

Topic model with Batch effects

Kushal K Dey

January 22, 2016

Introduction

In RNA-seq experiments, we often encounter samples coming from different batches. The batches may be determined by the amplification procedures used, or the sequencing machine or even the sequencing lane effects. When these batch effects or technical effects are present in the samples, it becomes difficult to often separate out the biological information from the technical information (the latter is often relatively stronger). The topic model or the grade-of membership model has been used to cluster the samples based on their RNA-seq reads counts data (see [paper](#)). In the paper, we have shown that the topic model is sensitive to the presence of batch effects, however we have not been able to present a solution to that problem. We address the issue of how one can tackle batch effects in a topic model type framework.

We first present the standard topic model framework

Standard Topic Model

Let c_{ng} be the counts of reads for sample n and gene g . Let c_{n+} be the sum of reads for sample n , also called the *library size*.

$$(c_{n1}, c_{n2}, \dots, c_{nG}) \sim \text{Mult}(c_{n+}, p_{n1}, p_{n2}, \dots, p_{nG})$$

$$p_{ng} = \sum_{k=1}^K \omega_{nk} \theta_{kg} \quad \sum_k \omega_{nk} = 1 \quad \forall n \quad \sum_g \theta_{kg} = 1 \quad \forall k$$

Here $\omega_{n\cdot}$ represents the topic proportions for n th samples. On the other hand $\theta_{k\cdot}$ represents the probability distribution on the genes for the k th topic or cluster.

Topic model with Batch effects

One way batch effects may be incorporated in the above model would be to make the topic distribution for each cluster/ topic a function of the batch the sample is coming from, as well. Then we can write the above model as

$$(c_{n1}, c_{n2}, \dots, c_{nG}) \sim \text{Mult}(c_{n+}, p_{n1}, p_{n2}, \dots, p_{nG}) \quad (1)$$

$$p_{ng} = \sum_{k=1}^K \omega_{nk} \theta_{b(n):k,g} \quad \sum_k \omega_{nk} = 1 \quad \forall n \quad \sum_g \theta_{b(n):k,g} = 1 \quad \forall k, \quad b(n) \in \{1, 2, \dots, B\} \quad (2)$$

Prior Specification

Note that the above the model is analogous to applying topic model separately for each batch. The problem with that approach is that we will not be able to track which cluster in Batch 1 corresponds to that cluster in Batch 2. Also, we expect each cluster distribution to have some common features across different batches despite getting effected by batch effects. In order to tackle this, we make the following assumption.

For each cluster k

$$(\theta_{b:k,1}, \theta_{b:k,2}, \dots, \theta_{b:k,G}) \sim \text{Dir}_G(\theta_{k1}, \theta_{k2}, \dots, \theta_{kG}) \quad b \in \{1, 2, \dots, B\} \quad (3)$$

Which is same as saying that for each batch, we are generating a sample from the cluster with mean $(\theta_{k1}, \theta_{k2}, \dots, \theta_{kG})$, which represents the cluster k . This is analogous to the assumption in the normal linear models with batch effects,

$$y_{ng} = \mu_{t(n):b(n),g} = \mu + \tau_{t(n)} + \beta_{b(n)} + e_{ng}$$

where $t(n)$ is the treatment effect and $b(n)$ is the batch effect. We often assume that

$$\beta_b \sim N(0, \sigma_b^2)$$

Then

$$\mu_{t(n):b(n)} \sim N(\mu + \tau_n, \sigma_b^2) := N(\mu_{t(n)}, \sigma_b^2)$$

You can see that the treatment effects under the different batches are then a random sample from a distribution whose mean is the treatment effect without the batch information. Note that σ_b term is there in normal models to tune the variance for each effect. We can also put such a scaling parameter in our model Equation 11.

