- **Expectation Maximization:**
  iterative algorithm for maximizing likelihood

- Given: vectors of "visible" variables \( v_n \)
  hidden: vectors of "hidden" variables \( h_n \)
  where \( n \) is example index

- Complete data set: \( V = \{ v_n \}, H = \{ h_n \} \)
  true data

- Model specifies a joint distribution,

\[
P(U, H | \Theta)
\]

4 parameters of model

- Usually i.i.d. data

\[
P(U, H | \Theta) = \prod_{n} P(v_n, h_n | \Theta)
\]

Then whole analysis "decomposes"!

- \( P(U | \Theta) = \sum_{H} P(U, H | \Theta) \)

  \[
  = \sum_{H} P(U | H, \Theta) P(H | \Theta)
  \]

  **Goal:**
  maximize \( \ln P(U | \Theta) \)

  \[
  \text{Note: Its log of a sum over the hidden variables}
  \]

  \[
  \text{variables}
  \]
Example 1: Mixture of $m$ fixed densities

\[ \Theta = \{ \Theta_1, \Theta_2, \ldots, \Theta_m \} \]

mixture coefficients

Statistics often use $\Theta$ for their model parameter

\[ P(x|\Theta) = \sum_i \Theta_i P(x|i) \]

\[ \text{visible} \quad \text{hidden} \]

Formally: \[ P(x|\Theta) = \sum_i P(x|i\Theta) \]

\[ \frac{\text{parameter}}{\text{visible}} \quad \Theta_i \]

\[ = \sum_i P(x|i, \Theta) P(i|\Theta) \]

\[ \frac{\text{fixed}}{\text{densities}} \]

Example 2: Mixture of Gaussians

\[ \Theta = \{ \Sigma_i \} \cup \{ \mu_i \} \cup \{ \Sigma_i \} \]

\[ \text{mixture} \quad \text{coefficients} \quad \text{means} \quad \text{covariance matrices} \]

\[ P(x|\Theta) = \sum_i P(i|\Theta) P(x|i, \Theta) \]

\[ = \sum_i \gamma_i \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} e^{-\frac{1}{2} (x-\mu_i)^T \Sigma_i^{-1} (x-\mu_i)} \]

Point $x \in \mathbb{R}^d$ visible

\[ \gamma_i \in \{ 1, 2, \ldots, m \} \quad \text{hidden} \]

\[ \# \text{of Gaussians} \]
Example 3: Hidden Markov Models

\[ P(\text{abc, 1231} | \Theta) = \frac{0.1 \times 0.1 \times 0.3 \times 0.6 \times 0.2 \times 0.1 \times 0.4 \times 0.1}{\text{observed sequence}} \times \frac{0.1 \times 0.2 \times 0.3 \times 0.7 \times 0.2 \times 0.3 \times 0.7 \times 0.2}{\text{hidden state sequence}} \]

PARAMETERS: \( \Theta = \{ \Theta_i \}_{i=1}^{m} \)
- \( \Theta_i \) initial state probabilities
- \( \Theta_i \) output probabilities
- \( \Theta_i \) transition probabilities

\[ P(x, s | \Theta) = \prod_{i=1}^{m} \Theta_i \cdot m_i(x, s) \]

Where \( m_i \) is the number of times \( \Theta_i \) occurs in \( (x, s) \)

\[ P(x | \Theta) = \sum_{s} P(x, s | \Theta) \]

sum over all hidden paths
\[ \text{Loss: } -\ln P(U \mid \Theta) \]

\[ = -\ln \sum_{H} P(U, H \mid \Theta) \]

Hard to minimize because log of sum!

