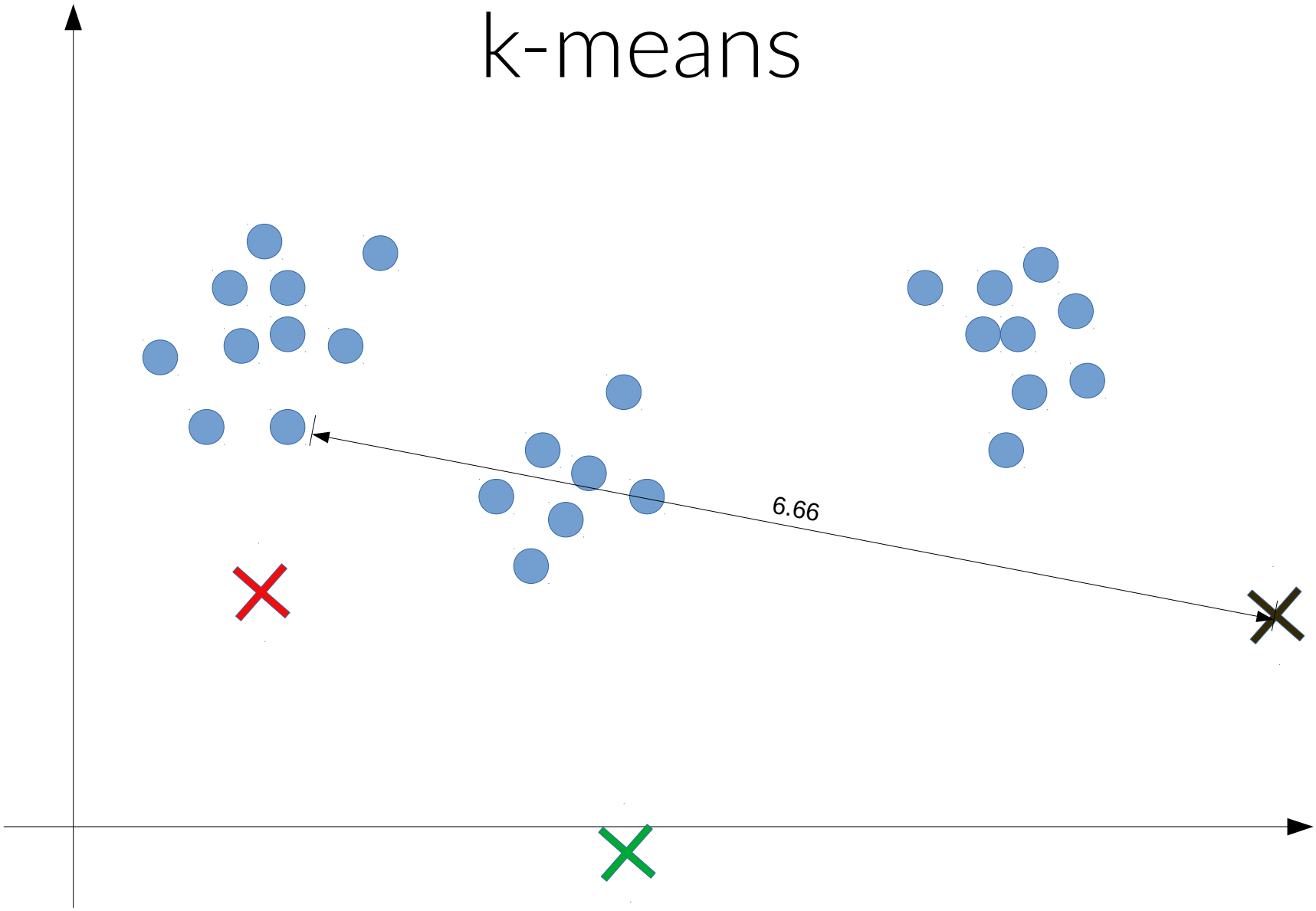
k-means



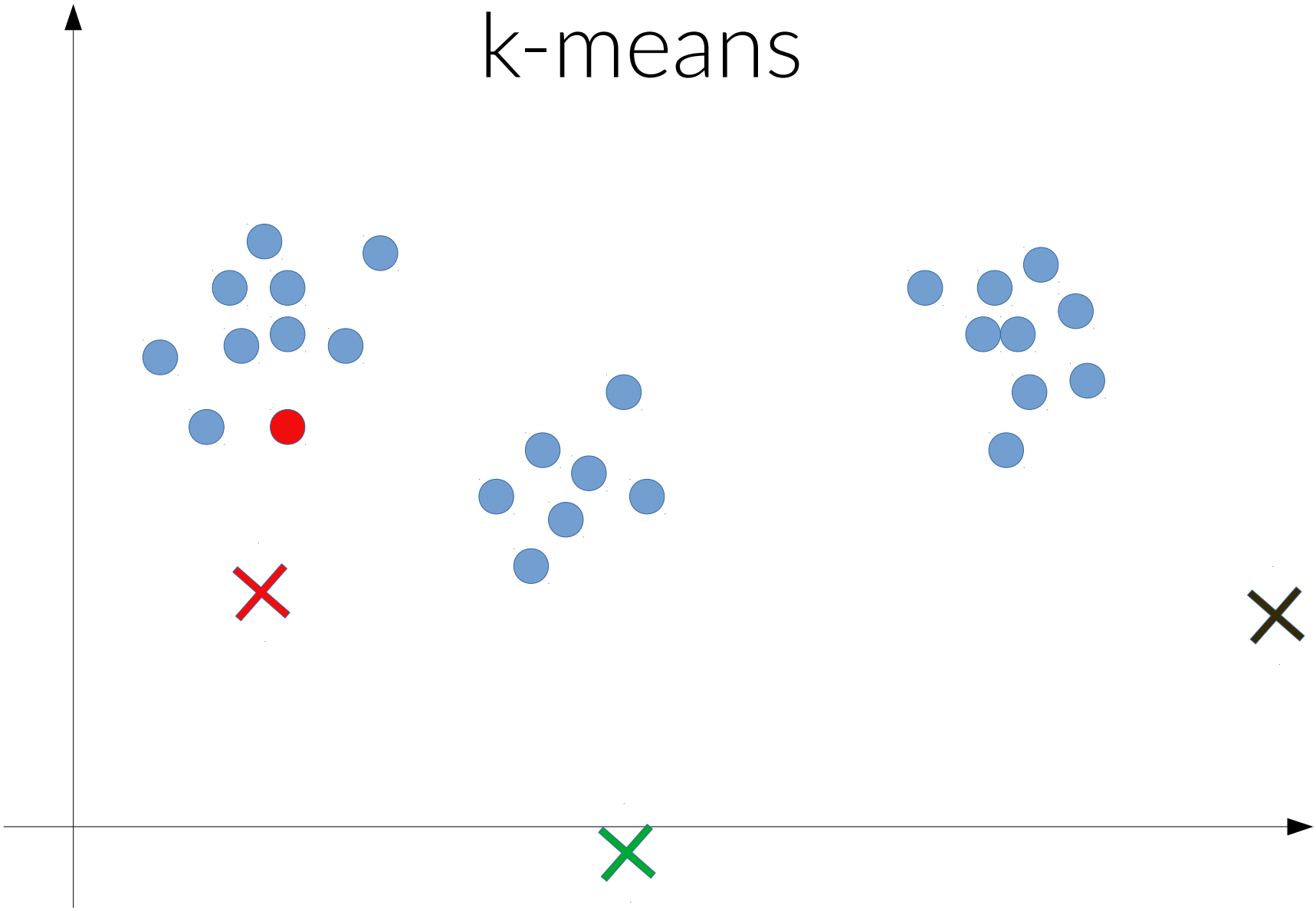
k-means



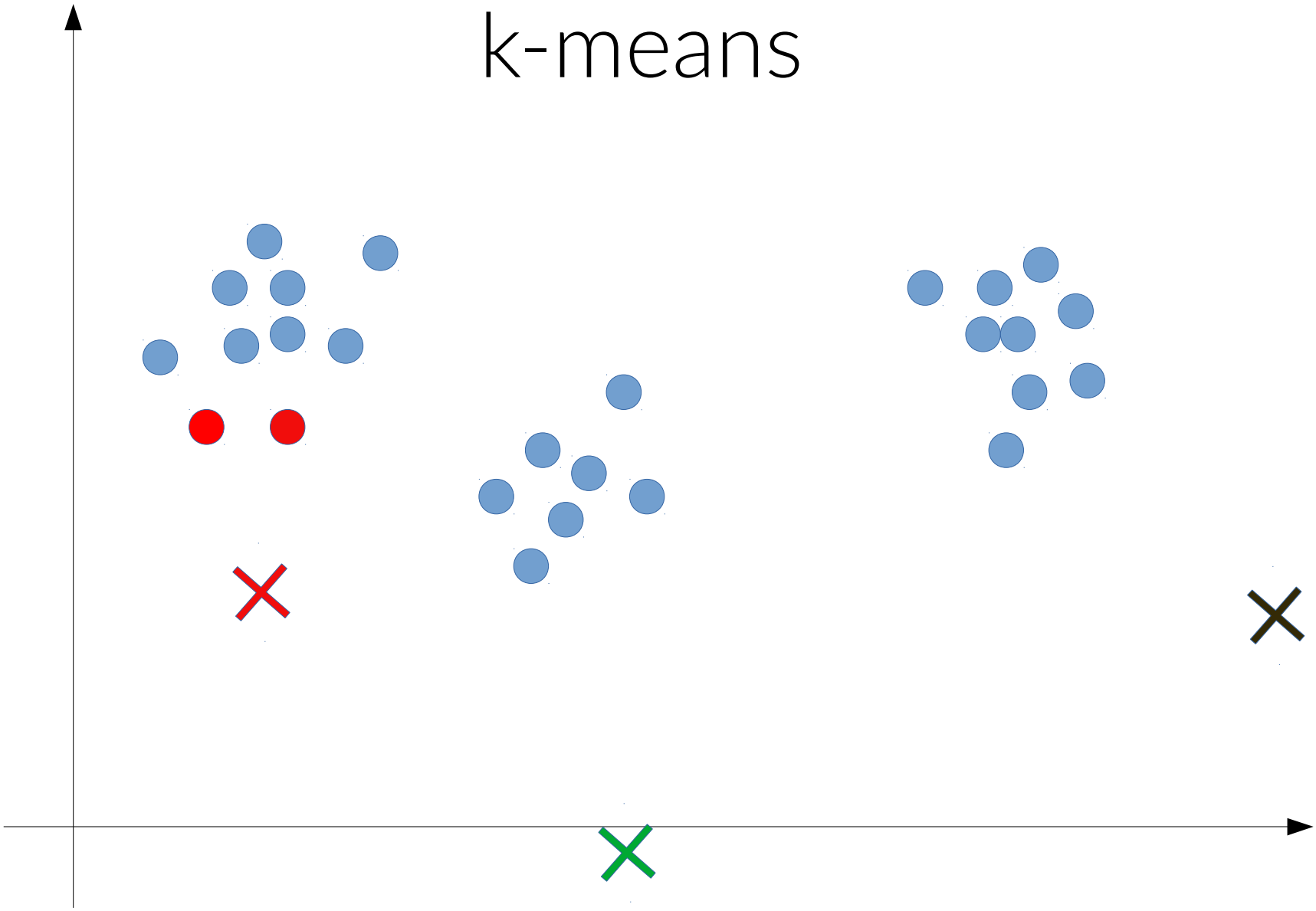
