# HDP for Alignment of Multiple LC-MS Data in Metabolomics

May 17, 2014

## 1 Introduction

Peak alignment is an important step in the pre-processing stages of LC-MS data. Errors produced during alignment will have a potentially significant impact in the later analysis stage (e.g. when performing differential analysis on the abundance of metabolites of interests). Existing methods do not take into account the dependencies of related peaks during alignments. We do that here by proposing a hierarchical Bayesian model that assigns related peaks to metabolite within and across files.

## 2 A Generative Model for LC-MS Data

Metabolites generates multiple clusters across files. In turn, clusters generates multiple peaks within files. We can model this process with a hierarchical mixture model of Gaussian components. Within each file, related peaks are grouped together. These groups are shared across files through top-level latent variables that represents the metabolites. We use Dirichlet Process (DP) prior to avoid setting the number of clusters a priori. The overall model fits into the Hierarchical Dirichlet Process (HDP) framework [1].

The observed data is the peak RT values $\mathbf{x} = (x_n^j)$ for all peaks across files. Within each file $j$, the peak RT value $x_n^j$ is drawn from its parent cluster's RT $t_k^j$, which is normally distributed.

$$x_n^j | t_k^j, \gamma \sim \mathcal{N}(t_k^j, \gamma^{-1}) \tag{1}$$

The cluster RT $t_{ij}$ in file $j$ is drawn from the metabolite RT $t_i$ that is shared across all files.

$$t_k^j | t_i, \delta \sim \mathcal{N}(t_i, \delta^{-1}) \tag{2}$$

The metabolite RT $t_i$ is drawn from a base distribution of predicted retention times of metabolites. This is a Gaussian for now.

$$t_i|\mu_0, \sigma_0 \sim \mathcal{N}(\mu_0, \sigma_0^{-1}) \tag{3}$$

*Add description about HDP, Chinese Restaurant Franchise and the stick breaking stuff*

## 2.1 Modelling Assumptions

Some current assumptions to think about in the model:

1. Every peak must be generated by a metabolite. How about signals that are just noise (not coming from any metabolite) ?
2. Top-level component (metabolites) can produce multiple clusters in the same file. This seems to be a reasonable assumption ?
3. Not considering the mass information yet (including adducts, isotope etc) ..
4. We model RT drifts across replicates using the clusters (which are basically the noisy realisation of metabolite in each file). Assume clusters are IID given the parent metabolite, i.e. no correlation in the drift over time.

## 2.2 Inference

Denote the assignment of peak $n$ to cluster $k$ in file $j$ by $z_{jnk} = 1$. The matrix $\boldsymbol{Z}$ contains this assignment for all indices $j$,$k$ and $n$ in the model. We also need another indicator value to set the assignment of clusters within each file to the top-level component (metabolite) shared across files. Denote by $v_{jki} = 1$ if cluster $k$ in file $j$ is assigned to metabolite $i$. Collectively, we store the assignments in the matrix $\boldsymbol{V}$ for all $i$, $j$ and $k$.

Given the observed data $\mathbf{x}$, our task is the infer the parameters $\boldsymbol{\theta} = (\boldsymbol{t}, \boldsymbol{t}^j, \boldsymbol{Z}, \boldsymbol{V})$. We use Gibbs sampling to sample from the posterior distribution $P(\boldsymbol{\theta}|\boldsymbol{x}) \propto P(\boldsymbol{x}|\boldsymbol{\theta})P(\boldsymbol{\theta})$. The likelihood term $P(\boldsymbol{x}|\boldsymbol{\theta})$ is

$$P(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_i \sum_j \sum_n \Big[ p(x_n^j|t_k^j) \cdot p(t_k^j|t_i) \cdot p(z_{jnk} = 1) \cdot p(v_{jki} = 1) \Big] \tag{4}$$

We set a DP prior with concentration parameter $\alpha$ on the assignment of peaks to clusters $p(z_{jnk} = 1)$, and another DP prior with concentration parameter $\alpha'$ on the assignment of clusters to metabolites $p(v_{jki} = 1)$. We use MCMC method to sample from the posterior distribution $P(\boldsymbol{\theta}|\boldsymbol{x})$. Specifically, we follow the posterior sampling in the Chinese restaurant franchise, as described in section 5.1 in [1]. Here, we explicitly assign the peaks to their parent clusters, which are then assigned to the parent metabolites.

