The EM Algorithm

Michael Collins

In fulfillment of the Written Preliminary Exam II requirement, September 1997

Contents

1 Introduction

The Expectation Maximization (EM) algorithm is a parameter estimation method which falls into the general framework of maximum-likelihood estimation, and is applied in cases where part of the data can be considered to be incomplete, or "hidden". It is essentially an iterative optimisation algorithm which, at least under certain conditions, will converge to parameter values at a local maximum of the likelihood function. There are many statistical models which turn out to be special cases of EM, for example: Hidden Markov Models (HMMs) (Baum 71); the generalisation of HMMs to Stochastic Context-Free Grammars (Baker 79); mixture models; and estimation in cases of missing data.

(Dempster, Laird and Rubin) (from here on referred to as **DLR**)) defined the EM algorithm, and proved certain properties, in particular that at each iteration the log-likelihood of the observed data is guaranteed to be non-decreasing. That is, if $L(\mathbf{\Theta})$ is the likelihood of the observed data given parameter values $\mathbf{\Theta}$, and $\mathbf{\Theta_t}$, $\mathbf{\Theta_{t+1}}$ are the parameter values at the t'th and $t+1$'th iterations respectively, then $L(\mathbf{\Theta}_{t+1}) \geq L(\mathbf{\Theta}_t)$. They also defined Generalised EM (GEM) algorithms, which include EM as a special case, and can be more computationally efficient, while still guaranteeing that $L(\mathbf{\Theta}_{t+1}) \geq L(\mathbf{\Theta}_t)$.

(Wu 1983) addressed two issues:

1. Given that L converges to some value L^*, then is L^* a global maximum, local maximum, saddle point or some other point? It is well known that L^* can not, in general, be guaranteed to be a global maximum. $L(\mathbf{\Theta}_{t+1}) \geq L(\mathbf{\Theta}_t)$ is one condition for convergence to a stationary point of L, (Wu 83) defines additional conditions for convergence of an EM or GEM algorithm to a stationary point. At least for EM algorithms, these conditions are quite mild. He also gave a condition for convergence to a local maximum as opposed to a saddle point, but this condition is difficult to verify in practice (and does not hold in many practical applications).

2. Under what conditions do the parameter estimates $\mathbf{\Theta}$ also converge to some point $\mathbf{\Theta}^*$? Convergence of L to a point L^* does not guarantee convergence of the parameter estimates to some $\mathbf{\Theta}^*$, particularly if there is more than one point $\mathbf{\Theta}$ satisfying $L(\mathbf{\Theta}) = L^*$.

(JJ 93) emphasise that EM is an optimisation algorithm for L, and show that it is approximately a steepest descent algorithm, an optimisation method which often converges slowly. They show that with a relatively minor increase in complexity the EM algorithm can be modified to a conjugate-gradient descent method, which is known to be an improved optimisation algorithm. They give experimental results showing that their algorithm typically converges around 3-10 times faster than standard EM, and can in some cases be 25-100 times faster.

The remainder of this paper gives some background about maximum-likelihood estimation in section 2; considers the major results of **DLR**, (Wu 83) and (JJ 77) in sections 3, 4 and 5; and concludes in section 6. For a summary of the major points of this paper the reader should refer at this point to the bullet points in section 6.

2 Preliminaries

Most of the results in this section are taken from [BD 77].

2.1 Notation

We use bold-face throughout to denote matrices, normal typeface to denote scalars. Given a vector $\mathbf{X}$, we write its i'th component as X_i. We use the $\mathbf{D}$ operator to denote differentiation. Where there is ambiguity regarding which variable differentiation is with respect to, we use superscripts on the $\mathbf{D}$ operator. For example, $\mathbf{D}^{10}Q(\mathbf{\Theta_1}, \mathbf{\Theta_2})$ is the first derivative of Q w.r.t. $\mathbf{\Theta_1}$, $\mathbf{D}^{01}Q(\mathbf{\Theta_1}, \mathbf{\Theta_2})$ is the first derivative w.r.t. $\mathbf{\Theta_2}$.

2.2 Maximum-likelihood Estimation

In general we have

- a *sample* $\mathbf{X} = \{X_1, X_2, ...X_n\}$ where each X_i is a random variable (a single value, or vector of values).
- A vector of *parameters* $\mathbf{\Theta}$ such that we can define the *likelihood* of the data $P(\mathbf{X}|\mathbf{\Theta})$. We can also define the *log-likelihood* $L(\mathbf{X}|\mathbf{\Theta}) = \log P(\mathbf{X}|\mathbf{\Theta})$. Often the X_is are independently identically distributed (i.i.d.) so that $L(\mathbf{X}|\mathbf{\Theta}) = \sum_{i=1...n} \log P(X_i|\mathbf{\Theta})$.

If Ω is the parameter space, maximum-likelihood (ML) estimation involves setting the ML estimate $\mathbf{\Theta}_{ML}$ such that

$$\mathbf{\Theta}_{ML} = \arg\max_{\mathbf{\Theta}\in\mathbf{\Omega}} L(\mathbf{X}|\mathbf{\Theta}) \tag{1}$$

2.2.1 An example

Suppose we toss a coin 6 times, and $X_i = 1$ if the i'th toss is heads, 0 if it is tails. Say our sample $\mathbf{x} = \{1, 0, 0, 0, 1, 0\}$. Assume the coin has a probability p of being heads, $1 - p$ of being tails, so that $\mathbf{\Theta} = p$. Then

$$\begin{aligned} L(\mathbf{X} = \mathbf{x}|\mathbf{\Theta}) &= \sum_{i=1}^{n} log(P(X_i = x_i|p)) \\ &= 2\log p + 4\log(1-p) \end{aligned} \tag{2}$$

We can maximize $\mathbf{L}$ by setting the derivative w.r.t. p equal to 0:

$$\frac{d\,L(\mathbf{X}=\mathbf{x}|\mathbf{\Theta})}{d\,p} \quad = \quad \frac{2}{p} - \frac{4}{1-p} = 0 \tag{3}$$

Solving this gives $p = \frac{2}{6}$, which is the "intuitive" estimate for p, the proportion of heads which have been seen in the sample.

Another common example of maximum-likelihood estimation is when the components of $\mathbf{X}$ are drawn i.i.d. from a normal distribution with unknown mean μ and known variance σ^2. It's simple enough to prove that the ML estimate for μ is $\frac{\sum X_i}{n}$, i.e., the sample mean.

2.3 Sufficient Statistics

A statistic $\mathbf{T}(\mathbf{X})$ is any real or vector-valued function of the data $\mathbf{X}$. Note that if $\mathbf{T}(\mathbf{X_1}) = \mathbf{T}(\mathbf{X_2})$ for two samples $\mathbf{X_1}$ and $\mathbf{X_2}$ such that $\mathbf{X_1} \neq \mathbf{X_2}$ then $\mathbf{T}$ reduces the data, by mapping different samples to the same value. $\mathbf{T}$ is *sufficient* if there are functions $g(\mathbf{T}(\mathbf{X}), \mathbf{\Theta})$ and $h(\mathbf{X})$ s.t.

$$P(\mathbf{X}|\mathbf{\Theta}) \quad = \quad g(\mathbf{T}(\mathbf{X}), \mathbf{\Theta})h(\mathbf{X}) \tag{4}$$

Typically, $g(\mathbf{T}(\mathbf{X}), \mathbf{\Theta}) = P(\mathbf{T}(\mathbf{X})|\mathbf{\Theta})$ and $h(\mathbf{X}) = P(\mathbf{X}|\mathbf{T}(\mathbf{X})$. The crucial point is that when maximizing $P(\mathbf{X}|\mathbf{\Theta})$ w.r.t. $\mathbf{\Theta}$ we can simply maximize $g(\mathbf{T}(\mathbf{X}), \mathbf{\Theta})$, so the sufficient statistics *summarise* the data – for ML estimation, once we know $\mathbf{T}$ we don't need to know anything else about the data.

2.3.1 An example

For the coin-tossing example, if the sample size is n and the number of heads in the sample is N_h, then

$$P(\mathbf{X}|\mathbf{\Theta}) \quad = \quad p^{N_h}(1-p)^{(n-N_h)} \tag{5}$$

So $\mathbf{T} = (N_h, n)$ is sufficient.

2.4 Exponential Families

An important class of distributions is the exponential family, where the likelihood can be written

$$P(\mathbf{X}|\mathbf{\Theta}) \quad = \quad \{\exp[\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + d(\mathbf{\Theta}) + S(\mathbf{X})]\}I_A(\mathbf{X}) \tag{6}$$

I_A is the indicator function over the set A, and A cannot depend on $\mathbf{\Theta}$. Note that $\mathbf{T}(\mathbf{X}) = \{T_1(\mathbf{X}), T_2(\mathbf{X})...T_n(\mathbf{X})\}$ is sufficient.

If we define the parameters $\mathbf{\Theta} = \{\Theta_1, \Theta_2, ...\Theta_n\}$ such that $C_i(\mathbf{\Theta}) = \Theta_i$ then these are called the *natural* parameters. This can be a useful simplification, for example if when maximizing $\mathbf{L}$ we differentiate w.r.t. $\mathbf{\Theta}$, where for the natural parameters the derivative is then a simple function involving $\mathbf{T}$.

2.4.1 An example: the normal distribution

$$
\begin{aligned}
P(X|\mathbf{\Theta}) &= \frac{1}{\sigma\sqrt{2\pi}}\exp[-\frac{(X-\mu)^2}{2\sigma^2}] \\
&= \exp[-\frac{X^2}{2\sigma^2}+\frac{\mu}{\sigma^2}X-\frac{\mu^2}{2\sigma^2}-\log\sigma\sqrt{2\pi}] \qquad (7)
\end{aligned}
$$

In this case $\mathbf{C}(\mathbf{\Theta}) = \{-\frac{1}{2\sigma^2}, \frac{\mu}{\sigma^2}\}$, $\mathbf{T}(\mathbf{X}) = \{X^2, X\}$, $d(\mathbf{\Theta}) = \frac{\mu^2}{2\sigma^2} - \log\sigma\sqrt{2\pi}$ The natural parameters are $\{-\frac{1}{2\sigma^2}, \frac{\mu}{\sigma^2}\}$, being functions of the conventional parameters $\{\mu, \sigma\}$.

