A Model-Based Clustering via Mixture of Bayesian Hierarchical Models with Covariate Adjustment for Detecting Differentially Expressed Genes from Paired Design

Yixin Zhang¹, Wei Liu^{1,*}, and Weiliang Qiu²

¹York University, Department of Mathematics and Statistics, Toronto, M3J 1P3, Canada

²Sanofi Genzyme, Department of Biostatistics and Programming, Framingham, 01701, USA

*liuwei@yorku.ca

ABSTRACT

The causes of many complex human diseases are still largely unknown. Genetics plays an important role in uncovering the molecular mechanisms of complex human diseases. A key step to characterize the genetics of a complex human disease is to unbiasedly identify disease-associated gene transcripts in whole-genome scale. Confounding factors could cause false positives. Paired design, such as measuring gene expression before and after treatment for the same subject, can reduce the effect of known confounding factors. Model-based clustering, such as mixtures of Bayesian hierarchical models, has been proposed to detect gene transcripts differentially expressed between paired samples. However, not all known confounding factors can be controlled in a paired/match design. To the best of our knowledge, no clustering methods have the capacity to adjust for the effects of covariates yet. In this article, we proposed a novel mixture of Bayesian hierarchical models with covariate adjustment in identifying differentially expressed transcripts using high-throughput whole-genome data from paired design. Both simulation study and real data analysis show the good performance of the proposed method

Introduction

Genome-wide differential gene expression analysis is widely used for the elucidation of the molecular mechanisms of complex human diseases. One popular and powerful approach to detect differentially expressed genes is the probe-wise linear regression analysis combined with the control of multiple testing, such as limma¹. That is, we first perform linear regression for each probe and then adjust p-values for controlling multiple testing. One advantage of this approach is its capacity to adjust for potential confounding factors.

Another approach for detecting differentially expressed genes is the model-based clustering via mixture of Bayesian hierarchical model (MBHM)²⁻⁷, which can borrow information across genes to cluster genes. Probe clustering based on MBHMs treats gene transcripts as “samples” and samples as “variables”. Therefore, transcript clustering based on MBHMs has large number of “samples” and relatively small number of “variables”, hence does not have the curse-of dimensionality problem. In addition, unlike transcript-specific tests that have several parameters per transcript, transcript clustering based on MBHMs has only a few hyperparameters per cluster to be estimated and could borrow information across transcripts to estimate model hyperparameters. These approaches generally assume that samples under two groups are obtained independently.⁸ proposed a constrained MBHM to identify genetic outcomes measured from paired/matched designs.

Paired design is commonly used in study design for its homogeneous external environment for comparing measurements under different conditions. However, not all known confounding factors can be controlled in a paired/match design. Hence, we might still need to adjust the effects of confounding factors for data from a paired/matched design. To best of our knowledge, no existing clustering methods, including the constrained MBHM⁸, have the capacity to adjust for potential confounding factors, i.e., age, sex, and/or batch effect.

In this article, we proposed a novel mixture of Bayesian hierarchical models with covariate adjustment in identifying differentially expressed transcripts using high-throughput whole genome data from paired design.

Materials and Methods

We assumed that gene transcripts can be roughly classified into 3 clusters based on their expression levels in subjects after treatment (denoted as condition 1) relative to those before treatment (denoted as condition 2):

1. transcripts after treatment have higher expression levels than those before treatment, i.e., over-expressed (**OE**) in condition 1;
2. transcripts after treatment have lower expression levels than those before treatment, i.e., under-expressed (**UE**) in condition 1;
3. transcripts after treatment have same expression levels than those before treatment, i.e., non-differentially expressed (**NE**) between condition 1 and matched condition 2.

We followed⁸ to directly model the marginal distributions of gene transcripts in the 3 clusters. In⁸, they proposed a mixture of three-component Bayesian hierarchical distributions to characterize the within-pair difference of gene expression. We extended their model by incorporating potential confounding factors (such as *Age* and *Sex*) in the mixture of Bayesian hierarchical models, which might affect the response of gene expression to drug treatment. We assumed that data have been processed so that the distributions of mRNA expression levels are close to normal distributions. For RNAseq data, we can apply VOOM transformation⁹ or countTransformers¹⁰.

A mixture of Bayesian hierarchical models

For the g^{th} gene transcript, let x_{gl} and y_{gl} denote the expression levels of the l^{th} subject under two different conditions, e.g., before and after treatment, $g = 1, \dots, G$, $l = 1, \dots, n$, where G is the number of transcripts and n is the number of subjects (i.e., the number of pairs). Let $d_{gl} = \log_2(y_{gl}) - \log_2(x_{gl})$ be the log2 differences for the l^{th} subject. Denote $d_g = (d_{g1}, \dots, d_{gn})^T$. We assumed that d_{gl} is conditionally normally distributed given mean vector and covariance matrix. Let X^T be the $n \times (p+1)$ design matrix, where p is the number of covariates. The first column of X^T is the vector of ones, indicating intercept. Let η be the $(p+1) \times 1$ vector of coefficients for the intercept and covariate effects. We assume following mixture of three-component Bayesian hierarchical models:

For the OE gene transcripts,

$$\begin{aligned} d_g | (\mu_g, \tau_g) &\sim N(\mu_g, \tau_g^{-1} I_n) \\ \mu_g | \tau_g &\sim N(\exp[X^T \eta_1], k_1 \tau_g^{-1} I_n) \\ \tau_g &\sim \Gamma(\alpha_1, \beta_1) \end{aligned}$$

where $k_1 > 0$, $\alpha_1 > 0$ and $\beta_1 > 0$. $\Gamma(\alpha_1, \beta_1)$ denotes the Gamma distribution with shape parameter α_1 and rate parameter β_1 . That is, we assume that (1) the mean vectors μ_g , $g = 1, \dots, G$, given the variance τ_g^{-1} follow a multivariate normal distribution with mean vector $\exp[X^T \eta_1]$ and covariance matrix $k_1 \tau_g^{-1} I_n$; and (2) the variances τ_g^{-1} , $g = 1, \dots, G$, follow a Gamma distribution with shape parameter α_1 and rate parameter β_1 .

For the UE gene transcripts,

$$\begin{aligned} d_g | (\mu_g, \tau_g) &\sim N(\mu_g, \tau_g^{-1} I_n) \\ \mu_g | \tau_g &\sim N(-\exp[X^T \eta_2], k_2 \tau_g^{-1} I_n) \\ \tau_g &\sim \Gamma(\alpha_2, \beta_2) \end{aligned}$$

where $k_2 > 0$, $\alpha_2 > 0$, $\beta_2 > 0$, and X^T is the design matrix.

For the NE gene transcripts,

$$\begin{aligned} d_g | \tau_g &\sim N(U^T \theta_g, \tau_g^{-1} I_n) \\ \theta_g | \tau_g &\sim N(\eta_3, k_3 \tau_g^{-1} I_p) \\ \tau_g &\sim \Gamma(\alpha_3, \beta_3) \end{aligned}$$

where $k_3 > 0$, $\alpha_3 > 0$ and $\beta_3 > 0$. U^T is the design matrix without intercept column.

The hyper-parameters α_c and β_c are shape and rate parameters for the Gamma distribution, respectively, $c = 1, 2, 3$. As for k_1, k_2 and k_3 , the variation of the mean vector μ_g should be smaller than that of the observations d_g . So we expect $0 < k_c < 1$, $c = 1, 2, 3$.

Note that the marginal distribution for each component of the mixture is a multivariate t distribution with special structure of mean vector and covariance matrix [11, Section 3.7.6].

For continuous covariates, we require that they are standardized so that they have mean zero and variance one. Standardizing continuous covariates would make $\exp(X^T \eta_1)$ and $\exp(X^T \eta_2)$ be numerically finite.

Ideally, we should require $\mu_g > 0$ ($\mu_g < 0$) for all transcripts in cluster 1 (cluster 2). To do so, we can assume a log normal prior distribution for μ_g in cluster 1, for instance. However, a log normal distribution could not be a conjugate prior for the mean of a normal distribution. It would increase the computational burden if non-conjugate priors were used. As an alternative, we require the mean $E(\mu_g) > 0$ ($E(\mu_g) < 0$) for cluster 1 (cluster 2) by assuming $E(\mu_g)$ for cluster 1 (cluster 2) to be $\exp[X^T \eta_1]$ ($-\exp[X^T \eta_2]$). Denote $\psi = (\alpha_1, \beta_1, k_1, \eta_1^T, \alpha_2, \beta_2, k_2, \eta_2^T, \alpha_3, \beta_3, k_3, \eta_3^T)^T$.

