Unsupervised learning: K-means and Gaussian Mixture Models

Machine Learning Summer Course 2020

Krishnakant Saboo

18th July 2020

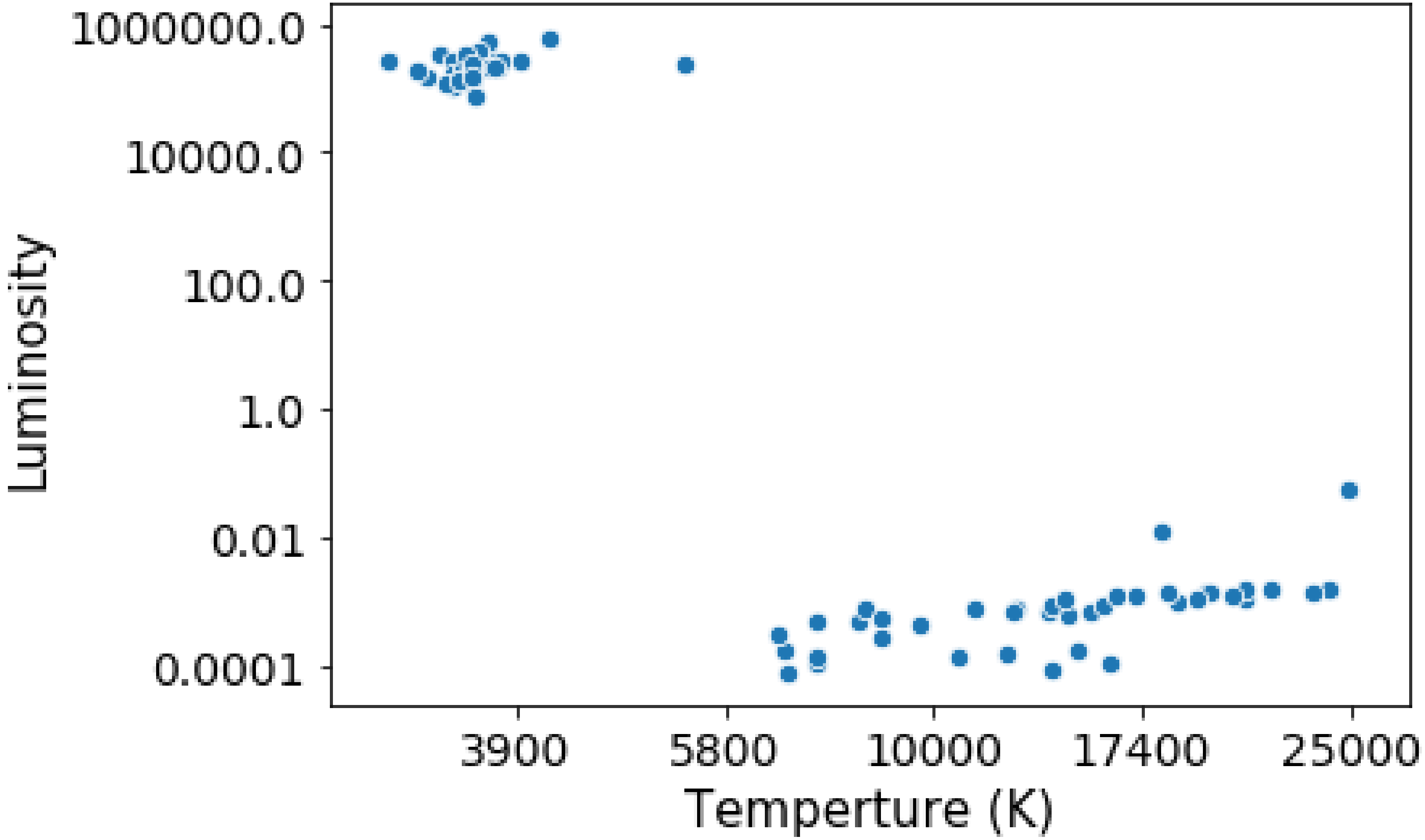
Sky full of stars

There are so many stars in the sky.

- Are all of them of the same type or are there different categories?
- How to find those categories?
- What are the properties of those categories?

Data from 74 stars.

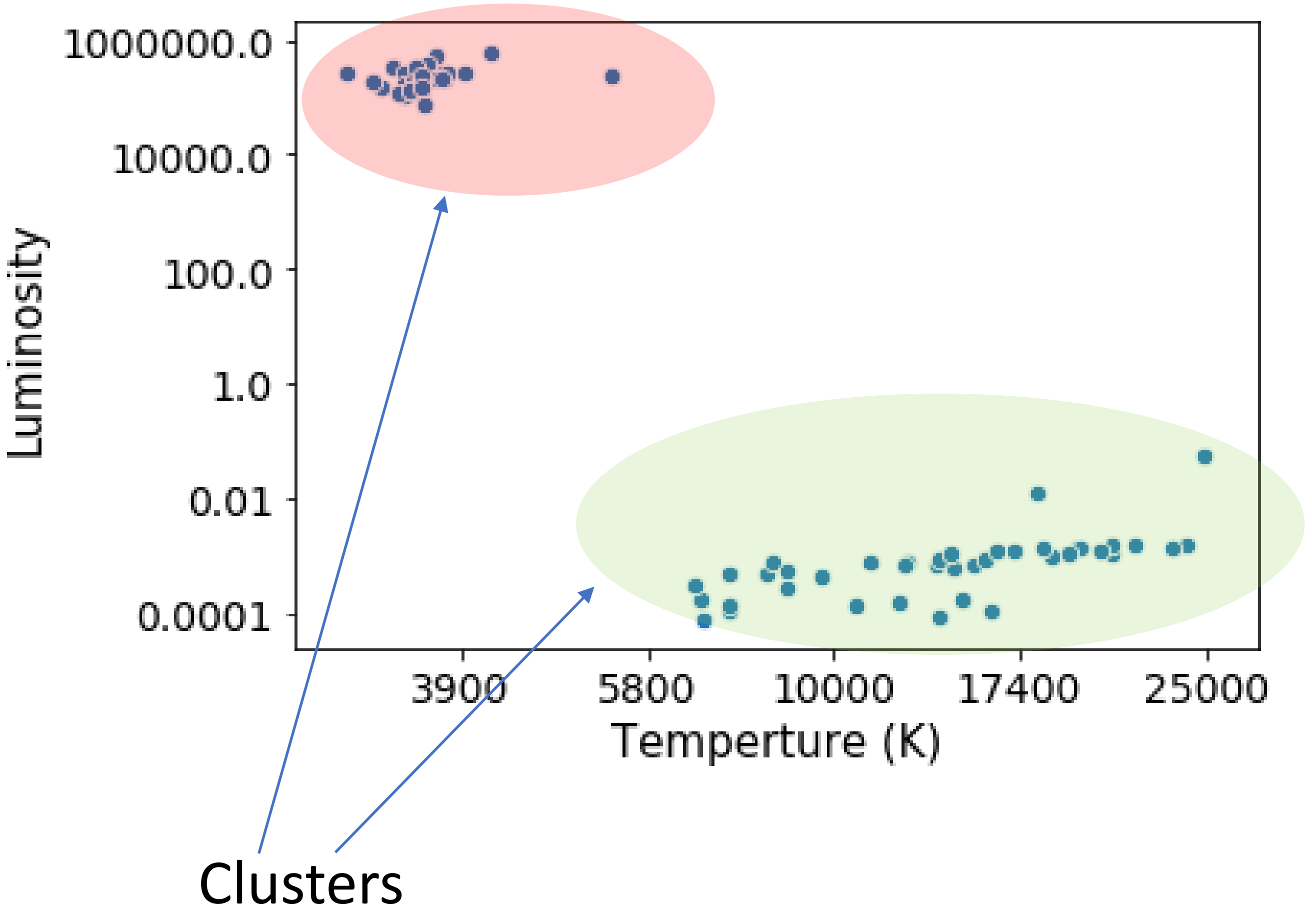
- Temperature at the surface of the star
- Luminosity: Brightness of the star relative to the sun



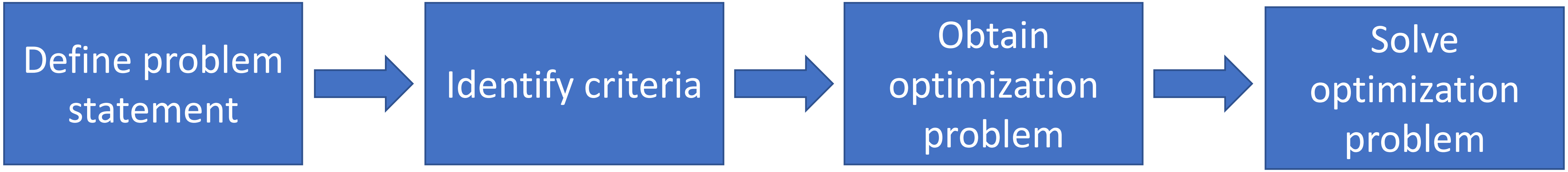
Data source: https://www.kaggle.com/deepu1109/star-dataset

Unsupervised learning

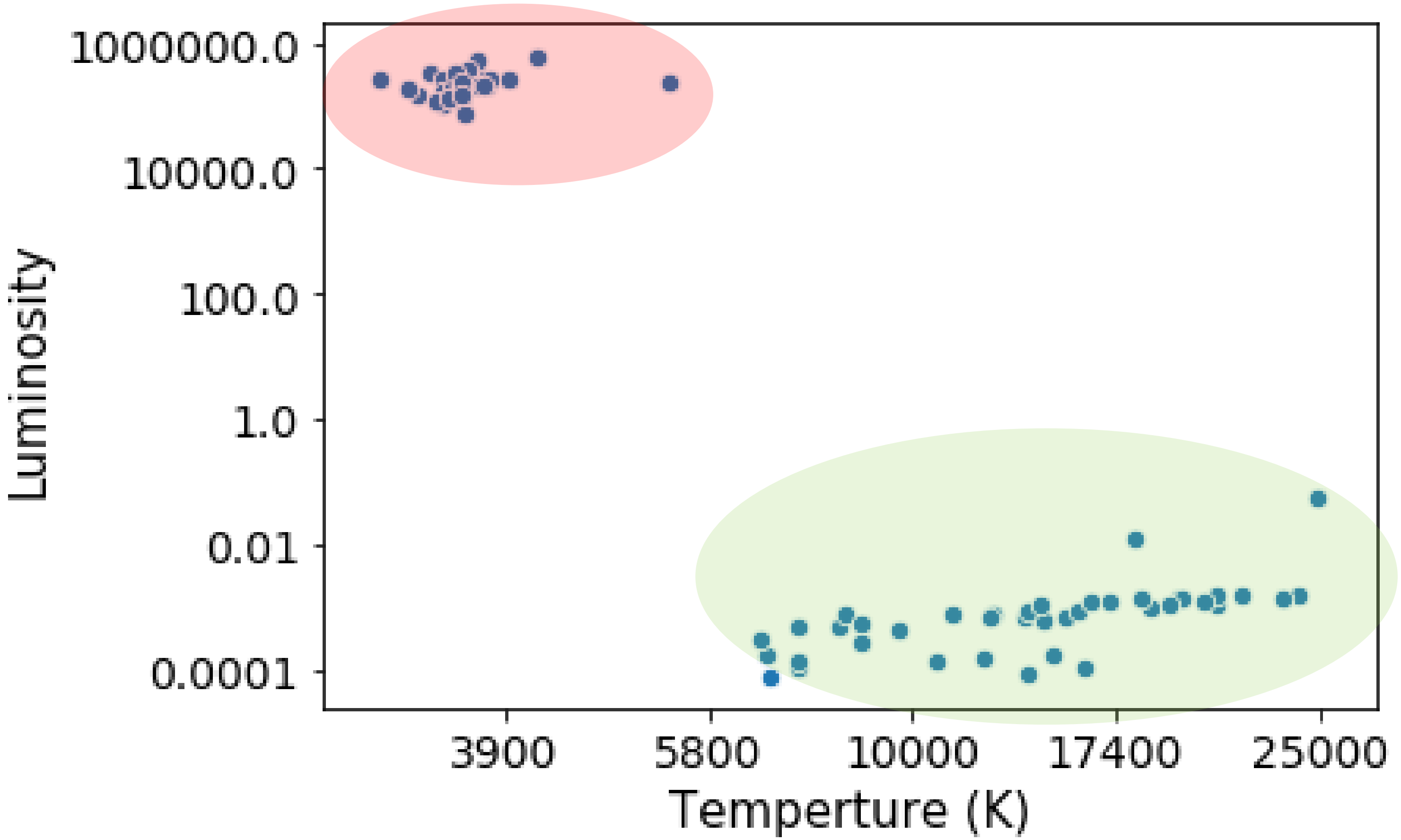
- Label (Y) is unavailable in training data
  - Can happen due to several practical reasons
- Unsupervised learning looks for previously undetected patterns in the data with no pre-existing labels and with minimum human supervision [Wikipedia]
- Goal of unsupervised learning may be to discover groups of similar examples within the data [Bishop 2006]
- Want to find ***clusters*** in the data



