Unsupervised Learning with Scikit-Learn

Jakub Powierza

Konferencja Inżynierii Oprogramowania

GDAŃSK UNIVERSITY OF TECHNOLOGY

Jakub Powierza

- **Student** of Gdańsk University of Technology
- Co-founder & Board Member at **Gradient**
- ML & DL Enthusiast
- Fan of **Computer Vision**
- **MedTagger**'s Team Leader
- **Graphics Software Engineer** at Intel Technology Poland

Agenda

1. Introduction to Machine Learning
2. Clustering using K-Means
3. Anomaly Detection using Gaussian Mixture Models
4. Hierarchical Clustering (*)

(*) Only if we will have enough time 😉

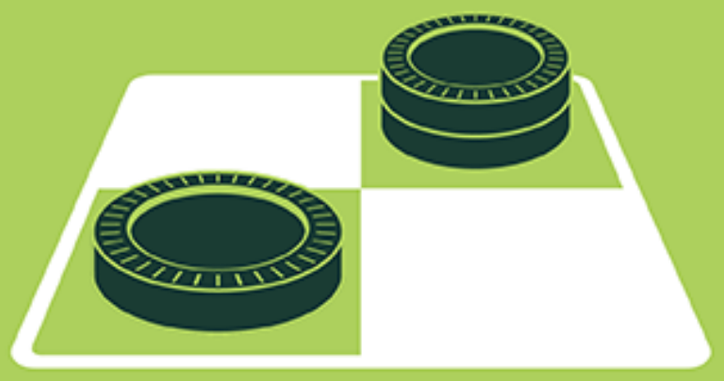
Ask Questions anytime! 😉

Introduction to Machine Learning

Before we will dive deep into Unsupervised Learning…

ARTIFICIAL INTELLIGENCE

Early artificial intelligence stirs excitement.



MACHINE LEARNING

Machine learning begins to flourish.

DEEP LEARNING

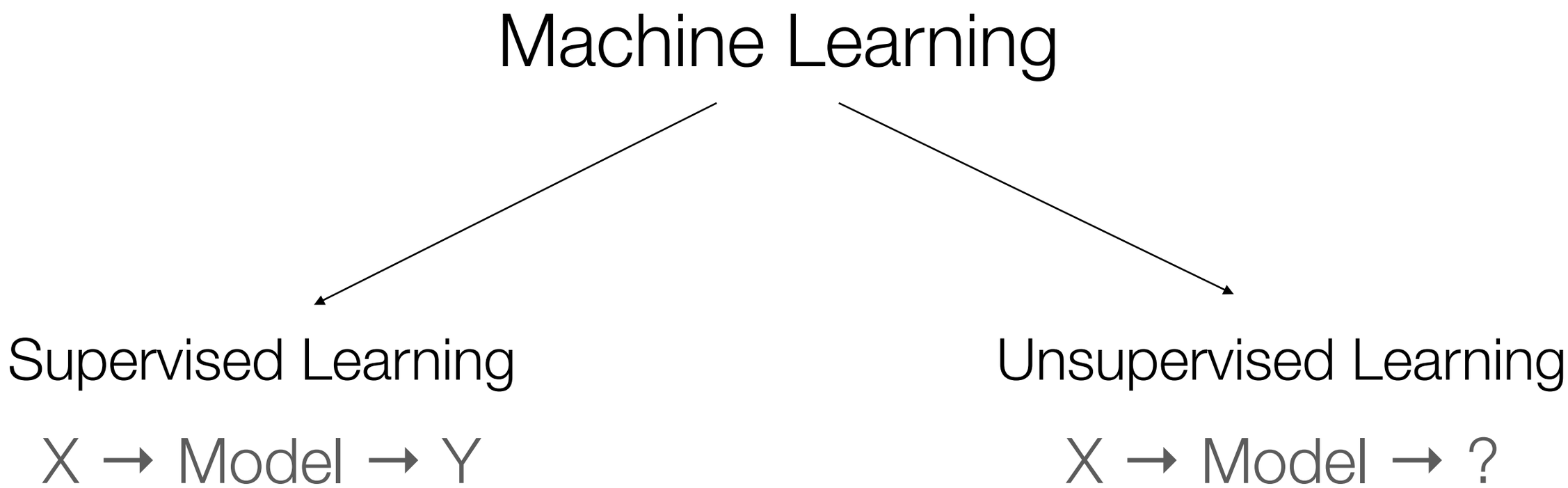
Deep learning breakthroughs drive AI boom.

1950's 1960's 1970's 1980's 1990's 2000's 2010's

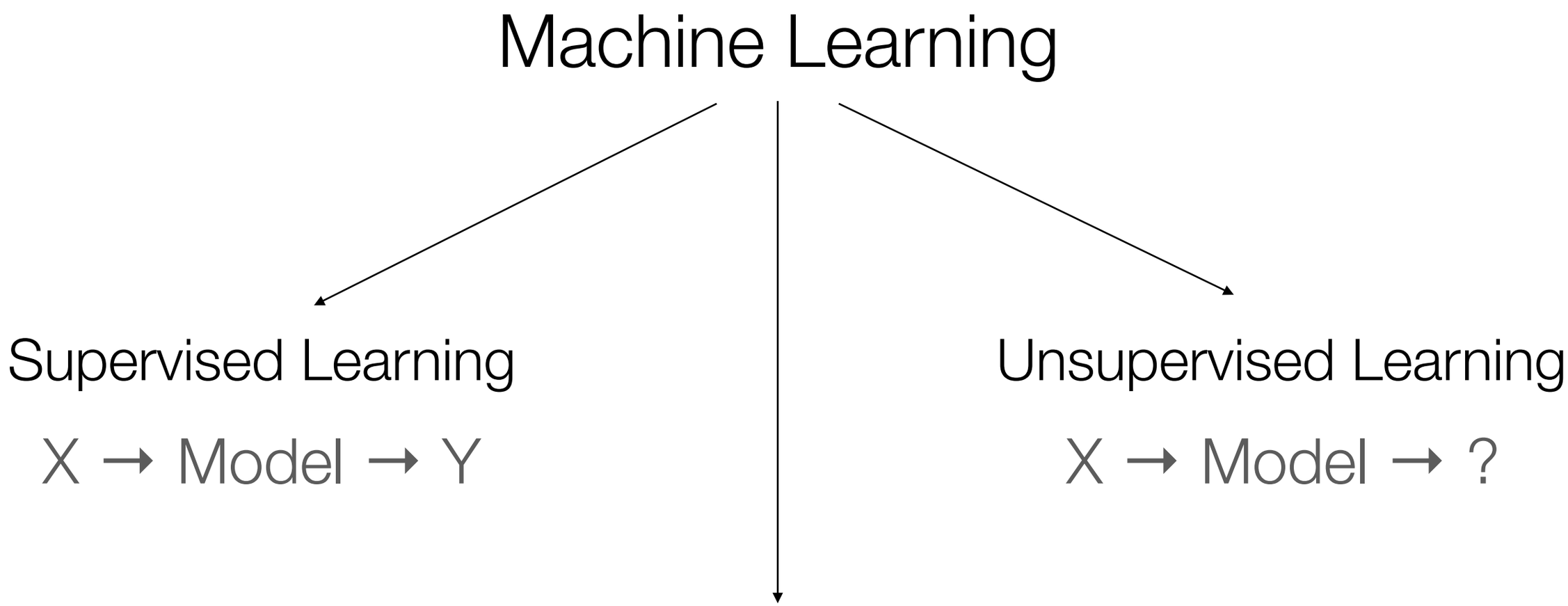
Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

Source: NVidia's blog

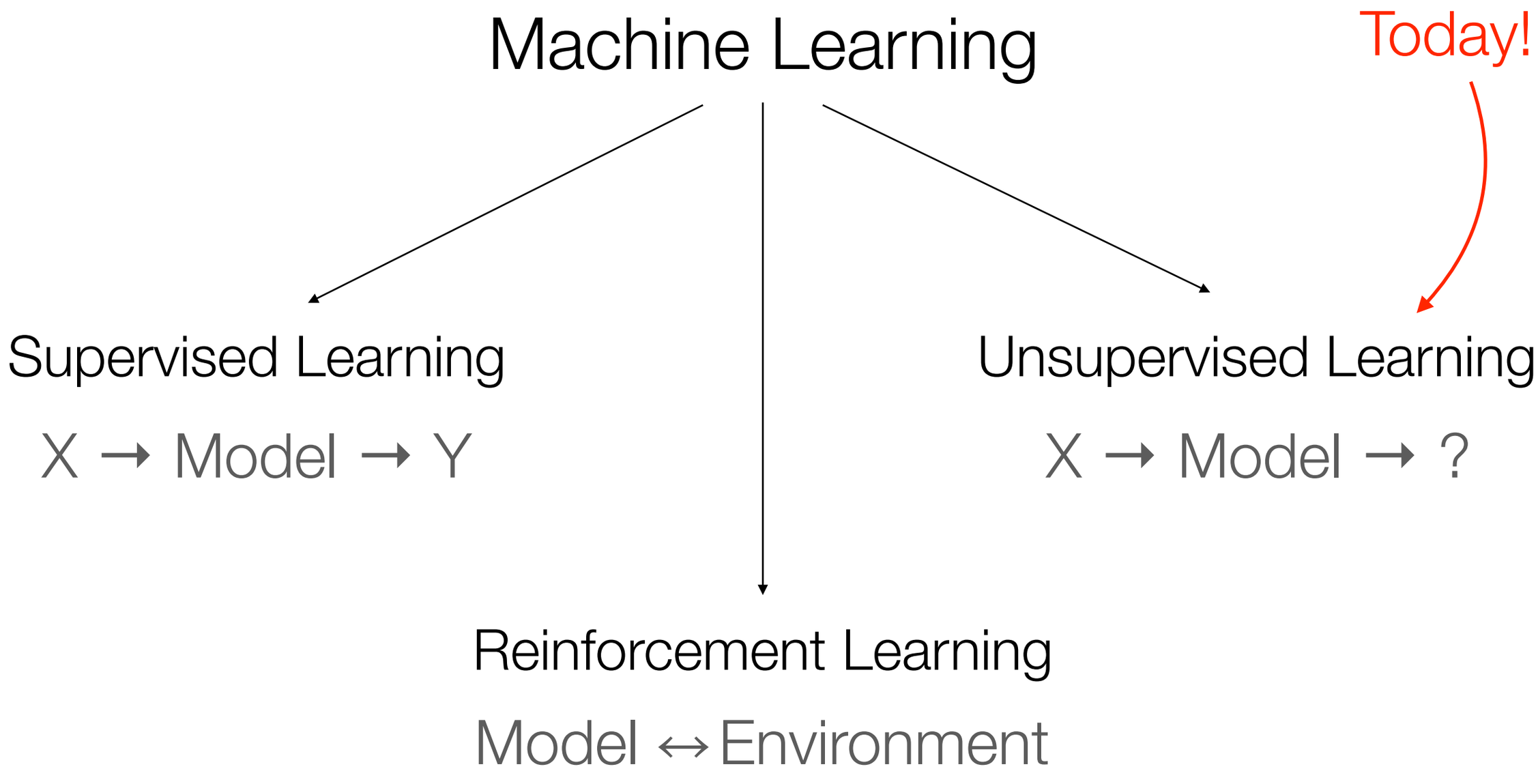
Machine Learning Tasks



Machine Learning Tasks



Machine Learning Tasks



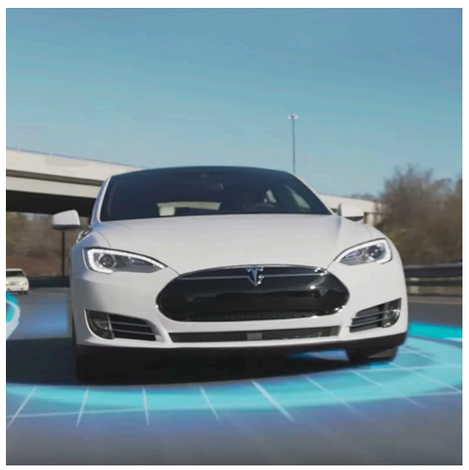
Machine Learning Problems

Supervised

Classification

“This is a cat!”