k-means



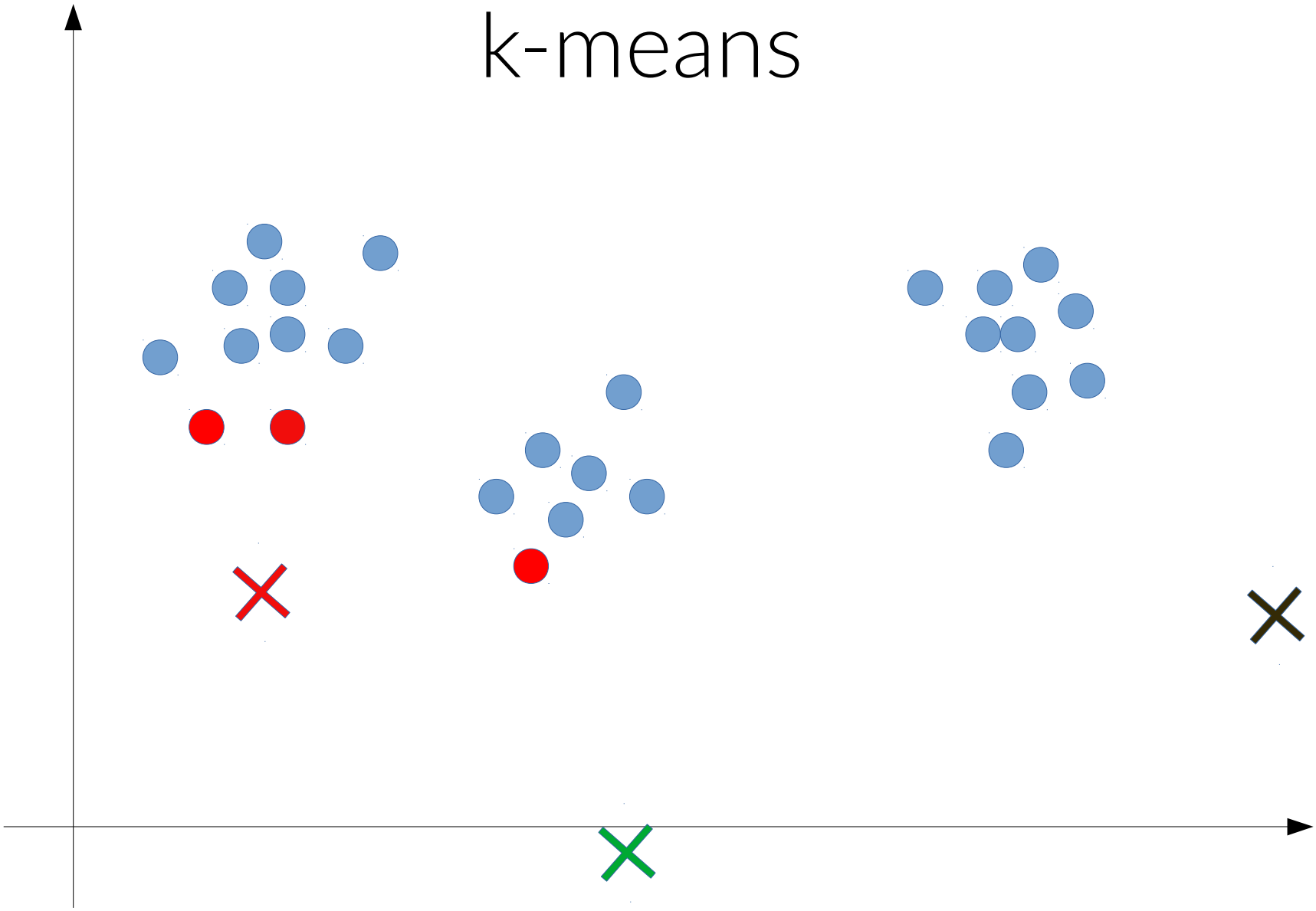
k-means



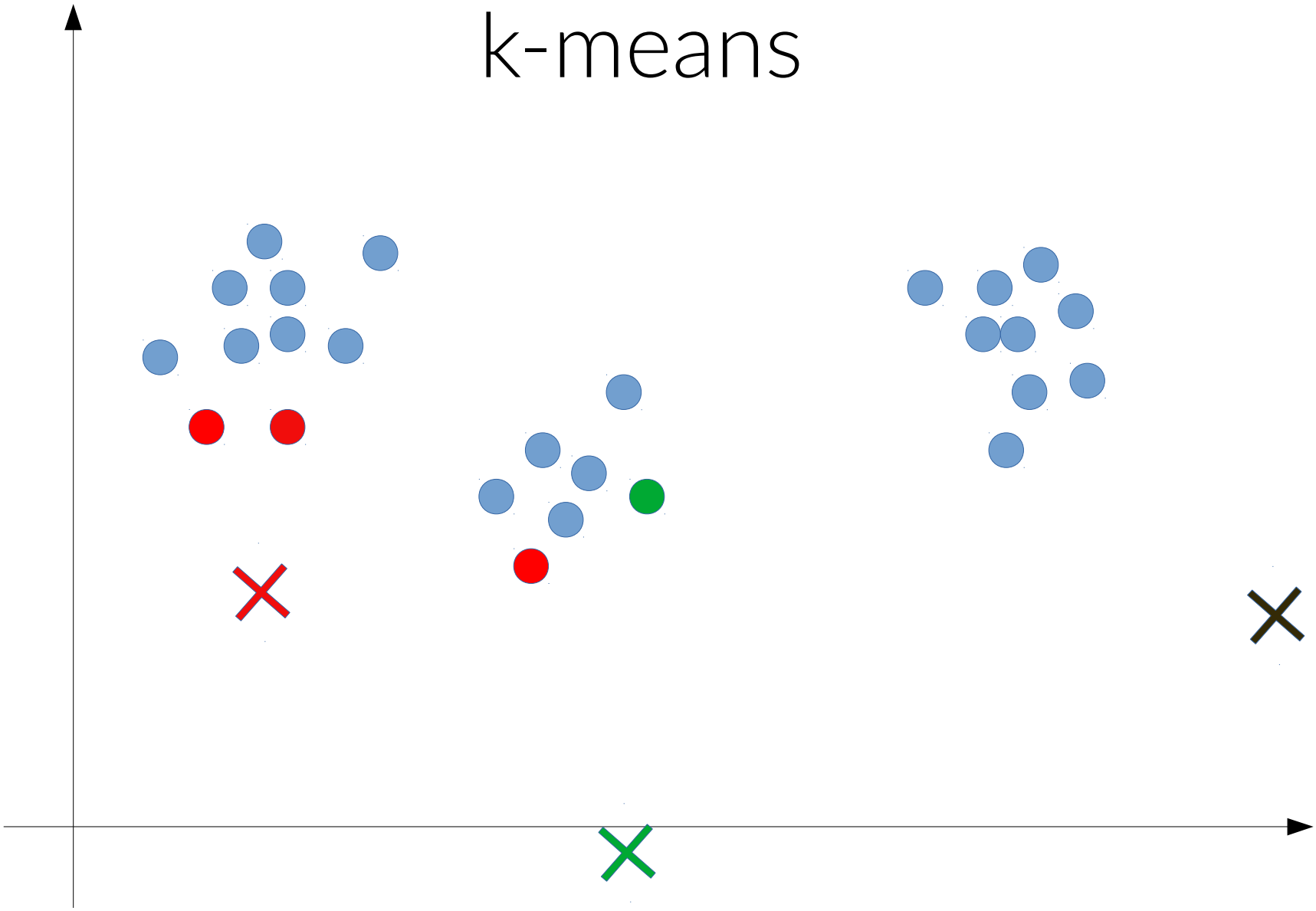
k-means