Then for each k ,

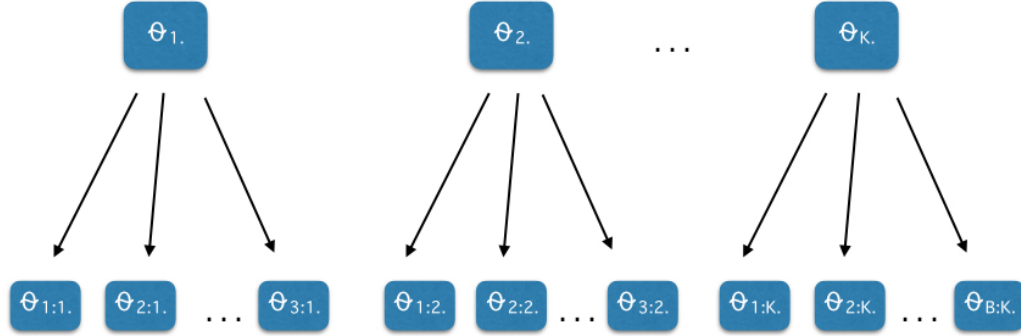
$$(\theta_{b:k,1}, \theta_{b:k,2}, \dots, \theta_{b:k,G}) \sim \text{Dir}_G(\alpha_b \theta_{k1}, \alpha_b \theta_{k2}, \dots, \alpha_b \theta_{kG}) \quad b \in \{1, 2, \dots, B\}$$

However, as of now, I am assuming that $\alpha_b = 1$ for all batches and working with the simpler model.

We assume a prior for θ_{kg} .

$$(\theta_{k1}, \theta_{k2}, \dots, \theta_{kG}) \sim \text{Dir}_G\left(\frac{1}{KG}, \frac{1}{KG}, \dots, \frac{1}{KG}\right) \quad \forall k \quad (4)$$

So, essentially we have a hierarchical structure in the θ 's, on combining Equation 11 and Equation 4.



We can assume the same prior for ω as in standard topic model, given by

$$(\omega_{n1}, \omega_{n2}, \dots, \omega_{nK}) \sim Dir_K \left(\frac{1}{K}, \frac{1}{K}, \dots, \frac{1}{K} \right) \quad \forall n$$

Model estimation

We can assume that

$$c_{n+} \sim Poi(\lambda_n) \tag{5}$$

Then combining Equation 1 and Equation 5, we get

$$c_{ng} \sim Poi \left(\lambda_n \sum_k \omega_{nk} \theta_{kg} \right) \tag{6}$$

Let z_{nkg} represents the number of counts from sample n and from feature g that comes from k th subgroup or cluster. By definition,

$$\sum_{k=1}^K z_{nkg} = c_{ng}$$

Since the summation of two independent Poisson random variables is also a Poisson variable with mean equal to the sum of the means of the original random variables, we can infer that

$$z_{nkg} \sim Poi(\lambda_n \omega_{nk} \theta_{b(n):k,g})$$

Let z_{bkg} be the latent variable representing the number of reads coming from the b th batch, k th subgroup/cluster and gene g .

$$z_{bkg} | \theta_{b:k,g} \sim Poi\left(\theta_{b:k,g} \sum_{b(n):b} \omega_{nk} \lambda_n\right)$$

$$z_{kg} | \theta_{b:k,g} \sim Poi\left(\sum_b \theta_{b:k,g} \sum_{b(n):b} \omega_{nk} \lambda_n\right)$$

We can write

$$\begin{aligned} E(z_{kg} | \theta_{kg}) &= E\left(\sum_b E(z_{bkg} | \theta_{b:k,g}) | \theta_{kg}\right) \\ &= E\left(\sum_b \theta_{b:k,g} \sum_{b(n):b} \omega_{nk} \lambda_n | \theta_{kg}\right) \\ &= \theta_{kg} \sum_n \omega_{nk} \lambda_n \end{aligned}$$

Note that under the above set of assumptions, the distribution

$$z_{bkg} | \theta_{kg} \sim \prod_{b=1}^B \frac{\Gamma(\sum_g \theta_{kg})}{\Gamma(\sum_g z_{bkg} + \sum_g \theta_{kg})} \prod_{g=1}^G \frac{\Gamma(z_{bkg} + \theta_{kg})}{\Gamma(\theta_{kg})} \theta_{kg}^{1/KG} \quad (7)$$

and

$$z_{kg} = \sum_b z_{bkg} \quad (8)$$

This leads to a completed distribution for z_{kg} . this density is difficult to handle and will be time expensive to solve for θ using this density function. Therefore, we resort to a more simplified approach, through EM algorithm to obtain parameter updates at each step.

