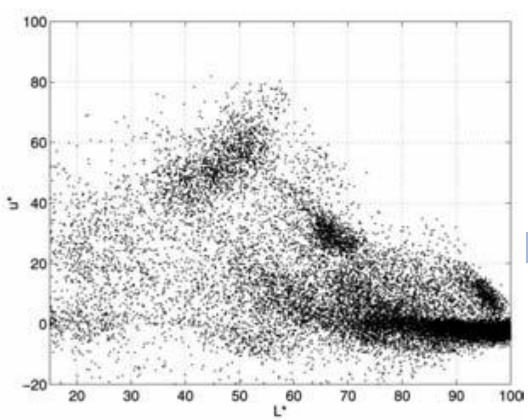
CSE 176 Introduction to Machine Learning

Lecture 6: Gaussian Mixture Model and EM

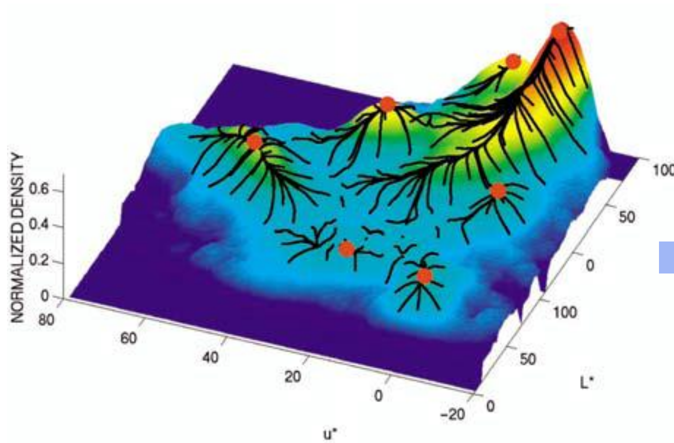
Some materials from Yuri Boykov

From “**means**” towards “**modes**” clustering:

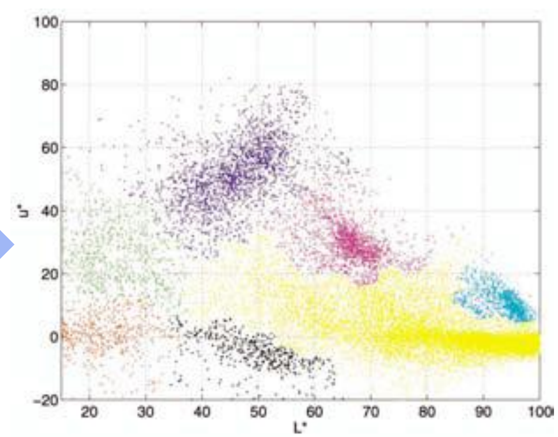
Recap: ***mode clustering***



data points



data histogram and its modes



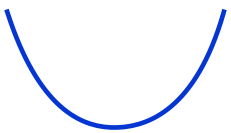
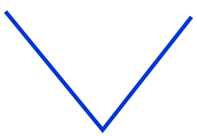
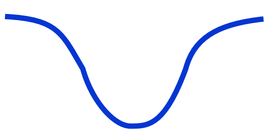
clustering

(generalization)

Recap: Distortion Clustering

can use different "distortion" measures

$$E(S,\mu) \;=\; \sum_{k=1}^{K} \sum_{p \in S_k} \|f_p - \mu_k\|_d$$

examples of distortion measure $\|\cdot\|_d$			interpretation of parameters μ_k
	$\|\cdot\|_d \;=\; \|\cdot\|^2$	squared L_2 norm	**K-means**
	$\|\cdot\|_d \;=\; \|\cdot\|$	absolute L_2 norm	**K-medians**
	$\|\cdot\|_d \;=\; 1 - \exp(-\|\cdot\|^2)$		**K-modes**

Recap: Kernel Density Estimation

Kernel density estimate with *bandwidth* σ: a mixture having one component for each data point:

$$p(\mathbf{x}) = \sum_{n=1}^{N} p(\mathbf{x}|n)p(n) = \frac{1}{N\sigma^D} \sum_{n=1}^{N} K\left(\frac{\mathbf{x}-\mathbf{x}_n}{\sigma}\right) \qquad \mathbf{x} \in \mathbb{R}^D.$$

Usually the kernel K is Gaussian: $K\left(\frac{\mathbf{x}-\mathbf{x}_n}{\sigma}\right) = (2\pi)^{-D/2} \exp\left(-\frac{1}{2}\|(\mathbf{x}-\mathbf{x}_n)/\sigma\|^2\right)$.

Recap: Mean-shift

Mean-shift algorithm: starting from an initial value of $\mathbf{x}$, it iterates the following expression:

$$\mathbf{x} \leftarrow \sum_{n=1}^{N} p(n|\mathbf{x})\mathbf{x}_n \qquad \text{where} \qquad p(n|\mathbf{x}) = \frac{p(\mathbf{x}|n)p(n)}{p(\mathbf{x})} = \frac{\exp\left(-\frac{1}{2}\|(\mathbf{x}-\mathbf{x}_n)/\sigma\|^2\right)}{\sum_{n'=1}^{N}\exp\left(-\frac{1}{2}\|(\mathbf{x}-\mathbf{x}_{n'})/\sigma\|^2\right)}.$$

