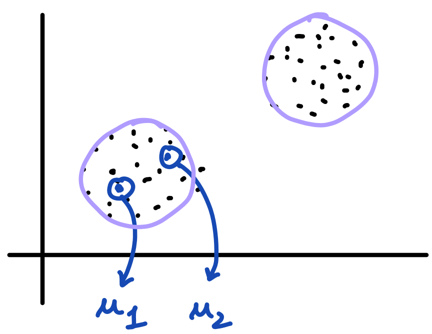
Clustering

- Centroid based clustering (K Means) ✓
- Distribution based clustering (GMM)
- Connectivity based clustering (Hierarchial)
- Density based clustering (DBSCAN)

CENTROID BASED CLUSTERING:

Illustration

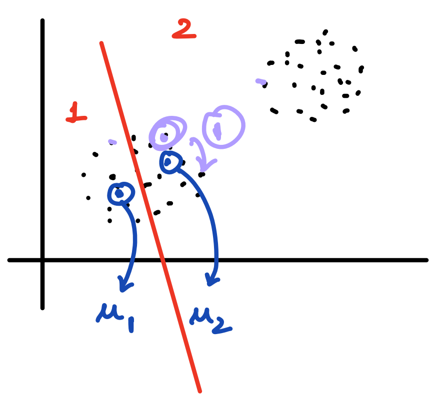


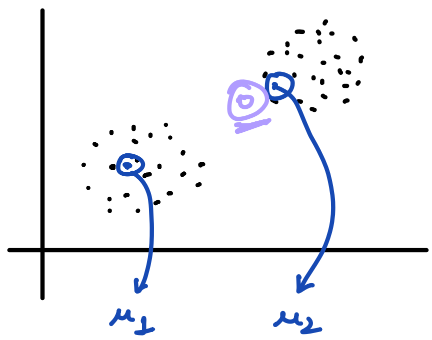
Initialization

Iteration 1

(A) find cluster assignment (assign each point to closest center)

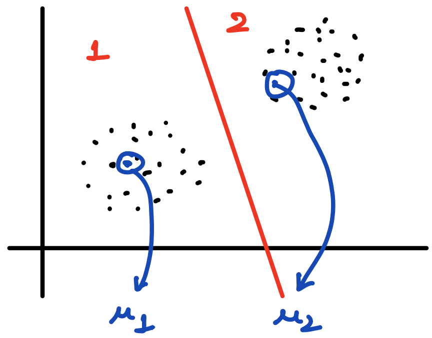
(B) Recompute μ



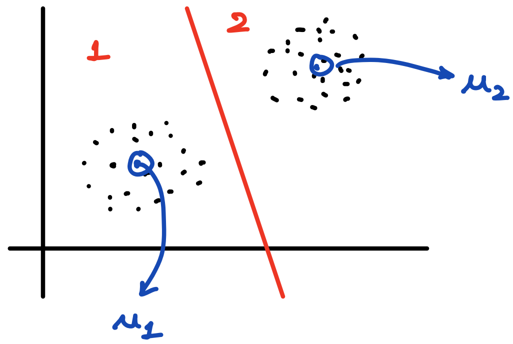


Iteration 2 :

A

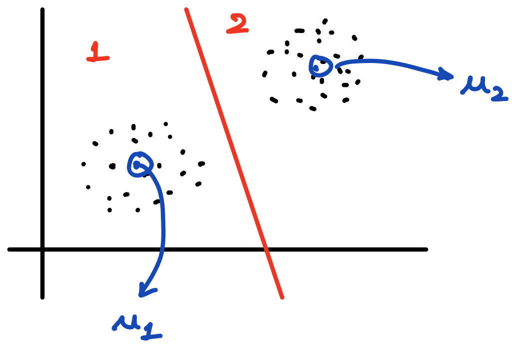


B



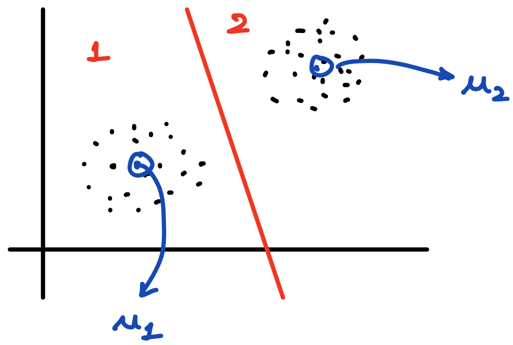
Iteration 3 : Algo converged (Assignment and μ doesnot change)

