Problem Set-up : Two Variance Components

Two variance components

The $p \times n$ data matrix Y has measurements in p probes for n individuals. Then we use the following model:

$$Y_{ij} = \phi_i \Theta_{ij} + \epsilon_{ij} \tag{1}$$

ϕ_i is the i'th probe effect where Θ_{ij} is a true count. ϵ_{ij} is an error term that follows a distribution of $\sim N(0, \delta_j^2 + \epsilon_i^2)$. Our objective is therefore to minimize the following:

$$\sum_{i=1}^{p} \sum_{j=1}^{n} \frac{\left(Y_{ij} - \phi_i \Theta_{ij}\right)^2}{\sigma_i^2 + \delta_j^2} + \lambda \frac{\|\Theta_{i+1,\cdot} - \Theta_{i,\cdot}\|}{w_i} \tag{2}$$

Alternating Descent

We will build on the existing algorithm in that we add a conditional update of σ and δ given ϕ and Θ. Therefore, our three steps will be

- Update σ and δ given ϕ and Θ through likelihood maximization with gradient descent
- Update Θ given ϕ, σ, δ through Group Fused Lars
- Update ϕ given Θ, σ, δ by solving least squares

Estimating σ and δ given Θ and ϕ

Assuming normality and mean 0 in the data Y, we can estimate σ and δ through maximizing the likelihood. The log-likelihood of Y is

$$\ell(Y; \sigma, \delta) = -\frac{np}{2} log(2\pi) - \frac{1}{2} \sum_{i,j} log(\sigma_i^2 + \delta_j^2) - \sum_{i,j} \frac{Y_{ij}^2}{2(\sigma_i^2 + \delta_j^2)} \tag{3}$$

and this leads to the gradient functions

$$\frac{\partial \ell}{\partial \sigma_i} = \sum_j \frac{-\sigma_i^3 + \sigma_i (Y_{ij}^2 - \delta_j^2)}{(\sigma_i^2 + \delta_j^2)^2} \tag{4}$$

$$\frac{\partial \ell}{\partial \delta_j} = \sum_i \frac{-\delta_j^3 + \delta_j (Y_{ij}^2 - \sigma_i^2)}{(\sigma_i^2 + \delta_j^2)^2} \tag{5}$$

In order to make the above score functions to be zero, and assuming that σ_i and δ_j are not 0 for all i and j,

$$\sum_{j=1}^{n} \frac{\sigma_i^2}{(\sigma_i^2 + \delta_j^2)^2} = \sum_{j=1}^{n} \frac{Y_{ij}^2 - \delta_j^2}{(\sigma_i^2 + \delta_j^2)^2} \tag{6}$$

$$\sum_{i=1}^{p} \frac{\delta_j^2}{(\sigma_i^2 + \delta_j^2)^2} = \sum_{i=1}^{p} \frac{Y_{ij}^2 - \sigma_i^2}{(\sigma_i^2 + \delta_j^2)^2} \tag{7}$$

Codes implementing gradient descent with line search based on the above are implemented in cnvJoint.

Estimating Θ given ϕ, σ and δ

In order to take advantage of the Group Fused Lars algorithm, we need to define the $(p-1) \times p$ change matrix Δ derived from the matrix Θ :

$$\Delta_{i\cdot} = \frac{\Theta_{i+1\cdot} - \Theta_{i\cdot}}{w_i} \tag{8}$$

and naturally

$$\Theta_{i\cdot} = \Theta_{1\cdot} + \sum_{s=0}^{i-1} w_s \Delta_{s\cdot} \tag{9}$$

Our objective is then

$$Q_2(\phi, \Theta_{1\cdot}, \Delta) = \sum_{i=1}^{p} \sum_{j=1}^{n} \left(\frac{Y_{ij} - \phi_i \Theta_{1j} - \phi_i \sum_{x=0}^{i-1} w_s \Delta_{s\cdot}}{\sqrt{\sigma_i^2 + \delta_j^2}} \right)^2 + \lambda \sum_{i=1}^{p-1} \|\Delta_{i\cdot}\|_2 \tag{10}$$

where $\Theta_{1\cdot}$ given Δ and ϕ is like below.