The next sub-sections describe the update steps during Gibbs sampling of the posterior distribution

### 2.2.1 Initialisation

1. For every file, assign all peaks under 1 cluster.
2. Across files, all clusters are assigned under 1 global metabolite.
3. The metabolite's RT is sampled from the base distribution (eq. 3), and the cluster's RT is sampled conditioned on it's parent metabolite RT (eq. 2).

### 2.2.2 Useful identities

The integral of the product of two pdf $p(x|y,c) \cdot p(y|a,b)$ where both are normal densities

$$\int \left[\mathcal{N}(x|y,c^{-1}) \cdot \mathcal{N}(y|a,b^{-1})\right] \, dy \quad \propto \quad \mathcal{N}(x|a,(c^{-1}+b^{-1})) \tag{5}$$

*Proof to follow ...*

### 2.2.3 Updating peak assignments

To update the assignment of peak $n$ to cluster $k$ in file $j$, we need the conditional probability of $p(z_{jnk} = 1)$ given other parameters. This is

$$p(z_{jnk} = 1|\ldots) \propto \begin{cases} c_{jk} \cdot L(x_n^j|z_{jnk} = 1) \\ \alpha \cdot L(x_n^j|z_{jnk^*} = 1) \end{cases} \tag{6}$$

For existing cluster $k$, the likelihood of the peak is proportional to $c_{jk}$, the number of peaks inside that cluster.

$$L(x_n^j|z_{jnk} = 1) = \mathcal{N}(x_n^j|t_k^j, \gamma^{-1}) \tag{7}$$

For a new cluster $k^*$, this is proportional to $\alpha$. The likelihood then depends on whether the peak is assigned to an existing metabolite $i$ or a new metabolite $i^*$.

$$L(x_n^j|z_{jnk^*} = 1) = \sum_i \left[\frac{c_i}{\alpha' + \sum_i c_i} p(x_i^j|v_{jk^*i} = 1)\right] + \frac{\alpha'}{\alpha' + \sum_i c_i} p(x_i^j|v_{jk^*i^*} = 1) \tag{8}$$

For existing metabolite $i$, $p(x_i^j|v_{jk^*i} = 1)$, the likelihood of the peak under that metabolite, can be computed by marginalising over all possible values of clusters RT $t_k^j$. Note that we use the identity in equation 5 to go from equation 10 to equation 11.

$$
\begin{aligned}
p(x_n^j|v_{jk^*i}=1) &\propto \int \left[p(x_n^j|t_k^j)p(t_k^j|t_i)\right] dt_k^j &(9)\\
&\propto \int \left[\mathcal{N}(x_n^j|t_k^j,\gamma^{-1})\cdot\mathcal{N}(t_k^j|t_i,\delta^{-1})\right] dt_k^j &(10)\\
&\propto \mathcal{N}(x_n^j|t_i,(\gamma^{-1}+\delta^{-1})) &(11)
\end{aligned}
$$

For new metabolite $i^*$, the likelihood is obtained by marginalising over all possible values of clusters RT $t_k^j$ and metabolite RT $t_i$

$$
\begin{aligned}
p(x_n^j|v_{jk^*i^*}=1) &\propto \int\int \left[p(x_n^j|t_k^j)p(t_k^j|t_i)\right] dt_k^j\, dt_i &(12)\\
&\propto \int\int \left[\mathcal{N}(x_n^j|t_k^j,\gamma^{-1})\cdot\mathcal{N}(t_k^j|t_i,\delta^{-1})\cdot\mathcal{N}(t_i|\mu_0,\sigma_0^{-1})\right] dt_k^j\, dt_i &(13)\\
&\propto \mathcal{N}(x_n^j|\mu_0,(\sigma_0^{-1}+\gamma^{-1}+\delta^{-1})) &(14)
\end{aligned}
$$

