THE UNIVERSITY OF CHICAGO

EMPIRICAL BAYES METHODS FOR COUNT DATA

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF STATISTICS

BY
DONGYUE XIE

CHICAGO, ILLINOIS
AUGUST 2023

To my family

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I deeply value the time I have spent at the University of Chicago. Being accepted into the Ph.D. program in the Department of Statistics has been an incredible opportunity, and has played a significant role in shaping my research. Completing this thesis and other doctoral work involved support of many people I have met during my study here. Their support made this all possible.

First and foremost, I am extremely grateful to my dissertation advisor, Matthew Stephens FRS, for bringing me into his lab and for his invaluable guidance, exceptional mentorship, as well as endless support over the years. His insights into statistical methods and scientific problems are profound, and have inspired and motivated me to develop statistical methods to explore science. I am truly honored to do research with Matthew - an exceptional statistician, geneticist and scientist.

I extend my deep gratitude to Dan Nicolae and Jingshu Wang, for their enlightening recommendations, meaningful discussions, and constructive feedback on my research. I thoroughly enjoyed working with Dan and Jingshu and my research work with them has also had tremendous inspirations on the research in this dissertation. Their invaluable contributions have significantly enhanced the quality of this dissertation.

I would like to thank all faculty members in the Departments of Statistics, for their continuous supports and endeavors.

I would also like to thank my friends and past and present peers in the Department of Statistics, CAM, and the Stephens Lab for their encouragement, support and for the delightful conversations and valuable discussions: Wanrong Zhu, Daniel Xiang, Solomon Quinn, Peter Carbonetto, Zihao Wang, Hai Tran Bach, Yi Wang, Yuxin Zou, Yunqi Yang, Karl Tayeb, Saikat Banerjee, Carlos F. Buen Abad Najar, Andrew Goldstein, Kaixuan (Kevin) Luo,

William Denault, Kushal K. Dey, Gao Wang, Yusha Liu, Hussein Al-Asadi, Joonsuk Kang, Nathan Lapierre, Wesley Crouse, Jean Morrison, Fabio Morgante, Abhishek Sarkar, Jason Willwerscheid, Yuguan Wang, Yi Wei, Chih-Hsuan Wu, Xiao (Annie) Xie, Jinwen Yang, Lijia Zhou, Zhen Dai and many others.

I would love to thank my parents, Chuanhong Xie and Lingdi Wang for their love and support. They have always been absolutely supportive throughout all my education journey. Thank you for always being there throughout this process.

ABSTRACT

High-throughput sequencing (HTS) techniques such as RNA-seq, ChIP-seq and ATAC-seq have enabled researchers to investigate complex biological processes in unprecedented detail. One common feature of HTS data is that they often consist of counts. For example, in RNA-seq, the counts typically represent the number of times a RNA molecule has been sequenced and are a proxy for the expression level. Recently, the advent of single-cell sequencing techniques such as scRNA-seq and scATAC-seq has unveiled the transcriptome at cell-level resolution. However, the single-cell count data are sparse and come with high levels of technical noise. With the emergence of large, sparse and noisy sequencing data, there is a need for rigorous statistical methods that can accurately model these counts.

On the other hand, due to the complex structure of the sequencing data exhibited, the statistical methods developed for the data should be flexible enough to incorporate different assumptions and structural information. For instance, matrix factorization has been extensively employed to uncover the latent structure of gene expression across a variety of cell types. The incorporation of sparsity assumptions into these latent structures has been shown to yield a more parsimonious representation and enhance the interpretability of results. Consequently, it would be beneficial to integrate sparsity assumptions when modeling the structure of sequencing data.

In this thesis, we focus on developing flexible empirical Bayes (EB) methods for statistical modeling and inference in the field of genomics. We first explore EB Poisson mean models as a fundamental component for developing sophisticated models and as a simple problem for evaluating different approaches. Then we study EB smoothing methods that can account for extra variation or over-dispersion in sequencing data, and apply the methods to visualize gene expression patterns along the genome. We further introduce a general variational inference method for non-Gaussian data, and develop an EB Poisson matrix factorization method, with

applications to single cell RNA sequencing data. Finally, we extend Poisson non-negative matrix factorization methodologies to accommodate spatially-structured or sparse factors and loadings.

CHAPTER 1

INTRODUCTION

In recent years, high-throughput sequencing (HTS) techniques such as RNA-seq, ChIP-seq and ATAC-seq have revolutionized the field of genomics and many other areas of biology. By producing massive amounts of data, HTS has enabled researchers to investigate complex biological processes in unprecedented detail. One common feature of HTS data is that they often consist of counts. The counts represent the number of times a particular event is observed in the sample. For example the counts in RNA-seq typically represents the number of times a RNA molecule, usually a transcript or a gene, has been sequenced in a sample. These counts are a proxy for the abundance or expression level of the corresponding RNA molecules.

As for the statistical analysis of sequencing count data, some methods model the counts directly while the others transform them so that one can apply Gaussian distribution-based methods. For example, in differential expression analysis for bulk RNA-seq data, methods such as limma-voom (Ritchie et al. [2015], Law et al. [2014]) have focused on transforming the count data then applying Gaussian linear regression. Methods directly modeling count have also been developed such as DESeq2 (Love et al. [2014]) and edgeR (Robinson et al. [2010]). Bulk RNA-seq data typically exhibit large counts, and from a statistical perspective, properly transforming these large counts can result in a reasonably valid Gaussian distribution representation. Recently, the advent of single-cell sequencing techniques such as scRNA-seq and scATAC-seq have enabled researchers to investigate the transcriptome at cell-level resolution. The single-cell count data are sparse and have high levels of technical noise. Sarkar and Stephens [2021] argued that the Poisson measurement model is preferred from both theoretical and practical point of view. A number of methods have been developed for directly modelling single-cell sequencing data using count models, typically assuming Poisson

or negative binomial distributions (Huang et al. [2018], Vallejos et al. [2015], Wang et al. [2018], Townes et al. [2019], Eraslan et al. [2019], Lopez et al. [2018], Levitin et al. [2019], Sun et al. [2019], Risso et al. [2018]).