$$\Theta_{1\cdot} = \frac{\sum_{i=1}^{p} \phi_i Y_{i\cdot} - \sum_{i=1}^{p} \phi_i^2 \sum_{s=0}^{i-1} w_s \Delta_{s\cdot}}{\sum_{i=1}^{p} \phi_i^2} = \frac{\phi^T Y - \phi^T T \Delta}{\phi^T \phi} \tag{11}$$

where T is known:

$$T = \begin{bmatrix} 0 & 0 & 0 & ... & 0 \\ \phi_2 w_1 & 0 & 0 & ... & 0 \\ \phi_3 w_1 & \phi_3 w_2 & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ \phi_{p-1} w_1 & \phi_{p-1} w_2 & \phi_{p-1} w_3 & ... & 0 \\ \phi_p w_1 & \phi_p w_2 & \phi_p w_3 & ... & \phi_p w_{p-1} \end{bmatrix}$$

So far, we only focused on removing Θ and write the objective with Δ in order to simplify the penalty term. Our final objective with no Θ term looks like this.

$$Q_3(\phi, \Delta, \sigma, \delta) = \sum_{i=1}^{p} \sum_{j=1}^{n} \left(\frac{Y_{ij} - \phi_i \left(\frac{\phi^T Y_{\cdot j}}{\phi^T \phi} - \frac{\phi^T T \Delta_{\cdot j}}{\phi^T \phi} \right) - \phi_i \sum_{s=0}^{i-1} w_s \Delta_{sj}}{\sqrt{\sigma_i^2 + \delta_j^2}} \right)^2 + \lambda \sum_{i=1}^{p-1} \|\Delta_{i\cdot}\|_2 \tag{12}$$

Above is equivalent to

$$Q_3(\phi, \Delta, \sigma, \delta) = \sum_{i=1}^{p} \sum_{j=1}^{n} \frac{((Q_\phi Y - Q_\phi T \Delta)_{i,j})^2}{\sigma_i^2 + \delta_j^2} + \lambda \sum_{i=1}^{p-1} \|\Delta_{i\cdot}\|_2 \tag{13}$$

with $Q_\phi = I - (\phi^T \phi)^{-1} \phi \phi^T$.

Our goal is to modify the above objective $min_\Delta Q_3(\phi, \Delta, \sigma, \delta)$ to something like below with known A and B.

$$min_\Delta \|A - B\Delta\|_F^2 + \lambda \sum_{i=1}^{p} \|\Delta_{i,\cdot}\| \tag{14}$$

Therefore, we need to transform $Q_\phi Y$ and $Q_\phi T$ in terms of σ and δ so that, with $A(\phi, \sigma, \delta, w)$ and $B(\phi, \sigma, \delta, w)$,

$$\sum_{i=1}^{p} \sum_{j=1}^{n} \frac{((Q_\phi Y - Q_\phi T \Delta)_{i,j})^2}{\sigma_i^2 + \delta_j^2} = \|A - B\Delta\|_F^2 \tag{15}$$

Note the dimension of the matrices

$$\begin{aligned} Q_\phi &: p \times p \\ Y &: n \times p \\ T &: p \times (p-1) \\ \Delta &: (p-1) \times n \end{aligned}$$

- **get A**
  Since Y has dimension $p \times n$, we can do element-wise division of Y to get

  $$\tilde{Y} = Y \odot \Lambda$$

  where $\odot$ is Hadamard multiplication and Λ is the $n \times p$ variance matrix so that

  $$\Lambda_{ij} = \sqrt{\frac{1}{\sigma_i^2 + \delta_j^2}}$$

  Then,

  $$A = Q_\phi \tilde{Y}$$

- **get B**
  Since we need to keep Δ intact, we will have to modify $Q_\phi T$ which is a $p \times (p-1)$ matrix. Since the variance component has the cell index j, we will need n different versions of $Q_\phi T$ scaled differently for each j. This leads to a three-dimensional array $p \times (p-1) \times n$, and I don't think Group Fused Lars can handle this? :(

Data Simulation

We assume the model (1) is true, and create data Y according to the model. The simulation follows the process below.