The above assignment steps can be illustrated using the Chinese Restaurant Franchise metaphor. Here, a peak is a customer, a file is a restaurant, a cluster is a table, and a metabolite is a dish. During the Gibbs sampling step, a new customer $n+1$ arrives in the restaurant. We then need to determine whether this new customer is to be seated on a new or existing table. For an existing table, the likelihood proportional to the number of people already sitting on that table ($c_{jk}$) times the likelihood of the new customer belonging on the table (eq. 7). For a new table, the likelihood is proportional to the DP concentration parameter $\alpha$ times the likelihood of ordering any of the existing or new dishes (eq. 8). When deciding which dish to order, we marginalise over all possible tables in eq. 9 and eq. 12 (so we don't have to care about which table is actually selected).

Having computed both likelihood values of the customer sitting on existing tables or a new table (eq. 6), we do a coin toss to decide which table $k$ is actually selected. At this point, we should have maintained a restaurant-level variable $K$, which is the counter of all tables that exist in the restaurant. If $k \leq K$, we send the customer to sit on table $k$ and share the dish already ordered by his friends sitting on table $k$. Otherwise if $k > K$, we increment $K$ by 1, create a new table $k+1$ and offers the customer the menu of dishes to order. At this point, the new table $k+1$ is created but not assigned any retention time value (mixture parameter) yet. We will return to this later.

Since the customer is the first to sit on the table, he would order the dish to be shared on that table. We reuse the computation done for eq. 8 to compute the likelihood of the new customer ordering existing dishes or new dish, and do a coin toss to select the dish $i$. If $i \leq I$, where $I$ is the global counter of all dishes that exist *across restaurants*, we serve the customer the dish $i$. Otherwise, we get our chef to devise a new dish $i+1$ for him. The new dish $i+1$ can subsequently be ordered by any new customer that arrives in any of the restaurant in our franchise. The mixture parameter (retention time value) $t_i$ of the new dish $i+1$ can be computed given the data $x_n^j$ using Bayes' rule:

$$\begin{aligned} p(t_i|x_n^j) &\propto p(x_n^j|t_i)p(t_i) && (15)\\ &\propto \left[\int p(x_n^j|t_k^j)p(t_k^j|t_i)\,dt_k^j\right]\cdot p(t_i) && (16)\\ &\propto \left[\int \mathcal{N}(x_n^j|t_k^j,\gamma^{-1})\cdot\mathcal{N}(t_k^j|t_i,\delta^{-1})\,dt_k^j\right]\cdot\mathcal{N}(t_i|\mu_0,\sigma_0^{-1}) && (17)\\ &\propto \mathcal{N}(x_n^j|t_i,\gamma^{-1}+\delta^{-1})\,\cdot\,\mathcal{N}(t_i|\mu_0,\sigma_0^{-1}) && (18)\end{aligned}$$

In the LHS of the equation above, $p(t_i|x_n^j)$ is a Normal distribution. Let's say it is parameterised by $\mathcal{N}(t_i|\mu_i,\gamma_i^{-1})$.

$$\begin{aligned} \mathcal{N}(t_i|\mu_i,\gamma_i^{-1}) &\propto \mathcal{N}(x_n^j|t_i,\gamma^{-1}+\delta^{-1})\,\cdot\,\mathcal{N}(t_i|\mu_0,\sigma_0^{-1}) && (19)\\ exp\left(\frac{-\gamma_i}{2}(t_i-\mu_i)^2\right) &\propto exp\left(\frac{-(\gamma^{-1}+\delta^{-1})^{-1}}{2}(x_n^j-t_i)^2+\frac{-\sigma_0}{2}(t_i-\mu_0)^2\right) && (20)\end{aligned}$$

Collecting the quadratic term containing $(t_i)^2$, we can solve for $\gamma_i$

$$\begin{aligned} \frac{-\gamma_i}{2}(t_i)^2 &= \frac{(\gamma^{-1}+\delta^{-1})^{-1}}{2}(t_i)^2+\frac{-\sigma_0}{2}(t_i)^2 && (21)\\ \gamma_i &= (\gamma^{-1}+\delta^{-1})^{-1}+\sigma_0 && (22)\end{aligned}$$