EM algorithm

Suppose at the end of update n , the current estimates we have are $\omega_{nk}^{(m)}$ and $\theta_{kg}^{(m)}$. We use these iterates to obtain refined estimates $\omega_{nk}^{(m+1)}$ and $\theta_{kg}^{(m+1)}$. This idea is similar to the standard topic model iterative scheme. The only major difference is that there are intermediate parameters containing batch effect information, namely $\theta_{b(n):k,g}^{(m+1)}$.

$$\mathcal{L}(z_{bkg}|\theta_{b:k,g}) := Poi\left(\theta_{b:k,g} \sum_{b(n):b} \omega_{nk} \lambda_n\right) \quad (9)$$

or

$$\mathcal{L}(z_{bk1}, z_{bk2}, \dots, z_{bkG}|\theta_{b:k,.}) := Mult(z_{bk+}, \theta_{b:k,1}, \theta_{b:k,2}, \dots, \theta_{b:k,G}) \quad (10)$$

$$\pi(\theta_{b:k,.}|\theta_{k.}) \propto \prod_{g=1}^G \theta_{b:k,g}^{\theta_{kg}} \quad (11)$$

We define the E-step of the EM algorithm as follows (done separately for each k)

$$\mathcal{Q}(\theta_{b:k,.}|C_{N \times G}, \theta_{b:k,.}^{(m)}, \theta_{k.}^{(m)}, \omega^{(m)}) = \mathbb{E}_{Z|C_{N \times G}, \theta_{b:k,.}^{(m)}, \theta_{k.}^{(m)}, \omega^{(m)}} \left(\log \mathcal{L}(z_{bk1}, z_{bk2}, \dots, z_{bkG}|\theta_{b:k,.}) + \log \pi(\theta_{b:k,.}|\theta_{k.}^{(m)}) \right) \quad (12)$$

We next perform the M-step and we obtain the following solutions for $\theta_{b:k,g}$ s.

$$\theta_{b:k,g}^{(m+1)} := \frac{\mathbb{E}\left(z_{b:k,g}|C_{N \times G}, \theta_{k.}^{(m)}, \omega^{(m)}\right) + \theta_{kg}^{(m)}}{\sum_g \mathbb{E}\left(z_{b:k,g}|C_{N \times G}, \theta_{k.}^{(m)}, \omega^{(m)}\right) + 1} \quad (13)$$

The expectation over $[z_{b:k,g}|C_{N \times G}, \theta_{b:k,.}^{(m)}, \theta_{k.}^{(m)}, \omega^{(m)}]$ is given by the following

$$\mathbb{E}\left(z_{b:k,g}|C_{N \times G}, \theta_{b:k,.}^{(m)}, \theta_{k.}^{(m)}, \omega^{(m)}\right) := c_{ng} \frac{\omega_{nk}^{(m)} \theta_{b:k,g}^{(m)}}{\sum_{h=1}^K \omega_{nh}^{(m)} \theta_{b:h,g}^{(m)}}$$

Once we obtain the estimates $\theta_{b:k,g}^{(m+1)}$, then we would like to update $\theta_{kg}^{(m+1)}$ conditional on $\theta_{b:k,g}^{(m+1)}$ for all $b \in \{1, 2, \dots, B\}$.

One can assume $\theta_{b:k,.}^{(m+1)}$ for all b to be a random sample of size B (same as number of batches) given $\theta_{k.}$ and given the data, we want to estimate the parameters $\theta_{k.}$. However estimating MLE of the Dirichlet parameters is not easy, and requires Newton-Raphson Method (check [paper](#)). One can obtain a MOM (Methods of Moments) type estimator with a tuning parameter ν as follows.

$$\theta_{kg}^{(m+1)} = \frac{1}{B + \nu} \sum_{b=1}^B \theta_{b:k,g}^{(m+1)} + \frac{\nu}{B + \nu} \frac{1}{KG}$$

instead of the MLE estimator derived from the conditional distribution of z_{kg} given θ_{kg} as given in Equation 7 and Equation 8. However as discussed earlier, the original conditional distribution of $z_{kg}|\theta_{kg}$ seems difficult to handle.

We fix the batch b . Then given $\theta_{b:k,g}^{(m+1)}$, we can estimate $\omega_{nk}^{(m+1)}$ from $\omega_{nk}^{(m)}$ and $\theta_{b:k,g}^{(m+1)}$ using similar convex optimization technique used by Matt Taddy in his [paper](#).

At the end of these steps, we will have $\omega_{nk}^{(m+1)}$, $\theta_{b(n):k,g}^{(m+1)}$ and $\theta_{kg}^{(m+1)}$. We can use these to update the parameters further and we continue till the log-likelihood converges.