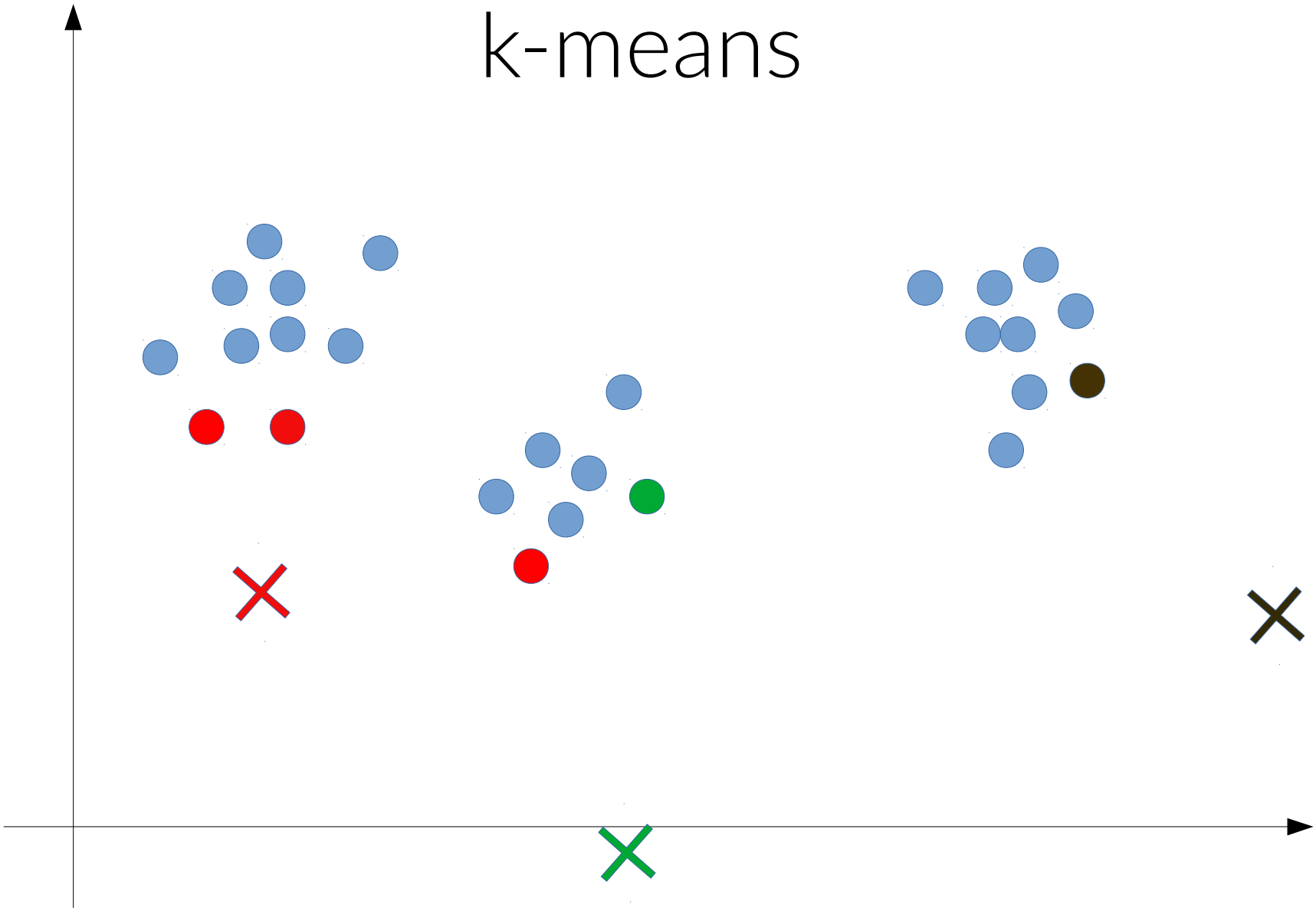
k-means



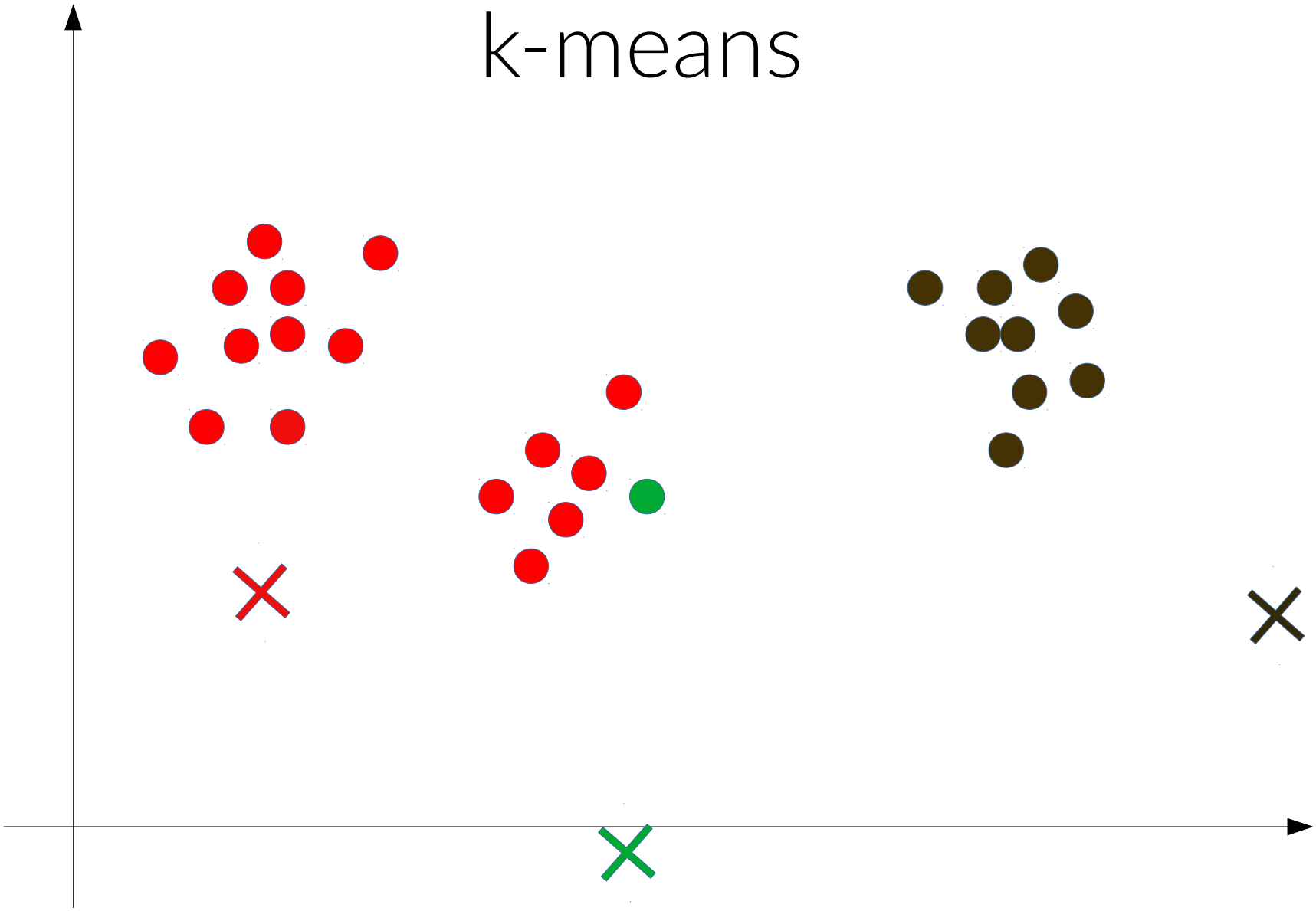
k-means



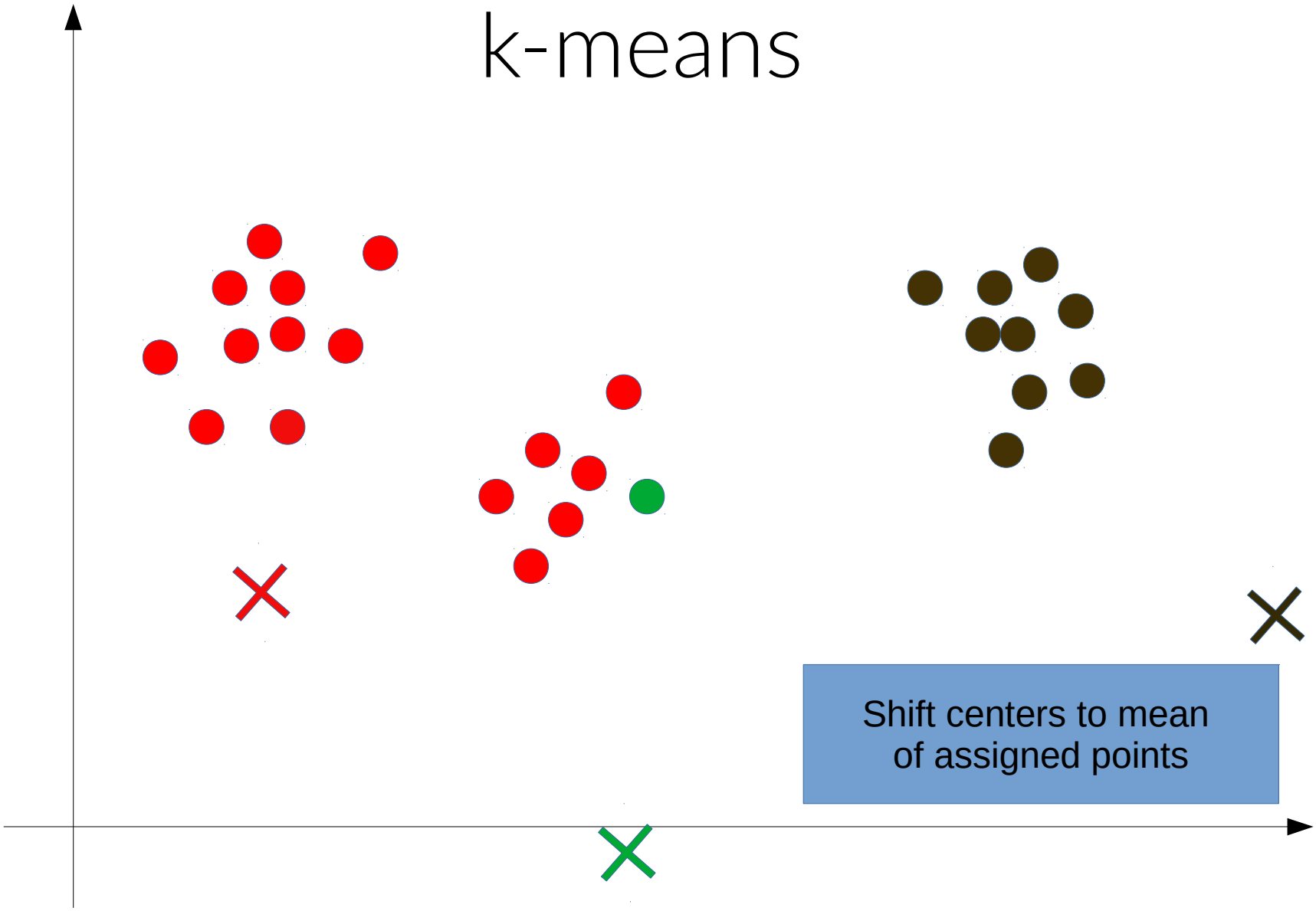