A

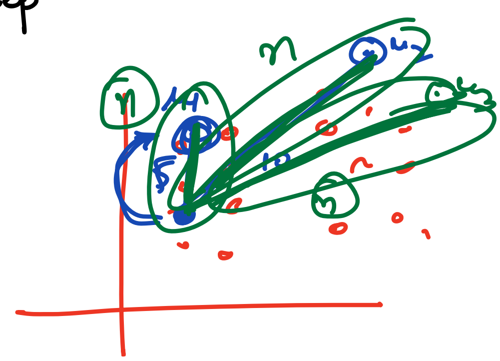


E Step

B



M Step



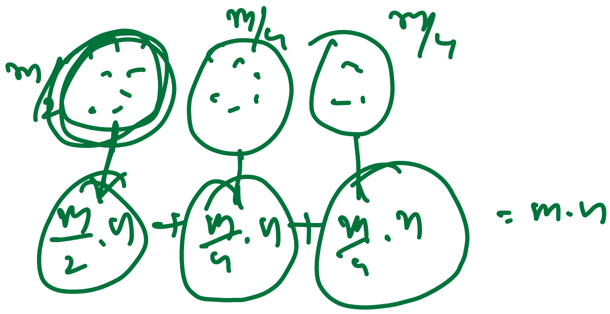
Code (E Step)

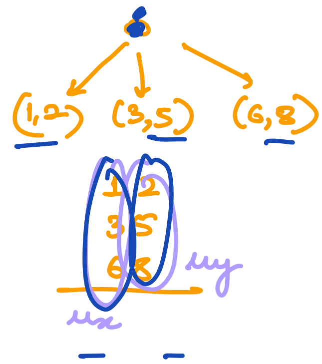
- go to every data point → m
  - calculate the distance of this data point from μ of every cluster → m:k | 5,10
- then find out the minimum distance → 5
- Assign data point to the cluster with min distance → 1st cluster.

Code (M Step)

- Iterate over all clusters

Iterate over all points present in the cluster and take the mean





Time Complexity:

$x^{(i)} \in R^n$

K : no of clusters

m = # points

$O(\underbrace{m \cdot k \cdot n}_{\text{E Step}} + \underbrace{n \cdot m}_{\text{M Step}})$

for a particular cluster: $n \cdot m_k$ → no. of points in k cluster

$\sum_{k=1}^{K} n m_k = n m_1 + n m_2 + \dots \; n m_k$

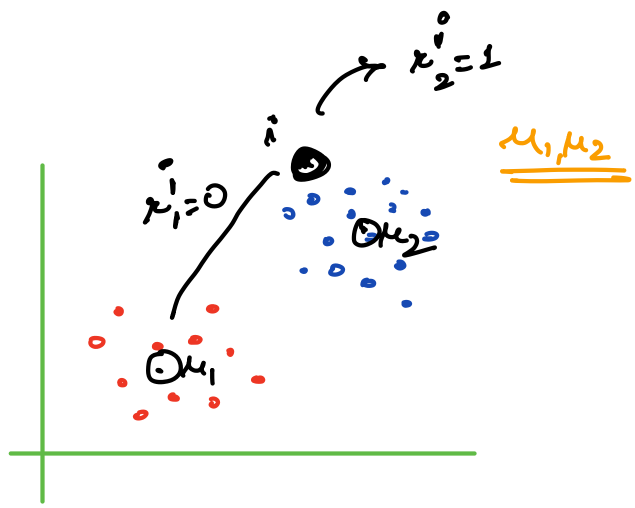
$= n \cdot m$

Math:

$x = \{x^1, x^2, x^3 \dots \; x^m\}$

$\mu = \{\mu_1, \mu_2 \dots \dots \; \mu_k\}$

k = clusters



Loss/Inertia

$$J = \underbrace{\sum_{i=1}^{m}}_{\text{all datapoints}} \underbrace{\sum_{k=1}^{K}}_{\text{all clusters}} x_k^{(i)} \underbrace{(x^{(i)} - \mu_k)^2}_{\text{distance}}$$

compute the distance of every point from every cluster.

