

Expectation Maximization

Aaron C. Courville
Université de Montréal

Note: Material for the slides is taken directly from a presentation
prepared by Christopher M. Bishop

Learning in DAGs

- Two things could be learned:
 - Graph structure
 - Parameters governing the conditional probability distribution
- Learning the structure often involves a search over candidate structures and a method to score each structure.
 - In practice, it is often difficult to extract the conditional independence relationships that make DAGs so appealing in the first place.
 - MCMC methods are also used to search over the space of structures.
- We will focus on the problem of learning the parameters.

Maximum Likelihood Learning

- Consider the parameter set $\theta = (\theta_1, \dots, \theta)$ which govern the conditional probability distributions $P(X_i | Pa_i, \theta)$.
- One way to learn the parameters is to maximize the likelihood (or probability) of the *data* D (set of observed variables):

$$\begin{aligned}\theta_{ML} &= \arg \max_{\theta} \{ \ln P(D | \theta) \} \\ &= \arg \max_{\theta} \left\{ \ln \prod_{n=1}^N P(x_n | \theta) \right\} \\ &= \arg \max_{\theta} \left\{ \sum_{n=1}^N \ln P(x_n | \theta) \right\} \\ &= \arg \max_{\theta} \left\{ \sum_{n=1}^N \ln \sum_{h_n} P(x_n, h_n | \theta) \right\}\end{aligned}$$

Sum over H (latents)
could be problematic

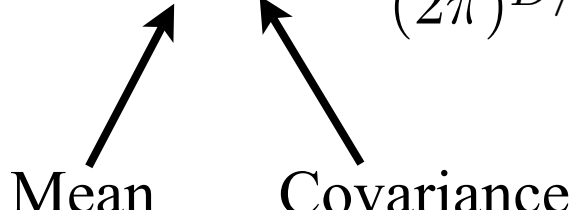
- When there are no latent variables, the situation is much easier:
 - If we can condition on all the variables, the graph factors by d-separation and we can estimate the parameters for all the $P(X_i | Pa_i, \theta)$ independently (eg. Naïve Bayes Classifier).

ML Example: Multivariate Gaussian

Multivariate Gaussian distribution:

$$p(X = x) = \mathcal{N}(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right\}$$

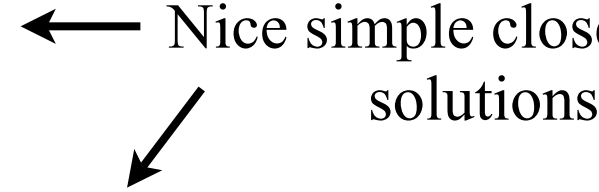
Mean Covariance



Maximum Likelihood Solution:

$$\mu_{ML} = \frac{1}{N} \sum_{n=1}^N x_n$$

← Nice simple closed-form solutions

$$\Sigma_{ML} = \frac{1}{N} \sum_{n=1}^N (x_n - \mu_{ML})(x_n - \mu_{ML})^T$$


Gaussian Mixture Models

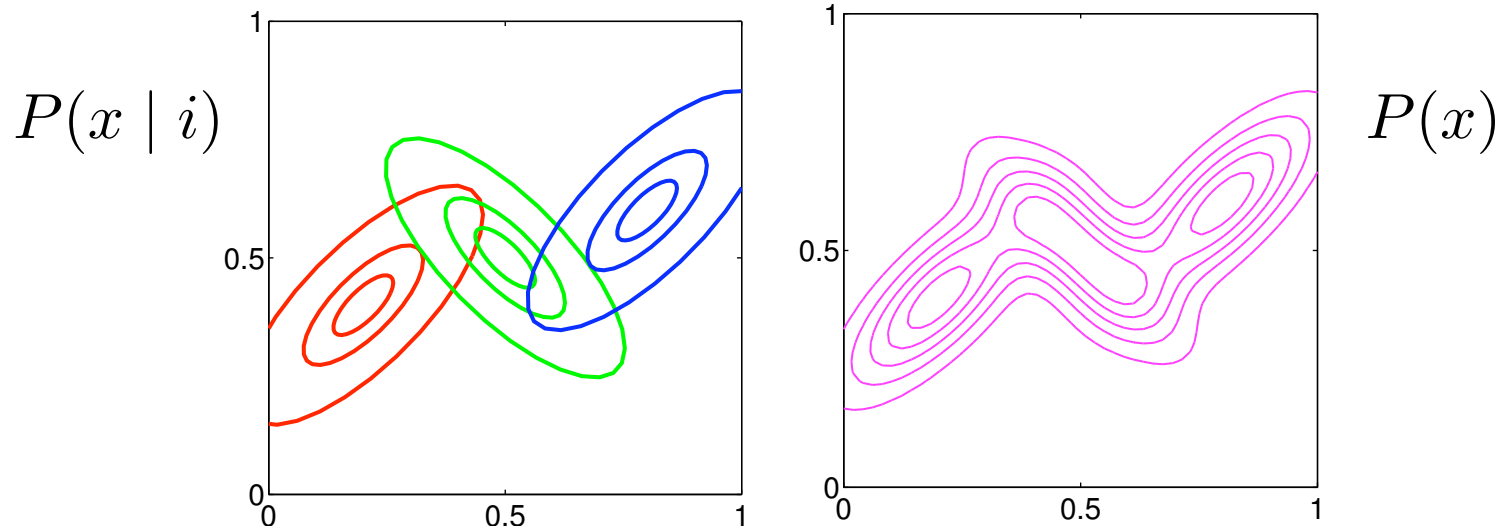
- Now let's consider a random variable distributed according to a mixture of Gaussians.
- **Conditional distributions for a D -dimensional X :**

$$P(I = i) = w_i$$

$$p(X = x | I = i) = \mathcal{N}(x | \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2} |\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i)\right)$$

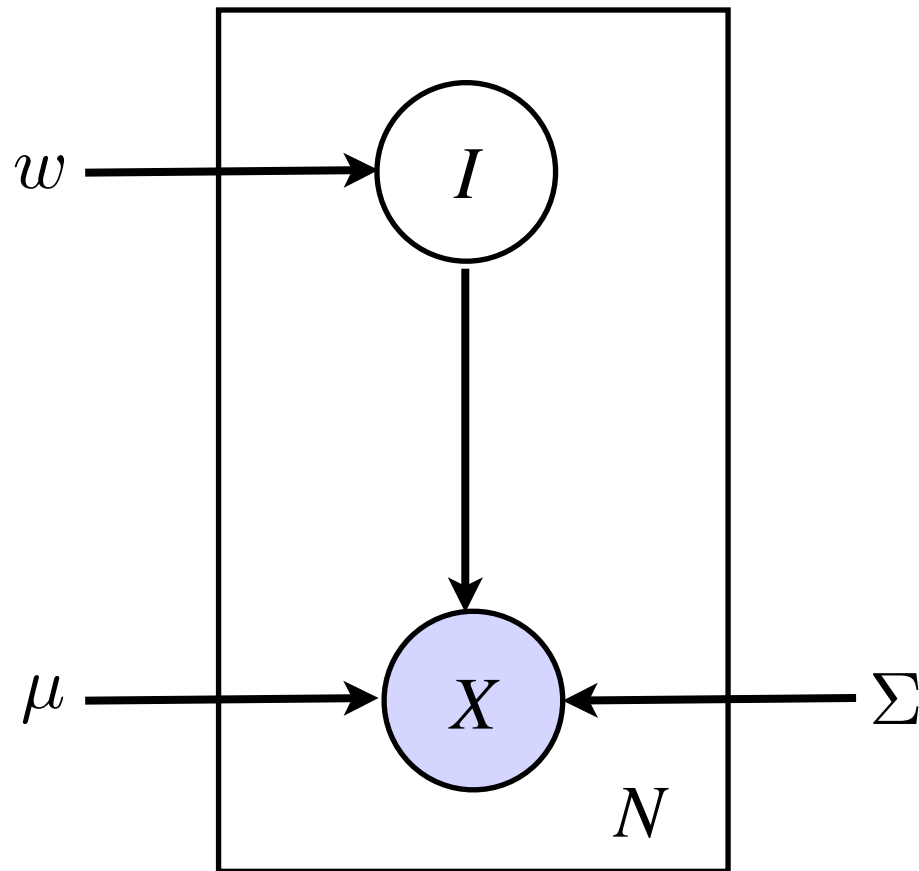
where I is an index over the *multivariate* Gaussian components in the mixture and the mixing proportion, w_i , is the marginal probability that X is generated by mixture component i .