On the other hand, due to the complex structure of the sequencing data exhibited, the statistical methods developed for the data should be flexible enough to incorporate different assumptions and structural information. For instance, sparse matrix factorization (Wang and Stephens [2021], Witten et al. [2009]) has demonstrated its ability to yield a more parsimonious representation and enhance the interpretability of results. Consequently, we could integrate sparsity assumptions when modelling the structure of sequencing data. Another example is when modelling RNA-seq data along the genome or ChIP-seq data, the underlying signal is spatially-structured, which should be considered in the modeling process.

In this thesis, we focus on developing flexible empirical Bayes (EB) methods designed for count sequencing data analysis. In Chapter 2, we start with an exploration of EB Poisson mean models, which are analogous to empirical Bayes normal mean problems (EBNM, Willwerscheid and Stephens [2021]). The EB Poisson mean model serves as a fundamental component for the development of more sophisticated models. Chapter 3 gives a comprehensive examination of various EB smoothing methods for count data, allowing for extra variation or over-dispersion. Subsequently, these methods are applied to de-noise and visualize gene expression patterns along the genome. In chapter 4, we introduce a novel variational inference method for non-Gaussian data, and develop an EB Poisson matrix factorization method (with log link) for scRNA-seq data. In chapter 5, we revisit EB matrix factorization (Wang and Stephens [2021]), and EB Poisson matrix factorization (with identity link) models. Furthermore, we extend these two methodologies to enable factors and/or loadings to be spatially-structured.

In the rest of this chapter, we review the basic concepts used in this thesis - empirical Bayes

and variational inference. Then we introduce the variational empirical Bayes method and show how the two concepts combine to give a flexible framework for statistical modelling and inference.

1.1 Empirical Bayes

In Bayesian data analysis, a typical model involves an observation model and a prior distribution on the unobserved quantities. Denote $\boldsymbol{y}$ as the observations and $\boldsymbol{\mu}$ as the unobserved quantities, the joint distribution is

$$p(\boldsymbol{y}, \boldsymbol{\mu}) = p(\boldsymbol{y}|\boldsymbol{\mu})g(\boldsymbol{\mu}), \tag{1.1}$$

where $g(\boldsymbol{\mu})$ denotes the prior distribution on $\boldsymbol{\mu}$.

Then the posterior distribution of $\boldsymbol{\mu}$ is given as

$$p(\boldsymbol{\mu}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\mu})g(\boldsymbol{\mu})}{p(\boldsymbol{y})}, \tag{1.2}$$

where $p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{\mu})g(\boldsymbol{\mu})d\boldsymbol{\mu}$. Note that in the posterior calculation we have assumed the prior distribution $g(\cdot)$ is fully specified.

Empirical Bayes allows for the estimation of unknown parameters in the prior from the observed data. An empirical Bayes approach typically has two steps. It first estimates the prior $g(\cdot)$ by maximizing the log marginal likelihood

$$\hat{g} = \arg\max_{g \in \mathcal{G}} \int p(\boldsymbol{y}|\boldsymbol{\mu})g(\boldsymbol{\mu})d\boldsymbol{\mu}, \tag{1.3}$$

where $\mathcal{G}$ is some specified family of prior distributions. The calculation of posterior distribution is conditional on the estimated prior $\hat{g}$. For a high level review of empirical Bayes and

its applications, see van de Wiel et al. [2019].

1.2 Variational empirical Bayes

Variational Inference (VI) is a powerful method used in approximate Bayesian inference. It is particularly useful when dealing with complex models where exact posterior inference is computationally expensive or analytically intractable. VI turns the inference problem into an optimization problem by approximating the true posterior with a simpler, more tractable distribution. A canonical review of VI is given by Blei et al. [2017].

The need for VI often arises when calculating the true posterior distribution often involves integrating over high-dimensional spaces, which can be difficult or impossible to solve analytically. Moreover, sampling-based methods like Markov Chain Monte Carlo (MCMC) can be computationally expensive and slow to converge, especially in large-scale problems such as matrix factorization for single cell RNA sequencing data. The VI finds

$$q^*(\boldsymbol{\mu}) = \underset{q \in \mathcal{Q}}{\arg\min}\, D_{KL}(q(\boldsymbol{\mu}) \| p(\boldsymbol{\mu}|\boldsymbol{y})), \tag{1.4}$$

where $\mathcal{Q}$ is a family of approximate densities, and D_{KL} is the Kullback-Leibler (KL) divergence. The family $\mathcal{Q}$ determines the complexity of this optimization and is typically selected to make computation easy while (ideally) being sufficiently adaptable to approximate the posterior $p(\boldsymbol{\mu}|\boldsymbol{y})$ closely.