Marginal density functions

Let $f_1(\mathbf{d}_g|\psi)$, $f_2(\mathbf{d}_g|\psi)$, $f_3(\mathbf{d}_g|\psi)$ be the marginal densities of the 3 Bayesian hierarchical models, and $\pi = (\pi_1, \pi_2, \pi_3)$ is the vector of cluster mixture proportions. Then the marginal density of $\mathbf{d}_g$ is :

$$f(\mathbf{d}_g|\psi) = \pi_1 f_1(\mathbf{d}_g|\psi) + \pi_2 f_2(\mathbf{d}_g|\psi) + \pi_3 f_3(\mathbf{d}_g|\psi).$$

Determining transcript cluster membership

The transcript-cluster membership is determined based on the posterior probabilities, $\zeta_{gc} = Pr(g^{th} \text{ gene transcript in cluster } c | d_g)$. We can get

$$\zeta_{gc} = \frac{\pi_c f_c(\mathbf{d}_g|\psi)}{\pi_1 f_1(\mathbf{d}_g|\psi) + \pi_2 f_2(\mathbf{d}_g|\psi) + \pi_3 f_3(\mathbf{d}_g|\psi)}, c = 1, 2, 3.$$

We determine a transcript's cluster membership as follows: If the maximum value among $\zeta_{gi}, i = 1, 2, 3$ is ζ_{gc} , then the transcript g belongs to cluster c .

The true values of π_1 , π_2 , π_3 , and ψ are unknown. We use estimated values to determine transcripts' cluster membership.

Parameter estimation via EM algorithm

We used expectation-maximization (EM) algorithm¹² to estimate the model parameters $\pi = (\pi_1, \pi_2, \pi_3)^T$ and ψ . Detailed derivations of the expectation Q of the log complete-data likelihood function and its first-order derivatives are shown in the Appendix.

We stop the expectation-maximization iterations if the sum of squared differences of model parameter estimates between current iteration and previous iteration is smaller than a small constant (e.g. 1.0×10^{-3}).

More details about the EM algorithm are shown in Appendix.

A real data study

We downloaded the dataset GSE24742¹³ from the Gene Expression Omnibus (<https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE24742>) to evaluate the performance of the proposed model-based clustering method (denoted as *eLNNpairedCov*).

The dataset is from a study that investigated the gene expression before and after administrating rituximab, a drug for treating anti-TNF resistant rheumatoid arthritis (RA). There are 12 subjects, each having 2 samples (one sample is before treatment and the other is after treatment). Age and sex are also available. Expression levels of 54,675 gene probes were measured for each of the 24 samples by using Affymetrix HUMAN Genome U133 Plus 2.0 array. The dataset has been preprocessed by the dataset contributor. We further kept only 43,505 gene probes in the autosomal chromosomes (i.e., chromosomes 1 to 22). We then performed log2 transformation for gene expression levels. We next obtained the within-subject difference of the log2 transformed expression levels (log2 expression after-treatment minus log2 expression before-treatment). By examining the histogram (Figure 5) of the estimated standard deviations of log2 differences of within-subject gene expression for the 43,505 gene probes, we found a bimodal distribution. We excluded gene probes having standard deviation < 1 . Finally, 23,948 gene probes kept in the down-stream analysis.

A simulation study

We performed a simulation study to compare the performance of the proposed method *eLNNpairedCov* with transcript-wise test *limma* and Li et. al.'s⁸ method (denoted as *eLNNpaired*). Both *eLNNpairedCov* and *limma* adjust covariate effects, while *eLNNpaired* does not.

The *limma* approach first performs an empirical-Bayes-based linear regression for each transcript. In this linear regression, the within-subject log2 difference of transcript expression is the outcome and intercept indicating if the transcript is over-expressed (intercept > 0), under-expressed (intercept < 0), or non-differentially expressed (intercept = 0), adjusting for potential confounding factors. Here condition 1 are observations after treatment and condition 2 are observations before treatment. A transcript is claimed as OE if its intercept estimate is positive and corresponding FDR-adjusted p-value < 0.05 , where

FDR stands for false discovery rate. A transcript is claimed as UE if its intercept estimate is negative and corresponding FDR-adjusted p-value < 0.05 . Other transcripts are claimed as NE.

In this simulation study, we considered two scenarios. In the first scenario (Scenario1), the number of subjects is equal to 30. In the second scenario (Scenario2), the number of subjects is equal to 100.

For each scenario, we generated 100 datasets. Each simulated dataset contains $G = 20,000$ gene transcripts, 0.0376% of which are OE and 0.346% of which are UE.

There are two covariates: standardized age (denoted as *Age.s*) and *Sex*. *Age.s* follows normal distribution with mean 0 and standard deviation 1. Seventy five percent (75%) of subjects are women.

The model hyperparameters are set as $\alpha_1 = 3.53$, $\beta_1 = 3.45$, $k_1 = 0.26$, $\eta_1 = (0.18, 0.00, -1.05)^T$, $\alpha_2 = 3.53$, $\beta_2 = 3.45$, $k_2 = 0.26$, $\eta_2 = (0.18, 0.00, -1.05)^T$, $\alpha_3 = 2.86$, $\beta_3 = 2.20$, $k_3 = 0.72$, and $\eta_3 = (-0.01, 0.00)^T$.

The parameter values (π , ψ , and proportion of women) in the simulation study are estimated from the analysis of the pre-processed real dataset GSE24742 described in Section .

Evaluation criteria

Two agreement indices and two error rates are used to compare the predicted cluster membership and true cluster membership of all genes. The two agreement indices are accuracy (i.e., proportion of predicted cluster membership equal to the true cluster membership) and Jaccard index¹⁴. For perfect agreement, these indices have a value of one. If an index takes a value close to zero, then the agreement between the true transcript cluster membership and the estimated transcript cluster membership is likely due to chance. The two error rates are false positive rate (FPR) and false negative rate (FNR). FPR is the percentage of detected DE transcripts among truly NE transcripts. FNR is the percentage of detected NE transcripts among truly DE transcripts. We also examined the user time and number of EM iterations for running each simulated dataset.

Results

Results of the real data analysis

For the real dataset, we adjusted standardized age and sex for *eLNNpairedCov* and *limma*. We standardized age so that it has mean zero and variance one. For each transcript, we also scaled its expression across subjects so that its variance is equal one.

The *limma* method detected 6 under-expressed gene transcripts (Figure 1 and Table 1), while *eLNNpaired* did not find any positive signals (i.e., $\hat{\pi}_3 = 1$). The proposed method *eLNNpairedCov* detected 9 OE transcripts (Upper panel of Figure 2 and Table 2) and 83 UE transcripts (Lower panel of Figure 2 and Table 3). The 83 UE transcripts detected by *eLNNpairedCov* include the 6 UE transcripts detected by *limma*.

Figure 1. Parallel boxplots of log2 within-subject difference of gene expression for 6 UE transcripts detected by *limma* for pre-processed GSE24742 dataset. Red horizontal line indicates log2 difference equal to zero.

Figure 2. Parallel boxplots of log2 within-subject difference of gene expression for differentially expressed transcripts detected by *eLNNpairedCov* for pre-processed GSE24742 dataset. Upper level: 9 OE transcripts; Lower panel: 83 UE transcripts. Red horizontal lines indicate log2 difference equal to zero.

It is assuring that several genes corresponding to the 92 DE transcripts identified by *eLNNpairedCov* have been associated to rheumatoid arthritis (RA) in literature. For example, Humby et. al. (2019)¹⁵ reported that genes *ZNF365* (OE), *IL36RN* (OE), *MRVII-AS1* (OE), *WFDC6* (UE), *UBE2H* (UE), are associated with RA.

Results of the simulation study

The results for Scenario 1 ($n = 30$) are shown in Figures 3 and 4, which are jittered scatter plots of the performance indices and the difference of performance indices versus methods, respectively.