I.i.d. case: Decomposition into a sum
\[ U = \{ u_n \}, \quad H = \{ h_n \} \quad \text{sequences of one fixed length} (u_n, h_n) \]

**Assumption:** \( P(U, H \mid \Theta) = \prod_{n=1}^{N} p(u_n, h_n \mid \Theta) \)

**This implies:**

\[ P(U \mid \Theta) = \sum_{H} P(U, H \mid \Theta) \]

\[ = \sum_{H} \prod_{n=1}^{N} p(u_n, h_n \mid \Theta) \]

\[ = \sum_{h_1, h_2, \ldots, h_N} \prod_{n=1}^{N} p(u_n, h_n \mid \Theta) \]

\[ = \prod_{n} \sum_{h_n} p(u_n, h_n \mid \Theta) \]

\[ = \prod_{n} P(u_n \mid \Theta) \]

\[ P(H \mid U, \Theta) = \frac{P(H, U \mid \Theta)}{P(U \mid \Theta)} = \prod_{n} \frac{P(h_n, u_n \mid \Theta)}{P(u_n \mid \Theta)} = \prod_{n} P(h_n \mid u_n, \Theta) \]

\[ -\ln P(U \mid \Theta) = -\sum_{n} \ln P(U \mid \Theta) \]

**Product becomes sum**

**Loss is essentially a relative entropy**

\[ \sum_{n} \frac{1}{N} \ln \frac{N}{P(u_n \mid \Theta)} = -\frac{1}{N} \sum_{n} \ln (P(u_n \mid \Theta)) - \ln N \]
T. I. D. makes relative entropies decompose as well

\[ \sum_{H} \frac{p(H|U,G)}{p(H|U,G)} \ln \frac{p(H|U,G)}{p(H|U,G)} \]

\[ = \sum_{H} \prod_{n} p(u_{n}|v_{n},G) \ln \frac{\prod_{n} p(u_{n}|v_{n},G)}{\prod_{n} p(u_{n}|v_{n},\bar{G})} \]

\[ = \sum_{n} \sum_{u_{n}} p(u_{n}|v_{n},G) \ln \frac{p(u_{n}|v_{n},G)}{p(u_{n}|v_{n},\bar{G})} \]

**Simplest Case**

\[ \sum_{x,y} p(x) p(y) \ln \frac{p(x)p(y)}{q(x)q(y)} \]

\[ = \sum_{x,y} p(x)p(y) \left( \ln \frac{p(x)}{q(x)} + \ln \frac{p(y)}{q(y)} \right) \]

\[ = \sum_{x} p(x) \ln \frac{p(x)}{q(x)} + \sum_{y} p(y) \ln \frac{p(y)}{q(y)} \]

**Upshot:**

All becomes sums over examples
\[
\sum_{H} P(H \mid V, \Theta) \ln \frac{P(H \mid V, \Theta)}{P(H \mid V, \Theta)} + \eta (-\ln P(V \mid \Theta)) \tag{*}
\]

\[
= \sum_{H} P(H \mid V, \Theta) \ln \frac{P(H, V \mid \Theta)}{P(H, V \mid \Theta)} - \eta \ln P(V \mid \Theta)
\]

\[
= \sum_{H} P(H \mid V, \Theta) \ln \frac{P(H, V \mid \Theta)}{P(H, V \mid \Theta)} - \eta \ln P(V \mid \Theta)
\]

\[
= \sum_{H} P(H \mid V, \Theta) \ln \frac{P(H, V \mid \Theta)}{P(H, V \mid \Theta)} - \eta \ln P(V \mid \Theta)
\]

\[
\eta = 1
\]

\[
\sum_{H} P(H \mid V, \Theta) \ln \frac{P(H, V \mid \Theta)}{P(H, V \mid \Theta)} - \eta \ln P(V \mid \Theta)
\]

\[
\text{Easy to minimize!}
\]

### Estimation Step:
- Compute posterior \( P(H \mid V, \Theta) \).

### Maximization Step:
\[
\Theta := \arg \max_{\Theta} \sum_{H} P(H \mid V, \Theta) \ln P(H, V \mid \Theta)
\]

### IID Case:
\[
P(V, H | \Theta) = \frac{1}{N} \prod_{n=1}^{N} P(v_n, h_n | \Theta)
\]

\( (\ast) \) becomes:
\[
- \sum_{n,m} \sum_{h_n} P(l_n \mid l_m, \Theta) \ln P(l_n, v_m | \Theta) + \text{constant}
\]