2.4.2 Other important properties

By noting that (by the definition of probability)

$$\int P(\mathbf{X}|\mathbf{\Theta})\, d\mathbf{X} = 1 \qquad (8)$$

it is easy to show

$$d(\mathbf{\Theta}) = -\log\int\{\exp[\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + S(\mathbf{X})]\}I_A(\mathbf{X})\ d\mathbf{X} \qquad (9)$$

Using[1] $\bigtriangledown(-\log f(\Theta)) = -\frac{f'(\Theta)}{f(\Theta)}$, and assuming that we're using natural parameters (hence $\bigtriangledown\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) = \mathbf{T}(\mathbf{X})$)

$$
\begin{aligned}
d'(\mathbf{\Theta}) &= -\frac{\bigtriangledown\int\{\exp[\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + S(\mathbf{X})]\}I_A(\mathbf{X})\ d\mathbf{X}}{\int\{\exp[\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + S(\mathbf{X})]\}I_A(\mathbf{X})\ d\mathbf{X}} \\
&= -\frac{\bigtriangledown\int\{\exp[\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + S(\mathbf{X})]\}I_A(\mathbf{X})\ d\mathbf{X}}{\exp[-d(\mathbf{\Theta})]} \\
&= -\frac{\int \mathbf{T}(\mathbf{X})\{\exp[\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + S(\mathbf{X})]\}I_A(\mathbf{X})\ d\mathbf{X}}{\exp[-d(\mathbf{\Theta})]} \\
&= -\int \mathbf{T}(\mathbf{X})\{\exp[\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + S(\mathbf{X}) + d(\mathbf{\Theta})]\}I_A(\mathbf{X})\ d\mathbf{X} \\
&= -\int \mathbf{T}(\mathbf{X})P(\mathbf{X}|\mathbf{\Theta})\ d\mathbf{X} \\
&= -\mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}] \qquad (10)
\end{aligned}
$$

Now note that the log-likelihood

$$L(\mathbf{X}|\mathbf{\Theta}) = \sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + d(\mathbf{\Theta}) + S(\mathbf{X}) \qquad (11)$$

[1] $\bigtriangledown$ refers to differentiation w.r.t. $\mathbf{\Theta}$

So to obtain the ML estimates by differentiating w.r.t. $\mathbf{\Theta}$ (again, assuming natural parameters)

$$\begin{aligned} L'(\mathbf{X}|\mathbf{\Theta}) &= \mathbf{T}(\mathbf{X}) + d'(\mathbf{\Theta}) \\ &= \mathbf{T}(\mathbf{X}) - \mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}] \end{aligned} \tag{12}$$

So setting $\mathbf{T}(\mathbf{X}) = \mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}]$ will give $L'(\mathbf{X}|\mathbf{\Theta}) = 0$, and maximize the log-likelihood. For example, for a binomial distribution, the sufficient statistic $\mathbf{T}(\mathbf{X}) = \sum X_i$ and $\mathbf{E}[\sum \mathbf{X_i}|\mathbf{p}] = np$ where n is the sample size and p is the binomial parameter. So solving $\sum X_i = np$ gives the ML estimate of p.

If we assume non-natural parameters, then (12) is modified to give

$$L'(\mathbf{X}|\mathbf{\Theta}) = \frac{d\mathbf{C}(\mathbf{\Theta})}{d\mathbf{\Theta}} [\mathbf{T}(\mathbf{X}) - \mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}]] \tag{13}$$

Solving $\mathbf{T}(\mathbf{X}) = \mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}]$ is also a solution to (13), but this solution may not always exist – it may be necessary to also solve (13) as it stands (see section 3.5.1 for an example where $\mathbf{T}(\mathbf{X}) = \mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}]$ has no solution, but $\frac{d\mathbf{C}(\mathbf{\Theta})}{d\mathbf{\Theta}} [\mathbf{T}(\mathbf{X}) - \mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}]] = 0$ does have a solution.)

3 The EM algorithm

The EM algorithm assumes the following problem definition: we have two sample spaces $\mathcal{X}$ and $\mathcal{Y}$, such that there is a many-one mapping $\mathbf{Y} = f(\mathbf{X})$ from an observation $\mathbf{X}$ in $\mathcal{X}$ to an observation $\mathbf{Y}$ in $\mathcal{Y}$. We define

$$\mathcal{X}(\mathbf{Y}) = \{\mathbf{X} : f(\mathbf{X}) = \mathbf{Y}\} \tag{14}$$

$\mathbf{X}$ is the *complete* data, and $\mathbf{Y}$ is the *observed* data. If the distribution $f(\mathbf{X}|\mathbf{\Theta})$ is well defined then the probability of $\mathbf{Y}$ given $\mathbf{\Theta}$ is

$$g(\mathbf{Y}|\mathbf{\Theta}) = \int_{\mathcal{X}(\mathbf{Y})} f(\mathbf{X}|\mathbf{\Theta}) \; d\mathbf{X} \tag{15}$$

EM attempts to solve the following problem: given a sample from $\mathbf{Y}$ is observed, but the corresponding $\mathbf{X}$ are unobserved, or *hidden*, find the maximum-likelihood estimate $\mathbf{\Theta}_{ML}$ which maximizes $L(\mathbf{\Theta}) = \log g(\mathbf{Y}|\mathbf{\Theta})$. In general, $\log f(\mathbf{X}|\mathbf{\Theta})$ will have an easily-defined, analytically solvable maximum, but maximization of $L(\mathbf{\Theta})$ has no analytic solution. EM is an iterative optimisation algorithm which defines a sequence of parameter settings through a mapping $\mathbf{\Theta}_t \to \mathbf{\Theta}_{t+1}$ such that $L(\mathbf{\Theta}_{t+1}) \geq L(\mathbf{\Theta}_t)$ with equality holding only at stationary points of $L(\mathbf{\Theta})$. Thus EM is a hill-climbing algorithm which, at least under certain conditions, will converge to a stationary point of $L(\mathbf{\Theta})$.

The mapping $\mathbf{\Theta}_t \to \mathbf{\Theta}_{t+1}$ is defined in two steps:

1. The **Estimation** step. Define $\tilde{p}(\mathbf{X}) = p(\mathbf{X}|\mathbf{Y}, \mathbf{\Theta}_t)$. (Note that $\tilde{p}(\mathbf{X}) = 0$ outside $\mathcal{X}(\mathbf{Y})$.) Calculate

$$Q(\mathbf{\Theta}', \mathbf{\Theta}_t) = E\left[\log f(\mathbf{X}|\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] = \int \tilde{p}(\mathbf{X}) \log f(\mathbf{X}|\mathbf{\Theta}') \; d\mathbf{X} \tag{16}$$

2. The **Maximization** step. Set $\mathbf{\Theta}_{t+1} = \arg\max_{\mathbf{\Theta}'} Q(\mathbf{\Theta}', \mathbf{\Theta}_t)$.

The intuition is as follows: if we had the complete data, we would simply estimate $\mathbf{\Theta}'$ to maximize $\log f(\mathbf{X}|\mathbf{\Theta}')$. But with some of the complete data missing we instead maximize the *expectation* of $\log f(\mathbf{X}|\mathbf{\Theta}')$ given the observed data and the current value of $\mathbf{\Theta}$.

3.1 An example

Say we observe a series of coin-tosses which we assume have been generated in the following way: a person has two coins in her pocket. Coin 1 has probability of heads $= p_1$, coin 2 has probability p_2. At each point she chooses coin 1 with probability λ, coin 2 with probability $1-\lambda$, and tosses it 3 times. Thus the observed data is a sequence of triples of coin tosses, e.g. $\mathbf{Y} = \{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$. The complete data $\mathbf{X}$, if we could observe it, would additionally show the coin chosen at each step, e.g. $\mathbf{X} = \{\langle HHH, 1\rangle, \langle TTT, 2\rangle, \langle HHH, 1\rangle, \langle TTT, 2\rangle\}$. The parameters, all of which are to be estimated, are $\mathbf{\Theta} = \{\lambda, p_1, p_2\}$.

Assume that $\mathbf{X}$ is unobserved. Then the EM steps are as follows.

1. The **estimation** step: define $\tilde{p}_i = P(X_i = \langle Y_i, 1\rangle \mid Y_i, \mathbf{\Theta})$, i.e. the probability of the i'th coin being coin 1, given the observed data and the current parameter settings. If $P_c(Y_i|p)$ is the probability of seeing Y_i given a coin with prob of heads $= p$, then we have

$$
\begin{aligned}
f_i(X_i = \langle Y_i, 1\rangle \mid \mathbf{\Theta}) &= \lambda P_c(Y_i|p_1) && (17)\\
g_i(Y_i \mid \mathbf{\Theta}) &= \lambda P_c(Y_i|p_1) + (1-\lambda) P_c(Y_i|p_2) && (18)\\
\tilde{p}_i &= \frac{f_i(X_i = \langle Y_i, 1\rangle \mid \mathbf{\Theta})}{g_i(Y_i \mid \mathbf{\Theta})} && (19)\\
&= \frac{\lambda P_c(Y_i|p_1)}{\lambda P_c(Y_i|p_1) + (1-\lambda) P_c(Y_i|p_2)} && (20)
\end{aligned}
$$

$\tilde{p}_i$ is the *posterior* probability of coin 1 having generated the i'th observation. If we define H_i as the number of heads in Y_i then $P_c(Y_i|p) = p^{H_i}(1-p)^{3-H_i}$. Say $\mathbf{\Theta}' = \{\lambda', p_1', p_2'\}$. As the samples are i.i.d., we can write

$$
\begin{aligned}
& E\left[\log f(\mathbf{X}|\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] \\
= & \sum E\left[\log f_i(\mathbf{X}|\mathbf{\Theta}') \mid \tilde{p}_i\right] \\
= & \sum \tilde{p}_i \log(f_i(X_i = \langle Y_i, 1\rangle \mid \mathbf{\Theta})) + (1-\tilde{p}_i)\log(f_i(X_i = \langle Y_i, 2\rangle \mid \mathbf{\Theta})) \\
= & \sum \tilde{p}_i \log \lambda' P_c(Y_i|p_1') + (1-\tilde{p}_i)\log(1-\lambda') P_c(Y_i|p_2') \\
= & \sum \tilde{p}_i \log \lambda' p_1'^{H_i}(1-p_1')^{3-H_i} + (1-\tilde{p}_i)\log(1-\lambda') p_2'^{H_i}(1-p_2')^{3-H_i} \\
= & \sum \tilde{p}_i \log \lambda' + (1-\tilde{p}_i)\log(1-\lambda') + \tilde{p}_i \log p_1'^{H_i}(1-p_1')^{3-H_i} + (1-\tilde{p}_i)\log p_2'^{H_i}(1-p_2')^{3-H_i}
\end{aligned}
\tag{21}
$$

2. The **Maximization** step: Maximizing this function by setting the differentials w.r.t. λ', p_1' and p_2' respectively to 0 gives the following update formulae:

$$\lambda' = \frac{\sum \tilde{p}_i}{n} \tag{22}$$

$$p_1' = \frac{\sum \frac{H_i}{3}\tilde{p}_i}{\sum \tilde{p}_i} \tag{23}$$

$$p_2' = \frac{\sum \frac{H_i}{3}(1-\tilde{p}_i)}{\sum (1-\tilde{p}_i)} \tag{24}$$

These formulae have a nicely intuitive interpretation. λ is the average posterior probability of coin 1 having generated the i'th sample. p_1 is a weighted average over the observations of the usual ML estimate, $\frac{H_i}{3}$, where the weight corresponds to $\tilde{p}_i$, the posterior probability of coin 1 for Y_i. Similarly, p_2 is a weighted average over the observations, where the weight corresponds to $1-\tilde{p}_i$, the posterior probability of coin 2 generating Y_i. See tables 1, 2 and 3 for examples of the EM algorithm for this problem.

