**Some applications of expectation maximization algorithm**

Expectation maximization (EM) algorithm is a popular and powerful mathematical method for statistical parameter estimation in case that there exist both observed data and hidden data. This book focuses on applications of EM in which the implicit relationship is essential to connect observed data and hidden data. In other words, such applications reinforce EM which in turn extends estimation methods like maximum likelihood estimation (MLE) or moment method.

Loc Nguyen

Loc Nguyen’s Academic Network, Vietnam

Email: ng\_phloc@yahoo.com

Phone: +84-975250362

Homepage: www.locnguyen.net

**Table of contents**

[Introduction to expectation maximization algorithm and mixture model 2](#_Toc98426668)

[Conditional mixture model and its application for regression model 8](#_Toc98426669)

[Learning dyadic data and predicting unaccomplished co-occurrent values by mixture model 16](#_Toc98426670)

[Conditional mixture model for modeling attributed dyadic data 30](#_Toc98426671)

[Handling missing data with expectation maximization algorithm 40](#_Toc98426672)

[Expectation Maximization Algorithm with Combinatorial Assumption 71](#_Toc98426673)

# **Introduction to expectation maximization algorithm and mixture model**

Loc Nguyen

Loc Nguyen’s Academic Network, Vietnam

Email: ng\_phloc@yahoo.com

Homepage: www.locnguyen.net

**Abstract**

Expectation maximization (EM) algorithm is a popular and powerful mathematical method for parameter estimation in case that there exist both observed data and hidden data. Therefore, EM is appropriate to applications which aim to exploit laten aspects under heterogeneous data. Mixture model is a popular and successful application of EM. This report focuses on introducing EM and mixture model as the essential chapter in this book.

**Keywords:** Expectation maximization (EM) algorithm, mixture model.

**1. Introduction to expectation maximization algorithm**

Literature of expectation maximization (EM) algorithm in this report is mainly extracted from the preeminent article “Maximum Likelihood from Incomplete Data via the EM Algorithm” by Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin (Dempster, Laird, & Rubin, 1977). For convenience, let DLR be reference to such three authors. The preprint “Tutorial on EM algorithm” (Nguyen, Tutorial on EM algorithm, 2020) by Loc Nguyen is also referred in this report.

Now we skim through an introduction of EM algorithm. Suppose there are two spaces ***X*** and ***Y***, in which ***X*** is *hidden space* whereas ***Y*** is *observed space*. We do not know ***X*** but there is a mapping from ***X*** to ***Y*** so that we can survey ***X***by observing ***Y***. The mapping is many-one function *φ*: ***X*** → ***Y*** and we denote *φ*–1(*Y*) = {: *φ*(*X*) = *Y*} as all such that *φ*(*X*) = *Y*. We also denote ***X***(*Y*) = *φ*–1(*Y*). Let *f*(*X* | Θ) be the probability density function (PDF) of random variable and let *g*(*Y* | Θ) be the PDF of random variable . Note, *Y* is also called observation. Equation 1.1 specifies *g*(*Y* | Θ) as integral of *f*(*X* | Θ) over *φ*–1(*Y*).

|  |  |
| --- | --- |
|  | (1.1) |

Where Θ is probabilistic parameter represented as a column vector, Θ= (*θ*1, *θ*2,…, *θr*)*T* in which each *θi* is a particular parameter. If *X* and *Y* are discrete, equation 1.1 is re-written as follows:

According to viewpoint of Bayesian statistics, Θ is also random variable. As a convention, let Ω be the domain of Θ such that and the dimension of Ω is *r*. For example, normal distribution has two particular parameters such as mean *μ* and variance *σ*2 and so we have Θ= (*μ*, *σ*2)*T*. Note that, Θ can degrades into a scalar as Θ = *θ*. The conditional PDF of *X* given *Y*, denoted *k*(*X* | *Y*, Θ), is specified by equation 1.2.

|  |  |
| --- | --- |
|  | (1.2) |

According to DLR (Dempster, Laird, & Rubin, 1977, p. 1), ***X*** is called *complete data* and the term “incomplete data” implies existence of ***X*** and ***Y*** where ***X*** is not observed directly and ***X*** is only known by the many-one mapping *φ*: ***X*** → ***Y***. In general, we only know ***Y***, *f*(*X* | Θ), and *k*(*X* | *Y*, Θ) and so our purpose is to estimate Θ based on such ***Y***, *f*(*X* | Θ), and *k*(*X* | *Y*, Θ). Like MLE approach, EM algorithm also maximizes the likelihood function to estimate Θ but the likelihood function in EM concerns ***Y*** and there are also some different aspects in EM which will be described later. Pioneers in EM algorithm firstly assumed that *f*(*X* | Θ) belongs to exponential family with note that many popular distributions such as normal, multinomial, and Poisson belong to exponential family. Although DLR (Dempster, Laird, & Rubin, 1977) proposed a generality of EM algorithm in which *f*(*X* | Θ) distributes arbitrarily, we should concern exponential family a little bit. Exponential family (Wikipedia, Exponential family, 2016) refers to a set of probabilistic distributions whose PDF (s) have the same exponential form according to equation 1.3 (Dempster, Laird, & Rubin, 1977, p. 3):

|  |  |
| --- | --- |
|  | (1.3) |

Where *b*(*X*) is a function of *X*, which is called base measure and *τ*(*X*) is a vector function of *X*, which is sufficient statistic. For example, the sufficient statistic of normal distribution is *τ*(*X*) = (*X*, *XXT*)*T*. Equation 1.3 expresses the canonical form of exponential family. Recall that Ω is the domain of Θ such that . Suppose that Ω is a convex set. If Θ is restricted only to Ω then, *f*(*X* | Θ) specifies a *regular exponential family*. If Θ lies in a curved sub-manifold Ω0 of Ω then, *f*(*X* | Θ) specifies a *curved exponential family*. The *a*(Θ) is *partition function* for variable *X*, which is used for normalization.

As usual, a PDF is known as a popular form but its exponential family form (canonical form of exponential family) specified by equation 1.3 looks unlike popular form although they are the same. Therefore, parameter in popular form is different from parameter in exponential family form.

For example, multinormal distribution with theoretical mean *μ* and covariance matrix Σ of random variable *X* = (*x*1, *x*2,…, *xn*)*T* has PDF in popular form is:

Hence, parameter in popular form is Θ = (*μ*, Σ)*T*. Exponential family form of such PDF is:

Where,

The exponential family form is used to represents all distributions belonging to exponential family as canonical form. Parameter in exponential family form is called exponential family parameter. As a convention, parameter Θ mentioned in EM algorithm is often exponential family parameter if PDF belongs to exponential family and there is no additional information.

Expectation maximization (EM) algorithm has many iterations and each iteration has two steps in which expectation step (E-step) calculates sufficient statistic of hidden data based on observed data and current parameter whereas maximization step (M-step) re-estimates parameter. When DLR proposed EM algorithm (Dempster, Laird, & Rubin, 1977), they firstly concerned that the PDF *f*(*X* | Θ) of hidden space belongs to exponential family. E-step and M-step at the *t*th iteration are described in table 1.1 (Dempster, Laird, & Rubin, 1977, p. 4), in which the current estimate is Θ(*t*), with note that *f*(*X* | Θ) belongs to regular exponential family.

|  |
| --- |
| *E-step*:  We calculate current value *τ*(*t*) of the sufficient statistic *τ*(*X*) from observed *Y* and current parameter Θ(*t*) according to following equation:  *M-step*:  Basing on *τ*(*t*), we determine the next parameter Θ(*t*+1) as solution of following equation:  Note, Θ(*t*+1) will become current parameter at the next iteration ((*t*+1)th iteration). |

**Table 1.1.** E-step and M-step of EM algorithm given regular exponential PDF *f*(*X*|Θ)

EM algorithm stops if two successive estimates are equal, Θ*\** = Θ(*t*) = Θ(*t*+1), at some *t*th iteration. At that time we conclude that Θ*\** is the optimal estimate of EM process. As a convention, the estimate of parameter Θ resulted from EM process is denoted Θ\* instead of in order to emphasize that Θ\* is solution of optimization problem.

For further research, DLR gave a preeminent generality of EM algorithm (Dempster, Laird, & Rubin, 1977, pp. 6-11) in which *f*(*X* | Θ) specifies arbitrary distribution. In other words, there is no requirement of exponential family. They define the conditional expectation *Q*(Θ’ | Θ) according to equation 1.4 (Dempster, Laird, & Rubin, 1977, p. 6).

|  |  |
| --- | --- |
|  | (1.4) |

If *X* and *Y* are discrete, equation 2.4 can be re-written as follows:

The two steps of generalized EM (*GEM*) algorithm aim to maximize *Q*(Θ | Θ(*t*)) at some *t*th iteration as seen in table 1.2 (Dempster, Laird, & Rubin, 1977, p. 6).

|  |
| --- |
| *E-step*:  The expectation *Q*(Θ | Θ(*t*)) is determined based on current parameter Θ(*t*), according to equation 1.4. Actually, *Q*(Θ | Θ(*t*)) is formulated as function of Θ.  *M-step*:  The next parameter Θ(*t*+1) is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ. Note that Θ(*t*+1) will become current parameter at the next iteration (the (*t*+1)th iteration). |

**Table 1.2.** E-step and M-step of GEM algorithm

DLR proved that GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the optimal estimate of EM process, which is an optimizer of *L*(Θ).

It is deduced from E-step and M-step that *Q*(Θ | Θ(*t*)) is increased after every iteration. How to maximize *Q*(Θ|Θ(*t*)) is the optimization problem which is dependent on applications. For example, the estimate Θ(*t*+1) can be solution of the equation created by setting the first-order derivative of *Q*(Θ|Θ(*t*)) regarding Θ to be zero, *DQ*(Θ|Θ(*t*)) = **0***T*. If solving such equation is too complex or impossible, some popular methods to solve optimization problem are Newton-Raphson (Burden & Faires, 2011, pp. 67-71), gradient descent (Ta, 2014), and Lagrange duality (Wikipedia, Karush–Kuhn–Tucker conditions, 2014).

In practice, if *Y* is observed as particular *N* observations *Y*1, *Y*2,…, *YN*. Let = {*Y*1, *Y*2,…, *YN*} be the observed sample of size *N* with note that all *Yi* (s) are mutually independent and identically distributed (iid). Given an observation *Yi*, there is an associated random variable *Xi*. All *Xi* (s) are iid and they are not existent in fact. Each is a random variable like *X*. Of course, the domain of each *Xi* is ***X***. Let = {*X*1, *X*2,…, *XN*} be the set of associated random variables. Because all *Xi* (s) are iid, the joint PDF of is determined as follows:

Because all *Xi* (s) are iid and each *Yi* is associated with *Xi*, the conditional joint PDF of given is determined as follows:

The conditional expectation *Q*(Θ’ | Θ) given samples ***X*** and ***Y*** is re-written according to equation 1.5.

|  |  |
| --- | --- |
|  | (1.5) |

Equation 1.5 is proved in (Nguyen, Tutorial on EM algorithm, 2020, pp. 45-47). In case that *f*(*X* | Θ) and *k*(*X* | *Yi*, Θ) belong to exponential family, equation 1.5 becomes equation 1.6 with an observed sample = {*Y*1, *Y*2,…, *YN*}.

|  |  |
| --- | --- |
|  | (1.6) |

Where,

DLR (Dempster, Laird, & Rubin, 1977, p. 1) called ***X*** as *complete data* because the mapping *φ*: ***X*** → ***Y*** is many-one function. There is another case that the complete space ***Z*** consists of hidden space ***X*** and observed space ***Y*** with note that ***X*** and ***Y*** are separated. There is no explicit mapping *φ* from ***X*** and ***Y*** but there exists a PDF of as the joint PDF of and .

The PDF of *Y* becomes:

The PDF *f*(*Y*|Θ) is equivalent to the PDF *g*(*Y*|Θ) mentioned in equation 1.1. Although there is no explicit mapping from ***X*** to ***Y***, the PDF of *Y* above implies an implicit mapping from ***Z*** to ***Y***. The conditional PDF of *X* given *Z* is specified according to Bayes’ rule as follows:

The conditional PDF *f*(*X*|*Y*, Θ) is equivalent to the conditional PDF *k*(*X*|*Y*, Θ) mentioned in equation 1.2. Of course, given *Y*, we always have:

Equation 1.7 specifies the conditional expectation *Q*(Θ’ | Θ) in case that there is no explicit mapping from ***X*** to ***Y*** but there exists the joint PDF of *X* and *Y*.

|  |  |
| --- | --- |
|  | (1.7) |

Where,

Note, ***X*** is separated from ***Y*** and the complete data ***Z*** = (***X***, ***Y***) is composed of ***X*** and ***Y***. For equation 1.7, the existence of the joint PDF *f*(*X*, *Y* | Θ) can be replaced by the existence of the conditional PDF *f*(*Y*|*X*, Θ) and the prior PDF *f*(*X*|Θ) due to:

In applied statistics, equation 1.4 is often replaced by equation 1.7 because specifying the joint PDF *f*(*X*, *Y* | Θ) is more practical than specifying the mapping *φ*: ***X*** → ***Y***. However, equation 1.4 is more general equation 1.7 because the requirement of the joint PDF for equation 1.7 is stricter than the requirement of the explicit mapping for equation 1.4. In case that *X* and *Y* are discrete, equation 1.7 becomes:

In practice, suppose *Y* is observed as a sample = {*Y*1, *Y*2,…, *YN*} of size *N* with note that all *Yi* (s) are mutually independent and identically distributed (iid). The observed sample is associated with a a hidden set (latent set) = {*X*1, *X*2,…, *XN*} of size *N*. All *Xi* (s) are iid and they are not existent in fact. Let be the random variable representing every *Xi*. Of course, the domain of *X* is ***X***. Equation 1.8 specifies the conditional expectation *Q*(Θ’ | Θ) given such .

|  |  |
| --- | --- |
|  | (1.8) |

Equation 1.8 is a variant of equation 1.5 in case that there is no explicit mapping between *Xi* and *Yi* but there exists the same joint PDF between *Xi* and *Yi*. If both *X* and *Y* are discrete, equation 1.8 becomes:

|  |  |
| --- | --- |
|  | (1.9) |

If *X* is discrete and *Y* is continuous such that *f*(*X*, *Y* | Θ) = *P*(*X*|Θ)*f*(*Y* | *X*, Θ) then, according to the total probability rule, we have:

Note, when only *X* is discrete, its PDF *f*(*X*|Θ) becomes the probability *P*(*X*|Θ). Therefore, equation 1.10 is a variant of equation 1.8, as follows:

|  |  |
| --- | --- |
|  | (1.10) |

Where *P*(*X* | *Yi*, Θ) is determined by Bayes’ rule, as follows:

Equation 1.10 is the base for estimating the probabilistic mixture model by EM algorithm, which is described in the next section.

**2. Mixture model**

As usual, let ***X*** be the hidden or latent space and let ***Y*** be the observed space. Especially, the random variable *X* in ***X*** represents latent class or latent component of random variable *Y* in ***Y***. Suppose *X* is discrete and ranges in ***X*** = {1, 2,…, *K*}. The so-called probabilisticfinite *mixture model* is represented by the PDF of *Y*, as seen in equation 2.1.

|  |  |
| --- | --- |
|  | (2.1) |

Where,

Note, *Y* can be discrete or continuous. Recall that the ultimate purpose of EM algorithm is to maximize *f*(*Y*|Θ) with subject to Θ. Each *fX*(*Y*|*θX*) is called the *X*th partial PDF of *Y* whose partial parameter is *θX*. Each *fX*(*Y*|*θX*) is also called the *X*th observational PDF of *Y*. It is really the conditional PDF of *Y* given *X*, as seen in equation 2.2.

|  |  |
| --- | --- |
|  | (2.2) |

From equation 2.1, the mixture model *f*(*Y*|Θ) is the mean of *K* partial PDFs. The variable *X* implies which partial PDF “generates” *Y* (Bilmes, 1998, p. 5).

Each *αX* is called mixture coefficient. It is really the probability of discrete *X*, as seen in equation 2.3. However, in mixture model, each *αX* is also considered as parameter, which is belongs to the compound parameter Θ.

|  |  |
| --- | --- |
|  | (2.3) |

The joint probabilistic distribution of *X* and *Y*, which implies the implicit mapping between ***X*** and ***Y***, is product of the mixture coefficient *αX* and the *X*th PDF of *Y*, as seen in equation 2.4.

|  |  |
| --- | --- |
|  | (2.4) |

This implies:

|  |  |
| --- | --- |
|  | (2.5) |

Equation 2.6 specifies the conditional probability of *X* given *Y*. Please pay attention to this important probability.

|  |  |
| --- | --- |
|  | (2.6) |

Following is the proof of equation 2.6. According to Bayes’ rule, we have:

Applying equation 2.3 and equation 2.4, we have:

In other words, equation 2.6 is established■

Now GEM algorithm is applied into mixture model for estimating the parameter Θ. Derived from equation 1.7, the conditional expectation *Q*(Θ’|Θ) of mixture model becomes:

|  |  |
| --- | --- |
|  | (2.7) |

In practice, suppose *Y* is observed as a sample = {*Y*1, *Y*2,…, *YN*} of size *N* in which all *Yi* (s) are mutually independent and identically distributed (iid). The observed sample is associated with a a hidden set (latent set) = {*X*1, *X*2,…, *XN*} of size *N*. All *Xi* (s) are iid and they are not existent in fact. Let be the random variable representing every *Xi*. Of course, the domain of *X* is ***X***. Derived from equation 1.10, equation 2.8 specifies *Q*(Θ’|Θ) given such .

|  |  |
| --- | --- |
|  | (2.8) |

Equation 2.8 is the general case of equation 2.7. At the *t*th iteration of GEM, given current parameter Θ(*t*) = (*α*1(*t*), *α*2(*t*),…, *αK*(*t*), *θ*1(*t*), *θ*2(*t*),…, *θK*(*t*))*T*, the conditional expectation specified by equation 2.8 is written as follows:

Thus, the unknown of *Q*(Θ|Θ(*t*)) is Θ = (*α*1, *α*2,…, *αK*, *θ*1, *θ*2,…, *θK*)*T*. Because *X* is discrete and ranges in {1, 2,…, *K*}, the conditional expectation *Q*(Θ|Θ(*t*)) is re-written as equation 2.9 for convenience.

|  |  |
| --- | --- |
|  | (2.9) |

Where the conditional probability *P*(*k* | *Y*, Θ(*t*)) is determined by equation 2.10 which is indeed equation 2.6.

|  |  |
| --- | --- |
|  | (2.10) |

At M-step of the current *t*th iteration, *Q*(Θ|Θ(*t*)) specified by equation 2.9 is maximized with subject to Θ. How to maximize *Q*(Θ|Θ(*t*)) with subject to Θ is dependent on types of partial PDFs *fk*(*Yi*|*θk*).

Because there is the constraint , we use Lagrange duality method to maximize to maximize *Q*(Θ|Θ(*t*)). The Lagrange function *la*(Θ, *λ* | Θ(*t*)) is sum of *Q*(Θ|Θ(*t*)) and the constraint , which is specified by equation 2.11.

|  |  |
| --- | --- |
|  | (2.11) |

Note, *λ* ≥ 0 is called Lagrange multiplier. Of course, *la*(Θ, *λ* | Θ(*t*)) is function of Θ and *λ*. The next parameters *αk*(*t*+1) that maximizes *Q*(Θ|Θ(*t*)) is solution of the equation formed by setting the first-order partial derivative of Lagrange function regarding *αk* and *λ* to be zero with suppose that the Lagrange function is first-order smooth function.

This implies:

|  |  |
| --- | --- |
|  | (2.12) |

Summing equation 2.12 over *K* classes {1, 2,…, *K*}, we have (Bilmes, 1998, p. 5):

Substituting *λ = N* into equation 2.12, the next parameters *αk*(*t*+1) is totally determined by equation 2.13.

|  |  |
| --- | --- |
|  | (2.13) |

Note, the conditional probability *P*(*k* | *Yi*, Θ(*t*)) is determined by equation 2.10.

When parameters *αk*(*t*+1) and *λ* are determined, the Lagrange function *la*(Θ, *λ* | Θ(*t*)) is now function of parameters *θk* as *la*(*θk*|*θk*(*t*)). The next parameters *θk*(*t*+1) is solution of the equation formed by setting the first-order partial derivative of Lagrange function regarding *θk* to be zero with suppose that the Lagrange function is first-order smooth function.

Thus, the next parameters *θk*(*t*+1) is solution of the equation 2.14.

|  |  |
| --- | --- |
|  | (2.14) |

The two steps of GEM algorithm for constructing mixture model at some *t*th iteration are shown in table 2.1. Note, suppose the Lagrange function is first-order smooth function.

|  |
| --- |
| *E-step*:  The conditional probability *P*(*k* | *Yi*, Θ(*t*)) is calculated based on current parameter Θ(*t*) = (*α*1(*t*), *α*2(*t*),…, *αK*(*t*), *θ*1(*t*), *θ*2(*t*),…, *θK*(*t*))*T*, according to equation 2.10.  *M-step*:  The next parameter Θ(*t*+1) = (*α*1(*t*+1), *α*2(*t*+1),…, *αK*(*t*+1), *θ*1(*t*+1), *θ*2(*t*+1),…, *θK*(*t*+1))*T*, which is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ, is calculated by equation 2.13 and equation 2.14. Note, *θk*(*t*+1) is solution of the equation 2.14. |

**Table 2.1.** E-step and M-step of GEM algorithm for constructing mixture model regarding first-order smooth Lagrange function

GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the optimal estimate of mixture model regarding first-order smooth Lagrange function.

Suppose that each PDF *fk*(*Yi*|*θk*) ) belongs to regular exponential family and then, solving equation 2.4 is easier as follows:

(Due to *fk*(*Yi*|*θk*) ) belongs to exponential family)

(Due to log’(*a*(*θk*)) = (*E*(*τ*(*Y*|*θk*)))*T*, please see table 1.2)

In general, the next parameters *θk*(*t*+1) is solution of the equation 2.15 within regular exponential family.

|  |  |
| --- | --- |
|  | (2.15) |

Where *Y* is the random variable representing all *Yi* (s) and,

The two steps of GEM algorithm for constructing mixture model at some *t*th iteration are shown in table 2.2 with suppose that each partial PDF *fX*(*Y*|*θX*) is assumed to belong regular exponential family.

|  |
| --- |
| *E-step*:  The conditional probability *P*(*k* | *Yi*, Θ(*t*)) is calculated based on current parameter Θ(*t*) = (*α*1(*t*), *α*2(*t*),…, *αK*(*t*), *θ*1(*t*), *θ*2(*t*),…, *θK*(*t*))*T*, according to equation 2.10.  *M-step*:  The next parameter Θ(*t*+1) = (*α*1(*t*+1), *α*2(*t*+1),…, *αK*(*t*+1), *θ*1(*t*+1), *θ*2(*t*+1),…, *θK*(*t*+1))*T*, which is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ, is calculated by equation 2.13 and equation 2.15. Note, *θk*(*t*+1) is solution of the equation 2.15. |

**Table 2.2.** E-step and M-step of GEM algorithm for constructing mixture model regarding regular exponential family

GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the optimal estimate of mixture model regarding regular exponential family.

There is a special case that each *fk*(*Yi*|*θk*) is normal distribution, which is popular in domain of mixture model, with note that normal distribution belongs to regular exponential family. Thus, let *Y* be random variable representing all *Yi*. Without loss of generality, suppose *Y* is vector so that each *fk*(*Y*|*θk*) is multinormal distribution. Recall that each *fk*(*Y*|*θk*) is called the *k*th partial PDF of *Y* or the *k*th observational PDF of *Y*. In this case, the mixture model is called *normal mixture model* (Gaussian mixture model) and it is easy to solve equation 2.14 or equation 2.15 for *θk*. Suppose random variable *Y* is vector of size *n*.

|  |  |
| --- | --- |
|  | (2.16) |

Where *μk* and Σ*k* are mean vector and covariance matrix of *fk*(*Y*|*θk*), respectively. The notation |.| denotes determinant of given matrix and the notation Σ*k*–1 denotes inverse of matrix Σ*k*. Note, Σ*k* is invertible and symmetric. Now we find other parameters *θk*(*t*+1) = (*μk*(*t*+1), Σ*k*(*t*+1))*T* by solving directly equation 2.14 or equation 2.15. Recall that each *Yi* conforms to multinormal distribution, according to equation 2.16.

Where *μk* and Σ*k* are mean and covariance matrix of *fk*(*Yi*|*θk*), respectively. The Lagrange function is re-written as follows:

Where *p* is the dimension of *Yi*; in other words, *p* is the dimension of space ***Y***.

The first-order partial derivative of Lagrange function with respect to *μk* is (Nguyen, 2015, p. 35):

The next parameter *μk*(*t*+1) that maximizes *Q*(Θ|Θ(*t*)) is solution of the equation formed by setting the first-order partial derivative of Lagrange function with regard to *μk* to be **0***T*. Note that **0** = (0, 0,…, 0)*T* is zero vector.

This implies equation 2.17 to specify the next parameter *μk*(*t*+1).

|  |  |
| --- | --- |
|  | (2.17) |

Note, the conditional probability *P*(*k* | *Yi*, Θ(*t*)) is determined by equation 2.10.

The first-order partial derivative of Lagrange function with respect to Σ*k* is:

Due to:

And

Because Bilmes (Bilmes, 1998, p. 5) mentioned:

Where tr(*A*) is trace operator which takes sum of diagonal elements of matrix .

This implies (Nguyen, 2015, p. 45):

Where Σ*k* is symmetric and invertible matrix. Substituting the next parameter *μk*(*t*+1) specified by equation 2.16 into the first-order partial derivative of Lagrange function with respect to Σ*k*, we have:

The next parameter Σ*k*(*t*+1) that maximizes *Q*(Θ|Θ(*t*)) is the solution of equation formed by setting the first-order partial derivative of Lagrange function regarding Σ*k* to zero matrix. Let (**0**) denote zero matrix.

We have:

This implies equation 2.18 to specify the next parameter Σ*k*(*t*+1).

|  |  |
| --- | --- |
|  | (2.18) |

Note, the conditional probability *P*(*k* | *Yi*, Θ(*t*)) is determined by equation 2.10 and the next parameter *μk*(*t*+1) is specified by equation 2.17.

As a result, the solution *θk*(*t*+1) = (*μk*(*t*+1), Σ*k*(*t*+1))*T* of equation 2.14 or equation 2.15 is specified by equation 2.17 and equation 2.18 when each *fk*(*Y*|*θk*) is multinormal distribution within normal mixture model. The two steps of GEM algorithm for constructing normal mixture model at some *t*th iteration are refined in table 2.3 (Bilmes, 1998, p. 7).

|  |
| --- |
| *E-step*:  The conditional probability *P*(*k* | *Yi*, Θ(*t*)) is calculated based on current parameter Θ(*t*) = (*α*1(*t*), *α*2(*t*),…, *αK*(*t*), *θ*1(*t*), *θ*2(*t*),…, *θK*(*t*))*T*, according to equation 2.10. Note, in normal mixture model, each observational PDF *fk*(*Y*|*θk*) is (multivariate) normal distribution with mean vector *μk* and covariance matrix Σ*k* such that *θk* = (*μk*, Σ*k*)*T*.  *M-step*:  The next parameter Θ(*t*+1) = (*α*1(*t*+1), *α*2(*t*+1),…, *αK*(*t*+1), *θ*1(*t*+1), *θ*2(*t*+1),…, *θK*(*t*+1))*T*, which is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ, is calculated by equation 2.13, equation 2.17, and equation 2.18 with current parameter Θ(*t*). |

**Table 2.3.** E-step and M-step of GEM algorithm for constructing normal mixture model

GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the optimal estimate of normal mixture model.

