Learning dyadic data and predicting unaccomplished co-occurrence values by mixture model

Loc Nguyen
Independent scholar, 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 sparseness 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 $f(X, Y | \Theta)$ where $\Theta$ is parameter. Given sample $\mathcal{Y} = \{Y_1, Y_2, \ldots, Y_N\}$ whose all $Y_i$ (s) are mutually independent and identically distributed (iid), it is required to estimate $\Theta$ based on $\mathcal{Y}$ whereas $X$ is unknown. Expectation maximization (EM) algorithm is applied to solve this problem when only $\mathcal{Y}$ 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 $\Theta^{(t)}$, the two steps are described as follows:

| E-step: |
|---|
| The expectation $Q(\Theta | \Theta^{(t)})$ is determined based on current parameter $\Theta^{(t)}$, according to equation 1.1 (Nguyen, 2020, p. 50). |

$$Q(\Theta | \Theta^{(t)}) = \sum_{i=1}^{N} \int f(X|Y_i, \Theta) \log(f(X, Y_i|\Theta'))dX$$

| M-step: |
|---|
| The next parameter $\Theta^{(t+1)}$ is a maximizer of $Q(\Theta | \Theta^{(t)})$ with subject to $\Theta$. Note that $\Theta^{(t+1)}$ 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 $\Theta^{(t)} = \Theta^{(t+1)} = \Theta^*$. Note, the estimate $\Theta^*$ 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 $\{1, 2, \ldots, K\}$. As a convention, let $k=X$. Note, because all $Y_i$ (s) are iid, let random variable $Y$ represent every $Y_i$. The so-called probabilistic finite mixture model is represented by the PDF of $Y$, as follows:
\[ f(Y|\Theta) = \sum_{k=1}^{K} \alpha_k f_k(Y|\theta_k) \quad (1.2) \]

Where,
\[ \Theta = (\alpha_1, \alpha_2, ..., \alpha_K, \theta_1, \theta_2, ..., \theta_K)^T \]
\[ \sum_{k=1}^{K} \alpha_k = 1 \]

Note, the superscript “\(^T\)” denotes transpose operator for vector and matrix. The \(Q(\Theta | \Theta^{(t)})\) is re-defined for finite mixture model as follows (Nguyen, 2020, p. 79):
\[ Q(\Theta | \Theta^{(t)}) = \sum_{i=1}^{N} \sum_{k=1}^{K} P(k|Y_i, \Theta^{(t)}) \log(\alpha_k f_k(Y_i|\theta_k)) \quad (1.3) \]

Where,
\[ P(k|Y_i, \Theta^{(t)}) = \frac{\alpha_k^{(t)} f_k(Y_i|\theta_k^{(t)})}{\sum_{i=1}^{K} \alpha_i^{(t)} f_i(Y_i|\theta_i^{(t)})} \quad (1.4) \]

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-occurring 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 \(\mathcal{X} = \{x_1, x_2, ..., x_N\}\) and \(\mathcal{Y} = \{y_1, y_2, ..., y_M\}\) with note that \(x_i\) (s) and \(y_j\) (s) represent \(\mathcal{X}\)-objects and \(\mathcal{Y}\)-objects, respectively; exactly, they are names of objects. The numbers of \(\mathcal{X}\)-objects and \(\mathcal{Y}\)-objects are \(|\mathcal{X}|=N\) and \(|\mathcal{Y}|=M\), respectively. For example, in information retrieval, \(x_i\) (s) are documents and \(y_j\) (s) are keywords. Hence, \(x_i\) and \(y_j\) are not evaluated as numbers. An observational pair \((x_i, y_j)\) \(\in \mathcal{X} \times \mathcal{Y}\) is called a co-occurrence of \(x_i\) and \(y_j\). Dyadic data or COD \(\mathcal{S}\) contains these co-occurrences with note that a co-occurrence \((x_i, y_j)\) can exist more than one time. So, each co-occurrence \((x_i, y_j)\) is indexed by an index \(r\). As a result, each co-occurrence is denoted by the triple \((x_i, y_j, r)\) and we have (Hofmann & Puzicha, 1998, p. 1):
\[ \mathcal{S} = \{ (x_i, y_j, r): 1 \leq r \leq |\mathcal{S}| \} \quad (2.1) \]

Where,
\[ x_i \in \mathcal{X} = \{ x_1, x_2, ..., x_{|\mathcal{X}|} \} \]
\[ y_j \in \mathcal{Y} = \{ y_1, y_2, ..., y_{|\mathcal{Y}|} \} \]

Of course, the size of \(\mathcal{S}\) is \(|\mathcal{S}|\). As a convention, \(x_i(r)\) and \(y_j(r)\) indicate that \(\mathcal{X}\)-object and \(\mathcal{Y}\)-object at the \(r^{th}\) co-occurrence are \(x_i\) and \(y_j\), respectively. Thus, the triplet \((x_i, y_j, r)\) can be denoted as \((x_i(r), y_j(r), r)\). For example, suppose \(\mathcal{X} = \{x_1, x_2, x_3\}\) and \(\mathcal{Y} = \{y_1, y_2\}\), and dyadic data of 4 co-occurrences, \(\mathcal{S} = \{ (x_1, y_1, 1), (x_1, y_1, 2), (x_1, y_2, 3), (x_1, y_1, 4) \}\), 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 \( X \)-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 \( Y \)-object at this co-occurrence is \( y_2 \).