Since the true posterior $p(\boldsymbol{\mu}|\boldsymbol{y})$ is unknown, we cannot directly minimize the KL divergence (1.4). Instead, we maximize the following Evidence Lower Bound (ELBO), which is a lower

bound on the log marginal likelihood $\log p(\boldsymbol{y}; g)$:

$$\begin{aligned} F(q; g, \boldsymbol{y}) &= \log p(\boldsymbol{y}; g) - D_{KL}(q(\boldsymbol{\mu}) \| p(\boldsymbol{\mu}|\boldsymbol{y})), \\ &= \mathbb{E}_{q(\boldsymbol{\mu})}(\log p(\boldsymbol{y}, \boldsymbol{\mu}) - \log q(\boldsymbol{\mu})). \end{aligned} \tag{1.5}$$

F a lower bound of the evidence $[\log p(\boldsymbol{y}; g)]$ because KL divergence is non-negative. Minimizing the KL divergence in (1.4) over q is equivalent to maximizing the ELBO $F(q; g, \boldsymbol{y})$ with respect to q.

Mean-field variational inference is the most prevalent method employed in variational inference, in which the posterior of $\boldsymbol{\mu}$ is assumed to factorize over each element. A popular optimization method for maximizing the ELBO in mean-field variational inference is coordinate ascent variational inference (Jordan et al. [1999]). This method iteratively optimizes the posterior density for each latent variable while keeping the others fixed. Although straightforward to implement, this approach can be inefficient for large-scale models. Alternative optimization methods have been explored in the literature, such as stochastic variational inference (Hoffman et al. [2013]), black box variational inference (Ranganath et al. [2014]), and Markov chain variational inference (Salimans et al. [2015]).

Variational empirical Bayes (VEB) combines variational inference and empirical Bayes in a single optimization problem, expressed as

$$q^*(\boldsymbol{\mu}), \hat{g} = \underset{q \in \mathcal{Q}, g \in \mathcal{G}}{\arg\max}\, F(q, g; \boldsymbol{y}). \tag{1.6}$$

In contrast to classical empirical Bayes, this method learns the prior $\hat{g}$ under the approximated posterior distribution q. Differing from variational inference, the prior g is not fully specified and is instead estimated from the data. We note that if the family $\mathcal{Q}$ includes all possible densities, then VEB is the same as regular EB, as shown in Lemma 1.2.1.

Lemma 1.2.1. *If the variational family $\mathcal{Q}$ includes all possible densities, then the optimal $q^*(\boldsymbol{\mu}), \hat{g}$ obtained from the VEB optimization problem* (1.6) *are*

$$\hat{g} = \arg\max_{g \in \mathcal{G}} \int p(\boldsymbol{y}|\boldsymbol{\mu}) g(\boldsymbol{\mu}) d\boldsymbol{\mu},$$
$$q^*(\boldsymbol{\mu}) = \frac{p(\boldsymbol{y}|\boldsymbol{\mu})\hat{g}(\boldsymbol{\mu})}{\int p(\boldsymbol{y}|\boldsymbol{\mu})\hat{g}(\boldsymbol{\mu}) d\boldsymbol{\mu}},$$

which are the same as the ones from the regular EB procedure.

Proof. For any given g, the optimal q is $p(\boldsymbol{\mu}|\boldsymbol{y}; g)$, the exact posterior. This is because when $q(\boldsymbol{\mu}) = p(\boldsymbol{\mu}|\boldsymbol{y}, g)$, the ELBO $F(q, g; \boldsymbol{y})$ is the same as the evidence, i.e. $F(q, g; \boldsymbol{y}) = \log p(\boldsymbol{y}; g)$. Therefore the profiled objective function for g is the evidence, and the optimal g is $\hat{g} = \arg\max_{g \in \mathcal{G}} \log p(\boldsymbol{y}; g)$, which is exactly the regular EB estimate. And the optimal posterior is $q^*(\boldsymbol{\mu}) = p(\boldsymbol{\mu}|\boldsymbol{y}, \hat{g})$, which is also the same as the one from regular EB procedure. □

CHAPTER 2

THE EMPIRICAL BAYES POISSON MEAN PROBLEM WITH A VEB SOLUTION

2.1 Introduction

The empirical Bayes normal means problem has been extensively studied and applied to various areas, including shrinkage estimation for mean parameter (Willwerscheid and Stephens [2021]), nonparametric regression or denoising problem (Xing et al. [2021]), and large-scale multiple testing (Stephens [2017]). It also serves as a fundamental component for more complex models, such as regression (Kim et al. [2022]) and matrix factorization (Wang and Stephens [2021]). However, in real-world applications, data is frequently non-normally distributed. For instance, RNA sequencing data consists of counts, and DNA methylation data (Lea et al. [2015]) is typically analyzed as binomial data. As a building block towards developing models for non-Gaussian data, in this chapter, we explore the empirical Bayes means problem for these non-Gaussian distributions.

Consider the EB Poisson mean model

$$\begin{aligned} y_j|\mu_j &\sim \text{Poisson}(s_j \times h(\mu_j)), \\ \boldsymbol{\mu} &\sim g(\cdot), \end{aligned} \tag{2.1}$$

for $j = 1, 2, ..., n$, where $h^{-1}(\cdot)$ is a link function, g is a prior distribution to be estimated and s_j is a known positive scaling scalar. Commonly used link functions are the log and identity links. In this chapter, we consider different variational approaches for estimating the prior $g(\cdot)$ and performing inference on the posterior of $\boldsymbol{\mu}$, denoted as $q_{\boldsymbol{\mu}}(\cdot)$. We refer to this procedure as the "variational empirical Bayes Poisson mean" (VEBPM) procedure. This

procedure defines a mapping from $(\boldsymbol{y}, \boldsymbol{s})$ to $(\hat{g}, q)$, and we denote such mapping as

$$(\hat{g}, q) = \text{VEBPM}(\boldsymbol{y}, \boldsymbol{s}). \tag{2.2}$$

Specifically, we focus mainly on the canonical link function of the Poisson distribution, which is the log link, $h(\mu) = \exp(\mu)$. The log-likelihood is

$$l(\boldsymbol{\mu}) = \log p(\boldsymbol{y}|\boldsymbol{\mu}) = \sum_j y_j \mu_j - s_j \exp(\mu_j) + \text{const}\,. \tag{2.3}$$