An interesting application of finite mixture model is soft clustering. Traditional clustering methods assign a fixed cluster to every data point in sample, which means that every data point belongs exactly to one cluster. There are some popular (hard) clustering methods such as *K*-means and *K*-medoids (Han & Kamber, 2006, pp. 451-457). Soft clustering is more flexible when every data point belongs to more than one cluster and the degree of assignment is represented by a probability. Concretely, GEM algorithm for normal mixture model described in table 2.3 is applied into soft clustering. Given sample = {*Y*1, *Y*2,…, *YN*} of size *N* in which all *Yi* (s) are iid and each *Yi* is also called a data point, soft clustering partitions into *K* clusters and each cluster *k* is considered as hidden variable (*X* = 1, 2,…, *K*) and is represented by the aforementioned normal PDF *fk*(*Y*|*θk*)

Where *θk* = (*μk*, Σ*k*)*T* includes mean vector *μk* and covariance matrix Σ*k* of *fk*(*Y*|*θk*), respectively. Especially, *μk* is considered as centroid of cluster *k*. Given cluster *k*, the degree of assignment that a data point *Y* belonging to cluster *k* is specified by such *fk*(*Y*|*θk*). Therefore, GEM algorithm for normal mixture model is used to learn Θ = (*α*1, *α*2,…, *αK*, *θ*1, *θ*2,…, *θK*)*T*. The parameter *αk* indicates degree of popularity of cluster *k*, which can be considered as capacity or size of cluster *k*. It can be also considered as coverage ratio of cluster *k*. The higher the *αk* is, the larger the cluster *k* is. Essentially, soft clustering is to estimate *αk* and *θk* by GEM. Suppose after GEM results out the best estimate Θ\* = (*α*1\*, *α*2\*,…, *αK*\*, *θ*1\*, *θ*2\*,…, *θK*\*)*T*, it is required to determine to which cluster a new data point *Y* is more likely to belong. We calculate *K* joint probabilities *p*1 = *α*1\**f*1(*Y*|*θ*1\*), *p*2 = *α*2\**f*2(*Y*|*θ*2\*),…, and *pK* = *αK*\**fK*(*Y*|*θK*\*). Indeed, each *pk* is the joint probability of *Y* and cluster *k* that come together. Suppose some *pj* is maximum then, *Y* is more likely to belong cluster *j*.

Of course, the probability of each data point *Y* within soft clustering for *K* clusters is

But this probability *f*(*Y*|Θ) is not important. The most important task of GEM for soft clustering is to compute the estimate Θ\* = (*α*1\*, *α*2\*,…, *αK*\*, *θ*1\*, *θ*2\*,…, *θK*\*)*T* from sample in order to determine clusters because each cluster *k* is represented by a pair {*αk*\*, *θk*\*}.

**Example 2.1.** Given sample = {*Y*1, *Y*2, *Y*3, *Y*4}, we apply GEM for soft clustering into *K*=2 clusters.

|  |  |  |
| --- | --- | --- |
|  | *y*1 | *y*2 |
| *Y*1 | 0 | 0 |
| *Y*2 | 0 | 1 |
| *Y*3 | 2 | 0 |
| *Y*4 | 2 | 1 |

Of course, we have *Y*1 = (*y*11=0, *y*12=0)*T*, *Y*2 = (*y*21=0, *y*22=1)*T*, *Y*3 = (*y*31=2, *y*32=0)*T*, and *Y*4 = (*y*41=2, *y*42=1)*T*. The parameter Θ = (*α*1, *α*2, *θ*1, *θ*2)*T* is initialized as follows:

Note, it is easy to calculate normal PDF *fk*(*Y*|*θk*) with known *θk* = (*μk*, Σ*k*)*T*.

At the 1st iteration, E-step we have:

At the 1st iteration, M-step we have:

At the 2nd iteration, E-step we have:

At the 2nd iteration, M-step we have:

Therefore, GEM stops at the 2nd iteration with the estimate Θ(2) = Θ(3) = Θ\* = (*α*1\*, *α*2\*, *θ*1\*, *θ*2\*)*T*.

Given new data point *Y* = (0.5, 0.5)*T*, it is required to determine to which cluster *Y* is more likely to belong. We calculate *K* joint probabilities as follows:

Due to some *p*1=*p*2, the likelihood that *Y* belongs to such two clusters is equal ■

# **Conditional mixture model and its application for regression model**

Loc Nguyen

Loc Nguyen’s Academic Network, Vietnam

Email: ng\_phloc@yahoo.com

Homepage: www.locnguyen.net

**Abstract**

Expectation maximization (EM) algorithm is a powerful mathematical tool for estimating statistical parameter when data sample contains hidden part and observed part. EM is applied to learn finite mixture model in which the whole distribution of observed variable is average sum of partial distributions. Coverage ratio of every partial distribution is specified by the probability of hidden variable. An application of mixture model is soft clustering in which cluster is modeled by hidden variable whereas each data point can be assigned to more than one cluster and degree of such assignment is represented by the probability of hidden variable. However, such probability in traditional mixture model is simplified as a parameter, which can cause loss of valuable information. Therefore, in this research I propose a so-called conditional mixture model (CMM) in which the probability of hidden variable is modeled as a full probabilistic density function (PDF) that owns individual parameter. CMM aims to extend mixture model. I also propose an application of CMM which is called adaptive regression model (ARM). Traditional regression model is effective when data sample is scattered equally. If data points are grouped into clusters, regression model tries to learn a unified regression function which goes through all data points. Obviously, such unified function is not effective to evaluate response variable based on grouped data points. The concept “adaptive” of ARM means that ARM solves the ineffectiveness problem by selecting the best cluster of data points firstly and then evaluating response variable within such best cluster. In order words, ARM reduces estimation space of regression model so as to gain high accuracy in calculation.

**Keywords:** expectation maximization (EM) algorithm, finite mixture model, conditional mixture model, regression model, adaptive regression model (ARM).

**1. Introduction**

Suppose data has two parts such as hidden part *X* and observed part *Y* and we only know *Y*. A relationship between random variable *X* and random variable *Y* is specified by the joint probabilistic density function (PDF) denoted *f*(*X*, *Y* | Θ) where Θ is parameter. Given sample = {*Y*1, *Y*2,…, *YN*} whose all *Yi* (s) are mutually independent and identically distributed (iid), it is required to estimate Θ based on whereas *X* is unknown. Expectation maximization (EM) algorithm is applied to solve this problem when only is observed. EM has many iterations and each iteration has two steps such as expectation step (E-step) and maximization step (M-step). At some *t*th iteration, given current parameter Θ(*t*), the two steps are described as follows:

*E-step*:

The expectation *Q*(Θ | Θ(*t*)) is determined based on current parameter Θ(*t*), according to equation 1.1 (Nguyen, Tutorial on EM tutorial, 2020, p. 50).

|  |  |
| --- | --- |
|  | (1.1) |

*M-step*:

The next parameter Θ(*t*+1) is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ. Note that Θ(*t*+1) will become current parameter at the next iteration (the (*t*+1)th iteration).

EM algorithm will converge after some iterations, at that time we have the estimate Θ(*t*) = Θ(*t*+1) = Θ\*. Note, the estimate Θ\* is result of EM.

Especially, the random variable *X* represents latent class or latent component of random variable *Y*. Suppose *X* is discrete and ranges in {1, 2,…, *K*}. As a convention, let *k*=*X*. Note, because all *Yi* (s) are iid, let random variable *Y* represent every *Yi*. The so-called probabilisticfinitemixture model is represented by the PDF of *Y*, as follows:

|  |  |
| --- | --- |
|  | (1.2) |

Where,

Note, the superscript “*T*” denotes transpose operator for vector and matrix. The *Q*(Θ | Θ(*t*)) is re-defined for finite mixture model as follows (Nguyen, Tutorial on EM tutorial, 2020, p. 79):

|  |  |
| --- | --- |
|  | (1.3) |

Where,

|  |  |
| --- | --- |
|  | (1.4) |

If every *fk*(*Y*|*θ*k) distributes normally with mean *μk* and covariance matrix Σ*k* such that *θk* = (*μk*, Σ*k*)*T*, the next parameter Θ(*t*+1) is calculated at M-step of such *t*th iteration given current parameter Θ(*t*) as follows (Nguyen, Tutorial on EM tutorial, 2020, p. 85):

|  |  |
| --- | --- |
|  | (1.5) |

Note, the conditional probability *P*(*k* | *Yi*, Θ(*t*)) is calculated at E-step.

In the traditional finite mixture model, the parameter *αk* is essentially the parameter of hidden random variable *X* when *X* is discrete, *αk* = *P*(*X*=*k*). In other words, *P*(*X*) is “reduced” at most. There is a problem of how to define and learn finite mixture model when *P*(*X*) is still a full PDF which owns individual parameter. Such problem is solved by the definition of conditional mixture model (CMM) in the next section.

**2. Conditional mixture model**

Now let *W* and *Y* be two random variables and both of them are observed. I define the conditional PDF of *Y* given *W* as follows:

|  |  |
| --- | --- |
|  | (2.1) |

Where *gk*(*W*|*φk*) is the *k*th PDF of *W* which can be considered PDF of *X* for the *k*th component. Equation 2.1 specifies the so-call conditional mixture model (CMM) when random variable *Y* is dependent on another random variable *W*. It is possible to consider that the parameter *αk* in the traditional mixture model specified by equation 1.2 is:

It is deduced that hidden variable *X*=*k* in CMM is represented by *gk*(*W*|*φk*) with a full of necessary parameters *φk*. When the sum is considered as constant, we have:

Where the sign “” indicates proportion. The quasi-conditional PDF of *Y* given *W* is defined to be proportional to the conditional PDF of *Y* given *W* as follows:

|  |  |
| --- | --- |
|  | (2.2) |

Where the parameter of CMM is Θ = (*φ*1, *φ*2,…, *φK*, *θ*1, *θ*2,…, *θK*)*T*. Of course, we have:

Given sample = {*Z*1 = {*W*1, *Y*1}, *Z*2 = {*W*2, *Y*2},…, }, *ZN* = {*WN*, *YN*})} of size *N* in which all *Xi* (s) are iid and all *yi* (s) are iid, we need to learn CMM. Let *W* and *Y* represent every *Wi* and every *Yi*, respectively. When applying EM along with the quasi-conditional PDF to estimate Θ, the *Q*(Θ | Θ(*t*)) is re-defined as follows:

|  |  |
| --- | --- |
|  | (2.3) |

Where *P*(*k* | *Wi*, *Yi*) is determined according to Bayes’ rule,

|  |  |
| --- | --- |
|  | (2.4) |

We need to maximize *Q*(Θ | Θ(*t*)) at M-step of some *t*th iteration given current parameter Θ(*t*). Expectedly, the next parameter Θ(*t*+1) is solution of the equation created by setting the first-order derivative of *Q*(Θ | Θ(*t*)) with regard to Θ to be zero. The first-order partial derivatives of *Q*(Θ | Θ(*t*)) with regard to *φk* and *θk* are:

Thus, the next parameter Θ(*t*+1) is solution of the following equation:

|  |  |
| --- | --- |
|  | (2.5) |

How to solve the equation 2.5 depends on individual applications. The next section describes an application of CMM.

**3. Adaptive regression model**

Traditional regression model is effective when data sample is scattered equally. If data points are grouped into clusters with their nature, regression model tries to learn a unified regression function which goes through all data points. Obviously, such unified function is not effective to evaluate response variable based on grouped data points. Alternately, if it is possible to select a right cluster for evaluating response variable, the value of response variable will be more precise. Therefore, selective evaluation is the main idea of adaptive regression model (ARM). The main ideology of ARM to group sample into clusters and build respective regression functions for clusters in parallel. CMM is applied to solve this problem, in other words, ARM is an application of CMM. There may be other applications of CMM but here I focus on ARM.

Given a *n*-dimension random variable *W* = (*w*1, *w*2,…, *wn*)*T* which is called regressors, a linear regression function is defined as

|  |  |
| --- | --- |
|  | (3.1) |

Where *y* is the random variable called response variable and each *βj* is called regressive coefficient. According to linear regression model, *y* conforms multinormal distribution, as follows:

|  |  |
| --- | --- |
|  | (3.2) |

Where *β* = (*β*0, *β*1,…, *βn*)*T* is called regressive parameter of *f*(*y* | *W*, *β*, *σ*2). Therefore, mean and variance of *f*(*y* | *W*, *β*, *σ*2) are *βTX* and *σ*2, respectively. Note, *f*(*y* | *W*, *β*, *σ*2) is called regressive PDF of *y*. As a convention, we denote:

Given sample = {*Z*1 = {*W*1, *y*1}, *Z*2 = {*W*2, *y*2},…, }, *ZN* = {*WN*, *yN*})} of size *N* in which all *Xi* (s) are iid and all *yi* (s) are iid. Let *W* = (*w*1, *w*2,…, *wn*)*T* and *y* represent every *Wi* = (*wi*1, *wi*2,…, *win*) and every *yi*, respectively. Let ***W*** and ***y*** be a matrix and a vector extracted from as follows:

|  |  |
| --- | --- |
|  | (3.3) |

When applying EM to estimate Θ, by following equation 2.3, the *Q*(Θ | Θ(*t*)) for ARM is re-defined as follows:

|  |  |
| --- | --- |
|  | (3.4) |

Where,

|  |  |
| --- | --- |
|  | (3.5) |

Where the parameter of ARM is Θ = (*φ*1, *φ*2,…, *φK*, *θ*1, *θ*2,…, *θK*)*T* but each *φk* and each *θk* are resolved more complexly. The definition of *Q*(Θ | Θ(*t*)) implies that sample can be grouped into *K* clusters.

The function *fk*(*y* | *W*, *θk*) is the *k*th regressive PDF of *y*.

|  |  |
| --- | --- |
|  | (3.6) |

Obviously, we have:

|  |  |
| --- | --- |
|  | (3.7) |

For convenience, suppose the *k*th PDF of *W* denoted *gk*(*W*|*φk*) is multinormal PDF as follows:

|  |  |
| --- | --- |
|  | (3.8) |

Where,

We need to maximize *Q*(Θ | Θ(*t*)) at M-step of some *t*th iteration given current parameter Θ(*t*). Expectedly, the next parameter Θ(*t*+1) is solution of the equation created by setting the first-order derivative of *Q*(Θ | Θ(*t*)) with regard to Θ to be zero.

The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *βk* is:

By referring (Nguyen & Shafiq, Mixture Regression Model for Incomplete Data, 2018, pp. 11-13), the next parameter *βk*(*t*+1) is solution of the equation where **0** is zero vector, as follows:

|  |  |
| --- | --- |
|  | (3.9) |

Where,

|  |  |
| --- | --- |
|  | (3.10) |

And,

|  |  |
| --- | --- |
|  | (3.11) |

The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *σk*2 is:

The next parameter (*σk*2)(*t*+1) which is solution of the equation is:

|  |  |
| --- | --- |
|  | (3.12) |

Where *βk*(*t*+1) is specified in equation 3.9.

The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *μk* is:

The next parameter *μk*(*t*+1) which is solution of the equation is:

|  |  |
| --- | --- |
|  | (3.13) |

The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to Σ*k* is (Nguyen, Tutorial on EM tutorial, 2020, pp. 83-84):

The next parameter Σ*k*(*t*+1) which is solution of the equation where (**0**) is zero matrix is:

|  |  |
| --- | --- |
|  | (3.14) |

Where *μk*(*t*+1) is specified in equation 3.13.

In general, at some *t*th iteration, given current parameter Θ(*t*), the two steps of EM for ARM are described as follows:

*E-step*:

The conditional probability *P*(*k* | *Yi*, Θ(*t*)) is calculated based on current parameter Θ(*t*) = (*φ*1(*t*), *φ*2(*t*),…, *φK*(*t*), *θ*1(*t*), *θ*2(*t*),…, *θK*(*t*))*T*, according to equation 3.5.

Where,

*M-step*:

The next parameter Θ(*t*+1) = (*φ*1(*t*+1), *φ*2(*t*+1),…, *φK*(*t*+1), *θ*1(*t*+1), *θ*2(*t*+1),…, *θK*(*t*+1))*T*, which is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ, is calculated by equation 3.19, equation 3.12, , equation 3.13, and equation 3.14 with current parameter Θ(*t*).

Where ***U****k*(*t*) and *Vk*(*t*) are specified by equation 3.10 and equation 3.11, respectively.

EM algorithm will converge after some iterations, at that time we have the estimate Θ(*t*) = Θ(*t*+1) = Θ\* = (*φ*1\*, *φ*2\*,…, *φK*\*, *θ*1\*, *θ*2\*,…, *θK*\*)*T*. As a result, ARM is specified by the estimate Θ\*. It can be said that Θ\* is ARM. Given any data point *W*, ARM select the best cluster *v* whose PDF *gv*(*W* | *φv*\*) is maximal, as follows:

|  |  |
| --- | --- |
|  | (3.15) |

Then ARM evaluates the response variable *Y* given regressor *W* with regard to such best cluster *v* as follows:

|  |  |
| --- | --- |
|  | (3.16) |

Instead of selecting the best cluster for evaluation, ARM can make an average over *K* clusters for evaluating *Y* as follows:

|  |  |
| --- | --- |
|  | (3.17) |

In general, equation 3.16 is the main one used to evaluate the regression function because the concept “adaptive” implies that ARM selects the best cluster (adaptive cluster) for evaluation.

**4. Conclusions**

The main ideology of CMM is to improve competence of mixture model, in which the probability of hidden variable is turned back its original form of PDF with full of parameters. As a result, its application ARM takes advantages of such hidden parameters in order to select best group or best cluster for making prediction of response value. In order words, ARM reduces estimation space of regression model so as to gain high accuracy in calculation. However, a new problem raised for CMM as well as ARM is how to pre-define the number *K* of clusters or components when CMM currently set fixed *K*. In the future, I will research some methods (Hoshikawa, 2013, p. 5) to pre-define *K*. Alternately, CMM can be improved or modified so that the number of clusters is updated in runtime (Nguyen & Shafiq, Mixture Regression Model for Incomplete Data, 2018, p. 16); in other words, there is no pre-definition of *K* and so *K* is determined dynamically.

# **Learning dyadic data and predicting unaccomplished co-occurrent values by mixture model**

Loc Nguyen

Loc Nguyen’s Academic Network, Vietnam

Email: ng\_phloc@yahoo.com

Homepage: www.locnguyen.net

**Abstract**

Dyadic data which is also called co-occurrence data (COD) contains co-occurrences of objects. Searching for statistical models to represent dyadic data is necessary. Fortunately, finite mixture model is a solid statistical model to learn and make inference on dyadic data because mixture model is built smoothly and reliably by expectation maximization (EM) algorithm which is suitable to inherent spareness of dyadic data. This research summarizes mixture models for dyadic data. When each co-occurrence in dyadic data is associated with a value, there are many unaccomplished values because a lot of co-occurrences are inexistent. In this research, these unaccomplished values are estimated as mean (expectation) of random variable given partial probabilistic distributions inside dyadic mixture model.

**Keywords:** dyadic data, co-occurrence data, expectation maximization (EM) algorithm, mixture model.

**1. Introduction**

Suppose data has two parts such as hidden part *X* and observed part *Y* and we only know *Y*. A relationship between random variable *X* and random variable *Y* is specified by the joint probabilistic density function (PDF) denoted *f*(*X*, *Y* | Θ) where Θ is parameter. Given sample = {*Y*1, *Y*2,…, *YN*} whose all *Yi* (s) are mutually independent and identically distributed (iid), it is required to estimate Θ based on whereas *X* is unknown. Expectation maximization (EM) algorithm is applied to solve this problem when only is observed. EM has many iterations and each iteration has two steps such as expectation step (E-step) and maximization step (M-step). At some *t*th iteration, given current parameter Θ(*t*), the two steps are described as follows:

|  |  |  |
| --- | --- | --- |
| *E-step*:  The expectation *Q*(Θ | Θ(*t*)) is determined based on current parameter Θ(*t*), according to equation 1.1 (Nguyen, Tutorial on EM algorithm, 2020, p. 50).   |  |  | | --- | --- | |  | (1.1) |   *M-step*:  The next parameter Θ(*t*+1) is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ. Note that Θ(*t*+1) will become current parameter at the next iteration (the (*t*+1)th iteration). |

**Table 1.1.** E-step and M-step of EM algorithm

EM algorithm will converge after some iterations, at that time we have the estimate Θ(*t*) = Θ(*t*+1) = Θ\*. Note, the estimate Θ\* is result of EM. The EM algorithm shown in table 1.1 is also called general EM or GEM.

Especially, the random variable *X* represents latent class or latent component of random variable *Y*. Suppose *X* is discrete and ranges in {1, 2,…, *K*}. As a convention, let *k*=*X*. Note, because all *Yi* (s) are iid, let random variable *Y* represent every *Yi*. The so-called probabilisticfinitemixture model is represented by the PDF of *Y*, as follows:

|  |  |
| --- | --- |
|  | (1.2) |

Where,

Note, the superscript “*T*” denotes transpose operator for vector and matrix. The *Q*(Θ | Θ(*t*)) is re-defined for finite mixture model as follows (Nguyen, Tutorial on EM algorithm, 2020, p. 79):

|  |  |
| --- | --- |
|  | (1.3) |

Where,

|  |  |
| --- | --- |
|  | (1.4) |

An interesting application of finite mixture model is soft clustering. Traditional clustering methods assign a fixed cluster to every data point in sample, which means that every data point belongs exactly to one cluster. Soft clustering is more flexible when every data point belongs to more than one cluster and the degree of assignment is represented by a probability. It is easy to recognize that when mixture model is applied into soft clustering, latent class *k* represents a cluster.

Every observation in ordinary sample is univariate or multivariate but there is a case that ordinary sample becomes dyadic sample related to two sets of objects, which causes some modifications of mixture model. *Dyadic data* which is also called co-occurrence data (COD) contains co-occurrent events of objects. It is necessary to obtain statistical models to represent dyadic data and fortunately, finite mixture model is the one. Recall that EM is applied to learn mixture model. The next section focuses on mixture model for dyadic data.

**2. Mixture models for dyadic data**

Given two finite sets = {*x*1, *x*2,…, *xN*) and = {*y*1, *y*2,…, *yM*) with note that *xi* (s) and *yj* (s) represent -objects and -objects, respectively; exactly, they are names of objects. The numbers of -objects and -objects are =*N* and =*M*, respectively. For example, in information retrieval, *xi* (s) are documents and *yj* (s) are keywords. Hence, *xi* and *yj* are not evaluated as numbers. An observational pair (*xi*, *yj*) is called a *co-occurrence* of *xi* and *yj*. Dyadic data or COD contains these co-occurrences with note that a co-occurrence (*xi*, *yj*) can exist more than one time. So, each co-occurrence (*xi*, *yj*) is indexed by an index *r*. As a result, each co-occurrence is denoted by the triple (*xi*, *yj*, *r*) and we have (Hofmann & Puzicha, 1998, p. 1):

|  |  |
| --- | --- |
|  | (2.1) |

Where,

Of course, the size of is . As a convention, *xi*(*r*) and *yj*(*r*) indicate that -object and -object at the *r*th co-occurrence are *xi* and *yj*, respectively. Thus, the triplet (*xi*, *yj*, *r*) can be denoted as (*xi*(*r*), *yj*(*r*), *r*). For example, suppose = {*x*1, *x*2, *x*3) and = {*y*1, *y*2), and dyadic data of 4 co-occurrences, = {(*x*1, *y*1, 1), (*x*1, *y*1, 2), (*x*1, *y*2, 3), (*x*1, *y*1, 4)}, we observe that *x*1 and *y*1 occur together three times at *r*=1, *r*=2, and *r*=4 where as *x*1 and *y*2 occur together one time at *r*=3. In the first co-occurrence (*x*1, *y*1, 1), the notation *x*1(1) indicate that the -object at this co-occurrence is *x*1. In the third co-occurrence (*x*1, *y*2, 3), the notation *y*2(3) indicate that the -object at this co-occurrence is *y*2.

If each co-occurrence of *xi* and *yj* is associated with a value *z* (Hofmann, Puzicha, & Jordan, Learning from Dyadic Data, 1998, p. 1), the triple (*xi*, *yj*, *r*) becomes the quadruplet (*xi*, *yj*, *z*, *r*) which is called *valued co-occurrence* of *xi* and *yj*. The value *z* is called associative value or co-occurrent value. If *z* is value of a variable *Z* then, *Z* is called associative variable or co-occurrent variable. As a result, the sample is called *valued dyadic data* or valued COD. Note, *Z* can be univariate or multivariate (vector).

|  |  |
| --- | --- |
|  | (2.2) |

Where,

As a convention, *Z*(*r*) or *z*(*r*) indicates that the associative value at *r*th co-occurrence is *Z*=*z*. Thus, the quadruplet (*xi*, *yj*, *Z*, *r*) can be denoted as (*xi*(*r*), *yj*(*r*), *Z*(*r*), *r*). For example, suppose = {*x*1, *x*2, *x*3) and = {*y*1, *y*2), and dyadic sample of 4 co-occurrences, = {(*x*1, *y*1, 6, 1), (*x*1, *y*1, 8, 2), (*x*1, *y*2, 7, 3), (*x*1, *y*1, 9, 4)}, we observe that *x*1 and *y*1 occur together three times at *r*=1, *r*=2, and *r*=4 where as *x*1 and *y*2 occur together one time at *r*=3. Moreover, at *r*=1, *r*=2, *r*=3, and *r*=4, associative values are *Z*(1)=6, *Z*(2)=7, *Z*(3)=8, and *Z*(4)=9, respectively. Valued dyadic data is special case of dyadic data. As a convention, dyadic data is default if there is no additional information.

Given fixed *xk*, let be the -partitioned subset of which contains co-occurrences whose -objects are fixed at *xk* (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 1). Note, can be empty. The size of is .

|  |  |
| --- | --- |
|  | (2.3) |

Dyadic data is partitioned into subsets .

Given fixed *yl*, let be the -partitioned subset of which contains co-occurrences whose -objects are fixed at *yl*. Note, can be empty. The size of is .

|  |  |
| --- | --- |
|  | (2.4) |

Dyadic data is partitioned into subsets .