Common framework so far…



Criteria for clustering



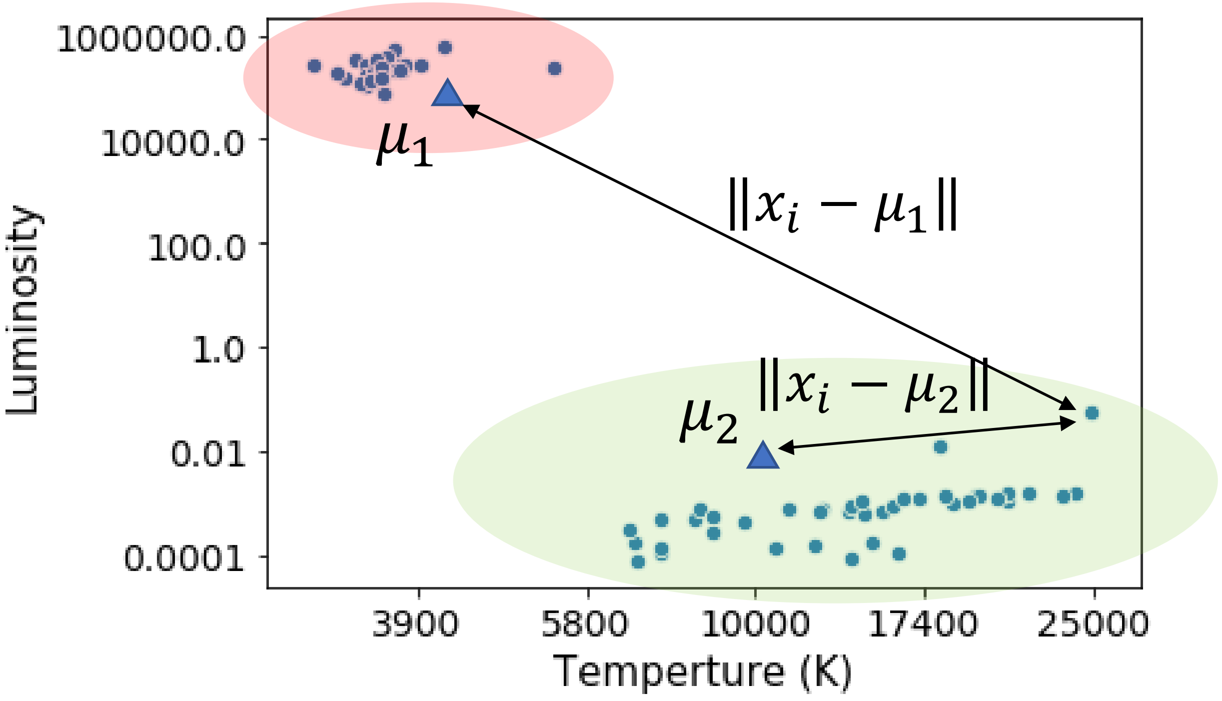
- Distance between points is one characteristic we can use
- Criterion: Samples within the same cluster are closer to each other compared to samples outside the cluster

Formulating the optimization problem

- Samples $x_1, x_2, \ldots, x_N$
- Two clusters (assumption)
- $r_i = [r_{i1}, r_{i2}]$ where

$$r_{ik} = \begin{cases} 1, & x_i \text{ in cluster } k \\ 0, & otherwise \end{cases}$$

- μ_k represents a typical point in cluster k
- Distance of x_i from μ_k: $\|x_i - \mu_k\|$



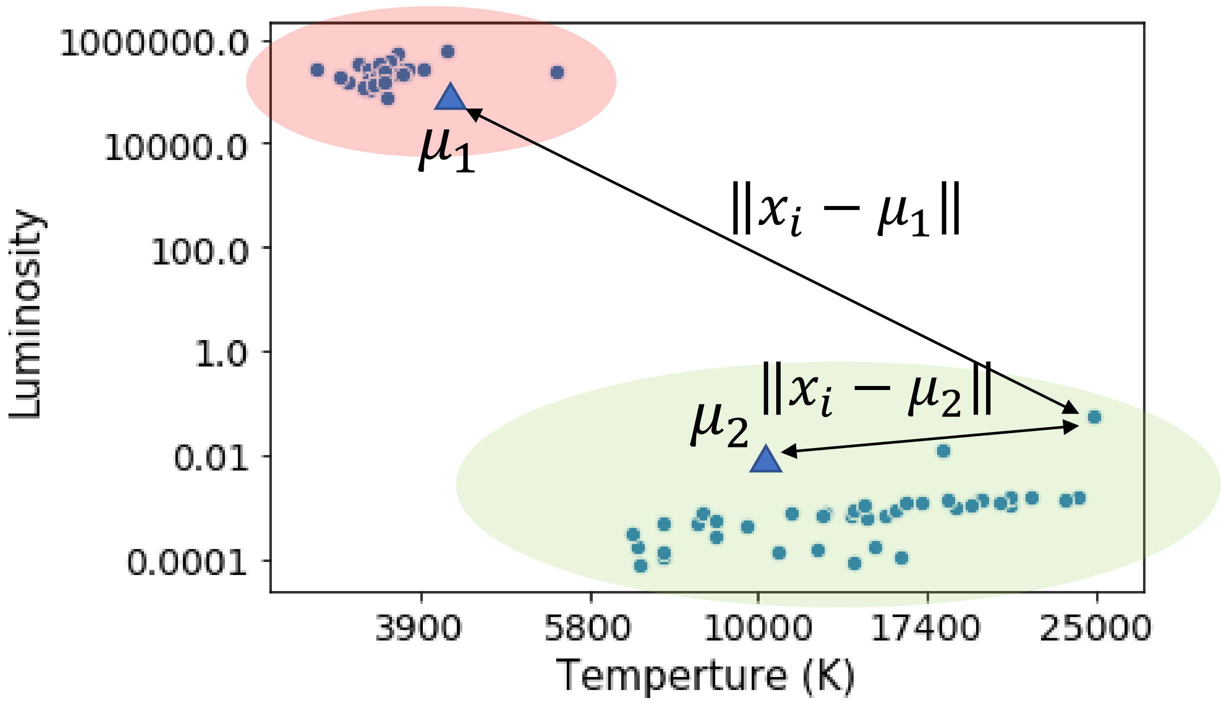
Example typical points for clusters

Formulating the optimization problem

- If μ_k are known, then which cluster should point x_i belong to i.e., what should be r_i?
- Solution: $\|x_i - \mu_2\| < \|x_i - \mu_1\|$
  - So $r_{i2} = 1, r_{i1} = 0$
- Consider the optimization for x_i

$$\min_{r_i} \sum_{k=1}^{2} r_{ik} \|x_i - \mu_k\|^2$$

- Claim: Solving the above optimization will give the cluster for x_i



Example typical points for clusters

Formulating the optimization problem

- We want to solve it together for all points $x_1, \dots, x_N$

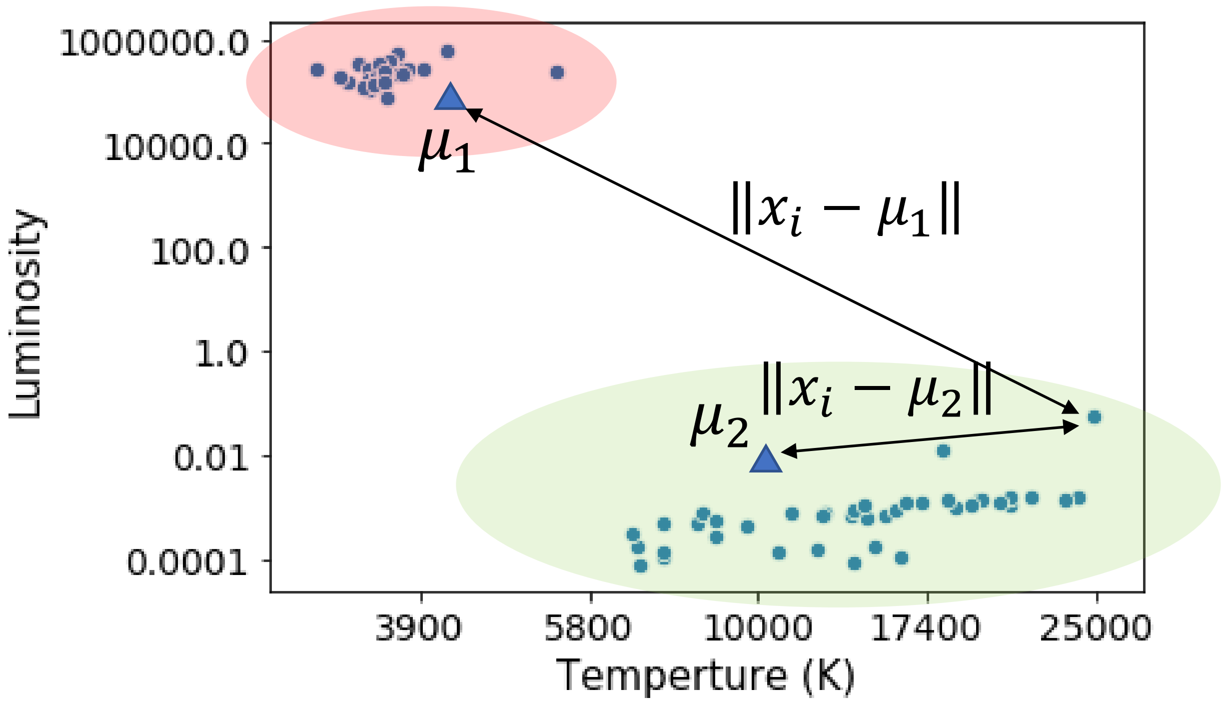
$$J = \sum_{i=1}^{N} \sum_{k=1}^{2} r_{ik} \|x_i - \mu_k\|^2$$

$$\min_{r_1, \dots, r_N} J$$

- But μ_k's are unknown, so we also want to find them

$$\min_{r_1, \dots, r_N, \mu_1, \mu_2} J$$

Distortion measure



Example typical points for clusters

Solving the optimization problem

Solve for r_i's and μ_k's that jointly satisfy

$$\min_{r_1,\ldots,r_N,\,\mu_1,\mu_2} \sum_{i=1}^{N} \sum_{k=1}^{2} r_{ik} \|x_i - \mu_k\|^2$$

No easy way to solve this directly! However, we can break the problem up into smaller problems and tackle them

If we knew μ_k's ....

Then r_i's can be easily found

μ_k is gone from the arguments

$$\min_{r_{1,\ldots,}\, r_N} \sum_{i=1}^{N} \sum_{k=1}^{2} r_{ik} \|x_i - \mu_k\|^2$$

- Observation 1: Cluster for sample x_i is not affected by cluster of sample x_j
  - So overall minimum is the same as minimizing for each x_i separately
- Observation 2: For point x_i, minimum is achieved when $r_{ik} = 1$ for k such that $\|x_i - \mu_k\|$ is the smallest

If we knew r_i's ....

Then μ_k's can be easily found

r_i is gone from the arguments

$$\min_{\mu_1,\mu_2} \sum_{i=1}^{N} \sum_{k=1}^{2} r_{ik} \|x_i - \mu_k\|^2$$

Standard calculus gives

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

$\boxed{\mu_k \text{ is the average of all the points that belong to cluster } k}$

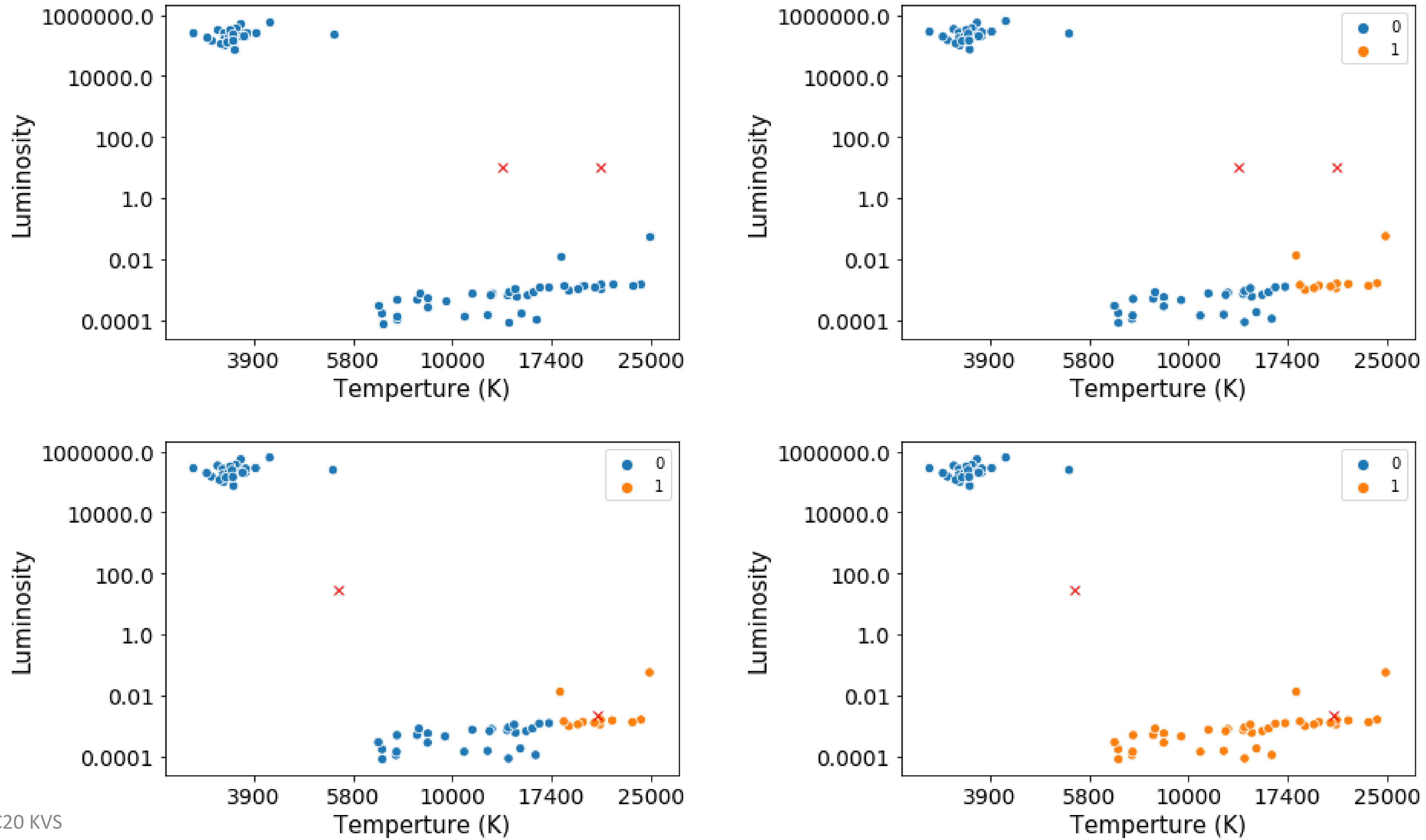
We are not done yet…

- If μ_k are known, then r_i can be found (re-assigning data)
- If r_i are known, then μ_k can be found (re-computing cluster means)
- But we don't know either to begin with…