Regression

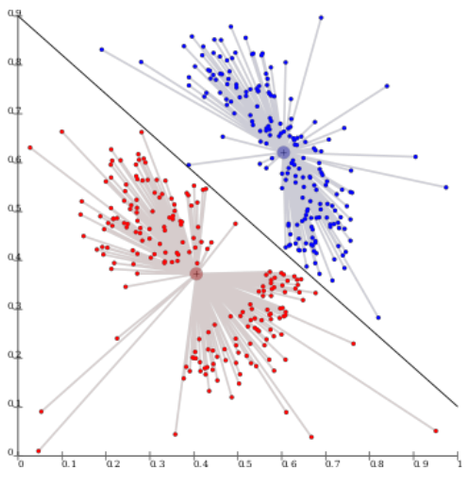


“5 degrees to the left”

Machine Learning Problems

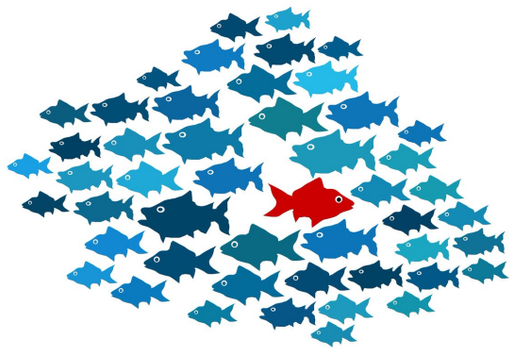
Unsupervised

Clustering



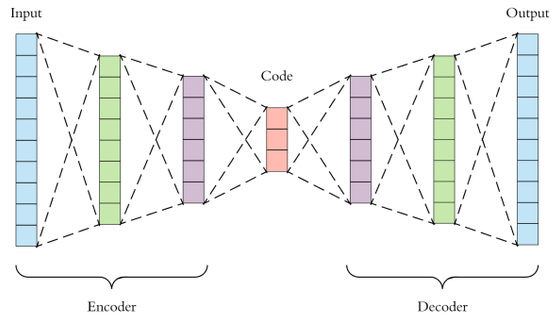
"Looks very similar to others"

Anomaly detection



"Looks abnormal…"

Autoencoders



"Here is the encoding for this example!"

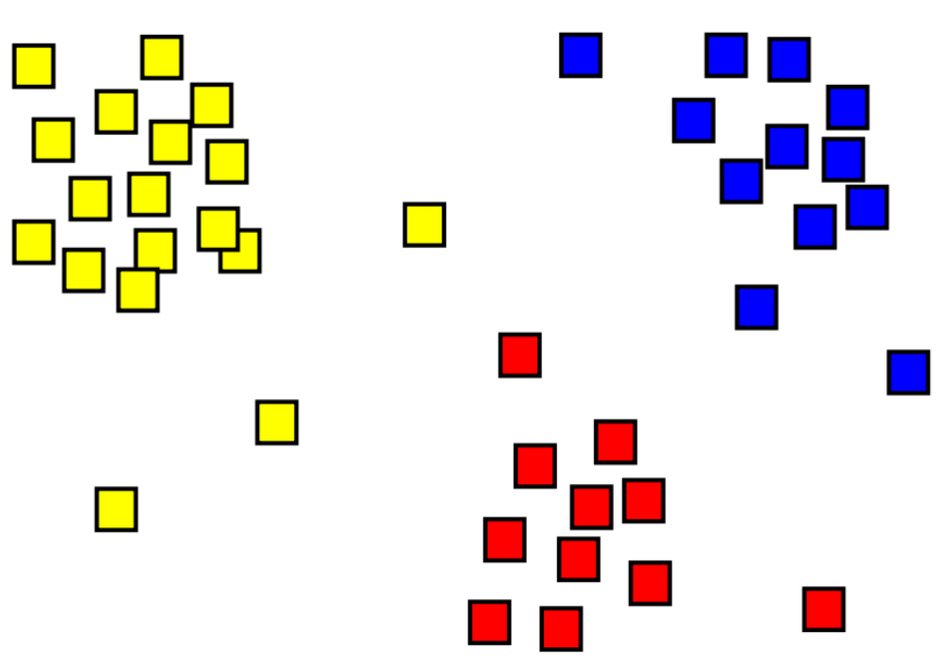
Questions? ☺️

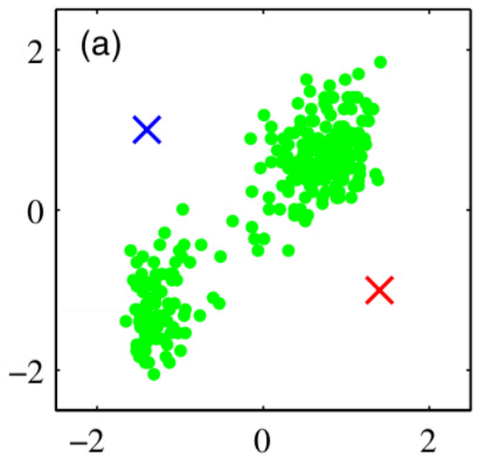
Clustering

A few words about K-Means algorithm

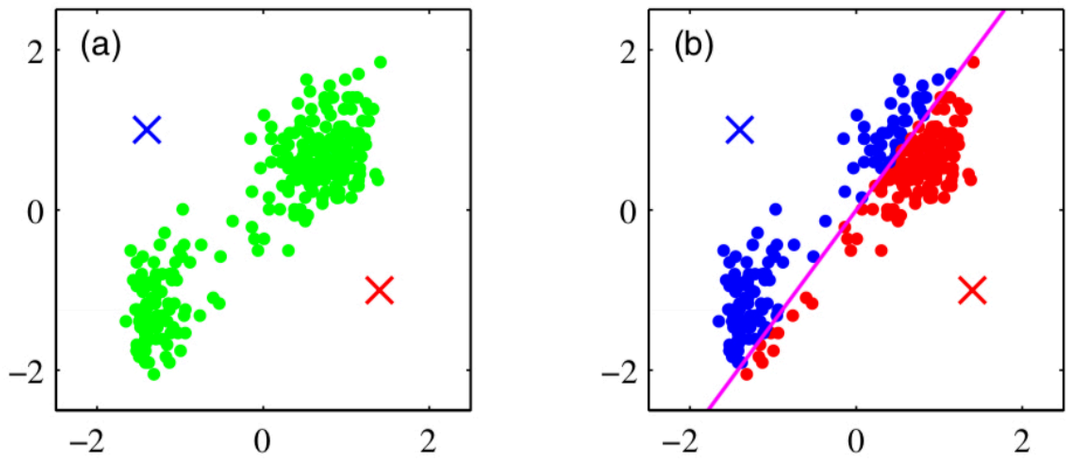
What is clustering?

A clustering algorithm will help you **group the data** into distinct groups, guaranteeing that data points in each group are **similar to each other**.

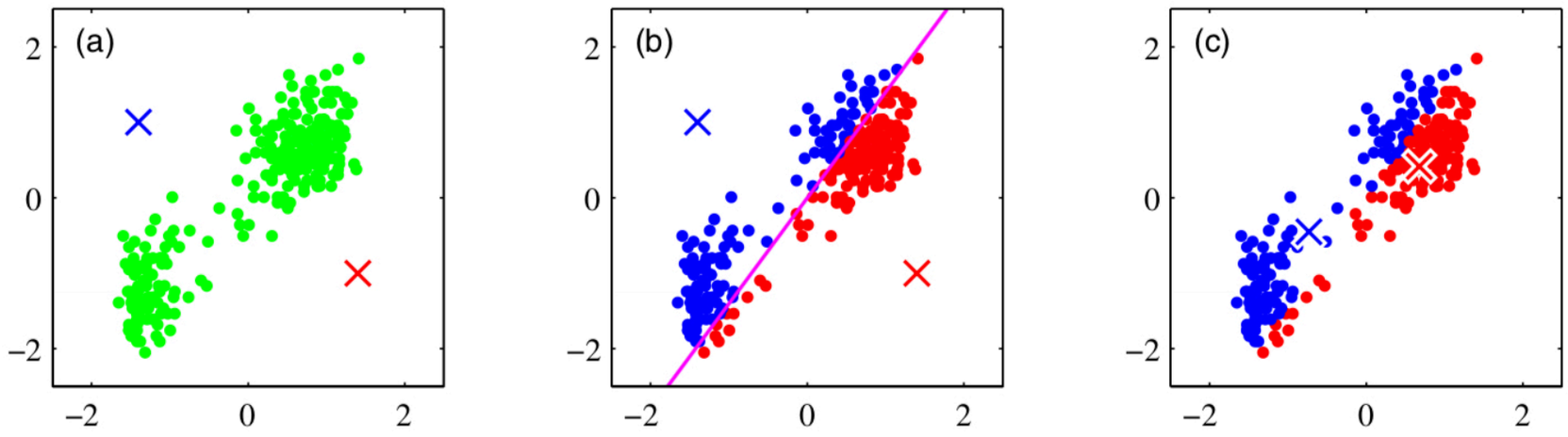




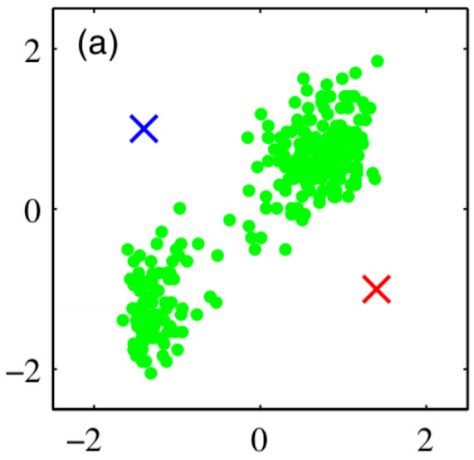
Source: Pattern Recognition and Machine Learning

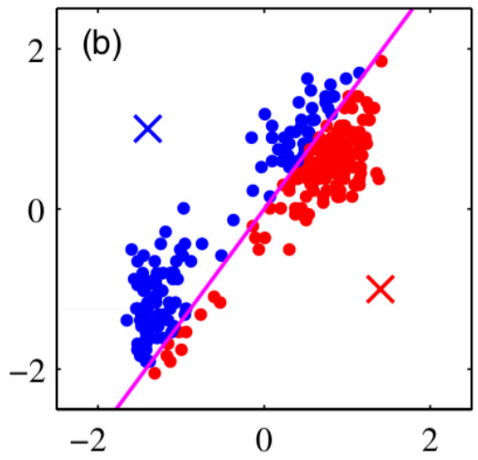


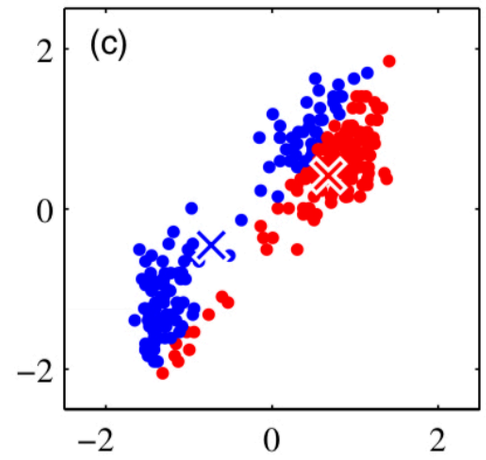
Source: Pattern Recognition and Machine Learning

