mash No Baseline

Sarah Urbut

July 18, 2016

Contents

1 Purpose

The purpose of this document is to propose a method for extending `mash` to estimate 'true' effects across conditions in a setting in which no obvious baseline exists. We assume that we observe noisy, uncentered averages $\hat{\boldsymbol{C}}_{jr}$ in each of R conditions, and seek to estimate the underlying true 'deviations' from average measurement across conditions and can be seen as the effects in `mash`.

Here, the use of bold-face notation indicates a vector, while matrix quantities are typeset in capital but unboldface letters.

2 Defining the Old Model

For a given gene-snp pair, $\boldsymbol{b}$ represents the R vector of unknown standardized effect. We model the prior distribution from which $\boldsymbol{b}$ is drawn as a mixture of multivariate *Normals.*

$$\boldsymbol{b}|\boldsymbol{\pi}, \mathbf{U} \sim \sum_{\mathbf{k},\mathbf{l}} \pi_{\mathbf{k},\mathbf{l}}\ N_{\mathbf{R}}(\mathbf{0}, \omega_{\mathbf{l}} \mathbf{U}_{\mathbf{k}}) \tag{1}$$

Furthermore, for a given gene-snp pair, the Likelihood on $\boldsymbol{b}$:

$$\hat{\boldsymbol{b}}|\boldsymbol{b} \sim N_R(\boldsymbol{b}, \hat{V}) \tag{2}$$

Now, we observe for each gene j a vector of uncentered noisy average feature expression $\hat{\boldsymbol{C}}$ across R conditions:

$$\hat{\boldsymbol{C}}|\boldsymbol{C} \sim N_R(\boldsymbol{C}, \hat{V}) \tag{3}$$

where the 'true' uncentered averages $\boldsymbol{C}$ can be written as follows:

$$\boldsymbol{C}|\mu, \boldsymbol{v} = \mu \mathbf{1} + \boldsymbol{v} \tag{4}$$

Where μ is a scalar that is the mean of the 'true' uncentered averages $\boldsymbol{C}$.

$\boldsymbol{v}$ is a zero-centered mixture of multivariate normals:

$$\boldsymbol{v}|\boldsymbol{\pi}, \mathbf{U} \sim \sum_{\mathbf{k},\mathbf{l}} \pi_{\mathbf{k},\mathbf{l}}\ N_{\mathbf{R}}(\mathbf{0}, \omega_{\mathbf{l}} \mathbf{U}_{\mathbf{k}}) \tag{5}$$

Critically, our quantity of interest now, $\boldsymbol{v}$ represents the true 'deviations' from average gene expression across each condition and can be seen as the effects in `mash`.

3 Applications

We will again apply a two-step process to our selection of covariance matrices, where we select a set of denoised 'pattern' matrices U_k by using the EM algorithm on the max effects across conditions, and then expanding this list by a fixed grid of scalar weights ω_l such that we conclude with a list of $P = KxL$ covariance matrices Σ. We can then:

- estimate the p prior weights $\boldsymbol{\pi}$ on this fixed P-list of covariance matrices from a training matrix of randomly selected feature expression measurements across conditions
- compute the posterior distribution $\boldsymbol{v}|L\hat{\boldsymbol{C}}, \boldsymbol{s_j}$

  Let

  $$L\boldsymbol{C} = L\mu\mathbf{1} + L\boldsymbol{v} \tag{6}$$

  L is the RxR centering matrix $L_r = I_r - \frac{1}{r}\mathbf{1}\mathbf{1}^\top$ which removes the mean of each R column vector.

  Then :

  $$\begin{aligned} L\boldsymbol{C} &= L\mu\mathbf{1} + L\boldsymbol{v} \\ L\boldsymbol{C} &= 0 + L\boldsymbol{v} \\ L\hat{\boldsymbol{C}} &= L\boldsymbol{v} + E \end{aligned} \tag{7}$$

  Where $E \sim \mathcal{N}(0, L\hat{V}L')$

3.1 Selecting The Covariance Matrices

We initiate our set of covariance matrices for the denoising step as before in `mash`, where now we compute the empirical covariance matrices and a variety of dimensional reductions on the feature-centered JxR *matrix* of maximum average, $L\hat{C}'$ instead of $\hat{C}$ alone. In practice, we actually use the matrix of maximum uncentered Z statistics.

Now for each gene J at each component k, integrating over $\boldsymbol{v}$,

$$\begin{aligned} L\boldsymbol{C} &\sim \mathcal{N}(0, LU_kL') \\ L\hat{\boldsymbol{C}} &\sim \mathcal{N}(0, LU_kL' + L\hat{V}L') \end{aligned} \tag{8}$$

And thus we can use the Bovy et al algorithm invoked in both the Extreme Deconvolution package and in 'Sarah's MixEm' where:

$$T_{jp} = LU_kL' + L\hat{V}_jL' \tag{9}$$

And as mentioned, L is the RxR centering matrix for each gene, and $w_j = L\hat{\boldsymbol{C}}_j$.