Figure 3. Jittered scatter plots of performance indices versus method for Scenario1 (number of pairs= 30). Red solid horizontal lines indicate the median performance indices of *limma*.

Figure 4. Jittered scatter plots of difference of performance indices versus method for Scenario1 (number of pairs= 30). Red solid horizontal lines indicate y-axis equal to zero, except for bottom-right plot, in which red solid horizontal line indicates y-axis equal to one.

The differences of performance indices are between *limma* and the other 2 methods (*eLNNpaired* and *eLNNpairedCov*). A positive difference indicates that the performance indices of the other 2 methods are larger than that of *limma*. A negative difference indicates that the performance indices of the other 2 methods are smaller than that of *limma*.

The upper panel of Figures 3 and 4 show that *eLNNpairedCov* has higher agreement indices (Jaccard and accuracy) than *limma*, which in turn has higher agreement indices than *eLNNpaired*.

The middle panel of Figures 3 and 4 show that the three methods have similar FPR, except that *eLNNpaired* has larger FPR than *eLNNpairedCov* and *limma* for a few simulated datasets. The middle panel also show that *eLNNpairedCov* has smaller FNR than *limma*, which in turn has smaller FNR than *eLNNpaired*.

The bottom panel of Figures 3 and 4 show that *limma* runs very fast, while both *eLNNpaired* and *eLNNpairedCov* run in reasonable time (i.e., less than 30 seconds per dataset that has $G = 20,000$ genes and $n = 30$ subjects). On average *eLNNpairedCov* spends a little more time than *eLNNpaired*. The bottom panel of Figures 3 and 4 also show that *eLNNpairedCov* uses only 2 EM iterations, while *eLNNpaired* tends to use more EM iterations. Note that the EM iteration number for *limma* is set to be one, which does not use EM algorithm to obtain parameter estimates.

The simulation results for Scenario 2 ($n = 100$) are shown in Figures 6 and 7 in the Appendix, which have similar patterns to those for Scenario 1 ($n = 30$), except that both *eLNNpaired* and *eLNNpairedCov* have smaller FPR than *limma*. Note that *eLNNpairedCov* tends to have smaller FNR than *limma*, while *eLNNpaired* has huge FNR (close 1).

1 Discussion

In this article, we extended Li et al.'s (2017) mixture of Bayesian hierarchical models to allow the adjustment for potential confounding factors. This is novel in that to the best of our knowledge, all existing clustering methods do not yet have the capacity to adjust for covariates.

The proposed approach is different from transcript-wise test with controlling multiple testing in that it does not involve hypothesis testing. Hence, no multiple test controlling is needed. The simulation study showed that if the difference of gene expression between samples before treatment and samples after treatment follows the mixture of Bayesian hierarchical models

in Section , then the proposed method can outperform *limma*, which currently is a fast and powerful transcript-wise test method. The real data analysis also showed the proposed method *eLNNpairedCov* can detect more differentially expressed gene transcripts, which include the transcripts detected by *limma*.

Although we classify genes to three distinct clusters, the transitions between these clusters could be smooth. This would be reflected by a gene's posterior probability that might be large in two of three clusters, e.g., 0.49 for cluster 1, 0.01 for cluster 2, and 0.5 for cluster 3. On the other hand, expression changes could be split up into more than 3 clusters, e.g., groups behaving differently. In this article, we are only interested in identifying three clusters of genes: over-expressed in condition 1, under-expressed in condition 1, and non-differentially expressed.

There are other model-based clustering methods in literature, such as¹⁶. However, they were not designed to detect differentially expressed genes. For example, we can set the number K of clusters as 3 for their model. However, there is no constraints that the intercepts for the three clusters have to be positive, negative, and zero. That is, the three clusters identified might not correspond to over-expressed, under-expressed, and non-differentially expressed genes.

RNAseq and single-cell RNAseq data are cutting-edge tools to investigate molecular mechanisms of complex human diseases. However, it is quite challenging to analyze these count data with inflated zero counts. In future, we will evaluate if *eLNNpairedCov* can be used to analyze single-cell RNAseq data by first transforming counts to continuous scale (e.g., via VROOM⁹ or countTransformers¹⁰) and then to apply *eLNNpairedCov* to the transformed data.

We implemented the proposed methods to an R package *eLNNpairedCov*, which will be freely available to researchers.

References

1. Smyth, GK. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Stat. Appl. Genet. Mol. Biol.* **3**, Article 3 (2004).
2. Newton, MA, Kendziorski, CM, Richmond, CS, Blattner, FR & Tsui, KW. On differential variability of expression ratios: Improving statistical inference about gene expression changes from microarray data. *J. Comput. Biol.* **8**, 37–52 (2001).
3. Baldi, P & Long, AD. A bayesian framework for the analysis of microarray expression data: regularized t -test and statistical inferences of gene changes. *Bioinformatics* **17**, 509–519 (2001).
4. Kendziorski, CM, Newton, MA, Lan, H & Gould, MN. On parametric empirical bayes methods for comparing multiple groups using replicated gene expression profiles. *Stat. Medicine* **22**, 3899–3914 (2003).
5. Gottardo, R, Pannucci, JA, Kuske, CR & Brettin, T. Statistical analysis of microarray data: a bayesian approach. *Biostatistics* **4**, 597–620 (2003).
6. Lo, K & Gottardo, R. Statistical analysis of microarray data: a bayesian approach. *Bioinformatics* **23**, 328–335 (2007).
7. Zuyderduyn, S. Statistical analysis and significance testing of serial analysis of gene expression data using a poisson mixture model. *BMC bioinformatics* **8**, 282–282 (2007).
8. Li, Y *et al.* Detecting disease-associated genomic outcomes using constrained mixture of bayesian hierarchical models for paired data. *PLOS ONE* **12**, 1–16 (2017).
9. Law, CW, Chen, Y, Shi, W & Smyth, GK. voom: Precision weights unlock linear model analysis tools for rna-seq read counts. *Genome biology* **15**, R29 (2014).
10. Zhang, Z *et al.* Novel data transformations for rna-seq differential expression analysis. *Sci. Reports* **9**, 4820 (2019).
11. Lenk, P. Bayesian inference and markov chain monte carlo. <https://webuser.bus.umich.edu/plenk/Bam2%20Short.pdf> (2001).
12. Dempster, AP, Laird, NM & Rubin, DB. Maximum likelihood from incomplete data via the em algorithm. *J. Royal Stat. Soc. Ser. B* **39**, 1–38 (1977).
13. Gutierrez-Roelens I, L. B. Effects of rituximab on global gene expression profiles in the ra synovium. *NCBI* <https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE24742> (2010).
14. Milligan, GW & Cooper, MC. A study of the comparability of external criteria for hierarchical cluster analysis. *Multivar. Behav. Res.* **21**, 441–458 (1986).
15. Humby, F *et al.* Synovial cellular and molecular signatures stratify clinical response to csdmard therapy and predict radiographic progression in early rheumatoid arthritis patients. *Ann Rheum Dis.* **78**, 761–772 (2019).
16. Grün, B and Leisch, F. Fitting finite mixtures of generalized linear regressions in r. *Comput. Stat. & Data Analysis* **51**, 5247–5252 (2007).

17. Byrd, RH, Lu, P, Nocedal, J & Zhu, C. A limited memory algorithm for bound constrained optimization. *SIAM J. on Sci. Comput.* **16**, 1190–1208 (1995).
18. Petersen, KB & Pedersen, MS. The matrix cookbook. <https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf> (2012).

Author contributions statement

Conceptualization, W.L. and W.Q.; methodology, Y.Z., W.L. and W.Q.; software, Y.Z. and W.Q.; validation, Y.Z., W.L. and W.Q.; formal analysis, Y.Z. and W.Q.; investigation, W.L.; resources, W.L.; data curation, Y.Z.; writing—original draft preparation, Y.Z.; writing—review and editing, W.L. and W.Q.; visualization, Y.Z. and W.Q.; supervision, W.L. and W.Q.; project administration, W.L.; funding acquisition, W.L. All authors have read and agreed to the published version of the manuscript.

Funding

This article was supported by Canada Natural Sciences and Engineering Research Council (NSERC) grants 198662. W.Q. is a Sanofi employee and may hold shares and/or stock options in the company.

