Chapter 4 - Estimation by the EM algorithm

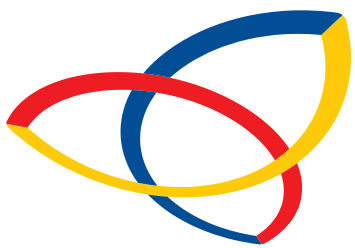
02433 - Hidden Markov Models

Pierre-Julien Trombe, Martin Wæver Pedersen, Henrik Madsen

Course week 4

MWP, compiled June 7, 2011

LUNDS UNIVERSITET
Lunds Tekniska Högskola



Interreg IV A
ÖRESUND – KATTEGAT – SKAGERRAK

DTU

Summary: The EM algorithm

In the context of hidden Markov models the expectation-maximization (EM) algorithm is known as the Baum-Welch algorithm.

The algorithm assumes:

- The Markov Chain is homogeneous (i.e. the transition probablities $\Pr(C_{s+t} = i | C_t = j)$ are independent of t).
- The Markov chain is not necessarily stationary. That is the initial distribution, $\boldsymbol{\delta}$, is to be estimated.

The algorithm is used to estimate all parameters of the HMM, i.e. the transition probabilities, the parameters of the state dependent distribution and the initial distribution $\boldsymbol{\delta}$. An advantage of the algorithm is that the likelihood is guaranteed to increase for each iteration, however convergence toward the maximum may be slow. The algorithm is furthermore derivative-free, i.e. it does not require an optimizer (such as nlm in R). It does, however, require more effort in the implementation phase.

Forward probabilities

Recall that the forward probability vector $\boldsymbol{\alpha}_t$ for $t = 1, \ldots, T$ is given by

$$\boldsymbol{\alpha}_t = \boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\ldots\boldsymbol{\Gamma}\mathbf{P}(x_t) = \boldsymbol{\delta}\mathbf{P}(x_1)\prod_{s=2}^{t}\boldsymbol{\Gamma}\mathbf{P}(x_s),$$

$$\boldsymbol{\alpha}_{t+1} = \boldsymbol{\alpha}_t\boldsymbol{\Gamma}\mathbf{P}(x_{t+1}).$$

In the supplementary slides for chapter 3 it was shown that the elements of $\boldsymbol{\alpha}_t$ are

$$\alpha_t(i) = \Pr(C_t = i, \mathbf{X}^{(t)} = \mathbf{x}^{(t)}).$$

Since $\alpha_t(i)$ is a joint probability of the observations by time t and the hidden state C_t it will for large values of t be a small number.

The reason for the name "forward probabilities" is that the probabilities are calculated by a recursion *forward* in time.

Backward probabilities 1/2

The backward probability vector for $t = 1, \ldots, T$ is

$$\boldsymbol{\beta}_t' = \boldsymbol{\Gamma}\mathbf{P}(x_{t+1})\boldsymbol{\Gamma}\mathbf{P}(x_{t+2})\ldots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}' = \left(\prod_{s=t+1}^{T}\boldsymbol{\Gamma}\mathbf{P}(x_s)\right)\mathbf{1}'$$

$$\boldsymbol{\beta}_t' = \boldsymbol{\Gamma}\mathbf{P}(x_{t+1})\boldsymbol{\beta}_{t+1}',$$

with the convention $\boldsymbol{\beta}_T = \mathbf{1}$.

For $t = 1, \ldots, T-1$ and for $i = 1, \ldots, m$ we have the recursion

$$\beta_t(i) = \Pr(\mathbf{X}_{t+1}^T = \mathbf{x}_{t+1}^T | C_t = i), \quad \text{with} \quad \mathbf{X}_{t+1}^T = (X_{t+1}, \ldots, X_T).$$

Note that $\beta_t(i)$ is a conditional probability in contrast with $\alpha_t(i)$ which is a joint probability.