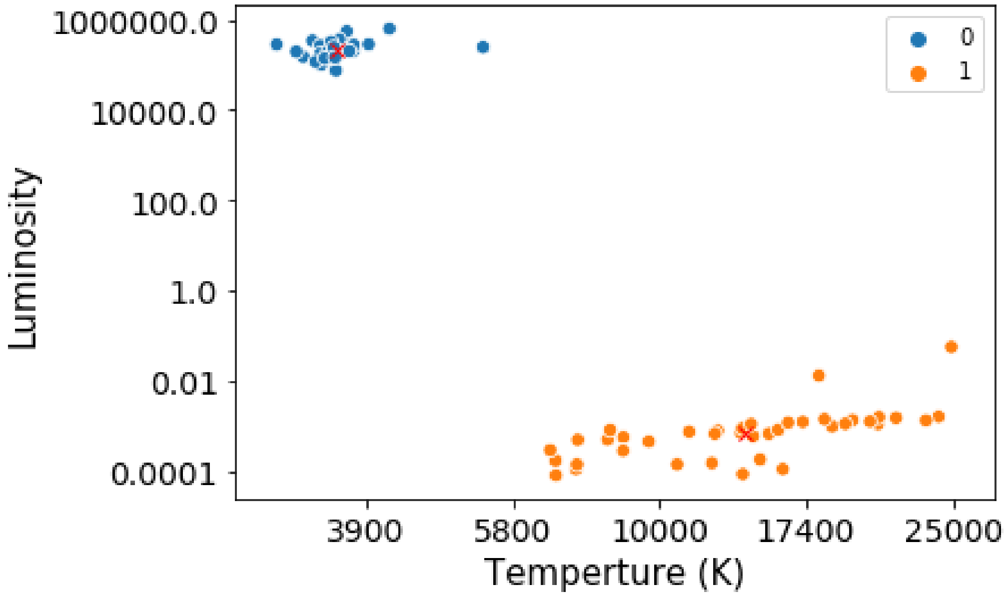
- Solution: Perform them alternatively till convergence

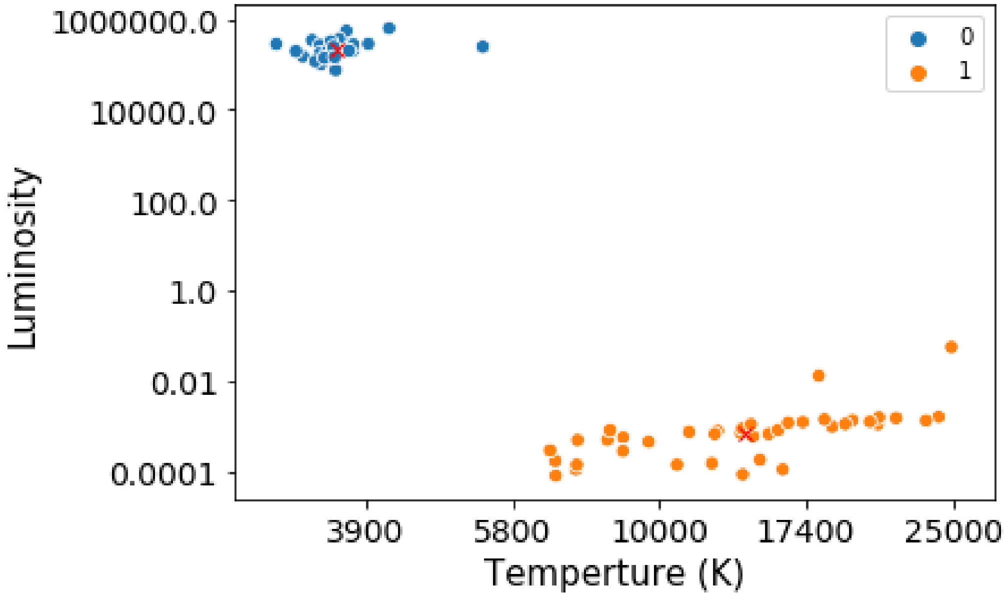
Image source: https://www.applicoinc.com/blog/7-strategies-solving-chicken-egg-problem-startup/

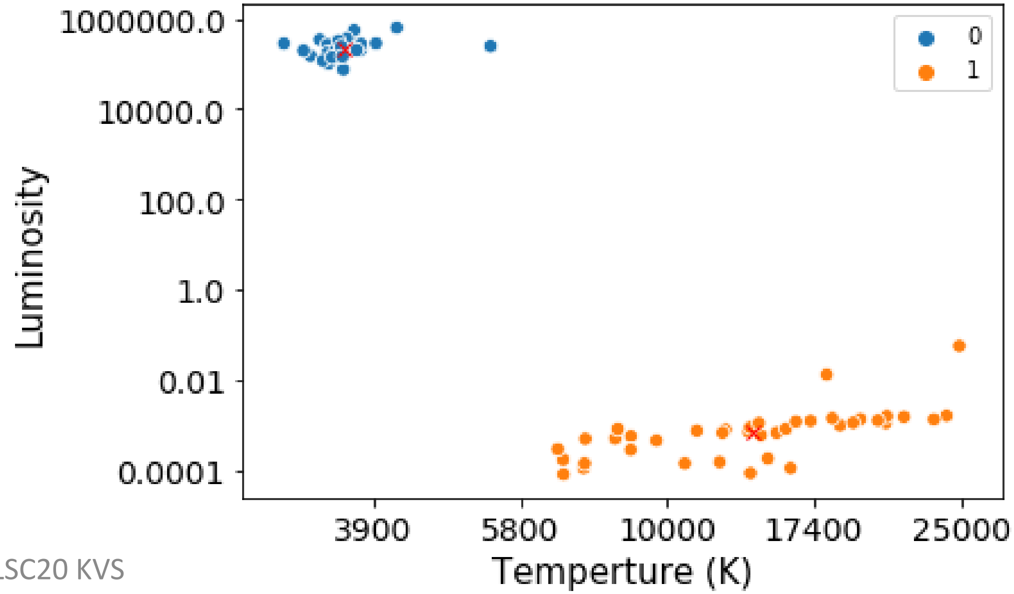
K-means in action



K-means in action







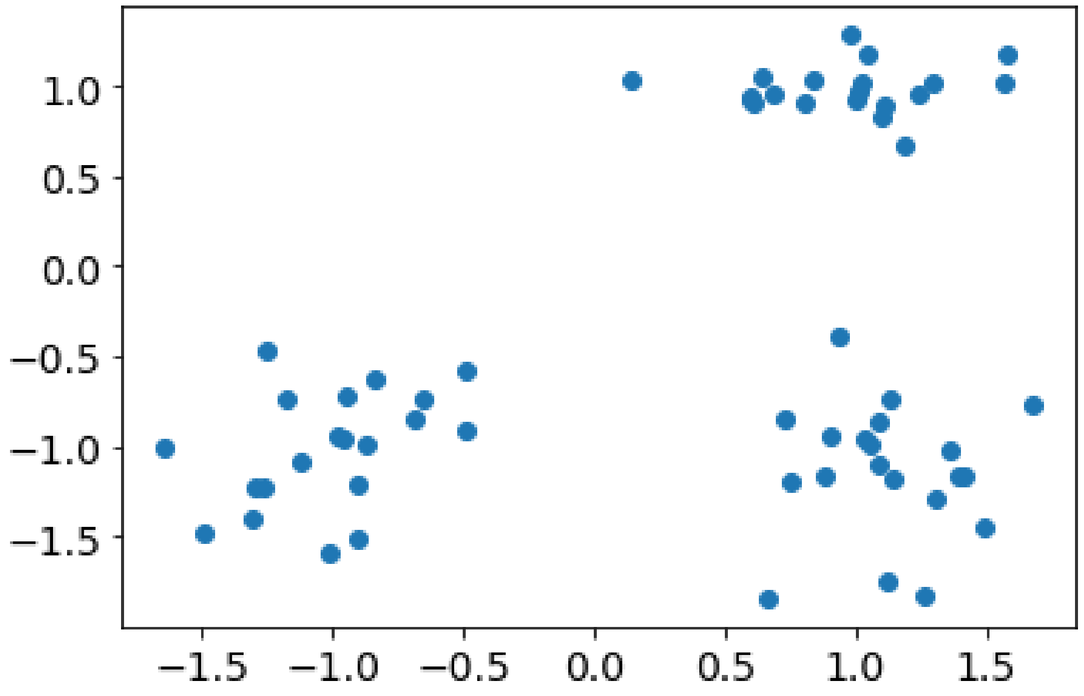
K-means algorithm

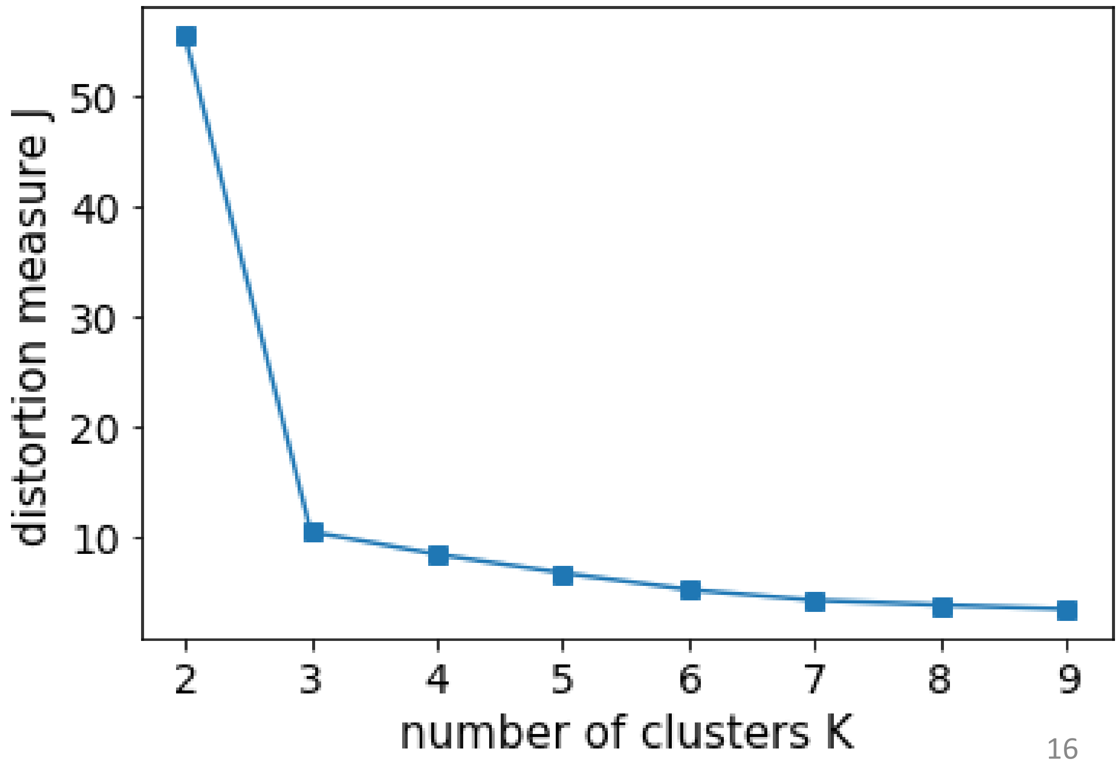
- Data: $x_1, \dots, x_N$ (no labels required)
- Choose number of clusters K
- Randomly select K data points as initial cluster centers (seeds)
- Step 1: Re-assign data to clusters based on new centers
- Step 2: Re-compute cluster means based on data assignment
- Repeat Step 1 and Step 2 alternatively until convergence

The above algorithm works for any number of clusters K and for multidimensional features

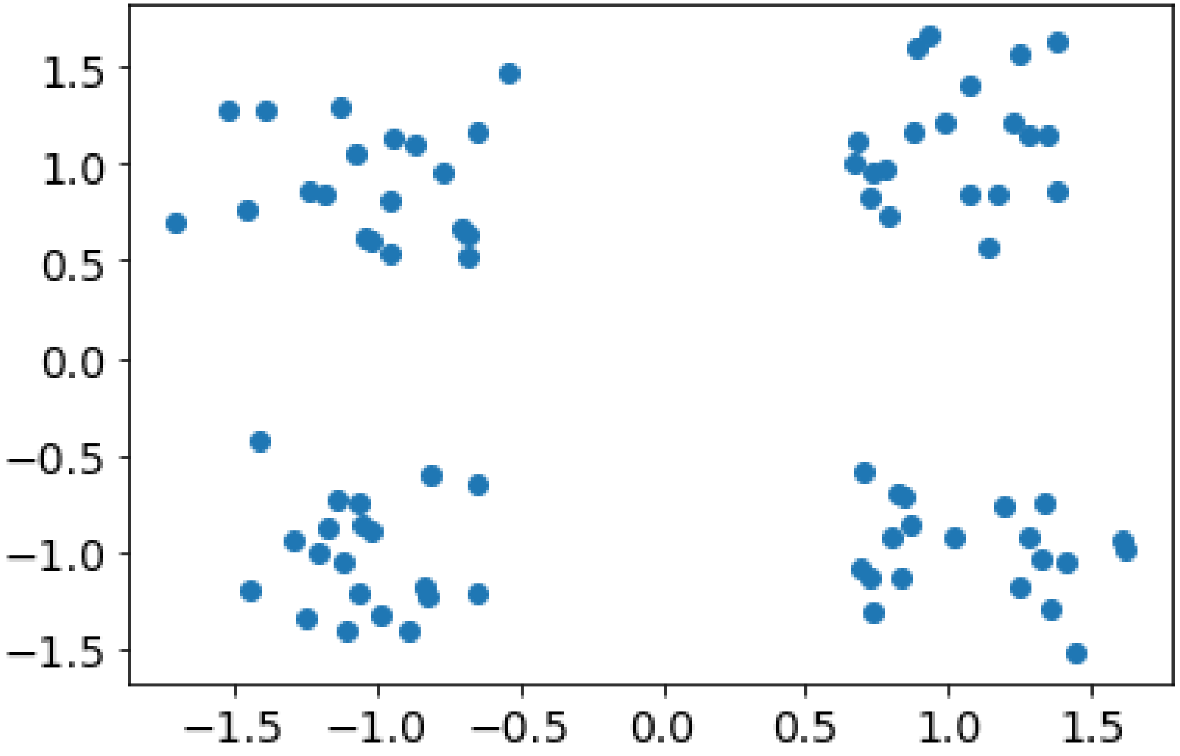
How to choose K?