- **Marginal distributions:** $p(X = x) = \sum_i p(X = x | I = i)P(I = i) = \sum_i w_i \mathcal{N}(x | \mu_i, \Sigma)$



Gaussian Mixture Models (cont.)

Graphical model:



Maximum Likelihood of GMM

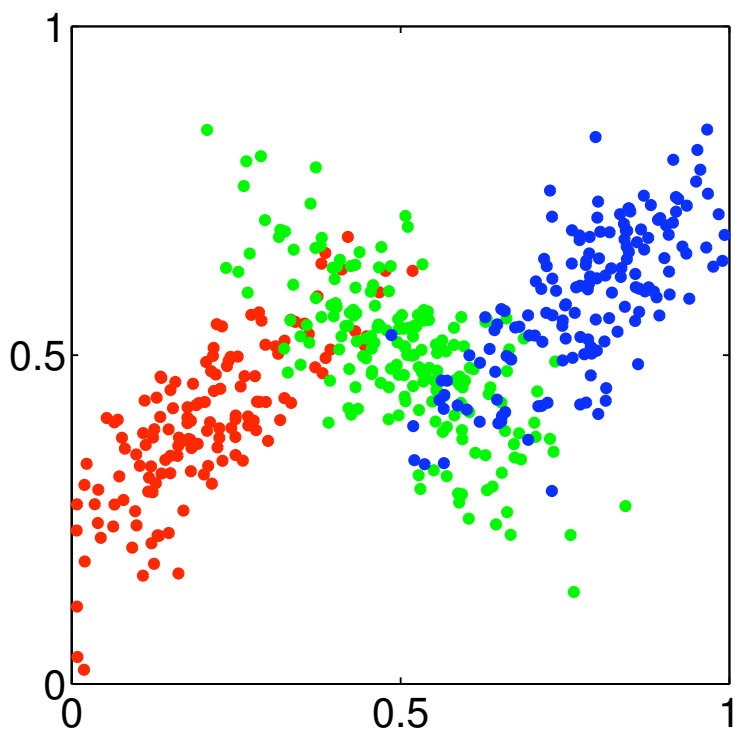
- Log likelihood function:

$$\ln p(D | w, \mu, \Sigma) = \sum_{n=1}^N \ln \left\{ \sum_i w_i \mathcal{N}(x_n | \mu_i, \Sigma_i) \right\}$$

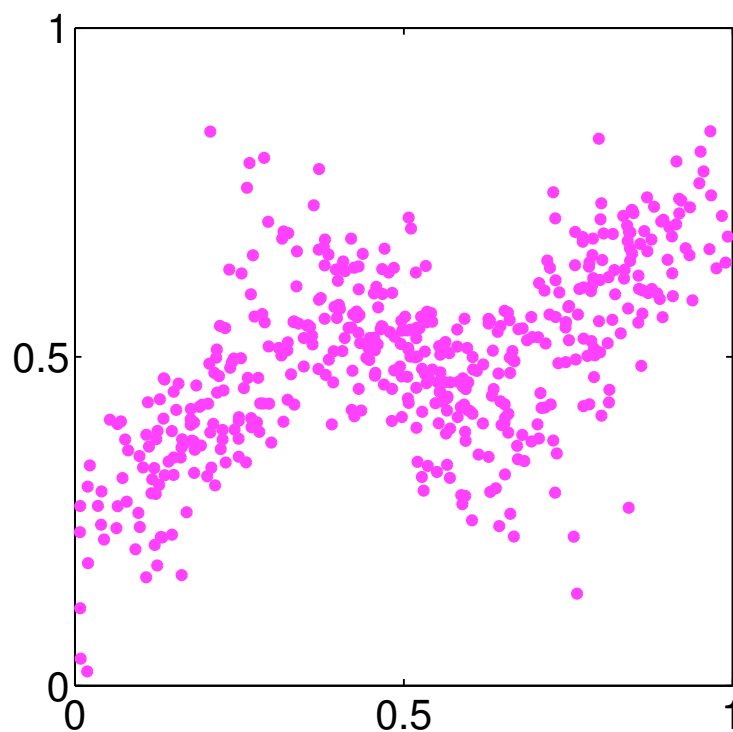
- Sum over mixture components appears inside the log
 - No closed form ML solution

Complete and Incomplete Data

- If we knew the mixture component identities, things would be easier.
 - This is the difference between *complete data* and *incomplete data*:



complete

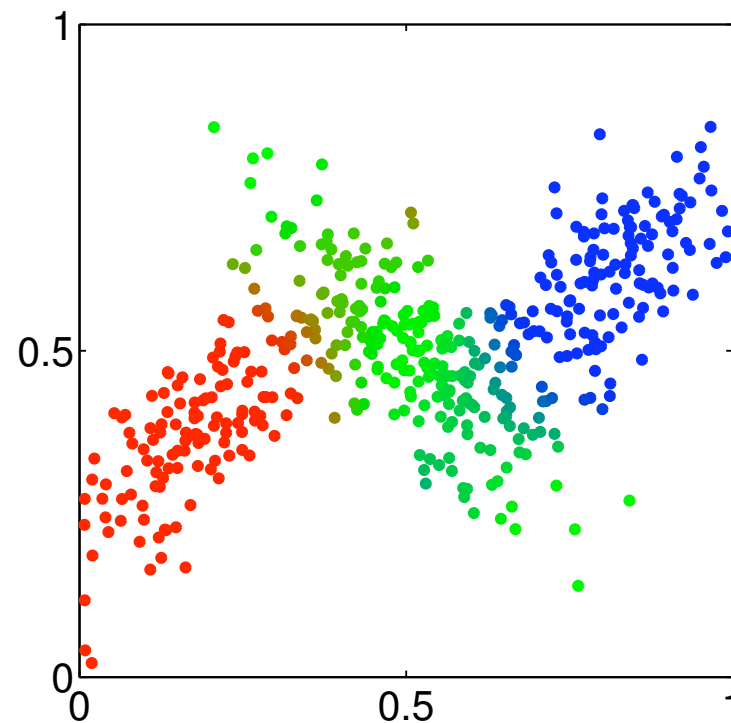
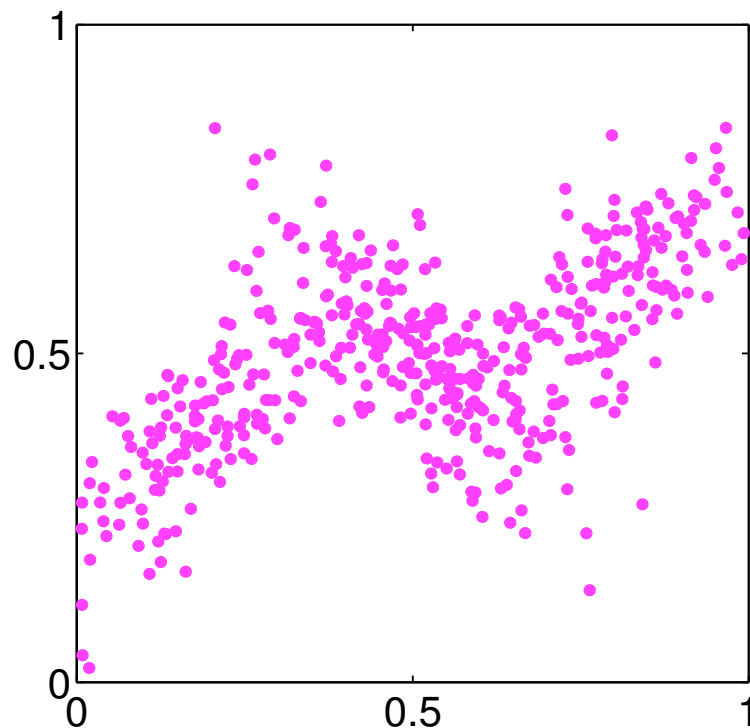


incomplete

- The complete data picture treats the latent variables as *missing data*.