- Fix the number of probes p and the number of cells n where $p > n$.
- Determine $\frac{n}{2}$ change points by taking a random sample from 1 to n.
- Create Θ. First divide p probes into blocks according to the change points earlier determined. Then sample a random integer from -10 to 10 for each block. All cells share the same Θ. ($\Theta_{\cdot j} = \Theta_{\cdot k}$ for all $j, k = 1, .., n$)
- Create ϕ by sampling p numbers from $N(0, 1)$.
- Create cell-specific true variance δ_j^2 from $unif(0.1, 1)$ for $j = 1, .., n$
- Create probe-specific true variance σ_i^2 from $unif(0.1, 1)$ for $i = 1, .., p$
- Sample $\epsilon_{ij} \sim N(0, \delta_j + \sigma_i)$
- Finally, make data Y by the formula $Y_{ij} = \phi_i \Theta_{ij} + \epsilon_{ij}$

Problem set-up : Add cell-specific fixed effects

$$Y_{ij} = \Theta_{ij}(\phi_i + \xi_j) + \epsilon_{ij},\ \epsilon_{ij} \sim N(0, \sigma^2) \tag{16}$$

$$min_{\phi,\xi,\Theta}\ (Y_{ij} - (\phi_i + \xi_j)\Theta_{ij})^2 + \lambda \frac{\|\Theta_{i+1,\cdot} - \Theta_{i,\cdot}\|}{w_i} \tag{17}$$

Alternating Descent

- Update Θ given (ϕ and ξ_j) through group fused lasso
- Update ϕ given (Θ and ξ) through solving least squares below where the left hand side is known

$$Y_{ij} - \Theta_{ij}\xi_j = \Theta_{ij}\phi_i + \epsilon_{ij}$$

- Update ξ given (Θ and ϕ) through solving least squares

$$Y_{ij} - \Theta_{ij}\phi_i = \Theta_{ij}\xi_j + \epsilon_{ij}$$

Modify group fused lasso to accommodate ξ

Define

$$\Theta_{ij} = \Theta_{1j} + \sum_{s=0}^{i-1} w_s \Delta_{sj}$$

Then our problem becomes

$$\min_{\Theta} \sum_{i,j} \left(Y_{ij} - (\phi_i + \xi_j)\Theta_{1j} - (\phi_i + \xi_j) \sum_{s=0}^{i-1} w_s \Delta_{sj} \right)^2 \tag{18}$$

and our goal is to change this in the form of $\|A - B\Delta\|_F^2$. Given ϕ, ξ, and Δ, we can solve for θ_{1j} like below

$$\begin{aligned} \Theta_{1j} &= \frac{\sum_i (\phi_i + \xi_j) Y_{ij} - \sum_i (\phi_i + \xi)^2 \sum_{s=0}^{i-1} w_s \Delta_{sj}}{\sum_i (\phi_i + \xi_j)^2} \\ &= \frac{\phi^T Y_{\cdot j} + (\xi_j 1_p^T) Y_{\cdot j} - \phi^T T \Delta_{\cdot j} - w \sum_i \phi_i \xi_j \sum_{s=0}^{i-1} w_s \Delta_{sj} + \sum_i \xi_j^2 \sum_{s=0}^{i-1} w_s \Delta_{sj}}{\phi^T \phi + \phi^T (\xi_j 1_p) + \xi_j^2} \end{aligned}$$

and this leads to the following vectorization

$$\Theta_{1\cdot} = \left(\phi^T Y + 1_p^T Y \Xi - \phi^T T \Delta - 2(\phi^T W \Delta + 1_p^T W \Delta \Xi) + 1_p^T W \Delta \Xi^2 \right) \oslash \left(\phi^T \phi 1_n + (\phi^T 1_p)\xi + \Xi \xi \right)$$

by introducing W matrix and Ξ matrix as following

$$W_{p\times(p-1)} = \begin{bmatrix} 0 & 0 & 0 & ... & 0 \\ w_1 & 0 & 0 & ... & 0 \\ w_1 & w_2 & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ w_1 & w_2 & w_3 & ... & 0 \\ w_1 & w_2 & w_3 & ... & w_{p-1} \end{bmatrix}$$

$$\Xi_{n\times n} = \begin{bmatrix} \xi_1 & 0 & ... & 0 \\ 0 & \xi_2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & \xi_n \end{bmatrix}$$

Also, the last part $(\phi_i + \xi_j)\sum_{s=0}^{i-1} w_s\Delta_{sj}$ can also be vectorized like the following.