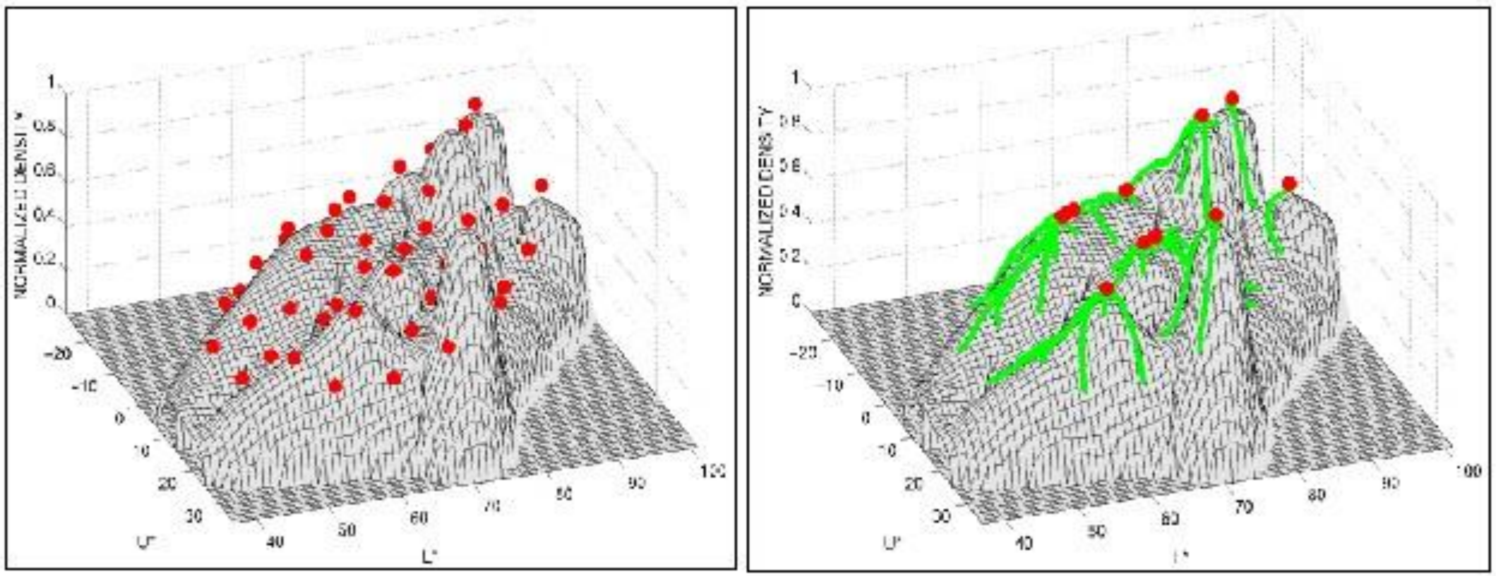
$\sum_{n=1}^{N} p(n|\mathbf{x})\mathbf{x}_n$ can be understood as the weighted average of the N data points using as weights the posterior probabilities $p(n|\mathbf{x})$. The mean-shift algorithm converges to a mode of $p(\mathbf{x})$. Which one it converges to depends on the initialization. By running mean-shift starting at a data point $\mathbf{x}_n$, we effectively assign $\mathbf{x}_n$ to a mode. We repeat for all points $\mathbf{x}_1, \dots, \mathbf{x}_N$.

Recap: Mean Shift

[Fukunaga and Hostetler 1975, Cheng 1995, Comaniciu & Meer 2002]

Multimodal Distributions

- Parallel processing of an initial tessellation.
- Pruning of mode candidates.
- Classification based on the basin of attraction.



Mean shift trajectories

Today's topic

- ❑Gaussian Mixture Model (GMM)
- ❑Expectation-Maximization (EM)

K-means and MLE (maximum likelihood estimation)

"hard" K-means

$$E(S,\mu) \;=\; -\sum_{k=1}^{K}\sum_{p\in S^k}\log P(f_p\,|\,\mu_k)$$

multi-variate (i.e. $x,\mu\in R^N$)
Gaussian distribution
(simple special case $\Sigma=\sigma^2\,\mathbf{I}$)

$$P(x|\mu) \;=\; \frac{1}{\sqrt{(2\pi\sigma^2)^N}}\;\exp-\frac{\|x-\mu\|^2}{2\,\sigma^2}$$

UCMERCED

Towards soft clustering…

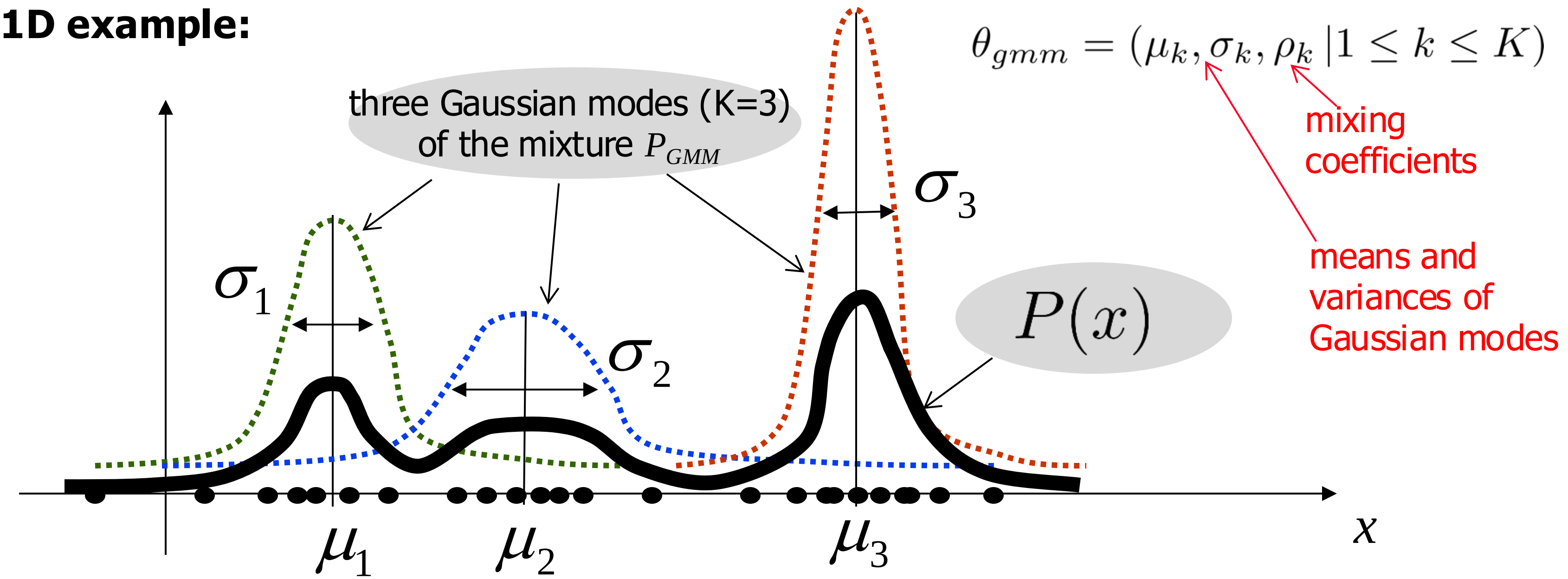
Gaussian Mixture Models (GMM)