**E-step:**
- Compute posteriors \( P(l_n | l_m, \Theta) \).

**M-step:**
\[
\Theta := \arg \max_{\Theta} \sum_{n,m} \sum_{h_n} P(l_n \mid l_m, \Theta) \ln P(l_n, v_m | \Theta)
\]
Example 1: \[ P(x|\theta) = \sum_i \frac{p(i|\theta) p(x|i, \theta)}{\sum_j p(x|i, \theta)} = \sum_i \theta_i p(x|i) \]

E-step: 
\[ P(i|x_n, \theta) = \frac{P(i, x_n|\theta)}{P(x_n|\theta)} = \frac{\theta_i p(x_n|i)}{\sum_j \theta_j p(x_n|j)} \]

H-step:
Maximize 
\[ \sum_n \sum_i P(i|x_n, \theta) \ln p(x_n|i, \tilde{\theta}) + \lambda \left( \sum_i \tilde{\theta}_i - 1 \right) \]

\[ = \sum_n \sum_i P(i|x_n, \theta) \ln \tilde{\theta}_i p(x|i) + \lambda \left( \sum_i \tilde{\theta}_i - 1 \right) \]

constant

\[ \frac{\partial}{\partial \tilde{\theta}_i} = \sum_n \frac{P(i|x_n, \theta)}{\tilde{\theta}_i} + \lambda = 0 \]

\[ \Rightarrow \sum_n P(i|x_n, \theta) + \lambda \tilde{\theta}_i = 0 \] (**)

\[ \sum_i \sum_n P(i|x_n, \theta) + \lambda \sum_i \tilde{\theta}_i = 0 \]

\[ \lambda = -N \]

From (**): \[ \tilde{\theta}_i = \frac{1}{N} \sum_n P(i|x_n, \tilde{\theta}) \]

\[ N = 1 \text{ Bayes Rule} \]
Example 2: Mixture of Gaussians

$p(x|\theta) \text{ not independent of } \theta$

\[
p(x|\theta) = \sum_{i} p(i|\theta) p(x|i, \theta)
\]
\[
= \sum_{i} \delta_{ij} (2\pi)^{-d/2} |\Sigma_{i}|^{-1/2} e^{-\frac{1}{2}(x-\mu_{i})^T \Sigma_{i}^{-1} (x-\mu_{i})}
\]

E-step:

\[
p(i|x_{n}, \theta) = \frac{p(i|x_{n}, \theta)}{p(x_{n}|\theta)} = \frac{\delta_{ij} p(x|i, \theta)}{\sum_{j} \delta_{ij} p(x|j, \theta)}
\]

M-step:

\[
\text{Maximize } \sum_{n} \sum_{i} p(i|x_{n}, \theta) \ln p(i, x_{n}|\theta)
\]
\[
= \sum_{n} \sum_{i} p(i|x_{n}, \theta) \ln \frac{p(i|\theta) p(x_{n}|i, \theta)}{\delta_{ij}}
\]

3 Classes of parameters:

a) Mixture coefficients \(\delta_{ij}\)

b) Means \(\mu_{i}\)

c) Covariance matrices \(\Sigma_{i}\)

Maximize above for one class at a time while keeping the other classes fixed.
a) \( \mu_i \) and \( \Sigma_i \) fixed

Thus \( P(x_n | \mu_i, \Sigma_i) = P(x_n | \mu_i, \theta) \)

(*) becomes \( \sum \sum P(i | x_n, \theta) \ln \tilde{\theta}_i + \text{constant} \)

Minimize as before with Lagrangeian:

\[
\tilde{\theta}_i = \frac{1}{N} \sum_{n} P(i | x_n, \theta)
\]

b) and c): \( \mu_i \) fixed

Thus \( P(i | \theta) = P(i | \theta) = \theta_i \)