Given fixed *xk* and fixed *yl*, let be the subset of the which contains co-occurrences whose -objects and -objects are fixed at *xk* and *yl*. Note, can be empty. The size of is .

|  |  |
| --- | --- |
|  | (2.5) |

Let *n*(*xi*) and *n*(*yj*) denote the number of *xi* and the number of *yj*, respectively.

|  |  |
| --- | --- |
|  | (2.6) |

Let *n*(*xi*, *yj*) denote the number of co-occurrences (*xi*, *yj*).

|  |  |
| --- | --- |
|  | (2.7) |

Let *n*(*xi*|*yj*) and *n*(*yj*|*xi*) denote the frequency of *xi* given *yj* and the frequency of *yj* given *xi*, respectively.

|  |  |
| --- | --- |
|  | (2.8) |

For example, suppose = {*x*1, *x*2, *x*3) and = {*y*1, *y*2), and dyadic data of 4 co-occurrences, = {(*x*1, *y*1, 1), (*x*1, *y*1, 2), (*x*1, *y*2, 3), (*x*1, *y*1, 4)}, we have = {(*x*1, *y*1, 1), (*x*1, *y*1, 2), (*x*1, *y*2, 3), (*x*1, *y*1, 4)}, = = Ø, = {(*x*1, *y*1, 1), (*x*1, *y*1, 2), (*x*1, *y*1, 4)}, = {(*x*1, *y*2, 3)}, = = {(*x*1, *y*1, 1), (*x*1, *y*1, 2), (*x*1, *y*1, 4)}, = {(*x*1, *y*2, 3)}, = = = = Ø, *n*(*x*1) = 1, *n*(*x*2) = *n*(*x*3) = 0, *n*(*y*1) = 3, *n*(*y*2) = 1, *n*(*x*1, *y*1) = 3, *n*(*x*1, *y*2) = 1, *n*(*x*2, *y*1) = *n*(*x*2, *y*2) = *n*(*x*3, *y*1) = *n*(*x*3, *y*2) = 0, *n*(*x*1 | *y*1) = 1, *n*(*x*1 | *y*2) = 1, *n*(*x*2 | *y*1) = *n*(*x*2 | *y*2) = *n*(*x*3 | *y*1) = *n*(*x*3 | *y*2) = 0, *n*(*y*1 | *x*1) = 3/4, *n*(*y*2 | *x*1) = 1/4.

Suppose each co-occurrence (*xi*, *yj*) belongs to a latent variable *C* and *C* has *K* values *ck* (s). These values *ck* (s) are called classes or aspects and thus, mixture model for dyadic data is also called aspect model or latent class model which aims to discover the latent variable *C*. Without loss of generality, let *ck* = *k* where *k* = 1, 2,…, *K*. The random variable *C* has discrete distribution such that every value has an associated probability *αk*. Of course, there are *K* probabilities *αk* (s). There are three kinds of dyadic mixture model for dyadic data such as symmetric mixture model (SMM), asymmetric mixture model (AMM), and product-space mixture model (PMM). This section only explains these models when they were introduced by Hofmann and Puzicha (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998).

The mixture model of dyadic data is called symmetric mixture model (SMM) if *αk* (s) are independent from both *xi* and *yj*. SMM is defined as follows (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 2):

|  |  |
| --- | --- |
|  | (2.9) |

Where *αk* is the probability of aspect *k*. Note, *P*(.) denote probability.

The is the probability of *xi* given aspect *k*.

The is the probability of *yj* given aspect *k*.

This implies that *xi* and *yj* are mutually independent in SMM.

The joint probability of *xi*, *yj*, and *k* is:

The parameter of SMM is Θ = (*αk*, *pi*|*k*, *qj*|*k*)*T* in which there are *K*( + + 1) partial parameters *αk*, *pi*|*k*, and *qj*|*k*. Note,

By applying GEM, given dyadic sample , at the *t*th iteration of GEM, given current parameter Θ(*t*) = (*αk*(*t*), *pi*|*k*(*t*), *qj*|*k*(*t*))*T*, the conditional expectation *Q*(Θ|Θ(*t*)) is:

|  |  |
| --- | --- |
|  | (2.10) |

Where,

|  |  |
| --- | --- |
|  | (2.11) |

Note, *n*(*xi*, *yj*) is the number of co-occurrences (*xi*, *yj*) in , which is specified by equation 2.7. Please refer to equation 1.4 to comprehend equation 2.11. Because there are three constraints

We use Lagrange duality method to maximize to maximize *Q*(Θ|Θ(*t*)). The Lagrange function *la*(Θ, *λ* | Θ(*t*)) is sum of *Q*(Θ|Θ(*t*)) and these constraints, as follows:

Note, *λ* = (*λ*1, *λ*2, *λ*3)*T* where *λ*1≥0, *λ*2≥0, and *λ*3≥0 are called Lagrange multipliers. Of course, *la*(Θ, *λ* | Θ(*t*)) is function of Θ and *λ*. The next parameters Θ(*t*+1) that maximizes *Q*(Θ|Θ(*t*)) at M-step of some *t*th iteration is solution of the equation formed by setting the first-order partial derivatives of Lagrange function regarding Θ and *λ* to be zero.

The first-order partial derivative of Lagrange function regarding *αk* is:

Setting this partial derivative to be zero, we obtain:

Summing the equation above over *K* aspects {1, 2,…, *K*}, we have:

This means the next parameters *αk*(*t*+1) is:

|  |  |
| --- | --- |
|  | (2.12) |

The first-order partial derivative of Lagrange function regarding *pi*|*k* is:

Setting this partial derivative to be zero, we obtain:

Summing the equation above over , we have:

This means the next parameters *pi*|*k*(*t*+1) is:

|  |  |
| --- | --- |
|  | (2.13) |

Similarly, the next parameters *qj*|*k*(*t*+1) is:

|  |  |
| --- | --- |
|  | (2.14) |

The two steps of GEM algorithm for SMM at some *t*th iteration are shown in table 2.1.

|  |
| --- |
| *E-step*:  The conditional probability *P*(*k* | *xi*, *yj*, Θ(*t*)) is calculated based on current parameter Θ(*t*) = (*αk*(*t*), *pi*|*k*(*t*), *qj*|*k*(*t*))*T*, according to equation 2.11.  *M-step*:  The next parameter Θ(*t*+1) = (*αk*(*t*+1), *pi*|*k*(*t*+1), *qj*|*k*(*t*+1))*T*, which is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ, is calculated by equation 2.12, equation 2.13, and equation 2.14. |

**Table 2.1.** E-step and M-step of GEM algorithm for SMM

GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the SMM itself. When SMM is applied into soft clustering, dyadic data is clustered according to blocks and each *αk* is coverage ratio of cluster *k* (aspect *k*).

The mixture model of dyadic data is called asymmetric mixture model (AMM) if *αk* (s) are only independent from *xi* or from *yj*. Without loss of generality, given *αk* (s) are only independent from *yj* (of course, it is dependent on *xi*), AMM is defined as follows (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 3):

|  |  |
| --- | --- |
|  | (2.15) |

The *αk*|*i* is the probability of aspect *k* given *xi*.

Where *pi* is the probability of *xi*.

The *qj*|*k* is the conditional probability of *yj* given aspect *k*. Suppose *yj* is dependent from *xi* given *k*, we have:

Note, *qj*|*i* is the conditional probability of *yj* given *xi*, which is defined as follows:

The joint probability of *xi*, *yj*, and *k* is:

The parameter of AMM is Θ = (*αk*|*i*, *pi*, *qj*|*k*)*T* in which there are *K*( + ) + partial parameters *αk*|*i*, *pi*, and *qj*|*k*. Note,

By applying GEM, given dyadic sample , at the *t*th iteration of GEM, given current parameter Θ(*t*) = (*αk*(*t*), *pi*|*k*(*t*), *qj*|*k*(*t*))*T*, the conditional expectation *Q*(Θ|Θ(*t*)) is:

|  |  |
| --- | --- |
|  | (2.16) |

Where,

|  |  |
| --- | --- |
|  | (2.17) |

Please refer to equation 1.4 to comprehend equation 2.17. Because there are three constraints

We use Lagrange duality method to maximize to maximize *Q*(Θ|Θ(*t*)). The Lagrange function *la*(Θ, *λ* | Θ(*t*)) is sum of *Q*(Θ|Θ(*t*)) and these constraints, as follows:

Note, *λ* = (*λ*1, *λ*2, *λ*3)*T* where *λ*1≥0, *λ*2≥0, and *λ*3≥0 are called Lagrange multipliers. Of course, *la*(Θ, *λ* | Θ(*t*)) is function of Θ and *λ*. The next parameters Θ(*t*+1) that maximizes *Q*(Θ|Θ(*t*)) at M-step of some *t*th iteration is solution of the equation formed by setting the first-order partial derivatives of Lagrange function regarding Θ and *λ* to be zero.

The first-order partial derivative of Lagrange function regarding *αk*|*i* is:

Setting this partial derivative to be zero, we obtain:

Summing the equation above over *K* aspects {1, 2,…, *K*}, we have:

This means the next parameters *αk*|*i*(*t*+1) is:

|  |  |
| --- | --- |
|  | (2.18) |

The first-order partial derivative of Lagrange function regarding *pi* is:

Setting this partial derivative to be zero, we obtain:

Summing the equation above over , we have:

This means the next parameters *pi*|*k*(*t*+1) is:

|  |  |
| --- | --- |
|  | (2.19) |

The first-order partial derivative of Lagrange function regarding *qj*|*k* is:

Setting this partial derivative to be zero, we obtain:

Summing the equation above over , we have:

This means the next parameters *qj*|*k*(*t*+1) is:

|  |  |
| --- | --- |
|  | (2.20) |

The two steps of GEM algorithm for AMM at some *t*th iteration are shown in table 2.2.

|  |
| --- |
| *E-step*:  The conditional probability *P*(*k* | *xi*, *yj*, Θ(*t*)) is calculated based on current parameter Θ(*t*) = (*αk*|*i*(*t*), *pi*(*t*), *qj*|*k*(*t*))*T*, according to equation 2.17.  *M-step*:  The next parameter Θ(*t*+1) = (*αk*|*i*(*t*+1), *pi*(*t*+1), *qj*|*k*(*t*+1))*T*, which is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ, is calculated by equation 2.18, equation 2.19, and equation 2.20. |

**Table 2.2.** E-step and M-step of GEM algorithm for AMM

GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the AMM itself. When AMM is applied into soft clustering, dyadic data is clustered vertically (horizontally) and each *αk*|*i* is coverage ratio of cluster *k* (aspect *k*) according to *xi*. Soft clustering with AMM is also called one-side clustering.

Product-space mixture model (PMM) is derived from SMM with a minor change that the aspect set {1, 2,…, *K*} is Cartesian product of -aspect set {1, 2,…, } and -aspect set {1, 2,…, }. In other words, the aspect space is still symmetric but is checked (stripped) according to two directions and .

|  |  |
| --- | --- |
|  | (2.21) |

For every *k* belongs to {1, 2,…, *K*}, there always exists a respective pair: and . However, for each or each , there are many respective *k*.

|  |  |
| --- | --- |
|  | (2.22) |

The sign “” denotes correspondence. PMM is defined as follows (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 4):

|  |  |
| --- | --- |
|  | (2.23) |

As usual, *αk* is the probability of aspect *ck* but is the probability of *xi* given of *k* and is the probability of *yj* given of *k*.

The joint probability of *xi*, *yj*, and *k* is:

The parameter of PMM is Θ = (*αk*, , )*T* in which there are *K* + + partial parameters *αk*, , and . Note,

Learning PMM is like learning SMM and so it is not necessary to duplicate the expansion of *Q*(Θ|Θ(*t*)). The two steps of GEM algorithm for PMM at some *t*th iteration are shown in table 2.3.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *E-step*:  The conditional probabilities *P*(*k* | *xi*, *yj*, Θ(*t*)), *P*( | *xi*, *yj*, Θ(*t*)), and *P*( | *xi*, *yj*, Θ(*t*)) are calculated based on current parameter Θ(*t*) = , according to equation 2.24, equation 2.25, and equation 2.26.   |  |  | | --- | --- | |  | (2.24) | |  | (2.25) | |  | (2.26) |   Please refer to equation 1.4 to comprehend equation 2.24.  *M-step*:  The next parameter Θ(*t*+1) = , which is the maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ, is calculated by equation 2.27, equation 2.28, and equation 2.29.   |  |  | | --- | --- | |  | (2.27) | |  | (2.28) | |  | (2.29) | |

**Table 2.3.** E-step and M-step of GEM algorithm for PMM

GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the PMM itself. When PMM is applied into soft clustering, dyadic data is clustered in checked (stripped) and each *αk* is coverage ratio of cluster *k* (aspect *k*) but such cluster *k* corresponds to a pair of cluster and cluster . Soft clustering with PMM is also called two-side clustering.

**3. Predicting unaccomplished co-occurrent values**

This section is the main subject of this research in which some extensions of dyadic mixture models are used to predict unaccomplished values in valued dyadic data. When is valued dyadic data in which every co-occurrence (*xi*, *yj*) is associated with value *z* from random variable *Z* then, SMM is reformed as follows:

|  |  |
| --- | --- |
|  | (3.1) |

AMM is reformed as follows:

|  |  |
| --- | --- |
|  | (3.2) |

PMM is reformed as follows:

|  |  |
| --- | --- |
|  | (3.3) |

Where *fk*(*Z*|*φk*) is the *k*th PDF of *Z* corresponding to the aspect *k*, in which *φk* is parameter of *fk*(*Z*|*φk*). Of course, the parameter Θ now must include all *φk*. It is possible to consider that

Moreover, *Z* is only dependent on *k*.

Note, suppose *xi* and *yj* (as well as *yj* given *xi*) are independent from *Z* given aspect *k*, which is the hint to reform these models.

For example, within SMM, the joint PDF of *xi*, *yj*, *Z*, and *k* is:

Within AMM, the joint PDF of *xi*, *yj*, *Z*, and *k* is:

Within PMM, the joint PDF of *xi*, *yj*, *Z*, and *k* is:

Here it is only necessary to estimate *φk* because how to estimate other partial parameters was mentioned in section 2. By reforming the conditional expectation *Q*(Θ|Θ(*t*)), it is easy to find out that the next parameter *φk*(*t*+1) is solution of following equation:

|  |  |
| --- | --- |
|  | (3.4) |

Where *P*(*k* | *xi*(*r*), *yj*(*r*), Θ(*t*)) is specified by equation 2.11, equation 2.17, and equation 2.24 for SMM, AMM, and PMM, respectively. Especially, if *fk*(*Z*|*φk*) distributed normally, the next parameter *φk*(*t*+1) = (*μk*(*t*+1), Σ*k*(*t*+1))*T* containing mean *μk*(*t*+1) and covariance matrix Σ*k*(*t*+1) is calculated as follows:

|  |  |
| --- | --- |
|  | (3.5) |

Where *P*(*k* | *xi*(*r*), *yj*(*r*), Θ(*t*)) is specified by equation 2.11, equation 2.17, and equation 2.24 for SMM, AMM, and PMM, respectively. Please refer to (Nguyen, Tutorial on EM algorithm, 2020, pp. 83-84) to comprehend equation 3.5.

In valued dyadic sample , many co-occurrences (*xi*, *yj*) are not existent and thus, it is required to predict or estimate *Z* value of inexistent co-occurrence (*xi*, *yj*). This *Z* value is called unaccomplished co-occurrent value or unaccomplished associative value. A so-called expected co-occurrent (EC) method is used to estimate *Z*. Firstly, it is necessary to define the conditional PDF of *Z* given *xi* and *yj*. According to Bayes’ rule, we have:

|  |  |
| --- | --- |
|  | (3.6) |

Then, *Z* value of inexistent co-occurrence (*xi*, *yj*) is estimated by an estimate which is the expectation of *Z* given the conditional PDF *f*(*Z* | *xi*, *yj*, Θ), as follows:

|  |  |
| --- | --- |
|  | (3.7) |

In short, EC method is specified by equation 3.6 and equation 3.7. Now we expend the two equations for SMM, AMM, and PMM. The conditional PDF *f*(*Z* | *xi*, *yj*, Θ) of SMM is:

|  |  |
| --- | --- |
|  | (3.8) |

Following is the proof of equation 3.8.

Similarly, the conditional PDF *f*(*Z* | *xi*, *yj*, Θ) of AMM is:

|  |  |
| --- | --- |
|  | (3.9) |

The conditional PDF *f*(*Z* | *xi*, *yj*, Θ) of PMM is:

|  |  |
| --- | --- |
|  | (3.10) |

Obviously, equation 3.8, equation 3.9, and equation 3.10 are extensions of equation 3.6.

The estimate for SMM is:

|  |  |
| --- | --- |
|  | (3.11) |

The estimate for AMM is:

|  |  |
| --- | --- |
|  | (3.12) |

The estimate for PMM is:

|  |  |
| --- | --- |
|  | (3.13) |

Where *Ek*(*Z*|*φk*) is expectation of *Z* given the *k*th PDF of *Z*:

|  |  |
| --- | --- |
|  | (3.14) |

If *fk*(*Z*|*φk*) is multinormal PDF with mean *μk* and covariance matrix Σ*k* then, we have *Ek*(*Z*|*φk*) = *μk*. Note, equation 3.11, equation 3.12, and equation 3.13 are extensions of equation 3.7.

Hofmann’s research (Hofmann, Latent Semantic Models for Collaborative Filtering, 2004) is different from EC method when Hofmann assumed that *fk*(*Z*|*φk*) is dependent on both *k* and *xi* so that *fk*(*Z*|*φk*) is replaced by .

Hofmann also assumed that (Hofmann & Puzieha, Latent Class Models for Collaborative Filtering, 1999, p. 690)

The sign “” indicates the proportion. Therefore, according to Hofmann, the conditional PDF *f*(*Z* | *xi*, *yj*, Θ) was defined as follows:

|  |  |
| --- | --- |
|  | (3.15) |

The estimate is still calculated by equation 3.7 except that *f*(*Z* | *xi*, *yj*, Θ) was defined by equation 3.15. As a result, equation 3.15 is the real mixture model of Hofmann in (Hofmann, Latent Semantic Models for Collaborative Filtering, 2004) and then Hofmann applied EM algorithm to learn parameters *αk*, *qj*|*k*, and *φik*. Therefore, Hofmann’s mixture model in (Hofmann, Latent Semantic Models for Collaborative Filtering, 2004) is not mixture models of co-occurrences (*xi*, *yj*) specified by equation 2.9 (SMM), equation 2.15 (AMM), and 2.23 (PMM). Hofmann’s mixture model is appropriate to collaborative filtering.

**4. Conclusions**

Essentially, learning dyadic data with models such as SMM, AMM, and PMM is unsupervised learning and it is easy to apply these models into soft clustering. Predicting or estimating unaccomplished values is essential to make a weighted sum of centroids over all clusters. Currently, an unaccomplished value is estimated based on pre-knowledge of an existent pair of two objects (-object and -object). As a result, an estimate is fixed if the two objects are fixed. In future, I try to find out another method to take advantages of more than two existent objects with a set of values. Combination of dyadic mixture model and regression model is a candidate method but how to prove and explain it is still fuzzy problem.

# **Conditional mixture model for modeling attributed dyadic data**

Loc Nguyen

Loc Nguyen’s Academic Network, Vietnam

Email: ng\_phloc@yahoo.com

Homepage: www.locnguyen.net

**Abstract**

Dyadic data contains co-occurrences of objects, which is often modeled by finite mixture model which in turn is learned by expectation maximization (EM) algorithm. Objects in traditional dyadic data are identified by names, causing the drawback which is that it is impossible to extract implicit valuable knowledge under objects. In this research, I propose the so-called attributed dyadic data (ADD) in which each object has an informative attribute and each co-occurrence of two objects is associated with a value. ADD is flexible and covers most of structures / forms of dyadic data. Conditional mixture model (CMM), which is a variant of finite mixture model, is applied into learning ADD. Moreover, a significant feature of CMM is that any co-occurrence of two objects is based on some conditional variable. As a result, CMM can predict or estimate co-occurrent values based on regression model, which extends applications of ADD and CMM.

**Keywords:** dyadic data, co-occurrence data, attributed dyadic data (ADD), mixture model, conditional mixture model (CMM), regression model.

**1. Introduction to dyadic data and mixture model**

Suppose data has two parts such as hidden part *X* and observed part *Y* and we only know *Y*. A relationship between random variable *X* and random variable *Y* is specified by the joint probabilistic density function (PDF) denoted *f*(*X*, *Y* | Θ) where Θ is parameter. Given sample {*Y*1, *Y*2,…, *YN*} whose all *Yi* (s) are mutually independent and identically distributed (iid), it is required to estimate Θ based on such sample whereas *X* is unknown. Expectation maximization (EM) algorithm is applied to solve this problem when only *Yi* (s) are observed. EM has many iterations and each iteration has two steps such as expectation step (E-step) and maximization step (M-step). At some *t*th iteration, given current parameter Θ(*t*), the two steps are described as follows:

*E-step*:

The expectation *Q*(Θ | Θ(*t*)) is determined based on current parameter Θ(*t*), according to equation 1.1 (Nguyen, Tutorial on EM tutorial, 2020, p. 50).

|  |  |
| --- | --- |
|  | (1.1) |

*M-step*:

The next parameter Θ(*t*+1) is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ. Note that Θ(*t*+1) will become current parameter at the next iteration (the (*t*+1)th iteration).

EM algorithm will converge after some iterations, at that time we have the estimate Θ(*t*) = Θ(*t*+1) = Θ\*. Note, the estimate Θ\* is result of EM.

Given two finite sets = {*x*1, *x*2,…, *xN*) and = {*y*1, *y*2,…, *yM*) with note that *xi* (s) and *yj* (s) represent -objects and -objects, respectively; exactly, they are names of objects. An observational pair (*xi*, *yj*) is called a *co-occurrence* of *xi* and *yj*. Dyadic data or co-occurrence data contains these co-occurrences with note that a co-occurrence (*xi*, *yj*) can exist more than one time. So, each co-occurrence (*xi*, *yj*) is indexed by an index *r*. As a result, each co-occurrence is denoted by the triple (*xi*, *yj*, *r*) and we have (Hofmann & Puzicha, 1998, p. 1):

|  |  |
| --- | --- |
|  | (1.2) |

Where,

Of course, the size of is . As a convention, *xi*(*r*) and *yj*(*r*) indicate that -object and -object at the *r*th co-occurrence are *xi* and *yj*, respectively. Thus, the triplet (*xi*, *yj*, *r*) can be denoted as (*xi*(*r*), *yj*(*r*), *r*).

Suppose each co-occurrence (*xi*, *yj*) belongs to a latent variable *C* and *C* has *K* values *ck* (s). These values *ck* (s) are called classes or aspects and thus, mixture model for dyadic data is also called aspect model or latent class model which aims to discover the latent variable *C*. Without loss of generality, let *ck* = *k* where *k* = 1, 2,…, *K*. The random variable *C* has discrete distribution such that every value has an associated probability *αk*. Of course, there are *K* probabilities *αk* (s). There are three kinds of dyadic mixture model (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 2) for dyadic data such as symmetric mixture model (SMM), asymmetric mixture model (AMM), and product-space mixture model (PMM). In this section, EM algorithm is applied to build up these mixture models.

SMM is defined as follows (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 2):

|  |  |
| --- | --- |
|  | (1.3) |

By applying EM, given dyadic sample , at some *t*th iteration, given current parameter Θ(*t*) = (*αk*(*t*), *pi*|*k*(*t*), *qj*|*k*(*t*))*T*, the next parameter Θ(*t*+1) = (*αk*(*t*+1), *pi*|*k*(*t*+1), *qj*|*k*(*t*+1))*T* of SMM is calculated at M-step as follows:

|  |  |
| --- | --- |
|  | (1.4) |

|  |  |
| --- | --- |
|  | (1.5) |

|  |  |
| --- | --- |
|  | (1.6) |

Where *n*(*xi*, *yj*) denotes the number of co-occurrences (*xi*, *yj*). The conditional probability *P*(*k* | *xi*, *yj*, Θ(*t*)) of SMM is calculated at E-step according to Bayes’ rule as follows:

|  |  |
| --- | --- |
|  | (1.7) |

The mixture model of dyadic data is called asymmetric mixture model (AMM) if *αk* (s) are only independent from *xi* or from *yj*. Without loss of generality, given *αk* (s) are only independent from *yj* (of course, it is dependent on *xi*), AMM is defined as follows (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 3):

|  |  |
| --- | --- |
|  | (1.8) |

By applying EM, given dyadic sample , at some *t*th iteration, given current parameter Θ(*t*) = (*αk*|*i*(*t*), *pi*(*t*), *qj*|*k*(*t*))*T*, the next parameter Θ(*t*+1) = (*αk*|*i*(*t*+1), *pi*(*t*+1), *qj*|*k*(*t*+1))*T* of AMM is calculated at M-step as follows:

|  |  |
| --- | --- |
|  | (1.9) |

|  |  |
| --- | --- |
|  | (1.10) |

|  |  |
| --- | --- |
|  | (1.11) |

Where the conditional probability *P*(*k* | *xi*, *yj*, Θ(*t*)) of AMM is calculated at E-step as follows:

|  |  |
| --- | --- |
|  | (1.12) |

Product-space mixture model (PMM) is derived from SMM with a minor change that the aspect set {1, 2,…, *K*} is Cartesian product of -aspect set {1, 2,…, } and -aspect set {1, 2,…, }. In other words, the aspect space is still symmetric but is checked (stripped) according to two directions and .

|  |  |
| --- | --- |
|  | (1.13) |

For every *k* belongs to {1, 2,…, *K*}, there always exists a respective pair: and . However, for each or each , there are many respective *k*.

|  |  |
| --- | --- |
|  | (1.14) |

The sign “” denotes correspondence. PMM is defined as follows (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 4):

|  |  |
| --- | --- |
|  | (1.15) |

By applying EM, given dyadic sample , at some *t*th iteration, given current parameter Θ(*t*) = , the next parameter Θ(*t*+1) = of PMM is calculated at M-step as follows:

|  |  |
| --- | --- |
|  | (1.16) |
|  | (1.17) |
|  | (1.18) |

Where (Nguyen, Learning Dyadic Data and Predicting Unaccomplished Co-Occurrent Values by Mixture Model, 2020, p. 10),

|  |  |
| --- | --- |
|  | (1.19) |
|  | (1.20) |
|  | (1.21) |

SMM, AMM, and PMM are defined in case of traditional dyadic data. The next section is main subject of this research where the so-called attributed dyadic data (ADD) is modeled by conditional mixture model (CMM).

**2. Learning attributed dyadic data by conditional mixture model**

In dyadic data , if each co-occurrence of *xi* and *yj* is associated with a value *z* (Hofmann, Puzicha, & Jordan, Learning from Dyadic Data, 1998, p. 1), the triple (*xi*, *yj*, *r*) becomes the quadruplet (*xi*, *yj*, *z*, *r*) which is called valued co-occurrence of *xi* and *yj*. The value *z* is called associative value or co-occurrent value. If *z* is value of a variable *Z* then, *Z* is called *associative variable*. As a result, the sample is called *valued dyadic data*. Note, *Z* can be univariate or multivariate (vector).

|  |  |
| --- | --- |
|  | (2.1) |

Where,

As a convention, *Z*(*r*) or *z*(*r*) indicates that the associative value at *r*th co-occurrence is *Z*=*z*. Thus, the quadruplet (*xi*, *yj*, *Z*, *r*) can be denoted as (*xi*(*r*), *yj*(*r*), *Z*(*r*), *r*).