The makings of an iterative scheme

- Problem: we don't know the values of the latent variables (they're missing!)
- **The EM idea:** instead maximize the expected value of the complete-data log likelihood
 - with the expectation w.r.t. $P(\text{latents} \mid \text{observed}, \text{parameters})$



Expectation-Maximization Algorithm

- E-step (expectation): evaluate the posterior distribution $P(Z | X, \theta^{old})$ using current estimate, θ^{old} , of the parameters.
- M-step (maximization): re-estimate θ by maximizing the expected *complete-data* log-likelihood:

$$\begin{aligned}\theta^{new} &= \arg \max_{\theta} Q(\theta, \theta^{old}) \\ &= \arg \max_{\theta} \left\{ \sum_Z P(Z | X, \theta^{old}) \ln P(X, Z | \theta) \right\}\end{aligned}$$

- Note that the log and the summation have been exchanged - this will often make the summation tractable.
- Iterate E and M steps until convergence. Guaranteed to converge to a local optimum with linear convergence rate

E Step: Mixture of Gaussians

- Calculate $P(I_n | X_n, \theta)$ for each observed example X_n

$$X_n = [X_{1,n}, \dots, X_{d,n}, \dots, X_{D,n}]^T$$

$$\begin{aligned} P(I_n = i | X_n, \theta) &= \frac{P(I_n = i | w_i)P(X_n | I_n = i, \mu_i, \Sigma_i)}{P(X_n)} \\ &= \frac{P(I_n = i | w_i)P(X_n | I_n = i, \mu_i, \Sigma_i)}{\sum_i P(I_n = i | w_i)P(X_n | I_n = i, \mu_i, \Sigma_i)} \\ &= \frac{w_i \mathcal{N}(X_n | \mu_i, \Sigma_i)}{\sum_j w_j \mathcal{N}(X_n | \mu_j, \Sigma_j)} \end{aligned}$$

Where $\theta = \{\theta_1, \dots, \theta_i, \dots, \theta_K\}$, $\theta_i = \{w_i, \mu_i, \Sigma_i\}$ and $\mathcal{N}(\cdot)$ is the multivariate Gaussian probability density function.

M Step: Mixture of Gaussians

- For mixtures of Gaussians:

$$\theta \leftarrow \arg \max_{\theta'} \left\{ \sum_n \sum_i P(I_n = i | X_n = x_n, \theta) \ln P(X_n = x_n, I_n = i | \theta') \right\}$$

- We already computed $P(I_n = i | X_n = x_n, \theta)$ in the E step and we can decompose the joint $P(X_n = x_n, I_n = i | \theta')$:

$$\begin{aligned} \sum_n \sum_i P(i_n | x_n, \theta) \ln p(x_n, i_n | \theta') &= \sum_n \sum_i P(i_n | x_n, \theta) \ln p(x_n | i_n, \theta') P(i_n | \theta') \\ &= \sum_n \sum_i P(i_n | x_n, \theta) \ln w'_i + \sum_n \sum_i P(i_n | x_n, \theta) \ln \mathcal{N}(x_n | \mu'_i, \Sigma'_i) \end{aligned}$$

- Now we maximize this expression w.r.t θ' (*on to the M step*)

M Step: Mixture of Gaussians (cont.)

- Let's consider updating w_i : (subject to the constraint $\sum_i w_i = 1$)

$$\frac{\partial}{\partial w'_i} \left[\sum_n \sum_i P(i_n | x_n, \theta) \ln w'_i + \lambda \left(\sum_i w'_i - 1 \right) \right] = 0$$

$$\sum_{n=1}^N \frac{1}{w'_i} P(i_n | x_n, \theta) + \lambda = 0$$

$$w_i \leftarrow \frac{1}{N} \sum_{n=1}^N P(i_n | x_n, \theta)$$

M Step: Mixture of Gaussians (cont.)

- Now consider updating the mean vectors μ_i :

$$\frac{\partial}{\partial \mu'_i} \left[\sum_n \sum_i P(i_n | x_n, \theta) \ln \mathcal{N}(x_n | \mu'_i, \Sigma'_i) \right] = 0$$

$$\frac{\partial}{\partial \mu'_i} \left[\sum_n \sum_i P(i_n | x_n, \theta) \left(-\frac{1}{2} \ln(|\Sigma'_i|) - \frac{1}{2} (x_n - \mu_i)^T \Sigma'^{-1}_i (x_n - \mu_i) \right) \right] = 0$$

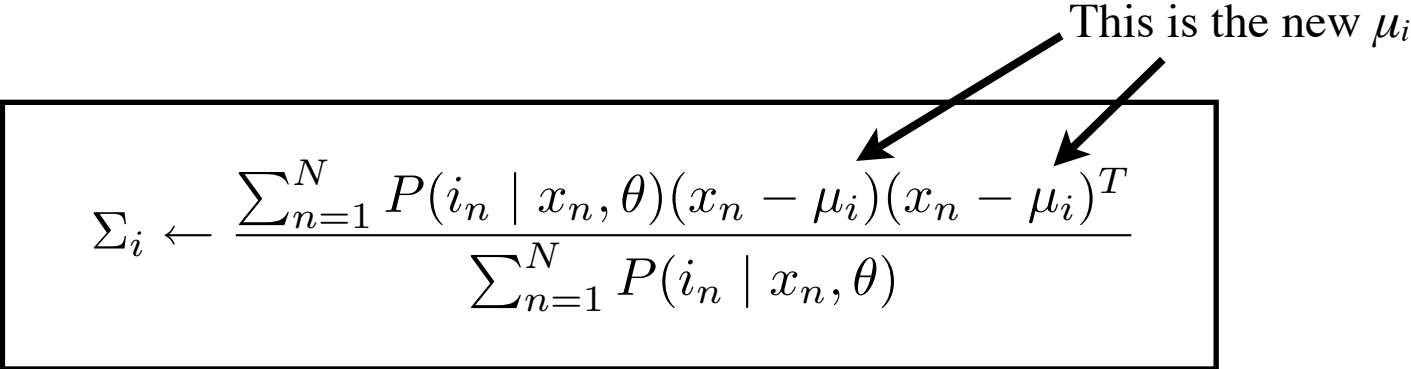
$$\mu_i \leftarrow \frac{\sum_{n=1}^N P(i_n | x_n, \theta) x_n}{\sum_{n=1}^N P(i_n | x_n, \theta)}$$

M Step: Mixture of Gaussians (cont.)

- Finally, let's consider updating the covariance matrices Σ_i :

$$\frac{\partial}{\partial \Sigma'_i} \left[\sum_n \sum_i P(i_n | x_n, \theta) \ln \mathcal{N}(x_n | \mu'_i, \Sigma'_i) \right] = 0$$

$$\frac{\partial}{\partial \Sigma'_i} \left[\sum_n \sum_i P(i_n | x_n, \theta) \left(-\frac{1}{2} \ln(|\Sigma'_i|) - \frac{1}{2} (x_n - \mu_i)^T \Sigma_i'^{-1} (x_n - \mu_i) \right) \right] = 0$$



This is the new μ_i

$$\Sigma_i \leftarrow \frac{\sum_{n=1}^N P(i_n | x_n, \theta) (x_n - \mu_i)(x_n - \mu_i)^T}{\sum_{n=1}^N P(i_n | x_n, \theta)}$$

EM Gaussian Mixture: Summary

- Given observed X_1 to X_N and hidden variables I_1 to I_N (mixture component) iterate E and M steps until convergence.
- E step: for each data point n compute

$$P(i_n | x_n, \theta) = \frac{w_i \mathcal{N}(x_n | \mu_i, \Sigma_i)}{\sum_{j=1}^K w_j \mathcal{N}(x_n | \mu_j, \Sigma_j)}$$

- M step: Update the parameters of component i (from 1 to K) with

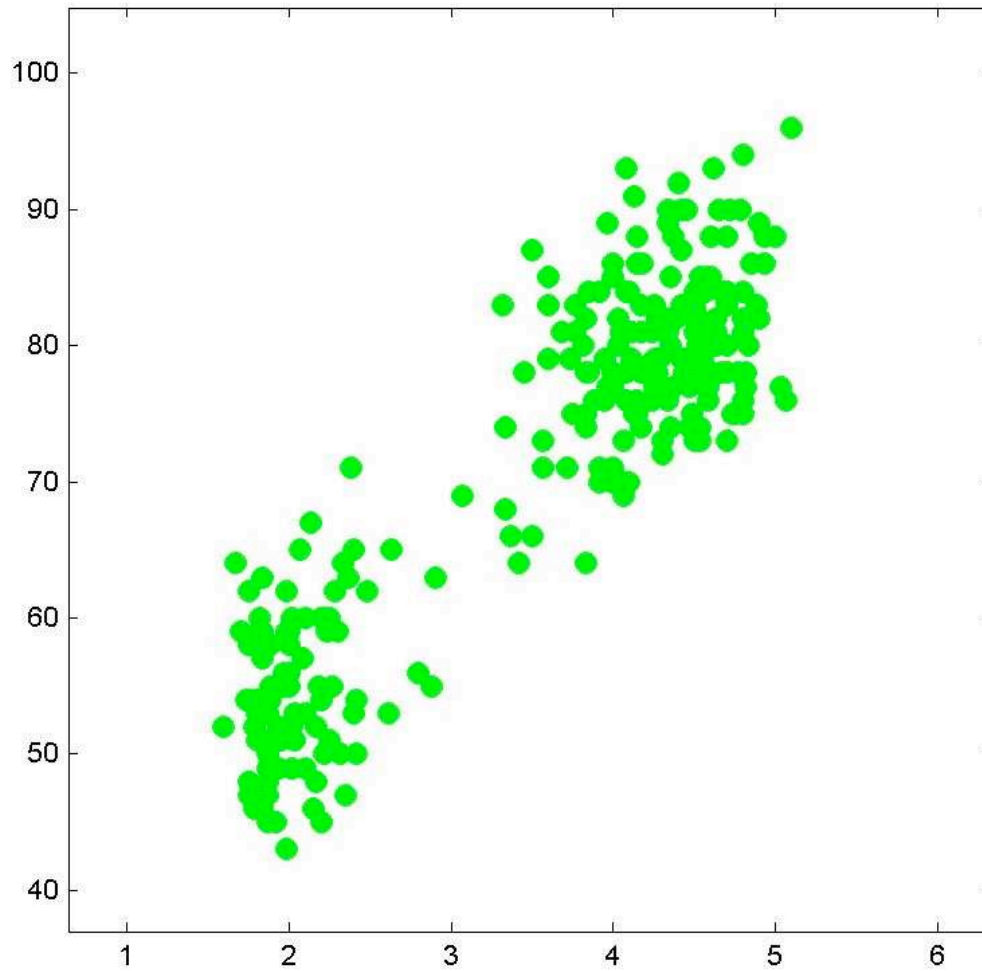
$$w_i \leftarrow \frac{1}{N} \sum_{n=1}^N P(i_n | x_n, \theta)$$