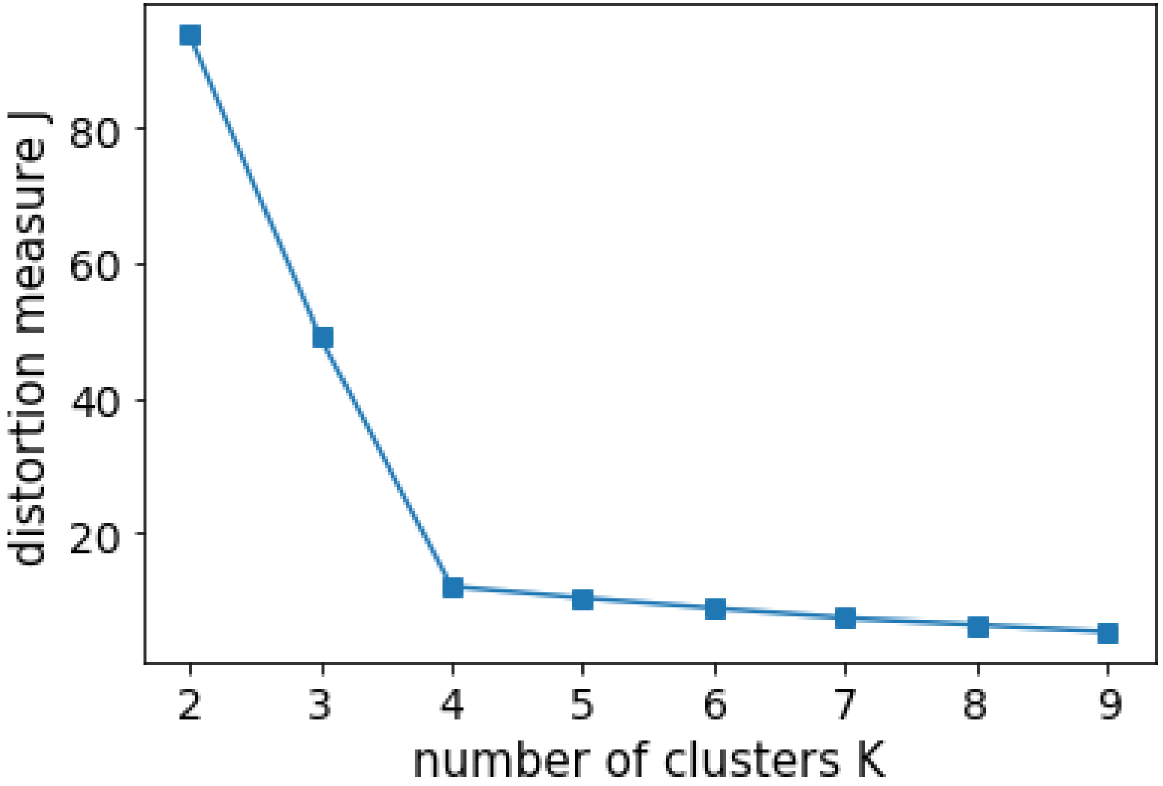
- Prior knowledge/domain knowledge
- Elbow method
  - Intuition: If K is the number of natural clusters, adding more clusters won't reduce J much





Another example for choosing K





What happens when K is not correct?

- Can get non-sensical clusters if K is not chosen appropriately

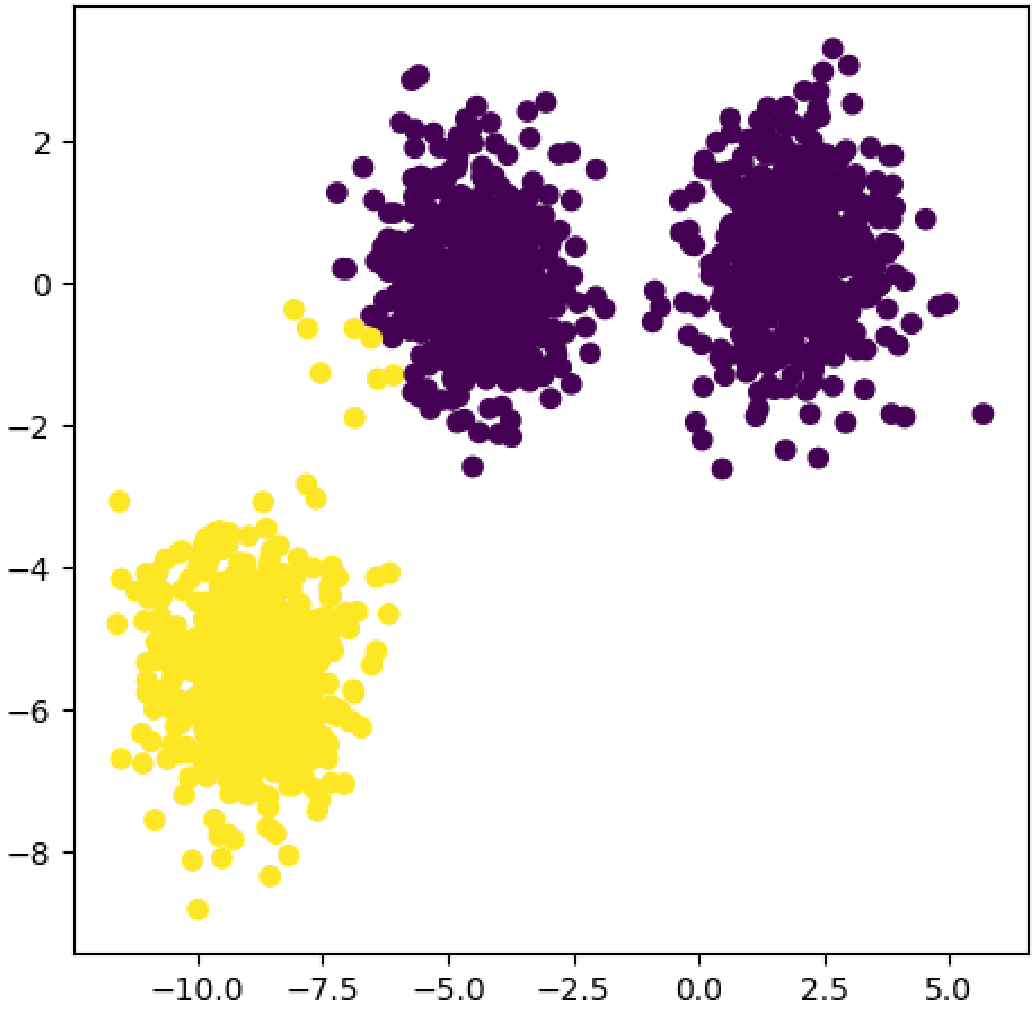
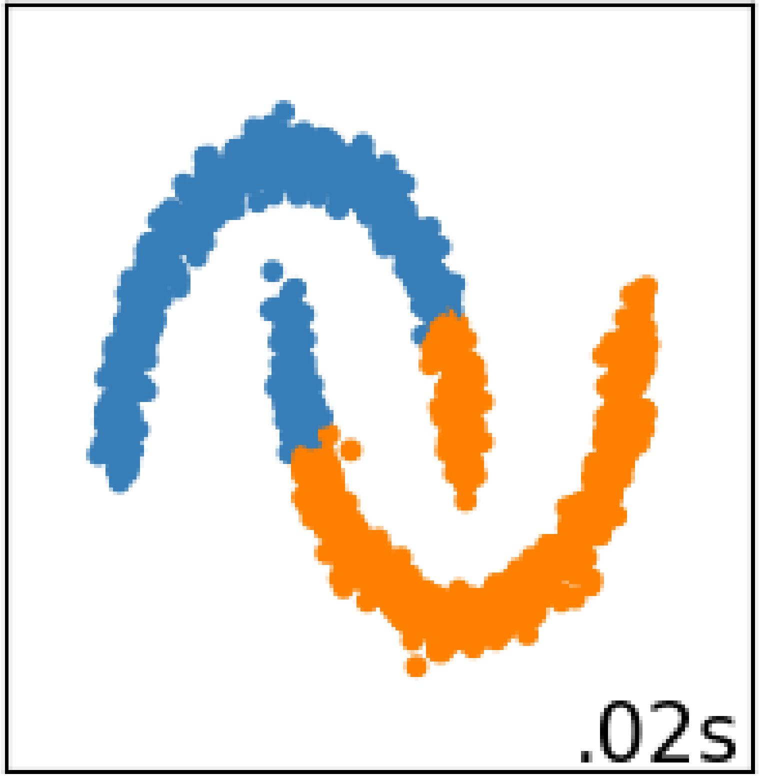
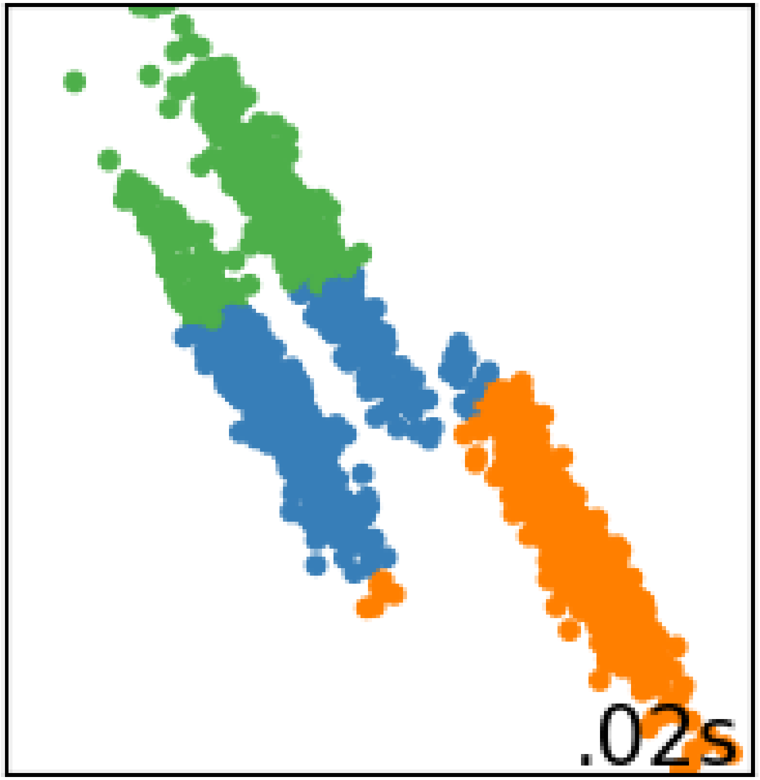


Image source: https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_assumptions.html

Data should be (roughly) spherical





- Data is expected to be roughly spherical or ellipsoid in shape

Image source: https://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

Variance

- Expected clusters have different variances
- K-means ends up creating clusters with roughly the same variance