An extension of valued dyadic data is called attributed dyadic data in every *xi* has an attribute *Xi* and every *yj* has an attribute *Yj* with constraint that all *Xi* (s) are iid and all *Yj* (s) are iid. Of course, these attributes are considered as random variables. Let *X* and *Y* be random variable representing every *Xi* and every *Yj*, respectively. Note, *X* and *Y* can be univariate or multivariate (vector), which are called *attribute variable*. As a result, the sample is called *attributed dyadic data* (ADD).

As a convention, *Xr* and *Yr* indicate that -object attribute and -object attribute at the *r*th co-occurrence, respectively whereas *Zr* indicates associative variable at the *r*th co-occurrence. The attributed dyadic data is represented as follows:

|  |  |
| --- | --- |
|  | (2.2) |

Thus, each co-occurrence in attributed dyadic data is denoted as a triplet (*Xr*, *Yr*, *Zr*). The -object and -object of *Xr* and *Yr* are denoted (*r*) and (*r*) which are some *xi* and *yj*, respectively. Here it is required to extends SMM, AMM, and PMM to represent ADD.

The joint PDF of -object attribute *X* = (*x*1, *x*2,…, *xn*)*T*, -object attribute *Y* = (*y*1, *y*2,…, *ym*)*T*, aspect *k*, and associative variable *Z* = (*z*1, *z*2,…, *za*)*T* given conditional variable *W* = (*w*1, *w*2,…, *wb*)*T* is:

Please distinguish partial random variable *xj* and *yj* here from object names *xi* and *yj*. Dimensions of *X*, *Y*, *Z*, and *W* are *n*, *m*, *a*, and *b*, respectively. We have:

(Suppose the proposed model is symmetric such that *f*(*X*, *Y* | *k*) = *f*(*X* | *k*)*f*(*Y* | *k*))

Thus, the joint PDF of -object attribute *X*, -object attribute *Y*, aspect *k*, and associative variable *Z* given conditional variable *W* is defined as follows:

|  |  |
| --- | --- |
|  | (2.3) |

Where,

Of course, *αk*, *βk*, *γk*, and *θk* are partial parameters of *fk*(*W*|*αk*), *gk*(*X*|*βk*), *hk*(*Y*|*γk*), and *vk*(*Z* | *W*, *θk*), respectively. These functions are PDFs. The whole parameter is Θ = (*αk*, *βk*, *γk*, *θk*)*T*. The PDF *fk*(*W*|*αk*) implies distribution of aspect *k* given conditional *W*. The two PDFs *gk*(*X*|*βk*) and *hk*(*Y*|*γk*) imply distributions of attributes with regard to -object, -object, and aspect *k*. The PDF *vk*(*Z* | *W*, *θk*) is conditional PDF of *Z* given *W* with regard to aspect *k*; later on we will know that it is more useful if it is considered as regression model.

According to Bayes’ rule, the conditional probability of *k* given -object attribute *X*, -object attribute *Y*, associative variable *Z*, and conditional variable *W* is:

|  |  |
| --- | --- |
|  | (2.4) |

Symmetric model (SMM) for attributed dyadic data is called symmetric attributed mixture model (SAMM), which is defined based on the joint PDF *f*(*X*, *Y*, *k*, *Z* | *W*, Θ) and *K* aspects {1, 2,…, *K*} as follows:

|  |  |
| --- | --- |
|  | (2.5) |

Recall that the parameter of SAMM is Θ = (*αk*, *βk*, *γk*, *θk*)*T*. Obviously, SAMM is an extension of conditional mixture model (CMM) (Nguyen, Conditional Mixture Model and Its Application for Regression Model, 2020) when *fk*(*W*|*αk*) and *vk*(*Z* | *W*, *θk*) are based on the condition *W*. SAMM is symmetric, which correspond to SMM. Here I focus on building up SAMM by mixture model and EM. Asymmetric model and product-space model for ADD will be mentioned later.

SAMM represented by equation 2.5 will be more specific and useful if *fk*(*W*|*αk*), *gk*(*X*|*βk*), and *hk*(*Y*|*γk*) are multinormal PDFs and *vk*(*Z* | *W*, *θk*) is regressive PDF which represents regression model. Means and covariance matrices of *fk*(*W*|*αk*), *gk*(*X*|*βk*), and *hk*(*Y*|*γk*) are *αk* = (*μαk*, Σ*αk*)*T*, *βk* = (*μβk*, Σ*βk*)*T*, and *γk* = (*μγk*, Σ*γk*)*T*.

|  |  |
| --- | --- |
|  | (2.6) |

|  |  |
| --- | --- |
|  | (2.7) |

|  |  |
| --- | --- |
|  | (2.8) |

Note, the superscript “*T*” denotes transposition operator for vector and matrix. The regressive PDF of *Z* given *W* denoted *vk*(*Z* | *W*, *θk*) is:

|  |  |
| --- | --- |
|  | (2.9) |

Where,

|  |  |
| --- | --- |
|  | (2.10) |

Note, *a* and *b* are dimensions of *Z* and *W*. Mean and covariance matrix of *Z* given *W* are *ωkW* and Σ*θk*, respectively. The partial parameter *ωk* is called regressive coefficient matrix, which is *a*x*b* matrix having *a* rows and *b* columns. Note, the product *ωkW* is:

|  |  |
| --- | --- |
|  | (2.11) |

The equation 2.11 also specifies multivariate regression function. Of course, equation 2.11 implies that:

By applying EM algorithm, given attributed dyadic sample , at the *t*th iteration of GEM, given current parameter Θ(*t*) = (*αk*(*t*), *βk*(*t*), *γk*(*t*), *θk*(*t*))*T*, the conditional expectation *Q*(Θ|Θ(*t*)) of SAMM specified by equation 2.5 is:

Note, all *Xr* (s) are iid represented by *X*, all *Yr* (s) are iid represented by *Y*, all *Zr* (s) are iid represented by *Z*, and all *Wr* (s) are iid represented by *W*. The -object and -object of *Xr* and *Yr* are denoted (*r*) and (*r*) which are some *xi* and *yj*, respectively. In short, we obtain:

|  |  |
| --- | --- |
|  | (2.12) |

Following equation 2.4, the conditional probability *P*(*k* | *Xr*, *Yr*, *Zr*, *Wr*, Θ(*t*)) is calculated at E-step as follows:

|  |  |
| --- | --- |
|  | (2.13) |

The next parameter Θ(*t*+1) = (*αk*(*t*+1), *βk*(*t*+1), *γk*(*t*+1), *θk*(*t*+1))*T* that maximizes *Q*(Θ|Θ(*t*)) at M-step of some *t*th iteration is solution of the equation formed by setting the first-order partial derivatives of *Q*(Θ|Θ(*t*)) regarding Θ = (*αk*, *βk*, *γk*, *θk*)*T* to be zero.Because *fk*(*W*|*αk*), *gk*(*X*|*βk*), and *hk*(*Y*|*γk*) distribute normally, by referring to (Nguyen, 2020, p. 2), the next parameters *αk*(*t*+1) = (*μαk*(*t*+1), Σ*αk*(*t*+1))*T*, *βk*(*t*+1) = (*μβk*(*t*+1), Σ*βk*(*t*+1))*T*, and *γk*(*t*+1) = (*μγk*(*t*+1), Σ*γk*(*t*+1))*T* for SAMM are calculated at M-step as follows:

|  |  |
| --- | --- |
|  | (2.14) |

|  |  |
| --- | --- |
|  | (2.15) |

|  |  |
| --- | --- |
|  | (2.16) |

Because *vk*(*Z* | *W*, *θk*) is regressive PDF known as adaptive regression model (ARM), by referring to (Nguyen, 2020, pp. 5-6), the next parameter *θk*(*t*+1) = (*ωk*(*t*+1), Σ*θk*(*t*+1))*T* is calculated as follows:

|  |  |
| --- | --- |
|  | (2.17) |

|  |  |
| --- | --- |
|  | (2.18) |

Where,

|  |  |
| --- | --- |
|  | (2.19) |

And,

|  |  |
| --- | --- |
|  | (2.20) |

Note, *a* and *b* are dimensions of *Z* and *W*. The product *ωk*(*t*+1)*W* is calculated by following equation 2.11 and equation 2.17. The conditional probability *P*(*k* | *Xr*, *Yr*, *Zr*, *Wr*, Θ(*t*)) is calculated at E-step according to equation 2.13. Moreover, *wrl* and *zrj* are extracted from *Xr* and *Zr* of where *Wr* = (*wr*1, *wr*2,…, *wrb*)*T* and *Zr* = (*zr*1, *zr*2,…, *zra*)*T*. Let,

|  |  |
| --- | --- |
|  | (2.21) |

Asymmetric model is not appropriate to ADD because the PDF of aspect *k*, *fk*(*W*|*αk*) becomes impractical if it depends on both *W* and *X* as *fk*(*W* | *X*, *αk*). However, product-space model (PMM) for ADD is still practical. PMM for attributed dyadic data is called product-space attributed mixture model (PAMM), which is defined as follows:

|  |  |
| --- | --- |
|  | (2.22) |

The parameter of PAMM is Θ = (*αk*, , , *θk*)*T* where and . Recall that the aspect set {1, 2,…, *K*} in product-space model is Cartesian product of -aspect set {1, 2,…, } and -aspect set {1, 2,…, }. For every *k* belongs to {1, 2,…, *K*}, there always exists a respective pair: and . However, for each or each , there are many respective *k*. Of course, PAMM is an extension of CMM.

At M-step of some *t*th iteration, the next parameters *αk*(*t*+1) = (*μαk*(*t*+1), Σ*αk*(*t*+1))*T* and *θk*(*t*+1) = (*ωk*(*t*+1), Σ*θk*(*t*+1))*T* for PAMM are as same as the ones for SAMM but the next parameters and for PAMM are estimated particularly as follows:

|  |  |
| --- | --- |
|  | (2.23) |

|  |  |
| --- | --- |
|  | (2.24) |

Where,

|  |  |
| --- | --- |
|  | (2.25) |
|  | (2.26) |

Of course, the conditional probability *P*(*k* | *Xr*, *Yr*, *Zr*, *Wr*, Θ(*t*)) is calculated at E-step according to equation 2.13. Please see equation 1.20 and equation 1.21 to understand equation 2.25 and equation 2.26.

CMM (s) for ADD such as SAMM and PAMM can be used to estimate an unknown associative value *Z* given attribute variables (*X*, *Y*) and conditional variable *W*. Let be the estimate of *Z*.

|  |  |
| --- | --- |
|  | (2.27) |

It is easy to calculate the conditional PDF of *Z* given *X*, *Y*, and *W* according to CMM as follows:

|  |  |
| --- | --- |
|  | (2.28) |

The estimate is resolved:

|  |  |
| --- | --- |
|  | (2.29) |

Where,

When *vk*(*Z* | *W*, *θk*) is a regressive PDF, the estimate given *X*, *Y*, and *W* is calculated smoothly as follows:

|  |  |
| --- | --- |
|  | (2.30) |

The product *ωkW* is calculated by following equation 2.11. In general, equation 2.30 is the ultimate estimation formula of CMM (s) for ADD.

**3. Conclusions**

ADD is flexible and covers most of structures / forms of dyadic data. CMM (s) for ADD such as SAMM and PAMM help researchers to model ADD in a flexible, solid, reliable manner. For example, if aspects are independent from conditional variable, each PDF *fk*(*W*|*αk*) is reduced into a discrete probabilistic parameter. If each regressive PDF *vk*(*Z* | *W*, *θk*) is not formed in favor of regression model, it must be formed as probabilistic distribution like conditional multinormal PDF. An interesting application of SAMM is to build up a unified estimation model of content-based filtering, collaborative filtering, and context-awarded filtering, in which attributes along with *gk*(*X*|*βk*) and *hk*(*Y*|*γk*) are responsible for content-based filtering whereas associative variable along with *vk*(*Z* | *W*, *θk*) and *fk*(*W*|*αk*) are responsible for collaborative filtering and context-awarded filtering. The attributes *X* and *Y* represent information about users and items in rating data with note that users and items are knowns as objects. The associative variable *Z* represents rating values in rating data. The conditional variable *W* represents contexts. Equation 2.30 is the ultimate formula of the unified estimation model. I hope that researchers will concern such proposed model because it is not realized yet when I compose this paper.

# **Handling missing data with expectation maximization algorithm**

Loc Nguyen

Loc Nguyen’s Academic Network, Vietnam

Email: ng\_phloc@yahoo.com

Homepage: www.locnguyen.net

**Abstract**

Expectation maximization (EM) algorithm is a powerful mathematical tool for estimating parameter of statistical models in case of incomplete data or hidden data. EM assumes that there is a relationship between hidden data and observed data, which can be a joint distribution or a mapping function. Therefore, this implies another implicit relationship between parameter estimation and data imputation. If missing data which contains missing values is considered as hidden data, it is very natural to handle missing data by EM algorithm. Handling missing data is not a new research but this report focuses on the theoretical base with detailed mathematical proofs for fulfilling missing values with EM. Besides, multinormal distribution and multinomial distribution are the two sample statistical models which are concerned to hold missing values.

**Keyworks:** expectation maximization (EM), missing data, multinormal distribution, multinomial distribution.

**1. Introduction to expectation maximization algorithm**

Literature of expectation maximization (EM) algorithm in this report is mainly extracted from the preeminent article “Maximum Likelihood from Incomplete Data via the EM Algorithm” by Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin (Dempster, Laird, & Rubin, 1977). For convenience, let DLR be reference to such three authors. The preprint “Tutorial on EM algorithm” (Nguyen, Tutorial on EM algorithm, 2020) by Loc Nguyen is also referred in this report.

Now we skim through an introduction of EM algorithm. Suppose there are two spaces ***X*** and ***Y***, in which ***X*** is *hidden space* whereas ***Y*** is *observed space*. We do not know ***X*** but there is a mapping from ***X*** to ***Y*** so that we can survey ***X***by observing ***Y***. The mapping is many-one function *φ*: ***X*** → ***Y*** and we denote *φ*–1(*Y*) = {: *φ*(*X*) = *Y*} as all such that *φ*(*X*) = *Y*. We also denote ***X***(*Y*) = *φ*–1(*Y*). Let *f*(*X* | Θ) be the probability density function (PDF) of random variable and let *g*(*Y* | Θ) be the PDF of random variable . Note, *Y* is also called observation. Equation 1.1 specifies *g*(*Y* | Θ) as integral of *f*(*X* | Θ) over *φ*–1(*Y*).

|  |  |
| --- | --- |
|  | (1.1) |

Where Θ is probabilistic parameter represented as a column vector, Θ= (*θ*1, *θ*2,…, *θr*)*T* in which each *θi* is a particular parameter. If *X* and *Y* are discrete, equation 1.1 is re-written as follows:

According to viewpoint of Bayesian statistics, Θ is also random variable. As a convention, let Ω be the domain of Θ such that and the dimension of Ω is *r*. For example, normal distribution has two particular parameters such as mean *μ* and variance *σ*2 and so we have Θ= (*μ*, *σ*2)*T*. Note that, Θ can degrades into a scalar as Θ = *θ*. The conditional PDF of *X* given *Y*, denoted *k*(*X* | *Y*, Θ), is specified by equation 1.2.

|  |  |
| --- | --- |
|  | (1.2) |

According to DLR (Dempster, Laird, & Rubin, 1977, p. 1), ***X*** is called *complete data* and the term “incomplete data” implies existence of ***X*** and ***Y*** where ***X*** is not observed directly and ***X*** is only known by the many-one mapping *φ*: ***X*** → ***Y***. In general, we only know ***Y***, *f*(*X* | Θ), and *k*(*X* | *Y*, Θ) and so our purpose is to estimate Θ based on such ***Y***, *f*(*X* | Θ), and *k*(*X* | *Y*, Θ). Like MLE approach, EM algorithm also maximizes the likelihood function to estimate Θ but the likelihood function in EM concerns ***Y*** and there are also some different aspects in EM which will be described later. Pioneers in EM algorithm firstly assumed that *f*(*X* | Θ) belongs to exponential family with note that many popular distributions such as normal, multinomial, and Poisson belong to exponential family. Although DLR (Dempster, Laird, & Rubin, 1977) proposed a generality of EM algorithm in which *f*(*X* | Θ) distributes arbitrarily, we should concern exponential family a little bit. Exponential family (Wikipedia, Exponential family, 2016) refers to a set of probabilistic distributions whose PDF (s) have the same exponential form according to equation 1.3 (Dempster, Laird, & Rubin, 1977, p. 3):

|  |  |
| --- | --- |
|  | (1.3) |

Where *b*(*X*) is a function of *X*, which is called base measure and *τ*(*X*) is a vector function of *X*, which is sufficient statistic. For example, the sufficient statistic of normal distribution is *τ*(*X*) = (*X*, *XXT*)*T*. Equation 1.3 expresses the canonical form of exponential family. Recall that Ω is the domain of Θ such that . Suppose that Ω is a convex set. If Θ is restricted only to Ω then, *f*(*X* | Θ) specifies a *regular exponential family*. If Θ lies in a curved sub-manifold Ω0 of Ω then, *f*(*X* | Θ) specifies a *curved exponential family*. The *a*(Θ) is *partition function* for variable *X*, which is used for normalization.

As usual, a PDF is known as a popular form but its exponential family form (canonical form of exponential family) specified by equation 1.3 looks unlike popular form although they are the same. Therefore, parameter in popular form is different from parameter in exponential family form.

For example, multinormal distribution with theoretical mean *μ* and covariance matrix Σ of random variable *X* = (*x*1, *x*2,…, *xn*)*T* has PDF in popular form is:

Hence, parameter in popular form is Θ = (*μ*, Σ)*T*. Exponential family form of such PDF is:

Where,

The exponential family form is used to represents all distributions belonging to exponential family as canonical form. Parameter in exponential family form is called exponential family parameter. As a convention, parameter Θ mentioned in EM algorithm is often exponential family parameter if PDF belongs to exponential family and there is no additional information.

Expectation maximization (EM) algorithm has many iterations and each iteration has two steps in which expectation step (E-step) calculates sufficient statistic of hidden data based on observed data and current parameter whereas maximization step (M-step) re-estimates parameter. When DLR proposed EM algorithm (Dempster, Laird, & Rubin, 1977), they firstly concerned that the PDF *f*(*X* | Θ) of hidden space belongs to exponential family. E-step and M-step at the *t*th iteration are described in table 1.1 (Dempster, Laird, & Rubin, 1977, p. 4), in which the current estimate is Θ(*t*), with note that *f*(*X* | Θ) belongs to regular exponential family.

|  |
| --- |
| *E-step*:  We calculate current value *τ*(*t*) of the sufficient statistic *τ*(*X*) from observed *Y* and current parameter Θ(*t*) according to following equation:  *M-step*:  Basing on *τ*(*t*), we determine the next parameter Θ(*t*+1) as solution of following equation:  Note, Θ(*t*+1) will become current parameter at the next iteration ((*t*+1)th iteration). |

**Table 1.1.** E-step and M-step of EM algorithm given regular exponential PDF *f*(*X*|Θ)

EM algorithm stops if two successive estimates are equal, Θ*\** = Θ(*t*) = Θ(*t*+1), at some *t*th iteration. At that time we conclude that Θ*\** is the optimal estimate of EM process. As a convention, the estimate of parameter Θ resulted from EM process is denoted Θ\* instead of in order to emphasize that Θ\* is solution of optimization problem.

For further research, DLR gave a preeminent generality of EM algorithm (Dempster, Laird, & Rubin, 1977, pp. 6-11) in which *f*(*X* | Θ) specifies arbitrary distribution. In other words, there is no requirement of exponential family. They define the conditional expectation *Q*(Θ’ | Θ) according to equation 1.4 (Dempster, Laird, & Rubin, 1977, p. 6).

|  |  |
| --- | --- |
|  | (1.4) |

If *X* and *Y* are discrete, equation 2.4 can be re-written as follows:

The two steps of generalized EM (*GEM*) algorithm aim to maximize *Q*(Θ | Θ(*t*)) at some *t*th iteration as seen in table 1.2 (Dempster, Laird, & Rubin, 1977, p. 6).

|  |
| --- |
| *E-step*:  The expectation *Q*(Θ | Θ(*t*)) is determined based on current parameter Θ(*t*), according to equation 1.4. Actually, *Q*(Θ | Θ(*t*)) is formulated as function of Θ.  *M-step*:  The next parameter Θ(*t*+1) is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ. Note that Θ(*t*+1) will become current parameter at the next iteration (the (*t*+1)th iteration). |

**Table 1.2.** E-step and M-step of GEM algorithm

DLR proved that GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the optimal estimate of EM process, which is an optimizer of *L*(Θ).

It is deduced from E-step and M-step that *Q*(Θ | Θ(*t*)) is increased after every iteration. How to maximize *Q*(Θ|Θ(*t*)) is the optimization problem which is dependent on applications. For example, the estimate Θ(*t*+1) can be solution of the equation created by setting the first-order derivative of *Q*(Θ|Θ(*t*)) regarding Θ to be zero, *DQ*(Θ|Θ(*t*)) = **0***T*. If solving such equation is too complex or impossible, some popular methods to solve optimization problem are Newton-Raphson (Burden & Faires, 2011, pp. 67-71), gradient descent (Ta, 2014), and Lagrange duality (Wikipedia, Karush–Kuhn–Tucker conditions, 2014).

In practice, if *Y* is observed as particular *N* observations *Y*1, *Y*2,…, *YN*. Let = {*Y*1, *Y*2,…, *YN*} be the observed sample of size *N* with note that all *Yi* (s) are mutually independent and identically distributed (iid). Given an observation *Yi*, there is an associated random variable *Xi*. All *Xi* (s) are iid and they are not existent in fact. Each is a random variable like *X*. Of course, the domain of each *Xi* is ***X***. Let = {*X*1, *X*2,…, *XN*} be the set of associated random variables. Because all *Xi* (s) are iid, the joint PDF of is determined as follows:

Because all *Xi* (s) are iid and each *Yi* is associated with *Xi*, the conditional joint PDF of given is determined as follows:

The conditional expectation *Q*(Θ’ | Θ) given samples ***X*** and ***Y*** is re-written according to equation 1.5.

|  |  |
| --- | --- |
|  | (1.5) |

Equation 1.5 is proved in (Nguyen, Tutorial on EM algorithm, 2020, pp. 45-47). In case that *f*(*X* | Θ) and *k*(*X* | *Yi*, Θ) belong to exponential family, equation 1.5 becomes equation 1.6 with an observed sample = {*Y*1, *Y*2,…, *YN*}.

|  |  |
| --- | --- |
|  | (1.6) |

Where,

DLR (Dempster, Laird, & Rubin, 1977, p. 1) called ***X*** as *complete data* because the mapping *φ*: ***X*** → ***Y*** is many-one function. There is another case that the complete space ***Z*** consists of hidden space ***X*** and observed space ***Y*** with note that ***X*** and ***Y*** are separated. There is no explicit mapping *φ* from ***X*** and ***Y*** but there exists a PDF of as the joint PDF of and .

The PDF of *Y* becomes:

The PDF *f*(*Y*|Θ) is equivalent to the PDF *g*(*Y*|Θ) mentioned in equation 1.1. Although there is no explicit mapping from ***X*** to ***Y***, the PDF of *Y* above implies an implicit mapping from ***Z*** to ***Y***. The conditional PDF of *X* given *Z* is specified according to Bayes’ rule as follows:

The conditional PDF *f*(*X*|*Y*, Θ) is equivalent to the conditional PDF *k*(*X*|*Y*, Θ) mentioned in equation 1.2. Of course, given *Y*, we always have:

Equation 1.7 specifies the conditional expectation *Q*(Θ’ | Θ) in case that there is no explicit mapping from ***X*** to ***Y*** but there exists the joint PDF of *X* and *Y*.

|  |  |
| --- | --- |
|  | (1.7) |

Where,

Note, ***X*** is separated from ***Y*** and the complete data ***Z*** = (***X***, ***Y***) is composed of ***X*** and ***Y***. For equation 1.7, the existence of the joint PDF *f*(*X*, *Y* | Θ) can be replaced by the existence of the conditional PDF *f*(*Y*|*X*, Θ) and the prior PDF *f*(*X*|Θ) due to:

In applied statistics, equation 1.4 is often replaced by equation 1.7 because specifying the joint PDF *f*(*X*, *Y* | Θ) is more practical than specifying the mapping *φ*: ***X*** → ***Y***. However, equation 1.4 is more general equation 1.7 because the requirement of the joint PDF for equation 1.7 is stricter than the requirement of the explicit mapping for equation 1.4. In case that *X* and *Y* are discrete, equation 1.7 becomes:

In practice, suppose *Y* is observed as a sample = {*Y*1, *Y*2,…, *YN*} of size *N* with note that all *Yi* (s) are mutually independent and identically distributed (iid). The observed sample is associated with a a hidden set (latent set) = {*X*1, *X*2,…, *XN*} of size *N*. All *Xi* (s) are iid and they are not existent in fact. Let be the random variable representing every *Xi*. Of course, the domain of *X* is ***X***. Equation 1.8 specifies the conditional expectation *Q*(Θ’ | Θ) given such .

|  |  |
| --- | --- |
|  | (1.8) |

Equation 1.8 is a variant of equation 1.5 in case that there is no explicit mapping between *Xi* and *Yi* but there exists the same joint PDF between *Xi* and *Yi*. If both *X* and *Y* are discrete, equation 1.8 becomes:

|  |  |
| --- | --- |
|  | (1.9) |

If *X* is discrete and *Y* is continuous such that *f*(*X*, *Y* | Θ) = *P*(*X*|Θ)*f*(*Y* | *X*, Θ) then, according to the total probability rule, we have:

Note, when only *X* is discrete, its PDF *f*(*X*|Θ) becomes the probability *P*(*X*|Θ). Therefore, equation 1.10 is a variant of equation 1.8, as follows:

|  |  |
| --- | --- |
|  | (1.10) |

Where *P*(*X* | *Yi*, Θ) is determined by Bayes’ rule, as follows:

Equation 1.10 is the base for estimating the probabilistic mixture model by EM algorithm, which is not main subject of this report. Now we consider how to apply EM into handling missing data in which equation 1.8 is most concerned. The goal of maximum likelihood estimation (MLE), maximum a posteriori (MAP), and EM is to estimate statistical based on sample. Whereas MLE and MAP require complete data, EM accepts hidden data or incomplete data. Therefore, EM is appropriate to handle missing data which contains missing values. Indeed, estimating parameter with missing data is very natural for EM but it is necessary to have a new viewpoint in which missing data is considered as hidden data (*X*). Moreover, the GEM version with joint probability (without mapping function, please see equation 1.7 and equation 1.8) is used and some changes are required. Handling missing data, which is the main subject of this report is described in next section.