k-means



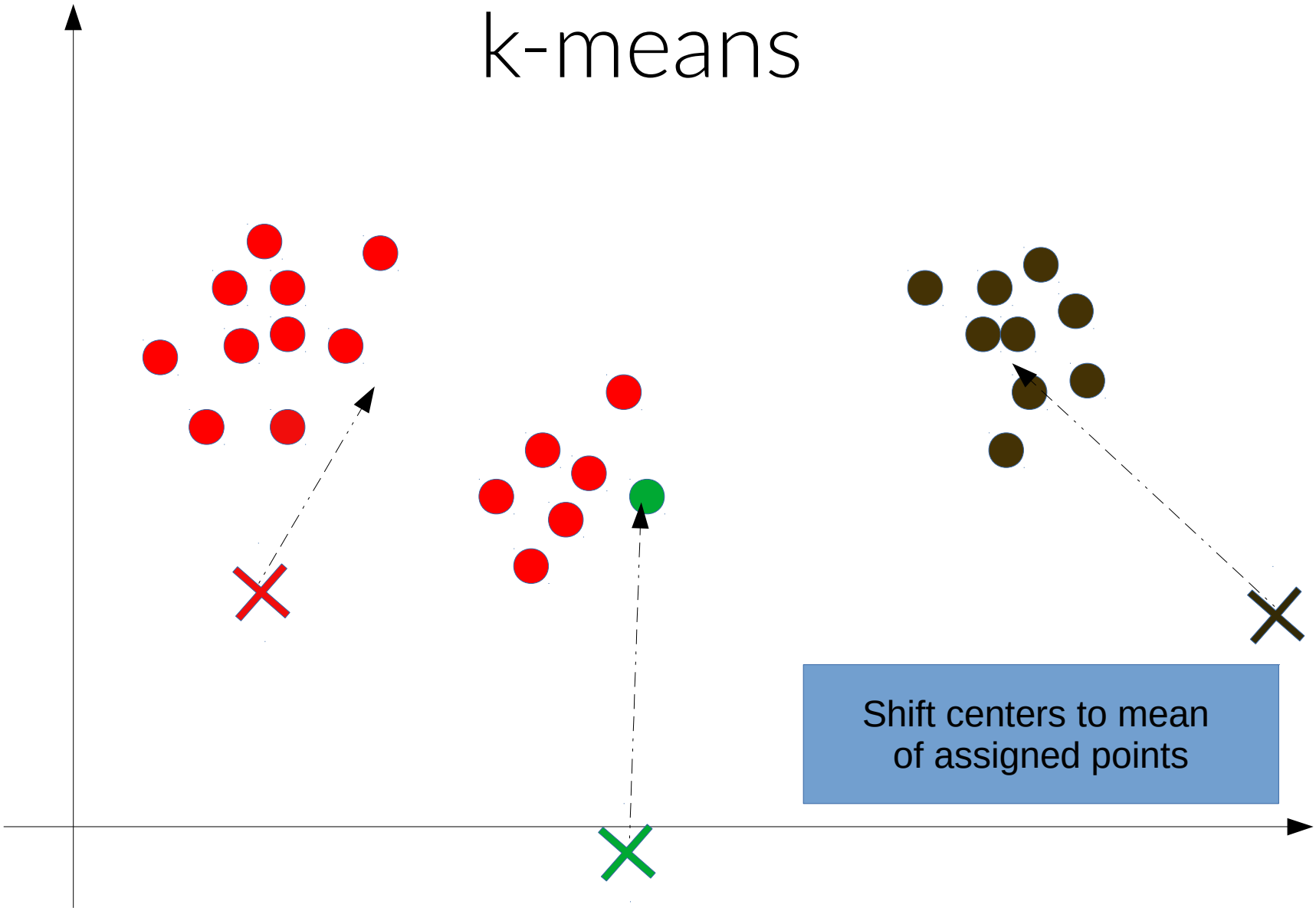
k-means



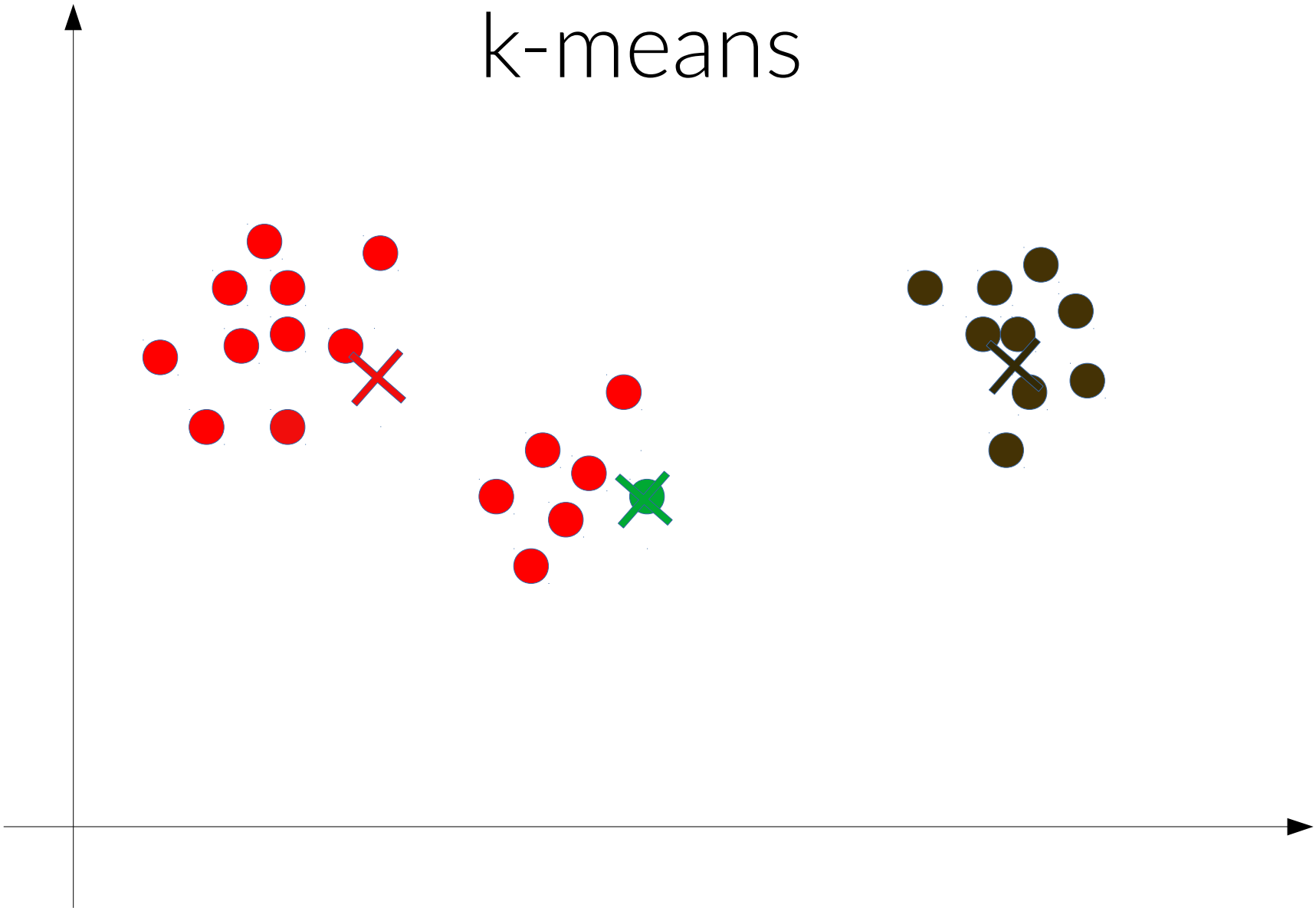
k-means



