The Expectation-Maximization (EM) Algorithm

• Reading Assignments

T. Mitchell, *Machine Learning*, McGraw-Hill, 1997 (section 6.12, *hard copy*).

S. Gong et al. *Dynamic Vision: From Images to Face Recognition*, Imperial College Pres, 2001 (Appendix C, hard copy).

A. Webb, *Statistical Pattern Recognition*, Arnold, 1999 (section 2.3, *hard copy*).

• Case Studies

B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696-710, 1997 (*on-line*).

S. McKenna, Y. Raja, and S. Gong, "Tracking color objects using adaptive mixture models", *Image and Vision Computing*, vol. 17, pp. 225-231, 1999 (*on-line*).

C. Stauffer and E. Grimson, "Adaptive background mixture models for real-time tracking", *IEEE Computer Vision and Pattern Recognition Conference*, Vol. 2, pp. 246-252, 1998 (*on-line*).

The Expectation-Maximization (EM) Algorithm

• Overview

- It is an iterative algorithm that starts with an initial estimate for θ and iteratively modifies θ to increase the likelihood of the observed data.

- Works best in situations where the data is incomplete or *can be thought of as being incomplete*.

- EM is typically used with mixture models (e.g., mixtures of Gaussians).

• The case of incomplete data

- Many times, it is impossible to apply ML estimation because we can not measure all the features or certain feature values are missing.

- The EM algorithm is ideal (i.e., it produces ML estimates) for problems with unobserved (missing) data.

$$\text{Actual data: } \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad \text{Observed data: } \mathbf{y} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\text{Complete pdf: } p(\mathbf{x}/\theta), \quad \text{Incomplete pdf: } p(\mathbf{y}/\theta)$$

- Incomplete pdf can be derived from complete pdf:

$$p(\mathbf{y}/\theta) = \int \cdots \int p(\mathbf{x}/\theta) d\mathbf{x}_{missing}$$

• **An example**

- Assume the following two classes in a pattern-recognition problem:

(1) A class of dark object
    (1.1) Round black objects
    (1.2) Square black objects

(2) A class of light objects

Complete data and pdf:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \begin{matrix} \textit{number of round dark objects} \\ \textit{number of square dark objects} \\ \textit{number of light objects} \end{matrix}$$

$$p(x_1, x_2, x_3/\theta) = (\frac{n!}{x_1!\ x_2!\, x_3!})(1/4)^{x_1}(1/4 + \theta/4)^{x_2}(1/2 - \theta/4)^{x_3}$$

Observed (incomplete) data and pdf:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 + x_2 \\ x_3 \end{bmatrix} \begin{matrix} \textit{number of dark objects} \\ \textit{number of light objects} \end{matrix}$$

(many-to-one mapping !!)

• EM: main idea and steps

- If x was available, then we could use ML to estimate θ, i.e.,

$$arg\ \max_\theta \ln\ p(D_x/\theta)$$

Idea: maximize the expectation of $p(\mathbf{x}/\theta)$ given the data $\mathbf{y}$ and our current estimate of θ.

1. Initialization step: initialize the algorithm with a guess θ^0

2. Expectation step: it is with respect to the unknown variables, using the current estimate of parameters and conditioned upon the observations.

$$Q(\theta; \theta^t) = E_{x_{unobserved}}(ln\ \ p(D_x/\theta)\ /\ D_y, \theta^t)$$

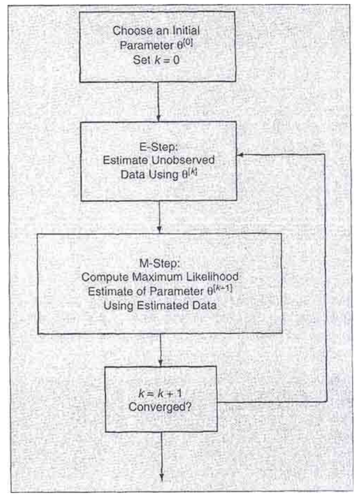
* Expectation is over the values of the unobserved variables since the observed data is fixed.