**2. Handling missing data**

Let *X* = (*x*1, *x*2,…, *xn*)*T* be *n*-dimension random variable whose *n* elements are partial random variables *xj* (s). Suppose *X* is composed of two parts such as observed part *Xobs* and missing part *Xmis* such that *X* = {*Xobs*, *Xmis*}. Note, *Xobs* and *Xmis* are considered as random variables.

|  |  |
| --- | --- |
|  | (2.1) |

When *X* is observed, *Xobs* and *Xmis* are determined. For example, given *X* = (*x*1, *x*2, *x*3, *x*4)*T*, when *X* is observed as *X* = (*x*1=1, *x*2=?, *x*3=4, *x*4=?, *x*5=9)*T* where question mask “?” denotes missing value, *Xobs* and *Xmis* are determined as *Xobs* = (*x*1=1, *x*3=4, *x*5=9)*T* and *Xmis* = (*x*2=?, *x*4=?)*T*. When *X* is observed as *X* = (*x*1=?, *x*2=3, *x*3=4, *x*4=?, *x*5=?)*T* then, *Xobs* and *Xmis* are determined as *Xobs* = (*x*2=3, *x*3=4)*T* and *Xmis* = (*x*1=?, *x*4=?, *x*5=?)*T*. Let *M* be a set of indices that *xj* (s) are missing when *X* is observed. *M* is called missing index set.

|  |  |
| --- | --- |
|  | (2.2) |

Suppose

|  |  |
| --- | --- |
|  | (2.3) |

Where,

Let is complementary set of the set *M* given the set {1, 2,…., *n*}.is called existent index set.

|  |  |
| --- | --- |
|  | (2.4) |

*M* or can be empty. They are mutual because can be defined based on *M* and vice versa.

Suppose

|  |  |
| --- | --- |
|  | (2.5) |

Where,

We have:

|  |  |
| --- | --- |
|  | (2.6) |

|  |  |
| --- | --- |
|  | (2.7) |

Obviously, dimension of *Xmis* is |*M*| and dimension of *Xobs* is = *n*–|*M*|. Note, when composing *X* from *Xobs* and *Xmis* as *X* = {*Xobs*, *Xmis*}, it is required a right re-arrangement of elements in both *Xobs* and *Xmis*.

Let *Z* = (*z*1, *z*2,…, *zn*)*T* be *n*-dimension random variable whose each element *zj* is binary random variable indicating if *xj* is missing. Random variable *Z* is also called missingness variable.

|  |  |
| --- | --- |
|  | (2.8) |

For example, given *X* = (*x*1, *x*2, *x*3, *x*4)*T*, when *X* is observed as *X* = (*x*1=1, *x*2=?, *x*3=4, *x*4=?, *x*5=9)*T*, we have *Xobs* = (*x*1=1, *x*3=4, *x*5=9)*T*, *Xmis* = (*x*2=?, *x*4=?)*T*, and *Z* = (*z*1=0, *z*2=1, *z*3=0, *z*4=1, *z*5=0)*T*.

Generally, when *X* is replaced by a sample = {*X*1, *X*2,…, *XN*} whose *Xi* (s) are iid, let = {*Z*1, *Z*2,…, *ZN*} be a set of missingness variables associated with . All *Zi* (s) are iid too. and can be represented as matrices. Given *Xi*, its associative quantities are *Zi*, *Mi*, and . Let *X* = {*Xobs*, *Xmis*} be random variable representing every *Xi*. Let *Z* be random variable representing every *Zi*. As a convention, *Xobs*(*i*) and *Xmis*(*i*) refer to *Xobs* part and *Xmis* part of *Xi*. We have:

|  |  |
| --- | --- |
|  | (2.9) |

For example, given sample of size 4, = {*X*1, *X*2, *X*3, *X*4} in which *X*1 = (*x*11=1, *x*12=?, *x*13=3, *x*14=?)*T*, *X*2 = (*x*21=?, *x*22=2, *x*23=?, *x*24=4)*T*, *X*3 = (*x*31=1, *x*32=2, *x*33=?, *x*34=?)*T*, and *X*4 = (*x*41=?, *x*42=?, *x*43=3, *x*44=4)*T* are iid. Therefore, we also have *Z*1 = (*z*11=0, *z*12=1, *z*13=0, *z*14=1)*T*, *Z*2 = (*z*21=1, *z*22=0, *z*23=1, *z*24=0)*T*, *Z*3 = (*z*31=0, *z*32=0, *z*33=1, *z*34=1)*T*, and *Z*4 = (*z*41=1, *z*42=1, *z*43=0, *z*44=0)*T.* All *Zi* (s) are iid too.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | *x*1 | *x*2 | *x*3 | *x*4 |  |  | *z*1 | *z*2 | *z*3 | *z*4 |
| *X*1 | 1 | ? | 3 | ? |  | *Z*1 | 0 | 1 | 0 | 1 |
| *X*2 | ? | 2 | ? | 4 |  | *Z*2 | 1 | 0 | 1 | 0 |
| *X*3 | 1 | 2 | ? | ? |  | *Z*3 | 0 | 0 | 1 | 1 |
| *X*4 | ? | ? | 3 | 4 |  | *Z*4 | 1 | 1 | 0 | 0 |

Of course, we have *Xobs*(1) = (*x*11=1, *x*13=3)*T*, *Xmis*(1) = (*x*12=?, *x*14=?)*T*, *Xobs*(2) = (*x*22=2, *x*24=4)*T*, *Xmis*(2) = (*x*21=?, *x*23=?)*T*, *Xobs*(3) = (*x*31=1, *x*32=2)*T*, *Xmis*(3) = (*x*33=?, *x*34=?)*T*, *Xobs*(4) = (*x*43=3, *x*44=4)*T*, and *Xmis*(4) = (*x*41=?, *x*42=?)*T*. We also have *M*1 = {*m*11=2, *m*12=4}, = {=1, =3}, *M*2 = {*m*21=1, *m*22=3}, = {=2, =4}, *M*3 = {*m*31=3, *m*32=4}, = {=1, =2}, *M*4 = {*m*41=1, *m*42=2}, and = {=3, =4}.

Both *X* and *Z* are associated with their own PDFs, as follows:

|  |  |
| --- | --- |
|  | (2.10) |

Where Θ and Φ are parameters of PDFs of *X* = {*Xobs*, *Xmis*} and *Z*, respectively. The goal of handling missing data is to estimate Θ and Φ given *X*. Sufficient statistic of *X* = {*Xobs*, *Xmis*} is composed of sufficient statistic of *Xobs* and sufficient statistic of *Xmis*.

|  |  |
| --- | --- |
|  | (2.11) |

How to compose *τ*(*X*) from *τ*(*Xobs*) and *τ*(*Xmis*) is dependent on distribution type of the PDF *f*(*X*|Θ).

The joint PDF of *X* and *Z* is main object of handling missing data, which is defined as follows:

|  |  |
| --- | --- |
|  | (2.12) |

The PDF of *Xobs* is defined as integral of *f*(*X*|Θ) over *Xmis*:

|  |  |
| --- | --- |
|  | (2.13) |

The PDF of *Xmis* is conditional PDF of *Xmis* given *Xobs* is:

|  |  |
| --- | --- |
|  | (2.14) |

The notation Θ*M* implies that the parameter Θ*M* of the PDF *f*(*Xmis* | *Xobs*, Θ*M*) is derived from the parameter Θ of the PDF *f*(*X*|Θ), which is function of Θ and *Xobs* asΘ*M* = *u*(Θ, *Xobs*). Thus, Θ*M* is not a new parameter and it is dependent on distribution type.

|  |  |
| --- | --- |
|  | (2.15) |

How to determine *u*(Θ, *Xobs*) is dependent on distribution type of the PDF *f*(*X*|Θ).

There are three types of missing data, which depends on relationship between *Xobs*, *Xmis*, and *Z* (Josse, Jiang, Sportisse, & Robin, 2018):

1. Missing data (*X* or ) is Missing Completely At Random (MCAR) if the probability of *Z* is independent from both *Xobs* and *Xmis* such that *f*(Z | *Xobs*, *Xmis*, Φ) = *f*(Z | Φ).
2. Missing data (*X* or ) is Missing At Random (MAR) if the probability of *Z* depends on only *Xobs* such that *f*(Z | *Xobs*, *Xmis*, Φ) = *f*(Z | *Xobs*, Φ).
3. Missing data (*X* or ) is Missing Not At Random (MNAR) in all other cases, *f*(Z | *Xobs*, *Xmis*, Φ) = *f*(Z | *Xobs*, *Xmis*, Φ).

There are two main approaches for handling missing data (Josse, Jiang, Sportisse, & Robin, 2018):

1. Using some statistical models such as EM to estimate parameter with missing data.
2. Inputting plausible values for missing values to obtain some complete samples (copies) from the missing data. Later on, every complete sample is used to produce an estimate of parameter by some estimation methods, for example, MLE and MAP. Finally, all estimates are synthesized to produce the best estimate.

Here we focus on the first approach with EM to estimate parameter with missing data. Without loss of generality, given sample = {*X*1, *X*2,…, *XN*} in which all *Xi* (s) are iid, by applying equation 1.8 for GEM with the joint PDF *f*(*Xobs*, *Xmis*, *Z* | Θ, Φ), we consider {*Xobs*, *Z*} as observed part and *Xmis* as hidden part. Let *X* = {*Xobs*, *Xmis*} be random variable representing all *Xi* (s). Let *Xobs*(*i*) denote observed part *Xobs* of *Xi* and let *Zi* be missingness variable corresponding to *Xi*, by following equation 1.8, the expectation *Q*(Θ’, Φ’ | Θ, Φ) becomes:

In short, *Q*(Θ’, Φ’ | Θ, Φ) is specified as follows:

|  |  |
| --- | --- |
|  | (2.16) |

Where,

Note, unknowns of *Q*(Θ’, Φ’ | Θ, Φ) are Θ’ and Φ’. Because it is not easy to maximize *Q*(Θ’, Φ’ | Θ, Φ) with regard to Θ’ and Φ’, we assume that the PDF *f*(*X*|Θ) belongs to exponential family.

|  |  |
| --- | --- |
|  | (2.17) |

Note,

It is easy to deduce that

|  |  |
| --- | --- |
|  | (2.18) |

Therefore,

We have:

Therefore, equation 2.19 specifies *Q*1(Θ’|Θ) given *f*(*X*|Θ) belongs to exponential family.

|  |  |
| --- | --- |
|  | (2.19) |

Where,

|  |  |
| --- | --- |
|  | (2.20) |

|  |  |
| --- | --- |
|  | (2.21) |

At M-step of some *t*th iteration, the next parameter Θ(*t*+1) is solution of the equation created by setting the first-order derivative of *Q*1(Θ’|Θ) to be zero. The first-order derivative of *Q*1(Θ’|Θ) is:

By referring table 1.2, we have:

Where,

Thus, the next parameter Θ(*t*+1) is solution of the following equation:

This implies the next parameter Θ(*t*+1) is solution of the following equation:

As a result, at E-step of some *t*th iteration, given current parameter Θ(*t*), the sufficient statistic of *X* is calculated as follows:

|  |  |
| --- | --- |
|  | (2.22) |

Where,

Equation 2.22 is variant of equation 2.11 when *f*(*X*|Θ) belongs to exponential family but how to compose *τ*(*X*) from *τ*(*Xobs*) and *τ*(*Xmis*) is not determined exactly yet.

As a result, at M-step of some *t*th iteration, given *τ*(*t*) and Θ(*t*), the next parameter Θ(*t*+1) is a solution of the following equation:

|  |  |
| --- | --- |
|  | (2.23) |

Moreover, at M-step of some *t*th iteration, the next parameter Φ(*t*+1) is a maximizer of *Q*2(Φ | Θ(*t*)) given Θ(*t*) as follows:

|  |  |
| --- | --- |
|  | (2.24) |

Where,

|  |  |
| --- | --- |
|  | (2.25) |

How to maximize *Q*2(Φ | Θ(*t*)) depends on distribution type of *Zi* which is also formulation of the PDF *f*(Z | *Xobs*, *Xmis*, Φ). For some reasons, such as accelerating estimation speed or ignoring missingness variable *Z* then, the next parameter Φ(*t*+1) will not be estimated.

In general, the two steps of GEM algorithm for handling missing data at some *t*th iteration are summarized in table 2.1 with assumption that the PDF of missing data *f*(*X*|Θ) belongs to exponential family.

|  |
| --- |
| *E-step*:  Given current parameter Θ(*t*), the sufficient statistic *τ*(*t*) is calculated according to equation 2.22.  Where,  *M-step*:  Given *τ*(*t*) and Θ(*t*), the next parameter Θ(*t*+1) is a solution of equation 2.23.  Given Θ(*t*), the next parameter Φ(*t*+1) is a maximizer of *Q*2(Φ | Θ(*t*)) according to equation 2.24.  Where, |

**Table 2.1.** E-step and M-step of GEM algorithm for handling missing data given exponential PDF

GEM algorithm converges at some *t*th iteration. At that time, Θ*\** = Θ(*t*+1) = Θ(*t*) and Φ*\** = Φ(*t*+1) = Φ(*t*) are optimal estimates. If missingness variable *Z* is ignored for some reasons, parameter Φ is not estimated. Because *Xmis* is a part of *X* and *f*(*Xmis* | *Xobs*, Θ*M*) is derived directly from *f*(*X*|Θ), in practice we can stop GEM after its first iteration was done, which is reasonable enough to handle missing data.

An interesting application of handling missing data is to fill in or predict missing values. For instance, suppose the estimate resulted from GEM is Θ*\**, missing values represented by *τ*(*Xmis*) are fulfilled by expectation of *τ*(*Xmis*) as follows:

|  |  |
| --- | --- |
|  | (2.26) |

Where,

Now we survey a popular case that sample = {*X*1, *X*2,…, *XN*} whose *Xi* (s) are iid is MCAR data and *f*(*X*|Θ) is multinormal PDF whereas missingness variable *Z* follows binomial distribution of *n* trials. Let *X* = {*Xobs*, *Xmis*} be random variable representing every *Xi*. Suppose dimension of *X* is *n*. Let *Z* be random variable representing every *Zi*. According to equation 2.9, recall that

The PDF of *X* is:

|  |  |
| --- | --- |
|  | (2.27) |

Therefore,

The PDF of *Z* is:

|  |  |
| --- | --- |
|  | (2.28) |

Therefore,

Where Θ = (*μ*, Σ)*T* and Φ = *p*. Note, given the PDF *f*(*X* | Θ), *µ* is mean and Σ is covariance matrix whose each element *σij* is covariance of *xi* and *xj*.

|  |  |
| --- | --- |
|  | (2.29) |

Suppose the probability of missingness at every partial random variable *xj* is *p* and it is independent from *Xobs* and *Xmis*. The quantity *c*(*Z*) is the number of *zj* (s) in *Z* that equal 1. For example, if *Z* = (1, 0, 1, 0)*T* then, *c*(*Z*) = 2. The most important task here is to define equation 2.11 and equation 2.15 in order to compose *τ*(*X*) from *τ*(*Xobs*), *τ*(*Xmis*) and to extract Θ*M* from Θ when *f*(*X*|Θ) distributes normally.

The conditional PDF of *Xmis* given *Xobs* is also multinormal PDF.

|  |  |
| --- | --- |
|  | (2.30) |

Therefore,

Where . We denote

Because only depends on within normal PDF whereas depends on *Xobs*(*i*). Determining the function = *u*(Θ, *Xobs*(*i*)) is now necessary to extract the parameter from Θ given *Xobs*(*i*) when *f*(*Xi*|Θ) is normal distribution.

Let Θ*mis* = (*μmis*, Σ*mis*)*T* be parameter of marginal PDF of *Xmis*, we have:

|  |  |
| --- | --- |
|  | (2.31) |

Therefore,

Where,

|  |  |
| --- | --- |
|  | (2.32) |

Obviously, Θ*mis*(*i*) is extracted from Θ given indicator *Mi*. Note, is covariance of and .

Let Θ*obs* = (*μobs*, Σ*obs*)*T* be parameter of marginal PDF of *Xobs*, we have:

|  |  |
| --- | --- |
|  | (2.33) |

Therefore,

Where,

|  |  |
| --- | --- |
|  | (2.34) |

Obviously, Θ*obs*(*i*) is extracted from Θ given indicator or *Mi*. Note, is covariance of and .

We have:

Therefore, it is easy to form the parameter from Θ*mis*(*i*) = (*μmis*(*i*), Σ*mis*(*i*))*T* and Θ*obs*(*i*) = (*μobs*(*i*), Σ*obs*(*i*))*T* as follows (Hardle & Simar, 2013, pp. 156-157):

|  |  |
| --- | --- |
|  | (2.35) |

Where from Θ*mis*(*i*) = (*μmis*(*i*), Σ*mis*(*i*))*T* and Θ*obs*(*i*) = (*μobs*(*i*), Σ*obs*(*i*))*T* are specified by equation 2.32 and equation 2.34. Moreover the *k*x*l* matrix which implies correlation between *Xmis* and *Xobs* is defined as follows (*k* = |*Mi*| and *l* = ):

|  |  |
| --- | --- |
|  | (2.36) |

Note, is covariance of and . The *l*x*k* matrix which implies correlation between *Xobs* and *Xmis* is defined as follows:

|  |  |
| --- | --- |
|  | (2.37) |

Therefore, equation 2.35 to extract from Θ given *Xobs*(*i*) is an instance of equation 2.15. For convenience let,

|  |  |
| --- | --- |
|  | (2.38) |

Equation 2.38 is result of equation 2.35. Given then, is estimated partial mean of and is estimated partial covariance of and given the conditional PDF *f*(*Xmis* | ).

At E-step of some *t*th iteration, given current parameter Θ(*t*), the sufficient statistic of *X* is calculated according to equation 2.22. Let,

It is necessary to calculate the sufficient with normal PDF *f*(*Xi*|Θ), which means that we need to define what *τ*1(*t*) and *τ*2(*t*) are. The sufficient statistic of *Xobs*(*i*) is:

The sufficient statistic of *Xmis*(*i*) is:

We also have:

Due to

Where and are and at current iteration, respectively. By referring to equation 2.38, we have

And

Where,

Therefore, *τ*1(*t*) is vector and *τ*2(*t*) is matrix and then, the sufficient statistic of *X* at E-step of some *t*th iteration, given current parameter Θ(*t*) is defined as follows:

|  |  |
| --- | --- |
|  | (2.39) |

Each is calculated as follows:

|  |  |
| --- | --- |
|  | (2.40) |

Please see equation 2.35 and equation 2.38 to know . Each is calculated as follows:

|  |  |
| --- | --- |
|  | (2.41) |

Equation 2.39 is an instance of equation 2.11, which compose *τ*(*X*) from *τ*(*Xobs*) and *τ*(*Xmis*) when *f*(*X*|Θ) distributes normally. Following is the proof of equation 2.41.

If and then, the partial statistic *xiuxiv* is kept intact because *xiu* and *xiv* are in *Xobs* are constant with regard to *f*(*Xmis* | ) If and then, the partial statistic *xiuxiv* is replaced by the expectation as follows:

If and then, the partial statistic *xiuxiv* is replaced by the expectation as follows:

If and then, the partial statistic *xiuxiv* is replaced by the expectation as follows:

At M-step of some *t*th iteration, given *τ*(*t*) and Θ(*t*), the next parameter Θ(*t*+1) = (*μ*(*t*+1), Σ(*t*+1))*T* is a solution of equation 2.23.

Due to

Equation 2.23 becomes:

Which means that

|  |  |
| --- | --- |
|  | (2.42) |

Please see equation 2.40 and equation 2.41 to know and .

Moreover, at M-step of some *t*th iteration, the next parameter Φ(*t*+1) = *p*(*t*+1) is a maximizer of *Q*2(Φ | Θ(*t*)) given Θ(*t*) according to equation 2.24.

Because the PDF of *Zi* is:

The *Q*2(Φ|Θ(*t*)) becomes:

The next parameter Φ(*t*+1) = *p*(*t*+1) is solution of the equation created by setting the first-order derivative of *Q*2(Φ|Θ(*t*)) to be zero, which means that:

It is easy to deduce that the next parameter *p*(*t*+1) is:

|  |  |
| --- | --- |
|  | (2.43) |

In general, given sample = {*X*1, *X*2,…, *XN*} whose *Xi* (s) are iid is MCAR data and *f*(*X*|Θ) is multinormal PDF whereas missingness variable *Z* follows binomial distribution of *n* trials, GEM for handling missing data is summarized in table 2.2.

|  |
| --- |
| *E-step*:  Given current parameter Θ(*t*) = (*μ*(*t*), Σ(*t*))*T*, the sufficient statistic *τ*(*t*) is calculated according to equation 2.39, equation 2.40, and equation 2.41.  Where and are specified in equation 2.35 and equation 2.38.  *M-step*:  Given *τ*(*t*) and Θ(*t*), the next parameter Θ(*t*+1) = (*μ*(*t*+1), Σ(*t*+1))*T* is specified by equation 2.42.  Given Θ(*t*), the next parameter Φ(*t*+1) = *p*(*t*+1) is specified by equation 2.43.  Where *c*(*Zi*) is the number of *zij* (s) in *Zi* that equal 1. |

**Table 2.2.** E-step and M-step of GEM algorithm for handling missing data given normal PDF

As aforementioned, an interesting application of handling missing data is to fill in or predict missing values. For instance, suppose the estimate resulted from GEM is Θ*\** = (*μ*\*, Σ\*)*T*, missing part is replaced by as follows:

|  |  |
| --- | --- |
|  | (2.44) |

Note, which is extracted from *μ*\* is estimated mean of the conditional PDF of *Xmis* (given *Xobs*) according to equation 2.35. Moreover, is estimated partial mean of given the conditional PDF *f*(*Xmis* | ), please see equation 2.38 for more details about . As aforementioned, in practice we can stop GEM after its first iteration was done, which is reasonable enough to handle missing data.

Now we survey another interesting case that sample = {*X*1, *X*2,…, *XN*} whose *Xi* (s) are iid is MCAR data and *f*(*X*|Θ) is multinomial PDF of *K* trials. We ignore missingness variable *Z* here because it is included in the case of multinormal PDF. Let *X* = {*Xobs*, *Xmis*} be random variable representing every *Xi*. Suppose dimension of *X* is *n*. According to equation 2.9, recall that

The PDF of *X* is:

|  |  |
| --- | --- |
|  | (2.45) |

Where *xj* are integers and Θ = (*p*1, *p*2,…, *pn*)*T* is the set of probabilities such that

Note, *xj* is the number of trials generating nominal value *j*. Therefore,

Where,

The most important task here is to define equation 2.11 and equation 2.15 in order to compose *τ*(*X*) from *τ*(*Xobs*), *τ*(*Xmis*) and to extract Θ*M* from Θ when *f*(*X*|Θ) is multinomial PDF.

Let Θ*mis* be parameter of marginal PDF of *Xmis*, we have:

|  |  |
| --- | --- |
|  | (2.46) |

Therefore,

Where,

|  |  |
| --- | --- |
|  | (2.47) |

Obviously, Θ*mis*(*i*) is extracted from Θ given indicator *Mi*.

Let Θ*obs* be parameter of marginal PDF of *Xobs*, we have:

|  |  |
| --- | --- |
|  | (2.48) |

Therefore,

Where,

|  |  |
| --- | --- |
|  | (2.49) |

Obviously, Θ*obs*(*i*) is extracted from Θ given indicator or *Mi*.

The conditional PDF of *Xmis* given *Xobs* is calculated based on the PDF of *X* and the marginal PDF of *Xobs* as follows:

This implies that the conditional PDF of *Xmis* given *Xobs* is multinomial PDF of *K* trials.

|  |  |
| --- | --- |
|  | (2.50) |

Therefore,

Where

Obviously, the parameter of the conditional PDF is:

|  |  |
| --- | --- |
|  | (2.51) |

Therefore, equation 2.51 to extract from Θ given *Xobs*(*i*) is an instance of equation 2.15. It is easy to check that

At E-step of some *t*th iteration, given current parameter Θ(*t*) = (*p*1(*t*), *p*2(*t*),…, *pn*(*t*))*T*, the sufficient statistic of *X* is calculated according to equation 2.22. Let,

The sufficient statistic of *Xobs*(*i*) is:

The sufficient statistic of *Xmis*(*i*) with regard to is:

We also have:

Therefore, the sufficient statistic of *X* at E-step of some *t*th iteration given current parameter Θ(*t*) = (*p*1(*t*), *p*2(*t*),…, *pn*(*t*))*T* is defined as follows:

|  |  |
| --- | --- |
|  | (2.52) |

Equation 2.52 is an instance of equation 2.11, which compose *τ*(*X*) from *τ*(*Xobs*) and *τ*(*Xmis*) when *f*(*X*|Θ) is multinomial PDF.

At M-step of some *t*th iteration, we need to maximize *Q*1(Θ’|Θ) with following constraint

According to equation 2.19, we have:

Where quantities *b*(*Xobs*(*i*), *Xmis*) and *a*(Θ’) belongs to the PDF *f*(*X*|Θ) of *X*. Because there is the constraint , we use Lagrange duality method to maximize *Q*1(Θ’|Θ). The Lagrange function *la*(Θ’, *λ* | Θ) is sum of *Q*1(Θ’|Θ) and the constraint , as follows:

Where Θ’ = (*p*1’, *p*2’,…, *pn*’)*T*. Note, *λ* ≥ 0 is called Lagrange multiplier. Of course, *la*(Θ’, *λ* | Θ) is function of Θ’ and *λ*. The next parameter Θ(*t*+1) that maximizes *Q*1(Θ’|Θ) is solution of the equation formed by setting the first-order derivative of Lagrange function regarding Θ’ and *λ* to be zero.

The first-order partial derivative of *la*(Θ’, *λ* | Θ) with regard to Θ’ is:

By referring table 1.2, we have:

Thus,

The first-order partial derivative of *la*(Θ’, *λ* | Θ) with regard to *λ* is:

Therefore, at M-step of some *t*th iteration, given current parameter Θ(*t*) = (*p*1(*t*), *p*2(*t*),…, *pn*(*t*))*T*, the next parameter Θ(*t*+1) is solution of the following equation:

This implies:

Where,

Due to

We obtain *n* equations *Kpj* = –*λ*/*N* + and 1 constraint . Therefore, we have:

Summing *n* equations above, we have:

Suppose every missing value is estimated by such that:

We obtain:

This implies

Such that

Therefore, at M-step of some *t*th iteration, given current parameter Θ(*t*) = (*p*1(*t*), *p*2(*t*),…, *pn*(*t*))*T*, the next parameter Θ(*t*+1) is specified by following equation.

|  |  |
| --- | --- |
|  | (2.53) |

In general, given sample = {*X*1, *X*2,…, *XN*} whose *Xi* (s) are iid is MCAR data and *f*(*X*|Θ) is multinomial PDF of *K* trials, GEM for handling missing data is summarized in table 2.3.

|  |
| --- |
| *M-step*:  Given *τ*(*t*) and Θ(*t*) = (*p*1(*t*), *p*2(*t*),…, *pn*(*t*))*T*, the next parameter Θ(*t*+1) is specified by equation 2.53. |

**Table 2.3.** E-step and M-step of GEM algorithm for handling missing data given multinomial PDF

In table 2.3, E-step is implied in how to perform M-step. As aforementioned, in practice we can stop GEM after its first iteration was done, which is reasonable enough to handle missing data. Next section includes two examples of handling missing data with multinormal distribution and multinomial distribution.