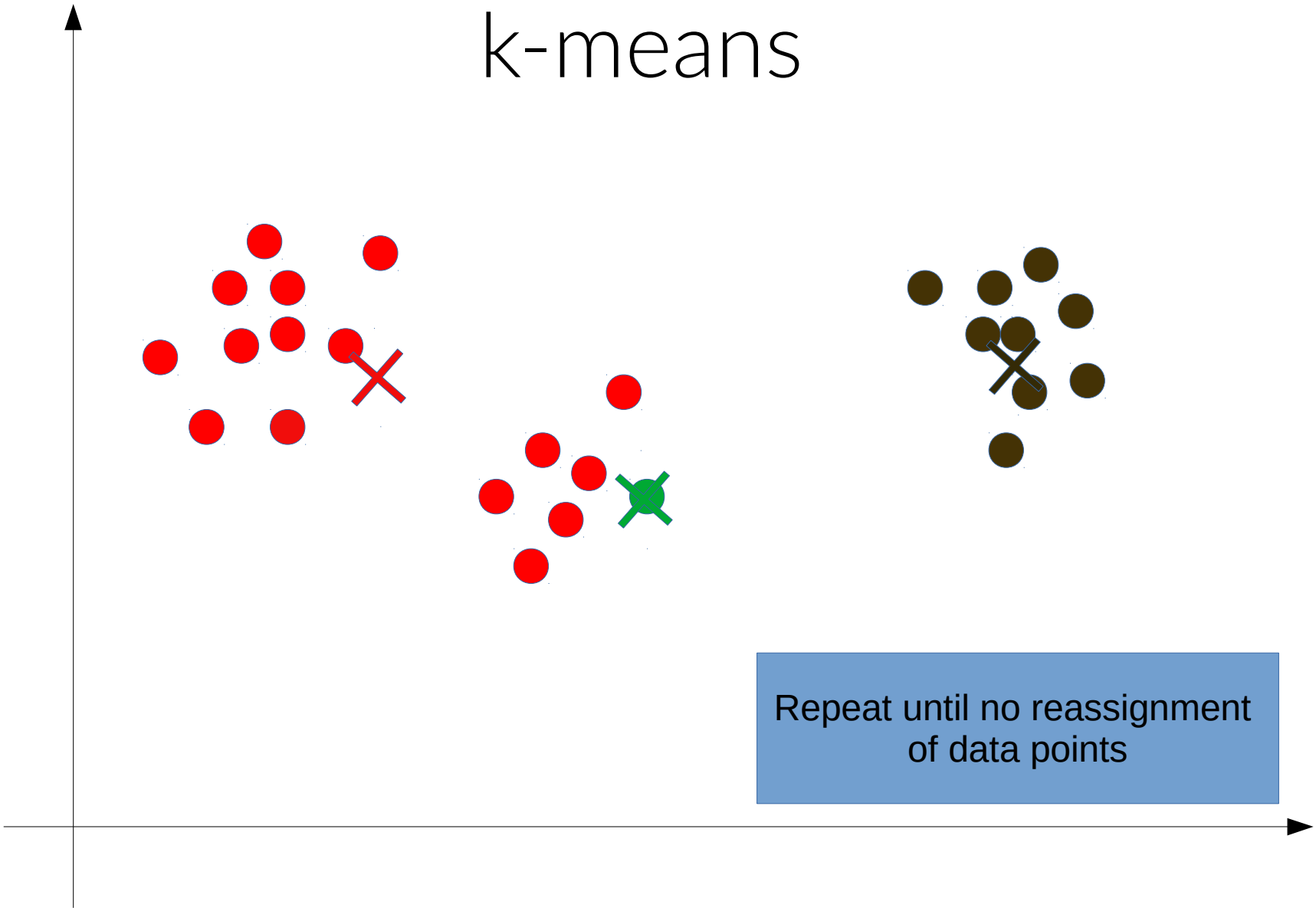
k-means



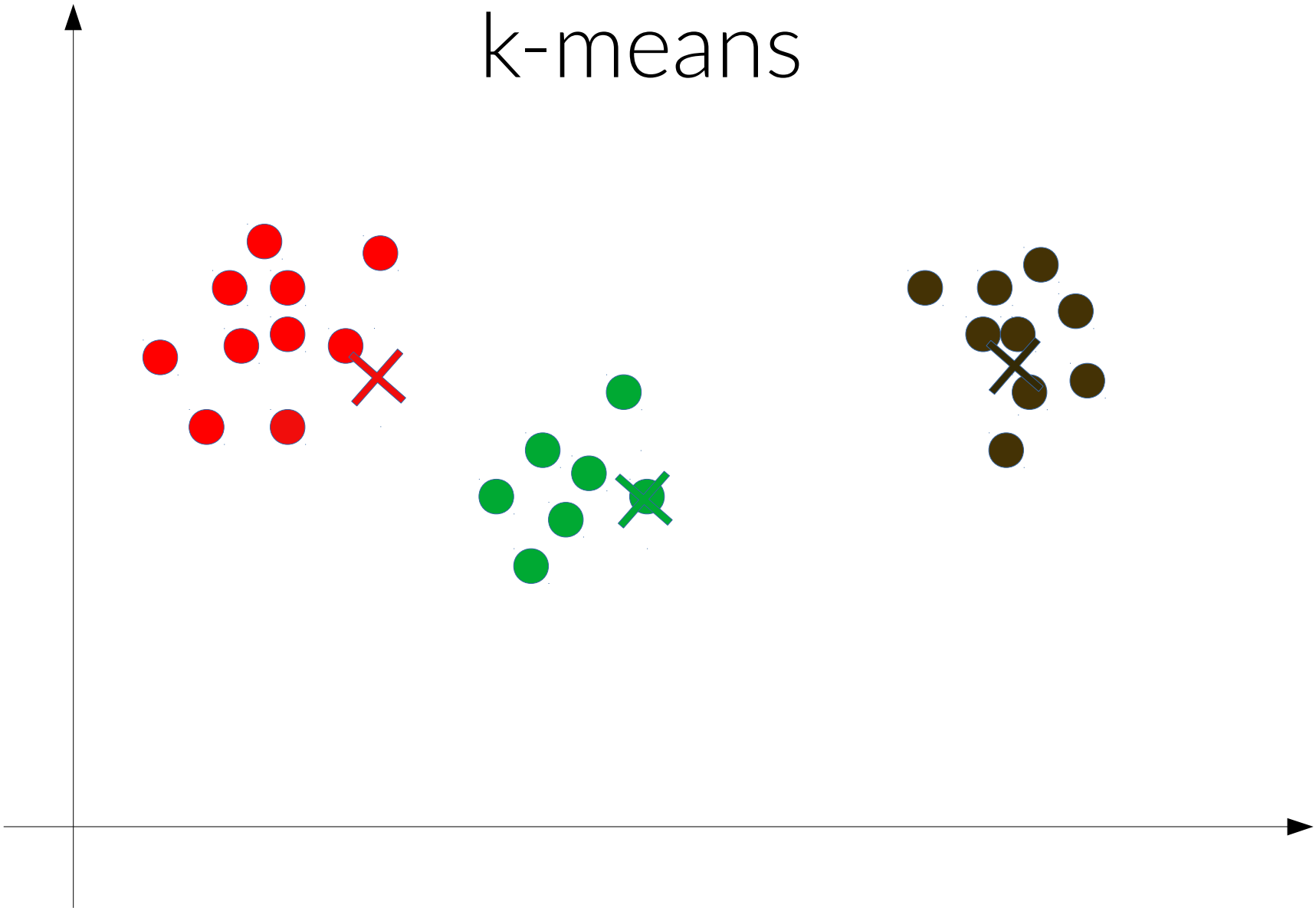
k-means



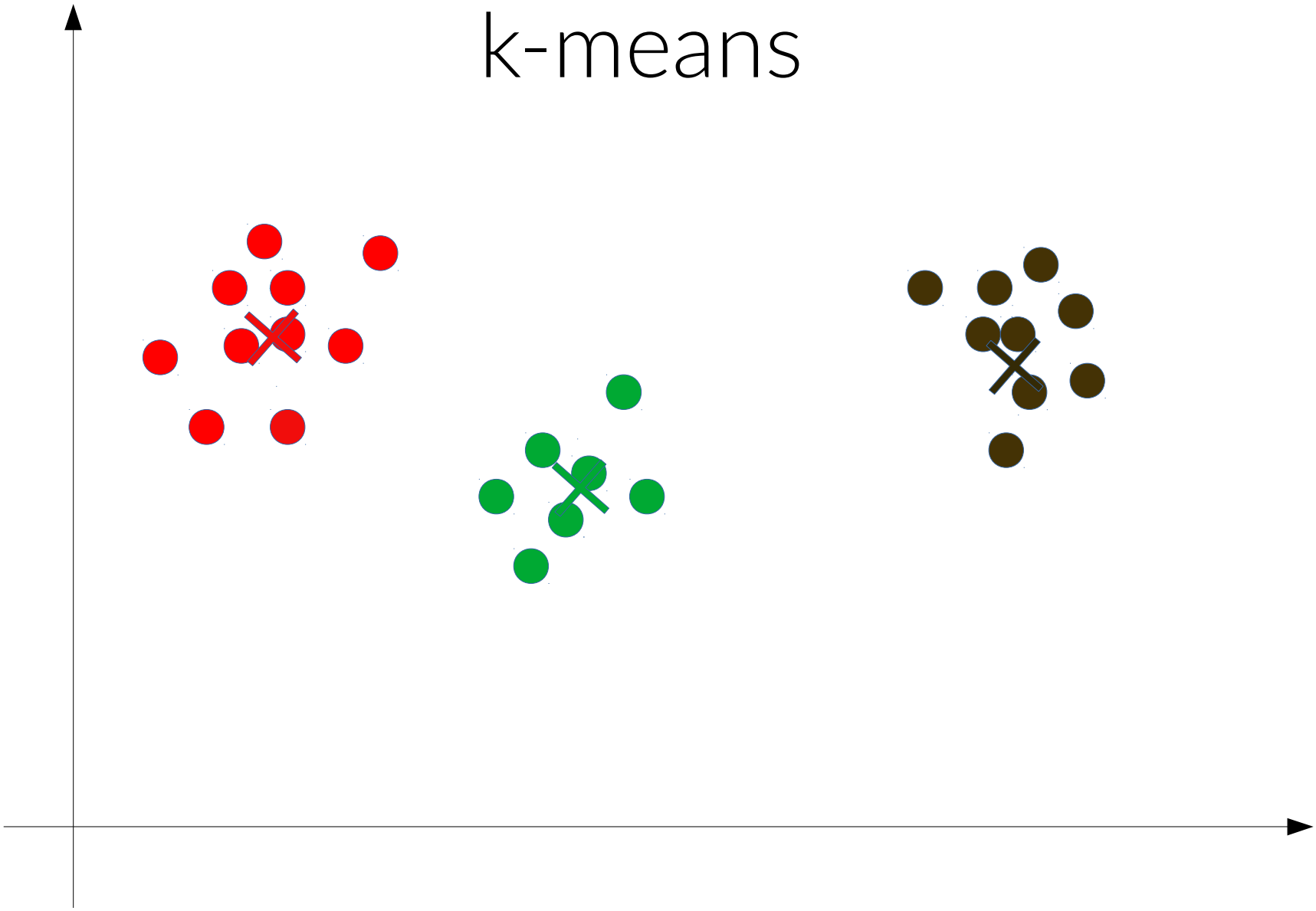