$$\mu_i \leftarrow \frac{\sum_{n=1}^N P(i_n | x_n, \theta) x_n}{\sum_{n=1}^N P(i_n | x_n, \theta)}$$

$$\Sigma_i \leftarrow \frac{\sum_{n=1}^N P(i_n | x_n, \theta) (x_n - \mu_i)(x_n - \mu_i)^T}{\sum_{n=1}^N P(i_n | x_n, \theta)}$$

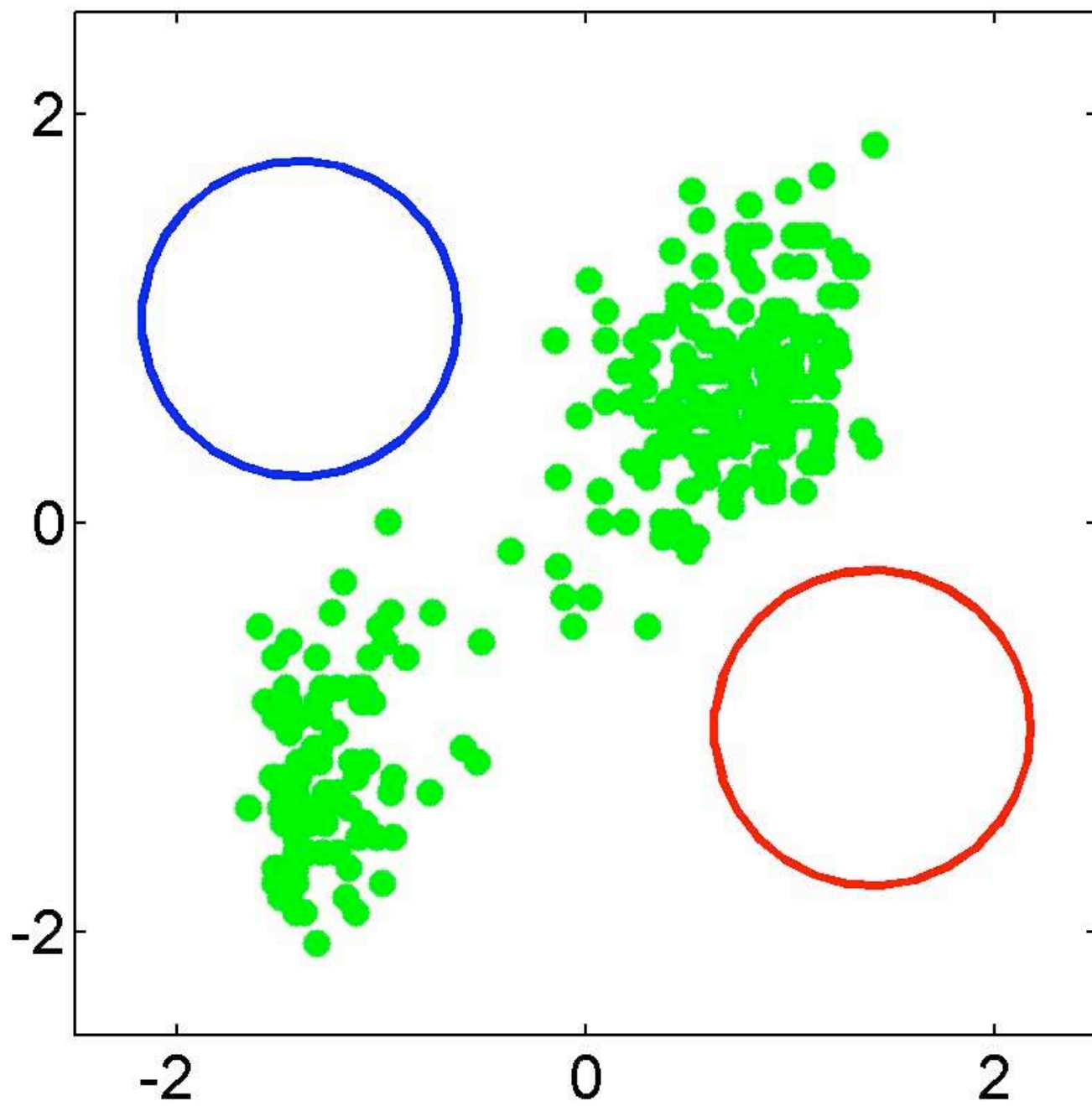
EM I: GMM model of Old Faithful data

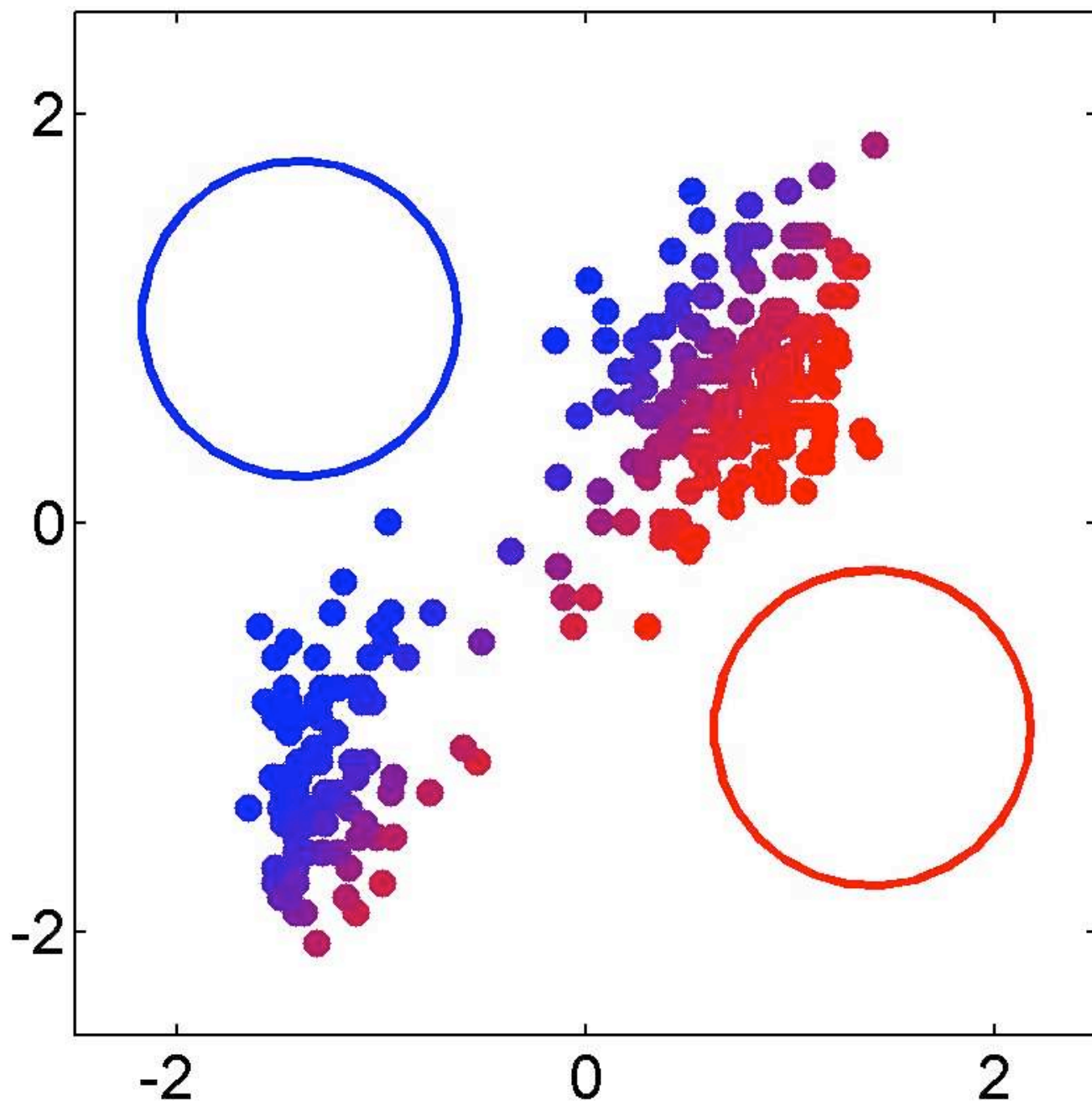
Time
between
eruptions
(minutes)