3.2 Proof that $L(\mathbf{\Theta})$ is non-decreasing at each iteration

A crucial property of the EM algorithm is that the log-likelihood $L(\mathbf{\Theta}) = \log g(\mathbf{Y}|\mathbf{\Theta})$ is non-decreasing at each iteration. Formally, if we define the EM mapping as $\mathbf{\Theta}_t \rightarrow \mathbf{\Theta}_{t+1}$ where $\mathbf{\Theta}_{t+1} = \arg\max_{\mathbf{\Theta}'} Q(\mathbf{\Theta}', \mathbf{\Theta}_t)$ then $L(\mathbf{\Theta}_{t+1}) \geq L(\mathbf{\Theta}_t)$. The proof rests on two results:

1. Define $k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta})$ to be the posterior likelihood of the complete data given the data $\mathbf{Y}$ and the parameters $\mathbf{\Theta}$, so that $k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}) = \frac{f(\mathbf{X}|\mathbf{\Theta})}{g(\mathbf{Y}|\mathbf{\Theta})}$. If we define $H(\mathbf{\Theta}',\mathbf{\Theta}) = E\left[\log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right]$, (as before, $\tilde{p}(\mathbf{X}) = p(\mathbf{X}|\mathbf{Y},\mathbf{\Theta})$), then

$$L(\mathbf{\Theta}') = Q(\mathbf{\Theta}',\mathbf{\Theta}) - H(\mathbf{\Theta}',\mathbf{\Theta}) \tag{25}$$

2.

$$\forall \mathbf{\Theta}' \quad H(\mathbf{\Theta}',\mathbf{\Theta}) \leq H(\mathbf{\Theta},\mathbf{\Theta}) \tag{26}$$

with equality iff $\log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}') = \log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta})$ almost everywhere.

Given (25),

$$L(\mathbf{\Theta}_{t+1}) - L(\mathbf{\Theta}_t) = \{Q(\mathbf{\Theta}_{t+1},\mathbf{\Theta}_t) - Q(\mathbf{\Theta}_t,\mathbf{\Theta}_t)\} - \{H(\mathbf{\Theta}_{t+1},\mathbf{\Theta}_t) - H(\mathbf{\Theta}_t,\mathbf{\Theta}_t)\} \tag{27}$$

But $\{Q(\mathbf{\Theta}_{t+1},\mathbf{\Theta}_t) - Q(\mathbf{\Theta}_t,\mathbf{\Theta}_t)\} \geq 0$ (by the definition of the M step), and from (26) $\{H(\mathbf{\Theta}_{t+1},\mathbf{\Theta}_t) - H(\mathbf{\Theta}_t,\mathbf{\Theta}_t)\} \leq 0$, so clearly $L(\mathbf{\Theta}_{t+1}) - L(\mathbf{\Theta}_t) \geq 0$.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.3000	0.6000	0.0508	0.6967	0.0508	0.6967
1	0.3738	0.0680	0.7578	0.0004	0.9714	0.0004	0.9714
2	0.4859	0.0004	0.9722	0.0000	1.0000	0.0000	1.0000
3	0.5000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000

Table 1: The coin example for $\mathbf{Y} = \{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$. The solution that EM reaches is intuitively correct: the coin-tosser has two coins, one which always shows up heads, the other which always shows tails, and is picking between them with equal probability ($\lambda = 0.5$). The posterior probabilities $\tilde{p}_i$ show that we are certain that coin 1 (tail-biased) generated Y_2 and Y_4, whereas coin 2 generated Y_1 and Y_3.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$	$\tilde{p}_5$
0	0.3000	0.3000	0.6000	0.0508	0.6967	0.0508	0.6967	0.0508
1	0.3092	0.0987	0.8244	0.0008	0.9837	0.0008	0.9837	0.0008
2	0.3940	0.0012	0.9893	0.0000	1.0000	0.0000	1.0000	0.0000
3	0.4000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000

Table 2: The coin example for $\{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle\}$. λ is now 0.4, indicating that the coin-tosser has probability 0.4 of selecting the tail-biased coin.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.3000	0.6000	0.1579	0.6967	0.0508	0.6967
1	0.4005	0.0974	0.6300	0.0375	0.9065	0.0025	0.9065
2	0.4632	0.0148	0.7635	0.0014	0.9842	0.0000	0.9842
3	0.4924	0.0005	0.8205	0.0000	0.9941	0.0000	0.9941
4	0.4970	0.0000	0.8284	0.0000	0.9949	0.0000	0.9949

Table 3: The coin example for $\mathbf{Y} = \{\langle HHT\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$. EM selects a tails-only coin, and a coin which is heavily heads-biased ($p_2 = 0.8284$). It's certain that Y_1 and Y_3 were generated by coin 2, as they contain heads. Y_2 and Y_4 could have been generated by either coin, but coin 1 is far more likely.

3.2.1 Proof of equation 25

By the rules of conditional probability,

$$\begin{aligned} k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}') &= \frac{f(\mathbf{X}|\mathbf{\Theta}')}{g(\mathbf{Y}|\mathbf{\Theta}')} \\ \log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}') &= \log f(\mathbf{X}|\mathbf{\Theta}') - \log g(\mathbf{Y}|\mathbf{\Theta}') \end{aligned} \tag{28}$$

We can now take expectations w.r.t. $\tilde{p}(\mathbf{X}) = p(\mathbf{X}|\mathbf{Y},\mathbf{\Theta})$:

$$\begin{aligned} E\left[\log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] &= E\left[\log f(\mathbf{X}|\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] - E\left[\log g(\mathbf{Y}|\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] \\ &= E\left[\log f(\mathbf{X}|\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] - \log g(\mathbf{Y}|\mathbf{\Theta}') \end{aligned} \tag{29}$$

(Note that $E\left[\log g(\mathbf{Y}|\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] = \log g(\mathbf{Y}|\mathbf{\Theta}')$ as $\log g(\mathbf{Y}|\mathbf{\Theta})$ does not depend on $\mathbf{X}$.) So by the definitions of H, Q and L,

$$H(\mathbf{\Theta}',\mathbf{\Theta}) = Q(\mathbf{\Theta}',\mathbf{\Theta}) - L(\mathbf{\Theta}') \tag{30}$$

3.2.2 Proof of equation 26

One thing to note is that $H(\mathbf{\Theta},\mathbf{\Theta}) - H(\mathbf{\Theta}',\mathbf{\Theta})$ is the Kullback-Liebler distance between $k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta})$ and $k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}')$, which is known to be ≥ 0 with equality only if the two distributions are equal.

A formal proof is through the following theorem stated in (Rao 1e.6.6): Let $f(\mathbf{X})$ and $g(\mathbf{X})$ be non-negative and integrable functions, and S be the region in which $f(\mathbf{X}) > 0$. The theorem states that if $\int_S (f(\mathbf{X}) - g(\mathbf{X}))d\,\mathbf{X} \geq 0$, then $\int_S f(\mathbf{X}) \log \frac{f(\mathbf{X})}{g(\mathbf{X})} d\,\mathbf{X} \geq 0$.

If we put $f(\mathbf{X}) = k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta})$ and $g(\mathbf{X}) = k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}')$ then clearly $\int_S (f(\mathbf{X}) - g(\mathbf{X}))d\,\mathbf{X} \geq 0$, as $\int_S f(\mathbf{X}) d\,\mathbf{X} = 1$ and by the laws of probability $\int_S g(\mathbf{X}) d\,\mathbf{X} \leq 1$. Hence

$$\int_S f(\mathbf{X}) \log \frac{f(\mathbf{X})}{g(\mathbf{X})} d\,\mathbf{X} = \int_S k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}) \log \frac{k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta})}{k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}')} \geq 0$$

But

$$\begin{aligned} H(\mathbf{\Theta},\mathbf{\Theta}) - H(\mathbf{\Theta}',\mathbf{\Theta}) &= E\left[\log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}) \mid \tilde{p}(\mathbf{X})\right] - E\left[\log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] \\ &= \int_S k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}) \log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}) - \int_S k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}) \log k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}') \\ &= \int_S k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}) \log \frac{k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta})}{k(\mathbf{X}|\mathbf{Y},\mathbf{\Theta}')} \\ &\geq 0 \end{aligned} \tag{31}$$

3.3 Proof that $L(\mathbf{\Theta})$ is increasing if $\mathbf{\Theta}$ is not a stationary point of L

The result given in **DLR**, that $L(\mathbf{\Theta}_{t+1}) \geq L(\mathbf{\Theta}_t)$, is not all that useful, as the likelihood could remain at the same value at any iteration: for example the trivial mapping $\mathbf{\Theta}_{t+1} = \mathbf{\Theta}_t$ would satisfy it with equality. (Wu 1983) proves the more useful result that if $\mathcal{L}$ is the set of stationary points of L in Ω, where Ω is the space of $\mathbf{\Theta}$, then $L(\mathbf{\Theta}_{t+1}) > L(\mathbf{\Theta}_t)$ for any $\mathbf{\Theta}_t \notin \mathcal{L}$. So this says that L will *increase* unless the algorithm has already reached a stationary point of L. This is one necessary condition for convergence to a stationary point of L.

The proof is as follows: from (25) we can write

$$L(\mathbf{\Theta}_t) = Q(\mathbf{\Theta}_t, \mathbf{\Theta}_t) - H(\mathbf{\Theta}_t, \mathbf{\Theta}_t) \tag{32}$$

Differentiating gives

$$\mathbf{D}L(\mathbf{\Theta}_t) = \mathbf{D}^{10}\, Q(\mathbf{\Theta}_t, \mathbf{\Theta}_t) - \mathbf{D}^{10}\, H(\mathbf{\Theta}_t, \mathbf{\Theta}_t) \tag{33}$$

From (26), $\mathbf{\Theta} = \mathbf{\Theta}_t$ maximizes $H(\mathbf{\Theta}, \mathbf{\Theta}_t)$, so $\mathbf{D}^{10}\, H(\mathbf{\Theta}_t, \mathbf{\Theta}_t) = 0$, therefore

$$\mathbf{D}L(\mathbf{\Theta}_t) = \mathbf{D}^{10}\, Q(\mathbf{\Theta}_t, \mathbf{\Theta}_t) \tag{34}$$

If $\mathbf{\Theta}_t \notin \mathcal{L}$ then $\mathbf{D}L(\mathbf{\Theta}_t) \neq 0$, so $\mathbf{D}^{10}\, Q(\mathbf{\Theta}_t, \mathbf{\Theta}_t) \neq 0$. Therefore we cannot be at a maximum of Q, hence given that $\mathbf{\Theta}_{t+1}$ maximizes $Q(\mathbf{\Theta}, \mathbf{\Theta}_t)$ we have

$$\forall \mathbf{\Theta}_t \notin \mathcal{L} \;\; Q(\mathbf{\Theta}_{t+1}, \mathbf{\Theta}_t) > Q(\mathbf{\Theta}_t, \mathbf{\Theta}_t) \tag{35}$$

From (25), (26) and (35) it is clear that

$$\forall \mathbf{\Theta}_t \notin \mathcal{L} \;\; L(\mathbf{\Theta}_{t+1}) > L(\mathbf{\Theta}_t) \tag{36}$$

3.4 Generalised EM (GEM) algorithms

DLR defined a GEM algorithm to be any iterative scheme $\mathbf{\Theta}_t \to \mathbf{\Theta}_{t+1}$ such that $Q(\mathbf{\Theta}_{t+1}, \mathbf{\Theta}_t) \geq Q(\mathbf{\Theta}_t, \mathbf{\Theta}_t)$. The point here is that it is not necessary to maximize Q at each step, instead it is sufficient for Q to simply increase at each step to ensure that $L(\mathbf{\Theta}_{t+1}) \geq L(\mathbf{\Theta}_t)$. In some situations it is less computationally demanding to increase Q at each step rather than to maximize it. Clearly EM algorithms are a special case of GEM algorithms.