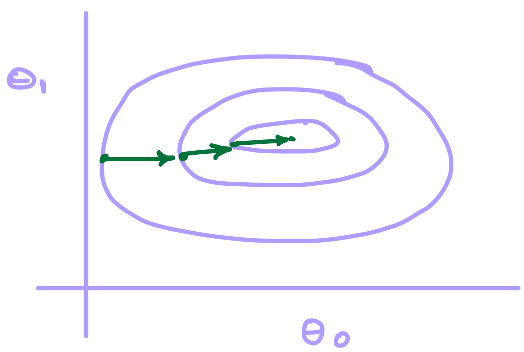
$$r_k^{(i)} = \begin{cases} 1 & \text{if } i\text{th point} \in k\text{th cluster} \\ 0 & \text{otherwise} \end{cases}$$

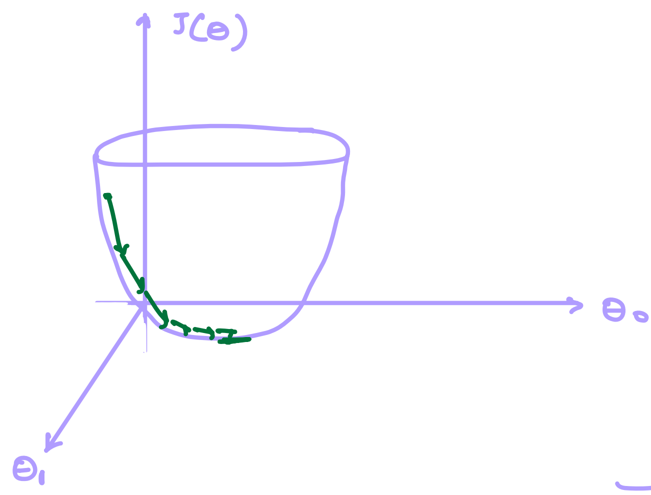
We want to learn $r_k^{(i)}$ and μ_k

This is done by using <u>COORDINATE DESCENT.</u>

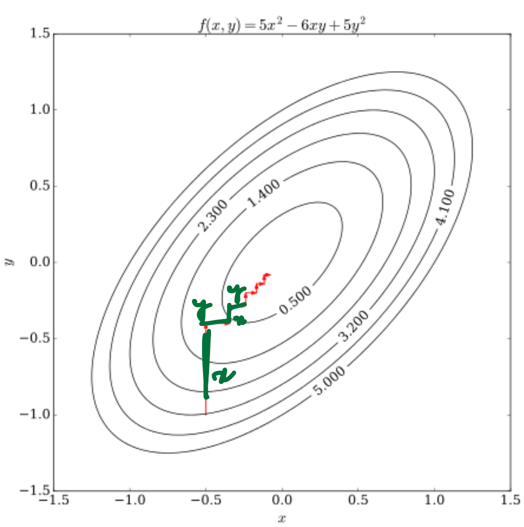
↓

optimise one set of variable at a time while keeping others fixed.





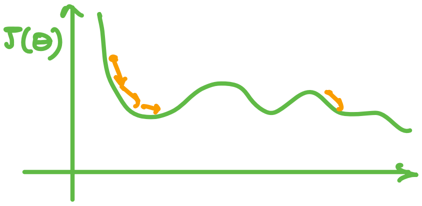
GRADIENT DESCENT



Minimize your loss wrt only one variable at a time.

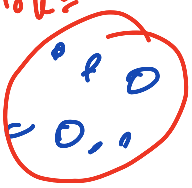
$$r_k^{(i)} = \begin{cases} 1 & \text{if } k = \arg\min_j \|x^{(i)} - \mu_j\|^2 \\ 0 & \text{otherwise} \end{cases}$$

$$J = \sum_{i=1}^{m} \sum_{k=1}^{K} r_k^{(i)} \left(x^{(i)} - \mu_k\right)^2$$



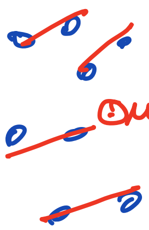
K Means loss fx^n is non convex.

$r_k^i = 0$



$$2 \sum_{i=1}^{m} r_k^{(i)} \left(x^{(i)} - \mu_k\right) = 0$$

$$\sum_{i=1}^{m} r_k^{(i)} x^{(i)} - \mu_k \sum_{i=1}^{m} r_k^{(i)} = 0$$



$r_k^i = 1$

μ_k

$$\mu_k = \frac{\sum_{i=1}^{m} r_k^{(i)} x^{(i)}}{\sum_{i=1}^{m} r_k^{(i)}}$$

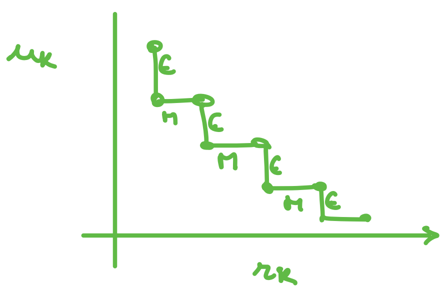
points in kth cluster.