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

\[
\mathcal{S} = \{(x_i, y_j, Z, r): 1 \leq r \leq |\mathcal{S}|\}
\]  
(2.2)

Where,
\[
x_i \in X = \{x_1, x_2, ..., x_{|X|}\}
\]
\[
y_j \in Y = \{y_1, y_2, ..., y_{|Y|}\}
\]

As a convention, \( Z(r) \) or \( z(r) \) indicates that the associative value at \( r \)th co-occurrence is \( Z = z \). Thus, the quadruplet \((x_i, y_j, Z, r)\) can be denoted as \((x(r), y_j(r), Z(r), r)\). For example, suppose \( X = \{x_1, x_2, x_3\} \) and \( Y = \{y_1, y_2\} \), and dyadic sample of 4 co-occurrences, \( \mathcal{S} = \{(x_1, y_1, 6, 1), (x_1, y_1, 8, 2), (x_1, y_2, 7, 3), (x_1, y_1, 9, 4)\} \), 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 \( x_k \), let \( \mathcal{S}_{x_k} \) be the \( X \)-partitioned subset of \( \mathcal{S} \) which contains co-occurrences whose \( X \)-objects are fixed at \( x_k \) (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 1). Note, \( \mathcal{S}_{x_k} \) can be empty. The size of \( \mathcal{S}_{x_k} \) is \(|\mathcal{S}_{x_k}|\).

\[
\mathcal{S}_{x_k} = \{(x_i, y_j, Z, r): x_i = x_k\}
\]  
(2.3)

Dyadic data \( \mathcal{S} \) is partitioned into \(|X|\) subsets \( \mathcal{S}_{x_k} \).

\[
\mathcal{S} = \bigcup_{k=1}^{\mid X \mid} \mathcal{S}_{x_k}
\]

\( \forall i \neq j, \mathcal{S}_{x_i} \cap \mathcal{S}_{x_j} = \emptyset \)

Given fixed \( y_l \), let \( \mathcal{S}_{y_l} \) be the \( Y \)-partitioned subset of \( \mathcal{S} \) which contains co-occurrences whose \( Y \)-objects are fixed at \( y_l \). Note, \( \mathcal{S}_{y_l} \) can be empty. The size of \( \mathcal{S}_{y_l} \) is \(|\mathcal{S}_{y_l}|\).

\[
\mathcal{S}_{y_l} = \{(x_i, y_j, Z, r): y_j = y_l\}
\]  
(2.4)

Dyadic data \( \mathcal{S} \) is partitioned into \(|Y|\) subsets \( \mathcal{S}_{y_l} \).

\[
\mathcal{S} = \bigcup_{l=1}^{\mid Y \mid} \mathcal{S}_{y_l}
\]

\( \forall i \neq j, \mathcal{S}_{y_i} \cap \mathcal{S}_{y_j} = \emptyset \)

Given fixed \( x_k \) and fixed \( y_l \), let \( \mathcal{S}_{x_k y_l} \) be the subset of the \( \mathcal{S} \) which contains co-occurrences whose \( X \)-objects and \( Y \)-objects are fixed at \( x_k \) and \( y_l \). Note, \( \mathcal{S}_{x_k y_l} \) can be empty. The size of \( \mathcal{S}_{x_k y_l} \) is \(|\mathcal{S}_{x_k y_l}|\).

\[
\mathcal{S}_{x_k y_l} = \{(x_i, y_j, Z, r): x_i = x_k, y_j = y_l\}
\]  
(2.5)

Let \( n(x_i) \) and \( n(y_j) \) denote the number of \( x_i \) and the number of \( y_j \), respectively.

\[
n(x_i) = |\mathcal{S}_{x_i}|
\]
\[
n(y_j) = |\mathcal{S}_{y_j}|
\]  
(2.6)
Let \( n(x_i, y_j) \) denote the number of co-occurrences \((x_i, y_j)\).

\[
n(x_i, y_j) = |S_{x_iy_j}|	ag{2.7}
\]

Let \( n(x_i|y_j) \) and \( n(y_j|x_i) \) denote the frequency of \( x_i \) given \( y_j \) and the frequency of \( y_j \) given \( x_i \), respectively.

\[
n(x_i|y_j) = \frac{n(x_i, y_j)}{n(y_j)}
\]

\[
n(y_j|x_i) = \frac{n(x_i, y_j)}{n(x_i)}
\]