Data availability statement

The data presented in this study are openly available in [<https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE24742>]

Additional information

The authors declare no potential conflict of interests.

Appendix

Histogram of estimated standard deviations of log2 differences of within-subject gene expression for GSE24742

Figure 5. Histogram of the estimated standard deviations of log2 differences of within-subject gene expression for 43,506 gene probes in GSE24742.

Additional results

Figure 6. Jittered scatter plots of performance indices versus method for Scenario2 (number of pairs= 100). Red solid horizontal lines indicate the median performance indices of *limma*.

Figure 7. Jittered scatter plots of difference of performance indices versus method for Scenario2 (number of pairs= 100). Red solid horizontal lines indicate y-axis equal to zero, except for bottom-right plot, in which red solid horizontal line indicates y-axis equal to one.

Marginal density functions

The marginal density function of $\mathbf{d}_g$ is :

$$f(\mathbf{d}_g|\psi) = \pi_1 f_1(\mathbf{d}_g|\psi) + \pi_2 f_2(\mathbf{d}_g|\psi) + \pi_3 f_3(\mathbf{d}_g|\psi)$$

where $f_1(\mathbf{d}_g|\psi)$, $f_2(\mathbf{d}_g|\psi)$, $f_3(\mathbf{d}_g|\psi)$ are the marginal densities of the 3 clusters, and $\pi = (\pi_1, \pi_2, \pi_3)$ is the cluster proportion.

Then we have the marginal distribution for each cluster:

$$\begin{aligned} f_1(d_g|\psi) &= \int_{-\infty}^{+\infty} \int_0^{\infty} N(d_g; \mu_g, \tau_g^{-1} I_n) N(\mu_g; \exp(X^T \eta_1), k_1 \tau_g^{-1} I_n) \Gamma(\alpha_1, \beta_1) d\mu_g d\tau_g \\ &= \frac{\beta_1^{\alpha_1}}{|2\pi|^{\frac{n}{2}} \Gamma(\alpha_1)} (k_1 + 1)^{-\frac{n}{2}} \int_0^{\infty} \tau_g^{\frac{n}{2} + \alpha_1 - 1} e^{-\tau_g(\beta_1 - M_1)} \\ &= \frac{\beta_1^{\alpha_1} \Gamma(\frac{n}{2} + \alpha_1)}{|2\pi|^{\frac{n}{2}} \Gamma(\alpha_1)} (k_1 + 1)^{-\frac{n}{2}} (\beta_1 - M_1)^{-(\frac{n}{2} + \alpha_1)}, \end{aligned}$$

$$\begin{aligned} f_2(d_g|\psi) &= \int_{-\infty}^{+\infty} \int_0^{\infty} N(d_g; \mu_g, \tau_g^{-1} I_n) N(\mu_g; -\exp(X^T \eta_2), k_2 \tau_g^{-1} I_n) \Gamma(\alpha_2, \beta_2) d\mu_g d\tau_g \\ &= \frac{\beta_2^{\alpha_2} \Gamma(\frac{n}{2} + \alpha_2)}{|2\pi|^{\frac{n}{2}} \Gamma(\alpha_2)} (k_2 + 1)^{-\frac{n}{2}} (\beta_2 - M_2)^{-(\frac{n}{2} + \alpha_2)}, \end{aligned}$$

where X^T is the $n \times (p + 1)$ design matrix.

$$\begin{aligned} f_3(\mathbf{d}_g|\psi) &= \int_0^{\infty} \int_{-\infty}^{\infty} N(d_g; U^T \theta_g, \tau_g I_n) N(\theta_g; \eta_3, k_3 \tau_g^{-1} I_p G(\tau_g; \alpha_3, \beta_3)) d\theta_g d\tau_g \\ &= \frac{\beta_3^{\alpha_3} \Gamma(\frac{n}{2} + \alpha_3)}{\Gamma(\alpha_3) (2\pi)^{\frac{n}{2}}} |I_p + k_3 U U^T|^{-\frac{1}{2}} \\ &\quad \left[\beta_3 + \frac{1}{2} d_g^T d_g + \frac{1}{2k_3} \eta_3^T \eta_3 \right. \\ &\quad \left. - \frac{1}{2k_3} (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3) \right]^{-(\frac{n}{2} + \alpha_3)}, \end{aligned}$$

where U^T is the $n \times p$ design matrix without intercept column.

Here

$$\begin{aligned} M_1 &= \frac{1}{2} \left(d_g^T I_n + (e^{X^T \eta_1})^T \frac{1}{k_1} I_n \right) (I_n + \frac{1}{k_1} I_n)^{-1} \left(I_n d_g + \frac{1}{k_1} I_n e^{X^T \eta_1} \right) \\ &\quad - \frac{1}{2} \left(d_g^T I_n d_g + (e^{X^T \eta_1})^T \frac{1}{k_1} I_n e^{X^T \eta_1} \right) \\ &= -\frac{1}{2} \frac{1}{k_1 + 1} d_g^T d_g + \frac{1}{k_1 + 1} d_g^T e^{X^T \eta_1} - \frac{1}{2} \frac{1}{k_1 + 1} (e^{X^T \eta_1})^T e^{X^T \eta_1} \\ \\ M_2 &= \frac{1}{2} \left(d_g^T I_n - (e^{X^T \eta_2})^T \frac{1}{k_2} I_n \right) (I_n + \frac{1}{k_2} I_n)^{-1} \left(I_n d_g - \frac{1}{k_2} I_n e^{X^T \eta_2} \right) \\ &\quad - \frac{1}{2} \left(d_g^T I_n d_g + (e^{X^T \eta_2})^T \frac{1}{k_2} I_n e^{X^T \eta_2} \right) \\ &= -\frac{1}{2} \frac{1}{k_2 + 1} d_g^T d_g - \frac{1}{k_2 + 1} d_g^T e^{X^T \eta_2} - \frac{1}{2} \frac{1}{k_2 + 1} (e^{X^T \eta_2})^T e^{X^T \eta_2} \end{aligned}$$

Note that each marginal density is the density of a multivariate t distribution with special structures of mean vector and covariance matrix.

Log marginal densities

$$\log [f_1(d_g|\psi)] = \alpha_1 \log(\beta_1) + \log \left[\Gamma \left(\frac{n}{2} + \alpha_1 \right) \right] - \frac{n}{2} \log(k_1 + 1) - \left(\frac{n}{2} + \alpha_1 \right) \log(\beta_1 - M_1) - \frac{n}{2} \log(2\pi) - \log[\Gamma(\alpha_1)]$$

$$\log [f_2(d_g|\psi)] = \alpha_2 \log(\beta_2) + \log \left[\Gamma \left(\frac{n}{2} + \alpha_2 \right) \right] - \frac{n}{2} \log(k_2 + 1) - \left(\frac{n}{2} + \alpha_2 \right) \log(\beta_2 - M_2) - \frac{n}{2} \log(2\pi) - \log[\Gamma(\alpha_2)]$$

$$\log [f_3(d_g|\psi)] = \alpha_3 \log(\beta_3) + \log \left[\Gamma \left(\frac{n}{2} + \alpha_3 \right) \right] - \frac{n}{2} \log(2\pi) - \log[\Gamma(\alpha_3)] - \frac{1}{2} \log |I_p + k_3 U U^T| - \left(\frac{n}{2} + \alpha_3 \right) \log(\beta_3 + M_3)$$

Here

$$M_3 = \frac{1}{2} d_g^T d_g + \frac{1}{2k_3} \eta_3^T \eta_3 - \frac{1}{2k_3} (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3)$$