M step

no of points in kth cluster

M step: find μ_k keeping $r_k^{(i)}$ fixed

E step: find $r_k^{(i)}$ keeping μ_k as fixed.



it will converge to local minima.

Algo:

initialize cluster centroids $\mu_1, \mu_2 \ldots \mu_k$ $\quad \mu \in R^n$

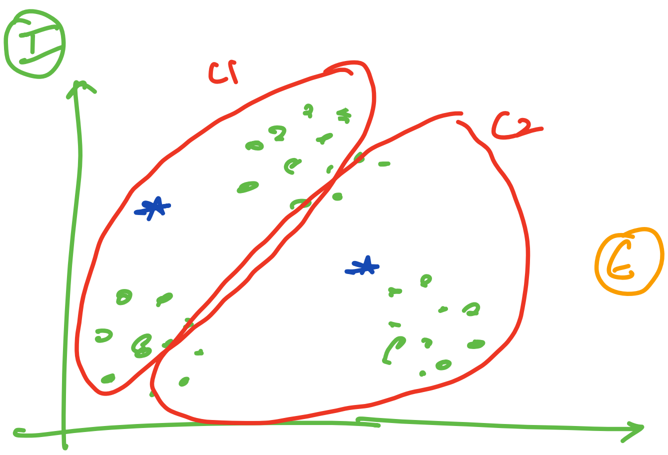
Repeat until convergence:

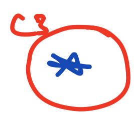
- for every data point i, set

$$r_k^{(i)} = \begin{cases} 1 & \text{if } k = \arg\min_j \|x^{(i)} - \mu_j\|^2 \\ 0 & \text{otherwise} \end{cases}$$

- for each k, set $\mu_k = \dfrac{\sum_{i=1}^{m} r_k^{(i)} x^{(i)}}{\sum_{i=1}^{m} r_k^{(i)}}$

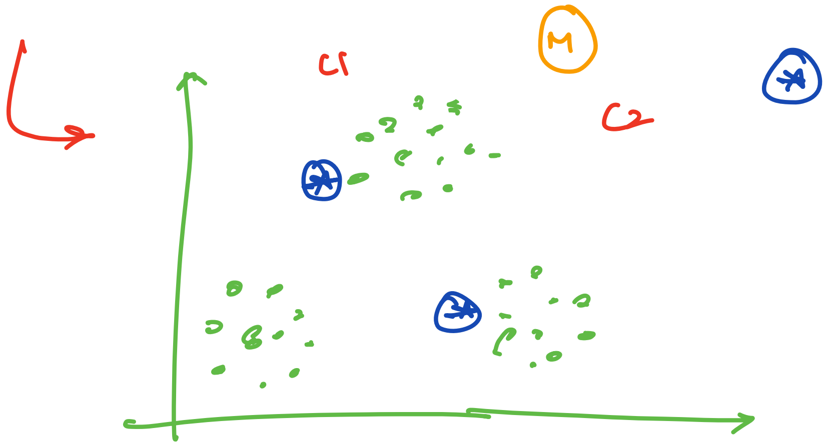
when k-means doesnot work ?
- random initialisation (empty)
- non linearly separable
- one cluster is large as compared to other.