Denote the approximate posterior of μ_j as q_{μ_j}, the posterior mean as $\bar{\mu}_j = \mathbb{E}_{q_{\mu_j}}(\mu_j)$, and posterior variance as $V_{\mu_j} = \text{Var}_{q_{\mu_j}}(\mu_j)$. The objective function we are maximizing is the evidence lower bound (ELBO) defined as,

$$\begin{aligned} F(q, g) &= \sum_j \left(\mathbb{E} \log p(y_j, \mu_j) - \mathbb{E} \log q_{\mu_j} \right), \\ &= \sum_j \left(\mathbb{E} \log p(y_j|\mu_j) + \mathbb{E} \log \frac{g(\mu_j)}{q_{\mu_j}} \right), \end{aligned} \tag{2.4}$$

where the expectation is over q_{μ_j}.

In the remaining section of the chapter, we describe and compare several VEB methods for solving the EB Poisson mean problem (2.1). Though we study the methods for the simple EB Poisson mean model, more broadly, we are interested in how the methods can be generalized to more complicated statistical models such as regression and matrix factorization.

We first discuss a direct variational inference method by assuming a specific parametric form for q_{μ_j}. A similar approach has been considered in Arridge et al. [2018] for the Poisson regression problem (log-link), where both the prior and posterior on the regression coefficients

are multivariate normal distributions. As we will see, this approach is not easily extended to more complex models. For each model, it requires separate algorithm derivations and non-trivial software development.

Next, we will study methods for VEBPM that could utilize existing empirical Bayes normal means (EBNM) methods. These methods typically require approximations of the ELBO but can be easily extended to other more complicated models, such as regression and matrix factorization. This is because they reduce the task of solving a Poisson problem to iteratively solving a normal problem.

Finally, we propose a novel penalty-based formulation of VEBPM, extending the normal mean penalty studied in Kim et al. [2022], and we show how to solve the optimization problem using existing solvers. The method can be generalized to other distributions, such as the Binomial distribution, and can also be extended to regression as well as matrix factorization problems.

2.2 Parametric VEBPM

In this section, we consider a parametric variational inference method for the VEBPM problem. We start with a simple case where both prior and posterior are Gaussian distributions. Then we study a more flexible model where both prior and posterior are mixture of Gaussians.

2.2.1 Gaussian prior and posterior

Consider the EB Poisson mean model (2.1), where the prior on each μ_j is

$$g(\mu_j) = N(\mu_j; \theta, \sigma^2), \tag{2.5}$$

and the approximate posterior is

$$q(\boldsymbol{\mu}) = \prod_j q_{\mu_j} = \prod_j N(\mu_j; \bar{\mu}_j, V_{\mu_j}). \tag{2.6}$$

The ELBO is $\sum_j F(\bar{\mu}_j, V_{\mu_j})$, where

$$F(\bar{\mu}, V_\mu) = y\bar{\mu} - se^{\bar{\mu}+V_\mu} - \frac{\bar{\mu}^2 + V_\mu - 2\bar{\mu}\theta}{2\sigma^2} + \frac{1}{2}\log V_\mu + \text{const.} \tag{2.7}$$

Following the VEB approach, a coordinate ascent variational inference (CAVI) algorithm iterates among the following steps until convergence:

(a). Update $\bar{\mu}_j, V_{\mu_j}$ as $\arg\max_{\bar{\mu}_j, V_{\mu_j}} F(\bar{\mu}_j, V_{\mu_j})$, for $j = 1, 2, ..., n$;

(b). Update $\sigma^2 \leftarrow \sum_j (\bar{\mu}_j^2 + V_{\mu_j} - 2\bar{\mu}_j\theta + \theta^2)/n$;

(c). Update $\theta \leftarrow \sum_j \bar{\mu}_j/n$.

2.2.2 Mixture of Gaussians prior and posterior

We assume the prior on each μ_j as

$$g(\mu_j) = \sum_{k=0}^{K} \pi_k N(\mu_j; \theta, \sigma_k^2), \tag{2.8}$$

where $\sigma_0^2 = 0$, and $\sigma_1^2, ..., \sigma_K^2$ are a large and dense grid of fixed positive values spanning a range from very small to very large. This is a slightly modified version of the adaptive shrinkage (ash, Stephens [2017]) prior that the prior mode is θ instead of being fixed at 0. We introduce latent variables z_{jk} such that $p(z_{jk} = 1) = \pi_k$ and $\mu_j | z_{jk} = 1 \sim N(\theta, \sigma_k^2)$. We restrict the posterior distribution to be a mixture of Gaussians and the posteriors factorize

Algorithm 1 CAVI algorithm for VEBPM with ash prior and Gaussian mixture posterior

Input: $\boldsymbol{y}, s$
Init: $\theta, \boldsymbol{\pi}$.
repeat
  Update $\bar{\mu}_{jk}, V_{\mu_{jk}}$ as $\arg\max_{\bar{\mu}_{jk}, V_{\mu_{jk}}} F(\bar{\mu}_{jk}, V_{\mu_{jk}})$, where F is defined as (2.7).
  Update $\phi_{jk} \propto \exp\left(\Delta_{jk}\right)$.
  Update $\pi_k \leftarrow \frac{\sum_j \phi_{jk}}{n}$.
  Update $\theta \leftarrow \frac{\sum_{j,k} \frac{\phi_{jk}}{\sigma_k^2} \bar{\mu}_{jk}}{\sum_{j,k} \frac{\phi_{jk}}{\sigma_k^2}}$.
until Converged
Output: $\hat{\theta}, \hat{\pi}$, and $q_{\boldsymbol{z}}, q_{\boldsymbol{\mu}|\boldsymbol{z}}$.

as

$$q = \prod_{j,k} (q(\mu_j | z_{jk} = 1)\phi_{jk})^{z_{jk}}, \tag{2.9}$$

where $q(\mu_j | z_{jk} = 1) = N(\mu_j; \bar{\mu}_{jk}, V_{\mu_{jk}})$ and $\phi_{jk} = q(z_{jk} = 1)$.