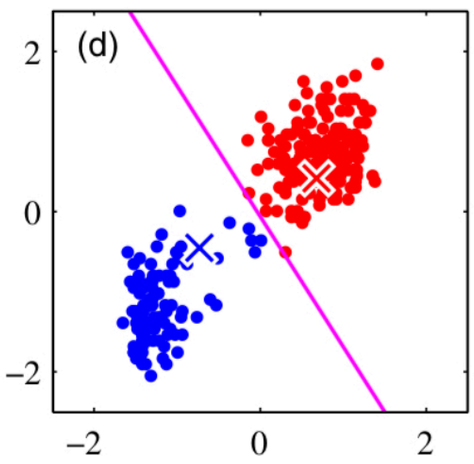


Source: Pattern Recognition and Machine Learning

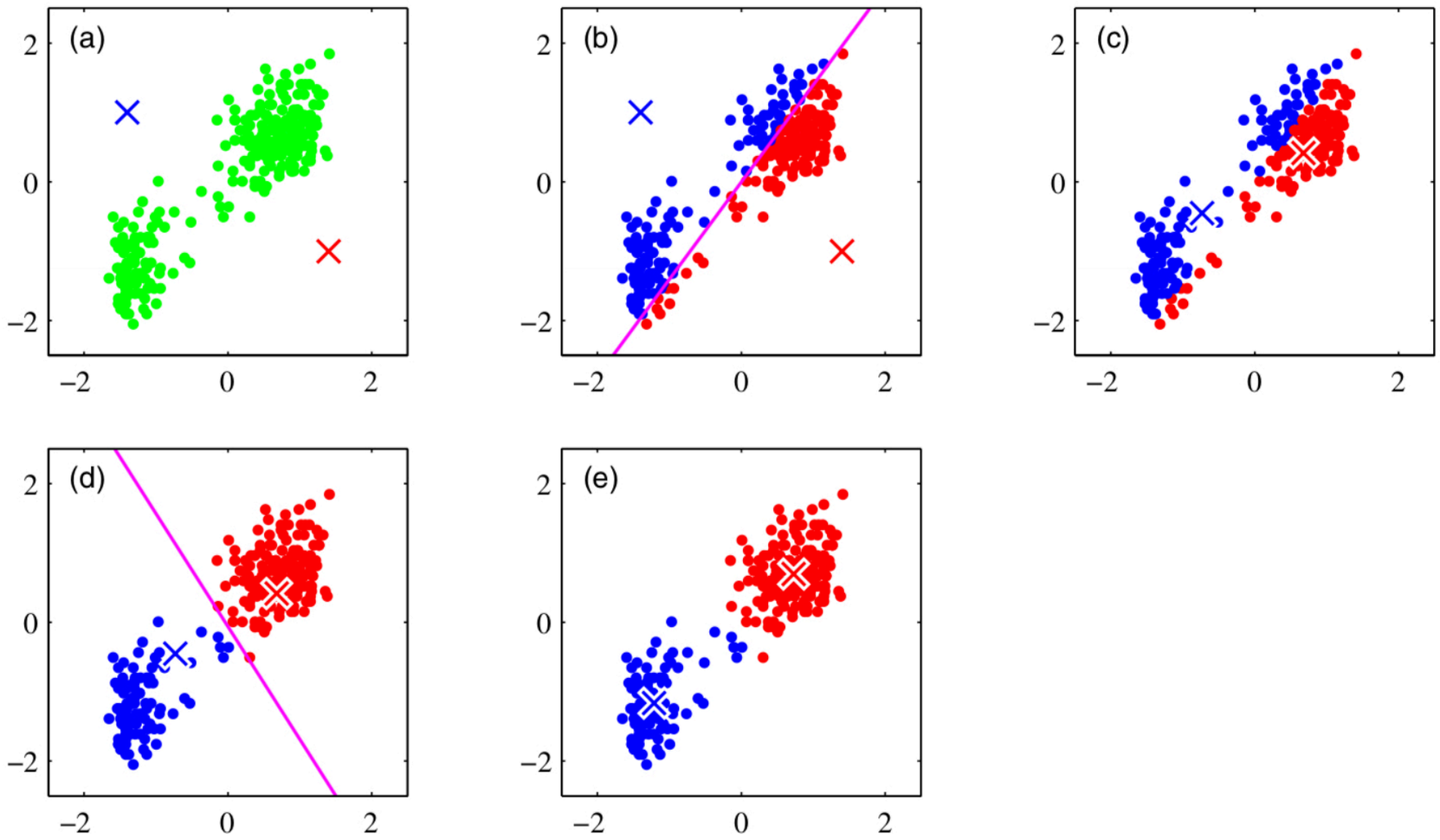




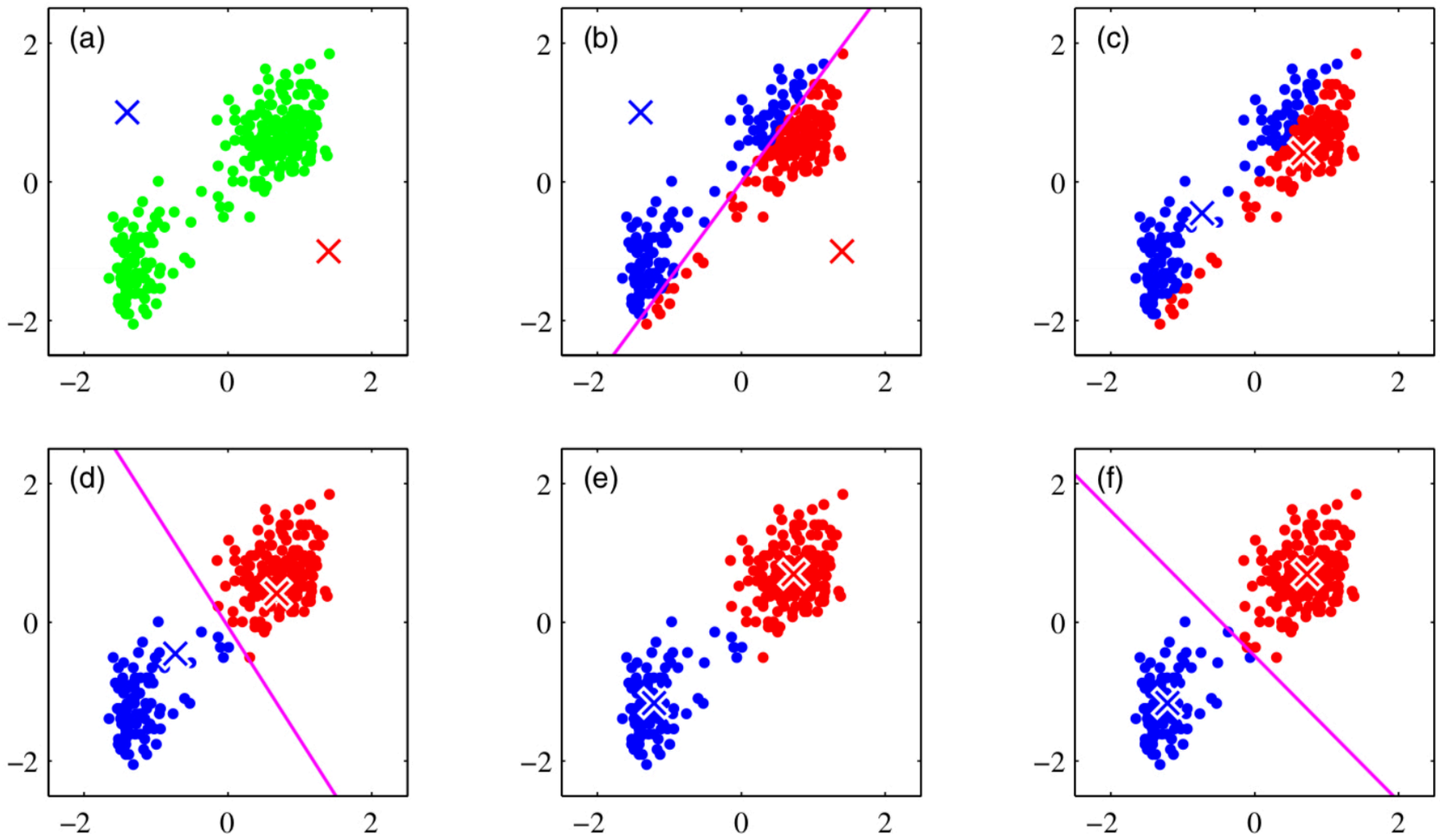




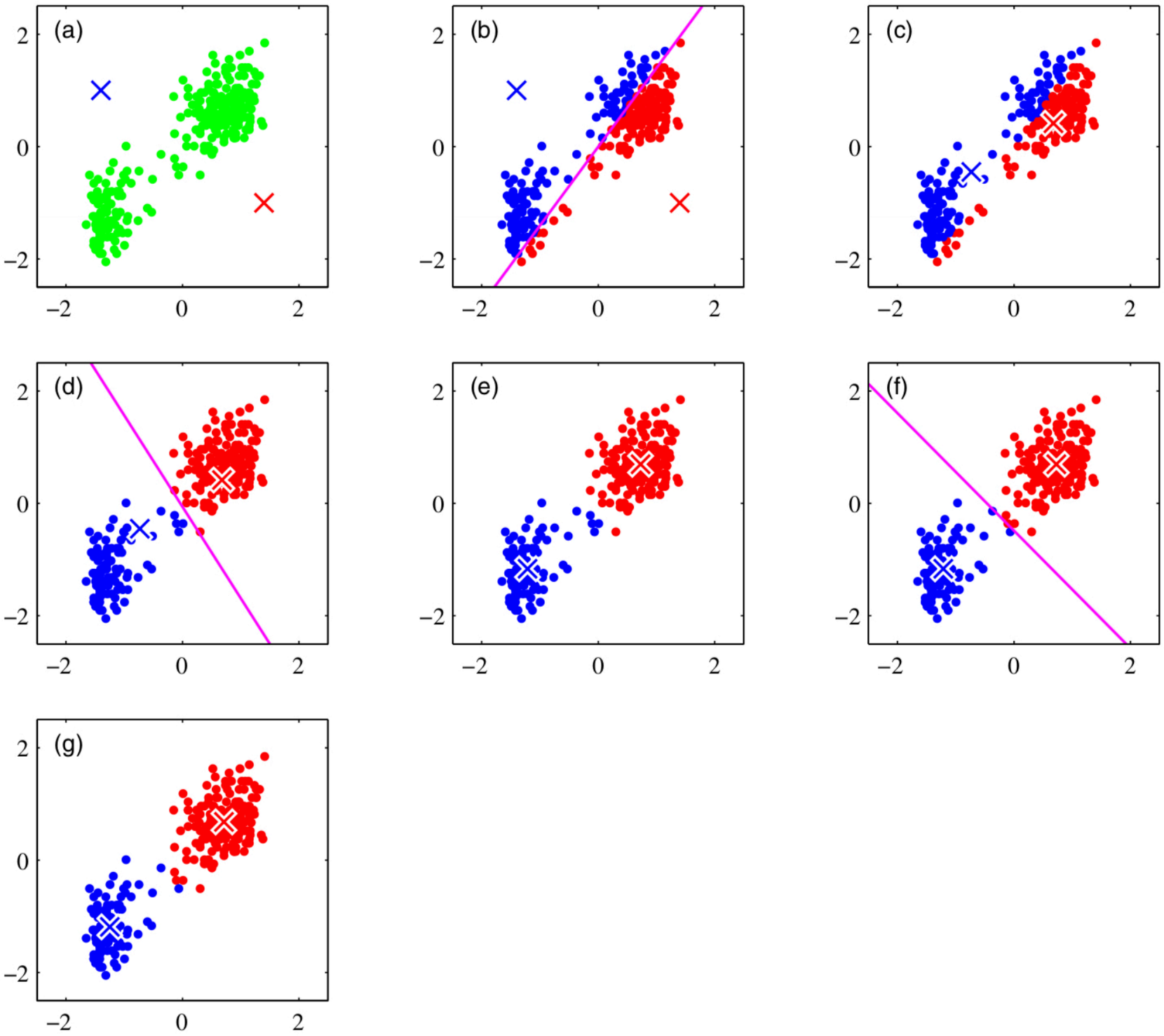
Source: Pattern Recognition and Machine Learning



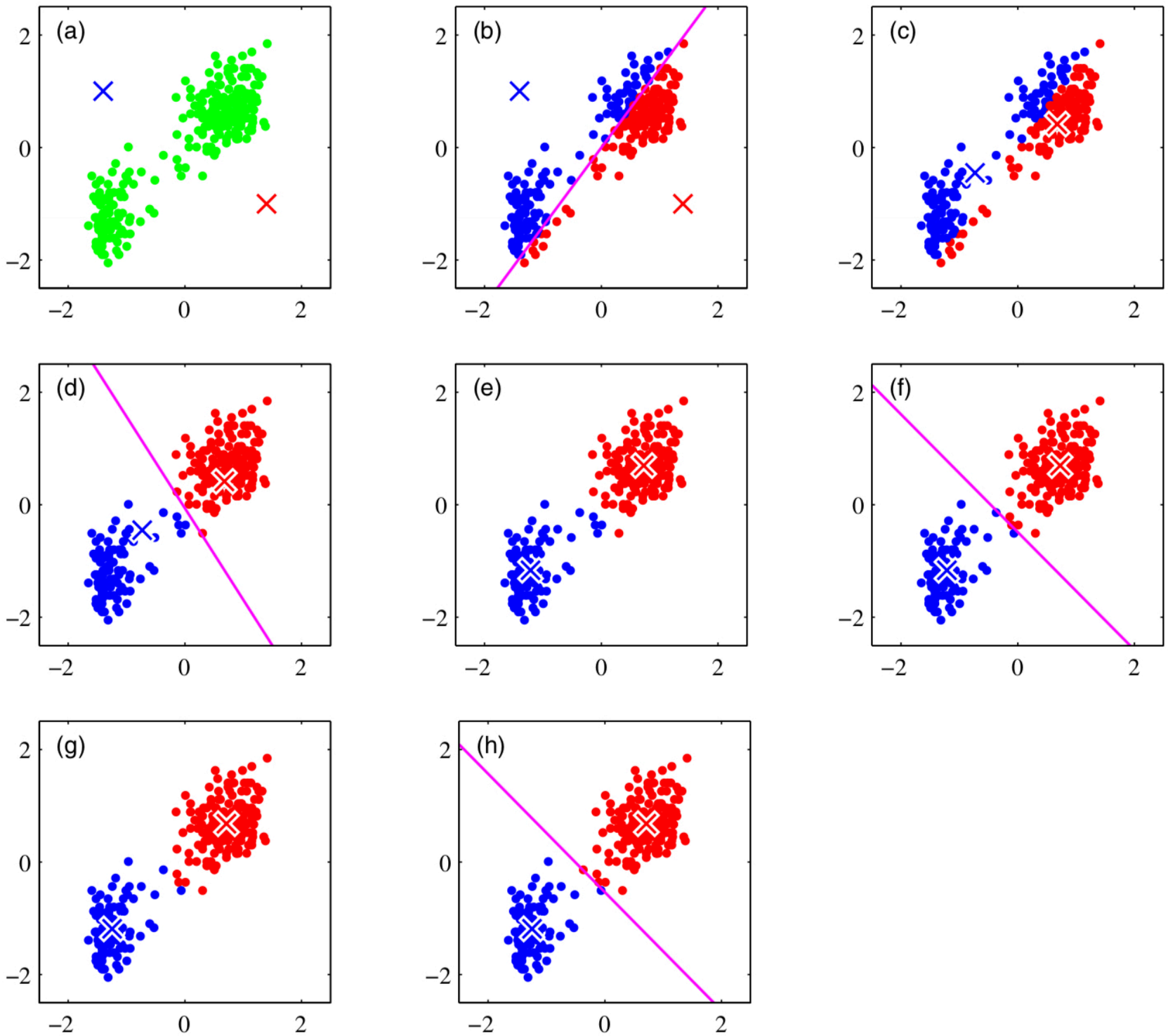
Source: Pattern Recognition and Machine Learning