$$\phi_i \sum_{s=0}^{i-1} w_s \Delta_{s\cdot} + \sum_{s=0}^{i-1} w_s \Delta_{s\cdot}\Xi$$

Lastly, note that the matrix T is equal to $1\phi^T W$

Combining the above vectorizations, we can re-write the objective (18):

$$\sum_i \left\| Y_{i\cdot} - \phi_i \left(\frac{\phi^T Y + 1_p^T Y\Xi - \phi^T 1 \phi^T W\Delta - 2\phi^T W\Delta - 21_p^T W\Delta\Xi + 1_p^T W\Delta\Xi^2}{\phi^T\phi 1_n + (\phi^T 1_p)\xi + \Xi\xi} \right) - \phi_i \sum_{s=0}^{i-1} w_s\Delta_{s\cdot}(I_n + \Xi) \right\|^2$$

$$= \sum_i \left\| Y_{i\cdot} - \phi_i \left(\frac{\phi^T Y + 1_p^T Y\Xi + (-\phi^T 1\phi^T - 2\phi^T)W\Delta - 1_p^T W\Delta(2\Xi - \Xi^2)}{\phi^T\phi 1_n + (\phi^T 1_p)\xi + \Xi\xi} \right) - 1\phi^T W\Delta(I_n + \Xi) \right\|^2$$

where the fraction shows element-wise division.

Problem set-up : Add cell-specific perturbations

$$Y_{ij} = (\Theta_{ij} + \xi_j)\phi_i + \epsilon_{ij},\ \epsilon_{ij} \sim N(0, \sigma^2) \tag{19}$$

$$min_{\phi,\xi,\Theta} \sum_{i,j}(Y_{ij} - \xi_j\phi_i - \phi_i\Theta_{ij})^2 + \sum_{i}^{p} \lambda \frac{\|\Theta_{i+1\cdot} - \Theta_{i\cdot}\|}{w_i} \tag{20}$$

Alternating Descent

- Update Θ given (ϕ and ξ) through group fused lasso
  This is exactly same as the original problem if we replace Y_{ij} with $Y_{ij} - \phi_i\xi_j$ for each i and j with known ϕ_i and ξ_j.
- Update ϕ given (ξ and Θ) by solving least squares
  $$\sum_{i,j}(Y_{ij} - (\Theta_{ij} + \xi_j)\phi_i)^2$$
  $$\Rightarrow \phi_i = \frac{\sum_j(\Theta_{ij} + \xi_j)Y_{ij}}{\sum_j(\Theta_{ij} + \xi_j)^2}$$
- Update ξ given (ϕ and Θ) by solving least squares
  $$\sum_{i,j}(Y_{ij} - \phi_i\Theta_{ij} - \phi_i\xi_j)^2$$
  $$\Rightarrow \xi_j = \frac{\sum_i(Y_{ij} - \phi_i\Theta_{ij})\phi_i}{\sum_i \phi_i^2}$$

Constraints for identifiability

We limit the L2 norm of ϕ to the length of the vector ϕ, and we constrain the mean of ξ to be 0.

Performance Comparison

Run the algorithm for

$$min_{\phi,\xi,\Theta} \sum_{i,j}(Y_{ij} - \xi_j\phi_i - \phi_i\Theta_{ij})^2 + \sum_{i}^{p} \lambda \frac{\|\Theta_{i+1\cdot} - \Theta_{i\cdot}\|}{w_i} \tag{21}$$

and

$$min_{\phi,\Theta} \sum_{i,j}(Y_{ij} - \phi_i\Theta_{ij})^2 + \sum_{i}^{p} \lambda \frac{\|\Theta_{i+1\cdot} - \Theta_{i\cdot}\|}{w_i} \tag{22}$$

And label the estimates from each algorithm (1) and (2). We compare $\Theta^{(1)}$ and $\Theta^{(2)}$, and so on.

Now compare,

- We look at $\xi_j^{(1)}\phi_i^{(1)} + \phi_i^{(1)}\Theta_{ij}^{(1)}$ and $\phi_i^{(2)}\Theta_{ij}^{(2)}$, and how close they are to Y_{ij} for each i and j (equivalent to comparing RSS).
- We look at $\Theta^{(1)}$ and $\Theta^{(2)}$ only which in theory should represent the true copy number of each sample.
- We look at $\Theta^{(1)} + \xi$ and $\Theta^{(2)}$
- We look at the minimized value that includes the penalty term, or the optimized value of (21) and (22).