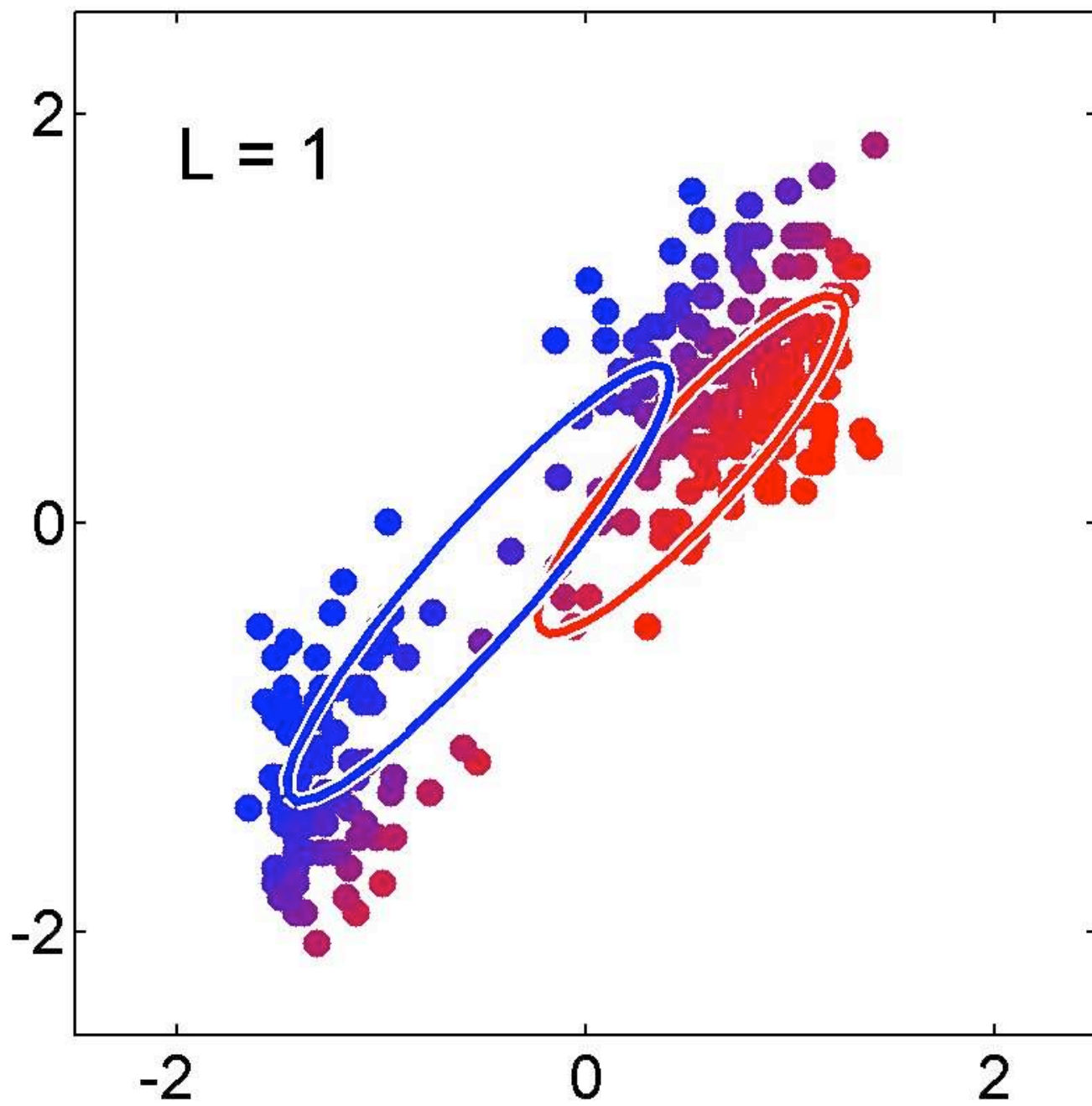


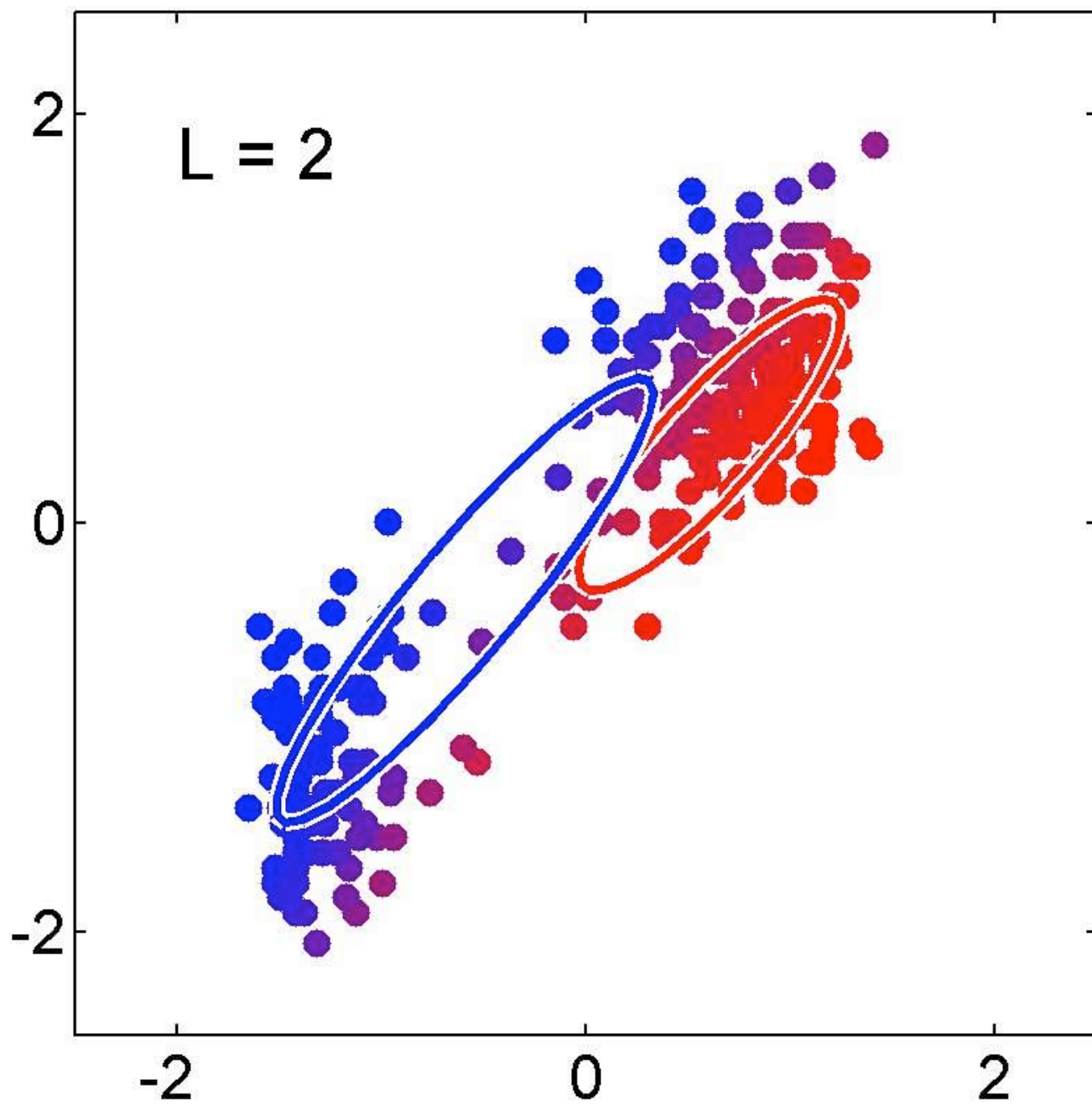
Duration of eruption (minutes)