* When ln $p(D_x/\theta)$ is a linear function of the unobserved variables, then the above step is equivalent to finding $E(x_{unobserved}/D_y, \theta^t)$

3. Maximization step: provides a new estimate of the parameters.

$$\theta^{t+1} = arg\ \max_\theta Q(\theta; \theta^t)$$

4. Convergence step: if $\|\theta^{t+1} - \theta^t\| < \varepsilon$, stop; otherwise, go to step 2.



- **An example (cont'd))**

- Assume the following two classes in a pattern-recognition problem:

(1) A class of dark object
(1.1) Round black objects
(1.2) Square black objects

(2) A class of light objects

Complete data and pdf:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \begin{matrix} \textit{number of round dark objects} \\ \textit{number of square dark objects} \\ \textit{number of light objects} \end{matrix}$$

$$p(x_1, x_2, x_3/\theta) = (\frac{n!}{x_1!\ x_2!\, x_3!})(1/4)^{x_1}(1/4 + \theta/4)^{x_2}(1/2 - \theta/4)^{x_3}$$

Observed (incomplete) data and pdf:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 + x_2 \\ x_3 \end{bmatrix} \begin{matrix} \textit{number of dark objects} \\ \textit{number of light objects} \end{matrix}$$

(many-to-one mapping !!)

Expectation step: compute $E(ln\ p(D_x/\theta)\ /\ D_y, \theta^t))$

$$p(D_x/\theta) = \prod_{i=1}^{n} p(\mathbf{x}_i/\theta) \quad ==> \quad \ln\ p(D_x/\theta) = \sum_{i=1}^{n} \ln\ p(\mathbf{x}_i/\theta) =$$

$$\sum_{i=1}^{n} ln(\frac{n!}{x_{i1}!\ x_{i2}!x_{i3}!}) + x_{i1}\ ln(1/4) + x_{i2}\ ln(1/4+\theta/4) + x_{i3}\ ln(1/2-\theta/4)$$

$$E[ln\ p(D_x/\theta)/D_y,\theta^t] = \sum_{i=1}^{n} E[ln(\frac{n!}{x_{i1}!\ x_{i2}!x_{i3}!})/D_y,\theta^t] + E[x_{i1}/D_y,\theta^t]\ ln(1/4) +$$

$$E[x_{i2}/D_y,\theta^t]\ ln(1/4+\theta/4) + x_{i3}ln(1/2-\theta/4)$$

<u>Maximization step:</u> compute θ^{t+1} by maximizing $E(ln\ p(D_x/\theta)\ /\ D_y,\theta^t)$

$$\frac{d}{d\theta}\ E[ln\ p(D_x/\theta)/D_y,\theta^t] = 0 ==> \theta^{t+1} = \frac{2+E[x_{i2}/D_y,\theta^t]-x_{i3}}{E[x_{i2}/D_y,\theta^t]+x_{i3}}$$

<u>Expectation step (cont'd):</u> estimating $E[x_{i2}/D_y,\theta^t]$

$$P(x_{i2}/y_{i1},y_{i2}) = P(x_{i2}/y_{i1}) = \binom{y_{i1}}{x_{i2}}(1/4)^{x_{i2}}(1/4+\theta/4)^{y_{i1}-x_{i2}}\ \frac{1}{(1/2+\theta/4)^{y^{i1}}}$$

$$E[x_{i2}/D_y,\theta^t] = y_{i1}\ \frac{1/4}{1/2+\theta^t/4}$$

Table 1. Results of the EM algorithm for an example using trinomial data

k	$x_1^{[k]}$	$x_2^{[k]}$	$p^{[k]}$
1	31.500000	31.500000	0.379562
2	26.475460	36.524540	0.490300
3	25.298157	37.701843	0.514093
4	25.058740	37.941260	0.518840
5	25.011514	37.988486	0.519773
6	25.002255	37.997745	0.519956
7	25.000441	37.999559	0.519991
8	25.000086	37.999914	0.519998
9	25.000017	37.999983	0.520000
10	25.000003	37.999997	0.520000

• Convergence properties of the EM algorithm

- At each iteration, a value of θ is computed so that the likelihood function does not decrease.

- It can be shown that by increasing $Q(\theta; \theta^t) = E_{x_{unobserved}}(ln\ p(D_x/\theta)\ /\ D_y, \theta^t)$ with the EM algorithm, we are also increasing ln $p(D_x/\theta)$.

- This does not guarantee that the algorithm will reach the ML estimate (*global* maximum) and, in practice, it may get stuck in a local optimum.

- The solution depends on the initial estimate θ^0.