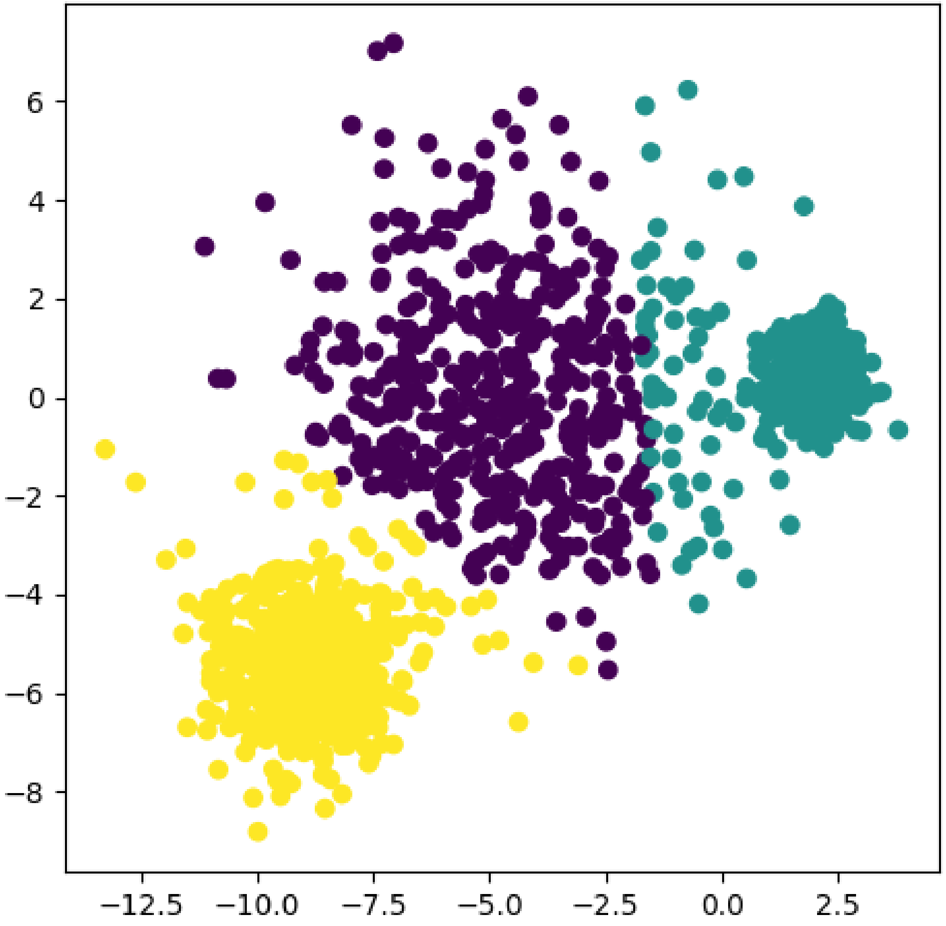


Image source: https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_assumptions.html

Gaussian Mixture Model

Gaussian distribution

$$f(X;\mu,\Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(X-\mu)^T\Sigma^{-1}(X-\mu)\right)$$

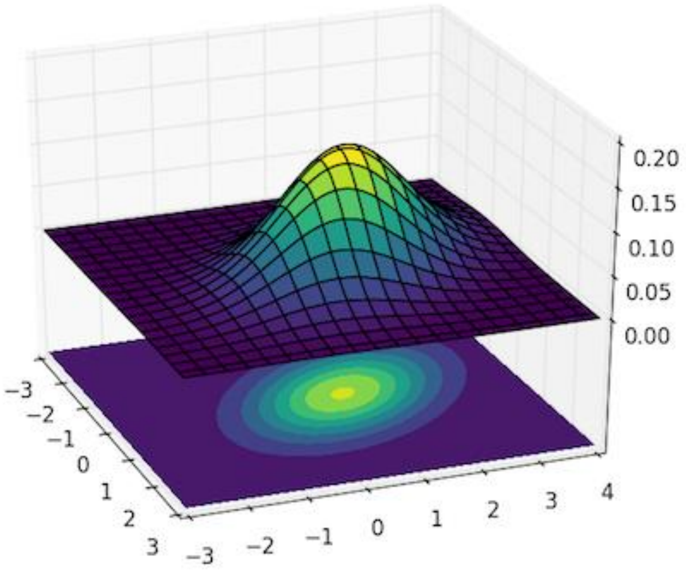
Multivariate Gaussian distribution has two parameters

Mean: $\mu \in R^d$

Covariance matrix: $\Sigma \in R^{d\times d}$

Example with d=2

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$



$\rho = 0 \quad \sigma_1 = 1 \quad \sigma_2 = 1$

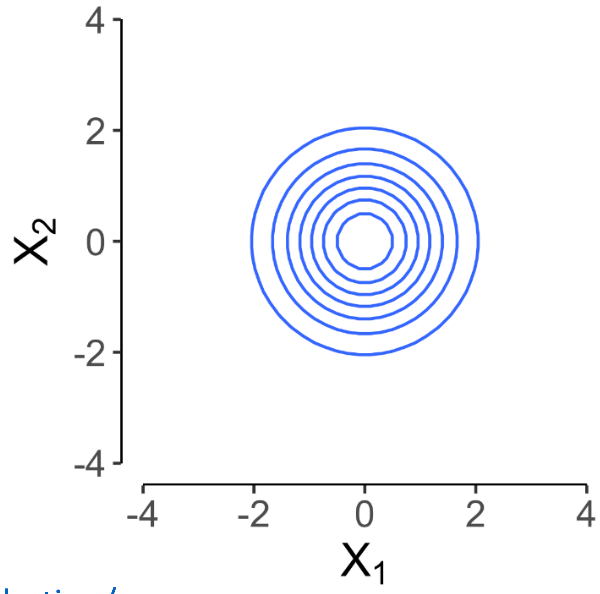


Image source: https://scipython.com/blog/visualizing-the-bivariate-gaussian-distribution/
Image source: https://fabiandablander.com/statistics/Two-Properties.html

Effect of varying the covariance matrix

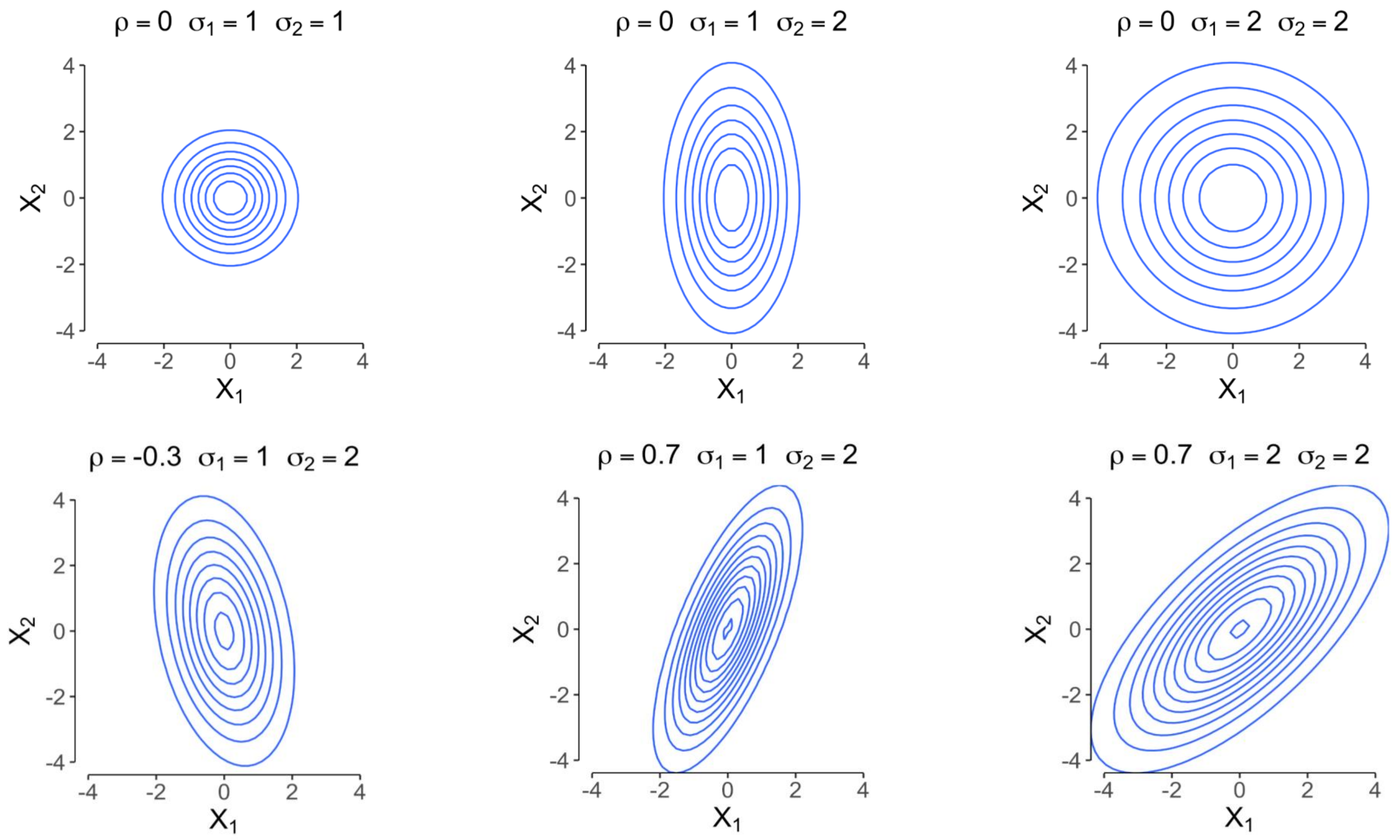


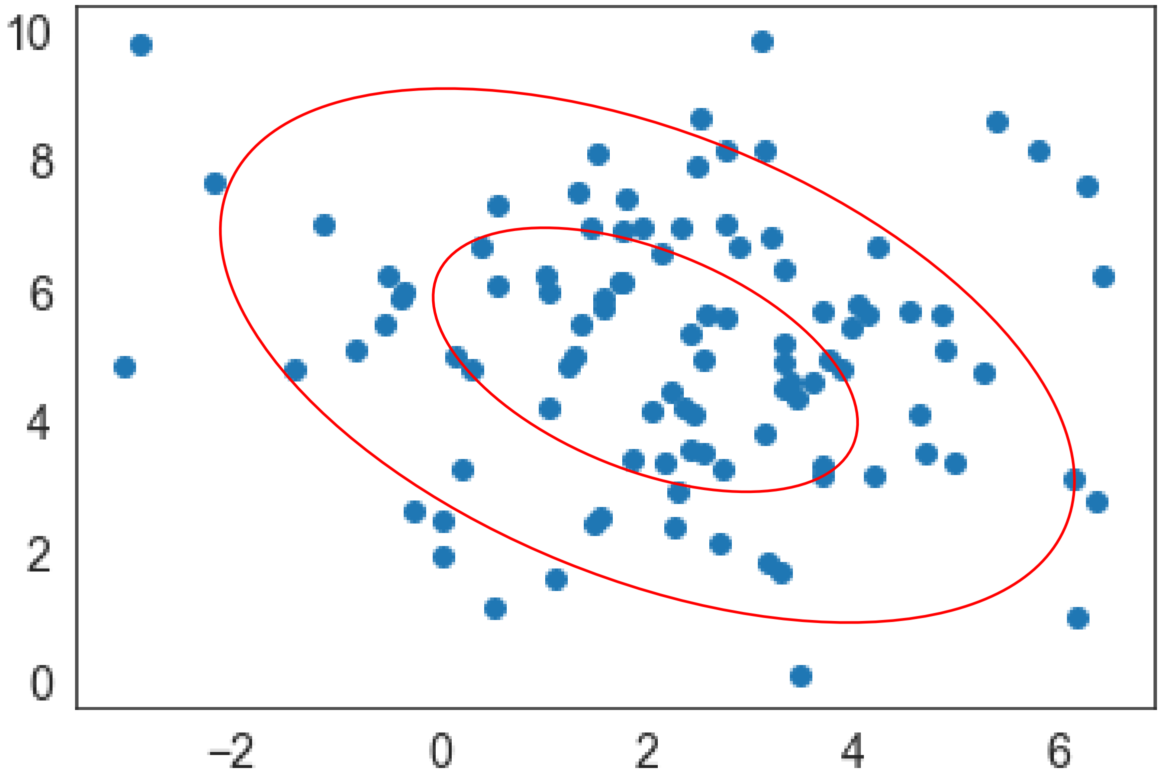
Image source: https://fabiandablander.com/statistics/Two-Properties.html

Estimating mean and covariance matrix of a Gaussian distribution

- If $X_1, X_2, \ldots$ are N samples from a d dimensional Gaussian distribution with mean μ and covariance matrix Σ, then the parameters can be estimated from the data as follows:

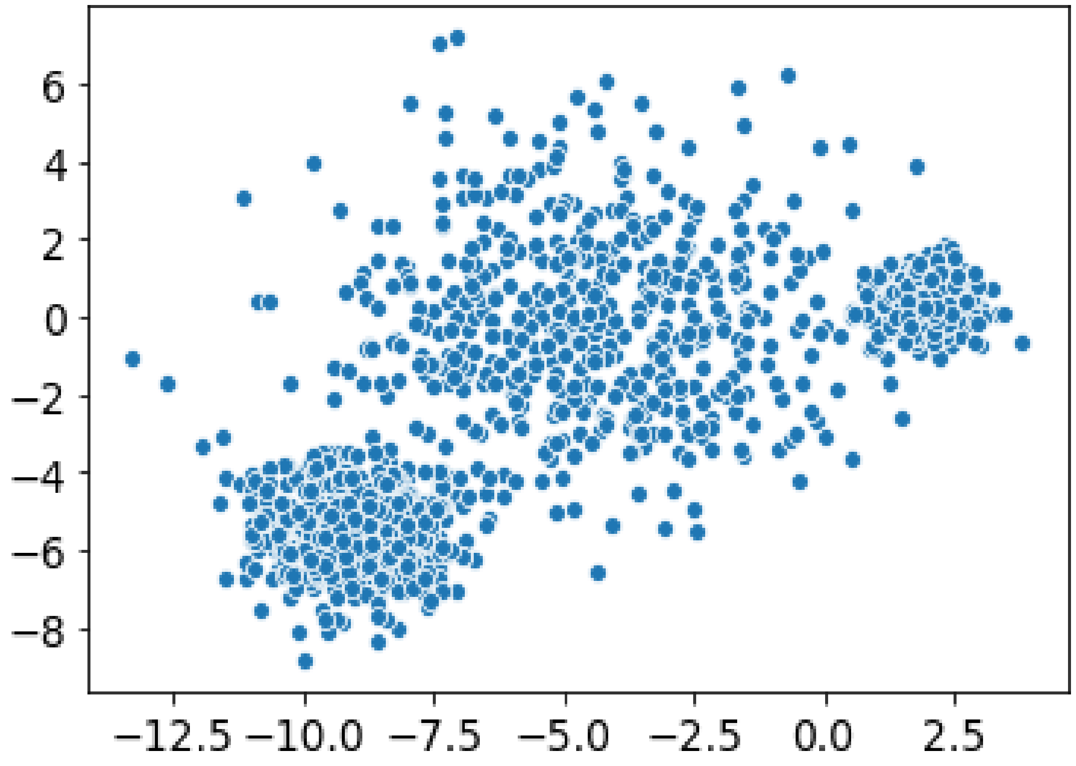
$$\bar{\mu} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

$$\bar{\Sigma} = \frac{1}{N}\sum_{i=1}^{N} (X_i - \bar{\mu})(X_i - \bar{\mu})^T$$