(*) becomes \( \sum \sum P(i | x_n, \theta) \ln P(x_n | i, \theta) \)

\[
\sum_{n} \sum_{i} P(i | x_n, \theta) \ln \left( \frac{1}{(2\pi)^{d/2} \Sigma_i^{1/2}} e^{-\frac{1}{2} (x_n - \mu_i)^T \Sigma_i^{-1} (x_n - \mu_i)} \right) + \text{const}
\]

\[
\sum_{n} \sum_{i} P(i | x_n, \theta) \left( -\frac{1}{2} \ln |\Sigma_i| - \frac{1}{2} (x_n - \tilde{\mu}_i)^T \Sigma_i^{-1} (x_n - \tilde{\mu}_i) \right) + \text{const}
\]

\[
\tilde{\mu}_i = \mu_i
\]
\[ \frac{\partial}{\partial \mu_i} \sum_i \Sigma_i = \sum_i \sum_m \rho(x_m, \theta) \Sigma_i \rho(x_m, \theta) = 0 \]

\[ \Rightarrow \sum_i \rho(x_m, \theta) - \Sigma_i \sum_m \rho(x_m, \theta) (x_n - \mu_i) = 0 \]

\[ \Rightarrow \sum_i \rho(x_m, \theta) \Sigma_i = \sum_m \rho(x_m, \theta) (x_n - \mu_i) (x_n - \mu_i)^T \]

High-level initialization; parameters are chosen unperturbed.
Intuition why EM works:

Minimizing \(-\ln P(U|\theta)\) = \(-\ln \sum_{H} P(U, H|\theta)\)

is hard because "in" of a sum.

Minimizing \(-\sum_{H} P(H|U, \theta) \ln P(H, U|\theta)\)

"easy" when \(P(H, U|\theta)\) is product!

EM avoids minimizing "in" of a sum directly.

By adding distance, minimization simplified.
\[-\ln P(U|\theta)\]

B) = \(-\ln \sum_{H} P(U, H|\theta)\) has many symmetries.

For example by renaming any local minima

for a mixture of \(m\) distinct Gaussians becomes

\(m\) local minima.

\[-\ln \sum_{H} P(H|U, \theta) \ln P(H, U|\theta)\] not as many

symmetries. Hidden variables are "coupled"

with visible variables.
Example 3: HMM's

\[ P(x_1:s_1 \theta) = \prod_{i=1}^{T} \theta_i^{n_i(x_1:s_1)} \]

Maximize

\[ \sum_{m} \sum_{s_m} P(s_m | x_m, \theta) \ln P(x_m, s_m | \tilde{\theta}) \]

\[ = \sum_{m} \sum_{s_m} P(s_m | x_m, \theta) \sum_{i} n_i(x_m, s_m) \ln \tilde{\theta}_i \]

\[ = \sum_{m} \sum_{i} \ln \tilde{\theta}_i \sum_{s_m} P(s_m | x_m, \theta) n_i(x_m, s_m) \]

\[ \tilde{n}_i(x, \theta) \]

Expected usage of param. \( \theta_i \)

Computable by dynamic program.

\[ \left[ i \right] = \left\{ j : \theta_i \text{ and } \theta_j \text{ in same "class"} \right\} \]

All parameters of a class must sum to one.

* two classes associated with a state.
* one class associated with the initial state probabilities.

Maximize

\[ \sum_{m} \sum_{i} \left( \ln \tilde{\theta}_i \right) \tilde{n}_i(x_m, \theta) + \sum_{[i]} \lambda_{[i]} \left( \sum_{j \in [i]} \theta_j - 1 \right) \]

\[ \frac{\partial}{\partial \tilde{\theta}_i} = \sum_{m} \frac{n_i(x_m, \theta)}{\tilde{\theta}_i} + \lambda_{[i]} = 0 \]

\[ \tilde{\theta}_i = \sum_{m} \frac{n_i(x_m, \theta)}{\lambda_{[i]} + \sum_{j \in [i]} \theta_j} \]

Enforcing constraint: \( \sum_{j \in [i]} \tilde{\theta}_j = 1 \)

\[ \tilde{\theta}_i = \frac{\sum_{m} \tilde{n}_i(x_m, \theta)}{\sum_{j \in [i]} \sum_{m} \tilde{n}_j(x_m, \theta)} \]
EM is too slow!