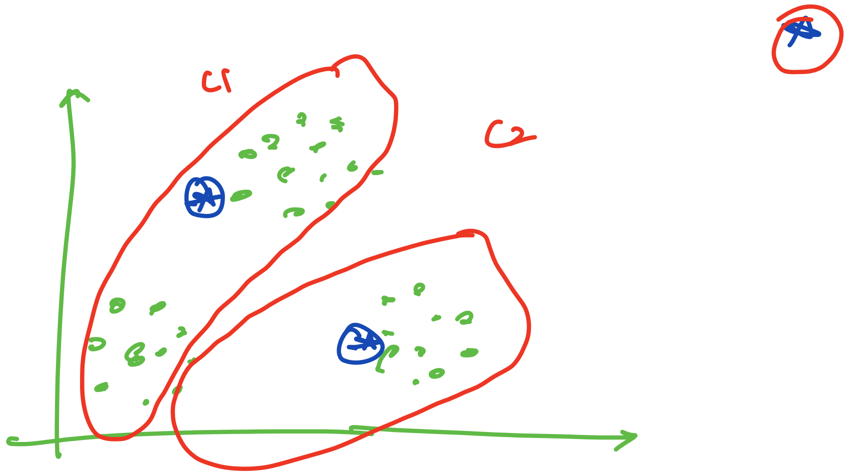


E

C1 → 10
C2 → 15
C3 → 0



II



Initialization

kmeans++ } Assignment

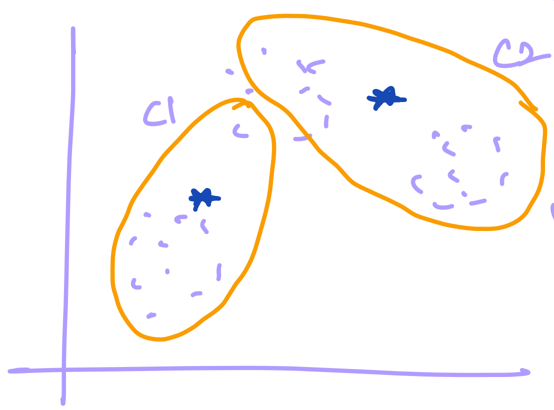
k = ?

→ hyperparameter

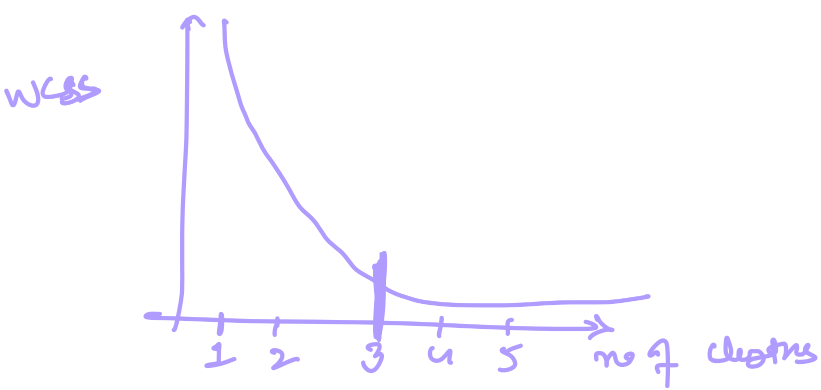
k = 1
2
3
4
5
.
.
.
.

Within cluster sum of squares (WCSS)

k = 2



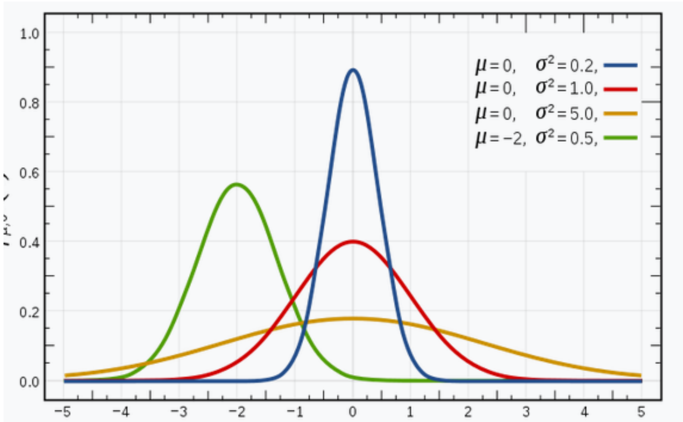
$$WCSS = \sum_{i \in C1} \text{distance}(x^{(i)}, \mu_1) + \sum_{i \in C2} \text{distance}(x^{(i)}, \mu_2)$$