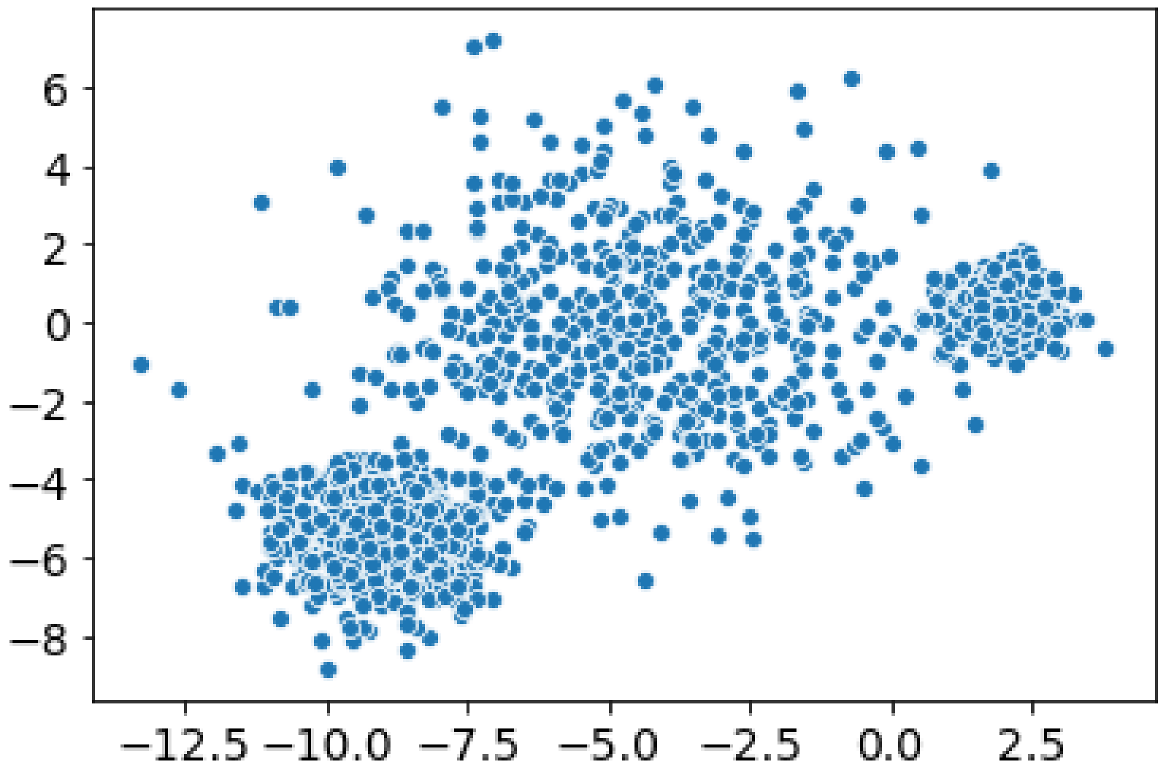
Hard clustering vs soft clustering

- Hard clustering: Sample belongs to only one cluster
  - For example, cluster belonging in K-Means
- Soft clustering: Sample belongs to multiple clusters with varying degree
- Gives a measure of confidence about clustering
- Can achieve soft clustering using concepts from probability
  - For example, if there are 3 clusters, a sample belongs to Cluster 1 wp 0.5, Cluster 2 wp 0.3, and Cluster 3 wp 0.2



Gaussian Mixture model (GMM)

- Goal: (Soft) Cluster the data
- Assumption:
  - Data consists of multiple Gaussian distributions
  - Each sample comes from one Gaussian distribution (unknown to us)
- Want to find the parameters of the Gaussian distributions and probability of choosing a given Gaussian distribution
- Number of Gaussians must be specified (like in K-means)

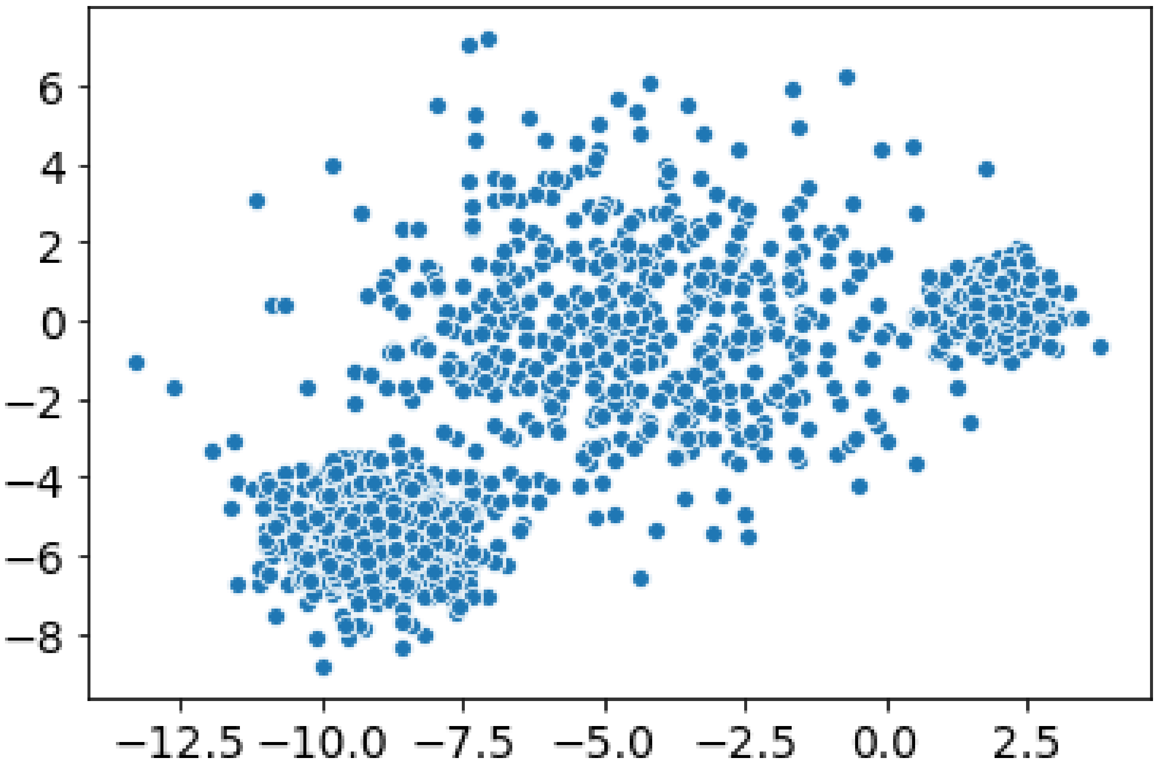


Formulation of optimization problem

- Samples $x_1, x_2, \dots, x_N$
- For each x_i, define z_i which represents the true (unknown) cluster (like r_i in k-means)

$$z_i = [z_{i1}, z_{i2}, z_{i3}]$$

- Parameters of Gaussian distribution: π_k, μ_k, Σ_k for $k = 1, 2, 3$



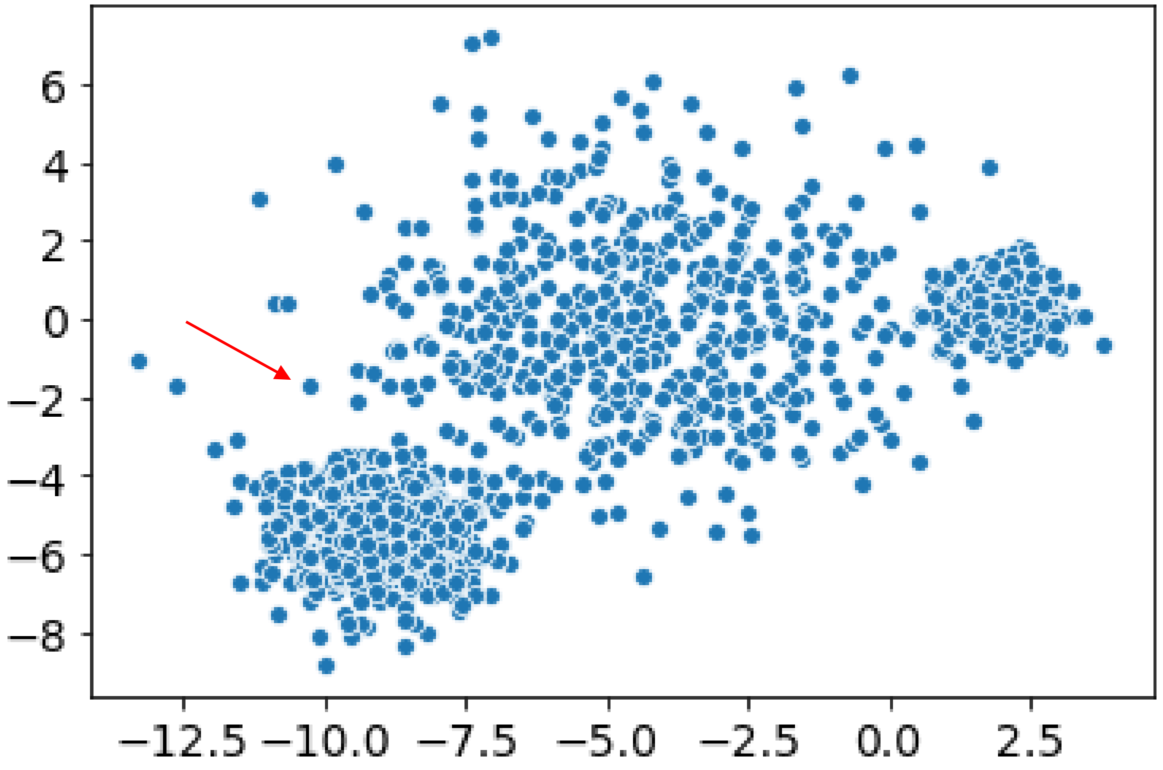
Formulation of optimization problem

- If parameters of Gaussian distribution: π_k, μ_k, Σ_k for $k = 1, 2, 3$ are known

$$P(z_{i1} = 1 | x_i) = ?$$

$$= \frac{p(x_i | z_{i1} = 1) P(z_{i1} = 1)}{p(x_i)}$$

$$= \frac{p(x_i | z_{i1} = 1) \pi_1}{\sum_{k=1}^{3} p(x_i | z_{ik} = 1) \pi_k}$$



$$p(x_i | z_{ik} = 1) = f(x_i; \mu_k, \Sigma_k)$$

For comparison with K-means, can think of $P(z_{i1} = 1 | x_i)$ as the "distance" of x_i from Cluster 1

Optimization problem

- Want to maximize the probability of observing the given data by appropriately choosing z_i's and π_k, μ_k, Σ_k's
- Optimization problem has a similar issue like in K-means
  - All terms cannot be optimized together
- Can we break up the problem into smaller problems in this case too?