As it stands, this definition is flawed. An additional criterion is required for L to converge to a stationary point, namely

$$\forall \mathbf{\Theta}_t \notin \mathcal{L} \;\; Q(\mathbf{\Theta}_{t+1}, \mathbf{\Theta}_t) > Q(\mathbf{\Theta}_t, \mathbf{\Theta}_t) \tag{37}$$

i.e. Q must be strictly increasing if we have not reached a stationary point of L. Note that EM algorithms automatically satisfy (37) by the proof in section 3.3. (Wu 83) identified this flaw in the definition of GEM algorithms, and states additional conditions (see Theorem 1 of (Wu 83), described in section 4.1) which guarantee convergence. Without the additional conditions non-convergent algorithms such as the trivial $\mathbf{\Theta}_{t+1} = \mathbf{\Theta}_t$ satisfy the GEM definition.

3.5 Special Cases of the EM Algorithm

The generality of the EM formulation is extremely useful, but also means that Q has to be defined for each problem, and furthermore that a method for maximizing Q must be found for each case. This section describes a few special cases of EM problems where Q has been defined and the maximization step has a simple analytic solution.

3.5.1 Exponential Families

Here the complete data is generated from a distribution which is a member of the exponential family, that is $f(\mathbf{X}|\mathbf{\Theta}) = \{\exp[\sum C_i(\mathbf{\Theta})T_i(\mathbf{X}) + d(\mathbf{\Theta}) + S(\mathbf{X})]\}I_A(\mathbf{X})$. **DLR** show that in this case the following is an EM algorithm:

1. **Expectation** step. As before, define $\tilde{p}(\mathbf{X}) = p(\mathbf{X}|\mathbf{Y}, \mathbf{\Theta}_t)$. Calculate
$$\mathbf{T}^p = E\left[\mathbf{T}(\mathbf{X}) \mid \tilde{p}(\mathbf{X})\right] \tag{38}$$

2. **Maximization** step. Find $\mathbf{\Theta}'$ such that
$$E\left[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}'\right] = \mathbf{T}^p \tag{39}$$

Note that (39) is very similar to the usual ML solution for exponential families, setting $\mathbf{T}(\mathbf{X}) = \mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}']$ (see section 2.4.2), except we set $\mathbf{E}[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}']$ to be the *expected value of the sufficient statistics* given the observed data and the current $\mathbf{\Theta}$, instead of calculating the sufficient statistics from the complete data which is unobserved.

The proof that this procedure maximizes Q is as follows:

$$\begin{aligned}
Q(\mathbf{\Theta}', \mathbf{\Theta}) &= \int \tilde{p}(\mathbf{X}) \log f(\mathbf{X}|\mathbf{\Theta}') \;\, d\mathbf{X} \\
&= \int \tilde{p}(\mathbf{X}) \left[\sum C_i(\mathbf{\Theta}')T_i(\mathbf{X}) + d(\mathbf{\Theta}') + S(\mathbf{X})\right] \;\, d\mathbf{X} \\
\mathbf{D}^{10}Q(\mathbf{\Theta}', \mathbf{\Theta}) &= \int \tilde{p}(\mathbf{X})[\frac{d\mathbf{C}(\mathbf{\Theta}')}{d\mathbf{\Theta}}T(\mathbf{X}) + d'(\mathbf{\Theta}')] \;\, d\mathbf{X} \\
&= E\left[\frac{d\mathbf{C}(\mathbf{\Theta}')}{d\mathbf{\Theta}}T(\mathbf{X}) \mid \tilde{p}(\mathbf{X})\right] + E\left[d'(\mathbf{\Theta}') \mid \tilde{p}(\mathbf{X})\right] \\
&= \frac{d\mathbf{C}(\mathbf{\Theta}')}{d\mathbf{\Theta}} E\left[T(\mathbf{X}) \mid \tilde{p}(\mathbf{X})\right] + d'(\mathbf{\Theta}') \\
&= \frac{d\mathbf{C}(\mathbf{\Theta}')}{d\mathbf{\Theta}} \left(E\left[T(\mathbf{X}) \mid \tilde{p}(\mathbf{X})\right] - E\left[T(\mathbf{X}) \mid \mathbf{\Theta}'\right]\right)
\end{aligned} \tag{40}$$

So (38), (39) give $\mathbf{D}^{10}Q(\mathbf{\Theta}', \mathbf{\Theta}) = 0$, and therefore maximize Q, the required result.

A solution to (39) may not actually exist, and a more general formulation of the maximization step is to set
$$\frac{d\mathbf{C}(\mathbf{\Theta}')}{d\mathbf{\Theta}} \left[\mathbf{T}^p - E\left[\mathbf{T}(\mathbf{X})|\mathbf{\Theta}'\right]\right] = 0 \tag{41}$$

An example. **DLR** present an initial motivating example for EM. Say the complete data $\mathbf{X} = (x_1, x_2, x_3, x_4, x_5)$ is drawn from a multinomial distribution $(\frac{1}{2}, \frac{\pi}{4}, \frac{1-\pi}{4}, \frac{1-\pi}{4}, \frac{\pi}{4})$. The observed data $\mathbf{Y} = (y_1, y_2, y_3, y_4) = (125, 18, 20, 34)$, and $y_1 = x_1 + x_2$, $y_2 = x_3$, $y_3 = x_4$, $y_4 = x_5$. Thus x_1 and x_2 are hidden, while $x_1 + x_2$ is observed. The log-likelihood of the complete data can be written
$$\log f(\mathbf{X} \mid \pi) = x_1 \log \frac{1}{2} + x_2 \log \frac{\pi}{4} + x_3 \log \frac{1-\pi}{4} + x_4 \log \frac{1-\pi}{4} + x_5 \log \frac{\pi}{4} + S(\mathbf{X}) \tag{42}$$

where $S(\mathbf{X})$ is a multinomial coefficient. This is then an exponential distribution, with sufficient statistics $\mathbf{T}(\mathbf{X}) = (x_1, x_2, x_3, x_4, x_5)$ and $\mathbf{C}(\mathbf{\Theta}) = (\log\frac{1}{2}, \log\frac{\pi}{4}, \log\frac{1-\pi}{4}, \log\frac{1-\pi}{4}, \frac{\pi}{4})$. While x_3, x_4, x_5 are observed, the expectation step involves estimating x_1^p and x_2^p given the current parameter settings and the observed data $\mathbf{Y}$. This gives

$$x_1^p = E\left[x_1 \mid \mathbf{Y}, \pi\right] = 125\frac{\frac{1}{2}}{\frac{1}{2} + \frac{\pi}{4}} \tag{43}$$

$$x_2^p = E\left[x_2 \mid \mathbf{Y}, \pi\right] = 125\frac{\frac{\pi}{4}}{\frac{1}{2} + \frac{\pi}{4}} \tag{44}$$

Then $E\left[\mathbf{T}(\mathbf{X}) \mid \mathbf{Y}, \pi\right] = (x_1^p, x_2^p, x_3, x_4, x_5)$. We can also calculate $E\left[\mathbf{T}(\mathbf{X}) \mid \pi\right] = n(\frac{1}{2}, \frac{\pi}{4}, \frac{1-\pi}{4}, \frac{1-\pi}{4}, \frac{\pi}{4})$ where $n = y_1 + y_2 + y_3 + y_4$. **DLR**'s maximization step (39) gives

$$\begin{aligned} E\left[\mathbf{T}(\mathbf{X}) \mid \pi'\right] &= E\left[\mathbf{T}(\mathbf{X}) \mid \mathbf{Y}, \pi\right] \\ n(\frac{1}{2}, \frac{\pi'}{4}, \frac{1-\pi'}{4}, \frac{1-\pi'}{4}, \frac{\pi'}{4}) &= (x_1^p, x_2^p, x_3, x_4, x_5) \end{aligned} \tag{45}$$

where π' is the new value for π. However this does not have a solution for π', so it fails as a maximization step (for example, it requires $x_3 = x_4 = n\frac{1-\pi'}{4}$, but $x_3 = 18$, $x_4 = 20$). If we instead use the more general formula in (41), noting that $\frac{d\mathbf{C}(\mathbf{\Theta})}{d\mathbf{\Theta}} = (0, \frac{1}{\pi}, \frac{-1}{1-\pi}, \frac{-1}{1-\pi}, \frac{1}{\pi})$, then the maximization step becomes

$$\begin{aligned} (0, \frac{1}{\pi'}, \frac{-1}{1-\pi'}, \frac{-1}{1-\pi'}, \frac{1}{\pi'}) \times \{n(\frac{1}{2}, \frac{\pi'}{4}, \frac{1-\pi'}{4}, \frac{1-\pi'}{4}, \frac{\pi'}{4}) - (x_1^p, x_2^p, x_3, x_4, x_5)\}^T &= 0 \\ \frac{x_2^p + x_5}{\pi'} - \frac{x_3 + x_4}{1-\pi'} &= 0 \\ \pi' &= \frac{x_2^p + x_5}{x_2^p + x_3 + x_4 + x_5} \end{aligned} \tag{46}$$

This is precisely the solution given in equation (1.5) of **DLR**.