**3. Numerical examples**

It is necessary to have an example for illustrating how to handle missing data with multinormal PDF.

**Example 3.1.** Given sample of size two, = {*X*1, *X*2 } in which *X*1 = (*x*11=1, *x*12=?, *x*13=3, *x*14=?)*T* and *X*2 = (*x*21=?, *x*22=2, *x*23=?, *x*24=4)*T* are iid. Therefore, we also have *Z*1 = (*z*11=0, *z*12=1, *z*13=0, *z*14=1)*T* and *Z*2 = (*z*21=1, *z*22=0, *z*23=1, *z*24=0)*T.* All *Zi* (s) are iid too.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | *x*1 | *x*2 | *x*3 | *x*4 |  |  | *z*1 | *z*2 | *z*3 | *z*4 |
| *X*1 | 1 | ? | 3 | ? |  | *Z*1 | 0 | 1 | 0 | 1 |
| *X*2 | ? | 2 | ? | 4 |  | *Z*2 | 1 | 0 | 1 | 0 |

Of course, we have *Xobs*(1) = (*x*11=1, *x*13=3)*T*, *Xmis*(1) = (*x*12=?, *x*14=?)*T*, *Xobs*(2) = (*x*22=2, *x*24=4)*T* and *Xmis*(2) = (*x*21=?, *x*23=?)*T*. We also have *M*1 = {*m*11=2, *m*12=4}, = {=1, =3}, *M*2 = {*m*21=1, *m*22=3}, and = {=2, =4}. Let *X* and *Z* be random variables representing every *Xi* and every *Zi*, respectively. Suppose *f*(*X*|Θ) is multinormal PDF and missingness variable *Z* follows binomial distribution of 4 trials according to equation 2.26 and equation 2.27. Dimension of *X* is 4. We will estimate Θ = (*μ*, Σ)*T* and Φ = *p* based on .

The parameters *μ* and Σ are initialized arbitrarily as zero vector and identity vector whereas *p* is initialized 0.5 as follows:

At 1st iteration, E-step, we have:

At 1st iteration, M-step, we have:

At 2nd iteration, E-step, we have:

At 2nd iteration, M-step, we have:

Because the sample is too small for GEM to converge to an exact maximizer with small enough number of iterations, we can stop GEM at the second iteration with Θ(3) = Θ\* = (*μ*\*, Σ\*)*T* and Φ(3) = Φ\* = *p*\* when difference between Θ(2) and Θ(3) is insignificant.

As aforementioned, because *Xmis* is a part of *X* and *f*(*Xmis* | Θ*M*) is derived directly from *f*(*X*|Θ), in practice we can stop GEM after its first iteration was done, which is reasonable enough to handle missing data.

As aforementioned, an interesting application of handling missing data is to fill in or predict missing values. For instance, the missing part *Xmis*(1) of *X*1 = (*x*11=1, *x*12=?, *x*13=3, *x*14=?)*T* is fulfilled by according to equation 2.44 as follows:

It is necessary to have an example for illustrating how to handle missing data with multinomial PDF.

**Example 3.2.** Given sample of size two, = {*X*1, *X*2 } in which *X*1 = (*x*11=1, *x*12=?, *x*13=3, *x*14=?)*T* and *X*2 = (*x*21=?, *x*22=2, *x*23=?, *x*24=4)*T* are iid.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | *x*1 | *x*2 | *x*3 | *x*4 |
| *X*1 | 1 | ? | 3 | ? |
| *X*2 | ? | 2 | ? | 4 |

Of course, we have *Xobs*(1) = (*x*11=1, *x*13=3)*T*, *Xmis*(1) = (*x*12=?, *x*14=?)*T*, *Xobs*(2) = (*x*22=2, *x*24=4)*T* and *Xmis*(2) = (*x*21=?, *x*23=?)*T*. We also have *M*1 = {*m*11=2, *m*12=4}, = {=1, =3}, *M*2 = {*m*21=1, *m*22=3}, and = {=2, =4}. Let *X* be random variable representing every *Xi*. Suppose *f*(*X*|Θ) is multinomial PDF of 10 trials. We will estimate Θ = (*p*1, *p*2, *p*3, *p*4)*T*. The parameters *p*1, *p*2, *p*3, and *p*2 are initialized arbitrarily as 0.25 as follows:

At 1st iteration, M-step, we have:

We stop GEM after the first iteration was done, which results the estimate Θ(2) = Θ\* = (*p*1\*, *p*2\*, *p*3\*, *p*4\*)*T* as follows:

**4. Conclusions**

In general, GEM is a powerful tool to handle missing data, which is not so difficult except that how to extract the parameter Θ*M* of the conditional PDF *f*(*Xmis* | *Xobs*, Θ*M*) from the whole parameter Θ of the PDF *f*(*X*|Θ*M*) is most important with note that only *f*(*X*|Θ) is defined firstly and then *f*(*Xmis* | *Xobs*, Θ*M*) is derived from *f*(*X*|Θ). Therefore, equation 2.15 is cornerstone of this method. Note, equation 2.35 and 2.51 are instances of equation 2.15 when *f*(*X*|Θ) is multinormal PDF or multinomial PDF.

# **Expectation maximization algorithm with combinatorial assumption**

Loc Nguyen

Loc Nguyen’s Academic Network, Vietnam

Email: ng\_phloc@yahoo.com

Homepage: www.locnguyen.net

**Abstract**

Expectation maximization (EM) algorithm is a popular and powerful mathematical method for parameter estimation in case that there exist both observed data and hidden data. The EM process depends on an implicit relationship between observed data and hidden data which is specified by either a mapping function in traditional EM or a joint probability density function (PDF) in practical EM. However, the mapping function is vague and impractical whereas the joint PDF is not easy to be defined because of heterogeneity between observed data and hidden data. The research aims to improve competency of EM by making it more feasible and easier to be specified, which removes the vagueness. Therefore, the research proposes an assumption that observed data is the combination of hidden data which is realized as an analytic function where data points are numerical. In other words, observed points are supposedly calculated from hidden points via regression model. Mathematical computations and proofs indicate feasibility and clearness of the proposed method which can be considered as an extension of EM.

**Keywords:** expectation maximization (EM), observed data, hidden data, mapping function, joint probability density function, combinatorial assumption, regression model.

**1. Introduction**

Expectation maximization (EM) algorithm developed by Dempster, Laird, and Rubin called DLR (Dempster, Laird, & Rubin, 1977) is an extension of maximum likelihood estimation (MLE) method when there exist both observed data *X* and hidden data *Y* and there is an implicit mapping *φ*: ***X*** → ***Y*** such that *φ*–1(*Y*) = {: *φ*(*X*) = *Y*}. Let *f*(*X* | Θ) be the density probability function (PDF) of random variable *X* and let *g*(*Y* | Θ) be the PDF of random variable *Y*. Note, Θ is (vector) parameter.

The conditional PDF of *X* given *Y*, denoted *k*(*X* | *Y*, Θ), is specified as follows:

Given sample = {*Y*1, *Y*2,…, *YN*} whose all *Yi* (s) are mutually independent and identically distributed (iid), EM has many iterations and each iteration has two steps in which expectation step called E-step determines the conditional expectation *Q*(Θ | Θ(*t*)) and maximization step (M-step) re-estimates parameter as follows:

*E-step*:

The expectation *Q*(Θ | Θ(*t*)) is determined based on current parameter Θ(*t*) (Nguyen, Tutorial on EM algorithm, 2020).

|  |  |
| --- | --- |
|  | (1.1) |

*M-step*:

The next parameter Θ(*t*+1) is a maximizer of *Q*(Θ | Θ(*t*)) with subject to Θ. Note that Θ(*t*+1) will become current parameter at the next iteration (the (*t*+1)th iteration).

EM converges at some *t*th iteration; at that time, Θ*\** = Θ(*t*+1) = Θ(*t*) is the optimal estimate of EM process. The expectation *Q*(Θ | Θ(*t*)) based on the mapping *φ*(*X*) represents the traditional EM proposed by DLR. Because it is too vague to specify the mapping *φ*(*X*), practical EM issues the joint PDF of *X* and *Y* denoted *f*(*X*, *Y* | Θ) as prerequisite condition to run EM such that:

The expectation *Q*(Θ | Θ(*t*)) becomes (Nguyen, Tutorial on EM algorithm, 2020):

|  |  |
| --- | --- |
|  | (1.2) |

However, if *X* and *Y* are too different in context due to data heterogeneity and they are not independent, it will be difficult to define the PDF *f*(*X*, *Y* | Θ). In general, it is not easy to specify both the mapping *φ*(*X*) in traditional EM and the joint PDF *f*(*X*, *Y* | Θ) in practical EM. Therefore, the purpose of this research is to extend competency of EM, in which the observed data *Y* is assumed to be the combination of *X*, called combinatorial assumption (CA). In other words, *Y* can be calculated by an analytic function of *X*, which is more feasible than specifying the mapping *φ*(*X*) and easier than specifying the joint PDF *f*(*X*, *Y*). The analytic function is arbitrary but it is linear called regression function in this research for convenience and feasibility. In some cases, it is possible to convert some analytic functions into linear functions. The next section is the main one which focuses on EM with CA. In general, this research is an extension of EM algorithm.

Almost there is no work to modify EM algorithm itself by support of regression model as methodological extension of EM but many researches utilize EM to learn regression model. Although the context of my methodological research is different, it is necessary to survey some other works related to both EM algorithm and regression model. Zang et al. (Zhang, Deng, & Su, 2016) used EM to learn linear regression model from partial missing data. Their experimental results showed EM algorithm is better than other estimation methods in their case. The essence of regression model estimation is to estimate error random variable with suppose that the error variable distributes normally. If there is no assumption of normal error distribution, EM algorithm may not be effective, which is a drawback of EM. Therefore, Barazandeh et al. (Barazandeh, Ghafelebashi, Razaviyayn, & Sriharsha, 2021) proposed a new algorithm based on combining the alternating direction method of multipliers (ADMM) with EM ideology in order to overcome this drawback. Kokic (Kokic, 2002) also used EM algorithm to learn multivariate regression model in case of missing data and applied the method into time series analysis. Zhang and Rockette (Zhang & Rockette, 2006) proposed an EM algorithm for finding the SML estimate and for variance estimation. Essentially, they utilized EM in case of missing data. Because Gaussian mixture is often learned by EM algorithm, the mixture model of regression functions is also learned by EM algorithm. Faria and Soromenho (Faria & Soromenho, 2009) made a comparison of three methods such as tradition EM, classification EM, and stochastic EM in learning such mixture of regression functions. Their experimental results showed that which method is better is dependent on the true regression line and the initialization step of EM. Griffith (Griffith & Smith, 2010)researched some simplifications of EM on regression model, especially, in case of missing data. In general, most of other researches related to EM and regression models focus on two problems: 1) Learning regression model in case of missing data by EM algorithm because EM is appropriate to impute missing values as hidden data. 2) Learning mixture model of regression functions because mixture model is a popular application of EM.

**2. EM with combinatorial assumption**

Given sample = {*Y*1, *Y*2,…, *YN*} of size *N* in which all *Yi* (s) are mutually independent and identically distributed (iid). Suppose *X* = (*x*1, *x*2,…, *xn*) which is vector of size *n* distributes normally with mean vector *μ* and covariance matrix Σ as follows:

|  |  |
| --- | --- |
|  | (2.1) |

Where the superscript “*T*” denotes vector (matrix) transposition operator. The *n*x1 mean *μ* and the *n*x*n* covariance matrix Σ:

Note, *σij*2 is covariance of *xi* and *xj* whereas *σi*2 is variance of *xi*. Let *Y* = (*y*1, *y*2,…, *ym*)*T* be random variable representing all sample random variable *Yi* = (*yi*1, *yi*2,…, *yim*)*T*. Note, *X* is *n*x1 vector and *Y* is *m*x1 vector. Suppose there is an assumption that *Y* is a combination of partial random variables (components) of *X* such that:

As a generalization, let *A* be *m*x*n* matrix whose elements are called regressive coefficients as follows:

This implies

|  |  |
| --- | --- |
|  | (2.2) |

When *Y* is vector, equation 2.2 specifies the so-called vector regressive model. As a convention, let:

Then

The equation above is regression function in which *Y* is called responsor and *X* is called regressor whereas *A* is called regressive matrix. The assumption is combinatorial assumption (CA) aforementioned and the method proposed here is called CA method or CA algorithm. Suppose *Y* distributes normally with mean and covariance matrix *S* as follows:

|  |  |
| --- | --- |
|  | (2.3) |

The *m*x*m* covariance matrix *S* is:

Note, *sij*2 is covariance of *yi* and *yj* whereas *si*2 is variance of *yi*. As a convention, let:

Then

As a convention, is the one which is like *Si* except that *S* is replaced by *S*–1. Note, is vector. Similarly, notations and imply such a meaning. The marginal PDF of *Y* is now defined by support of regression model as follows:

Where parameter Θ = (*μ*, Σ, *A*, *S*)*T* is compound parameter. The equation above is not real total probability rule but it implies that the conditional PDF *f*(*X* | *Y*, Θ) is substituted by regression model. Consequently, the expectation *Q*(Θ | Θ(*t*)) becomes:

|  |  |
| --- | --- |
|  | (2.4) |

It is necessary to specify the conditional PDF *f*(*X* | *Y*, Θ). Indeed, we have:

The joint PDF *f*(*X*, *Y* | Θ) which is the numerator of *f*(*X* | *Y*, Θ) is defined:

Where,

The expression is approximated with *μ* as follows:

As a result, *f*0(*X*, *Y* | Θ) and *f*(*X*, *Y* | Θ) is approximated as follows:

And

The approximation by removing *X*-dependency from the expression is reasonable because the PDF *f*(*X* | *Y*, Θ) contains second-order proportion with the built-in expression and this PDF also reflects regression model with another built-in expression including parameter *S*2. In other words, the dependency of on *X* is unnecessary. Moreover, EM process will adjust parameters by the best way later. In following proofs and computations, we will see that such dependency removal also makes the research easy to apply shifted Gaussian integral in appendix A1.

The denominator of *f*(*X* | *Y*, Θ) which is *f*(*Y* | Θ) is the integral of *f*(*X*, *Y* | Θ) over *X*:

Where *B* is the integral of *f*0(*X*, *Y* | Θ) over *X*:

It requires to calculate *B* to determine *f*(*X* | *Y*, Θ). By referring the appendix A1 given variable *X*, we can denote:

Obviously, *B* is totally determined. Thus, *f*(*Y* | Θ) is approximated as follows:

As a result, the PDF *f*(*X* | *Y*, Θ) is approximated as follows:

Let *k*(*Y*|Θ) be the constant with subject to *X* but it is function of *Y* with parameter Θ, which is defined as follows:

|  |  |
| --- | --- |
|  | (2.5) |

Shortly, the conditional PDF *f*(*X* | *Y*, Θ(*t*)) is specified (approximated) at E-step of some *t*th iteration process as follows:

|  |  |
| --- | --- |
|  | (2.6) |

Consequently, the expectation *Q*(Θ | Θ(*t*)) at E-step of some *t*th iteration is totally determined. At M-step of current *t*th iteration, *Q*(Θ|Θ(*t*)) is maximized by setting its partial derivatives regarding Θ to be zero. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *μ* with note that *Q*(Θ | Θ(*t*)) is analytic function is:

By referring to the appendix A1, we have:

Note, Σ is invertible and symmetric. As a convention, I denote:

The next parameter *μ*(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is solution of the equation , with note that **0** = (0, 0,…, 0)*T* is zero vector, as follows:

|  |  |
| --- | --- |
|  | (2.7) |

The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to Σ with note that *Q*(Θ | Θ(*t*)) is analytic function is:

The next parameter Σ(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is solution of equation formed by setting to zero matrix. Let (**0**) denote zero matrix.

We have:

Note, *μ* is replaced by *μ*(*t*). Thus, the next parameter Σ(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is obtained:

By referring to the appendix A1, we obtain:

|  |  |
| --- | --- |
|  | (2.8) |

As a convention, I denote:

The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *A*0 with note that *Q*(Θ | Θ(*t*)) is analytic function is:

By referring to the appendix A1, we obtain:

Note, is replaced by at current *t*th iteration. Therefore, the next parameter *A*0(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is obtained by setting the partial derivative to be zero:

|  |  |
| --- | --- |
|  | (2.9) |

Note, the next parameter *μ*(*t*+1) is specified by equation 2.7. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to with note that *Q*(Θ | Θ(*t*)) is analytic function is (Saliba, 2016):

Where denotes Kronecker product and *Im* is *m*x*m* identity matrix. The next parameter at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is obtained by setting the partial derivative to be zero. Because the Kronecker product occurs, the equation being equal to zero is equivalent to 1 equation and *n* equations , where

And

Where,

It is necessary to determine every partial expectation to determine the expectation . Indeed, by referring to the appendix A1, we have:

As a result, the next parameter is solution of *n*+1 equations as follows:

|  |  |
| --- | --- |
|  | (2.10) |

Note, *μ* is replaced by the current *μ*(*t*). The equation 2.10 can be solved by Newton-Raphson method. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *S* with note that *Q*(Θ | Θ(*t*)) is analytic function is:

The next parameter *S*(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is solution of equation formed by setting to zero matrix. By referring to the appendix A1, we have:

Therefore, we obtain:

|  |  |
| --- | --- |
|  | (2.11) |

Note, *μ*, *A*0 and are replaced by *μ*(*t*), *A*0(*t*) and , respectively. In general, CA method is EM process with two steps as follows:

*E-step*:

Determining the conditional PDF *f*(*X* | *Y*, Θ(*t*)) specified by equation 2.6 based on current parameter Θ(*t*).

*M-step*:

Calculating next parameters Θ(*t*+1) = (*μ*(*t*+1), Σ(*t*+1), *A*(*t*+1), *S*(*t*+1))*T* based on *f*(*X* | *Y*, Θ(*t*)) determined in the E-step, specified by equations 2.7, 2.8, 2.9, 2.10, and 2.11.

|  |
| --- |
|  |

In practice, it is not necessary to compute the covariance matrix Σ(*t*+1) and the variance *S*(*t*+1) because computational cost is high and it is also really ineffective to estimate Σ and *S*. Note, the condition that both Σ and *S* are invertible along with |Σ| ≠ 0 and |*S*| ≠ 0 is not easy to assert over many computational iterations. The most important parameters are *μ* and *A* and we should fix the other parameters Σ and *S* with hints of predefined bias or background knowledge.

**3. Case of scalar responsor**

Recall that given sample = {*Y*1, *Y*2,…, *YN*} of size *N* in which all *Yi* (s) are iid, equations 2.7, 2.8, 2.9, 2.10, and 2.11 for estimating CA model become complicated although they are general. Here I simplify CA model when every random vector variable *Yi* degrades into random scalar variable *yi* so that the conditional expectation *Q*(Θ | Θ(*t*)) becomes:

|  |  |
| --- | --- |
|  | (3.1) |

The PDF *f*(*X* | *μ*, Σ) which is specified by equation 2.1 is not changed whereas the PDF *f*(*y* | *X*, *α*, *σ*2) is specified as follows:

|  |  |
| --- | --- |
|  | (3.2) |

Note, the random variable *y* represents every random variable *yi* from the sample . Of course, the vector regressive model becomes scalar regressive model as follows:

|  |  |
| --- | --- |
|  | (3.3) |

Where regressive coefficients are specified as follows:

|  |  |
| --- | --- |
|  | (3.4) |

Therefore, the parameter Θ is compound parameter Θ = (*μ*, Σ, *α*, *σ*2)*T*. The parameter estimation process is the same to the aforementioned one but it is simpler. It is necessary to specify the conditional PDF *f*(*X* | *y*, Θ). Indeed, we have:

The joint PDF *f*(*X*, *y* | Θ) which is the numerator of *f*(*X* | *y*, Θ) is defined:

Where,

The expression is approximated with *μ* as follows:

As a result, *f*0(*X*, *y* | Θ) and *f*(*X*, *y* | Θ) is approximated as follows:

And

The approximation by removing *X*-dependency from the expression is reasonable because the PDF *f*(*X* | *y*, Θ) contains second-order proportion with the built-in expression and this PDF also reflects regression model with another built-in expression including parameter *σ*2. In other words, the dependency of on *X* is unnecessary.

The denominator of *f*(*X* | *y*, Θ) which is *f*(*y* | Θ) is the integral of *f*(*X*, *y* | Θ) over *X*:

Where *B* is the integral of *f*0(*X*, *y* | Θ) over *X*:

It requires to calculate *B* to determine *f*(*X* | *y*, Θ). By applying appendix A1, we have:

Thus, *f*(*y* | Θ) is approximated as follows:

As a result, the PDF *f*(*X* | *y*, Θ) is approximated as follows:

Let *k*(*y*|Θ) be the constant with subject to *X* but it is function of *y* with parameter Θ, which is defined as follows:

|  |  |
| --- | --- |
|  | (3.5) |

Shortly, the conditional PDF *f*(*X* | *y*, Θ) is specified at E-step of some *t*th iteration process as follows:

|  |  |
| --- | --- |
|  | (3.6) |

Consequently, the expectation *Q*(Θ | Θ(*t*)) at E-step of some *t*th iteration is totally determined. At M-step of current *t*th iteration, *Q*(Θ|Θ(*t*)) is maximized by setting its partial derivatives regarding Θ to be zero. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *μ* is:

By applying appendix A1, we have:

Note, Σ is invertible and symmetric. The next parameter *μ*(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is solution of the equation as follows:

|  |  |
| --- | --- |
|  | (3.7) |

Where,

|  |  |
| --- | --- |
|  | (3.8) |

Please pay attention to the quantity *δ*(*y* | Θ) because it is the main quantity which is calculated at E-step of every *t*th iteration. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to Σ is:

The next parameter Σ(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is solution of equation formed by setting to zero matrix. Let (**0**) denote zero matrix, we have:

Note, *μ* is replaced by *μ*(*t*). Thus, the next parameter Σ(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is obtained:

By applying appendix A1, we have:

|  |  |
| --- | --- |
|  | (3.9) |

The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *α*0 is:

By applying appendix A1, we obtain:

Note, is replaced by at current *t*th iteration. Therefore, the next parameter *α*0(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is obtained by setting the partial derivative to be zero:

|  |  |
| --- | --- |
|  | (3.10) |

Note, the next parameter *μ*(*t*+1) is specified by equation 3.7. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to is:

Replacing *μ* and *α*0 by *μ*(*t*) and *α*0(*t*), respectively at current *t*th iteration along with applying appendix A1, we have:

Therefore, the next parameter at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is obtained by setting the partial derivative to be zero:

|  |  |
| --- | --- |
|  | (3.11) |

Where,

|  |  |
| --- | --- |
|  | (3.12) |

Note, the superscript “–1” denotes matrix inversion. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *σ*2 is:

By approximating with *μ*(*t*) as follows:

And replacing *μ*, *α*0, and by *μ*(*t*), *α*0(*t*), and , we obtain:

Therefore, the next parameter (*σ*2)(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is obtained by setting the partial derivative to be zero:

|  |  |
| --- | --- |
|  | (3.13) |

Note, the quantity is calculated by equation 3.12. In general, CA method is EM process with two steps as follows:

*E-step*:

Calculating the quantities *δ*(*yi* | Θ(*t*)) specified by equation 3.8 based on current parameter Θ(*t*).

*M-step*:

Calculating next parameters Θ(*t*+1) = (*μ*(*t*+1), Σ(*t*+1), *α*(*t*+1), (*σ*2) (*t*+1))*T* based on the quantities *δ*(*yi* | Θ(*t*)) calculated in the E-step, specified by equations 3.7, 3.9, 3.10, 3.11, and 3.13.

|  |
| --- |
|  |

Shortly, estimation equations in case of scalar responsor are simpler and easier to compute than the ones in general case when responsor is vector variable. In other words, they are more practical.

**4. Without combinatorial assumption**

Combinatorial assumption (CA) is not always feasible when there is no clear relationship between observed data *Y* and hidden data *X*. In the most general case when regressive model is not supported, we need to specify the joint PDF *f*(*X*, *Y* | Θ) such that

Given random variable *Y* represents every random variable *Yi*, suppose *f*(*X*, *Y* | Θ) distributes normally as multinormal distribution. If we assume *X* and *Y* are mutually independent, it is unreal. Thus, given *X* = (*x*1, *x*2,…, *xn*)*T* is *n*-dimension vector and *Y* = (*y*1, *y*2,…, *ym*)*T* is *m*-dimension vector, let *Z* is the composite random variable such that *Z* = = (*z*1, *z*2,…, *zn*, *zn*+1, *zn*+1,…, *yn*+*m*)*T* is *m*-*n*-dimension vector where

Hence *f*(*X*, *Y* | Θ) can be denoted as *f*(*Z* | Θ).

|  |  |
| --- | --- |
|  | (4.1) |

Where *μ* and Σ are mean and covariance matrix given random variable *Z* and so we have Θ = (*μ*, Σ)*T* such that (Hardle & Simar, 2013, p. 156):

|  |  |
| --- | --- |
|  | (4.2) |

Note, indices “1” and “2” are corresponding to *X* and *Y*, respectively. The conditional PDF *f*(*X* | *Y*, Θ) is specified as follows (Hardle & Simar, 2013, p. 157):

|  |  |
| --- | --- |
|  | (4.3) |

Where *μ*12(*Y*) is conditional mean of *X* given *Y* specified as follows (Hardle & Simar, 2013, p. 157):

|  |  |
| --- | --- |
|  | (4.4) |

Please pay attention to the condition mean *μ*12(*Y*) because it is the core of EM without CA. The covariance matrix of *X* given *Y* is specified as follows (Hardle & Simar, 2013, p. 156):

|  |  |
| --- | --- |
|  | (4.5) |

When both *f*(*X*, *Y*, Θ) and *f*(*X* | *Y*, Θ) are specified, the expectation *Q*(Θ | Θ(*t*)) is totally determined. At M-step of current *t*th iteration, *Q*(Θ|Θ(*t*)) is maximized by setting its partial derivatives regarding Θ = (*μ*, Σ)*T* to be zero. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to *μ* is:

The next parameter *μ*(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is solution of the equation as follows:

|  |  |
| --- | --- |
|  | (4.6) |

Note, *μ*12(*t*)(*Yi*) denotes the conditional mean *μ*12(*Yi*) at the *t*th iteration. The first-order partial derivative of *Q*(Θ | Θ(*t*)) with regard to Σ is:

The next parameter Σ(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is solution of equation formed by setting to zero matrix. Let (**0**) denote zero matrix, we have:

Note, *μ* is replaced by *μ*(*t*). Thus, the next parameter Σ(*t*+1) at M-step of some *t*th iteration that maximizes *Q*(Θ|Θ(*t*)) is obtained:

|  |  |
| --- | --- |
|  | (4.7) |

In general, if there is no relationship between observed data and hidden data, EM process without CA has two steps as follows:

*E-step*:

Determining the condition means *μ*12(*Yi*) specified by equation 4.4 based on current parameter Θ(*t*).