The reason for the name "backward probabilities" is that a recursion backward in time is used to calculate $\boldsymbol{\beta}_t$.

Backward probabilities 2/2

Derivation of the expression for $\beta_t(i)$ using the Markov property that $\Pr(\mathbf{X}_{t+1}^T|C_{t+1}) = \Pr(\mathbf{X}_{t+1}^T|C_{t+1},\ldots,C_1)$:

$$
\begin{aligned}
\beta_T(i) &= \mathbf{1} \\
\beta_{T-1}(i) &= \Pr(X_T = x_T|C_{T-1} = i) \\
\beta_{T-2}(i) &= \sum_j \gamma_{ij} p_j(x_{T-1})\beta_{T-1}(j) \\
&= \sum_j \Pr(C_{T-1} = j|C_{T-2} = i)\Pr(X_{T-1} = x_{T-1}|C_{T-1} = j) \\
&\times \Pr(X_T = x_T|C_{T-1} = j) \\
&= \sum_j \Pr(X_T = x_T, X_{T-1} = x_{T-1}, C_{T-1} = j|C_{T-2} = i) \\
&= \Pr(X_T = x_T, X_{T-1} = x_{T-1}|C_{T-2} = i) \\
&\vdots \\
\beta_t(i) &= \Pr(\mathbf{X}_{t+1}^T = \mathbf{x}_{t+1}^T|C_t = i).
\end{aligned}
$$

Forward/backward probabilities properties

For $t = 1, \ldots, T$ and $i = 1, \ldots, m$,

$$\alpha_t(i)\beta_t(i) = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}, C_t = i)$$

$$\boldsymbol{\alpha}_t\boldsymbol{\beta}_t' = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}) = L_T,$$

owing to the conditional independence of $\mathbf{X}_1^t$ and $\mathbf{X}_{t+1}^T$:

$$\Pr(\mathbf{X}^{(T)}|C_t = i) = \Pr(\mathbf{X}_1^t|C_t = i)\Pr(\mathbf{X}_{t+1}^T|C_t = i).$$

Note that L_T can be calculated in T ways using the above equation.

The following two quantities are useful in applying the EM algorithm:
For$t = 1, \ldots, T$

$$\Pr(C_t = i|\mathbf{X}^{(t)} = \mathbf{x}^{(t)}) = \frac{\Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}, C_t = i)}{\Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)})} = \alpha_t(i)\beta_t(i)/L_T.$$

For $t = 2, \ldots, T$

$$\Pr(C_{t-1} = i, C_t = j|\mathbf{X}^{(t)} = \mathbf{x}^{(t)}) = \alpha_{t-1}(i)\gamma_{ij}p_j(x_t)\beta_t(j)/L_T.$$

EM in general

Intuition: The unobserved states of the Markov chain are considered as missing data and replaced by their conditional expectations. This is advantageous if the complete-data log-likelihood (CDLL) is straightforward to maximize. The CDLL is the log-likelihood of the parameters based on the observed and missing data.

The EM iterations:

- Choose the starting values to the parameters to be estimated.
- E-step: Compute the conditional expectations of those functions of the missing data appear in the complete-data log-likelihood.
- M-step: Maximization of the log-likelihood with respect to the set of parameters to be estimated (the missing data are substituted by their conditional expectation).
- Assess convergence (with respect to some criterion) and repeat the E and M-steps until convergence is reached.

Important: In the following $\boldsymbol{\delta}$ is the initial distribution of the hidden state (i.e. the hidden process is not assumed to be stationary).

EM for HMMs (1/2)

Some notation:

- $u_j(t) = 1$ if and only if $c_t = j$, $(t = 1, 2, \ldots, T)$.
- $v_{jk}(t) = 1$ if and only if $c_{t-1} = j$ and $c_t = k$, $(t = 2, 3, \ldots, T)$.