3.5.2 Algebraic Models

(Lafferty) describes an important class of EM problem — EM applied to algebraic models. Say $\mathbf{\Theta} = \{p_1, p_2, ...p_n\}$ is the combination of m multinomial distributions $\Omega_1, \Omega_2, ..\Omega_m$ such that the Ωs are disjoint subsets (forming a partition of $\{1, 2, 3..n\}$) of the integers $\{1, 2, ..., n\}$ and $\{p_i : i \in \Omega_j\}$ are the parameters of the j'th multinomial, so that $\sum_{i\in\Omega_j} p_i = 1$. The probability of the complete data can be written

$$f(\mathbf{X}|\mathbf{\Theta}) = \prod_{i\in\Omega_1} p_i^{C(i,\mathbf{X})} \prod_{i\in\Omega_2} p_i^{C(i,\mathbf{X})} ... \prod_{i\in\Omega_m} p_i^{C(i,\mathbf{X})} \tag{47}$$

where $C(i, \mathbf{X})$ is the count of the event in $\mathbf{X}$ which corresponds to p_i — $C(i, \mathbf{X})$ are the sufficient statistics for $\mathbf{X}$. In this case, if $\mathbf{\Theta}' = \{p_1, p_2, ...p_n\}$ then the Q function is:

$$Q(\mathbf{\Theta}', \mathbf{\Theta}) = \sum \tilde{p}(\mathbf{X}) \sum_{i\in\Omega_1} C(i, \mathbf{X}) \log p_i + \sum \tilde{p}(\mathbf{X}) \sum_{i\in\Omega_2} C(i, \mathbf{X}) \log p_i + ... \sum \tilde{p}(\mathbf{X}) \sum_{i\in\Omega_m} C(i, \mathbf{X}) \log p_i \tag{48}$$

We can maximize Q by maximizing each of these double sums separately. This gives m constrained optimisation problems, for example maximize

$$Q_1(\mathbf{\Theta}', \mathbf{\Theta}) \quad = \quad \sum \tilde{p}(\mathbf{X}) \sum_{i \in \Omega_1} C(i, \mathbf{X}) \log p_i \tag{49}$$

subject to the constraint

$$\sum_{i \in \Omega_1} p_i = 1 \tag{50}$$

Using Lagrange multipliers, the unconstrained problem is to maximize

$$\hat{Q}_1(\mathbf{\Theta}', \mathbf{\Theta}) \quad = \quad \sum \tilde{p}(\mathbf{X}) \sum_{i \in \Omega_1} C(i, \mathbf{X}) \log p_i - \lambda \sum_{i \in \Omega_1} p_i \tag{51}$$

Taking partial derivatives w.r.t. p_j and setting them to 0 gives

$$\begin{aligned} \frac{\mathbf{d}\hat{Q}_1(\mathbf{\Theta}', \mathbf{\Theta})}{\mathbf{d}p_j} &= \sum \frac{\tilde{p}(\mathbf{X})C(j, \mathbf{X})}{p_j} - \lambda = 0 \\ \Rightarrow p_j &= \frac{\sum \tilde{p}(\mathbf{X})C(j, \mathbf{X})}{\lambda} \end{aligned} \tag{52}$$

If we define $\tilde{C}(j, \mathbf{X}) = \sum \tilde{p}(\mathbf{X})C(j, \mathbf{X})$ and find λ s.t. $\sum_{i \in \Omega_1} p_i = 1$, then

$$p_j \quad = \quad \frac{\tilde{C}(j, \mathbf{X})}{\sum_{i \in \Omega_1} \tilde{C}(i, \mathbf{X})} \tag{53}$$

$\tilde{C}(j, \mathbf{X})$ can be interpreted as the *expected* count corresponding to parameter p_j, and p_j is then the normalised expected count.

Examples of algebraic models Hidden markov models are an important class of algebraic model. An EM algorithm for HMMs was first suggested by [Baum 71]. An HMM has n states $\{s_1, s_2 ... s_n\}$. There are $n \times n$ transition probabilities, $P(s_i \to s_j)$ such that $\sum_{j=1..n} P(s_i \to s_j) = 1$. Initial and final states are defined, without loss of generality we take the initial state to be s_1, the final state to be s_n. Given an output alphabet Σ s.t. $|\Sigma| = m$, there are also $n \times m$ emission probabilities $p(i \uparrow j)$, where this is the probability of state s_i emitting symbol j. The observed data is a sequence of symbols from Σ, and the complete data is this sequence together with the underlying sequence of states which generated this data.

Say we observe an output sequence $\{o_1, o_2, ...o_l\}$, and the state sequence is $\{q_1, q_2, ...q_l\}$ (excluding the initial and final states). Then the probability of the complete data is

$$\begin{aligned} & f(\{o_1, o_2, ...o_l\}, \{q_1, q_2, ...q_l\} \mid \mathbf{\Theta}) = \\ \prod p(s_1 \to q_1)p(q_1 \to q_2)p(q_2 \to q_3)...p(q_l \to s_n) & \times \prod (q_1 \uparrow o_1)(q_2 \uparrow o_2)....(q_l \uparrow o_l) \end{aligned} \tag{54}$$

If $C(s_i \rightarrow s_j)$ is the number of times we see $p(s_i \rightarrow s_j)$ in the first product, and $C(s_i \uparrow j)$ is the number of times we see $p(s_i \uparrow j)$ in the second product, then we can re-write this as

$$f(\{o_1, o_2, ...o_l\}, \{q_1, q_2, ...q_l\} \mid \mathbf{\Theta}) = \\ \prod_{i=1..n} \prod_{j=1..n} p(s_i \rightarrow s_j)^{C(s_i \rightarrow s_j)} \times \prod_{i=1..n} \prod_{j=1..m} p(s_i \uparrow j)^{C(s_i \uparrow j)} \tag{55}$$

Thus this defines an algebraic model with $2n$ multinomial distributions. We can define the EM algorithm for HMMs using the general method shown above. The result is a set of expected counts, $\tilde{C}(s_i \rightarrow s_j)$ and $\tilde{C}(s_i \uparrow j)$, with

$$p(s_i \rightarrow s_j) = \frac{\tilde{C}(s_i \rightarrow s_j)}{\sum_{k=1..n} \tilde{C}(s_i \rightarrow s_k)} \qquad p(s_i \uparrow j) = \frac{\tilde{C}(s_i \uparrow j)}{\sum_{k=1..m} \tilde{C}(s_i \uparrow k)} \tag{56}$$

Note that $\sum_{k=1..n} \tilde{C}(s_i \rightarrow s_k) = \sum_{k=1..m} \tilde{C}(s_i \uparrow k)$ is the expected number of times the model was in state s_i given the observed data and the current parameter values. A naive algorithm for this problem would be the following: for every possible state sequence, calculate $f(\{o_1, o_2, ...o_l\}, \{q_1, q_2, ...q_l\} \mid \mathbf{\Theta})$, and the sufficient statistics for $\mathbf{X} = \{\{o_1, o_2, ...o_l\}, \{q_1, q_2, ...q_l\}\}$. From f calculate the marginal $k(\{q_1, q_2, ...q_l\} \mid \{o_1, o_2, ...o_l\}, \mathbf{\Theta})$. From the marginal probabilities and sufficient statistics for each state sequence, calculate the expected counts and from these the parameter values. This algorithm is unworkable though, given that the number of state sequences n^l is exponential. Fortunately the forward-backward algorithm given in [Baum 71] gives a dynamic programming algorithm for calculation of the expected counts, which runs in $\mathcal{O}(n^2 l)$ time.

3.6 Summary of the 4 Theorems in DLR

In the following section we take $M(\mathbf{\Theta})$ to define a GEM algorithm, that is $Q(\mathbf{M}(\mathbf{\Theta}), \mathbf{\Theta}) \geq Q(\mathbf{\Theta}, \mathbf{\Theta})$ for all $\mathbf{\Theta}$. **DLR** states 4 central theorems with respect to GEM algorithms.

3.6.1 Theorem 1

For every GEM algorithm, $L(\mathbf{M}(\mathbf{\Theta})) \geq L(\mathbf{\Theta})$, with equality if and only if both $Q(\mathbf{M}(\mathbf{\Theta}), \mathbf{\Theta}) = Q(\mathbf{\Theta}, \mathbf{\Theta})$ and $k(\mathbf{X} \mid \mathbf{Y}, \mathbf{M}(\mathbf{\Theta})) = k(\mathbf{X} \mid \mathbf{Y}, \mathbf{\Theta})$ almost everywhere. The proof is in section 3.2.

Corollary 1. Suppose for some $\mathbf{\Theta}^*$, $L(\mathbf{\Theta}^*) \geq L(\mathbf{\Theta})$ for all $\mathbf{\Theta}$, i.e. $\mathbf{\Theta}^*$ is a (possibly non-unique) global maximum of L. Then $L(\mathbf{M}(\mathbf{\Theta}^*)) = L(\mathbf{\Theta}^*)$, $Q(\mathbf{M}(\mathbf{\Theta}^*), \mathbf{\Theta}^*) = Q(\mathbf{\Theta}^*, \mathbf{\Theta}^*)$ and $k(\mathbf{X} \mid \mathbf{Y}, \mathbf{M}(\mathbf{\Theta}^*)) = k(\mathbf{X} \mid \mathbf{Y}, \mathbf{\Theta}^*)$ almost everywhere. Hence if EM reaches a global maximum, the likelihood remains fixed at this point.

Corollary 2. Suppose for some $\mathbf{\Theta}^*$, $L(\mathbf{\Theta}^*) > L(\mathbf{\Theta})$ for all $\mathbf{\Theta} \neq \mathbf{\Theta}^*$, i.e. $\mathbf{\Theta}^*$ is a unique global maximum of L. Then for every GEM algorithm $\mathbf{M}(\mathbf{\Theta}^*) = \mathbf{\Theta}^*$. So if EM has reached a unique global maximum, the parameter values remain unchanged at each iteration.

3.6.2 Theorems 2 and 3

Theorems 2 and 3 attempted to show that under certain conditions $\mathbf{\Theta}$ converges to some point $\mathbf{\Theta}^*$. Note that $L(\mathbf{\Theta})$ will usually converge to some value L^* ((Wu 83) defines the exact conditions required for this to be the case), but this does not imply that $\mathbf{\Theta}$ also converges – $\mathbf{\Theta}$ could, for example, be oscillating between points nearby two local maxima with the same maximal value L^*. Unfortunately, as noted by (Wu 1983), a key step of the **DLR** proof is wrong, namely the application of the triangle inequality to go from step (3.13) to (3.14). Because of this both theorems 2 and 3 are invalid.

3.6.3 Theorem 4

It is useful to be able to calculate the *rate of convergence* of an EM algorithm. For the one parameter case where Θ converges to a point Θ^* we can define the rate of convergence $R_{\Theta_p} = \frac{\Theta_{p+1}-\Theta^*}{\Theta_p-\Theta^*}$. This can be interpreted as the reduction in the distance to Θ^* when going from Θ_p to Θ_{p+1}, for example if $\Theta^* = 0.5$, $\Theta_p = 1.0$, $\Theta_{p+1} = 0.75$ then $R_{\Theta_p} = 0.5$, i.e. the distance to Θ^* is halved. It's possible to prove that $lim_{\Theta_p \to \Theta^*} R_{\Theta_p} = M'(\Theta^*)$ where $M'(\Theta) = \frac{dM(\Theta)}{d\Theta}$. So if we can calculate $M'(\Theta^*)$, then we have an estimate of the rate of convergence close to $\mathbf{\Theta}^*$.

Proof. Say $\Theta_p = \Theta^* + \delta$. Then $\Theta_{p+1} = M(\Theta^* + \delta)$. Noting also that $M(\Theta^*) = \Theta^*$, as Θ^* is a maximum of L,

$$\begin{aligned} R_{\Theta_p} &= \frac{\Theta_{p+1} - \Theta^*}{\Theta_p - \Theta^*} \\ &= \frac{M(\Theta^* + \delta) - \Theta^*}{\Theta^* + \delta - \Theta^*} \\ &= \frac{M(\Theta^* + \delta) - M(\Theta^*)}{\delta} \end{aligned} \tag{57}$$

So,

$$\lim_{\Theta_p \to \Theta^*} R_{\Theta_p} = \lim_{\delta \to 0} \frac{M(\Theta^* + \delta) - M(\Theta^*)}{\delta} = \frac{dM(\Theta)}{d\Theta} \tag{58}$$

This can be generalised to the multiparameter case: $\mathbf{R} = \frac{d\mathbf{M}(\mathbf{\Theta})}{d\mathbf{\Theta}} = \mathbf{DM}(\mathbf{\Theta})$ is a vector of rates of convergence, where the i'th component of $\mathbf{R}$ is the rate of convergence of the i'th parameter in $\mathbf{\Theta}$.