1 Results

We ran the chromosome 22 data on both models, with and without ξ. For the reference, the points and lines labeled 'after xi' is

$$Y_{ij} = (\Theta_{ij} + \xi_j) * \phi_i + \epsilon_{ij}$$

and those labeled 'before xi' is

$$Y_{ij} = \Theta_{ij} * \phi_i + \epsilon_{ij}$$

In this section, we only focus on the first four samples and the last four samples (1, 2, 3, 4, 29, 30, 31, 32). This is designed to check the performance on the two distinct cell types. It is also appropriate in that top 5 samples with the largest ξ in magnitude are 3, 27, 12, 23, and 30, and two of them (samples 3 and 30) are included in our panels.

The first thing I checked was the AIC, BIC, and RSS error plots in Figure 1. The red line is after implementing ξ.

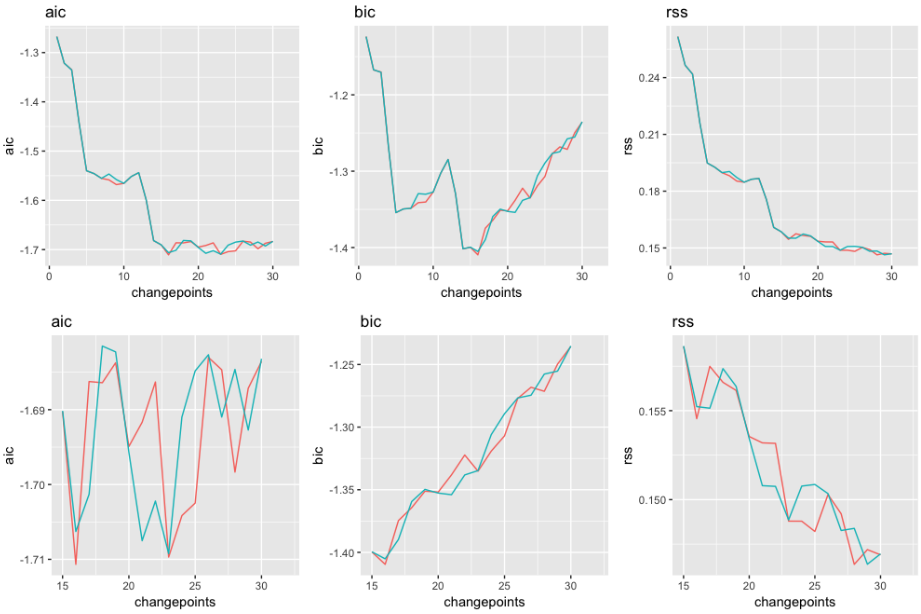

Figure 1: The error rates are mostly similar before and after the implementation if xi. Even when we zoom into the low-error region, it is difficult to decide which method performs better. Both the minimum of aic and bic error is achieved by the model with xi, but the error is so small that it seems possible that they're due to numercal instability.

Then, we checked $\Theta_{ij}\phi_i$. The primary goal is to check the alignment with zero. The original model without cell specific effects seem to perform better assuming that certain sections actually have true values of 0. See Figure 3.