- Soft clustering using Gaussian Mixture Model (GMM)

  - no "hard" assignments of points to K distinct (Gaussian) clusters S^k
  - all points are used to estimate parameters of one complex K-mode distribution (GMM)

GMMs estimate "true" data distributions
(continuous density analog of histograms)

simple 1D example:



GMM distribution: $$P_{gmm}(x\,|\,\theta) := \sum_k \rho_k \, P(x\,|\,\mu_k, \sigma_k)$$

Towards soft clustering…

Gaussian Mixture Models (GMM)

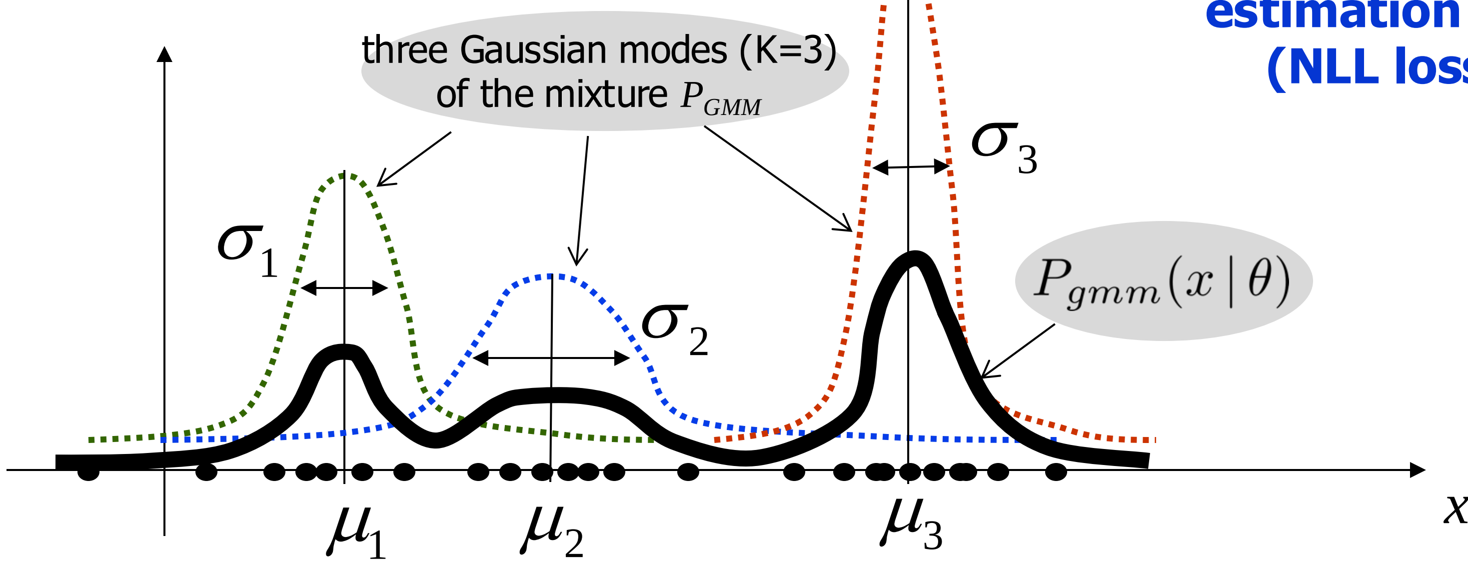
- Soft clustering using Gaussian Mixture Model (GMM)

  - no "hard" assignments of points to K distinct (Gaussian) clusters S^k
  - all points are used to estimate parameters of one complex K-mode distribution (GMM)

approximate optimization via *EM algorithm*

$$E_{gmm}(\theta) \;:=\; -\sum_p \log P_{gmm}(x_p \,|\, \theta)$$

maximum likelihood estimation of θ (NLL loss)



GMM distribution: $$P_{gmm}(x\,|\,\theta) \;:=\; \sum_k \rho_k \, P(x\,|\,\mu_k, \sigma_k)$$