Theorem 4 of **DLR** states that

$$\mathbf{DM}(\mathbf{\Theta}^*) = \mathbf{D^{20}H}(\mathbf{\Theta}^*, \mathbf{\Theta}^*) \left[\mathbf{D^{20}Q}(\mathbf{\Theta}^*, \mathbf{\Theta}^*)\right]^{-1} \tag{59}$$

under the following conditions:

1. $\mathbf{\Theta^p}$ converges to some $\mathbf{\Theta}^*$.
2. $\mathbf{D^{10}}Q(\mathbf{\Theta_{p+1}}, \mathbf{\Theta_p}) = 0$. This means that Q is maximized at each iteration, EM algorithms satisfy this condition, as do GEM algorithms which find a stationary point of Q at each iteration.
3. $\mathbf{D^{20}}Q(\mathbf{\Theta_{p+1}}, \mathbf{\Theta_p})$ is negative definite with eigenvalues bounded away from 0.

An example. If we return to the example in section 3.5.1, we can calculate $\mathbf{D^{20}}Q(\pi', \pi)$ and $\mathbf{D^{20}}H(\pi', \pi)$ as follows:

From (40)

$$\begin{aligned}
\mathbf{D^{10}}Q(\pi', \pi) &= (0, \frac{1}{\pi'}, \frac{-1}{1-\pi'}, \frac{-1}{1-\pi'}, \frac{1}{\pi'}) \times \{n(\frac{1}{2}, \frac{\pi'}{4}, \frac{1-\pi'}{4}, \frac{1-\pi'}{4}, \frac{\pi'}{4}) - (x_1^p, x_2^p, x_3, x_4, x_5)\}^T \\
&= \frac{x_2^p + x_5}{\pi'} - \frac{x_3 + x_4}{1-\pi'} \qquad (60)
\end{aligned}$$

(This is the quantity we set to 0 in the maximization step (46)). Differentiating again gives

$$\mathbf{D^{20}}Q(\pi', \pi) = -\frac{x_2^p + x_5}{\pi'^2} - \frac{x_3 + x_4}{(1-\pi')^2} \qquad (61)$$

This is the value given on page 10 of **DLR**. To calculate $\mathbf{D^{20}}H(\pi', \pi)$ we note that $L(\pi') = Q(\pi', \pi) - H(\pi', \pi)$, so that $\mathbf{D^{20}}H(\pi', \pi) = \mathbf{D^{20}}Q(\pi', \pi) - \mathbf{D^2}L(\pi')$ We have

$$\begin{aligned}
L(\pi') &= y_1 \log(\frac{2+\pi'}{4}) + x_3 \log\frac{1-\pi'}{4} + x_4 \log\frac{1-\pi'}{4} + x_5 \log\frac{\pi'}{4} + S(\mathbf{X}) \\
\mathbf{D^1}L(\pi') &= \frac{y_1}{2+\pi'} - \frac{x_3 + x_4}{1-\pi'} + \frac{x_5}{\pi'} \\
\mathbf{D^2}L(\pi') &= -\frac{y_1}{(2+\pi')^2} - \frac{x_3 + x_4}{(1-\pi')^2} - \frac{x_5}{\pi'^2}
\end{aligned} \qquad (62)$$

So

$$\begin{aligned}
\mathbf{D^{20}}H(\pi', \pi) &= \mathbf{D^{20}}Q(\pi', \pi) - \mathbf{D^2}L(\pi') \\
&= -\frac{x_2^p + x_5}{\pi'^2} - \frac{x_3 + x_4}{(1-\pi')^2} - \{-\frac{y_1}{(2+\pi')^2} - \frac{x_3 + x_4}{(1-\pi')^2} - \frac{x_5}{\pi'^2}\} \\
&= -\frac{x_2^p}{\pi'^2} + \frac{y_1}{(2+\pi')^2} \qquad (63)
\end{aligned}$$

Again, this is the value given on page 10 of **DLR**.

4 (Wu 83)'s Commentary on the EM algorithm

(Wu 83) addresses two points concerning the EM algorithm:

1. If L converges to some L^*, what is the nature of L^*? (A global maximum, local maximum, stationary value or other point?) He shows that in general there can only be a guarantee that L^* is a stationary value (i.e. a local/global maximum or a saddle point), and specifies conditions under which L^* falls into these categories (without these conditions, L^* could potentially be *any* value).

2. Under what conditions does $\mathbf{\Theta}$ converge to some $\mathbf{\Theta}^*$? Note that even if L converges, $\mathbf{\Theta}$ may not converge, for example it could oscillate between points on two local maxima which have the same maximum.

(Wu 83) makes the following assumptions throughout, so they can be taken as preconditions of every theorem stated in this section:

- $\mathbf{\Omega}$ is a subset in the r-dimensional Euclidean space R^r.
  ($\mathbf{\Omega}$ is the parameter space so $\mathbf{\Theta} \in \mathbf{\Omega}$). (64)
- $\mathbf{\Omega_{\Theta_0}} = \{\mathbf{\Theta} \in \mathbf{\Omega} : L(\mathbf{\Theta}) \geq L(\mathbf{\Theta_0})\}$ is compact for any $L(\mathbf{\Theta_0}) > -\infty$. (65)
- L is continuous in $\mathbf{\Theta}$ and differentiable in the interior of $\mathbf{\Omega}$. (66)

As a consequence of these conditions it follows that

- $\{L(\mathbf{\Theta_p})\}_{p \geq 0}$ is bounded above for any $\mathbf{\Theta_0} \in \mathbf{\Omega}$. (67)

4.1 Is L^* a global maximum, local maximum or stationary value?

We define the following subsets of $\mathbf{\Omega}$:

- $\mathcal{M}$ is the set of local maxima in the interior of $\mathbf{\Omega}$.
- $\mathcal{L}$ is the set of stationary points in the interior of $\mathbf{\Omega}$.

From this it follows that $\mathcal{L} \setminus \mathcal{M}$ is the set of saddle points in $\mathbf{\Omega}$ (the set of stationary points which are not local maxima).

4.1.1 Theorem 1

Let $\{\mathbf{\Theta}_p\}$ be a GEM sequence generated by $\mathbf{\Theta_{p+1}} \in M(\mathbf{\Theta_p})$. Then L converges monotonically to $L^* = L(\mathbf{\Theta}^*)$ for some $\mathbf{\Theta}^* \in \mathcal{L}$ under the following conditions:

i) M is a closed point-to-set map over the complement of $\mathcal{L}$ (we define closed point-to-set maps in section 4.1.6 below).

ii) $L(\mathbf{\Theta_{p+1}}) > L(\mathbf{\Theta_p})$ for all $\mathbf{\Theta_p} \notin \mathcal{L}$

This theorem also holds if we replace every mention of $\mathcal{L}$ with $\mathcal{M}$ – this gives a similar theorem but for the conditions for convergence to a local maximum rather than just any stationary point.

4.1.2 Theorem 2

Section 3.3 showed that for EM algorithms (as opposed to any GEM algorithm) condition (ii) of theorem 1 holds for the $\mathcal{L}$ (stationary value) case. M can be shown to satisfy condition (i) if $Q(\mathbf{\Theta}', \mathbf{\Theta})$ is continuous in both $\mathbf{\Theta}'$ and $\mathbf{\Theta}$. This leads to theorem 2:

If $Q(\mathbf{\Theta}', \mathbf{\Theta})$ is continuous in both $\mathbf{\Theta}'$ and $\mathbf{\Theta}$, then all the limit points of an EM algorithm are stationary points of L and L converges monotonically to $L^* = L(\mathbf{\Theta}^*)$ for some $\mathbf{\Theta}^* \in \mathcal{L}$.

4.1.3 Theorem 3

Theorem 2 guarantees convergence to a stationary value, but this stationary value could be a saddle point (a member of $\mathcal{L} \setminus \mathcal{M}$). The problem is that EM satisfies condition (ii) of Theorem 1 for stationary values $\mathbf{\Theta_p} \notin \mathcal{L}$, but there may be saddle points $\mathbf{\Theta_s} \in \mathcal{L} \backslash \mathcal{M}$ such that $L(M(\mathbf{\Theta_s})) = L(\mathbf{\Theta_s})$. Theorem 3 states that convergence to a *local maximum* is guaranteed if every saddle point of L is *not* a global maximum of $Q(\mathbf{\Theta}', \mathbf{\Theta})$ w.r.t $\mathbf{\Theta}'$. From (34) $\mathbf{D}Q = \mathbf{0}$ at any saddle point of L, so this must mean that any saddle point of L is a saddle point or local maximum of Q, but *not* the global maximum of Q. If this condition is satisfied then (given that EM maximizes Q at each step) Q will increase even at the saddle point, and L will also increase.

4.1.4 Summary of Theorems 1, 2 and 3

To summarise these theorems, when designing a GEM or EM algorithm:

- For GEM algorithms check that conditions (i) and (ii) of theorem 1 hold, and the algorithm will then converge to some point in $\mathcal{L}$ (or $\mathcal{M}$ for the version of theorem 1 regarding $\mathcal{M}$).
- For EM algorithms check that $Q(\mathbf{\Theta}', \mathbf{\Theta})$ is continuous in both $\mathbf{\Theta}'$ and $\mathbf{\Theta}$, then by theorem 2 the algorithm will converge to some point in $\mathcal{L}$. In addition, if it can be shown that every saddle point of L is *not* a global maximum of Q, then L will converge to some point in $\mathcal{M}$.

4.1.5 Example of Convergence to a Saddle Point

If we return to the example in section 3.1, but initialise p_1 and p_2 to the same value, we get the behaviour in table 4.

Theorem 3 is violated for this example, and in general it's hard to guarantee this theorem's requirement. (Wu 83) suggests that it's important when running EM to try several starting points, and to randomly select initial parameter values, in particular to avoid symmetries such as $p_1 = p_2$ in the last example. Note that if we initialise p_1 and p_2 even slightly differently from each other we get convergence to the global maximum, see table 5.

4.1.6 Proof of Theorem 1

The proof rests on the Global Convergence Theorem, which is stated and proved in (Zangwill 69):

Say $\{x_k\}_{k=0}^{\infty}$ is generated by $x_{k+1} \in M(x_k)$, where M is a point-to-set map on X. (A point-to-set map on X is a function from points in X to subsets of X). Let a solution set Γ be given. Suppose that:

1. All points x_k are contained in a compact set $S \subset X$.
2. M is closed over the complement of Γ. "Closedness" is a generalisation to point-to-set maps from continuity of a point-to-point map; for a point-to-point map continuity implies closedness.
3. There is a continuous function α on X such that

(a) if $x \notin \Gamma$, $\alpha(y) > \alpha(x)$ for all $y \in M(x)$.
(b) if $x \in \Gamma$, $\alpha(y) \geq \alpha(x)$ for all $y \in M(x)$.