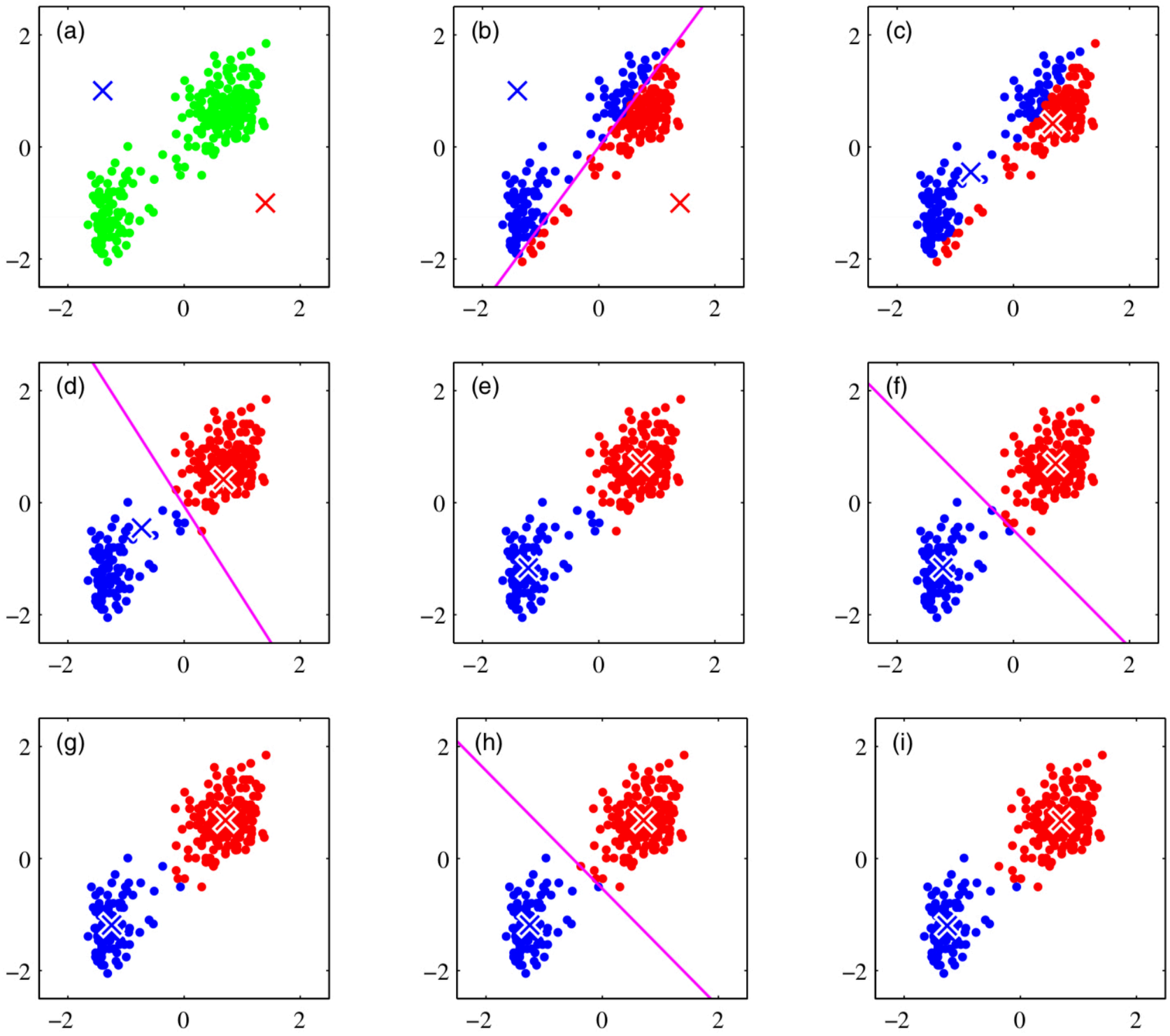
Source: Pattern Recognition and Machine Learning



Source: Pattern Recognition and Machine Learning



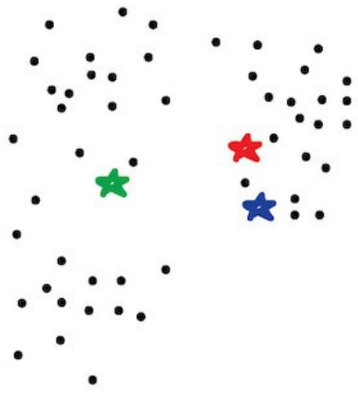
Source: Pattern Recognition and Machine Learning



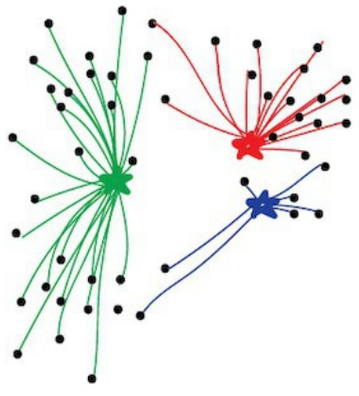
Source: Pattern Recognition and Machine Learning

PUT KEBAB KIOSKS IN THE OPTIMAL WAY

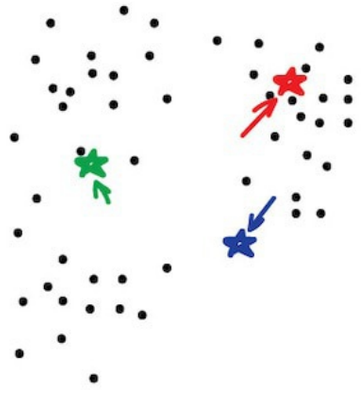
(also illustrating the K-means method)



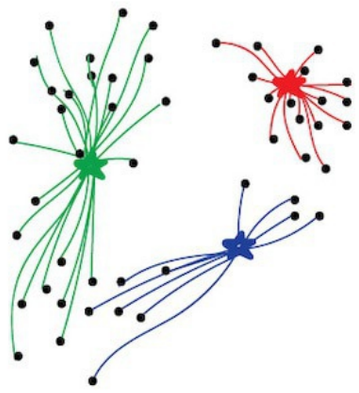
1. Put kebab kiosks in random places in city



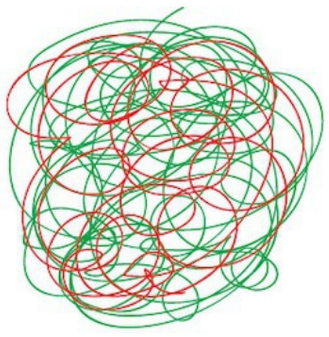
2. Watch how buyers choose the nearest one



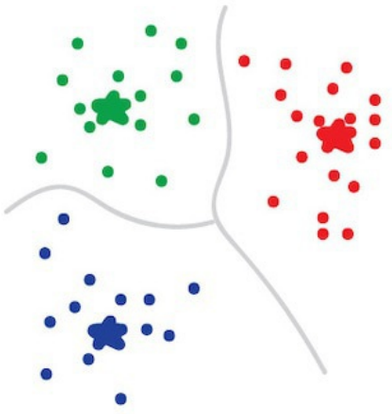
3. Move kiosks closer to the centers of their popularity



4. Watch and move again



5. Repeat a million times



6. Done!
You're god of kebabs!

Source: Vas3k's blog

K-Means Algorithm

1. Select K clusters randomly,
2. For each point select the closest cluster,
3. Update your clusters by taking mean value of all your data points,
4. Check for convergence and move to Step 2 if needed.

How to measure the distance?

Euclidean distance

$$\|a - b\|_2 = \sqrt{\sum_i (a_i - b_i)^2}$$

Squared Euclidean distance

$$\|a - b\|_2^2 = \sum_i (a_i - b_i)^2$$

Manhattan distance

$$\|a - b\|_1 = \sum_i |a_i - b_i|$$

Why/Why not to use K-Means Clustering?

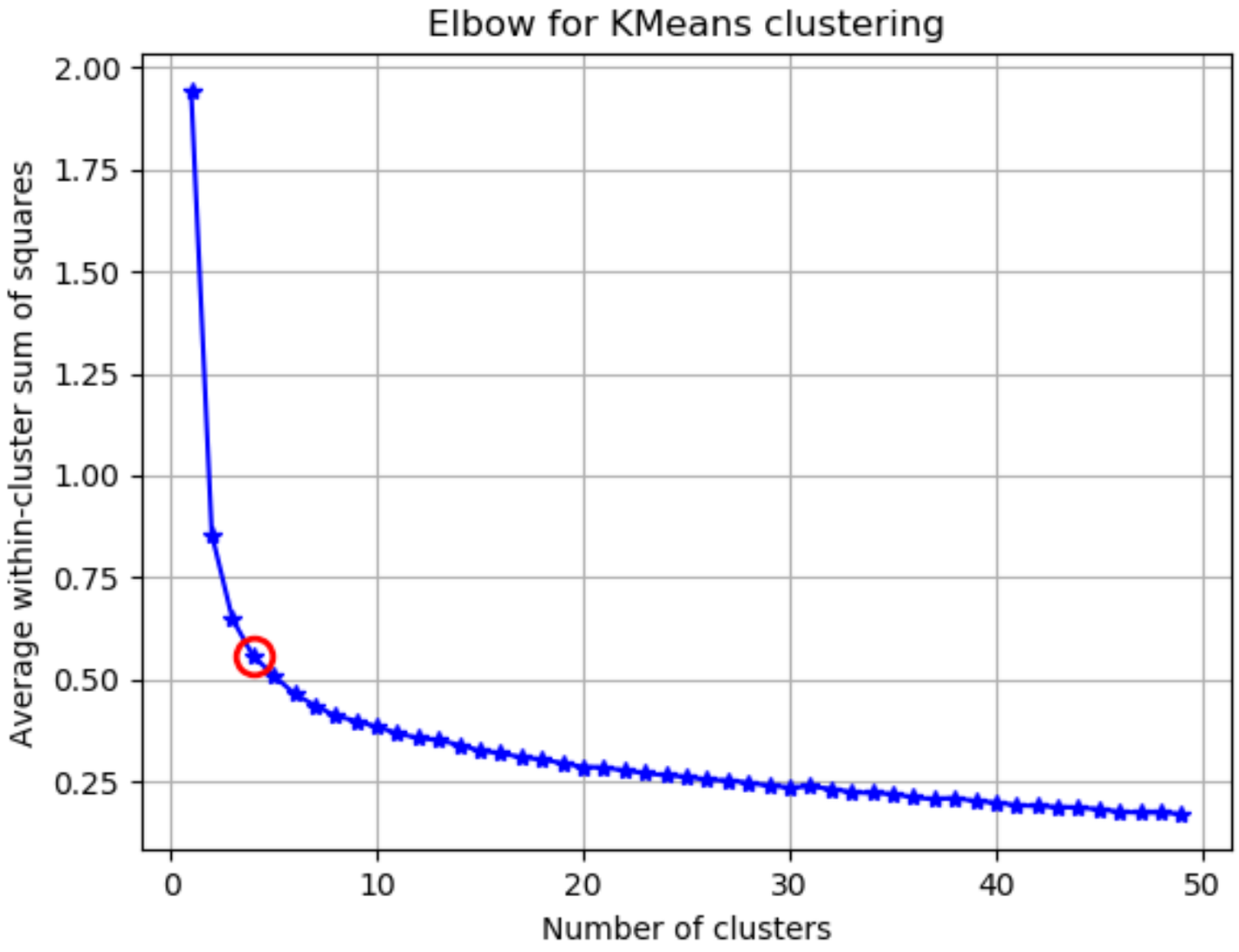
Advantages:

- Easy to implement,
- Relatively fast and efficient,
- Only one hyperparameter - K.

Disdvantages:

- Random points at the beginning,
- Hyperparameter for number of clusters.

How to choose the number of clusters?