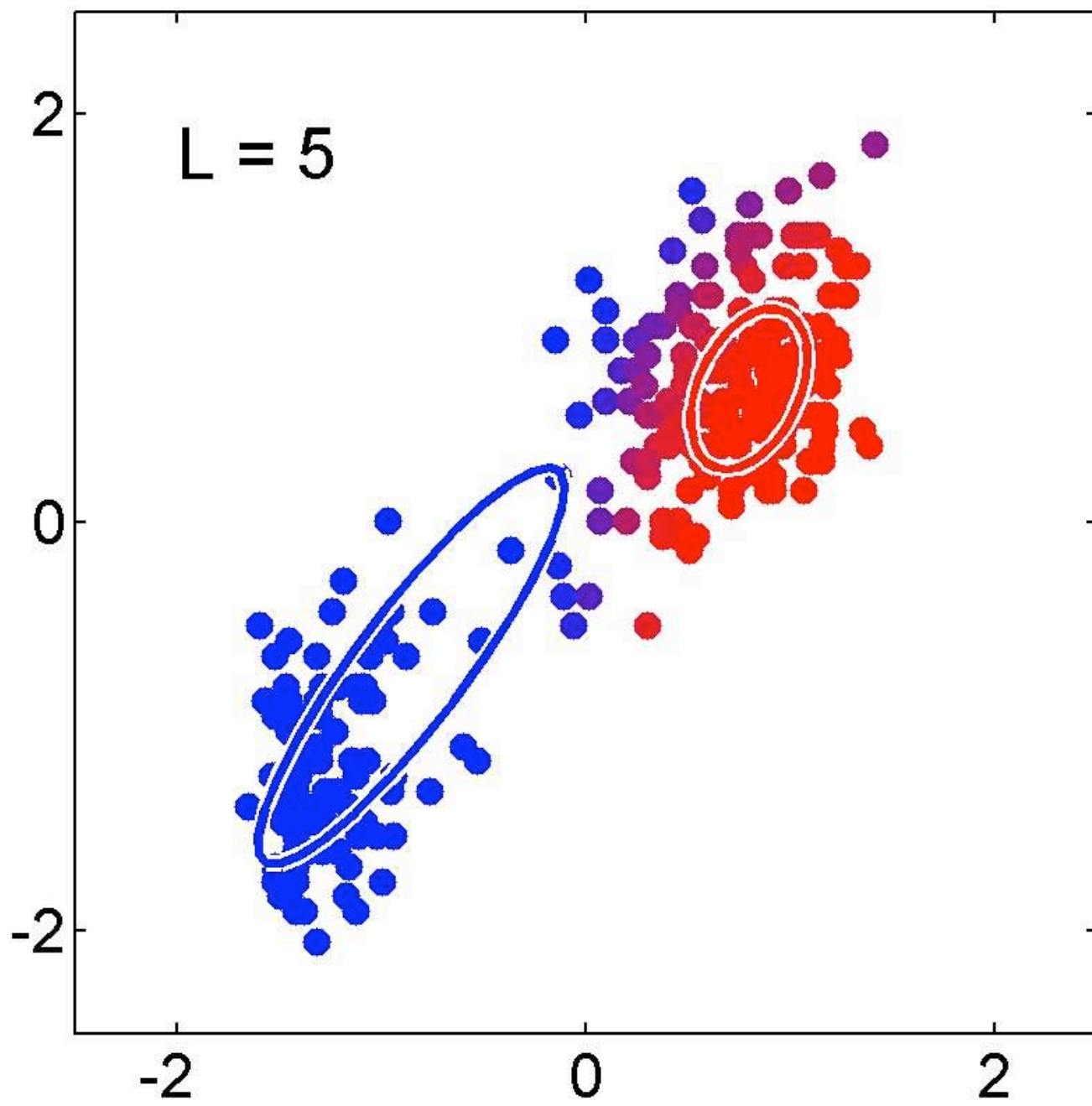


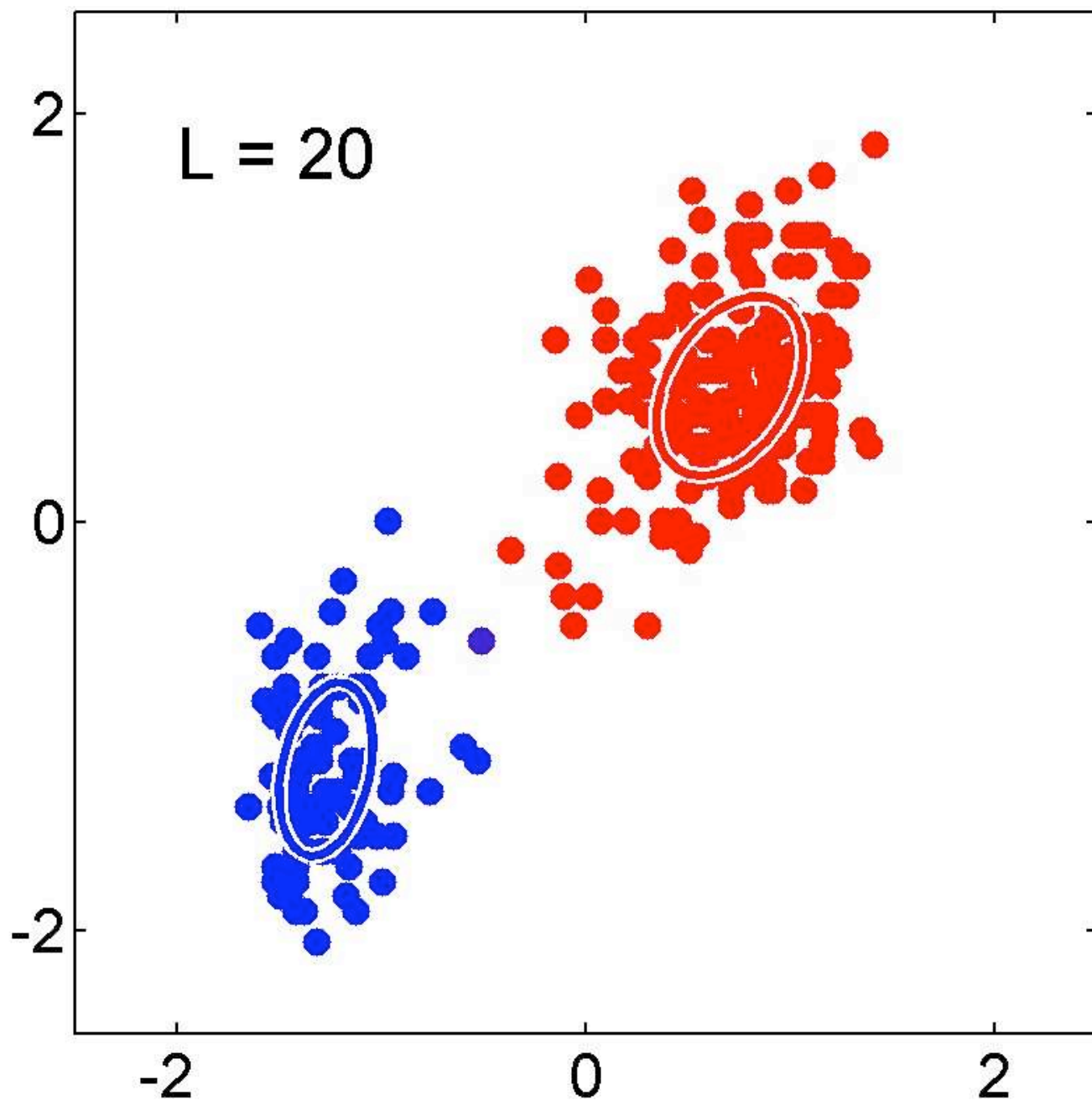








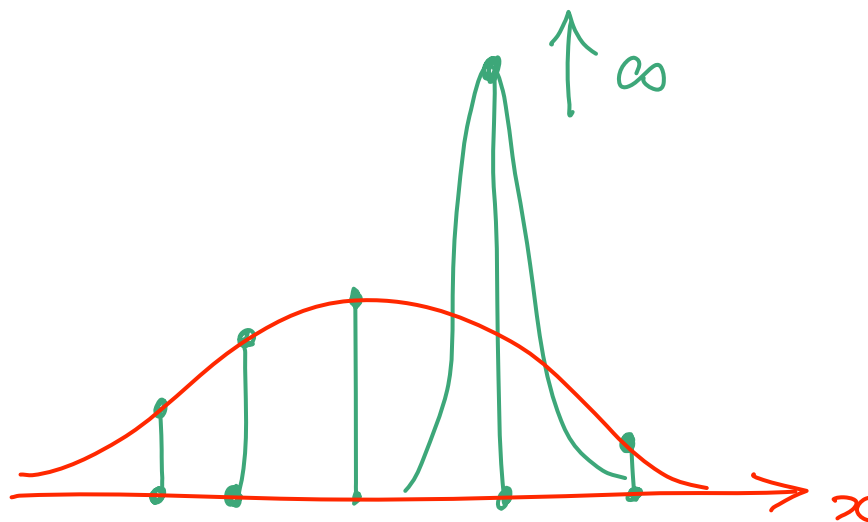




Over-fitting with Gaussian Mixtures

- Singularities (infinities) occur in the likelihood function when components “collapse” onto a data point