EM algorithm

Let $z_g = (z_{g1}, z_{g2}, z_{g3})$ to be the indicator vector indicating if gene transcript g belongs to a cluster or not. To stabilize the estimate of π when π_c is very small, we assume that the cluster mixture proportions π follows a symmetric Dirichlet $D(\mathbf{b})$ distribution, i.e., $f(\pi) = \frac{\Gamma(\sum_{c=1}^3 b_c)}{\prod_{c=1}^3 \Gamma(b_c)} \prod_{c=1}^3 \pi_c^{b_c-1}$. Therefore, the likelihood function for the complete data $(\mathbf{d}, \mathbf{z}, \pi)$ is

$$\begin{aligned} L(\psi|\mathbf{d}, \mathbf{z}, \pi) &= f(\mathbf{d}, \mathbf{z}, \pi|\psi) \\ &= f(\mathbf{d}, \mathbf{z}|\pi, \psi) f(\pi|\psi) \\ &= f(\mathbf{d}, \mathbf{z}|\pi, \psi) f(\pi) \\ &= \left(\prod_{g=1}^G f(\mathbf{d}_g, \mathbf{z}_g|\psi, \pi) \right) Dir(\mathbf{b}) \\ &= \left(\prod_{g=1}^G (\pi_1 f_1(\mathbf{d}_g|\psi))^{z_{g1}} (\pi_2 f_2(\mathbf{d}_g|\psi))^{z_{g2}} (\pi_3 f_3(\mathbf{d}_g|\psi))^{z_{g3}} \right) \frac{\Gamma(\sum_{c=1}^3 b_c)}{\prod_{c=1}^3 \Gamma(b_c)} \prod_{c=1}^3 \pi_c^{b_c-1} \end{aligned}$$

Then the log complete-data likelihood function is:

$$\begin{aligned} l(\psi|\mathbf{d}, \mathbf{z}, \pi) &= \sum_{g=1}^G ((z_{g1} \log f_1(\mathbf{d}_g|\psi) + z_{g2} \log f_2(\mathbf{d}_g|\psi)) + z_{g3} \log f_3(\mathbf{d}_g|\psi)) \\ &\quad + \sum_{g=1}^G (z_{g1} \log \pi_1 + z_{g2} \log \pi_2 + z_{g3} \log \pi_3) \\ &\quad + \log \left(\frac{\Gamma(\sum_{c=1}^3 b_c)}{\prod_{c=1}^3 \Gamma(b_c)} \right) + \sum_{c=1}^3 (b_c - 1) \log \pi_c \end{aligned}$$

The EM algorithm is used to estimate parameters π and ψ . Since z is unknown random vector, we integrate it out from the

log complete-data likelihood function. Here, $\mathbf{z}_g = (z_{g1}, z_{g2}, z_{g3})$.

$$\begin{aligned}\zeta_{g1} &= E(z_{g1} | \mathbf{d}_g, \boldsymbol{\pi}, \boldsymbol{\psi}) \\ &= Pr(z_{g1} = 1 | \mathbf{d}_g, \boldsymbol{\pi}, \boldsymbol{\psi}) \\ &= \frac{\pi_1 f_1(\mathbf{d}_g | \boldsymbol{\psi})}{\pi_1 f_1(\mathbf{d}_g | \boldsymbol{\psi}) + \pi_2 f_2(\mathbf{d}_g | \boldsymbol{\psi}) + \pi_3 f_3(\mathbf{d}_g | \boldsymbol{\psi})} \\ \zeta_{g2} &= \frac{\pi_2 f_2(\mathbf{d}_g | \boldsymbol{\psi})}{\pi_1 f_1(\mathbf{d}_g | \boldsymbol{\psi}) + \pi_2 f_2(\mathbf{d}_g | \boldsymbol{\psi}) + \pi_3 f_3(\mathbf{d}_g | \boldsymbol{\psi})} \\ \zeta_{g3} &= \frac{\pi_3 f_3(\mathbf{d}_g | \boldsymbol{\psi})}{\pi_1 f_1(\mathbf{d}_g | \boldsymbol{\psi}) + \pi_2 f_2(\mathbf{d}_g | \boldsymbol{\psi}) + \pi_3 f_3(\mathbf{d}_g | \boldsymbol{\psi})}\end{aligned}\tag{1}$$

E-step. Denote $Q^{(t)}(\boldsymbol{\pi}, \boldsymbol{\psi} | d, z^{(t)}, \boldsymbol{\pi}^{(t)})$ as the expected log complete-data likelihood function at t -th iteration of the EM algorithm, we have

$$\begin{aligned}Q^{(t)} &= E_z \left[l(\boldsymbol{\psi} | d, z, \boldsymbol{\pi}) | d, z^{(t)}, \boldsymbol{\pi}^{(t)} \right] \\ &= \sum_{g=1}^G ((\zeta_{g1}^{(t)} \log f_1(\mathbf{d}_g | \boldsymbol{\psi}) + \zeta_{g2}^{(t)} \log f_2(\mathbf{d}_g | \boldsymbol{\psi})) + \zeta_{g3}^{(t)} \log f_3(\mathbf{d}_g | \boldsymbol{\psi})) \\ &\quad + \sum_{g=1}^G (\zeta_{g1}^{(t)} \log \pi_1 + \zeta_{g2}^{(t)} \log \pi_2 + \zeta_{g3}^{(t)} \log \pi_3) \\ &\quad + \log \left(\frac{\Gamma(\sum_{c=1}^3 b_c)}{\prod_{c=1}^3 \Gamma(b_c)} \right) + \sum_{c=1}^3 (b_c - 1) \log \pi_c,\end{aligned}$$

where

$$\begin{aligned}\zeta_{gc}^{(t)} &= E(z_{gc} | \mathbf{d}_g, \boldsymbol{\pi}^{(t)}, \boldsymbol{\psi}^{(t)}) \\ &= \frac{\pi_c^{(t)} f_c(\mathbf{d}_g | \boldsymbol{\psi}^{(t)})}{\pi_1^{(t)} f_1(\mathbf{d}_g | \boldsymbol{\psi}^{(t)}) + \pi_2^{(t)} f_2(\mathbf{d}_g | \boldsymbol{\psi}^{(t)}) + \pi_3^{(t)} f_3(\mathbf{d}_g | \boldsymbol{\psi}^{(t)})}.\end{aligned}\tag{2}$$

M-step. Maximize $Q^{(t)}(\boldsymbol{\pi}, \boldsymbol{\psi} | d, z^{(t)}, \boldsymbol{\pi}^{(t)})$ to find the optimal values of $\boldsymbol{\pi}$ and $\boldsymbol{\psi}$, and use these optimal values as estimates for the parameters $\boldsymbol{\pi}$ and $\boldsymbol{\psi}$.

To maximize $Q^{(t)}(\boldsymbol{\pi}, \boldsymbol{\psi} | d, z^{(t)}, \boldsymbol{\pi}^{(t)})$, we use the “L-BFGS-B” method developed by Byrd et al. (1995)¹⁷, which utilizes the first partial derivatives of $Q^{(t)}(\boldsymbol{\pi}, \boldsymbol{\psi} | d, z^{(t)}, \boldsymbol{\pi}^{(t)})$ and allows box constraints, that is each variable can be given a lower and/or upper bound.

First Derivatives

For π_1, π_2, π_3 , we have:

$$\begin{aligned}\frac{\partial Q^{(t)}}{\partial \pi_1} &= \sum_g \left(\frac{\zeta_{g1}^{(t)}}{\pi_1} - \frac{\zeta_{g3}^{(t)}}{1 - \pi_1 - \pi_2} \right) + \frac{b_1 - 1}{\pi_1} - \frac{b_3 - 1}{1 - \pi_1 - \pi_2} \\ \frac{\partial Q^{(t)}}{\partial \pi_2} &= \sum_g \left(\frac{\zeta_{g2}^{(t)}}{\pi_2} - \frac{\zeta_{g3}^{(t)}}{1 - \pi_1 - \pi_2} \right) + \frac{b_2 - 1}{\pi_2} - \frac{b_3 - 1}{1 - \pi_1 - \pi_2}\end{aligned}$$

Since $\pi_1 + \pi_2 + \pi_3 = 1$, $\sum_g (\zeta_{g1}^{(t)} + \zeta_{g2}^{(t)} + \zeta_{g3}^{(t)}) = G$, and letting $b_1 = b_2 = b_3 = 2$, $\partial Q^{(t)} / \partial \pi_1 = \partial Q^{(t)} / \partial \pi_2 = 0$, we have:

$$\begin{aligned}\frac{\sum \zeta_{g1}^{(t)} + 1}{\pi_1} &= \frac{\sum \zeta_{g2}^{(t)} + 1}{\pi_2} = \frac{\sum \zeta_{g3}^{(t)} + 1}{\pi_3} \\ \pi_1^{(t+1)} &= \frac{\sum \zeta_{g1}^{(t)} + 1}{G + 3}, \pi_2^{(t+1)} = \frac{\sum \zeta_{g2}^{(t)} + 1}{G + 3}, \pi_3^{(t+1)} = \frac{\sum \zeta_{g3}^{(t)} + 1}{G + 3}\end{aligned}$$