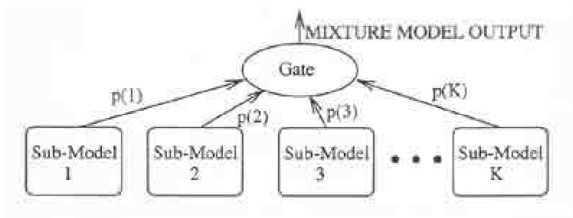
- The algorithm is guaranteed to be stable and to converge to a ML estimate (i.e., there is no chance of "overshooting" or diverging from the maximum).

Maximum Likelihood of mixtures via EM

• Mixture model

- In a mixture model, there are many "sub-models", each of which has its own probability distribution which describes how it generates data when it is active.

- There is also a "mixer" or "gate" which controls how often each sub-model is active.



- Formally, a mixture is defi ned as a weighted sum of K components where each component is a parametric density function $p(x/\theta_k)$:

$$p(x/\theta) = \sum_{k=1}^{K} p(x/\theta_k)\pi_k$$

• Mixture parameters

- The parameters θ to estimate are:

  * the values of π_k
  * the parameters θ_k of $p(x/\theta_k)$

- The component densities $p(x/\theta_k)$ may be of different parametric forms and are specifi ed using knowledge of the data generation process, if available.

- The weights π_k are the *mixing parameters* and they sum to unity:

$$\sum_{k=1}^{K} \pi_k = 1$$

- Fitting a mixture model to a set of observations D_x consists of estimating the set of mixture parameters that best describe this data.

- Two fundamental issues arise in mixture fitting:

  (1) Estimation of the mixture parameters.

  (2) Estimation of the mixture components.

• Mixtures of Gaussians

- In the mixtures of Gaussian model, $p(x/\theta_k)$ is the multivariate Gaussian distribution.

- In this case, the parameters θ_k are (μ_k, Σ_k).

• Mixture parameter estimation using ML

- As we have seen, given a set of data $D=(x_1, x_2, ..., x_n)$, ML seeks the value of θ that maximizes the following probability:

$$p(D/\theta) = \prod_{i=1}^{n} p(x_i/\theta)$$

- Since $p(x_i/\theta)$ is modeled as a mixture (i.e., $p(x_i/\theta) = \sum_{k=1}^{K} p(x_i/\theta_k)\pi_k$) the above expression can be written as:

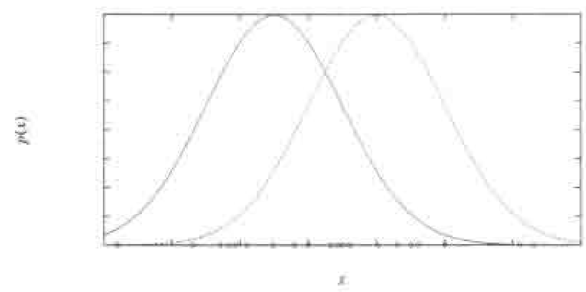
$$p(D/\theta) = \prod_{i=1}^{n} \sum_{k=1}^{K} p(x_i/\theta_k)\pi_k$$

- In general, it is not possible to solve $\frac{\partial p(D/\theta)}{\partial \theta} = 0$ explicitly for the parameters and iterative schemes must be employed.

Estimate the means of K Gaussians using EM (special case)

• Data generation process using mixtures

- Assume the data D is generated by a probability distribution that is a mixture of k Gaussians.



$k = 2$

$k = 2$

- Each instance is generated using a two-step process:

  (1) One of the K Gaussians is selected at random, with probabilities $\pi_1, \pi_2, \ldots, \pi_K$.

  (2) A single random instance x_i is generated according to this selected distribution.

- This process is repeated to generate a set of data points D.

• Assumptions (this example)

(1) $\pi_1 = \pi_2 = \cdots = \pi_K$ (uniform distribution)

(2) Each Gaussian has the same variance σ^2 which is known.

- The problem is to estimate the means of the Gaussians $\theta = (\mu_1, \mu_2, \ldots, \mu_K)$

Note: if we knew which Gaussian generated each datapoint, then it would be easy to find the parameters for each Gaussian using ML.

- **Involving hidden or unobserved variables**

- We can think of the full description of each instance x_i as

$$y_i=(x_i, z_i)=(x_i, z_{i1}, z_{i2}, \ldots, z_{iK})$$

where z_i is a class indicator vector (hidden variable):

$$z_{ij} = \begin{cases} 1 & \text{if } x_i \ was \ \ generated \ by \ j-th \ component \\ 0 & otherwise \end{cases}$$

- In this case, x_i are observable and z_i non-observable.