Questions? ☺️

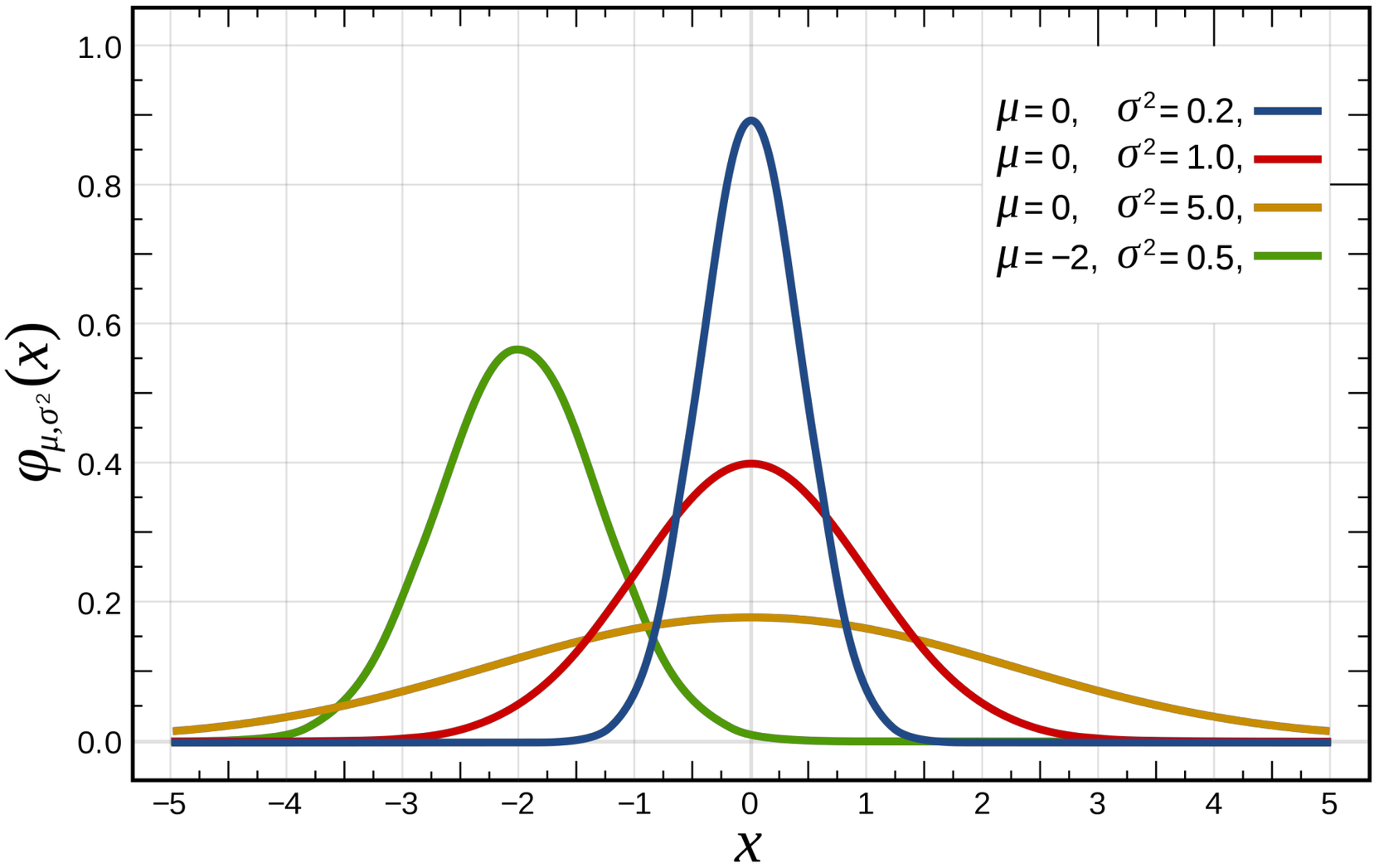
http://bit.ly/2OZetLC

Let’s use it in practice! 🚀

Anomaly Detection

Gaussian Mixture Models

Gaussian Distribution as a base

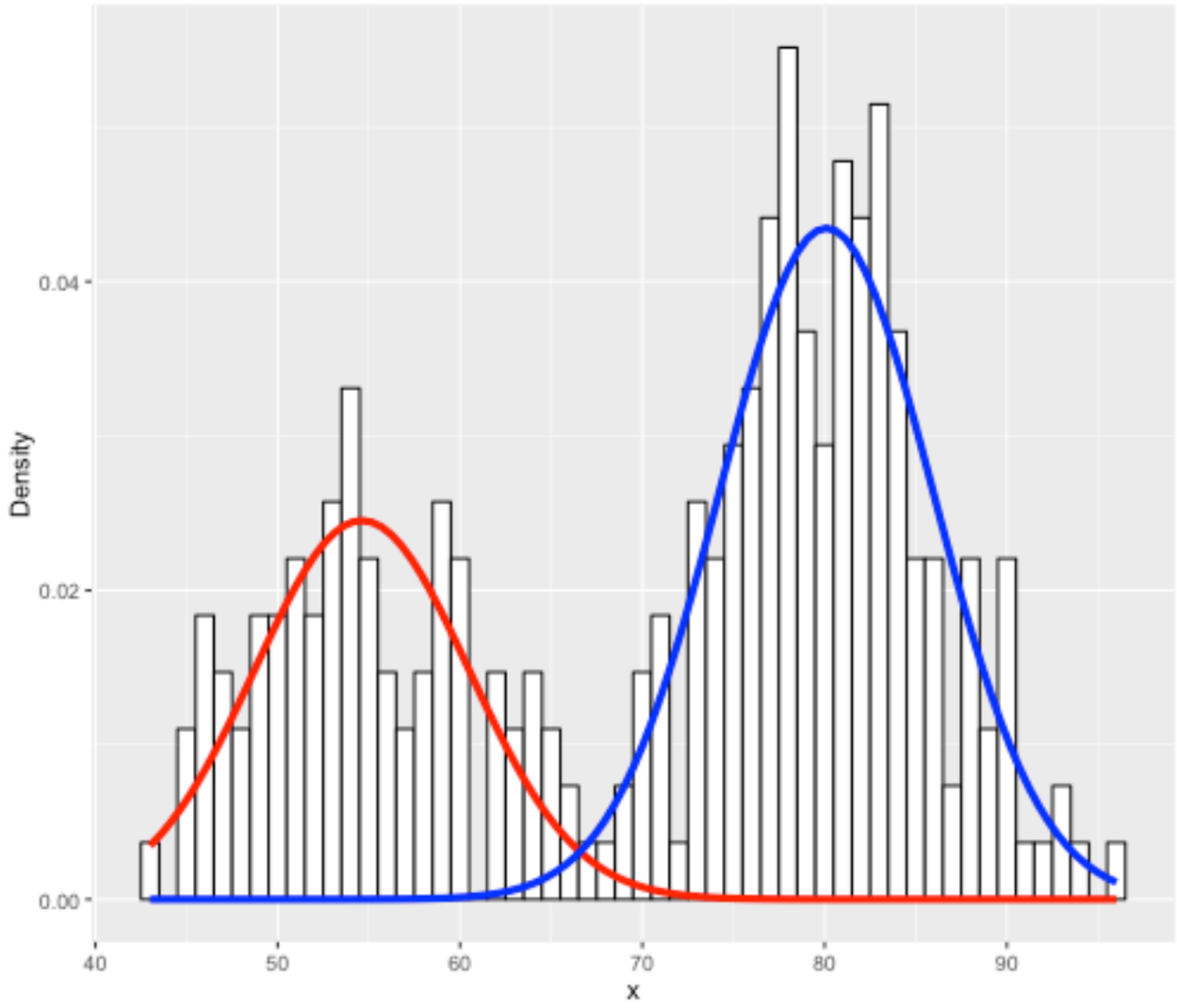


Source: Wikipedia

Mixture of Gaussians

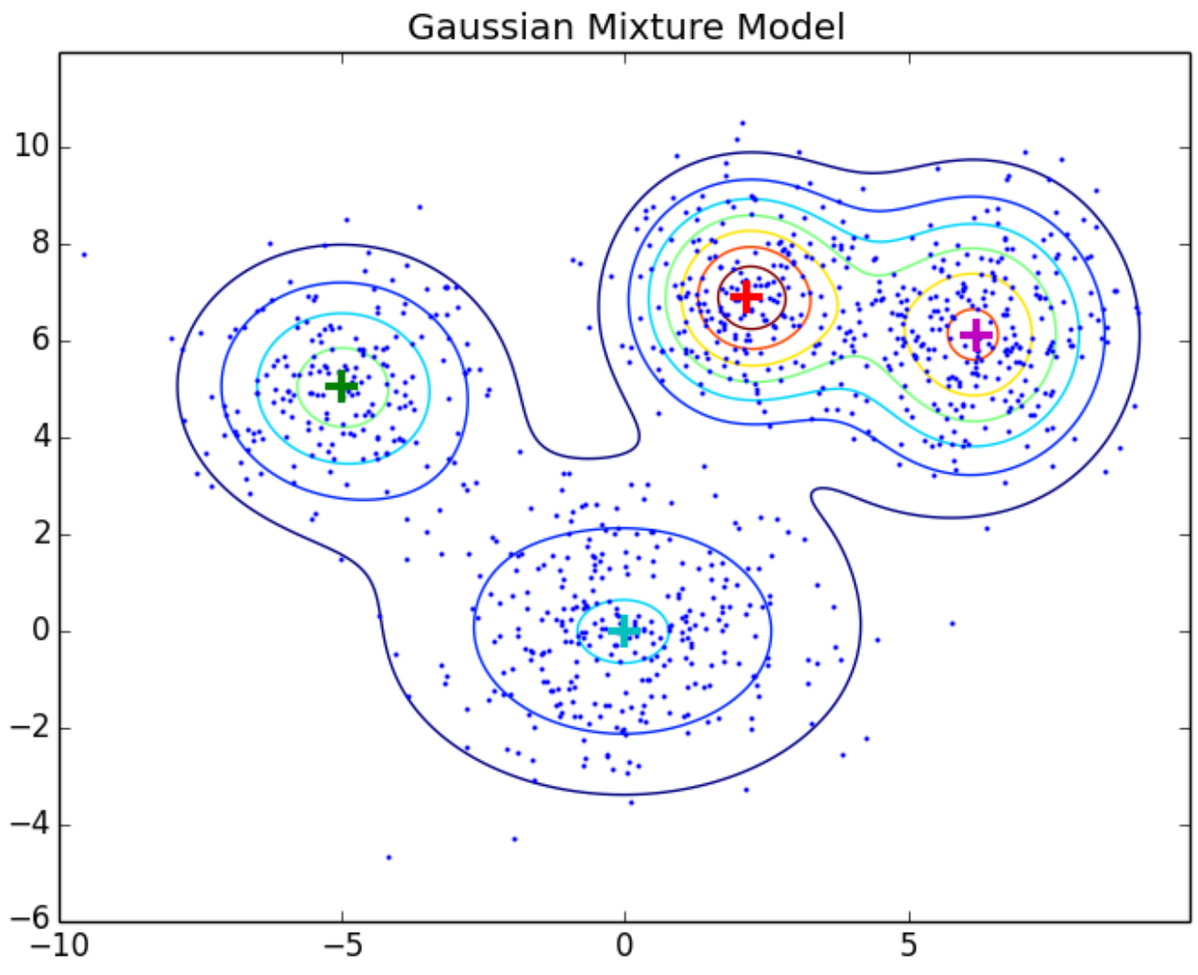
Source: Angus Turner's blog

Mixture of Gaussians

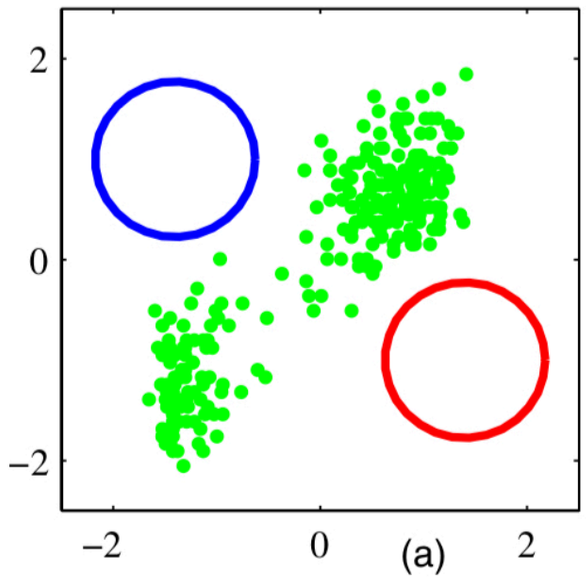


Source: Fong Chun Chan's blog

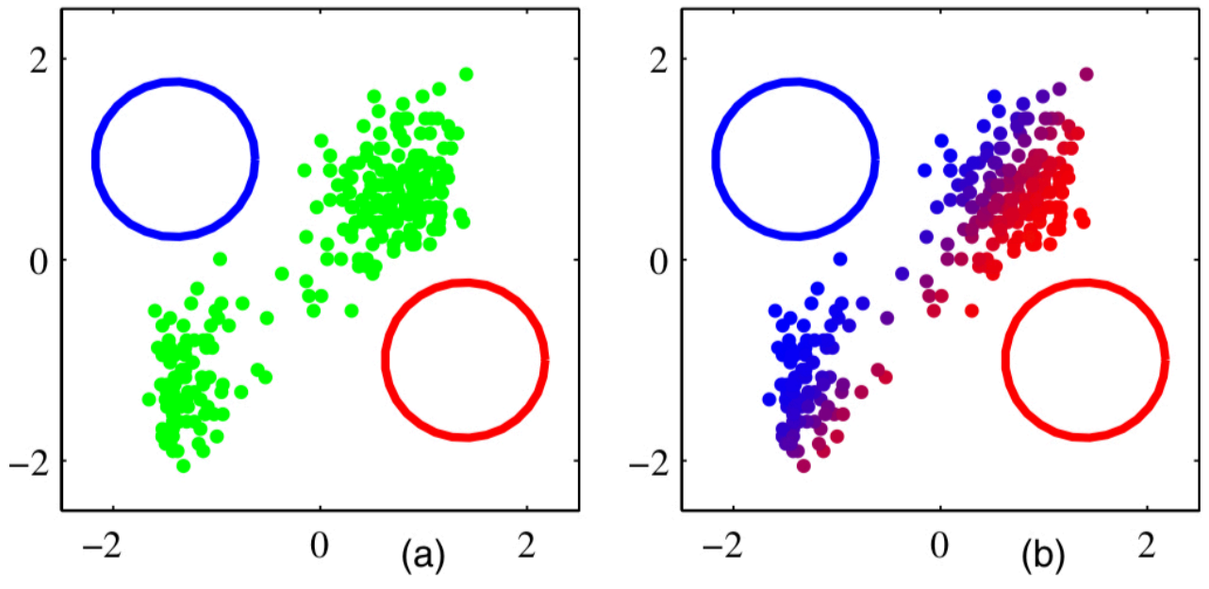
Mixture of Gaussians



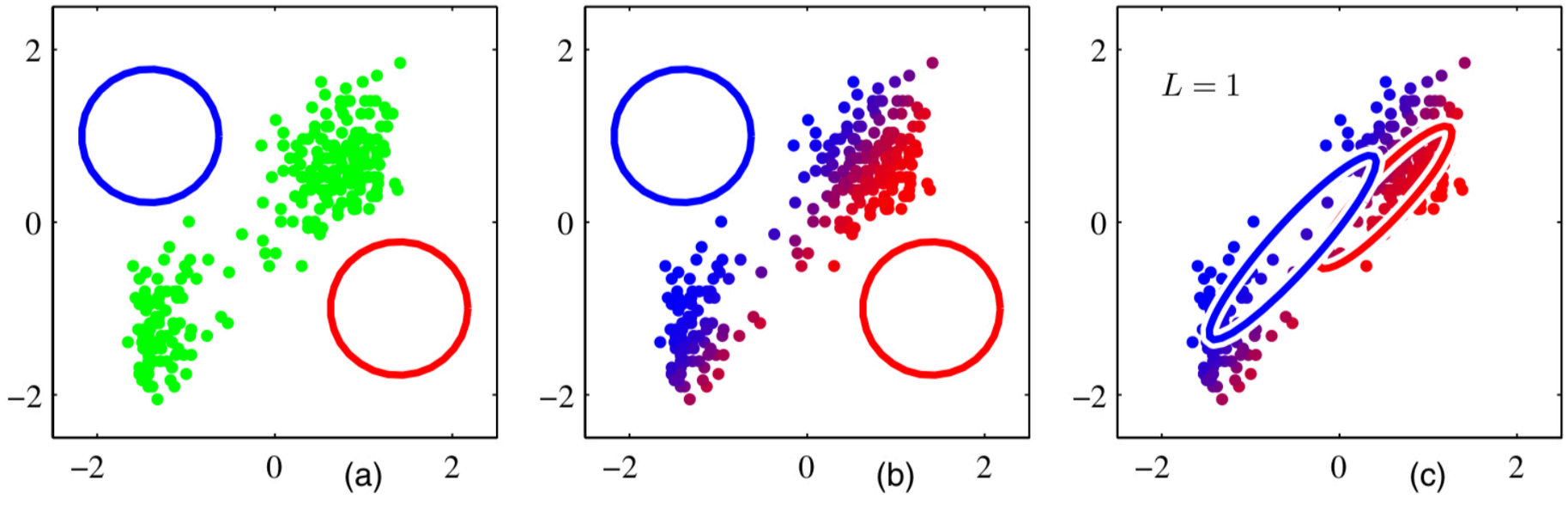
Source: Yu's Machine Learning Garden blog