For OE genes

For α_1, β_1, k_1 and η_1 , we have

$$\begin{aligned}\frac{\partial Q^{(t)}}{\partial \alpha_1} &= \sum_g \zeta_{g1}^{(t)} \frac{\partial \log f_1}{\partial \alpha_1} \\ \frac{\partial Q^{(t)}}{\partial \beta_1} &= \sum_g \zeta_{g1}^{(t)} \frac{\partial \log f_1}{\partial \beta_1} \\ \frac{\partial Q^{(t)}}{\partial k_1} &= \sum_g \zeta_{g1}^{(t)} \frac{\partial \log f_1}{\partial k_1} \\ \frac{\partial Q^{(t)}}{\partial \eta_1} &= \sum_g \zeta_{g1}^{(t)} \frac{\partial \log f_1}{\partial \eta_1}\end{aligned}$$

Now, we would like to take partial derivatives of $\log f_1$ with respect to α_1, β_1, k_1 and η_1 . And we have:

$$\begin{aligned}\log [f_1 (d_g | \psi)] &= \alpha_1 \log (\beta_1) + \log \left[\Gamma \left(\frac{n}{2} + \alpha_1 \right) \right] - \frac{n}{2} \log (k_1 + 1) - \left(\frac{n}{2} + \alpha_1 \right) \log (\beta_1 - M_1) \\ &\quad - \frac{n}{2} \log (2\pi) - \log [\Gamma(\alpha_1)]\end{aligned}$$

$$\frac{\partial \log (f_1)}{\partial \alpha_1} = \log (\beta_1) + \text{digamma} \left(\frac{n}{2} + \alpha_1 \right) - \log (\beta_1 - M_1) - \text{digamma} (\alpha_1)$$

$$\text{digamma}(x) = d \log (\Gamma(x)) / dx$$

$$\frac{\partial \log (f_1)}{\partial \beta_1} = \frac{\alpha_1}{\beta_1} - \frac{\left(\frac{n}{2} + \alpha_1 \right)}{(\beta_1 - M_1)}$$

$$\frac{\partial \log (f_1)}{\partial \eta_1} = \frac{\left(\frac{n}{2} + \alpha_1 \right)}{(\beta_1 - M_1)} \frac{\partial M_1}{\partial \eta_1}$$

Denote

$$\xi_1 = \exp (X^T \eta_1) = \begin{pmatrix} \exp (X_1^T \eta_1) \\ \vdots \\ \exp (X_n^T \eta_1) \end{pmatrix},$$

where

$$X_l = \begin{pmatrix} x_{l0} \\ x_{l1} \\ \vdots \\ x_{lp} \end{pmatrix}, \quad l = 1, \dots, n.$$

Then we can rewrite M_1 as

$$M_1 = -\frac{1}{2} \frac{1}{k_1 + 1} d_g^T d_g + \frac{1}{k_1 + 1} d_g^T \xi_1 - \frac{1}{2} \frac{1}{k_1 + 1} \xi_1^T \xi_1$$

$$\frac{\partial M_1}{\partial \eta_1} = \frac{1}{k_1 + 1} X \text{diag}(d_g) \xi_1 - \frac{1}{k_1 + 1} X \text{diag}(\xi_1) \xi_1$$

As for k_1 , we have

$$\frac{\partial \log f_1}{\partial k_1} = -\frac{n}{2} \frac{1}{k_1 + 1} + \frac{\frac{n}{2} + \alpha_1}{\beta_1 - M_1} \frac{\partial M_1}{\partial k_1}$$

$$\frac{\partial M_1}{\partial k_1} = \frac{1}{2} \frac{1}{(k_1 + 1)^2} d_g^T d_g - \frac{1}{(k_1 + 1)^2} d_g^T \xi_1 + \frac{1}{2} \frac{1}{(k_1 + 1)^2} \xi_1^T \xi_1$$

For UE gene probes

For α_2, β_2, k_2 and η_2 , we have

$$\frac{\partial Q^{(t)}}{\partial \alpha_2} = \sum_g \zeta_{g^2}^{(t)} \frac{\partial \log f_2}{\partial \alpha_2}$$

$$\frac{\partial Q^{(t)}}{\partial \beta_2} = \sum_g \zeta_{g^2}^{(t)} \frac{\partial \log f_2}{\partial \beta_2}$$

$$\frac{\partial Q^{(t)}}{\partial k_2} = \sum_g \zeta_{g^2}^{(t)} \frac{\partial \log f_2}{\partial k_2}$$

$$\frac{\partial Q^{(t)}}{\partial \eta_2} = \sum_g \zeta_{g^2}^{(t)} \frac{\partial \log f_2}{\partial \eta_2}$$

Take partial derivatives of $\log f_2$ with respect to α_2, β_2, k_2 and η_2 . And we have:

$$\begin{aligned} \log [f_2(d_g | \psi)] &= \alpha_2 \log(\beta_2) + \log \left[\Gamma \left(\frac{n}{2} + \alpha_2 \right) \right] - \frac{n}{2} \log(k_2 + 1) - \left(\frac{n}{2} + \alpha_2 \right) \log(\beta_2 - M_2) \\ &\quad - \frac{n}{2} \log(2\pi) - \log[\Gamma(\alpha_2)] \end{aligned}$$

$$\frac{\partial \log(f_2)}{\partial \alpha_2} = \log(\beta_2) + \text{digamma} \left(\frac{n}{2} + \alpha_2 \right) - \log(\beta_2 - M_2) - \text{digamma}(\alpha_2)$$

$$\frac{\partial \log(f_2)}{\partial \beta_2} = \frac{\alpha_2}{\beta_2} - \frac{\left(\frac{n}{2} + \alpha_2 \right)}{(\beta_2 - M_2)}$$

$$\frac{\partial \log(f_2)}{\partial \eta_2} = \frac{\left(\frac{n}{2} + \alpha_2 \right)}{(\beta_2 - M_2)} \frac{\partial M_2}{\partial \eta_2}$$

Denote

$$\xi_2 = \exp(X^T \eta_2) = \begin{pmatrix} \exp(X_1^T \eta_2) \\ \vdots \\ \exp(X_n^T \eta_2) \end{pmatrix},$$

where

$$X_l = \begin{pmatrix} x_{l0} \\ x_{l1} \\ \vdots \\ x_{lp} \end{pmatrix}, \quad l = 1, \dots, n.$$

Then we can rewrite M_2 as

$$M_2 = -\frac{1}{2} \frac{1}{k_2 + 1} d_g^T d_g - \frac{1}{k_2 + 1} d_g^T \xi_2 - \frac{1}{2} \frac{1}{k_2 + 1} \xi_2^T \xi_2$$

$$\frac{\partial M_2}{\partial \eta_2} = -\frac{1}{k_2 + 1} X \text{diag}(d_g) \xi_2 - \frac{1}{k_2 + 1} X \text{diag}(\xi_2) \xi_2$$

For k_2 , we have

$$\frac{\partial \log f_2}{\partial k_2} = -\frac{n}{2} \frac{1}{k_2 + 1} + \frac{\frac{n}{2} + \alpha_2}{\beta_2 - M_2} \frac{\partial M_2}{\partial k_2}$$

$$\frac{\partial M_2}{\partial k_2} = \frac{1}{2} \frac{1}{(k_2 + 1)^2} d_g^T d_g + \frac{1}{(k_2 + 1)^2} d_g^T \xi_2 + \frac{1}{2} \frac{1}{(k_2 + 1)^2} \xi_2^T \xi_2$$

For NE gene probes

For α_3, β_3 , we have

$$\begin{aligned} \frac{\partial Q^{(t)}}{\partial \alpha_3} &= \sum_g \zeta_{g^3}^{(t)} \frac{\partial \log f_3}{\partial \alpha_3} \\ \frac{\partial Q^{(t)}}{\partial \beta_3} &= \sum_g \zeta_{g^3}^{(t)} \frac{\partial \log f_3}{\partial \beta_3} \\ \log [f_3(d_g | \psi)] &= \alpha_3 \log(\beta_3) + \log \left[\Gamma \left(\frac{n}{2} + \alpha_3 \right) \right] - \left(\frac{n}{2} + \alpha_3 \right) \log(\beta_3 + M_3) \\ &\quad - \frac{n}{2} \log(2\pi) - \log[\Gamma(\alpha_3)] \end{aligned}$$