Then all limit points of $\{x_k\}$ are in the solution set Γ and $\alpha(x_k)$ converges monotonically to $\alpha(x)$ for some $x \in \Gamma$.

If we take M as a GEM algorithm, $\alpha = L$ and $\Gamma = \mathcal{L}$ (or $\mathcal{M}$ for Theorem 1 for $\mathcal{M}$), GEM algorithms always satisfy the following conditions:

- **Condition (1).** If $\mathbf{\Theta_0}$ is such that $L(\mathbf{\Theta_0}) > -\infty$ then, given that L is non-decreasing for a GEM algorithm, all points x_k are contained in $\mathbf{\Omega_{\Theta_0}}$, which by assumption (65) is compact.
- **Condition (3b).** This comes from L being non-decreasing at each iteration of a GEM algorithm.

The remaining conditions — (2) and (3a) — are conditions (i) and (ii) of Theorem 1, hence if Theorem 1 holds then the global convergence theorem holds. Theorems 2 and 3 follow from Theorem 1.

4.1.7 Corollary 1

(Wu 83) states one additional result concerning the convergence of L to L^*, corollary 1 to Theorem 6:

Suppose $L(\mathbf{\Theta})$ is continuous in $\mathbf{\Omega}$ with $\mathbf{\Theta}^*$ being the only stationary point and that $\mathbf{D^{10}}Q(\mathbf{\Theta}', \mathbf{\Theta})$ is continuous in both $\mathbf{\Theta}'$ and $\mathbf{\Theta}$. Then for any EM sequence $\{\mathbf{\Theta_p}\}$, $\mathbf{\Theta_p}$ converges to the unique maximizer $\mathbf{\Theta}^*$ of $L(\mathbf{\Theta})$.

4.2 Does $\mathbf{\Theta}$ Converge to a point $\mathbf{\Theta}^*$?

Say $\{\mathbf{\Theta}_p\}$ is an instance of a GEM algorithm which satisfies theorem 1. Define $\mathcal{L}(\alpha) = \{\mathbf{\Theta} : L(\mathbf{\Theta}) = \alpha\}$. Then by theorem 1 L converges to L^* and all the limit points of $\{\mathbf{\Theta}_p\}$ are in $\mathcal{L}(L^*)$. Theorems 4 and 5 of (Wu 83) then give conditions where $\mathbf{\Theta}$ converges to a point $\mathbf{\Theta}^*$.

4.2.1 Theorem 4

If $\mathcal{L}(L^*) = \{\mathbf{\Theta}^*\}$, that is there is only one stationary point of L at which L is L^*, then $\mathbf{\Theta} \to \mathbf{\Theta}^*$.

4.2.2 Theorem 5

If $||\mathbf{\Theta_{p+1}} - \mathbf{\Theta_p}|| \to 0$ as $p \to \infty$ then all the limit points of $\mathbf{\Theta_p}$ are in a connected and compact subset of $\mathcal{L}(L^*)$. In particular, if $\mathcal{L}(L^*)$ is discrete, then $\mathbf{\Theta} \to \mathbf{\Theta}^*$ where $\mathbf{\Theta}^*$ is some member of $\mathcal{L}(L^*)$.

4.3 The Non-convergent GEM Algorithm given in (Boyles 83)

(Boyles 83) gives an algorithm which satisfies the GEM definition in **DLR**, and is interesting in two respects:

1. It does not converge to a stationary point of the likelihood function.
2. $\mathbf{\Theta}$ does not converge to some point $\mathbf{\Theta}^*$.

It is useful to examine this example in the context of (Wu 83)'s theorems. The basic idea of (Boyles 83) is to define a GEM algorithm for a two parameter problem, maximization of $L(\theta_1, \theta_2)$. If y_1, y_2 are the ML-estimates for θ_1, θ_2, then in their example it can be shown that all points θ_1, θ_2 which lie on a circle of radius r centered on y_1, y_2 give the same value for $L(\theta_1, \theta_2)$. Moreover, the lower the value of the radius r the more L increases. The GEM algorithm given then defines the mapping $\theta_1^k, \theta_2^k \rightarrow \theta_1^{k+1}, \theta_2^{k+1}$ such that the parameter values spiral in, with the radius decreasing at each point, such that the limit of the parameter values is to rotate on the circle at $r = 1$. Clearly the algorithm fails to reach the maximal point at $r = 0$, and $\mathbf{\Theta}$ does not converge, instead in the limit the parameters continually traverse the circle. So which convergence criteria in (Wu83) does the algorithm fail?

First, it fails Theorem 1 condition (ii). For any $r \leq 1$ the radius remains constant at each iteration, hence the likelihood also remains constant. But $r = 0$ is the only stationary point of L in the problem, so Theorem 1 condition (ii) is violated.

Second, it fails the requirements of both theorems 4 and 5. L converges to a value L^* which has many values of θ_1, θ_2 such that $L(\theta_1, \theta_2) = L^*$, in fact any θ_1, θ_2 which lie on the circle of radius 1. And it is clear that $||\mathbf{\Theta_{p+1}} - \mathbf{\Theta_p}|| \rightarrow 0$ is not satisfied.

5 (Jamshidian and Jennrich 93)

(JJ 93) further emphasise that the EM algorithm is an optimisation algorithm, and apply a standard optimisation algorithm, generalised conjugate gradient descent.

5.1 Optimisation of Quadratic Functions

(Zangwill 69 Chapter 6) describes Conjugate-Gradient methods for optimisation. Taking a Taylor's expansion (page 326 of Zangwill 69) about a local maximum of L at $\mathbf{\Theta}^*$, gives

$$L(\mathbf{X}|\mathbf{\Theta}^* + \delta\mathbf{\Theta}) \approx L(\mathbf{X}|\mathbf{\Theta}^*) + \mathbf{D^1L}(\mathbf{X}|\mathbf{\Theta}^*)\delta\mathbf{\Theta} + \frac{1}{2}\delta\mathbf{\Theta}^\mathbf{T}\mathbf{D^2L}(\mathbf{X}|\mathbf{\Theta}^*)\delta\mathbf{\Theta} \quad (68)$$

Where $\mathbf{D^1}$... and $\mathbf{D^2}$... are the first and second derivatives of L with respect to $\mathbf{\Theta}$. Taylor's theorem shows that this approximation becomes increasingly accurate as $\delta\mathbf{\Theta} \rightarrow 0$. Thus we have a quadratic (second-order) approximation to L close to a maximal point.

Gradient-based optimisation algorithms draw heavily on results for optimisation of a quadratic function such as

$$f(\mathbf{Z}) = c + \mathbf{b^T Z} + \frac{1}{2}\mathbf{Z^T A Z} \quad (69)$$

The most obvious optimisation algorithm is steepest descent: at each point calculate the gradient $\mathbf{D}f$, and move in this direction to the point which maximizes $f(\mathbf{Z}+\alpha\mathbf{D}f)$ where α is the distance moved in the gradient's direction.

A crucial result is that steepest descent algorithms can be very poor as optimisation algorithms for quadratic functions, whereas conjugate gradient methods maximize a quadratic in at most p steps for a p-variate quadratic. If we assume that the function being optimised is well approximated by some quadratic, then we can assume that these results carry over to the function: steepest gradient will be considerably poorer than a conjugate gradient method based on the quadratic approximation of the function.

5.1.1 Conjugate Gradient Methods

Given an $n \times n$ symmetric matrix $\mathbf{A}$, the directions $d_1, d_2 ... d_n$ are said to be $\mathbf{A}$-conjugate if they are linearly independent, and $\mathbf{d_i^t A d_j} = 0$ for $i \neq j$.

It can then be shown that for a quadratic function $f(\mathbf{Z}) = c + \mathbf{b^T Z} + \frac{1}{2}\mathbf{Z^T A Z}$, given n conjugate directions and a starting point $\mathbf{Z_0}$, then the following algorithm will maximize a quadratic in n steps (i.e. $\mathbf{Z_n}$ maximizes f):

- For $k = 1...n$, find the α_k which maximizes $f(\mathbf{Z_{k-1}} + \alpha_k \mathbf{d_k})$. Set $\mathbf{Z_k} = \mathbf{Z_{k-1}} + \alpha_k \mathbf{d_k}$.

All that remains, then, is to find an algorithm which constructs $d_1, d_2..d_n$ for a given quadratic. Conjugate gradient methods are remarkable in that they find $d_1, d_2 ... d_n$ *without knowledge of* $\mathbf{A}$. This is important when maximizing a function using a Taylor approximation such as (68), in that there is no need to calculate the Hessian of the function (in the example, $\mathbf{D^2L}$), a step which can be computationally expensive. A conjugate algorithm proceeds as follows:

- For an arbitrary starting point $\mathbf{Z_0}$, let $d_1 = f'(\mathbf{Z_0})$, where f' is the first derivative of f.
- For $k = 1...n$,
  - set $\mathbf{Z_k} = \mathbf{Z_{k-1}} + \alpha_k d_k$, where α_k maximizes $f(\mathbf{Z_{k-1}} + \alpha_k d_k)$.
  - Calculate $d_{k+1} = f'(\mathbf{Z_k}) + \beta_k d_k$.

The most commonly used values for β_k are (where $g_k = f'(\mathbf{Z_k})$):

- The Fletcher-Reeves version $\beta_k = \frac{\mathbf{g_{k+1}^T g_{k+1}}}{\mathbf{g_k^T g_k}}$
- The Polak-Ribiere algorithm $\beta_k = \frac{\mathbf{(g_{k+1} - g_k)^T g_{k+1}}}{\mathbf{g_k^T g_k}}$. When maximizing a quadratic this is the same as the Fletcher-Reeves algorithm (as $\mathbf{g_k^T g_{k+1}} = 0$). The difference becomes important when maximizing an approximation to a quadratic, where Polak-Ribiere updates have been shown to be better in some cases.

5.1.2 Generalised Conjugate Gradient Methods

A generalised conjugate gradient method is identical to the algorithm in section 5.1.1, except the gradients are modified by some positive definite matrix $\mathbf{W}$. So every mention of f' in section 5.1.1 is replaced by $\mathbf{W}^{-1}f'$. (JJ 93) state that the use of an appropriate matrix $\mathbf{W}$ can significantly improve the performance of any optimisation algorithm which uses gradients.