Similarly, collecting the linear term containing $t_i$, we can solve for $\mu_i$

$$\begin{aligned} \frac{-\gamma_i}{2}(2t_i\mu_i) &= \frac{-(\gamma^{-1}+\delta^{-1})^{-1}}{2}(2x_n^j t_i)+\frac{-\sigma_0}{2}(2t_i\mu_0) && (23)\\ -\gamma_i\mu_i &= -(\gamma^{-1}+\delta^{-1})^{-1}x_n^j-\sigma_0\mu_0 && (24)\\ \mu_i &= \frac{1}{\gamma_i}\left[(\gamma^{-1}+\delta^{-1})^{-1}x_n^j+\sigma_0\mu_0\right] && (25)\end{aligned}$$

So, when creating a new dish $t_i$, we sample its value from a Normal distribution $\mathcal{N}(t_i|\mu_i,\gamma_i^{-1})$ with mean $\mu_i$ and precision $\gamma_i$ given above. Similarly, we can sample from the posterior distribution of table value $t_k^j$ given its parent $t_i$ and the data $x_n^j$:

$$p(t_k^j|x_n^j) \propto p(x_n^j|t_k^j)p(t_k^j|t_i)=\mathcal{N}(t_k^j|\mu_k,\gamma_k^{-1}) \qquad (26)$$

where

$$\gamma_k = \gamma + \delta \tag{27}$$

$$\mu_k = \frac{1}{\gamma_k}\left[\gamma x_n^j + \delta t_i\right] \tag{28}$$

The specific steps to update $p(z_{jnk} = 1)$ is given in algorithm 1.

**Algorithm 1** Updating peak assignments (assign_peaks.m)

```
I = 0, K = 0
j = loop randomly over files
    n = loop randomly over peaks

        remove peak n from model
        if this results in empty cluster k
            delete cluster k from model
            if this results in empty metabolite i
                delete metabolite i from model
            end
        end

        P1 = compute cjk * L(peak n|existing cluster, eq 7, 8)
        P2 = compute alpha * L(peak n|new cluster, eq 7, 9)
        new_k = pick_cluster(P1, P2)
        if new_k > K

            K = K+1
            P3 = compute ci * L(peak n|existing metabolite, eq 10)
            P4 = compute alpha' * L(peak n|new metabolite, eq 11)
            new_i = pick_metabolite(P3, P4)

            if new_i > I
                I = I+1
                ti = sample new metabolite RT given peak n (eq. 16)
                set metabolite RT = ti
            end
            tij = sample new cluster RT given ti and peak n (eq. 27)
            set cluster RT = tij

        end
        assign Zjnk = 1 for peak n to cluster k in file j

    end
end
```

### 2.2.4 Updating cluster assignments

Given some existing cluster $k$, we would like to assign this cluster (a whole block of peaks $\{x_n^j\}$ under $k$) into a new metabolite $i$.

The probability is proportional to $c_i$ for existing metabolite and $\alpha'$ for new metabolite. The likelihood term for existing table is

$$p(t_k^j|t_i) = \mathcal{N}(t_i, \delta^{-1}) \tag{29}$$

For new table, it is

$$\begin{aligned} p(t_k^j|t_{i^*}) &\propto \int \left[ p(t_k^j|t_i) \cdot p(t_i|\mu_0, \sigma_0) \right] dt_i && (30) \\ &\propto \int \left[ \mathcal{N}(t_k^j|t_i, \delta^{-1}) \cdot \mathcal{N}(t_i|\mu_0, \sigma_0^{-1}) \right] dt_i && (31) \\ &\propto \mathcal{N}(t_i|\mu_0, (\delta^{-1} + \sigma_0^{-1})) && (32) \end{aligned}$$