Here

$$M_3 = \frac{1}{2} d_g^T d_g + \frac{1}{2k_3} \eta_3^T \eta_3 - \frac{1}{2k_3} (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3)$$

Therefore, take the derivatives of $\log f_3$, we have

$$\frac{\partial \log f_3}{\partial \alpha_3} = \log \beta_3 + \text{digamma} \left(\frac{n}{2} + \alpha_3 \right) - \log(\beta_3 + M_3) - \text{digamma}(\alpha_3)$$

$$\frac{\partial \log f_3}{\partial \beta_3} = \frac{\alpha_3}{\beta_3} - \left(\frac{n}{2} + \alpha_3 \right) \frac{1}{\beta_3 + M_3}$$

$$\frac{\partial \log(f_3)}{\partial \eta_3} = \frac{\left(\frac{n}{2} + \alpha_3 \right)}{(\beta_3 + M_3)} \frac{\partial M_3}{\partial \eta_3}$$

For $\frac{\partial M_3}{\partial \eta_3}$, we consider the following two components,

$$\frac{\partial \eta_3^T \eta_3}{\partial \eta_3} = 2\eta_3$$

And we have (Matrix Cookbook¹⁸, equation(85)), given $\mathbf{W}$ is symmetric,

$$\frac{\partial}{\partial \mathbf{s}} (\mathbf{x} - \mathbf{s})^T \mathbf{W} (\mathbf{x} - \mathbf{s}) = 2\mathbf{W}(\mathbf{x} - \mathbf{s})$$

Then,

$$\frac{\partial}{\partial \eta_3} (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3) = 2(I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3)$$

$$\frac{\partial M_3}{\partial \eta_3} = \frac{1}{k_3} \eta_3 - \frac{1}{k_3} (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3)$$

For k_3 , we have

$$\begin{aligned}
 \frac{\partial \log(f_3)}{\partial k_3} &= \frac{\left(\frac{n}{2} + \alpha_3\right)}{(\beta_3 + M_3)} \frac{\partial M_3}{\partial k_3} \\
 &= d[(k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3)] \\
 &= (U d_g d k_3)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3) \\
 &\quad + (k_3 U d_g + \eta_3)^T d[(I_p + k_3 U U^T)^{-1}] (k_3 U d_g + \eta_3) \\
 &\quad + (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} (U d_g d k_3) \\
 &= 2(k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} U d_g d k_3 \\
 &\quad + (k_3 U d_g + \eta_3)^T [-(I_p + k_3 U U^T)^{-1} (U U^T d k_3) (I_p + k_3 U U^T)^{-1}] (k_3 U d_g + \eta_3) \\
 &= 2(k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} U d_g d k_3 \\
 &\quad + (k_3 U d_g + \eta_3)^T [-(I_p + k_3 U U^T)^{-1} U U^T (I_p + k_3 U U^T)^{-1}] (k_3 U d_g + \eta_3) d k_3 \\
 \\
 \frac{\partial}{\partial k_3} (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3) \\
 &= 2(U d_g)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3) \\
 &\quad - (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} U U^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3) \\
 \\
 \frac{\partial M_3}{\partial k_3} &= -\frac{1}{2k_3^2} \eta_3^T \eta_3 + \frac{1}{2k_3^2} (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3) \\
 &\quad - \frac{1}{k_3} (U d_g)^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3) \\
 &\quad + \frac{1}{2k_3} (k_3 U d_g + \eta_3)^T (I_p + k_3 U U^T)^{-1} U U^T (I_p + k_3 U U^T)^{-1} (k_3 U d_g + \eta_3)
 \end{aligned}$$

Initial parameter estimates

We first run *limma* to get false discovery rate (FDR) adjusted p-value of moderated t-test for each gene transcript. We then partition the gene transcripts to 3 clusters based on the FDR-adjusted p-values and signs of the test statistics: (1) gene transcripts with positive test statistics and FDR-adjusted p-values < 0.05 (denoted as OE); (2) gene transcripts with negative test statistics and FDR-adjusted p-values < 0.05 (denoted as UE); and (3) the remaining gene transcripts (denoted as NE). If numbers of transcripts in OE and/or UE are smaller than 5, we then use 100 transcripts having the smallest un-adjusted p-values to find initial OE and UE transcripts. We next estimate the cluster mixture proportions (π_1, π_2, π_3) by the 3 cluster sizes dividing the total number of gene transcripts.

Within each cluster, we use moment estimator to get the initial estimates of model parameters. In our case, we use sample median (med) and sample median absolute deviation (mad) to get robust estimates of mean and standard deviation, based on which we get the initial estimate of ψ .

OE gene transcripts

$$\begin{aligned}
 \hat{\tau}_g^{(0)} &= \frac{1}{(\text{mad}_l(d_{gl}))^2} \\
 \hat{\alpha}_1^{(0)} &= \left(\frac{\text{med}_g(\hat{\tau}_g^{(0)})}{\text{mad}_g(\hat{\tau}_g^{(0)})} \right)^2 \\
 \hat{\beta}_1^{(0)} &= \frac{\text{med}_g(\hat{\tau}_g^{(0)})}{(\text{mad}_g(\hat{\tau}_g^{(0)}))^2} \\
 \hat{\eta}_1^{(0)} &= \left(X_{(p+1) \times n} X_{n \times (p+1)}^T \right)^{-1} X \log(|\hat{\mu}|) \\
 \hat{k}_1 &= \bar{\tau}_g^{(0)} s^2.
 \end{aligned}$$

where

$$s^2 = \frac{1}{n} \sum_{l=1}^n [\hat{\mu}_l - \tilde{\mu}]^2, \tilde{\mu} = \frac{1}{n} \sum_{l=1}^n \hat{\mu}_l, \tilde{\tau}_g^{(0)} = \frac{1}{G_1} \sum_{g=1}^{G_1} \hat{\tau}_g^{(0)}.$$

UE gene transcripts

$$\begin{aligned} \hat{\tau}_g^{(0)} &= \frac{1}{(\text{mad}_l(d_{gl}))^2} \\ \hat{\alpha}_2^{(0)} &= \left(\frac{\text{med}_g(\hat{\tau}_g^{(0)})}{\text{mad}_g(\hat{\tau}_g^{(0)})} \right)^2 \\ \hat{\beta}_2^{(0)} &= \frac{\text{med}_g(\hat{\tau}_g^{(0)})}{(\text{mad}_g(\hat{\tau}_g^{(0)}))^2} \\ \hat{\eta}_2^{(0)} &= \left(X_{(p+1) \times n} X_{n \times (p+1)}^T \right)^{-1} X \log(|-\hat{\mu}|) \\ \hat{k}_2 &= \tilde{\tau}_g^{(0)} s^2. \end{aligned}$$

where

$$s^2 = \frac{1}{n} \sum_{l=1}^n [\hat{\mu}_l - \tilde{\mu}]^2, \tilde{\mu} = \frac{1}{n} \sum_{l=1}^n \hat{\mu}_l, \tilde{\tau}_g^{(0)} = \frac{1}{G_2} \sum_{g=1}^{G_2} \hat{\tau}_g^{(0)}.$$

Here, we use $|\hat{\mu}|$ to make sure it is positive.