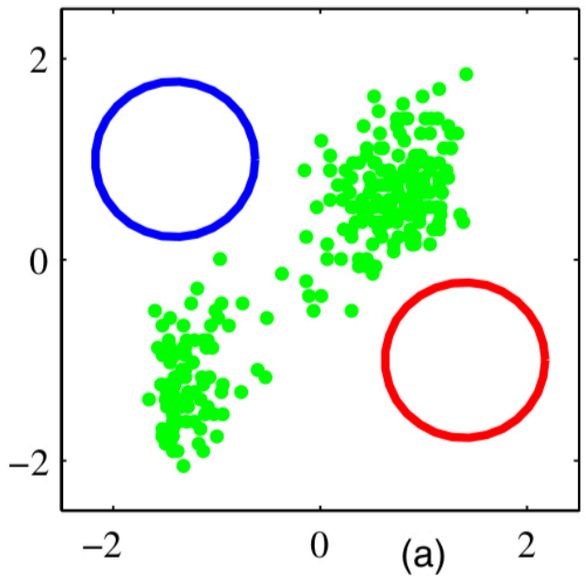
Source: Pattern Recognition and Machine Learning

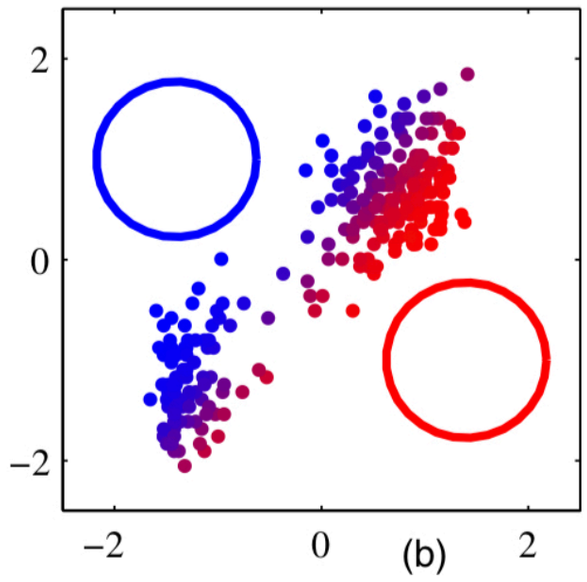


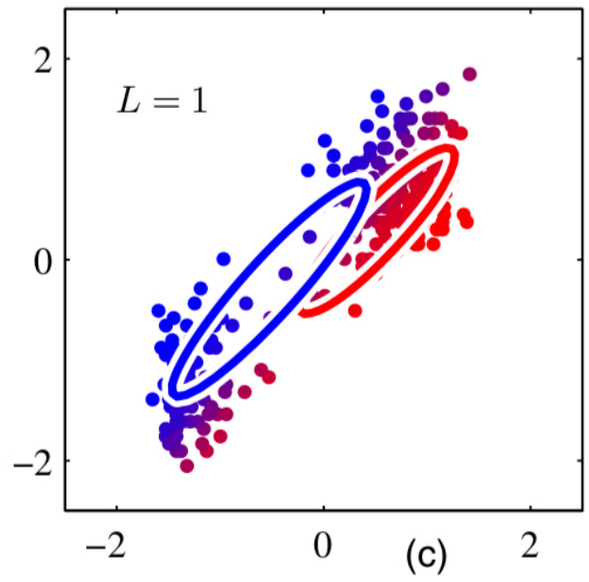
Source: Pattern Recognition and Machine Learning

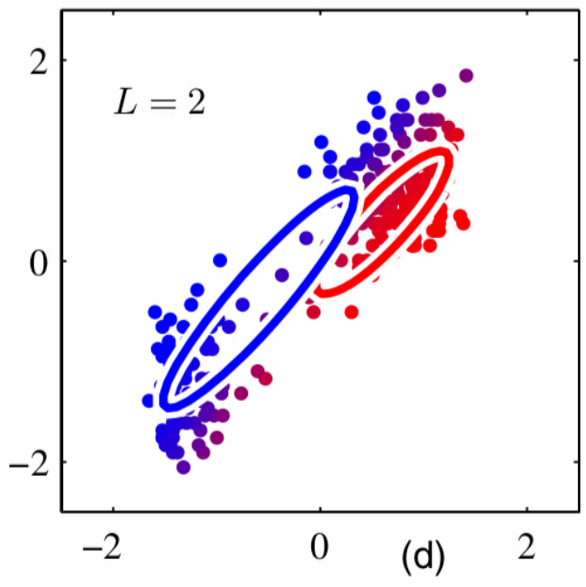


Source: Pattern Recognition and Machine Learning

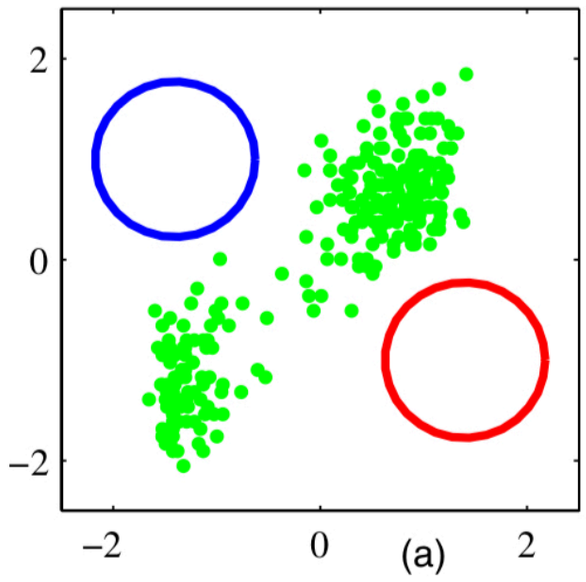


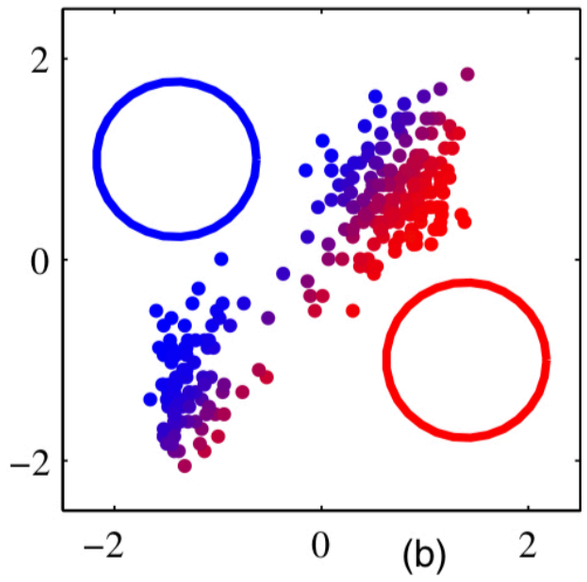


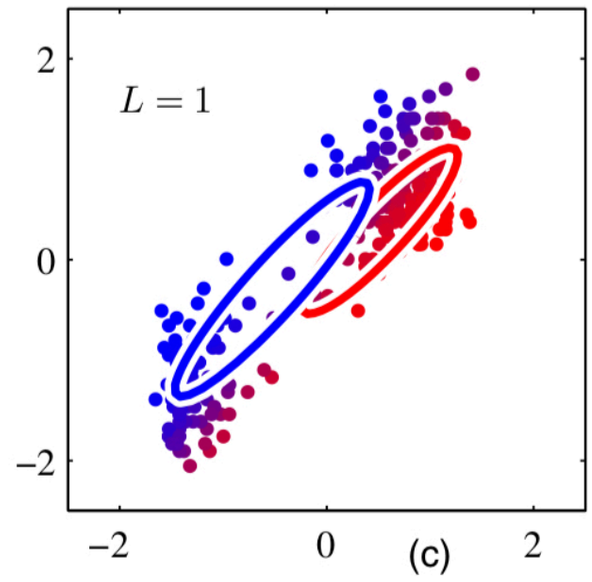


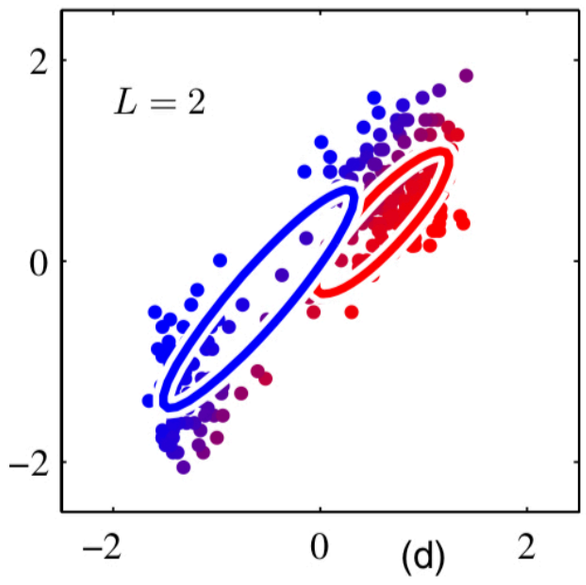


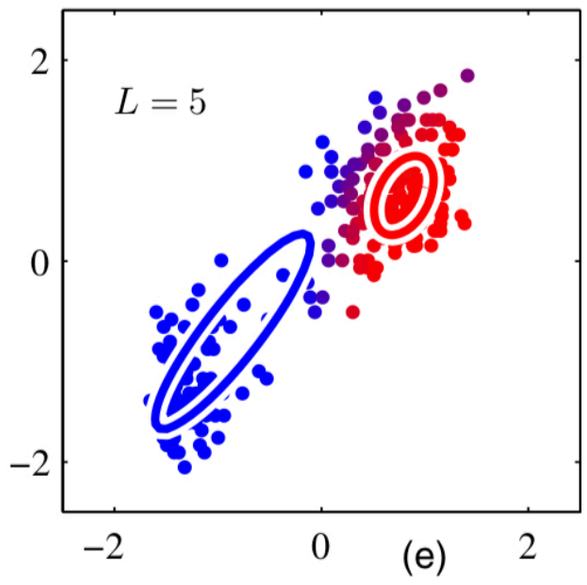
Source: Pattern Recognition and Machine Learning