If we marginalise out $t_k^j$, the metabolite RT $t_i$ is known but the new cluster RT $t_k^j$ is unknown, so we can no longer assume conditional independence between the peaks. To compute the probability of this block of peaks under the new metabolite $i$, we have to apply the chain rule:

$$p(x_1^j, x_2^j, \ldots, x_n^j|t_i) = p(x_1^j|t_i) \cdot p(x_2^j|t_i, x_1^j) \cdot p(x_3^j|t_i, x_1^j, x_2^j) \ldots p(x_n^j|t_i, x_1^j, x_2^j, \ldots, x_{n-1}^j) \tag{33}$$

where

$$p(x_n^j|t_i, x_1^j, x_2^j, \ldots, x_{n-1}^j) = \int \left[ p(x_n^j|t_k^j) \cdot p(t_k^j|t_i, x_1^j, x_2^j, \ldots, x_{n-1}^j) \right] dt_k^j \tag{34}$$

*More to come ..*

### 2.2.5 Updating metabolite RTs

The conditional probability of metabolite RTs $p(t_i|\ldots)$ is given by the product of the prior probability on $t_i$ multiplied by the likelihood of the RT values of metabolite $i$'s child clusters. $t_{v_{jki}=1}^j$ denotes the retention time of cluster $k$ in file $j$ that is assigned to metabolite $i$. There is a total of $J$ files altogether. So, for every metabolite $i$, we update its RT:

$$p(t_i|\ldots) \propto p(t_i|\mu_0, \sigma_0^{-1}) \prod_j^J \prod_k^K p(t_{v_{jki}=1}^j|t_i, \delta^{-1}) = \mathcal{N}(\mu_a, \gamma_a^{-1}) \tag{35}$$

where

$$\mu_a = \frac{1}{\gamma_a}\left[\mu_0\sigma_0 + \delta\sum_j\sum_k t^j_{v_{jki}=1}\right] \tag{36}$$

$$\gamma_a = \sigma_0 + c_{v_{jki}=1}\delta \tag{37}$$

$$c_{v_{jki}=1} = \sum_j\sum_k I(v_{jki} = 1) \tag{38}$$

### 2.2.6 Updating cluster RTs

Similar to above, for every $j$ file, there is a total of $N$ peaks inside. $x^j_{z_{jnk}=1}$ denotes the retention time of peak $n$ that is assigned to cluster $k$ in file $j$. So, for every cluster $k$ in file $j$, we update its RT:

$$p(t^j_k|\ldots) \propto p(t^j_k|t_i,\delta^{-1})\prod_n^N p(x^j_{z_{jnk}=1}|t^j_k,\gamma^{-1}) = \mathcal{N}(\mu_b,\gamma_b^{-1}) \tag{39}$$

where

$$\mu_b = \frac{1}{\gamma_b}\left[t_i\delta + \gamma\sum_n x^j_{z_{jnk}=1}\right] \tag{40}$$

$$\gamma_b = \delta + c_{z_{jnk}=1}\gamma \tag{41}$$

$$c_{z_{jnk}=1} = \sum_n I(z_{jnk} = 1) \tag{42}$$

### 2.2.7 Results

Some plots ..

## References

[1] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical Dirichlet Processes. pages 1–30, 2005.

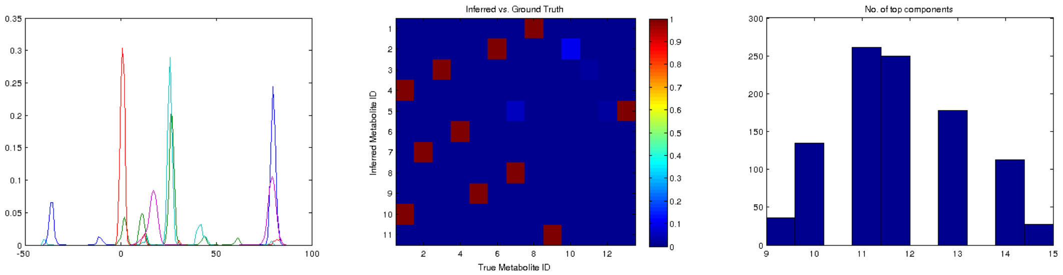

Figure 1: Sample result