Towards soft clustering…

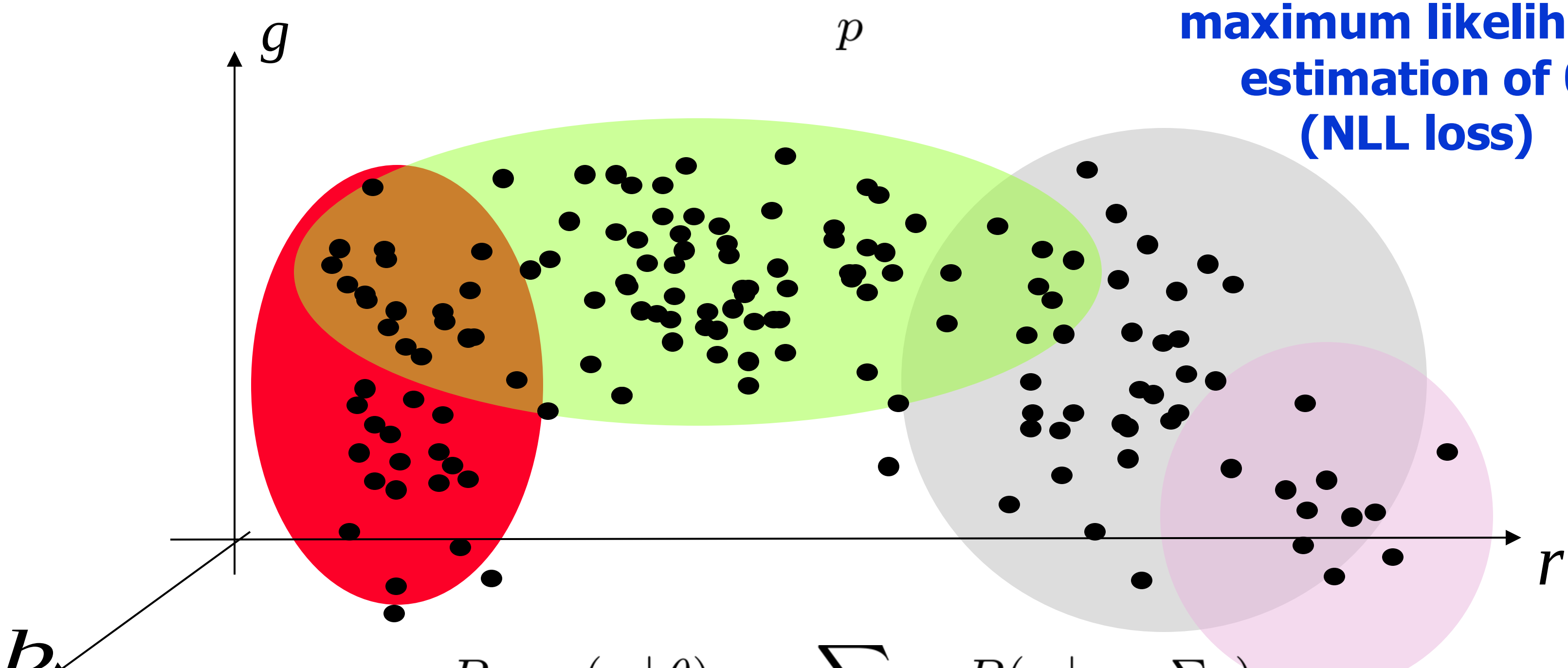
Gaussian Mixture Models (GMM)

- Soft clustering using Gaussian Mixture Model (GMM)

  - no “hard” assignments of points to K distinct (Gaussian) clusters S^k
  - all points are used to estimate parameters of one complex K-mode distribution (GMM)

$$E_{gmm}(\theta) \;:=\; -\sum_p \log P_{gmm}(x_p \,|\, \theta)$$

maximum likelihood estimation of θ (NLL loss)

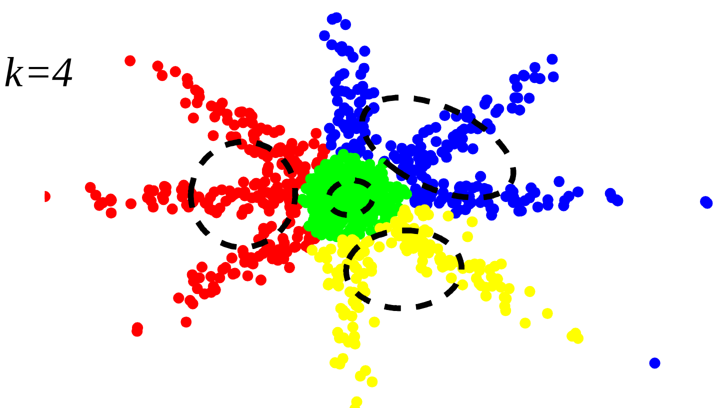


GMM distribution: $P_{gmm}(x \,|\, \theta) \;:=\; \sum_k \rho_k \, P(x \,|\, \mu_k, \Sigma_k)$

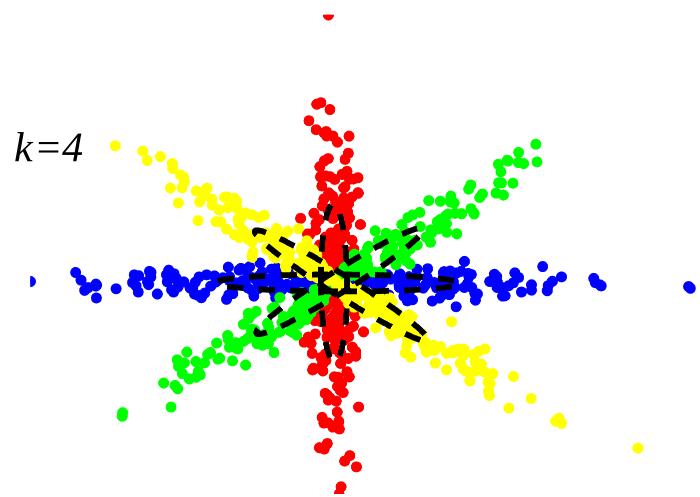
Gaussian clusters/modes in:

(basic) K-means vs. GMM (or fuzzy K-means)

- *hard* assignment to clusters
  - separates data points into multiple Gaussian blobs
- only estimates means μ_i
  - Σ_i can also be added as a cluster parameter (*elliptic K-means*)



- *soft* mode searching
  - estimates data distribution with multiple Gaussian modes
- estimates both mean μ_i and (co)variance Σ_i for each mode