For example, suppose \( X = \{x_1, x_2, x_3\} \) and \( Y = \{y_1, y_2\} \), and dyadic data of 4 co-occurrences, \( S = \{(x_1, y_1, 1), (x_1, y_1, 2), (x_1, y_2, 3), (x_1, y_1, 4)\} \), we have \( S_{x_1} = \{(x_1, y_1, 1), (x_1, y_1, 2), (x_1, y_1, 4)\} \), \( S_{x_2} = \emptyset \), \( S_{y_1} = \{(x_1, y_1, 1), (x_1, y_1, 2)\} \), \( S_{y_2} = \{(x_1, y_2, 3)\} \), \( S_{x_1y_1} = \{(x_1, y_1, 1), (x_1, y_1, 2), (x_1, y_1, 4)\} \), \( S_{x_1y_2} = \emptyset \), \( S_{x_2y_1} = \emptyset \), \( S_{x_2y_2} = \emptyset \), \( n(x_1) = 1 \), \( n(x_2) = 0 \), \( n(y_1) = 3 \), \( n(y_2) = 1 \), \( n(x_1, y_1) = 3 \), \( n(x_1, y_2) = 1 \), \( n(x_2, y_1) = n(x_2, y_2) = 0 \), \( n(x_3, y_1) = n(x_3, y_2) = 0 \), \( n(x_1 | y_1) = 1 \), \( n(x_1 | y_2) = 1 \), \( n(x_2 | y_1) = n(x_2 | y_2) = n(x_3 | y_1) = n(x_3 | y_2) = 0 \), \( n(y_1 | x_1) = 3/4 \), \( n(y_2 | x_1) = 1/4 \).

Suppose each co-occurrence \((x_i, y_j)\) belongs to a latent variable \( C \) and \( C \) has \( K \) values \( c_k \) (s). These values \( c_k \) (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 \( c_k = k \) where \( k = 1, 2, \ldots, K \). The random variable \( C \) has discrete distribution such that every value has an associated probability \( \alpha_k \). Of course, there are \( K \) probabilities \( \alpha_k \) (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 \( \alpha_k \) (s) are independent from both \( x_i \) and \( y_j \). SMM is defined as follows (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 2):

\[
P(x_i, y_j | \Theta) = \sum_{k=1}^{K} \alpha_k p(x_i, y_j | k) = \sum_{k=1}^{K} \alpha_k p_{i|k} q_{j|k}
\]

Where \( \alpha_k \) is the probability of aspect \( k \). Note, \( P(.) \) denote probability.

\[
\alpha_k = P(k)
\]

The \( p_{i|k} \) is the probability of \( x_i \) given aspect \( k \).

\[
p_{i|k} = P(x_i | k)
\]

The \( q_{j|k} \) is the probability of \( y_j \) given aspect \( k \).

\[
q_{j|k} = P(y_j | k)
\]

This implies that \( x_i \) and \( y_j \) are mutually independent in SMM.

\[
P(x_i, y_j | k) = P(x_i | k) P(y_j | k)
\]

The joint probability of \( x_i, y_j, \) and \( k \) is:

\[
P(x_i, y_j, k) = P(k) P(x_i, y_j | k) = \alpha_k p(x_i | k) p(y_j | k) = \alpha_k p_{i|k} q_{j|k}
\]

The parameter of SMM is \( \Theta = (\alpha_k, p_{i|k}, q_{j|k})^T \) in which there are \( K(|X| + |Y| + 1) \) partial parameters \( \alpha_k, p_{i|k}, \) and \( q_{j|k} \). Note,

\[
\sum_{k=1}^{K} \alpha_k = 1, \sum_{i=1}^{|X|} p_{i|k} = 1, \sum_{j=1}^{|Y|} q_{j|k} = 1
\]

By applying GEM, given dyadic sample \( S \), at the \( e \)th iteration of GEM, given current parameter \( \Theta^{(e)} = (\alpha_k^{(e)}, p_{i|k}^{(e)}, q_{j|k}^{(e)})^T \), the conditional expectation \( Q(\Theta | \Theta^{(e)}) \) is:
\[ Q(\Theta|\Theta^{(t)}) = \sum_{r=1}^{[|X|]} \sum_{k=1}^{[|Y|]} P(k|x_i(r),y_j(r),\Theta^{(t)}) \log(\alpha_k p_{ij|k} q_{jk}) \]

\[ = \sum_{i=1}^{[|X|]} \sum_{j=1}^{[|Y|]} n(x_i,y_j) \sum_{k=1}^{K} P(k|x_i,y_j,\Theta^{(t)}) \left( \log(\alpha_k) + \log(p_{ij|k}) \right) + \log(q_{jk}) \]  

(2.10)

Where,

\[ P(k|x_i,y_j,\Theta^{(t)}) = \frac{\alpha_k^{(t)} p_{ij|k}^{(t)} q_{jk}^{(t)}}{\sum_{l=1}^{K} \alpha_l^{(t)} p_{ij|l}^{(t)} q_{jk}^{(t)}} \]  

(2.11)

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

\[ \sum_{k=1}^{K} \alpha_k = 1, \sum_{i=1}^{[|X|]} p_{ij|k} = 1, \sum_{j=1}^{[|Y|]} q_{jk} = 1 \]