- **Main steps using EM**

- The EM algorithm searches for a ML hypothesis through the following iterative scheme:

(1) Initialize the hypothesis $\theta^0=(\mu_1^0, \mu_2^0, \ldots, \mu_K^0)$

(2) Estimate the expected values of the hidden variables z_{ij} using the current hypothesis $\theta^t=(\mu_1^t, \mu_2^t, \ldots, \mu_K^t)$

(3) Update the hypothesis $\theta^{t+1}=(\mu_1^{t+1}, \mu_2^{t+1}, \ldots, \mu_K^{t+1})$ using the expected values of the hidden variables from step 2.

- Repeat steps (2)-(3) until convergence.

- **Derivation of the Expectation-step**

- We must derive an expression for $Q(\theta; \theta^t) = E_{z_i}(ln\ p(D_y/\theta) \ /\ D_x, \theta^t)$

(1) Derive the form of ln $p(D_y/\theta)$:

$$p(D_y/\theta) = \prod_{i=1}^{n} p(y_i/\theta)$$

- We can write $p(y_i/\theta)$ as follows:

$$p(y_i/\theta) = p(x_i, z_i/\theta) = p(x_i/z_i, \theta)p(z_i/\theta) = p(x_i/\theta_j)\pi_j$$

(assuming z_{ij}=1 and z_{ik}=0 for $k \neq j$)

- We can rewrite $p(x_i/\theta_j)\pi_j$ as follows:

$$p(y_i/\theta) = \prod_{k=1}^{K} [p(x_i/\theta_k)\pi_k]^{z_{ik}}$$

- Thus, $p(D_y/\theta)$ can be written as follows (π_k's are all equal):

$$p(D_y/\theta) = \prod_{i=1}^{n}\prod_{k=1}^{K} [p(x_i/\theta_k)]^{z_{ik}}$$

- We have assumed the form of $p(x_i/\theta_k)$ to be Gaussian:

$$p(x_i/\theta_k) = \frac{1}{\sigma\sqrt{2\pi}} exp[-\frac{(x_i-\mu_k)^2}{2\sigma^2}], \quad \text{thus}$$

$$\prod_{k=1}^{K} [p(x_i/\theta_k)]^{z_{ik}} = \frac{1}{\sigma\sqrt{2\pi}} exp[-\frac{1}{2\sigma^2}\sum_{k=1}^{K} z_{ik}(x_i-\mu_k)^2]$$

which leads to the following form for $p(D_y/\theta)$:

$$p(D_y/\theta) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} exp[-\frac{1}{2\sigma^2}\sum_{k=1}^{K} z_{ik}(x_i-\mu_k)^2]$$

- Let's compute now ln $p(D_y/\theta)$:

$$\ln\ p(D_y/\theta) = \sum_{i=1}^{n}(ln\ \frac{1}{\sigma\sqrt{2\pi}} - \frac{1}{2\sigma^2}\sum_{k=1}^{K} z_{ik}(x_i-\mu_k)^2)$$

(2) Take the expected value of ln $p(D_y/\theta)$:

$$E_{z_i}(ln\ p(D_y/\theta)/D_x, \theta^t) = E(\sum_{i=1}^{n}(ln\ \frac{1}{\sigma\sqrt{2\pi}} - \frac{1}{2\sigma^2}\sum_{k=1}^{K} z_{ik}(x_i - \mu_k^t)^2))) =$$

$$\sum_{i=1}^{n}(ln\ \frac{1}{\sigma\sqrt{2\pi}} - \frac{1}{2\sigma^2}\sum_{k=1}^{K} E(z_{ik})(x_i - \mu_k^t)^2)$$

- $E(z_{ik})$ is just the probability that the instance x_i was generated by the k-th component (i.e., $E(z_{ik}) = \sum_j z_{ij} P(z_{ij}) = P(z_{ik}) = P(k/x_i)$):

$$E(z_{ik}) = \frac{exp[-\frac{(x_i - \mu_k^t)^2}{2\sigma^2}]}{\sum_{j=1}^{K} exp[-\frac{(x_i - \mu_j^t)^2}{2\sigma^2}]}$$

- **Derivation of the Maximization-step**

- Maximize $Q(\theta; \theta^t) = E_{z_i}(ln\ p(D_y/\theta)\ /\ D_x, \theta^t)$

$$\frac{\partial Q}{\partial \mu_k} = 0 \quad \text{or} \quad \mu_k^{t+1} = \frac{\sum_{i=1}^{n} E(z_{ik}) x_i}{\sum_{i=1}^{n} E(z_{ik})}$$