k-means



k-means

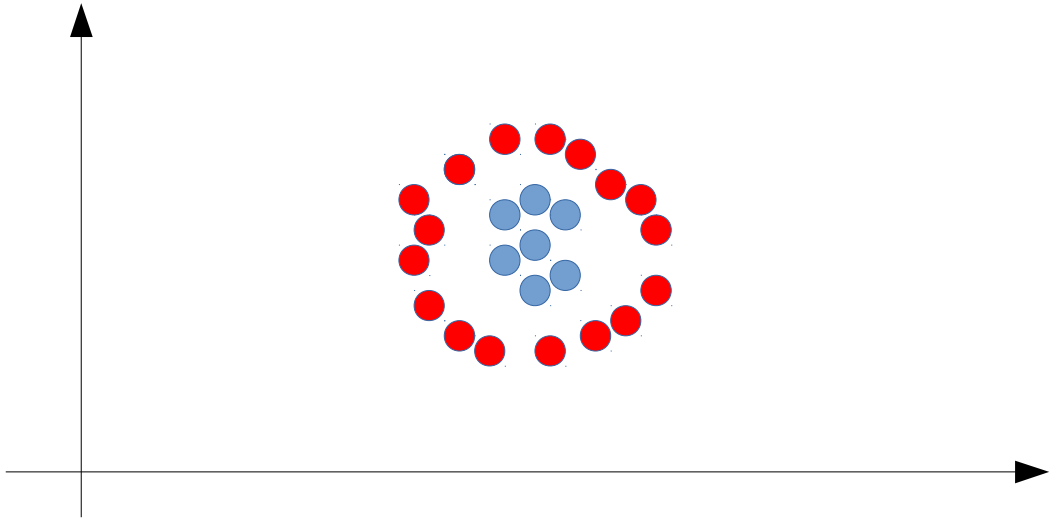


k-means



Revision of Lecture

- Drawbacks of k-means?
  - number of centers fixed
  - Assumes spherical shape of clusters



Revision of Lecture

- How to adapt params of 2nd layer?