We use Lagrange duality method to maximize to maximize \(Q(\Theta|\Theta^{(t)})\). The Lagrange function \(la(\Theta, \lambda | \Theta^{(t)})\) is sum of \(Q(\Theta|\Theta^{(t)})\) and these constraints, as follows:

\[ la(\Theta, \lambda | \Theta^{(t)}) = Q(\Theta|\Theta^{(t)}) + \lambda_1 \left( 1 - \sum_{k=1}^{K} \alpha_k \right) + \lambda_2 \left( 1 - \sum_{i=1}^{[|X|]} p_{ij|k} \right) + \lambda_3 \left( 1 - \sum_{j=1}^{[|Y|]} q_{jk} \right) \]

\[ = \sum_{i=1}^{[|X|]} \sum_{j=1}^{[|Y|]} n(x_i,y_j) \sum_{k=1}^{K} P(k|x_i,y_j,\Theta^{(t)}) \left( \log(\alpha_k) + \log(p_{ij|k}) + \log(q_{jk}) \right) + \lambda_1 \left( 1 - \sum_{k=1}^{K} \alpha_k \right) + \lambda_2 \left( 1 - \sum_{i=1}^{[|X|]} p_{ij|k} \right) + \lambda_3 \left( 1 - \sum_{j=1}^{[|Y|]} q_{jk} \right) \]

Note, \( \lambda = (\lambda_1, \lambda_2, \lambda_3)^T \) where \( \lambda_1 \geq 0, \lambda_2 \geq 0, \) and \( \lambda_3 \geq 0 \) are called Lagrange multipliers. Of course, \( la(\Theta, \lambda | \Theta^{(t)}) \) is function of \( \Theta \) and \( \lambda \). The next parameters \( \Theta^{(t+1)} \) that maximizes \( Q(\Theta|\Theta^{(t)}) \) at \( M \)-step of some \( l \)th iteration is solution of the equation formed by setting the first-order partial derivatives of Lagrange function regarding \( \Theta \) and \( \lambda \) to be zero.

The first-order partial derivative of Lagrange function regarding \( \alpha_k \) is:

\[ \frac{\partial la(\Theta, \lambda | \Theta^{(t)})}{\partial \alpha_k} = \sum_{i=1}^{[|X|]} \sum_{j=1}^{[|Y|]} n(x_i,y_j) \frac{1}{\alpha_k} P(k|x_i,y_j,\Theta^{(t)}) - \lambda_1 \]

Setting this partial derivative to be zero, we obtain:

\[ \sum_{i=1}^{[|X|]} \sum_{j=1}^{[|Y|]} n(x_i,y_j) P(k|x_i,y_j,\Theta^{(t)}) - \alpha_k \lambda_1 = 0 \]

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

\[ \sum_{i=1}^{[|X|]} \sum_{j=1}^{[|Y|]} n(x_i,y_j) \sum_{k=1}^{K} P(k|x_i,y_j,\Theta^{(t)}) - \lambda_1 \sum_{k=1}^{K} \alpha_k = 0 \]

\[ \Leftrightarrow \sum_{i=1}^{[|X|]} \sum_{j=1}^{[|Y|]} n(x_i,y_j) - \lambda_1 = 0 \Leftrightarrow \lambda_1 = \sum_{i=1}^{[|X|]} \sum_{j=1}^{[|Y|]} n(x_i,y_j) \]

This means the next parameters \( \alpha_k^{(t+1)} \) is:
\[ a_k^{(t+1)} = \frac{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j)} \]  
(2.12)

The first-order partial derivative of Lagrange function regarding \( p_{ik} \) is:

\[ \frac{\partial \ln(\Theta, \lambda | \Theta^{(t)})}{\partial p_{ik}} = \sum_{j=1}^{|Y|} n(x_i, y_j) \frac{1}{p_{ik}} P(k | x_i, y_j, \Theta^{(t)}) - \lambda_2 \]

Setting this partial derivative to be zero, we obtain:

\[ \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)}) - p_{ik} \lambda_2 = 0 \]

Summing the equation above over \( X \), we have:

\[ \sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)}) - \lambda_2 \sum_{i=1}^{|X|} p_{ik} = 0 \]

\[ \Rightarrow \lambda_2 = \sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)}) \]

This means the next parameters \( p_{ik}^{(t+1)} \) is:

\[ p_{ik}^{(t+1)} = \frac{\sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})} \]  
(2.13)

Similarly, the next parameters \( q_{jk}^{(t+1)} \) is:

\[ q_{jk}^{(t+1)} = \frac{\sum_{i=1}^{|X|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})} \]  
(2.14)

The two steps of GEM algorithm for SMM at some \( r^{th} \) iteration are shown in table 2.1.