If we knew the parameters π_k, μ_k, Σ_k …

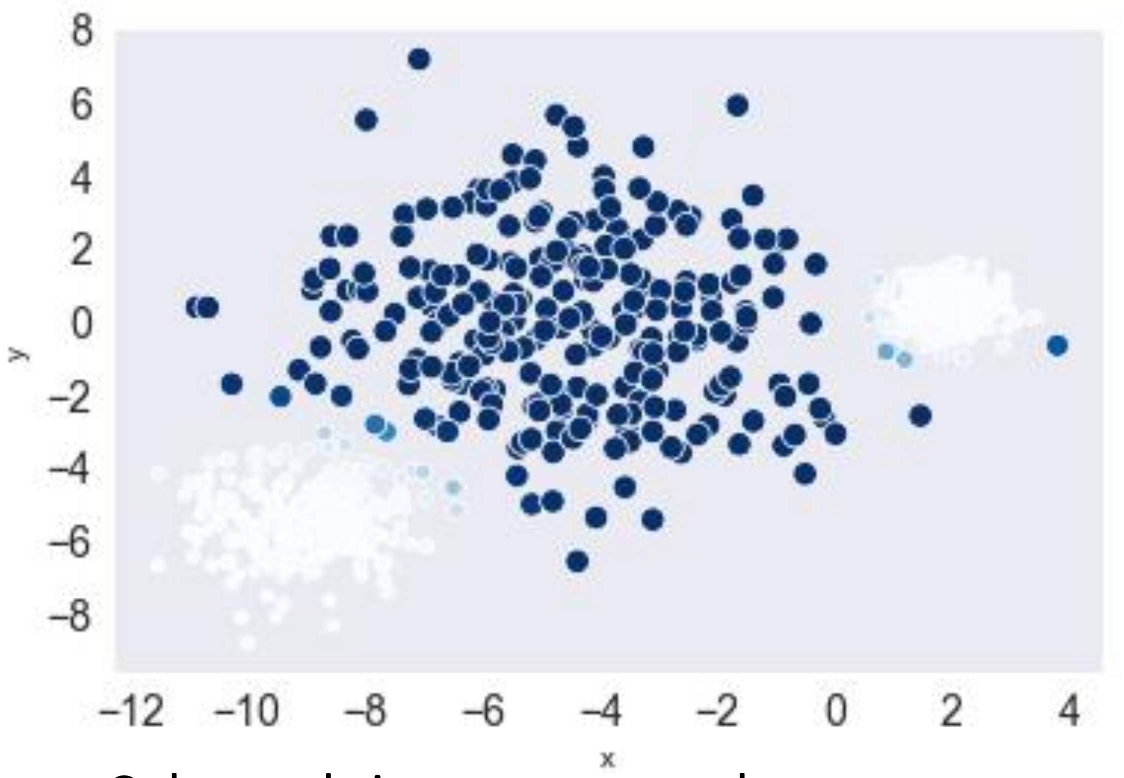
- What is the best choice of $z_1, z_2, \dots, z_N$?
- Observation 1: Samples are independent, so solving maximization for each one separately and combining them gives the correct answer
- Observation 2: For sample x_i, the correct cluster would be the one that has the maximum probability $P(z_{ik} = 1|x_i)$
  - Compute $P(z_{i1} = 1|x_i), P(z_{i2} = 1|x_i), P(z_{i3} = 1|x_i)$
  - Hard assignment: Choose the maximum out of them
  - Soft assignment: These probability values itself are the soft assignment

If we knew the soft assignments ….

- Then can the parameters be computed?

Let us look at Cluster 1 (z_{i1}'s)

- Belongingness of x_i to Cluster 1 is $P(z_{i1} = 1|x_i)$
  - Example, $P(z_{i1} = 1|x_1) = 0.70, P(z_{i1} = 1|x_2) = 0.18$
  - Which on the above should contribute more to the parameters of Cluster 1?
- Intuition: Sample will contribute to parameter based on their belongingness



Color and size represents the belongingness (larger and darker is higher)

$$\mu_1 = \frac{\sum_{i=1}^{N} 1 \times x_i}{\sum_{i=1}^{N} 1} \quad \Rightarrow \quad \mu_1 = \frac{\sum_{i=1}^{N} P(z_{i1} = 1|x_i) x_i}{\sum_{i=1}^{N} P(z_{i1} = 1|x_i)}$$

Like r_{i1} in K-means but allowed to be in [0,1]

If we knew the soft assignments ….

- Intuition: Sample will contribute to parameter based on their belongingness

Let us look the Cluster 1 (z_{i1}’s)

- Belongingness of for x_i to Cluster 1 is $P(z_{i1} = 1|x_i)$

$$\mu_1 = \frac{\sum_{i=1}^{N} P(z_{i1} = 1|x_i)x_i}{\sum_{i=1}^{N} P(z_{i1} = 1|x_i)}$$

$$\Sigma_1 = \frac{\sum_{i=1}^{N} P(z_{i1} = 1|x_i)(x_i - \mu_1)(x_i - \mu_1)^T}{\sum_{i=1}^{N} P(z_{i1} = 1|x_i)}$$

$$\pi_1 = \frac{\sum_{i=1}^{N} P(z_{i1} = 1|x_i)}{N}$$

- Same formulae hold for the parameters of the other clusters also with the probability terms $P(z_{ik} = 1|x_i)$ being used for Cluster k

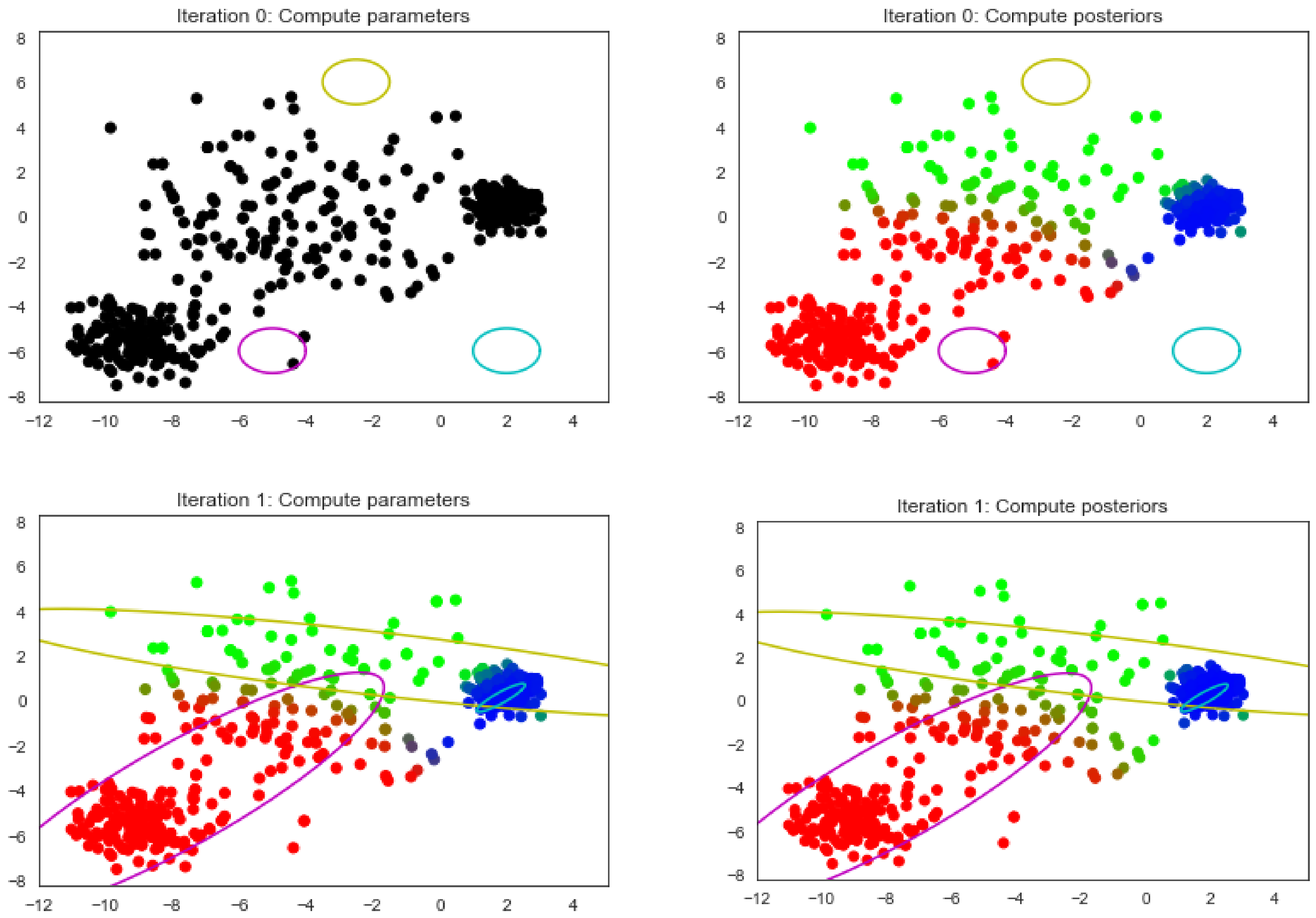
We are not done yet…

- If parameters are known, then probabilities can be found (soft clustering of data)
- If probabilities are known, then parameters can be found (re-computing Gaussian parameters)
- But we don't know either to begin with…

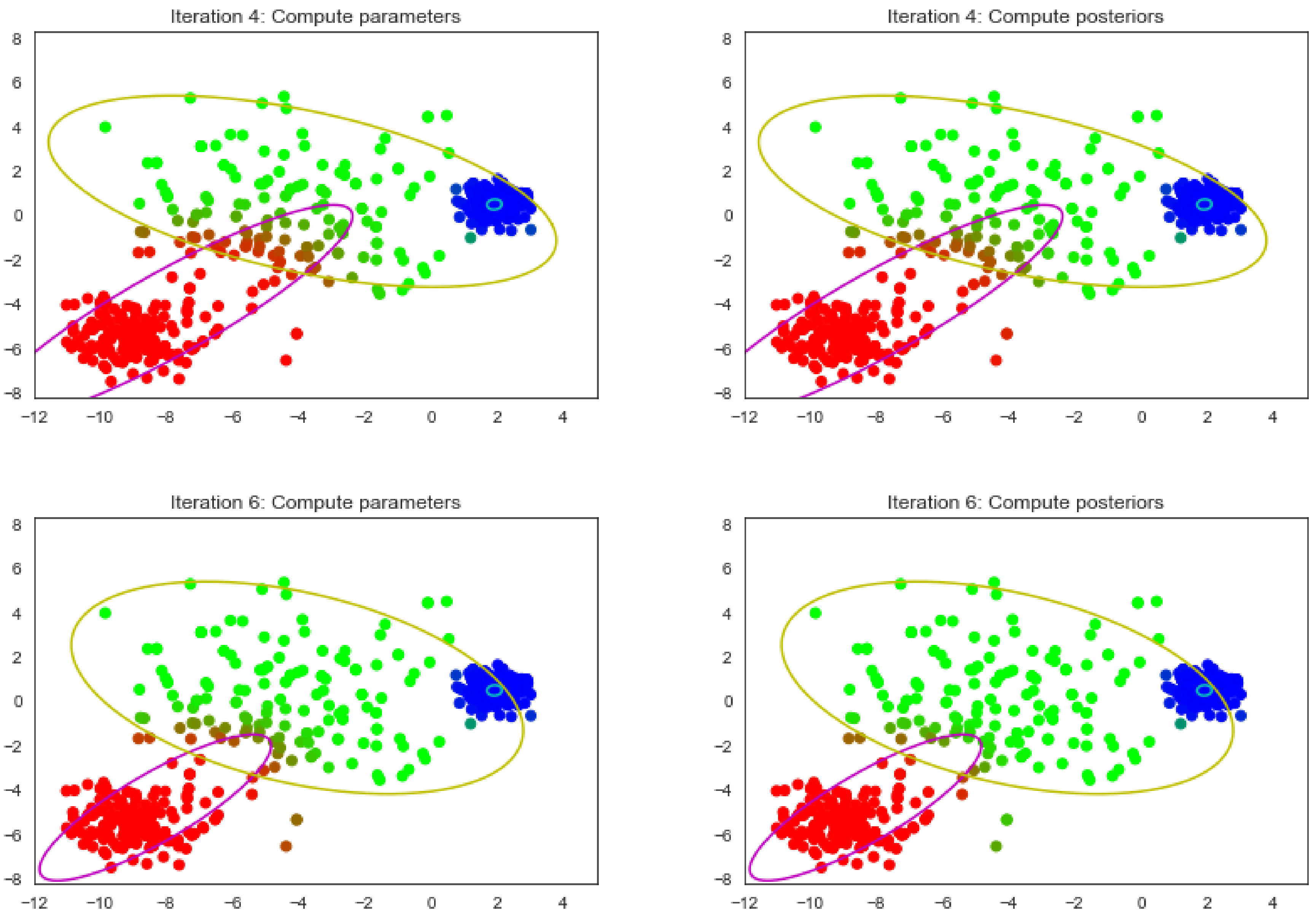
- Solution: Perform them alternatively till convergence

Image source: https://www.applicoinc.com/blog/7-strategies-solving-chicken-egg-problem-startup/