*M-step*:

Calculating next parameters Θ(*t*+1) = (*μ*(*t*+1), Σ(*t*+1))*T* based on *μ*12(*Yi*) determined in the E-step, specified by equations 4.6 and 4.7.

Therefore, estimation equations for EM without CA are much simpler but it will not be precise as CA method in the case that there is a clear regressive relationship between observed data and hidden data. Moreover, it is not easy to specify the joint PDF *f*(*X*, *Y* | Θ) in case of data heterogeneity.

**5. Discussions and conclusions**

Although combinatorial assumption (CA) is subjective assumption, it will reach high usefulness and high effectiveness if there is strong regressive relationship between observed data and hidden data in many cases. The regression function may not be linear but it is easy to convert nonlinear functions into linear function in some cases. For instance, there are some nonlinear functions as follows:

|  |  |
| --- | --- |
| Logarithm function |  |
| Exponent function |  |
| Product function |  |

Given logarithm function, the transformation is and *uj* = log(*xj*). For exponent function, the transformation is and *v* = log(*y*). For product function, the transformation is , *v* = log(*y*), *uj* = log(*xj*), and *β*0 = log(*α*0).

Recall that the dimensions of *X* and *Y* are *n* and *m*, respectively. Note, if *m* ≥ *n*, there is no information loss which is ideal case of CA method. Otherwise, if *m* < *n*, the important parameters such as *μ* and *A* vary much with large amplitude due to information loss of dimension reduction. The problem is called large-scale variation. Therefore, in practice there should be restrictions on *μ* and *A* specified as constraints *u*(*μ*) ≤ 0 and *v*(*A*) ≤ 0; for example, slope of regressive hyperplane specified by the normal vectors , which varies in a predefined interval, is a constraint. The solution is to apply Lagrange duality method into maximizing *Q*(Θ|Θ(*t*)), in which a Lagrange function *l*(Θ, *κ*, *λ* | Θ(*t*)) is defined as sum of *Q*(Θ|Θ(*t*)) and constraints *u*(*μ*) ≤ 0 and *v*(*A*) ≤ 0 as follows:

Note *Q*(Θ|Θ(*t*)) is maximized indirectly by maximizing the Lagrange function *l*(Θ, *κ*, *λ* | Θ(*t*)), in which *κ* ≥ 0and *λ* ≥ 0 called Lagrange multipliers are concerned. Another simple trick to alleviate the large-scale variation of *μ* and *A* is to initialize appropriate values *μ*(0) and *A*(0) at the first iteration of EM process.

An application of CA is to learn Bayesian parameter. According to Bayesian statistics, the parameter Θ is considered as random variable. Given sample *D* = {*X*1, *X*2,…, *XN*) whose all observations *Xi* (s) are iid, the posterior probability of Θ given *D* denoted *P*(Θ | *D*, *ξ*) is calculated based on *D* and the prior probability of Θ denoted *P*(Θ | *ξ*).

Where *ξ* denotes background knowledge about Θ which can be considered sub-parameter but we do not focus on learning *ξ*. Let *X* be random variable representing all *Xi*. The most popular method to learn Bayesian parameter is to use binomial sample and beta distribution. Here CA method considers the parameter Θ as hidden data and random variable *X* as observed data.

Therefore, the posterior probability *P*(Θ | *D*, *ξ*) is the same to the conditional PDF *f*(*X* | *y*, Θ) aforementioned. After EM process is finished, *f*(*X* | *y*, Θ) or *P*(Θ | *D*, *ξ*) is obviously determined.

In general, CA is an extension of EM, whose strong point is feasibility and simplicity with clearness of mathematic formulas. Especially, in the ideal case that dimension of *Y* (*y*) is larger than or equal to dimension of *X*, CA method will result out best estimates because there is no information loss. However, its drawback is the large-scale variation in determining important parameters *μ* and *A* when dimension of *Y* (*y*) is smaller than dimension of *X*. Therefore, in the future I will research Lagrange duality method to maximize the expectation *Q*(Θ|Θ(*t*)) with constraints on *μ* and *A*.

**Appendices**

**A1.**

This appendix focuses on calculating integrals related to the following Gaussian function:

Note, ***x*** = (*x*1, *x*2,…, *xh*)*T* is *h*x1 vector, *A* is *n*x*h* matrix, *B* is *m*x*h* matrix, and *c* is *h*x1 vector whereas *R* is *n*x*n* matrix and *S* is *m*x*m* matrix. Moreover, *R* and *S* are invertible and positive definite. Firstly, we calculate the integral of *f*(***x***).

Diagonalizing *R* and *S* as follows:

Where *n*x*n* matrix *U* and *m*x*m* matrix *V* are orthonormal and *n*x*n* matrix Γ and *m*x*m* matrix Λ are eigenmatrices:

We have (Mathprof, 2013):

Let,

We have:

Let

This implies

Suppose ***x*** distributes normally with mean *μ* = (*μ*1, *μ*2,…, *μh*)*T*, the product *xjxl* is approximated by *μjμl* and then, the quantities *yi*2 and *zi*2 are re-written as follows:

Let

We have:

The integral is re-written as follows:

Let *w* be the maximum value among *n*, *m*, and *h* and let

As a convention, let

Where Θ implies parameters of *f*(***x***) such as *A*, *R*, *B*, *S*, *c*. We have (Wikipedia, 2021):

Note, if *Fi* = 0, the respective expression is removed from the product series. We calculate the integral *E*(*xi* | *f*(***x***|Θ)) as follows (Wikipedia, 2021):

Consequently, the integral *E*(*xi*2 | *f*(***x***|Θ)) as follows (Wikipedia, 2021):

Moreover, the integral *E*(*xixj* | *f*(***x***|Θ)) as follows:

In general, we have:

By extending these integrals with vector ***x***, we determine:

# **Early fetal weight estimation with expectation maximization algorithm**

Loc Nguyen

Sunflower Soft Company, An Giang, Vietnam

Thu-Hang T. Ho

Vinh Long General Hospital, Vinh Long, Vietnam

**Abstract**

Fetal weight estimation before delivery is important in obstetrics, which assists doctors diagnose abnormal or diseased cases. Linear regression based on ultrasound measures such as bi-parietal diameter (*bpd*), head circumference (*hc*), abdominal circumference (*ac*), and fetal length (*fl*) is common statistical method for weight estimation but the regression model requires that time points of collecting such measures must not be too far from last ultrasound scans. Therefore this research proposes a method of early weight estimation based on expectation maximization (EM) algorithm so that ultrasound measures can be taken at any time points in gestational period. In other words, gestational sample can lack some or many fetus weights, which gives facilities to practitioners because practitioners need not concern fetus weights when taking ultrasound examinations. The proposed method is called dual regression expectation maximization (DREM) algorithm. Experimental results indicate that accuracy of DREM decreases insignificantly when completion of ultrasound sample decreases significantly. So it is proved that DREM withstands missing values in incomplete sample or sparse sample.

**Keywords**: fetal weight estimation, regression model, ultrasound measures, expectation maximization algorithm.

**1. Introduction**

According to the regression approach of fetal weight estimation, without loss of generality, an estimation formula is a linear regression function *Z = α*0 *+ α*1*X*1 *+ α*2*X*2 *+ … + αnXn* where *Z* is estimated fetal weight whereas *Xi* (s) are gestational ultrasound measures such as bi-parietal diameter (*bpd*), head circumference (*hc*), abdominal circumference (*ac*), fetal length (*fl*). Variable *Z* is called response variable or dependent variable. Each *Xi* is called regression variable, regressor, predictor, regression variable, or independent variable. Each *αi* is called regression coefficient. Researches focusing on the regression approach aim to estimate coefficients from gestational sample of ultrasound measures. For example, Hadlock et al. (Hadlock, Harrist, Sharman, Deter, & Park, 1985) proposed regression models for weight estimation based on head size, abdominal size, and femur length, which is better than those based on measurements of head and body. Error means in percentage of their models are 1.3%, 1.5%, 0.4%, 1.4%, 2.3%, and –0.7% whereas error standard deviations are 10.1%, 9.8%, 7.7%, 7.3%, 7.4%, 7.3%.

Phan (Phan, 1985) proposed some amazing regression formulas for estimating fetal age and weight based on *bpd*, *hc*, *ac*, abdominal area (*aa*), abdominal diameter (*ad*), average abdominal diameter (*aad*). Pham (Phạm, 2000) proposed some amazing regression formulas for estimating fetal weight based on *bpd*, ad, arm length (*al*), abdominal diameter (*ad*), average abdominal diameter (*aad*). Ho (Ho T. T., Nghiên Cứu Phương Pháp Ước Lượng Trọng Lượng Thai, Tuổi Thai Bằng Siêu Âm Hai và Ba Chiều, 2011) produced some amazing regression formulas for estimating fetal age and weight based on *bpd*, *ac*, *hc*, thigh volume, and thigh volume in her PhD dissertation. Some of Ho’s formulas (Ho T. T., Nghiên Cứu Phương Pháp Ước Lượng Trọng Lượng Thai, Tuổi Thai Bằng Siêu Âm Hai và Ba Chiều, 2011) are log(*weight*) = 1.746 + 0,0124\**bpd* + 0,001906\**ac* with R = 0.962, *weight* = –13099.1862 + 125.662\**ac* – 0.3818\**ac*\**ac* + 0.00045\**ac*\**ac*\**ac* with R = 0.9247, *weight* = –3306 + 55.477\**bpd* + 13.483\**thigh\_volume* with R = 0.9663, *age* = 167.0791 – 1,5537\**ac* + 0.00556\**ac*\**ac* – 0.00000618\**ac*\**ac*\**ac* with R = 0.8980, *age* = 331.0223 – 1.6118 \* (*hc* + *ac*) + 0.0028 \* (*hc* + *ac*) \* (*hc* + *ac*) – 0.0000015 \* (*hc* + *ac*) \* (*hc* + *ac*) \* (*hc* + *ac*) with R = 0.9212, *age* = 21.1148 + 0.2381 \* *thigh\_volume* – 0.001 \* *thigh\_volume* \* *thigh\_volume* + 0.000002 \* *thigh\_volume* \* *thigh\_volume* with R= 0,9959. Note that log(.) denotes logarithm function and R is correlation coefficient. The larger the R is, the better the formula is.

Deter, Rossavik, and Harrist (Deter, Rossavik, & Harrist, 1988) reassessed the weight estimation procedure of Rossavik (regression analysis) with particular emphasis on parameter estimation and performance over a wide weight range. As results, Deter, Rossavik, and Harrist assured that “there is no systematic errors over a 250 gram to 4750 gram weight range and random errors (± l standard deviation) of 10% to 13% below 200 gram and 6% to 8% above 2000 gram. The weights of small-and large-for-gestational age fetuses were systematically overestimated (4.1%) and underestimated (–3.0%), respectively, but systematic errors were not found in average-for-gestational age fetuses”.

Varol et al. (Varol, Saltik, Kaplan, Kilic, & Yardim, 2001) evaluated the growth curve of well-functioned regression models (Hadlock formulas, for example). Their purpose is to contribute to develop national standard growth curve of gestational age and birth weight. Percentile values and correlation coefficients were calculated and well-functioned regression models were produced for growth curve. As a result, the regression model for gestational age *age* = 4.945 + 0.606\**ac* + 0.105\**bpd* + 0.286\**fl* with adjusted R2 = 0.937 is optimal.

Salomon, Bernard, and Ville (Salomon, Bernard, & Ville, 2007) used polynomial regression approach to compute a new reference chart for weight estimation. Their resulted birth-weight chart showed that the weight estimation was noticeably larger at 25 – 36 weeks. At 28 – 32 weeks, the 50th centile of actual birth weight is approximated to the 50th centile of estimated weight.

A. R. Akinola, I. O. Akinola, and O. O. Oyekan (Akinola, Akinola, & Oyekan, 2009) evaluated many regression estimation models. Their results showed that models with *hc* and *ac* are not as good as those with *ac* and *bpd*. The combination of *fl* and *ac* did not improve accuracy. The use of multiple measures gives most accurate estimation.

Lee et al. (Lee, et al., 2009) used multiple linear regression model with standard measures (*bpd*, *fl*, *ac*) and their proposed biometrics such as fractional arm volume (*fav*) and fractional thigh volume (*ftv*). They produced six weight estimation models such as model 1, model 2, model 3, model 4, model 5, and model 6. The model 3 which is log(*weight*) = 0.5046 + 1.9665\*log(*bpd*) – 0.3040\*log(*bpd*)\*log(*bpd*) + 0.9675\*log(*ac*) + 0.3557\*log(*fav*) and model 6 which is log(*weight*) = –0.8297 + 4.0344\*log(*bpd*) – 0.7820\*log(*bpd*)\*log(*bpd*) + 0.7853\*log(*ac*) + 0.0528\*log(*ftv*)\*log(*ftv*) gain highest accuracy. Model 5 classified an additional 9.1% and 8.3% of fetuses within 5% and 10% of birth weight. Model 6 classified an additional 7.3% and 4.1% of infants within 5% and 10% of birth weight.

Bennini et al. (Bennini, Marussi, Barini, Faro, & Peralta, 2009) created a total of 210 pregnant women in their research into a formula-generating group (150 women) and prospective validation group (60 women). Polynomial regression is used to generate one formula based on two-dimension measures, one formula based on fetal thigh volume by multi-planar technique, and one formula based on fetal thigh volume by Virtual Organ Computer-aided Analysis. The experimental results showed that their models are significantly good and there is no significant difference between two-dimension model and three-dimension models. Note that their two-dimension model is *weight* = –562.824 + 11.962\**ac*\**fl* + 0.009\**bpd*\**bpd*\**ac*\**ac*. Their three-dimension models are *weight* = 1033.286 + 12.733\**thigh\_volume* and *weight* = 1025.383 + 12.775\* *thigh\_volume*.

Cohen et al. (Cohen, et al., 2010) used linear regression model to compare estimated weights for births after 6 days after last ultrasound scan and actual weights. Their results indicate that the mean ± standard deviation percentage among deliveries within 1 day of last ultrasound scan is 0.2 ± 9%.

Siggelkow et al. (Siggelkow, et al., 2010) proposed a new algorithm of isotonic regression to construct a birth weight prediction function that increases monotonically with each of input variables (ultrasound measures) and minimizes empirical quadratic loss. As a result, their isotonic regression function gains a small mean absolute error (312 gram).

Pinette et al. (Pinette, et al., 1999) used mean weight value from multiple formulas in order to improve the estimation. For instance, Pinette et al. calculated four estimated weight values *w*1, *w*2, *w*3, and *w*4 from formulas of Shepard, Hadlock, and Combs and then, they computed the mean *w* = (*w*1 + *w*2 + *w*3 + *w*4) / 4 as the optimal estimated value of birth weight.

When fetal weight is estimated based on gestational age, the weight-for-gestational chart is used. In such chart, if gestational age falls below 10th percentile then, it is impossible to estimate respective weight and so such problem is called small-for-gestational-age which often occurs because of missing data. Hutcheon and Platt (Hutcheon & Platt, 2008) applied standard epidemiologic approaches to correct the missing data problem. However such approaches does not use regression model. When gestational age is incompletely recorded, Eberg, Platt, and Filion (Eberg, Platt, & Filion, 2017) proposed four approaches to estimating missing gestational age: (1) generalized estimating equations for longitudinal data; (2) multiple imputation; (3) estimation based on fetal birth weight and sex; and (4) conventional approaches that assigned a fixed value (39 weeks for all or 39 weeks for full term and 35 weeks for preterm).

In general, most of researches required that the time point to take ultrasound measures is not too far from last ultrasound scan so that bias of actual birth weight at delivery time is small enough. If ultrasound examinations are taken soon, the gestational sample will lack weights because ultrasound measures do not conform to actual birth weight at delivery time. In the next section, we propose an algorithm which is an application of expectation maximization (EM) algorithm for estimating fetal weight in case of incomplete data. Here we should survey some researches related to how to apply EM algorithm into regression model. In literature of EM algorithm, missing values of data sample are estimated in expectation step and coefficients of regression model are re-estimated in maximization step. For example, Zhang, Deng, and Su (Zhang, Deng, & Su, 2016) used EM algorithm to build up linear regression model for studying glycosylated hemoglobin from partial missing data. In other words, they aim to discover relationship between independent variables (predictors) and diabetes. Therefore, the ideology of applying EM algorithm into regression model is not new but we propose a special technique to construct mutually two dual regression models at the same time from incomplete gestational sample. The algorithm that implements such technique is called dual regression expectation maximization (DREM) algorithm. DREM simplifies EM algorithm with assumption of normal distribution and moreover only weights in gestational sample are missing. DREM is described in next section.

**2. Methodology**

This research continues our previous research (Nguyen & Ho, A framework of fetal age and weight estimation, 2014) in which the used ultrasound samples are collected in fetal age from 28 weeks to 42 weeks because delivery time is not over 48 hours since last ultrasound scan. Hence, accuracy of weight estimation is only ensured when ultrasound examinations are performed after 28-week old fetal age. In the chapter “Phoebe Framework and Experimental Results for Estimating Fetal Age and Weight” of the book “E-Health” by Thomas F. Heston, we proposed an early weight estimation in which ultrasound measures can be taken before 28-week old fetal age. In this research, we implement such proposal as a so-called dual regression expectation maximization (DREM) algorithm and then make experiments on DREM. With DREM, the gestational sample can be totally collected at any appropriate time points in gestational period. In other words, the sample can lack fetal weights. This is a convenience for practitioners because they do not need to concern fetal weights when taking ultrasound examinations. Consequently, early weight estimation is achieved. As a convention, vectors are column vectors if there is no additional information.

Suppose we estimate two linear regression models *Z* = *α*0 *+ α*1*X*1 *+ α*2*X*2 *+ … + αnXn* and *Z* = *β*0 *+ β*1*Y* where *Z* is fetal weight and *Y* is fetal age whereas *Xi* (s) are gestational ultrasound measures such as *bpd*, *hc*, *ac*, and *fl*. Suppose both random variables *Y* and *Z* conform normal distribution, according to equation 1 (Lindsten, Schön, Svensson, & Wahlström, 2017, pp. 8-9). Note, only *Z* is random variable whereas *X* and *Y* are data.

|  |  |
| --- | --- |
|  | (1) |

Where, *α* = (*α*0, *α*1,…, *αn*)*T* and *β* = (*β*0, *β*1)*T* are parameter vectors where *X* = (1, *X*1, *X*2,…, *Xn*)*T* and (1, *Y*)*T* are data vectors. As a convention, linear function *Z = α*0 *+ α*1*X*1 *+ α*2*X*2 *+ … + αnXn* is called first function, first model, or first formula whereas linear function *Z =* *β*0 *+ β*1*Y* is called second function, second model, or second formula. Such two models are estimated in duality at the same time. The means of *Z* with regard to the first model and the second model are *αTX* and *βT*(1, *Y*)*T*, respectively whereas the variances of *Z* with regard to the first model and the second model are *σ*12 and *σ*22, respectively. Note that the superscript “*T*” denotes transposition operator in vector and matrix.

Let ***D*** = (***X***, ***y***, ***z***) be collected sample in which ***X*** is a set of sample measures, ***y*** is a set of sample fetal ages, and ***z*** is a set of fetal weights with note that ***z*** is empty or incomplete. If ***z*** is empty, there is no *zi* in ***z***. If ***z*** is incomplete, ***z*** has some values but there are also some missing values in ***z***. However, the constraint is that ***X*** is complete. Now we focus on estimating *α* and *β* based on ***D***. As a convention let *α\** and *β\** be estimates of *α* and *β*, respectively (Lindsten, Schön, Svensson, & Wahlström, 2017, p. 8).

Given *X* and *Y*, the entire probability of *Z* is defined product of *P*(*Z* | *X*, *α*) and *P*(*Z* | *Y*, *β*) according to equation 2.

|  |  |
| --- | --- |
|  | (2) |

Conditional expectation of sufficient statistic *Z* given *X* with regard to *P*(*Z* | *X*, *α*) is specified by equation 3.

|  |  |
| --- | --- |
|  | (3) |

Conditional expectation of sufficient statistic *Z* given *Y* with regard to *P*(*Z* | *Y*, *β*) is specified by equation 4.

|  |  |
| --- | --- |
|  | (4) |

Equation 2 indicates both explicit dependence via *P*(*Z* | *X*, *α*) and implicit dependence via *P*(*Z* | *Y*, *β*). There is a heuristic assumption that the explicit dependence and the implicit dependence share equal influence on *Z* if *E*(*Z* | *X*) specified by equation 3 is equal to *E*(*Z* | *Y*) specified by equation 4. Let *E*(*Z* | *X*, *Y*) be the expectation of *Z* with regard to the entire probability of Z specified by equation 2. If the heuristic assumption is satisfied, we have *E*(*Z* | *X*, *Y*) = *E*(*Z* | *X*) = *E*(*Z* | *Y*). This implies 2*E*(*Z* | *X*, *Y*) = *E*(*Z* | *X*) + *E*(*Z* | *Y*). Thus, from equations 3 and 4, we have equation 5 to specify the heuristic assumption.

|  |  |
| --- | --- |
|  | (5) |

Please pay attention to equation 5 because *Z* will be estimated by such expectation later.

For convenience, let Θ = (*α*, *β*)*T* be the compound parameter. The entire probability of *Z* given *X* and *Y* specified by equation 2 is re-written as follows:

(Due to all observations are independently and identically distributed)

The log-likelihood function is logarithm of the entire joint probability according to equation 6.

|  |  |
| --- | --- |
|  | (6) |

The optimal estimate Θ*\** is a maximizer of *L*(Θ), according to equation 7 (Lindsten, Schön, Svensson, & Wahlström, 2017, p. 9).

|  |  |
| --- | --- |
|  | (7) |

By taking first-order partial derivatives of *L*(Θ) with regard to Θ (Nguyen, Matrix Analysis and Calculus, 2015, p. 34), we obtain:

When first-order partial derivatives of *L*(Θ) are equal to zero, it gets locally maximal. In other words, Θ*\** is solution of the equation system 8 resulted from setting such derivatives to be zero.

|  |  |
| --- | --- |
|  | (8) |

The notation **0** = (0, 0,…, 0)*T* denotes zero vector. All equations in the system 8 are linear, whose unknowns are Θ = (*α*, *β*)*T*.

We apply expectation maximization (EM) algorithm into estimating Θ = (*α*, *β*)*T* with lack of fetal weights. Note that the entire probability *P*(*Z* | *X*, *Y*, *α*, *β*) specified by equation 2 is product of regular exponential distributions. EM algorithm has many iterations and each iteration has expectation step (E-step) and maximization step (M-step) for estimating parameters. Given current parameter Θ*t* = (*αt*, *βt*)*T* at the *t*th iteration, the two steps are shown in table 1 (Dempster, Laird, & Rubin, 1977, p. 4).

|  |  |  |
| --- | --- | --- |
| 1. E-step: Missing values *zi* (s) are estimated as expectations of themselves based on the current parameter Θ*t*, according to equation 5. 2. M-step: The next parameter Θ*t*+1 is a maximizer of *L*(Θ), which is the solution of equation system 8. The equation system 8 is solvable because missing values *zi* (s) were estimated in E-step. According to (Montgomery & Runger, 2010, pp. 456-459), the solution of system 8 at current iteration is specified by equation 9.  |  |  | | --- | --- | |  | (9) |   Where,  Note, Θ*t*+1 becomes current parameter for the next iteration. |

**Table 1.** E-step and M-step of EM algorithm

The EM algorithm stops if at some *t*th iteration, we have Θ*t* = Θ*t*+1 = Θ*\**. At that time, Θ*\** = (*α\**, *β\**)*T* is the optimal estimate of EM algorithm and hence the first model and the second model are determined with *α\** and *β\**. In practice, the algorithm can stop if the deviation between Θ*t* and Θ*t*+1 is smaller than a small enough terminated threshold. In this research such *terminated threshold* is *ε* = 0.001. The smaller the terminated threshold is, the more accurate the algorithm is.

The two steps shown in table 1, which is application of EM algorithm into linear regression model, is the proposed algorithm which is called dual regression expectation maximization (DREM) algorithm. The duality is implied by equation 5 which is used to estimate missing values *zi* in the E-step. If equation 3 is used to estimate *zi* for the first model and equation 4 is used to estimate *zi* for the second model then, dual option is turned off. If dual option is turned off but *yi* is missing then, equation 3 is used to estimate *zi* for both the first model and the second model. In general, DREM has two options such as dual and non-dual. As usual, dual option is default option of DREM. In other words, equation 5 is used to estimate missing values *zi* by default. Next section focuses on experimental results of DREM.

**3. Experimental results**

We uses the gestational sample of 127 cases in which each case includes ultrasound measures, fetus age, and fetus weight. Ultrasound measures are bi-parietal diameter (*bpd*), head circumference (*hc*), abdominal circumference (*ac*), and fetal length (*fl*). The unit of *bpd*, *hc*, *ac*, and *fl* is millimeter. The units of fetal age and fetal weight are week and gram, respectively. Ho and Phan (Ho & Phan, 2011), (Ho & Phan, 2011) collected the sample of pregnant women at Vinh Long General Hospital – Vietnam with obeying strictly all medical ethical criteria. These women and their husbands are Vietnamese. Their periods are regular and their last periods are determined. Each of them has only one alive fetus. Fetal age is from 28 weeks to 42 weeks. Delivery time is not over 48 hours since ultrasound scan.

Such sample of 127 cases is used as dataset to make experiment in this research. The dataset is split into two folders and each folder owns one training dataset and one testing dataset. Training dataset and testing dataset in the same folder are separated, which means that they do not have common cases. Training dataset is named *sample*{*i*}*.base* and testing dataset is named *sample*{*i*}.*test*. There are two training datasets and two testing datasets such as *sample1.base*, *sample1.test*, *sample2.base*, and *sample2.base*.

* Folder 1 has *sample1.base* of 513 cases and *sample1.test* of 514 cases.
* Folder 2 has *sample2.base* of 514 cases and *sample1.test* of 513 cases.