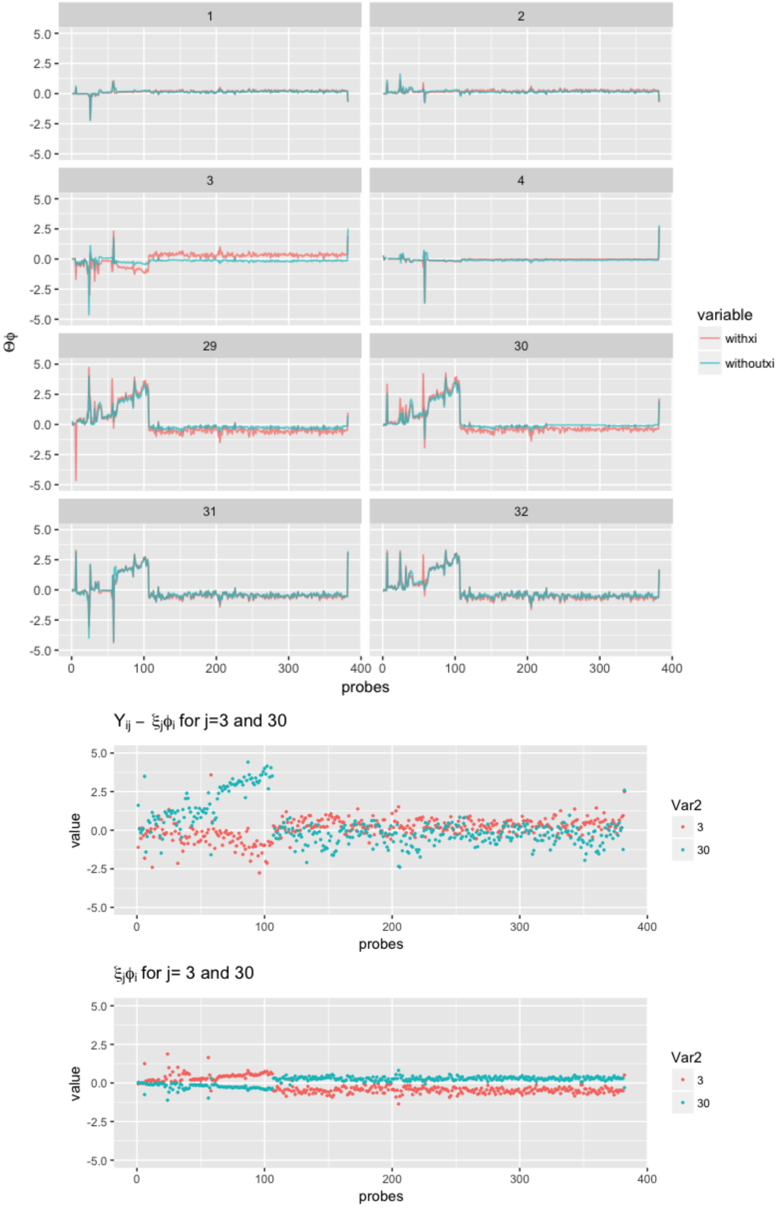

Figure 2: We assume the first four samples have zero over the entire chromosome. I also expect that for samples 29 to 32, probes from around 100 to the end have theta values zero. Some results are similar while samples 3, 29, and 30 show some difference. Overall, the model with ξ shows more variability, especially for samples where xi is large.

Finally, we compare Θ only. Θ is supposed to represent the true values of the copy number, so it is important that it is aligned at 0.