The CAVI algorithm is given in Algorithm 1. The iterations are stopped when the increase of ELBO (A.4) is smaller than a pre-specified tolerance. The initial value of θ is $\log(\sum x_j / \sum_j s_j)$ and the initial value of π_k is $1/K$.

2.3 Solve VEBPM problem via EBNM

In this section, we consider methods that leverage the existing empirical Bayes normal means (EBNM) problem to solve the EB Poisson mean problem. The EBNM problem has been extensively studied and a variety of prior classes are available, as well as a unified `R` package `ebnm` (Willwerscheid and Stephens [2021]). By utilizing EBNM, we can avoid developing separate algorithms for each prior class and take advantage of the fast and exact solutions.

The EBNM problem can be formulated as a mapping from observations to estimated prior

and posterior. We use the notation introduced in Wang and Stephens [2021] as

$$(\hat{g}, q) = \text{EBNM}(\boldsymbol{y}, \boldsymbol{s}), \tag{2.10}$$

where $\boldsymbol{y}$ is the observation vector, $\boldsymbol{s}$ is the standard errors, and $\hat{g}, q$ are estimated prior and computed posterior distributions respectively.

2.3.1 Quadratic approximation of log-likelihood

Replacing the Poisson log-likelihood in ELBO (2.4) by its quadratic approximation facilities the use of Gaussian methods, because of the quadratic form of the Gaussian log-likelihood. The new objective function is then an approximation of the ELBO. If the quadratic approximation is also a lower bound of the Poisson log-likelihood, then the new objective function is still a lower bound of the evidence, since it is a lower bound of the ELBO. However, for Poisson model with log-link function, there is no quadratic lower bound of the log-likelihood, because of the $\exp(\mu)$ term.

We consider two existing methods that enable us to solve the VEBPM problem by iteratively solving an EBNM problem. The first method uses an alternative link function (other than log-link) and the second method builds on the connection between Poisson and negative binomial distributions. Seeger and Bouchard [2012] proposed a quadratic lower bound of Poisson log-likelihood with softplus link function, based on Taylor's theorem. See Appendix A.3 for a detailed discussion. Another method is based on the fact that Poisson distribution is a limiting distribution of negative binomial distribution, and we can solve a negative binomial mean problem with fixed large r as an approximation to the Poisson mean problem (See Appendix A.4).

2.3.2 The splitting variational inference approach

The splitting variational inference method that will be introduced in Chapter 4 can be applied to solve the EB Poisson mean problem with log-link function. Recall the original Poisson mean model is

$$\begin{aligned} &y_j|\mu_j \sim \text{Poisson}(\exp(\mu_j)), \\ &\boldsymbol{\mu} \sim g(\cdot). \end{aligned} \tag{2.11}$$

We introduce a splitting variable b, such that the model is

$$\begin{aligned} &y_j|\mu_j \sim \text{Poisson}(\exp(\mu_j)), \\ &\mu_j|b_j \sim N(b_j, \sigma^2), \\ &\boldsymbol{b} \sim g(\cdot). \end{aligned} \tag{2.12}$$

Assume the posterior factorizes as

$$q(\boldsymbol{\mu}, \boldsymbol{b}) = \prod_j q_{\mu_j}(\mu_j) q_{\boldsymbol{b}}(\boldsymbol{b}),$$

where $q_{\mu_j}(\mu_j) = N(\mu_j; \bar{\mu}_j, V_{\mu_j})$. The ELBO is

$$F(q_{\boldsymbol{\mu}}, q_{\boldsymbol{b}}, g; \sigma^2) = \sum_j \mathbb{E} \log \frac{p(y_i|\mu_j)}{q_{\mu_j}(\mu_j)} + \sum_j \mathbb{E} \log p(\mu_j|b_j) + \mathbb{E} \log \frac{g(\boldsymbol{b})}{q_{\boldsymbol{b}}(\boldsymbol{b})}. \tag{2.13}$$

The splitting variational inference algorithm is given in Algorithm 2. It follows from the general splitting variational inference framework and a more detailed development is in Chapter 4.