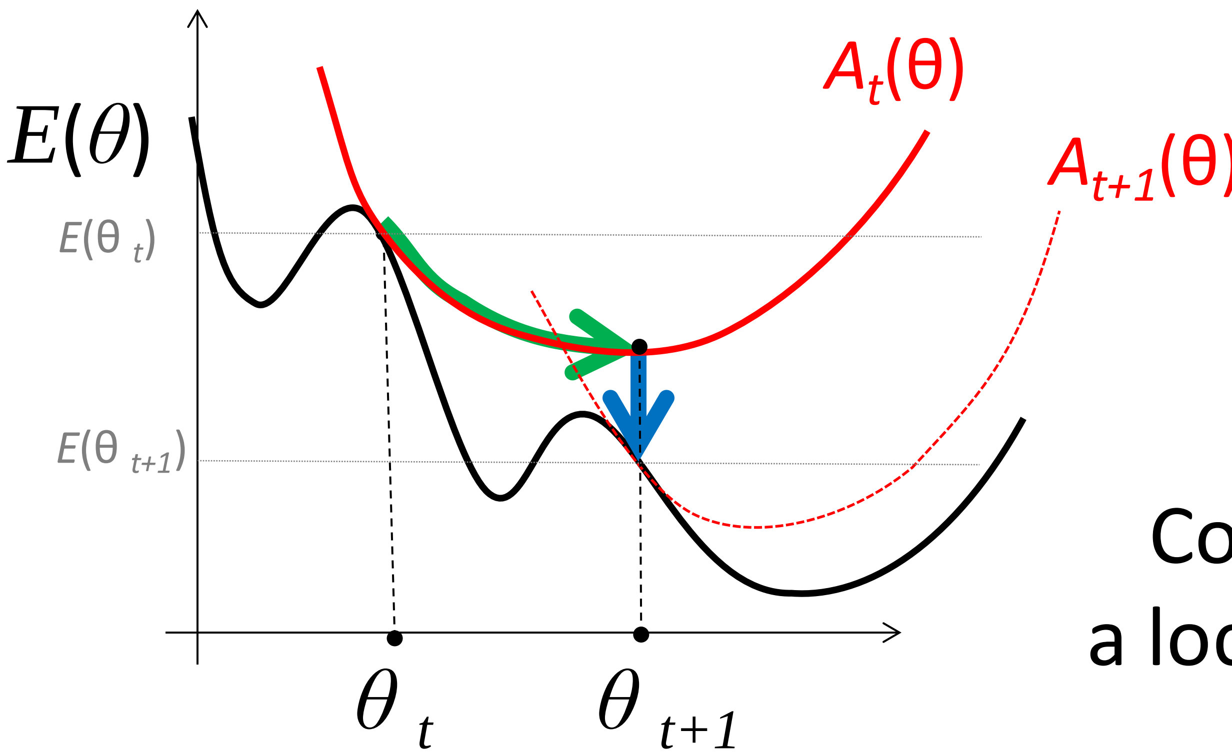


Optimization?

- How to estimate **mean**, **variance**, and **weights** of Gaussian components?
- Bound optimization in general

Bound optimization, in general

(Majorize-Minimize, Auxiliary Function, Surrogate Function)

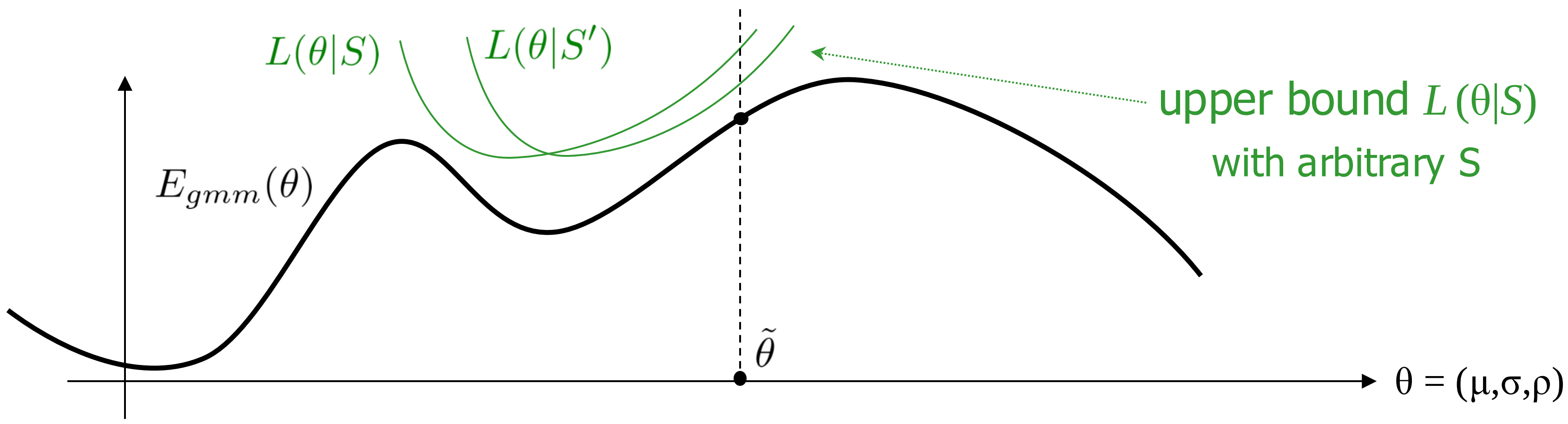


Converges to
a local minimum

Expectation-Maximization (EM)

GMM estimation - optimization of ML objective (sum of Negative Log Likelihoods, a.k.a. **NLL** loss)

$$E_{gmm}(\theta) \;:=\; -\sum_p \log P_{gmm}(x_p \,|\, \theta) \;\;\equiv\;\; -\sum_p \log\left(\sum_k \rho_k\, P(x_p \,|\, \mu_k, \sigma_k)\right)$$