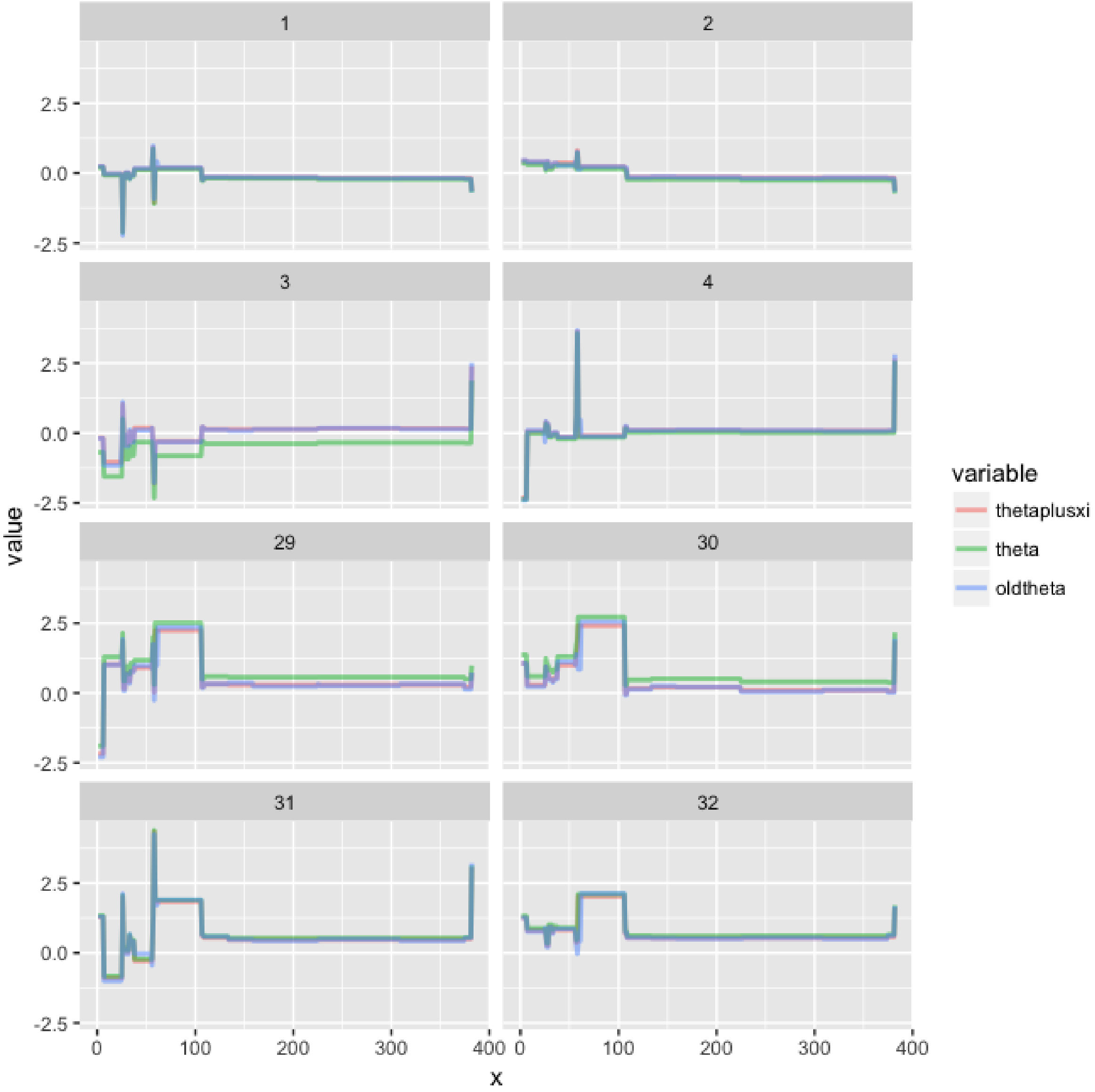

Figure 3: Again, they're very similar to one another except some samples including 3 and 30. And unfortunately, $\Theta + \xi$ is much more well aligned to zero than Θ. The sample-specific effect ξ is not performing as we expected.

2 Adding positive constraint on ϕ

2.1 Model

We model the copy number data for probe i of sample j like the following. Y_{ij} is the data we observe for sample j's probe i, and Θ_{ij} is the underlying true copy number for probe i in sample j. ϕ is a probe-effect while ξ is a cell-specific shifts which is the main attribution of this package.

$$Y_{ij} = (\Theta_{ij} + \xi_j)\phi_i + \epsilon_{ij},\ \epsilon_{ij} \sim N(0, \sigma^2) \tag{23}$$

In order to find make recover the piece-wise constant Θ, we solve the following penalized regression equation under the assumption that if there are change points in copy number, it is likely to be shared among the samples.

$$min_{\phi,\xi,\Theta} \sum_{i,j} (Y_{ij} - \xi_j\phi_i - \Theta_{ij}\phi_i)^2 + \sum_{i=1}^{p} \lambda \frac{\|\Theta_{i+1\cdot} - \Theta_{i\cdot}\|}{w_i} \tag{24}$$

2.2 Alternating Descent

We use an alternating descent algorithm to update Θ, ϕ, and ξ [?].

- **Update Θ given (ϕ and ξ) through Group Fused Lars [?]**
  If we substitute Y_{ij} to $Y_{ij} - \xi_j\phi_i$ for all i and j, the Group Fused Lars algorithm is the same with the method from Wang, Chen, & Zhao (2016) [?]

$$\Theta = argmin_{\Theta} \sum_{i,j} ((Y_{ij} - \xi_j\phi_i) - \phi_i\Theta_{ij})^2 + \sum_{i=1}^{p} \lambda \frac{\|\Theta_{i+1\cdot} - \Theta_{i\cdot}\|}{w_i}$$

- **Update ξ given (ϕ and Θ) by solving least squares**
  Similarly, we can get a closed form update for ξ.