| E-step: |
| --- |
| The conditional probability \( P(k | x_i, y_j, \Theta^{(t)}) \) is calculated based on current parameter \( \Theta^{(t)} = (\alpha_k^{(t)}, p_{ik}^{(t)}, q_{jk}^{(t)}) \), according to equation 2.11. |

\[ P(k | x_i, y_j, \Theta^{(t)}) = \frac{\alpha_k^{(t)} p_{ik}^{(t)} q_{jk}^{(t)}}{\sum_{i=1}^{K} \alpha_i^{(t)} p_{i|k}^{(t)} q_{jk}^{(t)}} \]

| M-step: |
| --- |
| The next parameter \( \Theta^{(t+1)} = (\alpha_k^{(t+1)}, p_{ik}^{(t+1)}, q_{jk}^{(t+1)}) \), which is a maximizer of \( Q(\Theta | \Theta^{(t)}) \) with subject to \( \Theta \), is calculated by equation 2.12, equation 2.13, and equation 2.14. |

\[ \alpha_k^{(t+1)} = \frac{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j)} \]

\[ p_{ik}^{(t+1)} = \frac{\sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})} \]

\[ q_{jk}^{(t+1)} = \frac{\sum_{i=1}^{|X|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) P(k | x_i, y_j, \Theta^{(t)})} \]

| Table 2.1. E-step and M-step of GEM algorithm for SMM |
| --- |
| GEM algorithm converges at some \( r^{th} \) iteration. At that time, \( \Theta^* = \Theta^{(t+1)} = \Theta^{(t)} \) is the SMM itself. When SMM is applied into soft clustering, dyadic data is clustered according to blocks and each \( \alpha_k \) is coverage ratio of cluster \( k \) (aspect \( k \)). |
The mixture model of dyadic data is called asymmetric mixture model (AMM) if \( \alpha_k(\mathbf{s}) \) are only independent from \( x_i \) or from \( y_j \). Without loss of generality, given \( \alpha_k(\mathbf{s}) \) are only independent from \( y_j \) (of course, it is dependent on \( x_i \)), AMM is defined as follows (Hofmann & Puzicha, Statistical Models for Co-occurrence Data, 1998, p. 3):

\[
P(x_i, y_j | \Theta) = p_i q_j \sum_{k=1}^{K} \alpha_{k|i} q_{j|k}
\]  

(2.15)

The \( \alpha_{k|i} \) is the probability of aspect \( k \) given \( x_i \).

\[
\alpha_{k|i} = P(k|x_i)
\]

Where \( p_i \) is the probability of \( x_i \).

\[
p_i = P(x_i)
\]

The \( q_{j|k} \) is the conditional probability of \( y_j \) given aspect \( k \). Suppose \( y_j \) is dependent from \( x_i \) given \( k \), we have:

\[
q_{j|k} = P(y_j|x_i, k) = P(y_j|k)
\]

Note, \( q_{j|i} \) is the conditional probability of \( y_j \) given \( x_i \), which is defined as follows:

\[
q_{j|i} = P(y_j|x_i) = \sum_{k=1}^{K} \alpha_{k|i} q_{j|k}
\]

The joint probability of \( x_i, y_j \), and \( k \) is:

\[
P(x_i, y_j, k) = P(x_i)P(y_j, k|x_i) = P(x_i)P(k|x_i)P(y_j|x_i, k) = p_i \alpha_{k|i} P(y_j|k) = p_i \alpha_{k|i} q_{j|k}
\]

The parameter of AMM is \( \Theta = (\alpha_{k|i}, p_i, q_{j|k})^T \) in which there are \( K(|X| + |Y|) + |X| \) partial parameters \( \alpha_{k|i}, p_i \), and \( q_{j|k} \). Note,

\[
\sum_{k=1}^{K} \alpha_{k|i} = 1, \sum_{i=1}^{|X|} p_i = 1, \sum_{j=1}^{|Y|} q_{j|k} = 1
\]

By applying GEM, given dyadic sample \( \mathcal{S} \), at the \( t \)th iteration of GEM, given current parameter \( \Theta^{(t)} = (\alpha^{(t)}_{k|i}, p^{(t)}_i, q^{(t)}_{j|k})^T \), the conditional expectation \( Q(\Theta | \Theta^{(t)}) \) is:

\[
Q(\Theta | \Theta^{(t)}) = \sum_{r=1}^{[\mathcal{S}]} \sum_{k=1}^{K} P(k|x_i(r), y_j(r), \Theta^{(t)}) \log(\alpha_{k|i} p_i q_{j|k})
\]

\[
= \sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) \sum_{k=1}^{K} P(k|x_i, y_j, \Theta^{(t)}) \left( \log(\alpha_{k|i}) + \log(p_i) + \log(q_{j|k}) \right)
\]  

(2.16)

Where,

\[
P(k|x_i, y_j, \Theta^{(t)}) = \frac{\alpha_{k|i}(t) p_i(t) q_{j|k}(t)}{\sum_{i=1}^{K} \alpha_{k|i}(t) p_i(t) q_{j|k}(t)}
\]  

(2.17)

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

\[
\sum_{k=1}^{K} \alpha_{k|i} = 1, \sum_{i=1}^{|X|} p_i = 1, \sum_{j=1}^{|Y|} q_{j|k} = 1
\]

We use Lagrange duality method to maximize to maximize \( Q(\Theta | \Theta^{(t)}) \). The Lagrange function \( l_a(\Theta, \lambda | \Theta^{(t)}) \) is sum of \( Q(\Theta | \Theta^{(t)}) \) and these constraints, as follows:
The first-order partial derivative of Lagrange function regarding Θ and λ is:

\[
\frac{\partial l_a(\Theta, \lambda | \Theta^{(t)})}{\partial \alpha_k} = \sum_{j=1}^{|Y|} n(x_i, y_j) \frac{1}{\alpha_k} p(k|x_i, y_j, \Theta^{(t)}) - \lambda_1
\]