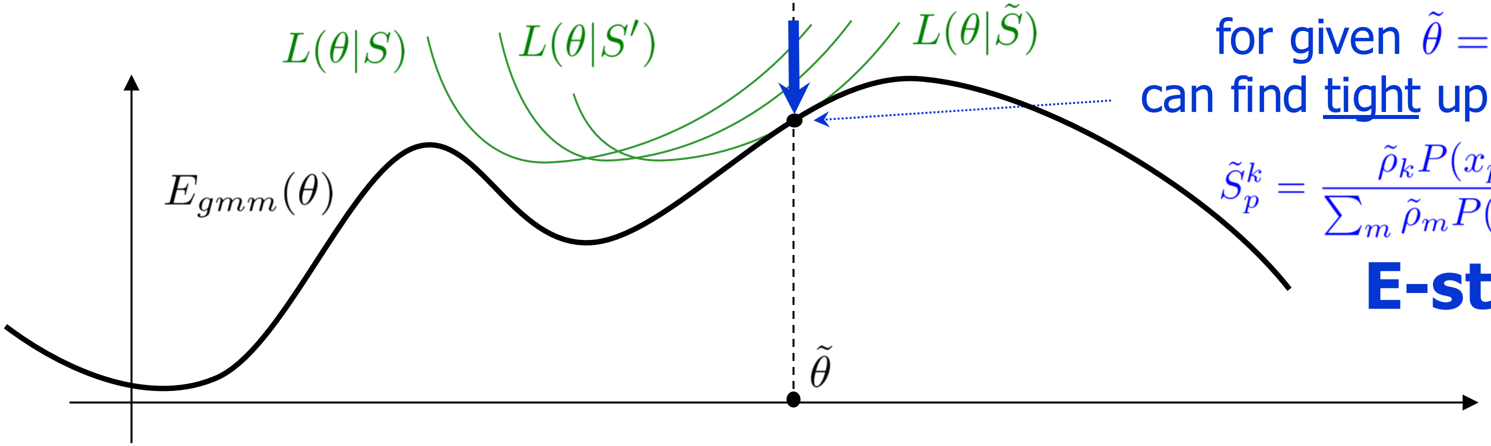
$L\,(\theta|S)$ - for any S defines an upper bounds for $E_{gmm}(\theta)$

$$\leq \;\; -\sum_k \left(\sum_p S_p^k\right) \log \rho_k \;\; - \;\; \sum_k \sum_p S_p^k \;\log P(x_p \,|\, \mu_k, \sigma_k) \; - \; \sum_p \;\; \mathbf{H}(S_p)$$

Expectation-Maximization (EM)

GMM estimation - optimization of ML objective (sum of Negative Log Likelihoods, a.k.a. **NLL** loss)

$$E_{gmm}(\theta) \;:=\; -\sum_p \log P_{gmm}(x_p \,|\, \theta) \;\equiv\; -\sum_p \log\left(\sum_k \rho_k\, P(x_p \,|\, \mu_k, \sigma_k)\right)$$



for given $\tilde{\theta} = (\tilde{\mu}, \tilde{\sigma}, \tilde{\rho})$
can find tight upper bound

$$\tilde{S}_p^k = \frac{\tilde{\rho}_k P(x_p|\tilde{\mu}_k, \tilde{\sigma}_k)}{\sum_m \tilde{\rho}_m P(x_p|\tilde{\mu}_m, \tilde{\sigma}_m)}$$

E-step

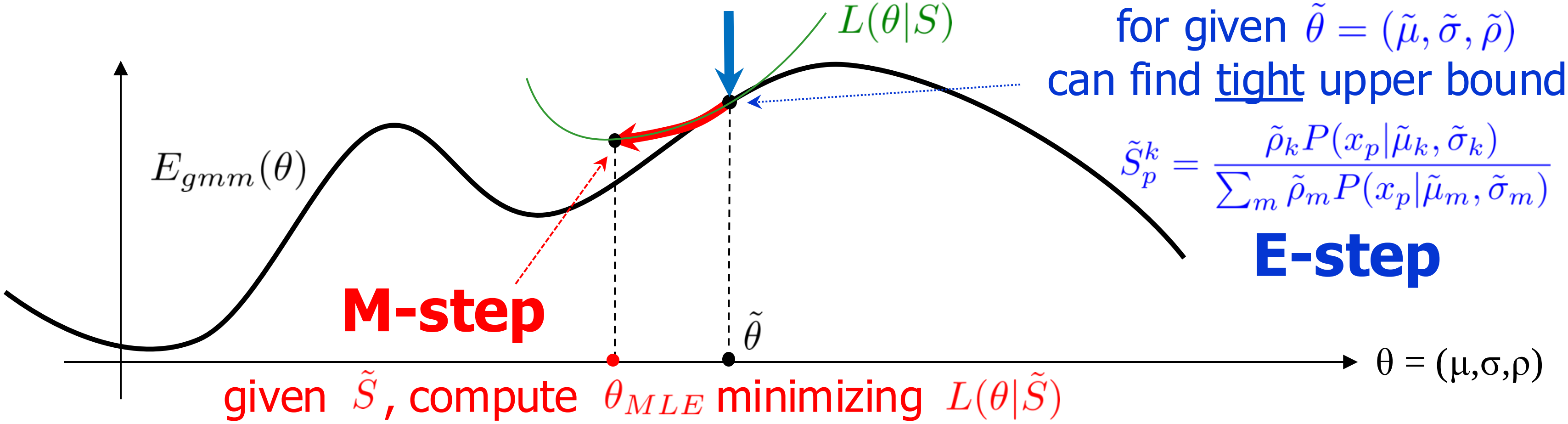
$L(\theta|S)$ - for any S defines an upper bounds for $E_{gmm}(\theta)$

$$\leq \quad -\sum_k \left(\sum_p S_p^k\right) \log \rho_k \;-\; \sum_k \sum_p S_p^k \,\log P(x_p \,|\, \mu_k, \sigma_k) \;-\; \sum_p \mathbf{H}(S_p)$$