The CDLL can be written as:

$$\log\left(\Pr(\mathbf{x}^{(T)}, \mathbf{c}^{(T)})\right) = \log\left(\delta_{c_1} \prod_{t=2}^{T} \gamma_{c_{t-1}, c_t} \prod_{t=1}^{T} p_{c_t}(x_t)\right)$$

$$= \sum_{j=1}^{m} u_j(1) \log \delta_j + \sum_{j=1}^{m} \sum_{k=1}^{m} \left(\sum_{t=2}^{T} v_{jk}(t)\right) \log \gamma_{jk}$$

$$+ \sum_{j=1}^{m} \sum_{t=1}^{T} u_j(t) \log p_j(x_t).$$

Note: the CDLL is partitioned into three terms that can be optimized separately. The first term depends only on the initial distribution, the second term depends only on the transition matrix, and the third term depends only on the parameters related to the state dependent distributions.

EM for HMMs (2/2)

E-step: Compute the conditional expectations of the $u_j(t)$ and $v_{jk}(t)$:

$$\hat{u}_j(t) = \Pr(C_t = j|\mathbf{x}^{(T)}) = \alpha_t(j)\beta_t(j)/L_T$$
$$\hat{v}_{jk}(t) = \Pr(C_{t-1} = j, C_t = k|\mathbf{x}^{(T)}) = \alpha_{t-1}(j)\gamma_{jk}p_k(x_t)\beta_t(k)/L_T.$$

M-step: Replace $u_j(t)$ and $v_{jk}(t)$ with $\hat{u}_j(t)$ and $\hat{v}_{jk}(t)$ respectively in the CDLL and maximize each term

- Term 1. $\sum_{j=1}^{m} \hat{u}_j(1) \log \delta_j$ with respect to $\boldsymbol{\delta}$.
- Term 2. $\sum_{j=1}^{m} \sum_{k=1}^{m} \left(\sum_{t=2}^{T} \hat{v}_{jk}(t) \right) \log \gamma_{jk}$ with respect to $\boldsymbol{\Gamma}$.
- Term 3. $\sum_{j=1}^{m} \sum_{t=1}^{T} \hat{u}_j(t) \log p_j(x_t)$ with respect to the state dependent parameters.

Solutions for the maxima of the terms

Term 1. Depends only on the initial distribution.

$$\delta_j = \frac{\hat{u}_j(1)}{\sum_{j=1}^{m} \hat{u}_j(1)} = \hat{u}_j(1).$$

Term 2. Depends only on the transition probability matrix.

$$\gamma_{jk} = \frac{f_{jk}}{\sum_{k=1}^{m} f_{jk}} \quad \text{with} \quad f_{jk} = \sum_{t=2}^{T} \hat{v}_{jk}(t).$$

Term 3. Depends on the type of the state dependent distribution. Some distributions (Poisson, Gaussian) have analytical solutions. Others require numerical maximization (Gamma, negative-binomial). Example, Gaussian distribution: $X_t \sim N(\mu_j, \sigma_j^2)$.

$$\hat{\mu}_j = \sum_{t=1}^{T} \hat{u}_j(t) x_t / \sum_{t=1}^{T} \hat{u}_j(t)$$

$$\hat{\sigma}_j^2 = \sum_{t=1}^{T} \hat{u}_j(t)(x_t - \hat{\mu}_j)^2 / \sum_{t=1}^{T} \hat{u}_j(t)$$

Advantages and disadvantages of the EM

Advantages

- Likelihood is guaranteed to increase for each iteration.
- Is a derivative-free optimizer.
- Is fast if analytical expressions for the M-step are available.
- Parameter constraints are often dealt with implicitly.

Disadvantages

- Requires both forward and backward probabilities (numerical optimization requires only forward).
- Significant implementational effort required compared to numerical optimization.
- Convergence may be slow if analytical expressions for the M-step are not available since numerical optimization must be applied.
- Hessian must be calculated manually.

Exercises

5,6