Setting this partial derivative to be zero, we obtain:

\[
\sum_{j=1}^{|Y|} n(x_i, y_j) p(k|x_i, y_j, \Theta^{(t)}) - \alpha_k \lambda_1 = 0
\]

Summing the equation above over \( K \) aspects \{1, 2, ..., \( K \)\}, we have:

\[
\sum_{j=1}^{|Y|} n(x_i, y_j) \sum_{k=1}^{K} p(k|x_i, y_j, \Theta^{(t)}) - \lambda_1 \sum_{k=1}^{K} \alpha_k = 0
\]

⇔ \( \sum_{j=1}^{|Y|} n(x_i, y_j) - \lambda_1 = 0 \) ⇔ \( \lambda_1 = \sum_{j=1}^{|Y|} n(x_i, y_j) \)

This means the next parameters \( \alpha_k^{(t+1)} \) is:

\[
\alpha_k^{(t+1)} = \frac{\sum_{j=1}^{|Y|} n(x_i, y_j) p(k|x_i, y_j, \Theta^{(t)})}{\sum_{j=1}^{|Y|} n(x_i, y_j)}
\]

The first-order partial derivative of Lagrange function regarding \( p_i \) is:

\[
\frac{\partial l_a(\Theta, \lambda | \Theta^{(t)})}{\partial p_i} = \sum_{j=1}^{|Y|} n(x_i, y_j) \frac{1}{p_i} - \lambda_2
\]

Setting this partial derivative to be zero, we obtain:

\[
\sum_{j=1}^{|Y|} n(x_i, y_j) - p_i \lambda_2 = 0
\]

Summing the equation above over \( \mathcal{X} \), we have:

\[
\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) - \lambda_2 \sum_{i=1}^{|X|} p_i = 0
\]

⇔ \( \lambda_2 = \sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} n(x_i, y_j) \)
This means the next parameters \( p^{(t+1)}_i \) is:

\[
p^{(t+1)}_i = \frac{\sum_{j=1}^{[y]} n(x_i, y_j)}{\sum_{i=1}^{[x]} \sum_{j=1}^{[y]} n(x_i, y_j)} \tag{2.19}
\]

The first-order partial derivative of Lagrange function regarding \( q_{jk} \) is:

\[
\frac{\partial l(\Theta, \lambda | \Theta^{(t)})}{\partial q_{jk}} = \sum_{i=1}^{[x]} n(x_i, y_j) \frac{1}{q_{jk}} p(k | x_i, y_j, \Theta^{(t)}) - \lambda_3
\]

Setting this partial derivative to be zero, we obtain:

\[
\sum_{i=1}^{[x]} \sum_{j=1}^{[y]} n(x_i, y_j) p(k | x_i, y_j, \Theta^{(t)}) - \lambda_3 \sum_{j=1}^{[y]} q_{jk} = 0
\]

Summing the equation above over \( y \), we have:

\[
\sum_{i=1}^{[x]} \sum_{j=1}^{[y]} n(x_i, y_j) p(k | x_i, y_j, \Theta^{(t)}) - \lambda_3 \sum_{j=1}^{[y]} q_{jk} = \sum_{i=1}^{[x]} \sum_{j=1}^{[y]} n(x_i, y_j) p(k | x_i, y_j, \Theta^{(t)})
\]

This means the next parameters \( q_{jk}^{(t+1)} \) is:

\[
q_{jk}^{(t+1)} = \frac{\sum_{i=1}^{[x]} n(x_i, y_j) p(k | x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{[x]} \sum_{j=1}^{[y]} n(x_i, y_j) p(k | x_i, y_j, \Theta^{(t)})} \tag{2.20}
\]

The two steps of GEM algorithm for AMM at some \( t^{th} \) iteration are shown in table 2.2.

| E-step: |
| --- |
| The conditional probability \( P(k | x_i, y_j, \Theta^{(t)}) \) is calculated based on current parameter \( \Theta^{(t)} = (\alpha^{(t)}_k, p^{(t)}_i, q^{(t)}_{jk})^T \), according to equation 2.17. |
| \( P(k | x_i, y_j, \Theta^{(t)}) = \frac{\alpha^{(t)}_{kl} p^{(t)}_i q^{(t)}_{jk}}{\sum_{l=1}^{K} \alpha^{(t)}_{kl} p^{(t)}_i q^{(t)}_{jl}} \) |

| M-step: |
| --- |
| The next parameter \( \Theta^{(t+1)} = (\alpha^{(t+1)}_k, p^{(t+1)}_i, q^{(t+1)}_{jk})^T \), which is a maximizer of \( Q(\Theta | \Theta^{(t)}) \) with subject to \( \Theta \), is calculated by equation 2.18, equation 2.19, and equation 2.20. |
| \( \alpha^{(t+1)}_{kl} = \frac{\sum_{j=1}^{[y]} n(x_i, y_j) p(k | x_i, y_j, \Theta^{(t)})}{\sum_{j=1}^{[y]} n(x_i, y_j)} \) |
| \( p^{(t+1)}_i = \frac{\sum_{j=1}^{[y]} n(x_i, y_j)}{\sum_{i=1}^{[x]} \sum_{j=1}^{[y]} n(x_i, y_j)} \) |
| \( q^{(t+1)}_{jk} = \frac{\sum_{i=1}^{[x]} n(x_i, y_j) p(k | x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{[x]} \sum_{j=1}^{[y]} n(x_i, y_j) p(k | x_i, y_j, \Theta^{(t)})} \) |