Synthetic Data 1: 2 State HMM

0.99

S1

(0.63, 0.37)

0.01

0.01

S2

(0.37, 0.63)

Entropic Update (eta=1.5) ×
Baum-Welch (EM) ×

LATER

-0.71
-0.705
-0.7
-0.695
-0.69
-0.685
-0.68
-0.67
Synthetic Data 1 (cont.)

Baum–Welch (EM)

Entropic Update (eta=1.5)
Framework for Parameter Update

- Add a penalty term to loss which will keep the new parameter set "close" to the old set.

\[
\begin{align*}
\Delta(\tilde{\Theta}, \Theta) & \quad \text{(1)} \\
L(X | \tilde{\Theta}) & \\
\Theta^* & \\
\end{align*}
\]

- Solve:

\[
\Theta^{t+1} = \arg \min_{\Theta} U^t(\Theta) \\
U^t(\Theta) = \Delta(\Theta, \Theta^t) + \eta \text{ loss}(S|\tilde{\Theta})
\]

\(\eta\) is a non-negative trade-off parameter that becomes the learning rate of the algorithm.

Any update of the form (1) is called Implicit Update.
Minimal Properties of the Divergence

(i) $\Delta(\Theta, \Theta) = 0$

(ii) $\Delta(\Theta, \Theta) > 0$ whenever $\Theta \neq \Theta$.

Key Lemma

If $U^t(\Theta) < U^t(\Theta)$
then $\text{loss}(S|\Theta) < \text{loss}(S|\Theta)$.

Proof:

$U^t(\Theta) = \Delta(\Theta, \Theta) + \eta \text{ loss}(S|\Theta)$
$\quad < U^t(\Theta) = \Delta(\Theta, \Theta) + \eta \text{ loss}(S|\Theta) \overset{(ii)}{=} \eta \text{ loss}(S|\Theta)$

This is equivalent to

$\text{loss}(S|\Theta) = \text{loss}(S|\Theta) - \frac{\Delta(\Theta, \Theta)}{\eta}$

$\quad \overset{(ii)}{<} \text{loss}(S|\Theta)$

For any implicit update s.t. $\Theta^{t+1} \neq \Theta^t$,

$\text{loss}(S|\Theta^{t+1}) < \text{loss}(S|\Theta^t)$
What's good about EM:

- Implicit update. Thus negative log likelihood decreases in each iteration
- Simple and elegant

Bad: Converges too slowly

\[ \sum_{H} \ln \frac{P(H | V, \theta)}{P(H | \theta)} - \eta \ln P(V | \theta) \]

Simplification:

\[ \sum_{H} P(H | V, \theta) \ln P(H | V, \theta) + \text{const.} \]

Want \( \eta > 1 \). In that case simplification does not work!!!

Idea 1: Use \( \eta > 1 \) but approximate \(-\ln P(V | \theta)\) by 1st order Taylor. Does not seem to work.

Idea 2: Use \( \eta > 1 \), different distance, and 1st order Taylor.

\[ \sum_{V'} \sum_{H} P(H, V' | \theta) \ln \frac{P(H, V' | \theta)}{P(H, \theta)} \]

- \( \eta (\ln P(V | \theta) + (\theta - \bar{\theta}) \frac{\partial \ln P(V | \theta)}{\partial \theta}) \)

Explicit!
Distance we use:
- different direction of entropy
- \( V \) integrated over domain
- avoids "ln" of sum in different way.

In all three examples our method converges faster.

\[
\frac{\partial - \ln P(V | \tilde{\Theta}_n)}{\partial \eta} \bigg|_{\eta=0} < 0 \quad \text{unless at extremum}
\]

provided \( \eta \) is close enough to 0, then loss decreases.

We don't know why our method is so good.