Recall that our previous approach was simplified by the fact that $\boldsymbol{w}_j$ was simply $\hat{\boldsymbol{b}_j}$ and the projection matrix was simply the I_r identity matrix. Our inference on $\boldsymbol{b}$ was analogous to their inference on $_j$.

As before, we are interested in returning the prior covariance U_k matrices of the 'true' deviations $\boldsymbol{v}$, which we will then rescale by choosing a set of ω that are appropriate to $L\hat{\boldsymbol{C}}$ to comprise a set of $P = KxL$ prior covariance matrices Σ.

and choose the set of π that maximizes compute the following likelihood at each of the P components:

$$L\hat{\boldsymbol{C}}_j \sim \mathcal{N}(0, L\Sigma_pL' + L\hat{V}_jL') \tag{10}$$

4 Posteriors

Now, as before we can compute a posterior distribution such that:

$$\boldsymbol{v}|L\hat{\boldsymbol{C}}, \pi, \Sigma, \boldsymbol{s} \sim N(\mu^1, U^1) \tag{11}$$

Where at each of the P components for each gene J

$$\begin{aligned} \mu^1_{jp} &= \Sigma_p L' T^{-1}_{jp} L\hat{\boldsymbol{C}}_j \\ U^1_{jp} &= \Sigma_p - \Sigma_p L' T^{-1}_{jp} L\Sigma_p \end{aligned} \tag{12}$$

In our old code, L was just the identity:

$$\begin{aligned} \mu^1_{jp} &= \Sigma_p T^{-1}_{jp} \hat{\boldsymbol{C}}_j \\ U^1_{jp} &= \Sigma_p - \Sigma_p T^{-1}_{jp} \Sigma_p \end{aligned} \tag{13}$$

and was equivalent to :

$$\begin{aligned} U^1_{jp} &= (\Sigma_p^{-1} + \hat{V}_j^{-1})^{-1} \\ \mu^1_{jp} &= U^1_{jp}(\hat{V}_j^{-1}\boldsymbol{w}) \end{aligned} \tag{14}$$

If we now replace

$$\begin{aligned} \Sigma{*}_p &= L\Sigma_p L' \\ \hat{V}{*} &= L\hat{V}_j L' \\ \boldsymbol{w} &= L\hat{\boldsymbol{C}}_j (15) \end{aligned}$$

And let the new L be the identity, we would get the following:

$$\begin{aligned} \mu^1_{jp} &= \Sigma *_p T_{jp}^{-1} L\hat{\boldsymbol{C}}_j \\ U^1_{jp} &= \Sigma *_p - \Sigma *_p T_{jp}^{-1} \Sigma{*}_p \end{aligned} \tag{16}$$

But as you can see that isn't going to work because in the Bovy equation, Σ should represent the prior covariance of the ’“true‘ deviation v, while here, $\Sigma*$ represent the covariance of $\boldsymbol{C}$, L which is $L\Sigma * L$

Then we can return our old posteriors:

$$\begin{aligned} U^1_{jp} &= (\Sigma_p^{-1} + \hat{V}_j^{-1})^{-1} \\ \mu^1_{jp} &= U^1_{jp}(\hat{V}_j^{-1}\boldsymbol{w}) \end{aligned} \tag{17}$$

And

$$\boldsymbol{v} \sim \sum_p \tilde{\pi}_p \mathcal{N}(\mu^1_p, U^1_p) \tag{18}$$

Where

$$\tilde{\pi}_{jp} = \frac{\mathcal{N}(L\hat{\boldsymbol{C}}; 0, T_{jp})}{\sum_p \mathcal{N}(L\hat{\boldsymbol{C}}; 0, T_{jp})} \tag{19}$$

5 Differences required over `mash` implementation

- We will now work with a matrix of observed column-centered gene averages, $L\hat{C}'$ in order to:

1. initialize our choice of U_k;
2. choose the maxes by which to denoise,
3. choose our set of scales, ω_l
4. compute our hierarchical weights, $\boldsymbol{\pi}_p$ as well as our posteriors.

- The new distribution we seek to estimate for each j is then $v|\boldsymbol{L\hat{C}}, \boldsymbol{s}_j$

Some questions:

- How is using $\boldsymbol{w}_j$ as $\boldsymbol{L\hat{C}}_j$ as the data from which we estimate our model different than initializing with the vectors of $\hat{\boldsymbol{b}}$ that have been computed on *feature centered data*?
- I think it is because we are now broadly incorporating the centering information into our prior on $\boldsymbol{LC}$ as well, such that at each component for each gene, $\boldsymbol{LC}_j \sim \mathcal{N}(0, L\Sigma_{jp}L')$.
- If this is the case, we will probably need to denoise all of the initiation matrices (not just the multirank Uk) since previously our projection matrix was the Identity
- Since our input will still be that matrix of uncentered noisy averages and their standard errors, our scaling parameter ω ought to be chosen consistent with $L\hat{C}$, and not $\hat{C}$, since this will tend to scale with the true deviations $\boldsymbol{v}_j$.