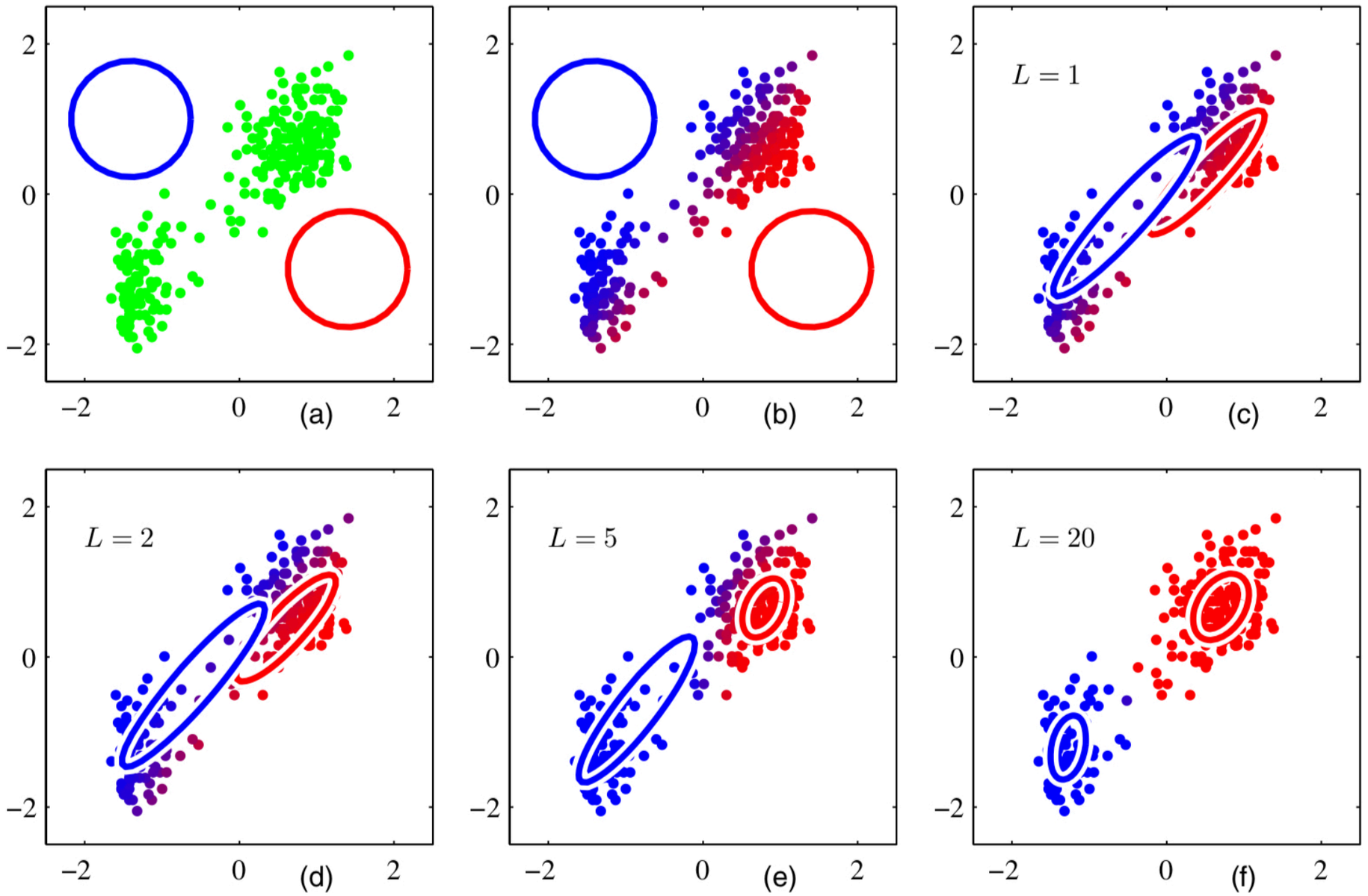








Source: Pattern Recognition and Machine Learning



Source: Pattern Recognition and Machine Learning

Gaussian Mixture Models Algorithm

E Step

$$w_{ik} = \frac{\omega_k \, N(x_i \mid \mu_k, \Sigma_k^2)}{\sum_{m=1}^{K} \omega_m \, N(x_i \mid \mu_m, \Sigma_m^2)}$$

Gaussian Mixture Models Algorithm

E Step

$$w_{ik} = \frac{\omega_k \, N(x_i \mid \mu_k, \Sigma_k^2)}{\sum_{m=1}^{K} \omega_m \, N(x_i \mid \mu_m, \Sigma_m^2)}$$

M Step

$$N_k = \sum_{n=1}^{N} w_{ik}$$

Gaussian Mixture Models Algorithm

E Step

$$w_{ik} = \frac{\omega_k \, N(x_i \mid \mu_k, \Sigma_k^2)}{\sum_{m=1}^{K} \omega_m \, N(x_i \mid \mu_m, \Sigma_m^2)}$$

M Step

$$N_k = \sum_{n=1}^{N} w_{ik}$$

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} x_i$$

Gaussian Mixture Models Algorithm

E Step

$$w_{ik} = \frac{\omega_k \, N(x_i \mid \mu_k, \Sigma_k^2)}{\sum_{m=1}^{K} \omega_m \, N(x_i \mid \mu_m, \Sigma_m^2)}$$

M Step

$$N_k = \sum_{n=1}^{N} w_{ik}$$

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} x_i$$

$$\Sigma_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} (x_i - \mu_k)(x_i - \mu_k)^T$$

Gaussian Mixture Models Algorithm

E Step

$$w_{ik} = \frac{\omega_k \, N(x_i \mid \mu_k, \Sigma_k^2)}{\sum_{m=1}^{K} \omega_m \, N(x_i \mid \mu_m, \Sigma_m^2)}$$

M Step

$$N_k = \sum_{n=1}^{N} w_{ik}$$

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} x_i$$

$$\Sigma_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} (x_i - \mu_k)(x_i - \mu_k)^T$$

$$\omega_k = \frac{N_k}{N}$$

Gaussian Mixture Models Algorithm

Repeat till convergence

E Step

$$w_{ik} = \frac{\omega_k \, N(x_i \mid \mu_k, \Sigma_k^2)}{\sum_{m=1}^{K} \omega_m \, N(x_i \mid \mu_m, \Sigma_m^2)}$$

M Step

$$N_k = \sum_{n=1}^{N} w_{ik}$$

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} x_i$$

$$\Sigma_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} (x_i - \mu_k)(x_i - \mu_k)^T$$

$$\omega_k = \frac{N_k}{N}$$

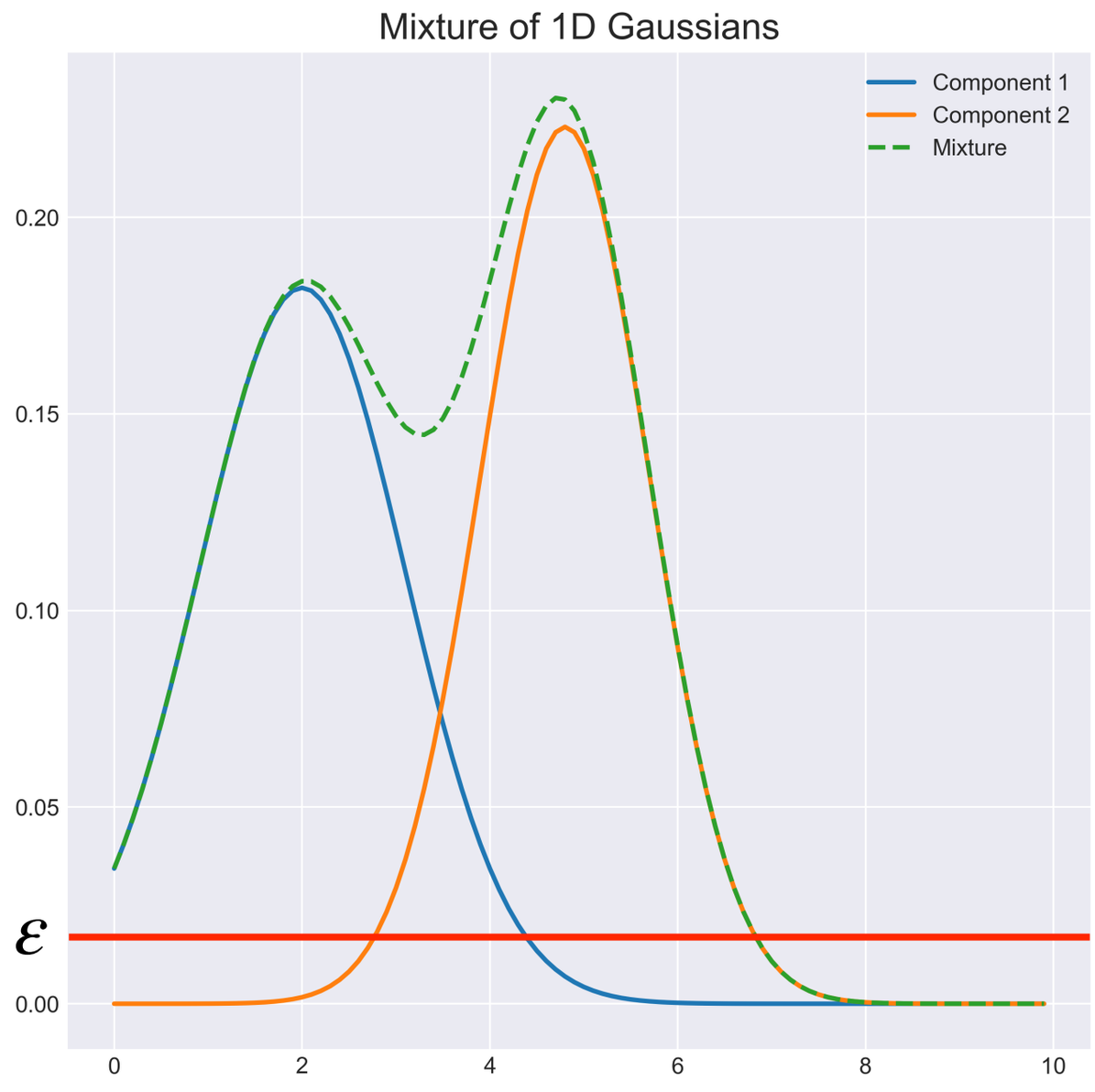
Anomaly Detection

Anomaly:

$p(x) < \varepsilon$

Expected:

$p(x) \geq \varepsilon$



Source: Angus Turner's blog

Why/Why not to use GMMs?

Advantages:

- Mixed membership based on probability,
- More flexible in terms of covariance.

Disdvantages:

- Optimization is not so trivial…
- May not work for datasets that doesn’t have “hidden” Gaussian distribution.