| Table 2.2. E-step and M-step of GEM algorithm for AMM |
| --- |
| GEM algorithm converges at some \( t^{th} \) iteration. At that time, \( \Theta^* = \Theta^{(t+1)} = \Theta^{(t)} \) is the AMM itself. When AMM is applied into soft clustering, dyadic data is clustered vertically (horizontally) and each \( \alpha_k \) is coverage ratio of cluster \( k \) (aspect \( k \)) according to \( x_i \). 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 \( \{1, 2, \ldots, K\} \) is Cartesian product of \( X\)-aspect set \( \{1, 2, \ldots, K_X\} \) and \( Y\)-aspect set \( \{1, 2, \ldots, K_Y\} \). In other words, the aspect space is still symmetric but is checked (stripped) according to two directions \( X \) and \( Y \).

\[
\{1,2, \ldots, K\} = \{1,2, \ldots, K_X\} \times \{1,2, \ldots, K_Y\}
\]

(2.21)

For every \( k \) belongs to \( \{1, 2, \ldots, K\} \), there always exists a respective pair: \( k_X \in \{1,2, \ldots, K_X\} \) and \( k_Y \in \{1,2, \ldots, K_Y\} \). However, for each \( k_X \) or each \( k_Y \), there are many respective \( k \).

\[
k \sim \{k_X, k_Y\} \quad \text{many} \quad k
\]

(2.22)

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

\[
P(x_i, y_j | \Theta) = \sum_{k=1}^{K} \alpha_k p_{i|k_X} q_{j|k_Y}
\]

(2.23)

As usual, \( \alpha_k \) is the probability of aspect \( c_k \) but \( p_{i|k_X} \) is the probability of \( x_i \) given \( k_X \) of \( k \) and \( q_{j|k_Y} \) is the probability of \( y_j \) given \( k_Y \) of \( k \).

\[
p_{i|k_X} = P(x_i | k_X)
\]

\[
q_{j|k_Y} = P(y_j | k_Y)
\]

The joint probability of \( x_i, y_j, \) and \( k \) is:

\[
P(x_i, y_j, k) = P(k)P(x_i, y_j | k) = \alpha_k P(x_i | k)P(y_j | k) = \alpha_k P(x_i | k_X)P(y_j | k_Y)
\]

The parameter of PMM is \( \Theta = (\alpha, p_{i|k_X}, q_{j|k_Y})^T \) in which there are \( K + K_X |X| + K_Y |Y| \) partial parameters \( \alpha_k, p_{i|k_X}, \) and \( q_{j|k_Y} \). Note,

\[
\sum_{k=1}^{K} \alpha_k = 1, \sum_{i=1}^{|X|} p_{i|k_X} = 1, \sum_{j=1}^{|Y|} q_{j|k_Y} = 1
\]

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

### E-step:

The conditional probabilities \( P(k | x_i, y_j, \Theta^{(t)}) \), \( P(k_X | x_i, y_j, \Theta^{(t)}) \), and \( P(k_Y | x_i, y_j, \Theta^{(t)}) \) are calculated based on current parameter \( \Theta^{(t)} = (\alpha^{(t)}, p_{i|k_X}^{(t)}, q_{j|k_Y}^{(t)})^T \), according to equation 2.24, equation 2.25, and equation 2.26.

\[
P(k | x_i, y_j, \Theta^{(t)}) = \frac{\alpha_k^{(t)} p_{i|k_X}^{(t)} q_{j|k_Y}^{(t)}}{\sum_{l=1}^{K} \alpha_k^{(t)} p_{i|l_X}^{(t)} q_{j|l_Y}^{(t)}}
\]

(2.24)

\[
P(k_X | x_i, y_j, \Theta^{(t)}) = \sum_{k:k_X=k} P(k | x_i, y_j, \Theta^{(t)})
\]

(2.25)

\[
P(k_Y | x_i, y_j, \Theta^{(t)}) = \sum_{k:k_Y=k} P(k | x_i, y_j, \Theta^{(t)})
\]

(2.26)

Please refer to equation 1.4 to comprehend equation 2.24.