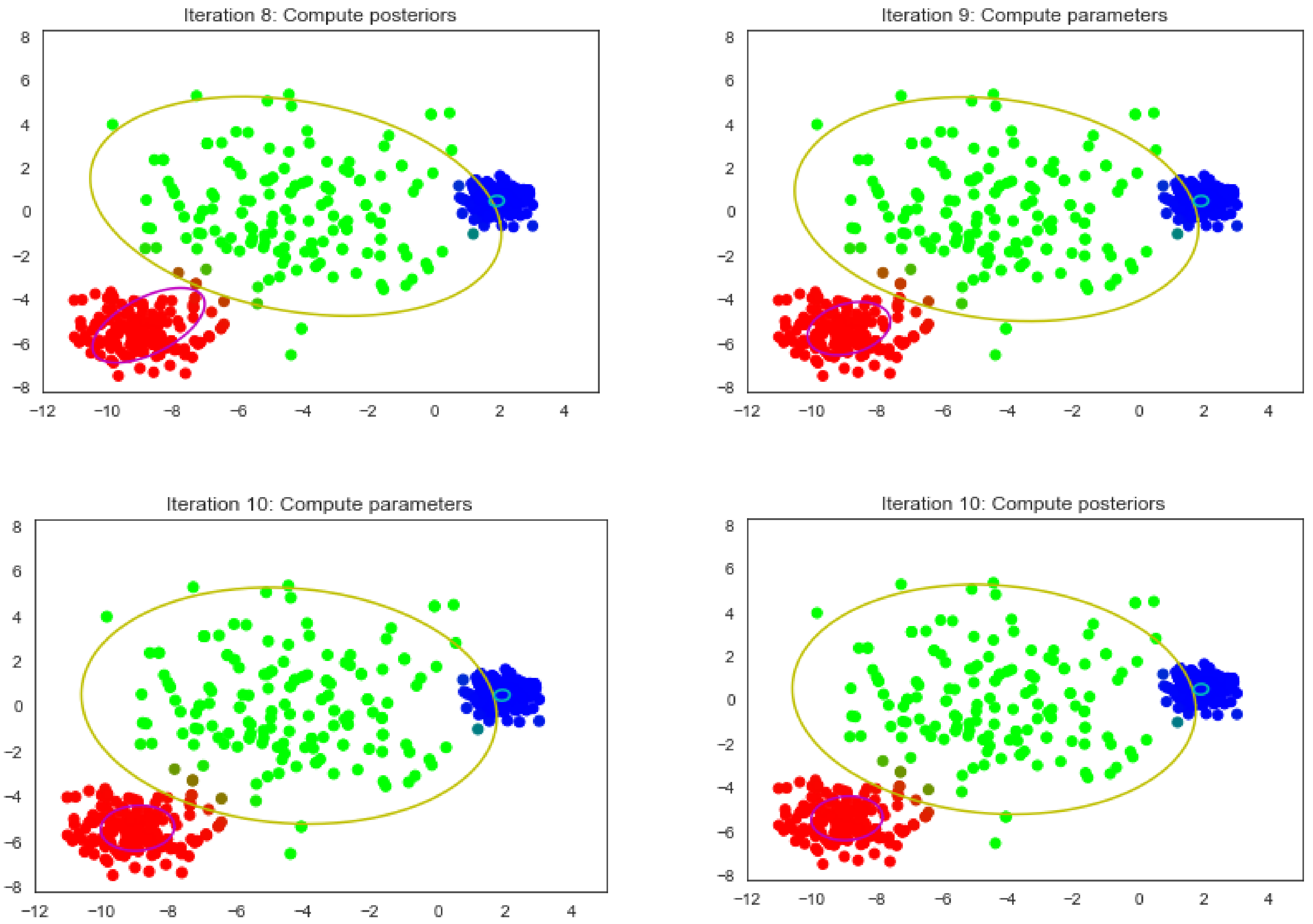
GMM in action



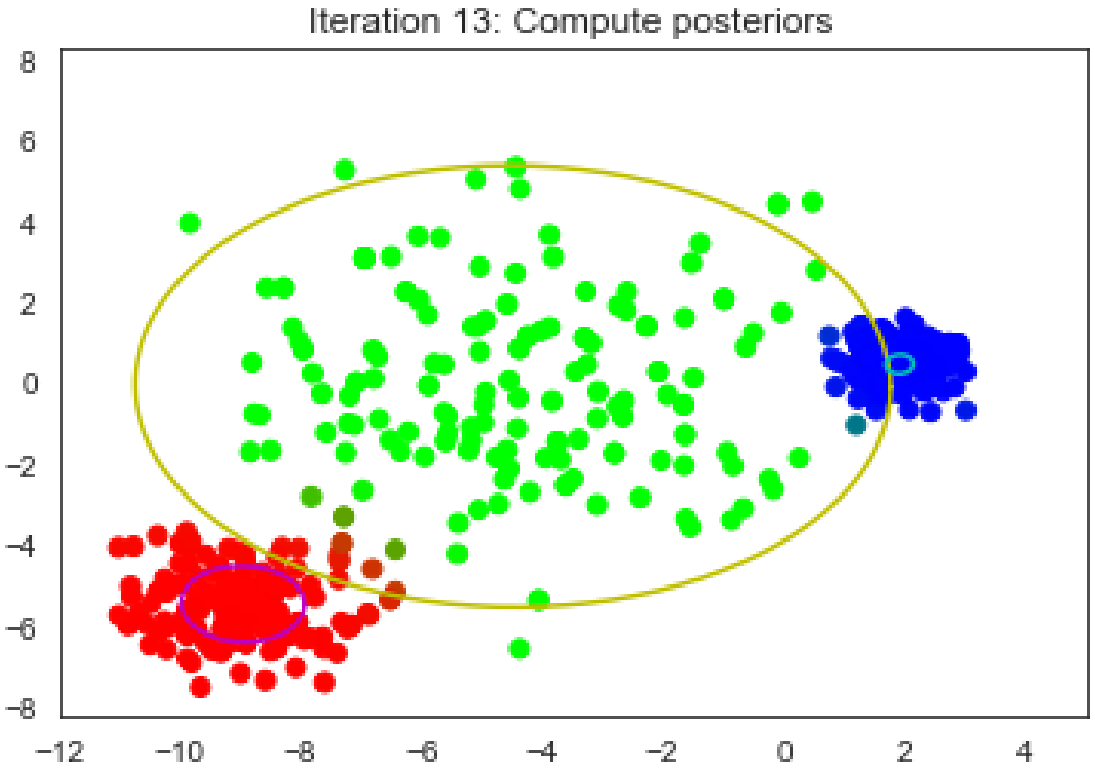
GMM in action

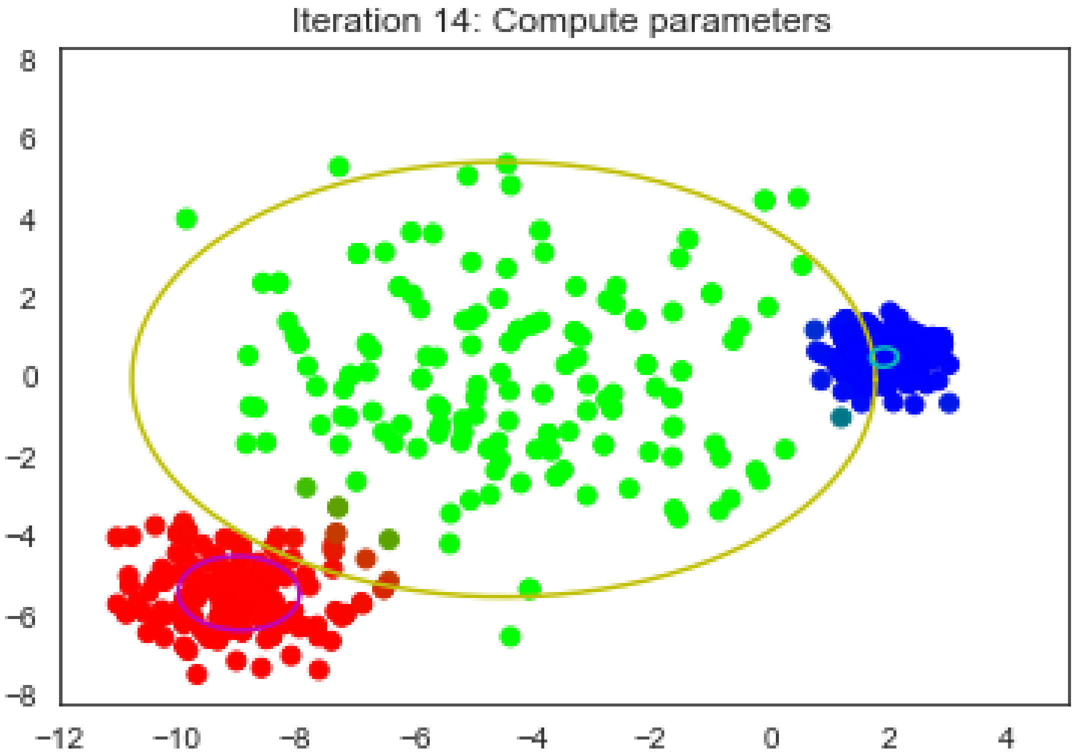


GMM in action



GMM in action





GMM algorithm

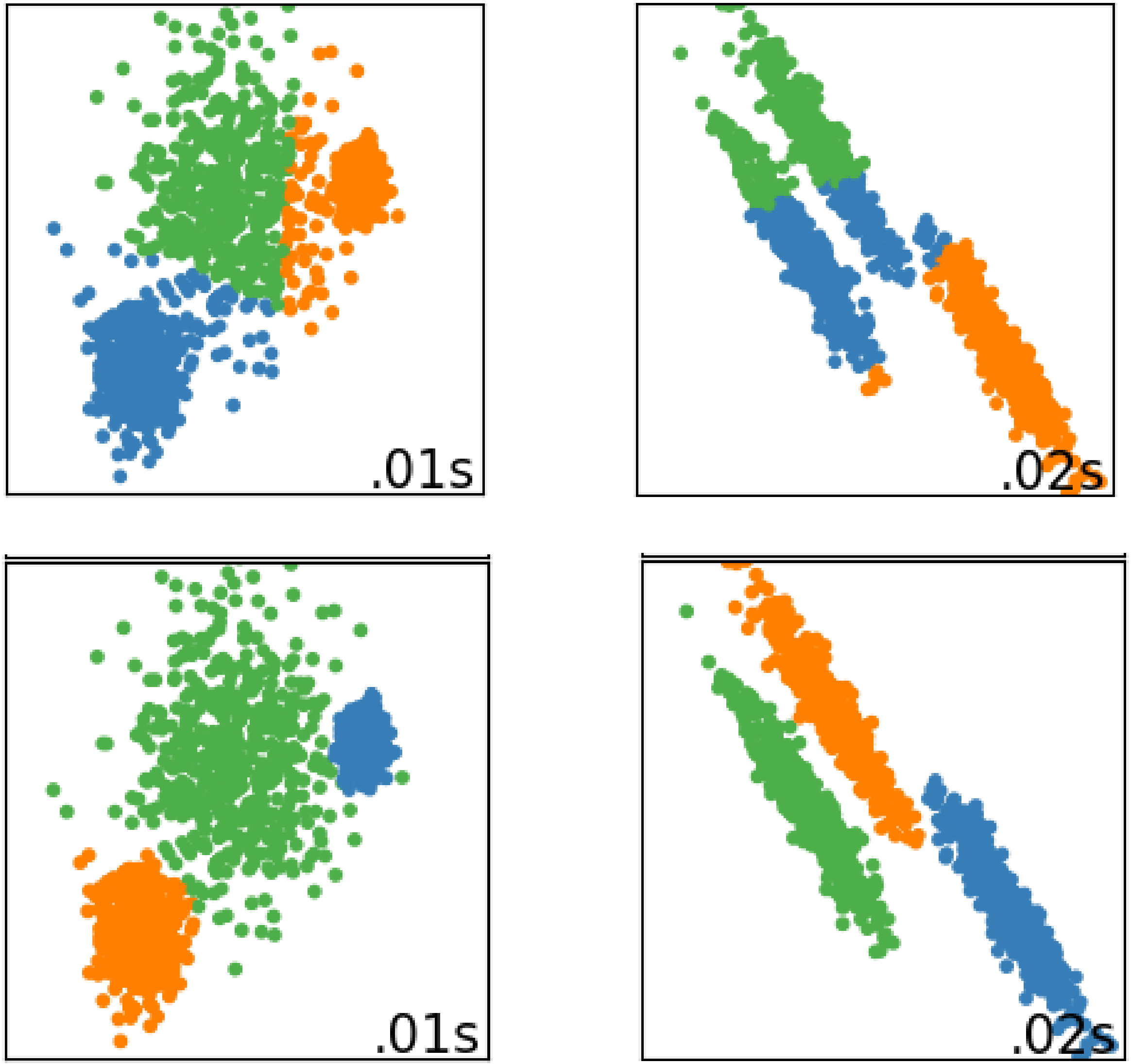
- Data: $x_1, \dots, x_N$ (no labels required)
- Choose number of components in the mixture K
- Randomly select K data points as initial cluster centers (μ_k). Also pick (randomly) Σ_k and non-zero π_k
- Step 1: Re-assign data to mixture softly based on new parameters
- Step 2: Re-compute parameters means based on data assignment
- Repeat Step 1 and Step 2 alternatively until convergence

The above algorithm works for any number of clusters K and for multidimensional features

Works better than K-Means in some cases

K-Means

GMM



GMM can handle clusters of different variances, shapes (ellipsoids), and sample sizes

Image source: https://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

Though it requires data from a cluster to be ellipsoid

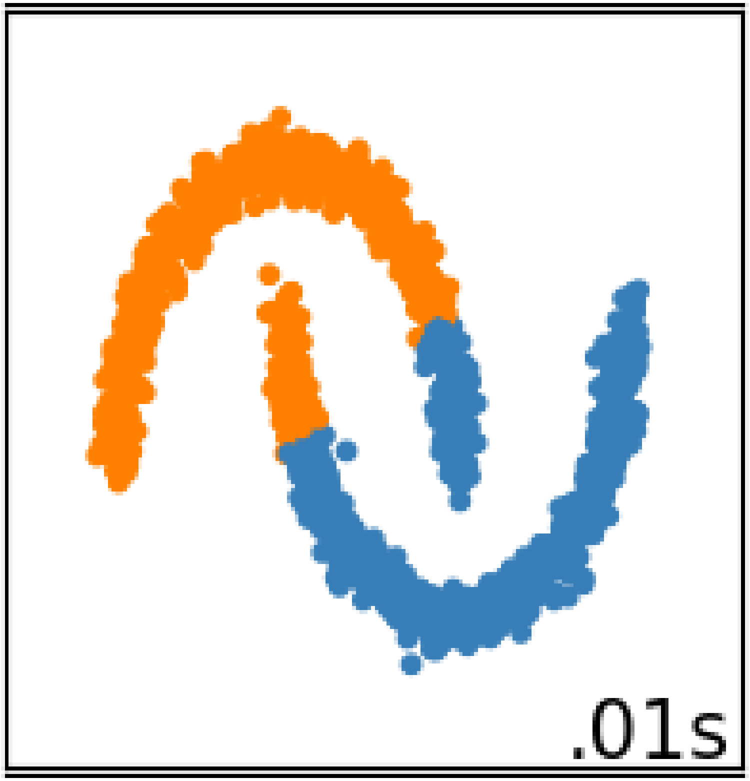


Image source: https://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

Questions?