5.2 Accelerating EM using Generalised Conjugate Gradients

(JJ 93) first show that EM is approximately a steepest descent algorithm optimising L with the generalised gradient $\tilde{\mathbf{g}} = \mathbf{W}^{-1}\mathbf{g}(\mathbf{\Theta})$ where $\mathbf{W} = (-\mathbf{D^{20}Q}(\mathbf{\Theta}', \mathbf{\Theta}'))$. They do this by proving that an EM step $\mathbf{\Theta} \to \mathbf{\Theta}'$ is such that

$$\mathbf{\Theta}' - \mathbf{\Theta} \approx (-\mathbf{D^{20}Q}(\mathbf{\Theta}', \mathbf{\Theta}'))^{-1}\mathbf{g}(\mathbf{\Theta}) \tag{70}$$

where $\mathbf{g}(\mathbf{\Theta}) = \mathbf{D^1}L(\mathbf{\Theta})$. (JJ 93) then define a conjugate gradient method based on the approximate generalised gradients $\tilde{\mathbf{g}}(\mathbf{\Theta}) = \mathbf{\Theta}' - \mathbf{\Theta}$. The k'th iteration of this algorithm is:

1. Perform the k'th EM step to find $\mathbf{\Theta}'$ and calculate $\tilde{\mathbf{g}}_{k+1} = \mathbf{\Theta}' - \mathbf{\Theta_k}$.
2. At the first and every p^{th} step thereafter, where $p =$ the number of parameters in $\mathbf{\Theta}$, set $d_{k+1} = \tilde{\mathbf{g}}(\mathbf{\Theta})$. At other steps:
   (a) Calculate $\mathbf{g}_k = \mathbf{D^1}L(\mathbf{\Theta_k})$.
   (b) Set $\beta_k = \frac{\tilde{\mathbf{g}}^{\mathbf{T}}_{\mathbf{k+1}}(\mathbf{g_{k+1}} - \mathbf{g_k})}{\mathbf{d^T_k}(\mathbf{g_{k+1}} - \mathbf{g_k})}$. This is an alternative to the Fletcher-Reeves and Polak-Ribiere updates.
   (c) Set $\mathbf{d_{k+1}} = \tilde{\mathbf{g}}_{\mathbf{k+1}} - \beta_k \mathbf{d_k}$.
3. Find α_k, the value of α which maximizes $L(\mathbf{\Theta_k} + \alpha\mathbf{d_{k+1}})$, and set $\mathbf{\Theta_{k+1}} = \mathbf{\Theta_k} + \alpha_k\mathbf{d_{k+1}}$.

It can be seen that the algorithm involves a modest increase in complexity over the EM algorithm, mainly the calculation of $\mathbf{g}_k = \mathbf{D^1}L(\mathbf{\Theta_k})$. (JJ 93) then pick a few example applications, and find that their algorithm is a around 3-10 times faster than EM on usual examples, and can be 25-100 times faster on cases where EM converges particularly slowly.

5.3 Discussion

While the results in (JJ 93) are impressive, there are a number of points which are unanswered in the paper:

- They justify the (unusual) choice of β_k because it gives conjugate gradients even when the line searchs for the α_k's aren't exact. But they neither prove this, nor provide a citation.

- They stress that the choice of $\mathbf{W}$ can be crucial when using generalised gradients. But they never given any justification of why $\mathbf{W} = (-\mathbf{D^{20}Q}(\mathbf{\Theta'}, \mathbf{\Theta'}))$ might be a good choice for $\mathbf{W}$. In particular they do not compare their algorithm to a standard conjugate gradient algorithm, which would also require the calculation of g_k but would not require the EM step to calculate $\tilde{g}_k$.

- The method presumably becomes increasingly useful as the maximum of L is approached, as the quadratic approximation becomes increasingly accurate. The method runs a few EM iterations at the start of the algorithm until change in likelihood falls below a certain threshold. No real justification is given for this threshold, nor are tests done to see how robust the method is to the choice of this parameter (and no indication is given of how much this parameter was tuned to the particular problems used).

6 Conclusions

The following is a summary of the major points in this paper:

- To derive an EM or GEM algorithm first find the function $Q(\mathbf{\Theta'}, \mathbf{\Theta})$ as defined in section 3. An EM mapping $f : \mathbf{\Theta_k} \rightarrow \mathbf{\Theta_{k+1}}$ is then $f(\mathbf{\Theta}) = \arg\max_{\mathbf{\Theta'}} Q(\mathbf{\Theta'}, \mathbf{\Theta})$. A GEM mapping is any $f : \mathbf{\Theta_k} \rightarrow \mathbf{\Theta_{k+1}}$ such that $Q(f(\mathbf{\Theta}), \mathbf{\Theta}) \geq Q(\mathbf{\Theta}, \mathbf{\Theta})$.

- Section 3.2 then contains the proof, given in **DLR**, that $L(\mathbf{\Theta_{k+1}}) \geq L(\mathbf{\Theta_k})$ for EM and GEM algorithms. (Wu 83) also proved that for any EM algorithm, $L(\mathbf{\Theta_{k+1}}) > L(\mathbf{\Theta_k})$ at any non-stationary point, (see section 3.3) though this is not necessarily true for any GEM algorithm.

- Section 3.5 gives a couple of special cases – namely the exponential family of distributions, and algebraic models – where it is easy to derive Q and the parameter values which maximize Q. Note that the M-step for exponential families given in **DLR**, (39), may not have a solution, and does not have a solution for the motivating multinomial example given at the start of **DLR**. When no solution is found, the more general (41) should be used.

- Section 3.6.3 describes a method for calculating the rate of convergence of an EM algorithm, as described in **DLR** Theorem 4. The theory requires calculation of the Hessians of the Q and H functions at the local maximum.

- (Wu 83) gives further conditions for the convergence of an EM or GEM algorithm to a stationary point of L. For an EM algorithm the most important criterion is that Q is continuous in both its parameters. See section 4.1 for the exact criteria for convergence. Major points are that convergence to a global maximum can not, in general, be guaranteed, and furthermore convergence to a saddle point is possible – theorem 3 of (Wu 83) gives criteria for convergence to a local maximum rather than a saddle point, but these criteria are unlikely to apply and are hard to prove. "Common wisdom" seems to be that parameter values should be randomly initialised to avoid symmetries which can lead to convergence to saddle points, and that EM should be run a few times with different starting points.

- Section 4.2 gives the theorems provided by (Wu 83) which state conditions for the estimate of $\mathbf{\Theta}$ to also converge. In most applications convergence of the estimate of $\mathbf{\Theta}$ is probably less important than convergence of L though.
- (JJ 93) show that EM is approximately a steepest descent algorithm, and describe a fairly simple modification that gives a conjugate-gradient optimisation algorithm. This is shown to improve performance on a number of problems. The major additional complexity of the method is the calculation of $\mathbf{D}L$ at each iteration.

In conclusion, **DLR** provided a very general framework for defining iterative algorithms which find local maxima of the log-likelihood in incomplete data problems. However, as they defined EM and GEM there was not a guarantee of convergence to a stationary point of L, (Wu 83) considerably tightened the theory by giving strict conditions for convergence. (JJ 93) show that EM, while having the advantage of simplicity, may be a poor algorithm in some situations, and used existing algorithms in the optimisation literature to improve convergence rates.

A major weakness in EM-style algorithms is the guarantee concerning the nature of the limit point — global maximum, local maximum or saddle point? While examples like that in section 4.1.5 suggest that perturbing the parameters slightly will result in the parameters diverging from the saddle point, and **DLR** page 10 suggest that this will always be the case, as far as we know there is no general theory about the "stability" of a saddle point, or a general method for perturbing the parameters in such a way that the algorithm diverges from the saddle point. Nor, as far as we know, is there any general theory about the nature of the log-likelihood surface — conditions under which it has a single global maximum, or under which it has no saddle points. While it may be very difficult to formulate theories about the general case, it seems that it would be possible for special (but extremely common) cases like HMMs or mixture models.

References

[Baker 79] Baker, J. (1979). Trainable Grammars for Speech Recognition. In Jared J. Wolf and Dennis H. Klatt, editors, *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, pages 547-550, MIT, Cambridge, Mass.

[Baum 71] Baum, L.E. (1971). An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes. In *Inequalities, III: Proceedings of a Symposium.* (Shish, Qved ed.). New York: Academic Press.

[BD 77] Bickel and Docksum (1977). *Mathematical Statistics: Basic Ideas and Selected Topics.* Prentice Hall, Englewood Cliffs, New Jersey.

[Boyles 83] Boyles, R.A. (1983). On the Convergence Properties of the EM Algorithm. *Journal of the Royal Statistical Society*, Ser B, 44, 47-50.

[DLR] Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum Likelihood from Incomplete Data Via the EM Algorithm, *Journal of the Royal Statistical Society*, Ser B, 39, 1-38.

[JJ 93] Jamshidian, M. and Jennrich, R.I. (1993). Conjugate Gradient Acceleration of the EM Algorithm , *Journal of the American Statistical Association*, Vol. 88, No 412, 221-228.

[Lafferty] Lafferty, J. Notes on the EM Algorithm. *Unpublished Course Notes.*

[Rao] Rao, C.R. (1965). *Linear Statistical Inference and its Applications.* New York, Wiley.

[Wu 83] Wu, C.F.Jeff (1983). On the Convergence Properties of the EM Algorithm, *The Annals of Statistics*, 11, 95-103.

[Zangwill 69] Zangwill, W.I. (1969). *Nonlinear Programming: A Unified Approach.* Prentice Hall, Englewood Cliffs, New Jersey.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.7000	0.7000	0.3000	0.3000	0.3000	0.3000
1	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
2	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
3	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
4	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
5	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
6	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000

Table 4: The coin example for $\mathbf{Y} = \{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$, with p_1 and p_2 initialised to the same value. EM is stuck at a saddle point

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.7001	0.7000	0.3001	0.2998	0.3001	0.2998
1	0.2999	0.5003	0.4999	0.3004	0.2995	0.3004	0.2995
2	0.2999	0.5008	0.4997	0.3013	0.2986	0.3013	0.2986
3	0.2999	0.5023	0.4990	0.3040	0.2959	0.3040	0.2959
4	0.3000	0.5068	0.4971	0.3122	0.2879	0.3122	0.2879
5	0.3000	0.5202	0.4913	0.3373	0.2645	0.3373	0.2645
6	0.3009	0.5605	0.4740	0.4157	0.2007	0.4157	0.2007
7	0.3082	0.6744	0.4223	0.6447	0.0739	0.6447	0.0739
8	0.3593	0.8972	0.2773	0.9500	0.0016	0.9500	0.0016
9	0.4758	0.9983	0.0477	0.9999	0.0000	0.9999	0.0000
10	0.4999	1.0000	0.0001	1.0000	0.0000	1.0000	0.0000
11	0.5000	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000
Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.6999	0.7000	0.2999	0.3002	0.2999	0.3002
1	0.3001	0.4998	0.5001	0.2996	0.3005	0.2996	0.3005
2	0.3001	0.4993	0.5003	0.2987	0.3014	0.2987	0.3014
3	0.3001	0.4978	0.5010	0.2960	0.3041	0.2960	0.3041
4	0.3001	0.4933	0.5029	0.2880	0.3123	0.2880	0.3123
5	0.3002	0.4798	0.5087	0.2646	0.3374	0.2646	0.3374
6	0.3010	0.4396	0.5260	0.2008	0.4158	0.2008	0.4158
7	0.3083	0.3257	0.5777	0.0739	0.6448	0.0739	0.6448
8	0.3594	0.1029	0.7228	0.0016	0.9500	0.0016	0.9500
9	0.4758	0.0017	0.9523	0.0000	0.9999	0.0000	0.9999
10	0.4999	0.0000	0.9999	0.0000	1.0000	0.0000	1.0000
11	0.5000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000

Table 5: The coin example for $\mathbf{Y} = \{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$. If we initialise p_1 and p_2 to be a small amount away from the saddle point $p_1 = p_2$, the algorithm diverges from the saddle point and eventually reaches the global maximum.