### M-step:
The next parameter $\Theta^{(t+1)} = \left( \alpha_k^{(t+1)}, p_{i|k_X}^{(t+1)}, q_j^{(t+1)} \right)^T$, which is the maximizer of $Q(\Theta | \Theta^{(t)})$ with subject to $\Theta$, is calculated by equation 2.27, equation 2.28, and equation 2.29.

\[
\alpha_k^{(t+1)} = \frac{\sum_{i=1}^{[X]} \sum_{j=1}^{[Y]} n(x_i, y_j) P(k|x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{[X]} \sum_{j=1}^{[Y]} n(x_i, y_j)} \\
p_{i|k_X}^{(t+1)} = \frac{\sum_{j=1}^{[Y]} n(x_i, y_j) P(k_X|x_i, y_j, \Theta^{(t)})}{\sum_{i=1}^{[X]} \sum_{j=1}^{[Y]} n(x_i, y_j)} \\
q_j^{(t+1)} = \frac{\sum_{i=1}^{[X]} n(x_i, y_j) P(k_Y|x_i, y_j, \Theta^{(t)})}{\sum_{j=1}^{[Y]} \sum_{i=1}^{[X]} n(x_i, y_j)} \tag{2.29}
\]

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

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

### 3. Predicting unaccomplished co-occurrence 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 $S$ is valued dyadic data in which every co-occurrence $(x_i, y_j)$ is associated with value $z$ from random variable $Z$ then, SMM is reformed as follows:

\[
f(x_i, y_j, Z|\Theta) = \sum_{k=1}^{K} \alpha_k p_{i|k_X} q_{j|k_Y} f_k(Z|\varphi_k) \tag{3.1}
\]

AMM is reformed as follows:

\[
f(x_i, y_j, Z|\Theta) = p_i \sum_{k=1}^{K} \alpha_k q_{j|k_Y} f_k(Z|\varphi_k) \tag{3.2}
\]

PMM is reformed as follows:

\[
f(x_i, y_j, Z|\Theta) = \sum_{k=1}^{K} p_i \alpha_k q_{j|k_Y} f_k(Z|\varphi_k) \tag{3.3}
\]

Where $f_i(Z|\varphi_k)$ is the $k^{th}$ PDF of $Z$ corresponding to the aspect $k$, in which $\varphi_k$ is parameter of $f_i(Z|\varphi_k)$. Of course, the parameter $\Theta$ now must include all $\varphi_k$. It is possible to consider that

\[
f_k(Z|\varphi_k) = f(Z|k, \varphi_k)
\]

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

\[
f(Z|x_i, k, \varphi_k) = f(Z|k, \varphi_k) = f_k(Z|\varphi_k)
\]

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

\[
P(x_i, y_j | k, Z) = P(x_i, y_j | k) \\
P(y_j | x_i, Z, k) = P(y_j | x_i, k)
\]

For example, within SMM, the joint PDF of $x_i, y_j, Z$, and $k$ is:

\[
f(x_i, y_j, Z, k) = P(k) P(x_i, y_j, Z|k) = \alpha_k P(x_i, y_j | k, Z) f(Z|k, \varphi_k) = \alpha_k P(x_i, y_j | k) f_k(Z|\varphi_k)
\]

Within AMM, the joint PDF of $x_i, y_j, Z$, and $k$ is:
Following is the proof of equation 3.8.

\[ f(z|x_i, y_j, \theta) = \frac{\sum_{k=1}^{K} \alpha_k p_{i|k} q_{j|k} f_k(Z|\varphi_k)}{\sum_{k=1}^{K} \alpha_k p_{i|k} q_{j|k} f_k(Z|\varphi_k)} = \frac{\sum_{k=1}^{K} \alpha_k p_{i|k} q_{j|k} f_k(Z|\varphi_k)}{\sum_{k=1}^{K} \alpha_k p_{i|k} q_{j|k} f_k(Z|\varphi_k)} \]

Similarly, the conditional PDF \( f(Z|x_i, y_j, \Theta) \) of AMM is:
Currently, an unaccomplished value is estimated based on pre-knowledge of an existent pair of unaccomplished values. It is essential to make a weighted sum of centroids over all clusters. Learning and it is easy to apply these models into soft clustering. Predicting or estimating

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 (\(X\)-object and \(Y\)-object). As a result, an estimate \(\hat{Z}\) 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.

References
Hofmann, T. (2004, January). Latent Semantic Models for Collaborative Filtering. *ACM Transactions on Information Systems (TOIS)*, 22(1), 89-115. doi:10.1145/963770.963774
Hofmann, T., & Puzicha, J. (1998). *Statistical Models for Co-occurrence Data*. Massachusetts Institute of Technology, Artificial Intelligence Laboratory. MIT Publisher. Retrieved from https://dspace.mit.edu/bitstream/handle/1721.1/7253/AIM-1625.pdf?sequence=2
Hofmann, T., & Puzieha, J. (1999). Latent Class Models for Collaborative Filtering. In T. Dean (Ed.), *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI ’99)* (pp. 688-693). San Francisco, CA, USA: Morgan Kaufmann. Retrieved from https://dl.acm.org/citation.cfm?id=687583
Hofmann, T., Puzicha, J., & Jordan, M. I. (1998). Learning from Dyadic Data. In M. J. Kearns, S. A. Solla, & D. A. Cohn (Ed.), *Advances in Neural Information Processing Systems 11 (NIPS 1998)*. 11, pp. 466-472. Denver: MIT Press. Retrieved from https://papers.nips.cc/paper/1503-learning-from-dyadic-data
Nguyen, L. (2020). *Tutorial on EM algorithm*. MDPI. Preprints. doi:10.20944/preprints201802.0131.v8