- **Summary of the two steps**

- Choose the number of components K

Initialization step

$$\theta_k^0 = \mu_k^0$$

Expectation step

$$E(z_{ik}) = \frac{exp[-\frac{(x_i - \mu_k^t)^2}{2\sigma^2}]}{\sum_{j=1}^{K} exp[-\frac{(x_i - \mu_j^t)^2}{2\sigma^2}]}$$

Maximization step

$$\mu_k^{t+1} = \frac{\sum_{i=1}^{n} E(z_{ik}) x_i}{\sum_{i=1}^{n} E(z_{ik})}$$

Estimate the mixture parameters (general case)

- If we knew which sub-model was responsible for generating each datapoint, then it would be easy to find the ML parameters for each sub-model.

(1) Use EM to estimate which sub-model was responsible for generating each datapoint.

(2) Find the ML parameters based on these estimates.

(3) Use the new ML parameters to re-estimate the responsibilities and iterate.

• Involving hidden variables

- We do not know which instance x_i was generated by which component (i.e., the missing data are the labels showing which sub-model generated each datapoint).

- Augment each instance x_i by the missing information:

$$y_i = (x_i, z_i)$$

where z_i is a class indicator vector $z_i = (z_{1i}, z_{2i}, \ldots, z_{Ki})$:

$$z_{ij} = \begin{cases} 1 & \text{if } x_i \textit{ generated by } j-th \textit{ component} \\ 0 & \textit{otherwise} \end{cases}$$

(x_i are observable and z_i non-observable)

- **Derivation of the Expectation step**

  - We must derive an expression for $Q(\theta;\theta^t) = E_{z_i}(ln\ p(D_y/\theta)\ /\ D_x, \theta^t)$

  <u>(1) Derive the form of ln $p(D_y/\theta)$:</u>

$$p(D_y/\theta) = \prod_{i=1}^{n} p(y_i/\theta)$$

  - We can write $p(y_i/\theta)$ as follows:

$$p(y_i/\theta) = p(x_i, z_i/\theta) = p(x_i/z_i, \theta)p(z_i/\theta) = p(x_i/\theta_j)\pi_j$$

(assuming z_{ij}=1 and z_{ik}=0 for $k \neq j$)

  - We can rewrite the above expression as follows:

$$p(y_i/\theta) = \prod_{k=1}^{K}[p(x_i/\theta_k)\pi_k]^{z_{ik}}$$

  - Thus, $p(D_y/\theta)$ can be written as follows:

$$p(D_y/\theta) = \prod_{i=1}^{n}\prod_{k=1}^{K}[p(x_i/\theta_k)\pi_k]^{z_{ik}}$$

  - We can now compute ln $p(D_y/\theta)$

$$\ln\ p(D_y/\theta) = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik} \ln\left(p(x_i/\theta_k)\pi_k\right) =$$

$$\sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik} \ln\left(p(x_i/\theta_k)\right) + \sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik} \ln\left(\pi_k\right)$$

(2) Take the expected value of ln $p(D_y/\theta)$:

$$E(ln\ p(D_y/\theta)/D_x, \theta^t) = \sum_{i=1}^{n}\sum_{k=1}^{K} E(z_{ik}) ln\ (p(x_i/\theta_k^t)) + \sum_{i=1}^{n}\sum_{k=1}^{K} E(z_{ik}) ln\ (\pi_k^t)$$

- $E(z_{ik})$ is just the probability that instance x_i was generated by the k-th component (i.e., $E(z_{ik}) = \sum_j z_{ij} P(z_{ij}) = P(z_{ik}) = P(k/x_i)$):

$$E(z_{ik}) = \frac{p(x_i/\theta_k^t)\pi_k^t}{\sum_{j=1}^{K} p(x_i/\theta_j^t)\pi_j^t}$$

- **Derivation of the Maximization step**

- Maximize $Q(\theta; \theta^t)$ subject to the constraint $\sum_{k=1}^{K} \pi_k = 1$:

$$Q'(\theta; \theta^t) = \sum_{i=1}^{n}\sum_{k=1}^{K} E(z_{ik}) ln\ (p(x_i/\theta_k)) + \sum_{i=1}^{n}\sum_{k=1}^{K} E(z_{ik}) ln\ (\pi_k) + \lambda(1 - \sum_{k=1}^{K} \pi_k)$$

where λ is the Langrange multiplier.