Algorithm 2 Splitting variational inference for VEBPM

Input: $\boldsymbol{y}$
Init: $q_{\boldsymbol{b}}, \sigma^2$
repeat
  Given $q_{\boldsymbol{b}}$ and σ^2, update q_{μ_j} by solving a VGA Poisson problem (A.3) with prior mean $\mathbb{E}\, b_j$ and prior variance σ^2, for $j = 1, 2, ..., n$.
  Given $q_{\boldsymbol{\mu}}$ and σ^2, update $q_{\boldsymbol{b}}, g$ by solving the EBNM problem $(q_{\boldsymbol{b}}, \hat{g}) = \text{EBNM}(\bar{\boldsymbol{\mu}}, \sigma^2)$
  Given $q_{\boldsymbol{\mu}}$ and $q_{\boldsymbol{b}}$, update σ^2 as $\sigma^2 = \sum_j \mathbb{E}(\mu_j - b_j)^2/n$.
until Converged
Output: $\hat{g}, \hat{\sigma}^2, q_{\boldsymbol{\mu}}, q_{\boldsymbol{b}}$

2.4 Gradient-based VEBPM

Kim et al. [2022] introduced a penalty formulation for the empirical Bayes normal mean problem, which potentially transforms the maximization of ELBO over distributions to the minimization of a loss function over parameters that take real values. In this section, we generalize this method to deal with non-Gaussian observations. Suppose we have observations y_j from a random variable with likelihood $p(y_j|b_j)$, and the parameters b_j are drawn from a common prior

$$b_j \sim g(\cdot). \tag{2.14}$$

We consider performing inference on the posterior of $\boldsymbol{b}$. Denote the log-likelihood of b_j as $l(b_j) := \log p(y_j|b_j)$ and the posterior distribution as $q_{b_j}(\cdot)$.

The variational empirical Bayes method finds $q, \hat{g}$ by maximizing the ELBO

$$F(q, g) = \sum_j \mathbb{E}_q\, l(b_j) - D_{KL}(q_{b_j}||g(b_j)). \tag{2.15}$$

In the following sections, when working on a single observation, we may omit the subscript j for notation simplicity.

2.4.1 Formulation of penalty term

Denote the posterior mean of b as $\bar{b}_q = \mathbb{E}_q(b)$, and posterior variance as $V_q = \mathrm{Var}_q(b)$. A second order Taylor series expansion of $l(b)$ around $\bar{b}_q$ gives

$$l(b) \approx \tilde{l}(b) := l(\bar{b}_q) + l'(\bar{b}_q)(b - \bar{b}_q) + \frac{1}{2} l''(\bar{b}_q)(b - \bar{b}_q)^2. \tag{2.16}$$

Replacing the log-likelihood in $F(q, g)$ by its Taylor series expansion (2.16), we have the approximated objective function as

$$\begin{aligned} \tilde{F}(q, g) &= \mathbb{E}_q \, \tilde{l}(b) - D_{KL}(q||g) \\ &= l(\bar{b}_q) + \frac{1}{2} l''(\bar{b}_q) V_q - D_{KL}(q||g). \end{aligned} \tag{2.17}$$

To formulate the optimization problem, we begin with re-writing the optimization problem in two-steps, as

$$\begin{aligned} \max_{q,g} \tilde{F}(q, g) &= \max_{\theta, g} \max_{q : \mathbb{E}_q \, b = \theta} \tilde{F}(q, g) \\ &= \max_{\theta, g} -h(\theta, g) \\ &= -\min_{\theta, g} h(\theta, g), \end{aligned} \tag{2.18}$$

where

$$\begin{aligned} h(\theta, g) &:= \min_{q : \mathbb{E}_q \, b = \theta} -\tilde{F}(q, g) \\ &= -l(\theta) + r_g(\theta), \end{aligned} \tag{2.19}$$

and

$$
\begin{aligned}
r_g(\theta) &= \min_{q:\mathbb{E}_q\, b=\theta} \left(\frac{1}{2s^2(\bar{b}_q)} V_q + D_{KL}(q||g) \right), \\
s^2(\cdot) &= -(l''(\cdot))^{-1}.
\end{aligned}
\tag{2.20}
$$

The optimization problem is

$$
\min_{\theta,g} h(\theta, g) = -l(\theta) + r_g(\theta). \tag{2.21}
$$

We list the form of $s^2(\cdot)$ for commonly used distributions here.

1. Normal $y \sim N(b, s^2)$, where s^2 is known. $s^2(\theta) = s^2$.

2. Poisson $y \sim \text{Poisson}(\exp(b))$. $s^2(\theta) = \exp(-\theta)$.

3. Binomial $y \sim \text{Binom}\left(n, \frac{1}{1+\exp(-b)}\right)$, where n is known. $s^2(\theta) = \frac{(1+\exp(\theta))^2}{n \exp(\theta)}$.

2.4.2 Evaluation of $r_g(\theta)$

We show that the optimal q in the $r_g(\theta)$, under the mean constraint, is a convolution of Gaussian density and prior g.

Theorem 2.4.1. *The value of $r_g(\theta)$ is*

$$
r_g(\theta) = -l_{NM}(z_g(\theta); g, s^2(\theta)) - \frac{(z_g(\theta) - \theta)^2}{2s^2(\theta)} - \frac{1}{2} \log 2\pi s^2(\theta), \tag{2.22}
$$

which is achieved when

$$
q(b) = \frac{g(b) N(z; b, s^2(\theta))}{c(z, s^2(\theta))}, \tag{2.23}
$$

where $l_{NM}(\cdot)$ is the marginal log-likelihood of normal mean model, $c(\cdot) = \int g(b) N(z; b, s^2) db$, $\theta = S_g(z, s^2(\theta))$ and $S_g(\cdot, \cdot)$ is the posterior mean operator under the normal mean model