Expectation-Maximization (EM)

GMM estimation - optimization of ML objective (sum of Negative Log Likelihoods, a.k.a. **NLL** loss)

$$E_{gmm}(\theta) := -\sum_p \log P_{gmm}(x_p \,|\, \theta) \;\equiv\; -\sum_p \log\left(\sum_k \rho_k \, P(x_p \,|\, \mu_k, \sigma_k)\right)$$



$L(\theta|S)$ - for any S defines an upper bounds for $E_{gmm}(\theta)$

$$\leq \; -\sum_k \overbrace{\left(\sum_p S_p^k\right)} \log \rho_k \;-\; \sum_k \sum_p S_p^k \, \log P(x_p \,|\, \mu_k, \sigma_k) \;-\; \sum_p \; \mathbf{H}(S_p)$$

Expectation-Maximization (EM)

GMM estimation - optimization of ML objective (sum of Negative Log Likelihoods, a.k.a. **NLL** loss)

$$E_{gmm}(\theta) \;:=\; -\sum_p \log P_{gmm}(x_p\,|\,\theta) \;\equiv\; -\sum_p \log\left(\sum_k \rho_k\, P(x_p\,|\,\mu_k,\sigma_k)\right)$$

$$\overset{\forall S_p \in \Delta_K}{\equiv} \; -\sum_p \log\left(\underbrace{\sum_k S_p^k}_{\mathbf{E}_{s_p}} \frac{\rho_k\, P(x_p\,|\,\mu_k,\sigma_k)}{S_p^k}\right)$$

In fact, **equality** holds specifically for

$$S_p^k = \frac{\rho_k P(x_p|\mu_k,\sigma_k)}{\sum_m \rho_m P(x_p|\mu_m,\sigma_m)}$$

(plug-in to check, very easy)

Jensen's inequality
move "log" inside expectation $\mathbf{E}$

$$\le\; -\sum_p \overbrace{\sum_k S_p^k}^{\mathbf{E}_{s_p}} \;\log \frac{\rho_k\, P(x_p\,|\,\mu_k,\sigma_k)}{S_p^k}$$

$$= \; -\sum_p\sum_k S_p^k \;\log \rho_k \;-\sum_p\sum_k S_p^k\; \log P(x_p\,|\,\mu_k,\sigma_k) + \underbrace{\sum_p\sum_k S_p^k \log S_p^k}$$

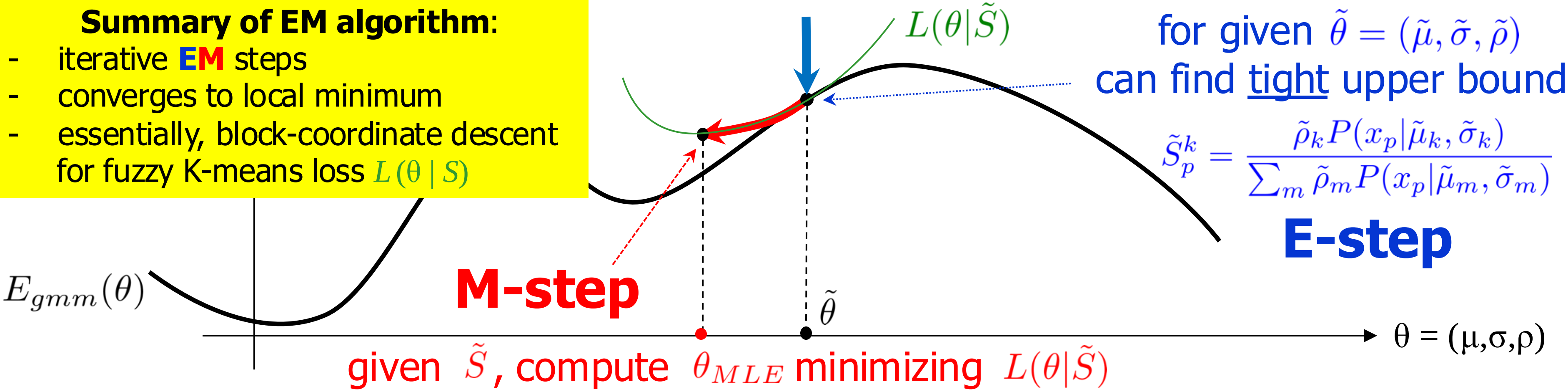
$$= \; -\sum_k\left(\sum_p S_p^k\right)\log\rho_k \;-\; \sum_k\sum_p S_p^k\;\log P(x_p\,|\,\mu_k,\sigma_k) \;-\; \sum_p \;\underset{\text{entropy}}{\mathbf{H}(S_p)}$$

Expectation-Maximization (EM)

GMM estimation - optimization of ML objective (sum of Negative Log Likelihoods, a.k.a. **NLL** loss)

$$E_{gmm}(\theta) := -\sum_p \log P_{gmm}(x_p \,|\, \theta) \;\equiv\; -\sum_p \log\left(\sum_k \rho_k\, P(x_p \,|\, \mu_k, \sigma_k)\right)$$

Summary of EM algorithm:
- iterative **EM** steps
- converges to local minimum
- essentially, block-coordinate descent for fuzzy K-means loss $L(\theta \,|\, S)$