NE gene transcripts

The initial values are

$$\begin{aligned} \hat{\tau}_g^{(0)} &= \frac{1}{(\text{mad}_l(d_{gl}))^2} \\ \hat{\alpha}_3^{(0)} &= \left(\frac{\text{med}_g(\hat{\tau}_g^{(0)})}{\text{mad}_g(\hat{\tau}_g^{(0)})} \right)^2 \\ \hat{\beta}_3^{(0)} &= \frac{\text{med}_g(\hat{\tau}_g^{(0)})}{(\text{med}_g(\hat{\tau}_g^{(0)}))^2} \end{aligned}$$

As for η_3 , we first need to get $\hat{\theta}_g$. With linear regression, we have

$$\hat{\theta}_g = (U_{p \times n} U_{p \times n}^T)^{-1} U_{p \times n} d_g$$

Then we can get

$$\hat{\eta}_3^{(0)} = \text{med}_g(\hat{\theta}_g)$$

And

$$\hat{k}_3 = \tilde{\tau}_g^{(0)} s^2$$

where

$$s^2 = \frac{1}{G_3} \sum_{l=1}^{G_3} (\hat{\theta}_g - \bar{\theta})^T (\hat{\theta}_g - \bar{\theta}), \bar{\theta} = \frac{1}{G_3} \sum_{g=1}^{G_3} (\hat{\theta}_g), \tilde{\tau}_g^{(0)} = \frac{1}{G_3} \sum_{g=1}^{G_3} \hat{\tau}_g^{(0)}.$$

Bounds for hyper-parameters

Although we can re-parameterize hyper-parameters to make sure $\alpha_c > 0$, $\beta_c > 0$, $0 < k_c < 1$, $c = 1, 2, 3$, and to transform constrained optimization problem to un-constrained optimization problem, the results are usually not good due to non-linear optimization in the M-step of the EM algorithm. To avoid un-expected large values of estimated α_c , β_c , and/or k_c in numerical optimization, we set constraints $0.001 < \alpha_c < 6$, $0.001 < \beta_c < 6$, $0.001 < k_c < 0.9999$.

If initial estimates $\hat{\alpha}_c$, $\hat{\beta}_c$, or $\hat{k}_c$ are outside their ranges, then we randomly choose a number between the range as the initial parameter estimates.

In the M-step of the EM iterations, we also require each element of η_c , $c = 1, 2, 3$, be within the interval $[-10, 10]$.

If at least one element of $\psi^{(t)}$ are outside their ranges in the t -th iteration of the EM algorithm, we then set final estimate of ψ as $\psi^{(t-1)}$, and then only update $\pi_c^{(t-1)}$, $c = 1, 2, 3$, until convergence criterion satisfied.

The M-step in the EM algorithm is to maximize Q function, which is a non-linear function of model parameters π_1 , π_2 , π_3 , ψ . It is difficult to optimize non-linear functions. The above constraints and procedure produced good results in the real and simulation data analyses in this article.

Information for differentially expressed transcripts for GSE24742

The information about the 6 UE transcripts detected by *limma* is shown in Table 1.

Table 1. Transcript IDs, gene symbols, chromosome numbers of the 6 UE transcripts detected by *limma*

ID	gene	chr
1560920_s_at	LOC102725382	9
202067_s_at	LDLR	19
207306_at	TCF15	20
219249_s_at	FKBP10	17
222419_x_at	UBE2H	7
224507_s_at	MGC12916	17

The information about the 9 OE transcripts detected by *eLNNpairedCov* is shown in Table 2.

Table 2. Transcript IDs, gene symbols, chromosome numbers of the 9 OE transcripts detected by *eLNNpairedCov*

ID	gene	chr
1552808_at	ADAMTSL1	9
1570121_at	ZNF365	10
222223_s_at	IL36RN	2
231595_at	MRVI1-AS1	11
232186_at	FITM2	20
233466_at	LOC101928729	17
234138_at	LOC105378749	1
238249_at	LOC101927499	22
244557_at	LINC00610	11

The information about the 83 UE transcripts detected by *eLNNpairedCov* is shown in Table 3.

Table 3. Transcript IDs, gene symbols, chromosome numbers of the 83 UE transcripts detected by *eLNNpairedCov*

ID	gene	chr	ID	gene	chr
1552396_at	WFDC6	20	214476_at	TFF2	21
1554256_a_at	PCNX2	1	214485_at	ODF1	8
1554259_at	GPSM1	9	216355_at	PCDHB17P	5
1554468_s_at	FBF1	17	216645_at	TCF3	19
1555308_at	CARD14	17	217397_at	YME1L1	10
1555377_at	OR4D2	17	217515_s_at	CACNA1S	1
1555704_at	CMTM3	16	217603_at	ATP6V0A2	12
1557866_at	CFAP157	9	219249_s_at	FKBP10	17
1558177_at	TMEM229B	14	219490_s_at	DCLRE1B	1
1558619_at	SNORA17B	9	219601_s_at	C10orf12	10
1560100_at	DLX1	2	220382_s_at	ARHGAP28	18
1560920_s_at	LOC102725382	9	220555_s_at	PDZD7	10
1562098_at	UHRF1BP1L	12	221165_s_at	IL22	12
1562623_at	LOC146513	16	221805_at	NEFL	8
1562631_at	TEX26-AS1	13	221835_at	DTX3	12
1564413_at	LINC01356	1	222419_x_at	UBE2H	7
1564444_at	LOC100130264	20	223488_s_at	GNB4	3
1564963_x_at	ZNF92	7	223761_at	FGF19	11
1569206_at	TCP11L2	12	224088_at	NMUR2	5
1569741_at	LINC01477	18	224507_s_at	MGC12916	17
200796_s_at	MCL1	1	226882_x_at	WDR4	21
201208_s_at	TNFAIP1	17	228068_at	GOLGA7B	10
201490_s_at	PIPF	10	228320_x_at	BICDL1	12
202067_s_at	LDLR	19	229045_at	SNX20	16
202827_s_at	MMP14	14	230895_at	HAPLN1	5
205695_at	SDS	12	231135_at	LOC151174	2
206619_at	DKK4	8	231732_at	SMPD3	16
206938_at	SRD5A2	2	234393_at	HDAC9	7
207116_s_at	GAPDHS	19	236730_at	GIPC3	19
207306_at	TCF15	20	237323_at	HKDC1	10
207528_s_at	SLC7A11	4	237765_at	SLC25A47	14
208071_s_at	LAIR1	19	238487_at	GNL1	6
208272_at	RANBP3	19	239132_at	NOS1	12
208889_s_at	NCOR2	12	239998_at	C10orf53	10
209261_s_at	NR2F6	19	241950_at	WWC1	5
209734_at	NCKAP1L	12	242150_at	CLSPN	1
209776_s_at	SLC19A1	21	242917_at	RASGEF1A	10
209842_at	SOX10	22	243426_at	LINC00667	18
210450_at	IGHV5-78	14	243506_at	AKAIN1	18
210955_at	CASP10	2	244437_at	LOC101927340	12
213250_at	CCDC85B	11	76897_s_at	FKBP15	9
214336_s_at	COPA	1			

OE (eLNNpairedCov)

UE (eLNNpairedCov)

log2(after) - log2(before)

UE (limma)

$\log_2(\text{after}) - \log_2(\text{before})$

Plot of accuracy by method

Plot of difference of accuracy from limma by method

method

- eLNNpaired

▲ eLNNpairedCov.max.w

method

Plot of difference of FNR from limma by method

Plot of difference of FPR from limma by method

Plot of difference of iteration number from limma by method

Plot of difference of Jaccard index from limma by method

Plot of difference of user time from limma by method

Plot of FNR by method

The scatter plot displays the relationship between 'number' (x-axis) and 'count' (y-axis). The x-axis ranges from 0 to 100, and the y-axis ranges from 0 to 100. A red horizontal line is drawn at count = 10. Data points are colored by 'category': red (category 1), green (category 2), and blue (category 3). Red points are clustered at low 'number' values (0-20) with 'count' values mostly near 10. Green points are clustered at higher 'number' values (40-60) with 'count' values mostly near 10, but there are two outliers with 'count' values around 40 and 50. Blue points are clustered at high 'number' values (80-100) with 'count' values mostly near 10.

- limma
- ▲ eLNNpaired
- eLNNpairedCov.max.w

Plot of iteration number by method

Plot of Jaccard index by method

Plot of user time (second) by method

Plot of accuracy by method

Plot of difference of accuracy from limma by method

Plot of difference of FNR from limma by method

Plot of difference of FPR from limma by method

Plot of difference of iteration number from limma by method

Plot of difference of Jaccard index from limma by method

Plot of difference of user time from limma by method

Plot of FNR by method

Plot of FPR by method

Plot of iteration number by method

Plot of Jaccard index by method

Plot of user time (second) by method

Histogram of sd