$$\frac{\partial Q'}{\partial \pi_k} = 0 \quad \text{or} \quad \sum_{i=1}^{n} E(z_{ik}) \frac{1}{\pi_k} - \lambda = 0 \quad \text{or} \quad \pi_k^{t+1} = \frac{1}{n} \sum_{i=1}^{n} E(z_{ik})$$

(the constraint $\sum_{k=1}^{K} \pi_k = 1$ gives $\sum_{k=1}^{K}\sum_{i=1}^{n} E(z_{ik}) = \lambda$)

$$\frac{\partial Q'}{\partial \mu_k} = 0 \quad \text{or} \quad \mu_k^{t+1} = \frac{1}{n\pi_k^{t+1}} \sum_{i=1}^{n} E(z_{ik}) x_i$$

$$\frac{\partial Q'}{\partial \Sigma_k} = 0 \quad \text{or} \quad \Sigma_k^{t+1} = \frac{1}{n\pi_k^{t+1}} \sum_{i=1}^{n} E(z_{ik})(x_i - \mu_k^{t+1})(x_i - \mu_k^{t+1})^T$$

- **Summary of steps**

  - Choose the number of components K

  Initialization step

$$\theta_k^0 = (\pi_k^0,\ \mu_k^0,\ \Sigma_k^0)$$

  Expectation step

$$E(z_{ik}) = \frac{p(x_i/\theta_k^t)\pi_k^t}{\sum_{j=1}^{K} p(x_i/\theta_j^t)\pi_j^t}$$

  Maximization step

$$\pi_k^{t+1} = \frac{1}{n}\sum_{i=1}^{n} E(z_{ik})$$

$$\mu_k^{t+1} = \frac{1}{n\pi_k^{t+1}}\sum_{i=1}^{n} E(z_{ik})x_i$$

$$\Sigma_k^{t+1} = \frac{1}{n\pi_k^{t+1}}\sum_{i=1}^{n} E(z_{ik})(x_i - \mu_k^{t+1})(x_i - \mu_k^{t+1})^T$$

  (4) If $\|\theta^{t+1} - \theta^t\| < \varepsilon$, stop; otherwise, go to step 2.

• Estimating the number of components K

- Use EM to obtain a sequence of parameter estimates for a range of values K

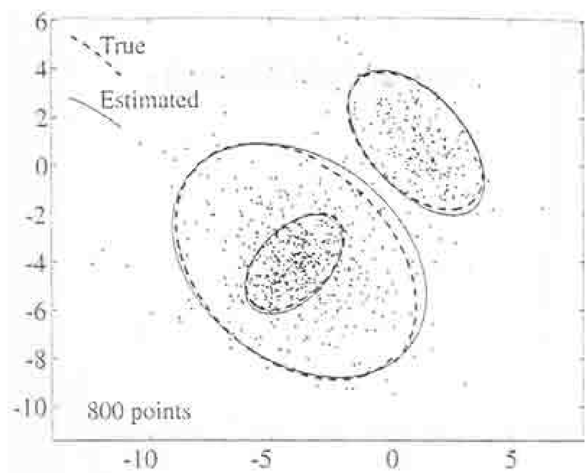
$$\{\Theta_{(K)}, K=K_{\min},...,K_{\max}\}$$

- The estimate of K is then defined as a minimizer of some cost function:

$$\hat{K} = arg\ \min_K(C(\Theta_{(K)}, K), K=K_{\min},...,K_{\max}$$

- Most often, the cost function includes ln $p(D_y/\theta)$ and an additional term whose role is to penalize large values of K.

- Several criteria have been used, e.g., Minimum description length (MDL)



Lagrange Optimization

- Suppose we want to maximize $f(x)$ subject to some constraint expressed in the form:

$$g(x) = 0$$

- To find the maximum, first we form the Lagrangian function:

$$L(x, \lambda) = f(x) + \lambda g(x)$$

(λ is called the Lagrange undetermined multiplier)

- Take the derivative and set it equal to zero:

$$\frac{\partial L(x, \lambda)}{\partial x} = \frac{\partial f(x)}{\partial x} + \lambda \frac{\partial g(x)}{\partial x} = 0$$

- Solve the resulting equation for λ and the value x that maximizes $f(x)$