$$\mathcal{N}(x_n | x_n, \sigma^2 \mathbf{I}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{\sigma} \rightarrow \infty \quad \text{as } \sigma \rightarrow 0$$



- Also, maximum likelihood cannot determine the number of mixture components (the likelihood always increases with more components).

EM II: Clustering Documents with Naïve Bayes

- Consider you have a collection of D unlabeled documents.
- Build an initial naïve Bayes classifier with parameters θ . Use EM to find the maximum likelihood estimation of the parameters.
- Naïve Bayes assumption for document clustering:
 - The probability of a document d_i given class c_j is the product of the probabilities of the words $w_{d_i,k}$ in the document given that class:

$$P(d_i | c_j, \theta) = \prod_k P(w_{d_i,k} | c_j, \theta)$$

- The model parameters are the probabilities of the words w_t given the class c_j : $\theta_{w_t | c_j}$ (consider this as a class specific vocabulary) and the marginal probabilities of the class c_j : θ_{c_j}
- Repeat until convergence:
 - E step: Use the current classifier (θ) to estimate component membership of each unlabeled document, i.e. the probability that each class generated each document $P(c_j | d_i, \theta)$.
 - M step: Re-estimate the classifier (θ) given the estimated component membership of each document.

EM II: Clustering Documents with Naïve Bayes

- E step:

$$\begin{aligned} P(y_i = c_j \mid d_i, \theta) &= \frac{P(c_j \mid \theta) P(d_i \mid c_j, \theta)}{P(d_i \mid \theta)} \\ &= \frac{P(c_j \mid \theta) \prod_{k=1}^{|d_i|} P(w_{d_i, k} \mid c_j, \theta)}{\sum_{r=1}^{\mathcal{C}} P(c_r \mid \theta) \prod_{k=1}^{|d_i|} P(w_{d_i, k} \mid c_r, \theta)} \end{aligned}$$

- M step:

$$\begin{aligned} \theta_{w_t | c_j} &\leftarrow P(w_t \mid c_j, \theta) = \frac{\sum_{i=1}^{|\mathcal{D}|} \text{Num}(w_t, d_i) P(y_i = c_j \mid d_i)}{\sum_{s=1}^{|\mathcal{V}|} \sum_{i=1}^{|\mathcal{D}|} \text{Num}(w_s, d_i) P(y_i = c_j \mid d_i)} \\ \theta_{c_j} &\leftarrow P(c_j \mid \theta) = \frac{\sum_{i=1}^{|\mathcal{D}|} P(y_i = c_j \mid d_i)}{|\mathcal{D}|} \end{aligned}$$

where $|\mathcal{D}|$ is the number of documents, \mathcal{C} is the number of classes, $|d_i|$ is the number of words in document d_i and w_t is the t -th word in the vocabulary of size $|\mathcal{V}|$.

EM: optimizing a lower bound

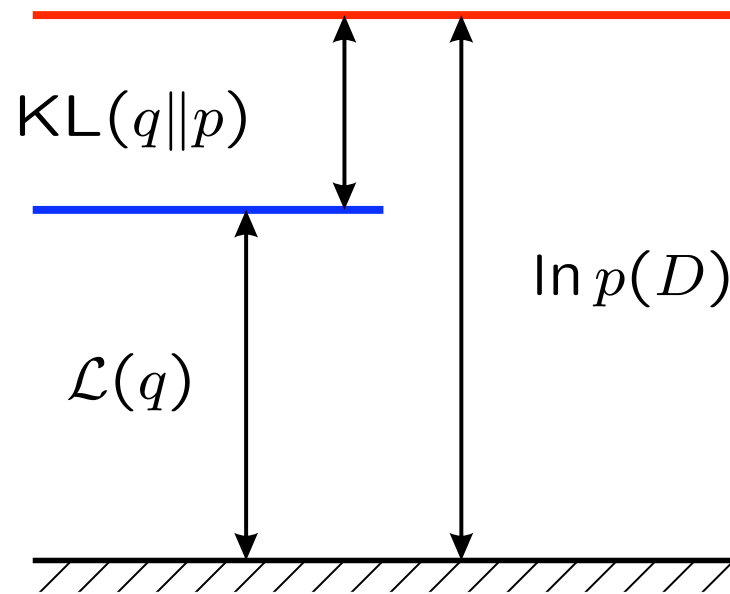
- Recall: our original goal is to maximize the likelihood $p(X | \theta)$.
- Suppose that direct optimization of $p(X | \theta)$ is difficult, but that optimizing the complete-data likelihood function $p(X, Z | \theta)$ is significantly easier.
- Introduce a distribution $q(Z)$ over the latents, for any choice of $q(Z)$:

$$\ln p(X | \theta) = \mathcal{L}(q, \theta) + \text{KL}(q||p)$$

where

$$\mathcal{L}(q, \theta) = \sum_Z q(Z) \ln \left\{ \frac{p(X, Z | \theta)}{q(Z)} \right\}$$

$$\text{KL}(q||p) = - \sum_Z q(Z) \ln \left\{ \frac{p(Z | X, \theta)}{q(Z)} \right\}$$



EM: optimizing a lower bound (cont.)

- Maximizing $\mathcal{L}(q, \theta)$ with respect to a free-form q distribution, we obtain the true posterior distribution:

$$q(Z) = p(Z | X, \theta)$$

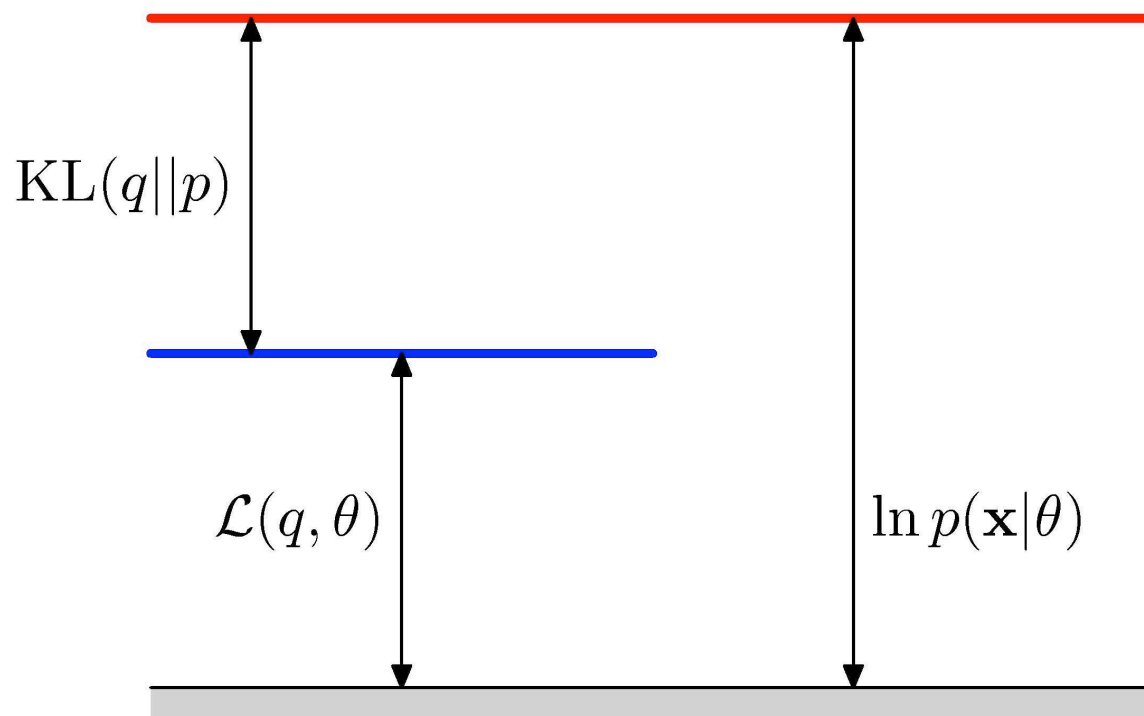
- The lower bound $\mathcal{L}(q, \theta)$ then becomes

$$\begin{aligned}\mathcal{L}(q, \theta) &= \sum_Z p(Z | X, \theta^{old}) \ln \left\{ \frac{p(X, Z | \theta)}{p(Z | X, \theta^{old})} \right\} \\ &= Q(\theta, \theta^{old}) + \text{const}\end{aligned}$$

which, as a function of θ is the expected complete-data log likelihood (up to an additive constant).