Elbow Method

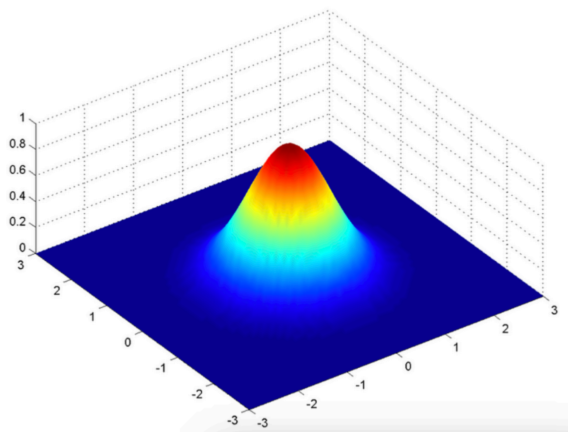
Gaussian Mixture Models

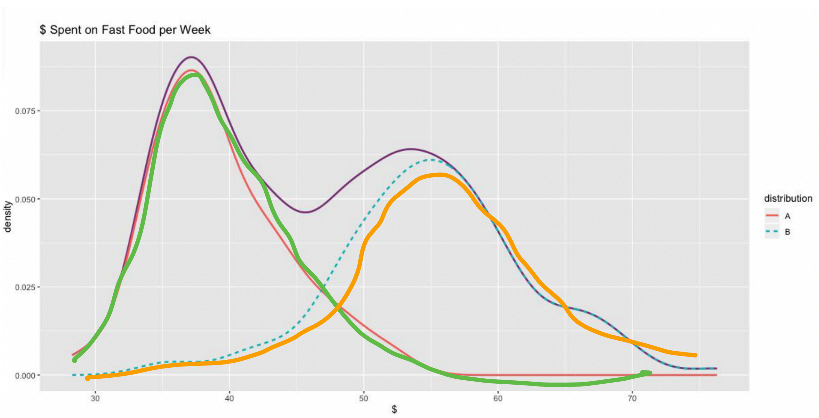
↳ Normal Distribution

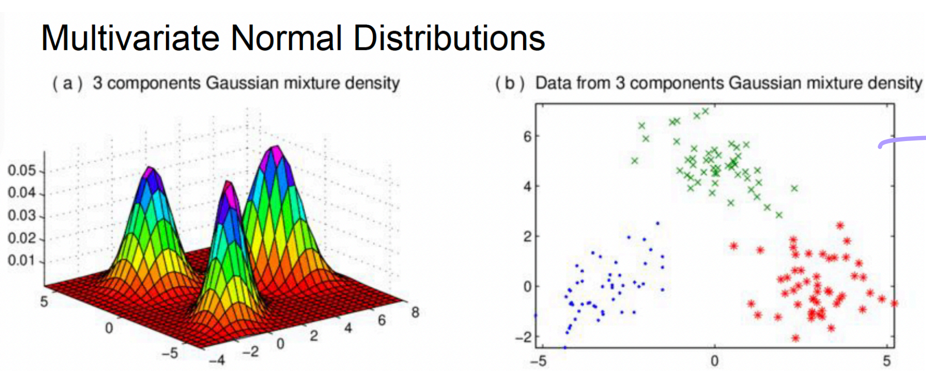


$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$\sigma^2 = \frac{\sum (x_i - \bar{x})^2}{n-1}$$







GMM

1/0

K means

- Hard Assignment
- All Variances the Same
- Roughly same # of datapoints

2

GMM

- Soft (probabilistic) Assignment
- Variances can be different
- Explicitly model # of data points

K-Means Algorithm

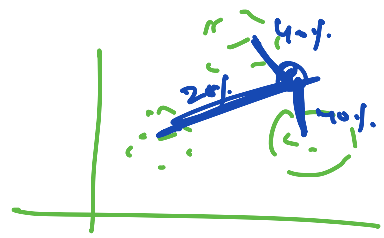
1. Choose **k** random points to be cluster centers
2. For each data point, assign it to the cluster whose center is closest
3. Using these assignments, recalculate the **centers**
4. Repeat 2 and 3 until either:
    a. Cluster membership does not change
    b. Centers change only a tiny amount

$$\sum_{i=1}^{m} \sum_{k=1}^{K} x_k^{(i)}$$

$$\sum_{n=1}^{N} \sum_{k=1}^{K} x_{nk}$$

Gaussian Mixture Model (EM Algorithm)

1. Choose **k** random points to be cluster centers (**or estimate using k-means...etc**)
2. For each data point, calculate the **probability** of belonging to each cluster
3. Using these probability weights, recalculate the **means + variances** (and weights)
4. Repeat 2 and 3 until **distributions converge.**



Data is a weighted combination of various distributions.