Proof. To solve $r_g(\theta)$, we formulate the Lagrangian multiplier as

$$L(q, \lambda_0, \lambda_1) = \frac{1}{2s^2(\theta)} \int q(b)(b-\theta)^2 + D_{KL}(q||g) + \lambda_0(\int q - 1) + \lambda_1(\int bq(b) - \theta), \quad (2.24)$$

where the last two terms come from the constraints that q is a density and its mean is θ. Taking derivative of $L(q, \lambda_0, \lambda_1)$ with respect to q and set it to 0, we have

$$\begin{aligned} &- \log g(b) + 1 + \log q(b) + \lambda_0 + \lambda_1 b + \frac{1}{2s^2(\theta)}(b-\theta)^2 = 0 \\ &\implies q(b) = g(b) e^{-\frac{1}{2s^2(\theta)}(b-\theta)^2 - \lambda_1(\theta) b - \lambda_0(\theta) - 1}. \end{aligned} \quad (2.25)$$

where $\lambda_0(\theta), \lambda_1(\theta)$ are solutions to $\partial L/\partial \lambda_0 = 0, \partial L/\partial \lambda_1 = 0$ respectively. The optimal q thus has the form

$$q(b) = \frac{g(b) N(z; b, s^2(\theta))}{c(z, s^2(\theta))}, \quad (2.26)$$

where $c(\cdot) = \int g(b) N(z; b, s^2) db$ and $\theta = S_g(z, s^2(\theta))$. The function $S_g(\cdot, \cdot)$ is the posterior mean operator under the normal mean model. Note that we can write z as $S_g^{-1}(\theta)$, and we will use notation $z_g(\theta)$ to highlight it's a function of θ.

The $r_g(\theta)$ now evaluates as

$$\begin{aligned} r_g(\theta) &= -\log c(z_g(\theta), s^2(\theta)) - \frac{(z_g(\theta) - \theta)^2}{2s^2(\theta)} - \frac{1}{2} \log 2\pi s^2(\theta) \\ &= -l_{\mathrm{NM}}(z_g(\theta); g, s^2(\theta)) - \frac{(z_g(\theta) - \theta)^2}{2s^2(\theta)} - \frac{1}{2} \log 2\pi s^2(\theta). \end{aligned} \quad (2.27)$$

□

Finally the optimization (2.21) has the form

$$\begin{aligned}\min_{\theta,g} h(\theta, g) &= -l(\theta) - l_{\mathrm{NM}}(z_g(\theta); g, s^2(\theta)) - \frac{(\theta - z_g(\theta))^2}{2s^2(\theta)} - \frac{1}{2}\log 2\pi s^2(\theta), \\ &:= -l(\theta) - \frac{1}{2}\log 2\pi s^2(\theta) + \rho_g(\theta),\end{aligned} \tag{2.28}$$

where $\rho_g(\theta)$ is a penalty term that shrinks posterior mean θ towards prior mean. We have transformed variational empirical Bayes problem over distributions into an optimization problem over real parameters. The new objective function $h(\theta, g)$ consists of three parts - the negative log-likelihood, the negative log observed Fisher information and a penalty term. This form is analogous to the widely used penalized regression, in which the estimator can usually be interpreted as the MAP estimator. However, our optimization problem finds the posterior mean, which is Bayes risk optimal under the squared-error loss function.

Kim et al. [2022] first proposed the penalty for the normal likelihood and it can be easily verified that for normal distribution, the penalty (2.28) is the same as the one in Lemma 9 of Kim et al. [2022]. However the derivation of the penalty here is more general and applicable to non-Gaussian data.

2.4.3 Solving the optimization problem

We consider two possible optimization approaches for finding the posterior mean. The first approach is called the inversion method, which directly optimizes over the posterior mean θ. It requires numerically inverting the normal mean operator, i.e., given the prior g and the posterior mean θ, finding the corresponding data z. The nice feature of this method is that it is an unconstrained optimization problem and we only need to optimize over θ. However, the downside is that the inversion operation is required for each iteration in the optimization algorithm, which can be computationally expensive.

The second approach is called the compound method that replaces θ by $S_g(z, s^2)$, then optimizes over z. In this case the optimization problem is

$$\begin{aligned} &\min_{z,s^2,g} h(z, s^2, g) = -l(S_g(z, s^2)) - l_{\text{NM}}(z; g, s^2) - \frac{(S_g(z, s^2) - z)^2}{2s^2} - \frac{1}{2}\log 2\pi s^2 \\ &\text{subject to } s^2 = (-l''(S_g(z, s^2)))^{-1}, s^2 > 0. \end{aligned} \tag{2.29}$$

We can use existing solvers to solve the equality-constrained optimization problem. To avoid the inequality constraint, we can re-parameterize s^2 as $s^2 = \exp(v)$. The compound method introduces an additional constraint but avoids the inversion of $S_g(z, s^2)$. In our simulations, we find that the two approaches often give similar results.

Figure 2.1 shows the plots of normal mean penalties in both inversion and compound methods. The prior is $g = \pi_0\delta_0 + \pi_1 N(0, \sigma^2)$ where $\pi_0 = 0.9$ or 0.5 and σ is 1 or 2. Both penalties are symmetric, and achieve the minimum at prior mean 0.

In this study, we use "adaptive shrinkage" (ash) prior from Stephens [2017]. Specifically, the prior is a mixture of normal distributions $g(\cdot) = \pi_k N(\cdot; 0, \sigma_k^2)$, and we re-parameterize $\pi_k = \exp(a_k)/\sum_{l=1}^K \exp(a_l)$ so that a_k is unconstrained. For the optimization method, we use L-BFGS-B (Byrd et al. [1995]) for unconstrained optimization problems (the inversion approach), as implemented in the `optim` function in `R`. The inversion step finds the root of the function $S_g(z, s^2(\theta)) - \theta$ with respect to z, and we use a bisection method because it's stable and fast. For the compound approach, we use gradient-based augmented Lagrangian method as implemented in the `nloptr` package in `R`.