$L(\theta|S)$ - for any S defines an upper bounds for $E_{gmm}(\theta)$

cluster cardinality term | fuzzy K-means loss

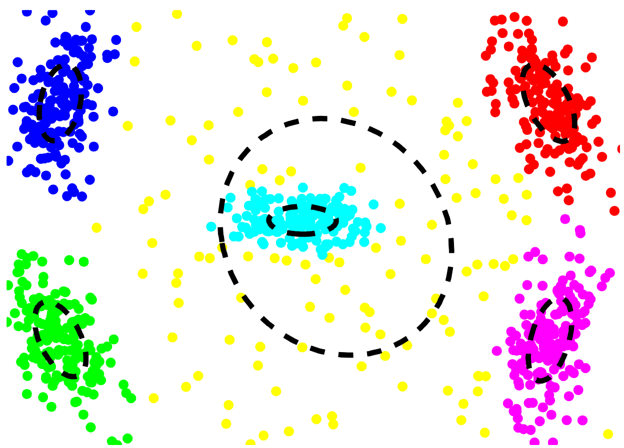
$$\leq \boxed{-\sum_k \Bigg(\overbrace{\sum_p S_p^k}\Bigg) \log \rho_k} \quad \boxed{-\sum_k \sum_p S_p^k \, \log P(x_p \,|\, \mu_k, \sigma_k) \; - \sum_p \mathbf{H}(S_p)}$$

Gaussian clusters/modes in:

(basic) K-means vs. GMM (or fuzzy K-means)

- *hard* assignment to clusters
  - separates data points into multiple Gaussian blobs
- only estimates means μ_i
  - Σ_i can also be added as a cluster parameter (*elliptic K-means*)

k=6

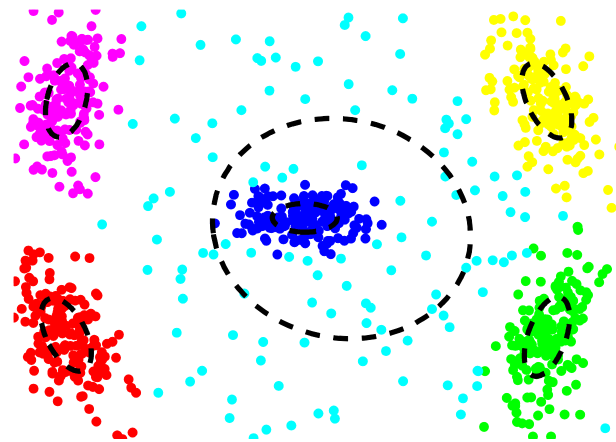


(elliptic) K-means

color indicates assigned cluster

- *soft* mode searching
  - estimates data distribution with multiple Gaussian modes
- estimates both mean μ_i and (co)variance Σ_i for each mode

k=6



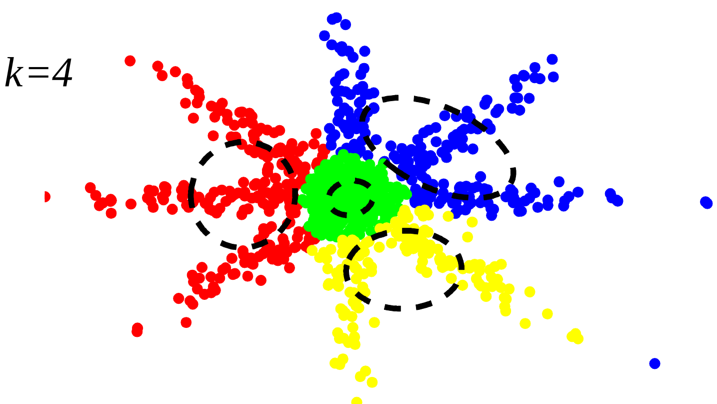
GMM

color indicates locally strongest mode

Gaussian clusters/modes in:

(basic) K-means vs. GMM (or fuzzy K-means)

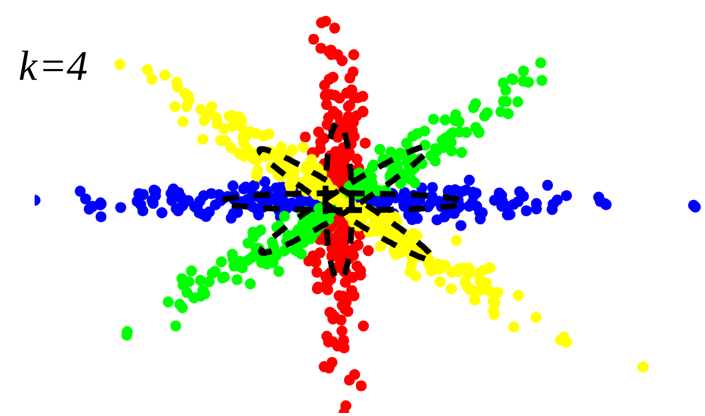
- *hard* assignment to clusters
  - separates data points into multiple Gaussian blobs
- only estimates means μ_i
  - Σ_i can also be added as a cluster parameter (*elliptic K-means*)



hard clustering may not work well when clusters overlap

(may not be a problem in image segmentation, since objects do not "overlap" in RGBXY)

- *soft* mode searching
  - estimates data distribution with multiple Gaussian modes
- estimates both mean μ_i and (co)variance Σ_i for each mode



While this is an optimal GMM, standard EM may converge to a bad solution (local minimum)

Gaussian clusters/modes in:

(basic) K-means vs. GMM (or fuzzy K-means)

K-means

- *hard* assignment to clusters
  - separates data points into multiple Gaussian blobs
- only estimates means μ_i
  - Σ_i can also be added as a cluster parameter (*elliptic K-means*)
- computationally cheap steps
  (block-coordinate descent, Lloyd's algorithm)
  unless estimating covariances Σ_k (elliptic case)
- sensitive to local minima

GMM

- *soft* mode searching
  - estimates data distribution with multiple Gaussian modes
- estimates both mean μ_i and (co)variance Σ_i for each mode
- expensive steps (mostly due to Σ_k)
  (iterative EM algorithm)
- sensitive to local minima
- **becomes slow to estimate Σ from high dimensional data, also needs lots of points**