$$p(x) = \sum_{k=1}^{K} w_k \, p_k(x)$$

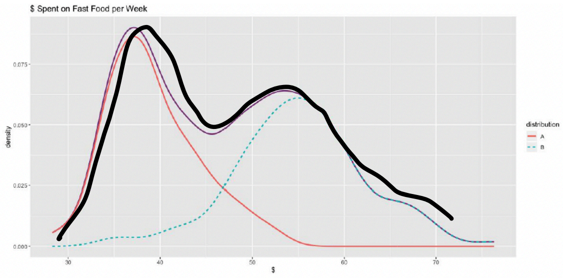
$$\mathrm{Cov}(x,x) = \mathrm{Variance}(x) = \frac{\sum_{i=1}^{n} (x-\bar{x})^2}{n-1}$$

$$\mathrm{Cov}(x,y) = \frac{\sum_{i=1}^{n} (x-\bar{x})(y-\bar{y})}{n-1}$$

$$p(x) = \sum_{k=1}^{K} w_k \boxed{p_k(x)}$$

In Gaussian, it is normal distribution.

$$p(x) = \sum_{k=1}^{K} w_k \boxed{\mathcal{N}(x|\underbrace{\mu_k}_{\text{mean}}, \underbrace{\Sigma_k}_{\text{Covariance}})}$$



$$p(x) = \sum_{k=1}^{K} w_k \, \mathcal{N}(x|\mu_k, \Sigma_k)$$

Probability of being in group k

Likelihood of seeing x in group k

$$p(x) = \sum_{k=1}^{K} \boxed{w_k} \boxed{\mathcal{N}(x|\mu_k, \Sigma_k)}$$

Posterior Probabilities

Prior probability of being in cluster k

Likelihood of seeing x in cluster k

$$\boxed{p(\text{cluster } k|x)} = \frac{\boxed{w_k}\boxed{\mathcal{N}(x|\mu_k, \Sigma_k)}}{\sum_{j=1}^{K} w_j \, \mathcal{N}(x|\mu_j, \Sigma_j)}$$

Posterior probability of being in cluster k

$r_k^{(i)}$

Maximum Likelihood Estimation

Probability of being in group k

Likelihood of seeing x in group k

$$p(x) = \sum_{k=1}^{K} \boxed{w_k} \boxed{\mathcal{N}(x|\mu_k, \Sigma_k)}$$

$$p(\mathbf{X}|\mathbf{w}, \mu, \mathbf{\Sigma}) = p(x_1, x_2, ..., x_n|\mathbf{w}, \mu, \mathbf{\Sigma}) =$$

$$\prod_{n=1}^{N} \sum_{k=1}^{K} \boxed{w_k} \boxed{\mathcal{N}(x_n|\mu_k, \Sigma_k)}$$

$k{=}1$

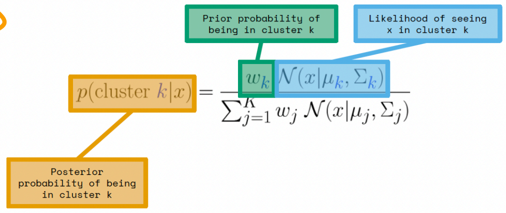
$$log(p(\mathbf{X}|\mathbf{w}, \mu, \mathbf{\Sigma})) = \sum_{n=1}^{N} log \left\{ \sum_{k=1}^{K} w_k \, \mathcal{N}(x_n|\mu_k, \Sigma_k) \right\}$$

Goal: choose w, μ, Σ that maximize the log likelihood

Formulas (E-Step)

$$r_{nk} = \frac{w_k N(x_n|\mu_k, \Sigma_k)}{\sum_j w_j N(x_n|\mu_j, \Sigma_j)}$$

Responsibilities are the posterior probability of a data point being in cluster k



Formulas (M-Step)

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} x_n \qquad N_k = \sum_{n=1}^{N} r_{nk}$$

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} (x_n - \mu_k)(x_n - \mu_k)^T$$

$x^2 = x \cdot x^T$

- GMM does **soft assignment**, every data point belongs to every cluster with some probability
- Data points that are more likely to be in a cluster have **more influence** over its parameters
- GMM uses the EM algorithm to iteratively update the cluster distributions. It first assigning a responsibility to each data point (**E-step**), and then using them to calculate weighted means and variances for each cluster (**M-step**)
- Responsibilities measure the **probability of a data point being in each cluster** (technically the **posterior probability**).
- Responsibilities contain information about **how common a cluster is** as well as the **likelihood of a data point belonging to that cluster**

Ref : https://www.youtube.com/watch?v=sAjOZ9hkn84