Questions? ☺️

http://bit.ly/2D6xA1y

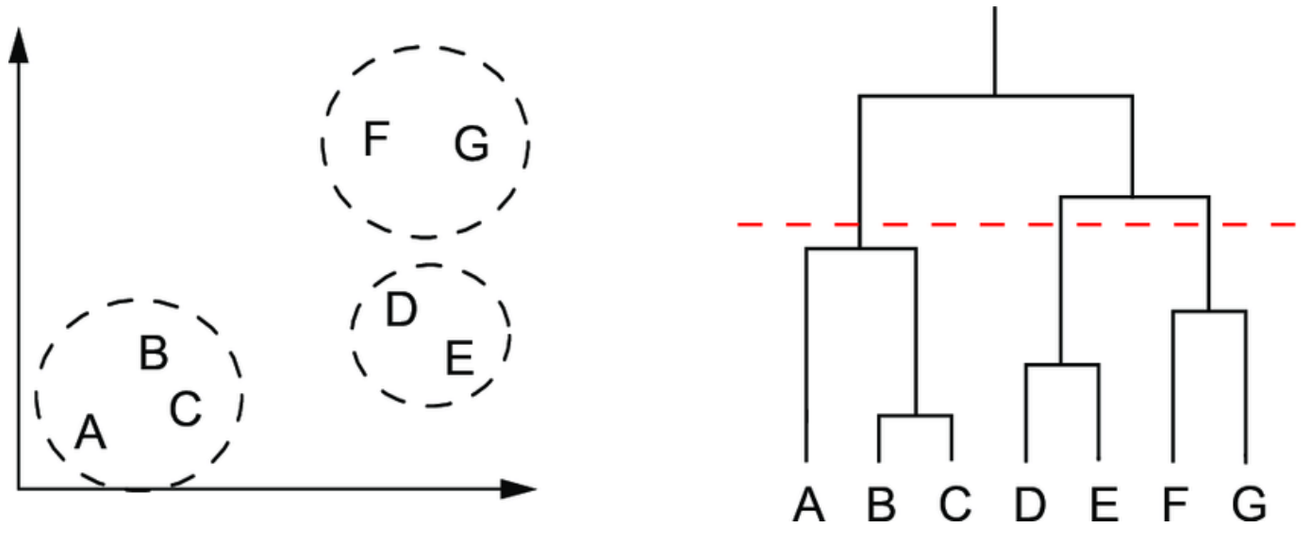
Let’s use it in practice! 🚀

Hierarchical Clustering

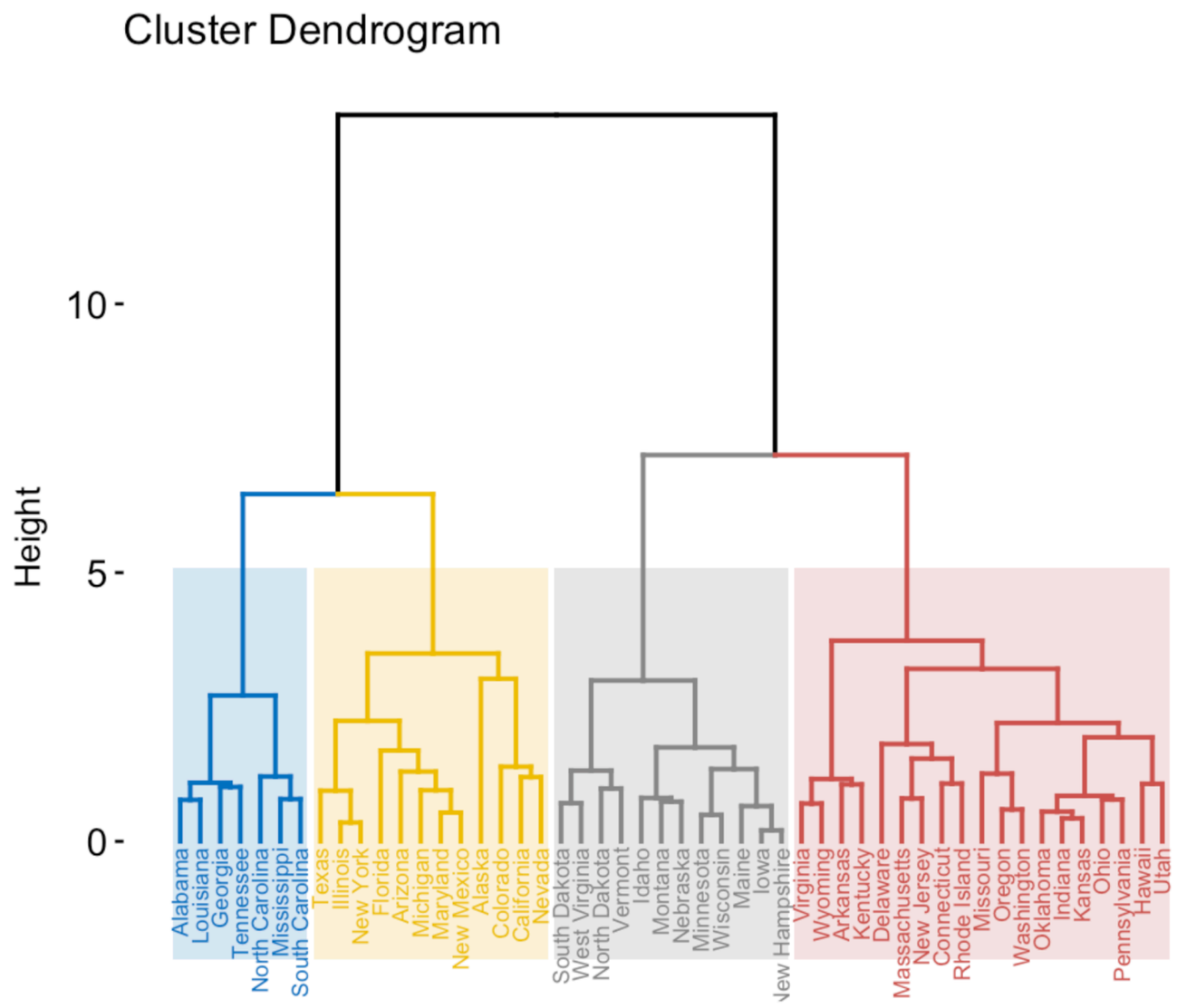
Is it really different than K-Means?

Hierarchical Clustering

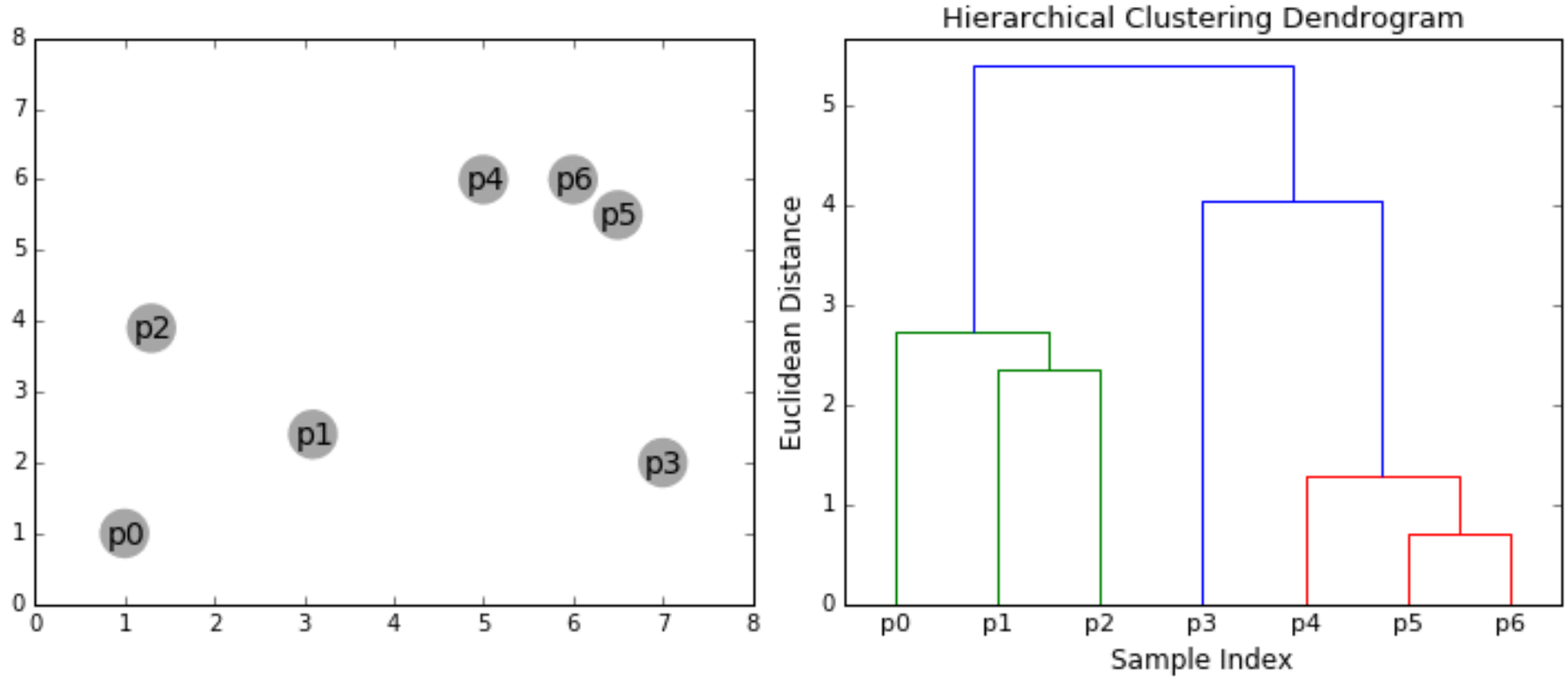
- Hierarchy of clusters is represented as a **tree**,
- **Root** is a unique cluster that gathers all samples,
- **Leaves** are clusters with only one sample.



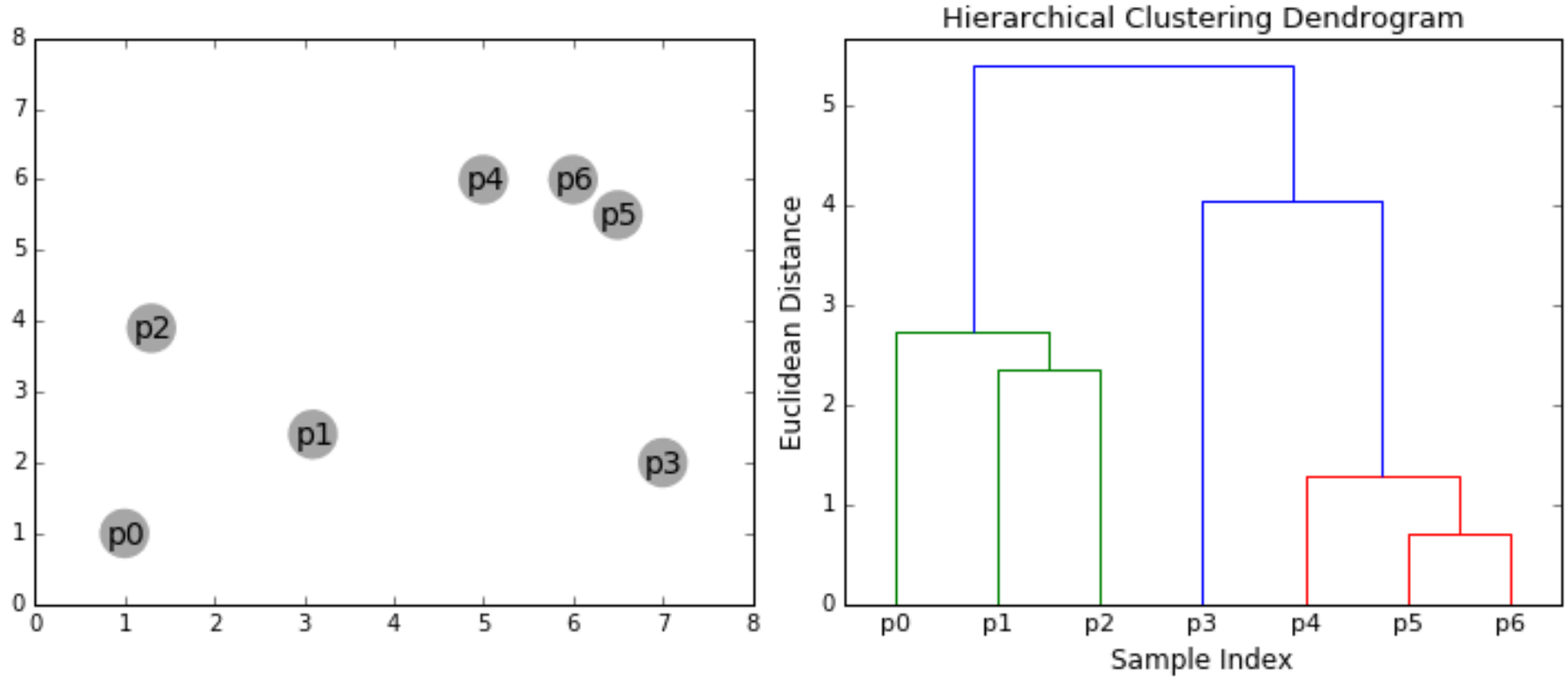
Dendrogram is your friend



Source: Practical Guide to Cluster Analysis in R



Source: George Seif's blogpost on Towards Data Science



Source: George Seif's blogpost on Towards Data Science

Agglomerative

Each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.

Divisive

All observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

Aglomerative Clustering Algorithm

1. Create separate cluster for each example in the dataset,
2. Find two closests clusters.
3. Merge them into one cluster.
4. Move to Step 2 if you have more than one cluster.

Why/Why not to use Hierarchical Clustering?

Advantages:

- No need to specify number of clusters a'priori,
- Works well with different metrics.

Disdvantages:

- Very low efficiency…

Questions? ☺️

http://bit.ly/2VxeyZw

Let’s use it in practice! 🚀

Resources

- Machine Learning Course by Andrew Ng - Coursera,
- “Pattern Recognition and Machine Learning”, Christopher M. Bishop,
- Machine Learning for Everyone,
- Andrew Ng about Anomaly Detection Problem,
- Fong Chun Chan’s blog about Mixture Models,
- Yu’s Machine Learning Garden about Gaussian Mixture Models,
- Angus Turner’s blogpost about Gaussian Mixture Models in PyTorch.

Resources

- Iris Dataset - Kaggle,
- Credit Card Fraud Detection Dataset - Kaggle,
- Clustering - Scikit-Learn User Guide,
- K-Means - Scikit-Learn Documentation,
- Agglomerative Clustering - Scikit-Learn Documentation,
- Gaussian Mixture Models - Scikit-Learn User Guide,
- Gaussian Mixture Models - Scikit-Learn Documentation.

Thanks

Jakub Powierza

jakub.powierza@icloud.com
https://linkedin.com/in/jakub-powierza/

https://github.com/jpowie01/
UnsupervisedLearningWorkshop_BeIT