We do not have ultrasound sample which has examination cases whose delivery time is over 48 hours since ultrasound scan so as to evaluate early weight estimation in real time. Therefore, we make training datasets sparse in order to test DREM algorithm. Each training dataset is made sparse with sparse ratios (0.2, 0.4, 0.6, 0.8) into incomplete (missing) training datasets. Each incomplete training dataset is named *sample*{*i*}*.base.*{*sparse-ratio*}.*miss*. There are eight incomplete training datasets such as *sample1.base.0.2.miss*, *sample2.base.0.2.miss*, *sample1.base.0.4.miss*, *sample2.base.0.4.miss*, *sample1.base.0.6.miss*, *sample2.base.0.6.miss*, *sample1.base.0.8.miss*, *sample2.base.0.8.miss*. For instance, with sparse ratio 0.2, there are 0.2\*513 = 102 cases in *sample1.base.0.2.miss* which has no fetus weight; in other words, *sample1.base.0.2.miss* loses 20% weight values. We test DREM algorithm with regard to ten testing pairs of complete and incomplete training datasets and testing datasets according to table 2:

|  |  |  |  |
| --- | --- | --- | --- |
| Pair | Training dataset | Testing dataset | Sparse ratio |
| 1 | *sample1.base* | *sample1.test* | 0% |
| 2 | *sample2.base* | *sample2.test* | 0% |
| 3 | *sample1.base.0.2.miss* | *sample1.test* | 20% |
| 4 | *sample2.base.0.2.miss* | *sample2.test* | 20% |
| 5 | *sample1.base.0.4.miss* | *sample1.test* | 40% |
| 6 | *sample2.base.0.4.miss* | *sample2.test* | 40% |
| 7 | *sample1.base.0.6.miss* | *sample1.test* | 60% |
| 8 | *sample2.base.0.6.miss* | *sample2.test* | 60% |
| 9 | *sample1.base.0.8.miss* | *sample1.test* | 80% |
| 10 | *sample2.base.0.8.miss* | *sample2.test* | 80% |

**Table 2.** Ten testing pairs

Note, training datasets in the 1st and 2nd pairs are complete so that we can simulate DREM in early weight estimation. The 1st and 2nd pairs are called completed pairs whereas 3rd, 4th, 5th, 6th, 7th, 8th, 9th, and 10th are called incomplete pairs. Experimental results from incomplete pairs are compared together and are aligned with experimental results from complete pairs so as to evaluate DREM with subject to both early weight estimation and withstanding of DREM for missing values. The terminated condition in DREM is that the deviation between estimated coefficients and current coefficients is smaller than *ε* = 0.001.

DREM algorithm always results out two regression models on weight estimation. The *first model* estimates fetal weight based on ultrasound measures whereas the *second model* estimates fetal weight based on age. DREM algorithm supports two options such as dual and non-dual. With dual option, the first model and the second model are mutually dependent and so they are mutually improved. In other words, by dual option, missing weights *zi* (s) are estimated by equation 5 for both the first model and the second model. With non-dual option, the first model and second model are independent. By non-dual option, missing weights *zi* (s) are estimated separately by equation 3 for the first model and by equation 4 for the second model. Table 3 shows the resulted first model and second model with dual and non-dual options.

|  |  |  |  |
| --- | --- | --- | --- |
| Pair | Dual | First model | Second model |
| 1 | Dual | *weight* = –5728.6373 + 45.1344\**bpd* + 2.5591\**hc* + 18.9092\**fl* + 7.9899\**ac* | *weight* = –3820.1479 + 181.9784\**age* |
| Non | *weight* = –5728.6373 + 45.1344\**bpd* + 2.5591\**hc* + 18.9092\**fl* + 7.9899\**ac* | *weight* = –3820.1479 + 181.9784\**age* |
| 2 | Dual | *weight* = –5685.0575 + 47.2727\**bpd* + 1.3630\**hc* + 13.7749\*fl + 9.5887\**ac* | *weight* = –3970.1408 + 187.3782\**age* |
| Non | *weight* = –5685.0575 + 47.2727\**bpd* + 1.3630\**hc* + 13.7749\**fl* + 9.5887\**ac* | *weight* = –3970.1408 + 187.3782\**age* |
| 3 | Dual | *weight* = –5645.2017 + 40.1234\**bpd* + 3.3449\**hc* + 21.1281\**fl* + 7.8674\**ac* | *weight* = –3829.7426 + 182.3679\**age* |
| Non | *weight* = –5647.4091 + 42.5970\**bpd* + 2.5708\**hc* + 19.3682\**fl* + 8.3290\**ac* | *weight* = –3910.2029 + 184.7158\**age* |
| 4 | Dual | *weight* = –5484.4714 + 40.5679\**bpd* + 2.1867\**hc* + 16.2284\**fl* + 9.4833\**ac* | *weight* = –3911.1008 + 185.9331\**age* |
| Non | *weight* = –5624.3980 + 44.3252\**bpd* + 1.3158\**hc* + 17.9969\**fl* + 9.4010\**ac* | *weight* = –3950.7140 + 187.0223\**age* |
| 5 | Dual | *weight* = –5402.6674 + 33.6115\**bpd* + 3.4910\**hc* + 28.4572\**fl* + 7.1784\**ac* | *weight* = –3768.3736 + 180.5090\**age* |
| Non | *weight* = –5797.9253 + 44.7521\**bpd* + 2.2256\**hc* + 22.1571\**fl* + 7.9819\**ac* | *weight* = –3849.9583 + 182.8097\**age* |
| 6 | Dual | *weight* = –5326.6660 + 40.0232\**bpd* + 1.7101\**hc* + 14.6771\**fl* + 9.9256\**ac* | *weight* = –3848.9083 + 184.1776\**age* |
| Non | *weight* = –5620.4169 + 46.6362\**bpd* + 0.5378\**hc* + 15.2852\**fl* + 10.0876\**ac* | *weight* = –4032.1445 + 189.3419\**age* |
| 7 | Dual | *weight* = –5105.4503 + 30.0349\**bpd* + 3.7805\**hc* + 32.7607\**fl* + 5.9909\**ac* | *weight* = –3547.3692 + 174.0874\**age* |
| Non | *weight* = –5718.8815 + 45.8368\**bpd* + 1.1911\**hc* + 26.7732\**fl* + 7.4468\**ac* | *weight* = –3739.6380 + 179.7493\**age* |
| 8 | Dual | *weight* = –5062.8630 + 37.3794\**bpd* + 2.7958\**hc* + 18.2244\**fl* + 7.9488\**ac* | *weight* = –3556.9181 + 175.6628\**age* |
| Non | *weight* = –5706.1052 + 52.9702\**bpd* + 1.1067\**hc* + 13.6221\**fl* + 8.3838\**ac* | *weight* = –3900.9242 + 185.2818\**age* |
| 9 | Dual | *weight* = –4755.9374 + 21.3966\**bpd* + 5.2579\**hc* + 27.8786\**fl* + 6.7601\**ac* | *weight* = –3427.7113 + 170.3515\**age* |
| Non | *weight* = –5874.9784 + 51.8042\**bpd* + 1.4529\**hc* – 0.1563\**fl* + 11.7492\**ac* | *weight* = –4175.1662 + 191.7896\**age* |
| 10 | Dual | *weight* = –4381.3747 + 20.6602\**bpd* + 3.5326\**hc* + 22.6394\**fl* + 8.7967\**ac* | *weight* = –3363.3951 + 170.9688\**age* |
| Non | *weight* = –5545.0467 + 49.0603\**bpd* – 0.0241\**hc* + 5.3808\**fl* + 11.8793\**ac* | *weight* = –4320.6100 + 198.5153\**age* |

**Table 3.** The first model and second model with dual and non-dual options

When sample is complete with the 1st pair and the 2nd pair, DREM results the same models as least squares method (Montgomery & Runger, 2010, pp. 452-459) does. As a convention, let *f*{*i*}{*d*/*n*} and *g*{*i*}{*d*/*n*} be the first model and the second model with regard to the testing pair *i*th and dual/non-dual option. For example, *f*1*d* is the first model derived from the 1st testing pair with dual option whereas *g*2*n* is the second model derived from the 2nd testing pair with non-dual option. By looking up table 3, we have *f*1*d* = *weight* = –5728.6373 + 45.1344\**bpd* + 2.5591\**hc* + 18.9092\**fl* + 7.9899\**ac* and *g*2*n* = *weight* = –3970.1408 + 187.3782\**age*.

We analyze the experimental results by two ways:

* By the first analyzed way, so-called *test 1*, we analyze results of REM algorithm on the first model with both dual and non-dual options. This test evaluates how good the first model is.
* By the second analyzed way, so-called *test 2*, we analyze results of REM algorithm on the second model with both dual and non-dual options. This test evaluates how good the second model is.

For example, given mean absolute error (MAE), by the test 1, for each case in the test dataset *sample*{*i*}.*test*, we compute the deviation between the weight in such case and the estimated weight resulted from the first model *f*{*i*}{*d*/*n*} in order to calculate MAE. Let *U* = {*u*1, *u*2,…, *uK*} and *V* = {*v*1, *v*2,…, *vK*} be sets of actual weights and estimated weights, respectively. In test 1, *vi* (s) are values of *f*{*i*}{*d*/*n*}. In test 2, *vi* (s) are values of *g*{*i*}{*d*/*n*}. Equation 10 specifies MAE metric.

|  |  |
| --- | --- |
|  | (10) |

The smaller the MAE is, the more accurate the DREM is. Table 4 shows MAE metric which evaluates our models (formulas) with test 1 and test 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pair | Test 1  (Dual) | Test 1  (Non-dual) | Test 2  (Dual) | Test 2  (Non-dual) |
| 1 | 174.4489 | 174.4489 | 242.5407 | 242.5407 |
| 2 | 162.3012 | 162.3012 | 249.5374 | 249.5374 |
| 3 | 175.7541 | 174.4708 | 241.7887 | 240.8631 |
| 4 | 162.5601 | 163.0694 | 250.8867 | 250.4620 |
| 5 | 180.1436 | 175.1376 | 242.9584 | 242.3857 |
| 6 | 162.1598 | 163.8128 | 251.4266 | 250.0164 |
| 7 | 186.9349 | 177.8223 | 246.0096 | 242.8904 |
| 8 | 164.6054 | 161.9048 | 254.0628 | 249.5981 |
| 9 | 196.2846 | 173.3726 | 249.6362 | 243.6904 |
| 10 | 187.4117 | 169.2374 | 261.7162 | 255.2253 |
| Average | 175.2604 | 169.5578 | 249.0563 | 246.7210 |

**Table 4.** MAE of test 1 and test 2

Let *rMAE*{*t*}{*i*≥3}{*d*/*n*} be the bias ratio between MAE of test *t* and pair *i* with dual or non-dual option and MAE of test *t* and pair 1 if *i* odd or pair 2 if *i* even. For example, we have:

These bias ratios indicates withstanding of DREM for incomplete data. For instance, the value *rMAE*13*d* = 0.0075 implies that the accuracy of dual DREM decreases 0.75% when the completion of training dataset of 3rd pair in test 1 decreases 20%. The value *rMAE*24*n* = 0.0037 implies that the accuracy of non-dual DREM decreases 0.37% when the completion of training dataset of 4th pair in test 2 decreases 20%. The bias ratios of test 1 and pairs 3rd (20% missing values), 5th (40% missing values), 7th (60% missing values), and 9th (80% missing values) with dual option are 0.37%, 0.36%, 0.89%, and 1.2%. Similarly, the bias ratios of test 1 and pairs 3rd, 5th, 7th, and 9th with non-dual option are 0.51%, 0.27%, 2.4%, and 7.6%. It is concluded that such bias ratios are much smaller than percentages of missing values and so the withstanding of DREM for missing values is significant. For instance, we make a one-way paired t-test of *X* = {20%, 40%, 60%, 80%} and *Y* = {0.75%, 3.26%, 7.16%, 12.52%}. Given significant level 95%, the statistic *t*0 is:

Where,

Note that = 0.4408 and *sD* = 0.2078 are sample mean and sample standard deviation of *D*. Because the *t*0 is larger than the percentage point *t*0.05, 3 = 2.353, difference between the percentage of missing values and the percentage of decrease in accuracy of DREM is significant within test 1, odd pairs (3rd, 5th, 7th, 9th), and dual option. Table 5 shows paired t-tests for test 1 and test 2 with dual / non-dual options, given MAE metric and significant level 95%. We use odd pairs (even pairs) in a same group which is compared with the 1st pair (2nd pair) because odd pairs (even pairs) share the same testing dataset *sample1.test* (*sample2.test*).

|  |  |  |
| --- | --- | --- |
|  | *t*0 | Difference |
| Test 1 & dual  Odd pairs | 4.2433 | Significant |
| Test 1 & dual  Even pairs | 4.5371 | Significant |
| Test 1 & non-dual  Odd pairs | 3.8295 | Significant |
| Test 1 & non-dual  Even pairs | 3.9633 | Significant |
| Test 2 & dual  Odd pairs | 4.0109 | Significant |
| Test 2 & dual  Even pairs | 3.9973 | Significant |
| Test 2 & non-dual  Odd pairs | 3.9488 | Significant |
| Test 2 & non-dual  Even pairs | 3.9248 | Significant |

**Table 5.** Paired t-tests given MAE metric where *t*0.05, 3 = 2.353

From paired t-tests in table 5, it is asserted that the withstanding of DREM for missing values with regard to MAE metric is significant because the bias ratios with regard to MAE metric are much smaller than percentages of missing values.

We compare the mean (average) of MAE metric with regard to dual option and non-dual option. In both test 1 and test 2, DREM with dual option decreases MAE metric very little (–5.70 ≈ 175.2604 – 169.5578 in test 1 and 2.34 ≈ 249.0563 – 246.7210 in test 2). In other words, dual DREM does not improve both the first model and the second model with subject to MAE metric. However, the deviation of MAE between test 1 and test 2 with dual option (73.7959 = 249.0563 – 175.2604) is smaller than the deviation of MAE between test 1 and test 2 with non-dual option (77.1632 = 246.7210 – 169.5578). This implies that dual option makes trade-off between the first model and the second model. Moreover we will know later that dual option will also increase convergence of DREM by decreasing the number of iterations.

We evaluate DREM with RMSE metric. Equation 11 specifies RMSE metric.

|  |  |
| --- | --- |
|  | (11) |

The smaller the RMSE is, the more accurate the DREM is. Table 6 shows RMSE metric which evaluates our models (formulas) with test 1 and test 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pair | Test 1  (Dual) | Test 1  (Non-dual) | Test 2  (Dual) | Test 2  (Non-dual) |
| 1 | 237.5107 | 237.5107 | 312.8357 | 312.8357 |
| 2 | 213.9920 | 213.9920 | 315.2571 | 315.2571 |
| 3 | 238.9250 | 237.2812 | 312.2347 | 311.5486 |
| 4 | 214.5607 | 214.9004 | 316.0840 | 316.0899 |
| 5 | 244.1085 | 237.1247 | 313.2679 | 312.7044 |
| 6 | 216.1667 | 216.6830 | 315.9654 | 316.7301 |
| 7 | 254.0856 | 239.4893 | 317.0741 | 313.2865 |
| 8 | 219.3093 | 214.6079 | 316.1866 | 314.3541 |
| 9 | 266.0722 | 239.1404 | 321.6692 | 313.9642 |
| 10 | 242.2878 | 226.1129 | 323.6832 | 327.9767 |
| Average | 234.7019 | 227.6843 | 316.4258 | 315.4747 |

**Table 6.** RMSE of test 1 and test 2

Table 7 shows paired t-tests for test 1 and test 2 with dual / non-dual options, given RMSE metric and significant level 95%.

|  |  |  |
| --- | --- | --- |
|  | *t*0 | Difference |
| Test 1 & dual  Odd pairs | 4.2559 | Significant |
| Test 1 & dual  Even pairs | 4.3904 | Significant |
| Test 1 & non-dual  Odd pairs | 3.9138 | Significant |
| Test 1 & non-dual  Even pairs | 4.0118 | Significant |
| Test 2 & dual  Odd pairs | 3.9979 | Significant |
| Test 2 & dual  Even pairs | 3.9485 | Significant |
| Test 2 & non-dual  Odd pairs | 3.9210 | Significant |
| Test 2 & non-dual  Even pairs | 3.9905 | Significant |

**Table 7.** Paired t-tests given RMSE metric where *t*0.05, 3 = 2.353

From paired t-tests in table 7, it is asserted that the withstanding of DREM for missing values with regard to RMSE metric is significant because the bias ratios with regard to RMSE metric are much smaller than percentages of missing values.

We compare the mean (average) of RMSE metric with regard to dual option and non-dual option. In both test 1 and test 2, DREM with dual option does not decrease RMSE. Even though, DREM with dual option changes RMSE metric very little (7.02 ≈ 234.7019 – 227.6843 in test 1 and 0.95 ≈ 316.4258 – 315.4747 in test 2). In other words, dual DREM does not improve both the first model and the second model with subject to RMSE metric. However, dual option makes trade-off between the first model and the second model because the deviation of RMSE between test 1 and test 2 with dual option (81.7239 = 316.4258 – 234.7019) is smaller than the deviation of RMSE between test 1 and test 2 with non-dual option (87.7904 = 315.4747 – 227.6843).

We evaluate DREM with error ranges. Equation 12 specifies error range metric.

|  |  |
| --- | --- |
|  | (12) |

The smaller both error mean *μ* and error standard deviation *σ* are, the more accurate the DREM is. Table 8 shows error ranges which evaluate our models (formulas) with test 1 and test 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pair | Test 1  (Dual) | Test 1  (Non-dual) | Test 2  (Dual) | Test 2  (Non-dual) |
| 1 | –18.4183  ±  236.7933 | –18.4183  ±  236.7933 | –40.1656  ±  310.2449 | –40.1656  ±  310.2449 |
| 2 | 17.3781  ±  213.2829 | 17.3781  ±  213.2829 | 38.1153  ±  312.9429 | 38.1153  ±  312.9429 |
| 3 | –14.8301  ±  238.4622 | –15.8939  ±  236.7462 | –36.0420  ±  310.1459 | –33.8197  ±  309.7059 |
| 4 | 26.9210  ±  212.8628 | 28.0853  ±  213.0549 | 46.8130  ±  312.5966 | 45.1457  ±  312.8477 |
| 5 | –21.2098  ±  243.1833 | –15.0595  ±  236.6439 | –40.1361  ±  310.6845 | –40.7017  ±  310.0426 |
| 6 | 28.5077  ±  214.2763 | 25.5652  ±  215.1672 | 47.8504  ±  312.3195 | 44.5196  ±  313.5841 |
| 7 | –28.0630  ±  252.5291 | –28.1997  ±  237.8212 | –45.2751  ±  313.8235 | –38.1546  ±  310.9528 |
| 8 | 25.0713  ±  217.8692 | 23.6332  ±  213.3003 | 43.2167  ±  313.2176 | 34.3020  ±  312.4754 |
| 9 | –40.1527  ±  263.0232 | –11.0795  ±  238.8815 | –57.1823  ±  316.5442 | –49.6731  ±  310.0082 |
| 10 | 57.7590  ±  235.3004 | 44.6905  ±  221.6502 | 73.2199  ±  315.2914 | 75.6217  ±  319.1380 |

**Table 8.** Error ranges of test 1 and test 2

Because error range is not scalar, it is incorrect to make paired t-tests and so we cannot assert the withstanding of DREM for missing values with regard to error range. It is also incorrect to calculate the mean (average) of error ranges but we assert that dual DREM does not improve the first model and the second model with subject to error range because error means and error standard deviations of dual option are often larger than those of non-dual option.

We evaluate DREM with correlation coefficient (R). Equation 13 specifies R metric.

|  |  |
| --- | --- |
|  | (13) |

The correlation coefficient *R* reflects adequacy of a given formula. The larger the *R* is, the better the formula is. Table 9 shows metric R which evaluates our models (formulas) with test 1 and test 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pair | Test 1  (Dual) | Test 1  (Non-dual) | Test 2  (Dual) | Test 2  (Non-dual) |
| 1 | 0.9544 | 0.9544 | 0.9205 | 0.9205 |
| 2 | 0.9632 | 0.9632 | 0.9192 | 0.9192 |
| 3 | 0.9538 | 0.9544 | 0.9205 | 0.9205 |
| 4 | 0.9633 | 0.9633 | 0.9192 | 0.9192 |
| 5 | 0.9525 | 0.9543 | 0.9205 | 0.9205 |
| 6 | 0.9628 | 0.9626 | 0.9192 | 0.9192 |
| 7 | 0.9508 | 0.9539 | 0.9205 | 0.9205 |
| 8 | 0.9636 | 0.9631 | 0.9192 | 0.9192 |
| 9 | 0.9484 | 0.9540 | 0.9205 | 0.9205 |
| 10 | 0.9600 | 0.9605 | 0.9192 | 0.9192 |
| Average | 0.9573 | 0.9584 | 0.9198 | 0.9198 |

**Table 9.** R metric of test 1 and test 2

Table 10 shows paired t-tests for test 1 and test 2 with dual / non-dual options, given R metric.

|  |  |  |
| --- | --- | --- |
|  | *t*0 | Difference |
| Test 1 & dual  Odd pairs | 3.8614 | Significant |
| Test 1 & dual  Even pairs | 3.8610 | Significant |
| Test 1 & non-dual  Odd pairs | 3.8713 | Significant |
| Test 1 & non-dual  Even pairs | 3.8640 | Significant |
| Test 2 & dual  Odd pairs | 3.8730 | Significant |
| Test 2 & dual  Even pairs | 3.8730 | Significant |
| Test 2 & non-dual  Odd pairs | 3.8730 | Significant |
| Test 2 & non-dual  Even pairs | 3.8730 | Significant |

**Table 10.** Paired t-tests given R metric where *t*0.05, 3 = 2.353

From paired t-tests in table 10, it is asserted that the withstanding of DREM for missing values with regard to R metric is significant because the bias ratios with regard to R metric are much smaller than percentages of missing values.

We compare the mean (average) of R metric with regard to dual option and non-dual option. In both test 1 and test 2, DREM with dual option does not increase R. Even though, DREM with dual option almost does not changes R (–0.001 ≈ 0.9573 – 0.9584 in test 1 and 0 = 0.9198 – 0.9198 in test 2). In other words, dual DREM does not improve both the first model and the second model with subject to R metric. However, dual option makes trade-off between the first model and the second model because the deviation of R between test 1 and test 2 with dual option (0.0375 = 0.9573 – 0.9198) is smaller than the deviation of R between test 1 and test 2 with non-dual option (0.0386 = 0.9584 – 0.9198).

With regard to metrics such as MAE, RMSE, and R then, DREM surely withstands incomplete data. Its accuracy decreases insignificantly when the percentages of missing values increases significantly. However dual option does not improve DREM in accuracy for both the first model and the second model although duality is a feature of DREM. However, there is an interesting discovery under duality of DREM. Table 11 lists the number of iterations of DREM with regard to dual option and non-dual option.

|  |  |  |
| --- | --- | --- |
| Pair | Dual | Non-dual |
| 1 | 2 | 2 |
| 2 | 2 | 2 |
| 3 | 12 | 12 |
| 4 | 12 | 13 |
| 5 | 17 | 19 |
| 6 | 19 | 20 |
| 7 | 30 | 33 |
| 8 | 31 | 35 |
| 9 | 57 | 100 |
| 10 | 62 | 72 |

**Table 11.** The number of iterations of DREM with dual / non-dual option

From table 11, the number of iterations in dual option is smaller, which means that the convergence of DREM is improved with dual option. The reason is that DREM with dual option takes advantages prior information from both the first model and the second model in order to speed up the convergence. Especially, the more the sparse ratio (percentage of missing values) increases, the more dual option brings into play. The sample in this research is not huge and so such improvement is insignificant.

An alternative technique to improve the convergence of DREM is to initialize the parameter Θ1 = (*α*1, *β*1)*T* at the first iteration of EM process in proper way instead of initializing Θ1 in arbitrary way. Note, by default, Θ1 is initialized as zero vector. Let ***X***’ be the complete matrix of ultrasound measures, which is created by removing rows whose respective weights *zi* (s) are missing from ***X***. Similarly, let ***Y***’ be the complete matrix of fetal ages, which is created by removing rows whose respective weights *zi* (s) are missing from ***Y***. Let ***z***’ be the complete vector of non-missing weights. The advanced Θ1 is initialized by equation 14.

|  |  |
| --- | --- |
|  | (14) |

Equation 14, which is solution of the equation system 8, is also a variant of equation 9 where ***X***, ***Y***, and ***z*** are replaced by ***X***’, ***Y***’, and ***z***’. Note, ***X***, ***X***’, ***Y***’, and ***z***’ are complete whereas ***Y*** and ***z*** can be incomplete. Table 12 lists the number of iterations of DREM with advanced Θ1 and terminated threshold *ε* = 0.001.

|  |  |  |
| --- | --- | --- |
| Pair | Dual | Non-dual |
| 1 | 1 | 1 |
| 2 | 1 | 1 |
| 3 | 7 | 1 |
| 4 | 9 | 1 |
| 5 | 14 | 1 |
| 6 | 15 | 1 |
| 7 | 23 | 1 |
| 8 | 25 | 1 |
| 9 | 42 | 1 |
| 10 | 44 | 1 |

**Table 12.** The number of iterations of DREM with advanced initialized parameter

From comparing table 12 with table 11, it is asserted that the advanced initialized parameter Θ1 improves the convergence of DREM. The interesting result is that non-dual option seems to be preeminent in speeding up DREM with advanced Θ1 whereas it is worse option with arbitrary Θ1. Now we test DREM with advanced Θ1 and very small terminated threshold *ε* = 10–10.

|  |  |  |
| --- | --- | --- |
| Pair | Dual | Non-dual |
| 1 | 1 | 1 |
| 2 | 1 | 1 |
| 3 | 18 | 3 |
| 4 | 19 | 6 |
| 5 | 31 | 8 |
| 6 | 34 | 8 |
| 7 | 53 | 13 |
| 8 | 58 | 15 |
| 9 | 107 | 40 |
| 10 | 110 | 27 |

**Table 13.** The number of iterations of DREM with advanced initialized parameter and very small terminated threshold

From table 13, there is no doubt that non-dual option is preeminent in speeding up DREM with advanced Θ1. In general, dual option is only useful if researchers want to build up two acceptable (mutual) regression models because dual option makes trade-off between the first model and the second model. Recall that the essence of DREM is to build up two mutual regression models in duality but scientists can turn off such dual option.

**4. Conclusions**

The analysis of experiments proves efficiency of DREM in withstanding sparse dataset but this good result is derived from preeminence of EM algorithm when EM algorithm estimates missing values by sufficient statistic. The duality of DREM currently is not an excellent feature although it improves the convergence of DREM. Note, in literature of EM, the convergence of EM will be improved with support of additional information like prior probability. So, the first model of DREM gives additional information to the second model and vice versa. In general, the ideology of DREM does not go beyond EM algorithm but DREM solves effectively the problem of incomplete sample, which in turn results out the early weight estimation in obstetrics. In this research, only weight values are missing. In the future, we will improve DREM to solve a hazard problem in which fetal weight, fetal ages, and ultrasound measures can be missing. We may also introduce another algorithm different from DREM which is also another implementation of the proposal mentioned in the chapter “Phoebe Framework and Experimental Results for Estimating Fetal Age and Weight” of the book “E-Health” by Thomas F. Heston. If such hazard problem is solved successfully, practitioners will have a lot of benefits when they will not be stressful in taking ultrasound examinations because some measures are allowed to be missing. In other words, it is acceptable for practitioners to make unintentional mistakes when taking ultrasound examinations. Moreover researchers also get benefits because they can receive estimation models from incomplete sample. In literature of EM algorithm, there are methods to estimate regression model with lack of some independent variables and so the improvement of DREM is feasible.