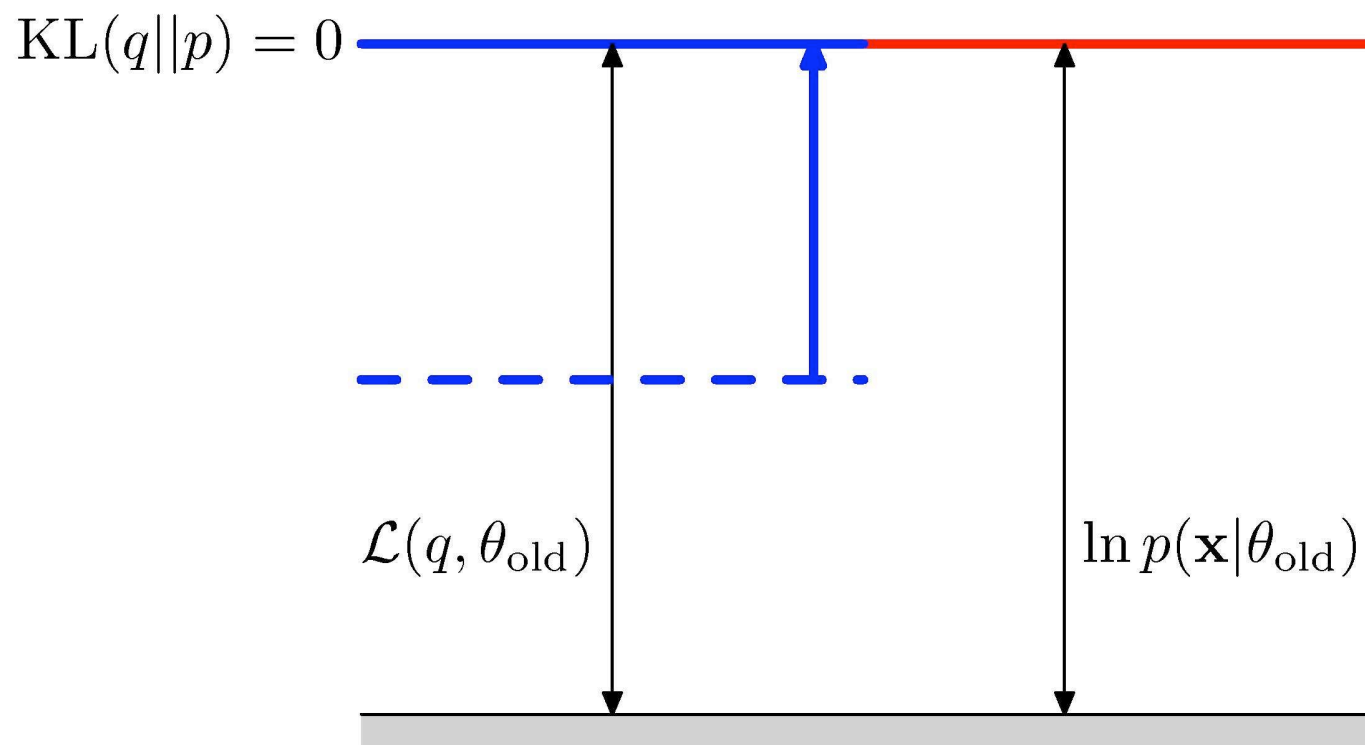
EM: optimizing a lower bound (cont.)

Initial Configuration:



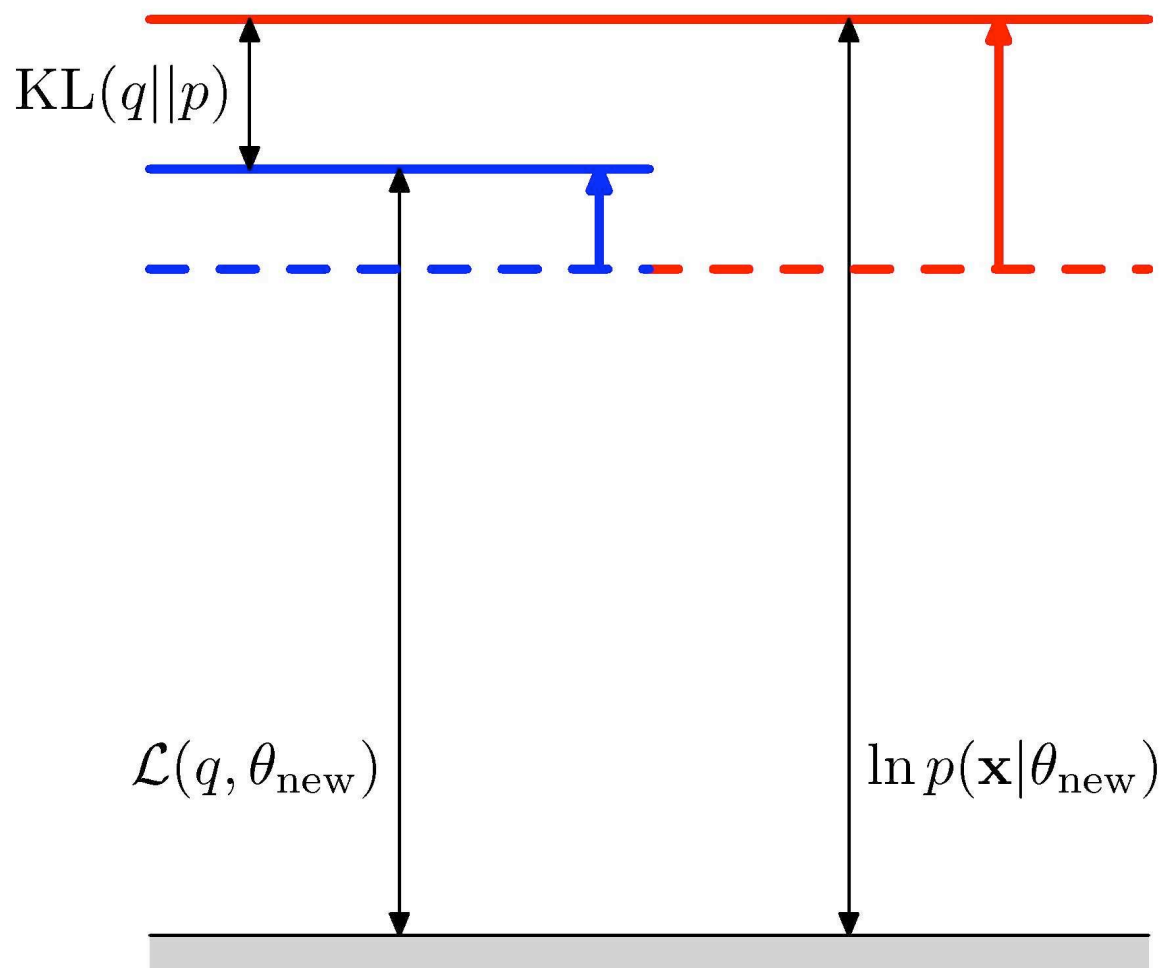
EM: optimizing a lower bound (cont.)

E-step:



EM: optimizing a lower bound (cont.)

M-step:



Acknowledgments

- The material presented here is taken from tutorials, notes and lecture slides from Yoshua Bengio, Christopher Bishop, Andrew Moore, Tom Mitchell and Scott Davies.

Note to other teachers and users of these slides. Andrew and Scott would be delighted if you found this source material useful in giving your own lectures. Feel free to use these slides verbatim, or to modify them to fit your own needs. PowerPoint originals are available. If you make use of a significant portion of these slides in your own lecture, please include this message, or the following link to the source repository of Andrew's tutorials: <http://www.cs.cmu.edu/~awm/tutorials>. Comments and corrections gratefully received.