Revision of Lecture

- How to adapt params of 2nd layer?
  - Linear regression by means of pseudo-inverse

Lineare Regression durch Pseudo-Inverse

Funktion des RBF-Netzes: $\boldsymbol{f(x) := \sum_{j=1}^{J} w_j h_j(x)}$

Trainingsdaten: $\boldsymbol{(x^m, r^m) \in \Omega_T \subset \mathbb{R}^I \times \mathbb{R}}$

Fehlerfunktion: $\boldsymbol{D(w) := \frac{1}{2} \sum_{m=1}^{M} (f(x^m) - r^m)^2}$

Optimaler Gewichtsfektor: $\boldsymbol{w^* := \underbrace{(H^T H)^{-1} \cdot H^T}_{\text{Pseudo-Inverse von } H} \cdot r}$

Es ist $\boldsymbol{w^*}$ der Vektor mit den $\boldsymbol{J}$ optimalen Gewichten, $\boldsymbol{H}$ die $(\boldsymbol{M \times J})$-Matrix mit Komponenten $\boldsymbol{h_j(x^m)}$, und $\boldsymbol{r}$ der Vektor mit den $\boldsymbol{M}$ Solldaten.

Revision of Lecture

- How to adapt params of 2nd layer?
  - Linear regression by means of pseudo-inverse

Lineare Regression durch Pseudo-Inverse

Funktion des RBF-Netzes: $\boldsymbol{f(x) := \sum_{j=1}^{J} w_j h_j(x)}$

Trainingsdaten: $\boldsymbol{(x^m, r^m) \in \Omega_T \subset \mathbb{R}^I \times \mathbb{R}}$

Fehlerfunktion: $\boldsymbol{D(w) := \frac{1}{2} \sum_{m=1}^{M} (f(x^m) - r^m)^2}$

Optimaler Gewichtsfektor: $\boldsymbol{w^* := \underbrace{(H^T H)^{-1} \cdot H^T}_{\text{Pseudo-Inverse von } H} \cdot r}$

Es ist $\boldsymbol{w^*}$ der Vektor mit den $\boldsymbol{J}$ optimalen Gewichten, $\boldsymbol{H}$ die $(\boldsymbol{M \times J})$-Matrix mit Komponenten $\boldsymbol{h_j(x^m)}$, und $\boldsymbol{r}$ der Vektor mit den $\boldsymbol{M}$ Solldaten.

Remember Adaline?

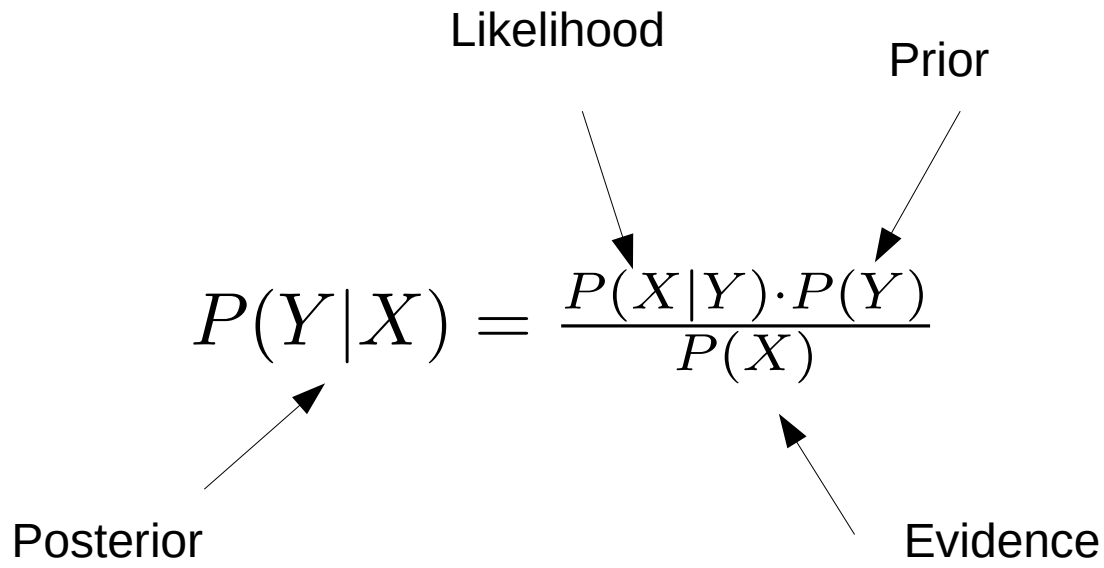
Reminder to (Bayesian)statistics

- Aim of each classifier is to output the posterior probability:
- Given an observation/a feature X, how probable is some output Y?

$$P(Y|X) = \frac{P(X|Y) \cdot P(Y)}{P(X)}$$

Reminder to (Bayesian)statistics

- Aim of each classifier is to output the posterior probability:
- Given an observation/a feature X, how probable is some output Y?



Discriminative vs Generative Models

- Models that aim to learn the likelihood (probability distribution of training data, given an output) are called *generative*

Discriminative vs Generative Models

- Models that aim to learn the likelihood (probability distribution of training data, given an output) are called *generative*
- These can generate more sample points from the learnt distribution

Discriminative vs Generative Models

- Models that aim to learn the likelihood (probability distribution of training data, given an output) are called *generative*
- These can generate more sample points from the learnt distribution

- Models that aim to directly learn the posterior distribution are called *discriminative*

Discriminative or Generative?

- RBF?

A: Generative

B: Discriminative

Discriminative or Generative?

- RBF? - Generative

Discriminative or Generative?

- RBF? - Generative
- MLP?

A: Generative

B: Discriminative

Discriminative or Generative?

- RBF? - Generative
- MLP? - Discriminative

Discriminative or Generative?

- RBF? - Generative
- MLP? - Discriminative
- CNN?

A: Generative

B: Discriminative

Discriminative or Generative?

- RBF? - Generative
- MLP? - Discriminative
- CNN? - Discriminative

Discriminative or Generative?

- RBF? - Generative
- MLP? - Discriminative
- CNN? - Discriminative
- Naive Bayes?

A: Generative

B: Discriminative

Discriminative or Generative?

- RBF? - Generative
- MLP? - Discriminative
- CNN? - Discriminative
- Naive Bayes? - Generative (although naive!)

Discriminative or Generative?

- RBF? - Generative
- MLP? - Discriminative
- CNN? - Discriminative
- Naive Bayes? - Generative (although naive!)
- Adaline?

A: Generative

B: Discriminative

Discriminative or Generative?

- RBF? - Generative
- MLP? - Discriminative
- CNN? - Discriminative
- Naive Bayes? - Generative (although naive!)
- Adaline? - Discriminative