$$argmin_{\xi_j} \sum_{i,j} (Y_{ij} - \phi_i\Theta_{ij} - \phi_i\xi_j)^2$$

$$= \frac{\sum_i (Y_{ij} - \phi_i\Theta_{ij})\phi_i}{\sum_i \phi_i^2}$$

- **Update ϕ given (ξ and Θ) by solving least squares**
  For all probe i, we can solve the least squares problem and get a closed-form update for ϕ_i.

$$argmin_{\phi_i} \sum_{i,j} (Y_{ij} - (\Theta_{ij} + \xi_j)\phi_i)^2 \text{ such that } \phi_i \geq 0 \text{ for all } i$$

  We can use quadratic programming to solve the above minimization problem. We need to construct the above in the form of

$$min_{\phi} \frac{1}{2}\phi^T H \phi + f^T \phi \text{ such that } A\phi \leq b$$

  To construct H, first define $\tilde{\Theta}$ as

$$\tilde{\Theta} = \begin{bmatrix} \Theta_{11} + \xi_1 & \cdots & \Theta_{1n} + \xi_n \\ \vdots & \ddots & \vdots \\ \Theta_{p1} + \xi_1 & \cdots & \Theta_{pn} + \xi_n \end{bmatrix}$$

and also define a j-specific vector D_j like below.

$$D_j = Q_j - R_j\phi = \begin{bmatrix} Y_{1j} \\ Y_{2j} \\ \vdots \\ Y_{pj} \end{bmatrix} - \begin{bmatrix} \tilde{\Theta}_{1j} & 0 & 0 & \cdots & 0 \\ 0 & \tilde{\Theta}_{2j} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & ... & \tilde{\Theta}_{pj} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_p \end{bmatrix}$$

Directly from the definition, e can see that

$$D_j^T D_j = \sum_{i=1}^{p} (Y_{ij} - \tilde{\Theta}_{ij}\phi_i)^2$$

and our optimization goal can be re-written as

$$argmin_\phi \sum_{j=1}^{n} D_j^T D_j$$

Therefore, we can construct H and f like the following.

$$\begin{aligned} & argmin_{\phi_i} \sum_{i,j} (Y_{ij} - (\Theta_{ij} + \xi_j)\phi_i)^2 \\ & \quad = argmin_\phi \sum_{j=1}^{n} D_j^T D_j \\ & \quad = argmin_\phi \sum_{j=1}^{n} (Q_j - R_j\phi)^T (Q_j - R_j\phi) \\ & \quad = argmin_\phi \phi^T (\sum_{j=1}^{n} R_j^T R_j)\phi - 2(\sum_{j=1}^{n} Q_j^T R)\phi + \sum_{j=1}^{n} Q_j^T Q_j \\ & \quad = argmin_\phi \phi^T \begin{bmatrix} \sum_{j=1}^{n} \tilde{\Theta}_{1j}^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sum_{j=1}^{n} \tilde{\Theta}_{pj}^2 \end{bmatrix} \phi - 2 \begin{bmatrix} \sum_{j=1}^{n} Y_{1j}\tilde{\Theta}_{1j} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sum_{j=1}^{n} Y_{pj}\tilde{\Theta}_{pj} \end{bmatrix} \phi \\ & \quad = argmin_\phi \phi^T \begin{bmatrix} \sum_{j=1}^{n} \tilde{\Theta}_{1j}^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sum_{j=1}^{n} \tilde{\Theta}_{pj}^2 \end{bmatrix} \phi - 2 \begin{bmatrix} \sum_{j=1}^{n} Y_{1j}\tilde{\Theta}_{1j} \\ \vdots \\ \sum_{j=1}^{n} Y_{pj}\tilde{\Theta}_{pj} \end{bmatrix}^T \phi \end{aligned}$$

Now we have our H, f, A, and b for the quadratic programming.

$$H = \begin{bmatrix} \sum_{j=1}^{n} \tilde{\Theta}_{1j}^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sum_{j=1}^{n} \tilde{\Theta}_{pj}^2 \end{bmatrix},\ f = \begin{bmatrix} \sum_{j=1}^{n} Y_{1j}\tilde{\Theta}_{1j} \\ \vdots \\ \sum_{j=1}^{n} Y_{pj}\tilde{\Theta}_{pj} \end{bmatrix},\ A = I_p,\ b = \mathbf{0}$$