2.4.4 A comparison of the penalized EBNM with ash

We compare the penalized EBNM method with ash (Stephens [2017]) on shrinkage estimation of mean parameters using a simple simulation example. For normal likelihood, the

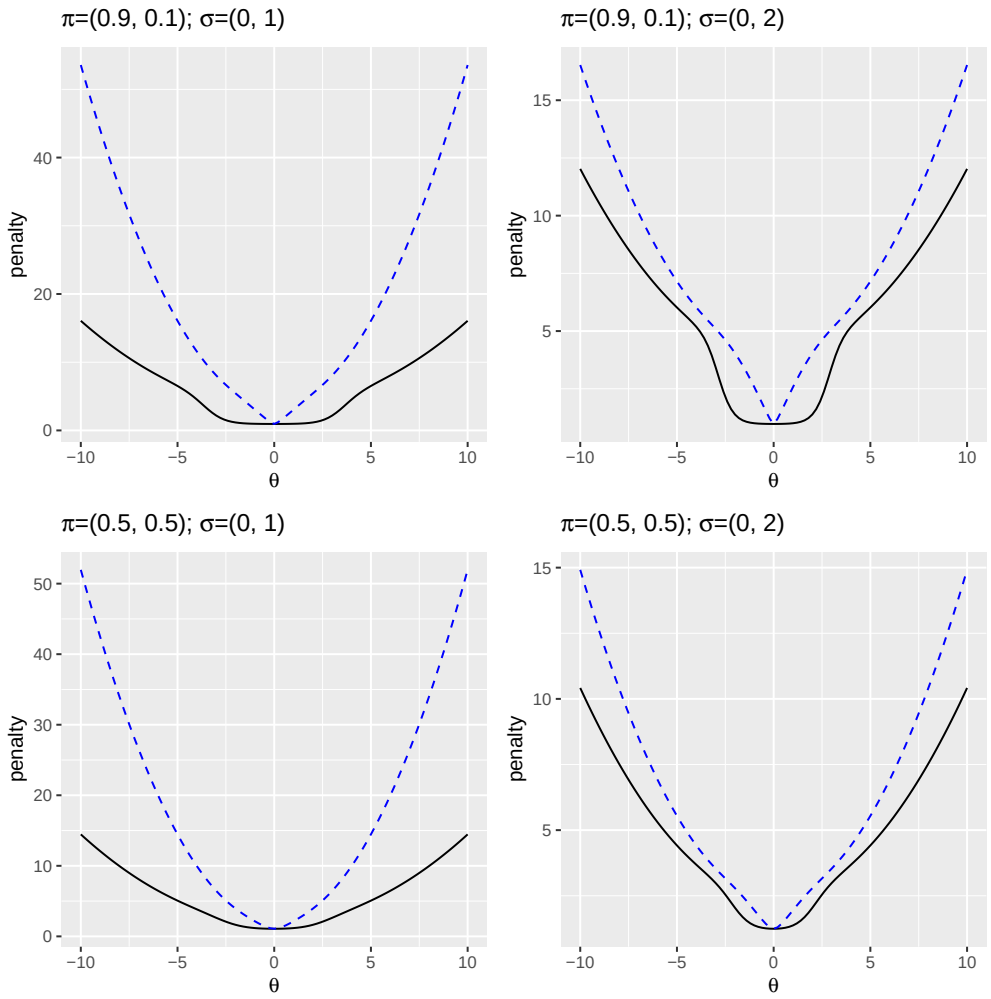


Figure 2.1: Compound and inversion (original) penalties for normal mean problem. The black line is the compound penalty, $\rho(S(\theta))$; the blue dashed line is the inversion (original) penalty $\rho(\theta)$.

optimization problems greatly simplify, because the variance term is no longer a function of θ.

Recall the EBNM model is

$$\begin{aligned} y_i|b_i &\sim N(b_i, s_i^2) \\ b_i &\sim g(\cdot), \end{aligned} \tag{2.30}$$

where s_i is known standard error.

Following the general optimization form (2.29) for the compound approach, the optimization problem for EBNM is

$$\min_{\boldsymbol{z},g} h(\boldsymbol{z}, g) = \sum_i \frac{1}{2s_i^2}(y_i - S_{g,s_i^2}(z_i))^2 - l_{\text{NM}}(z_i; g, s_i^2) - \frac{(S_{g,s_i^2}(z_i) - z_i)^2}{2s_i^2} - \frac{1}{2}\log 2\pi s_i^2. \tag{2.31}$$

where $S_{g,s_i^2}(\cdot))$ is the posterior mean operator under normal mean model (2.30).

The partial derivative of $h(\boldsymbol{z}, g)$ with respect to z_i is

$$\begin{aligned} \frac{\partial h(\boldsymbol{z}, g)}{\partial z_i} &= \frac{1}{s_i^2}(S_{g,s_i^2}(z_i) - y_i)S'_{g,s_i^2}(z_i) + S'_{g,s_i^2}(z_i)\frac{1}{s_i^2}(z_i - S_{g,s_i^2}(z_i)) \\ &= \frac{S'_{g,s_i^2}(z_i)}{s_i^2}(z_i - y_i), \end{aligned} \tag{2.32}$$

where we have used the fact that $l'_{NM}(z; g, s^2) = (S_{g,s^2}(z) - z)/s^2$. Note that $S'_{g,s^2}(z) = 1 + s^2 l''_{NM}(z; g, s^2)$. We can show that $l''_{NM}(z; g, s^2) \geq -1/s^2$ and the equality holds when $g(\cdot)$ is a point mass. Hence, the optimal z_i is actually the observation y_i. This is not a very surprising result. Consider the case where $g(\cdot)$ is known, it's obvious that the optimal z_i is y_i.

For the inversion method, following the general optimization form (2.28), we have

$$\min_{\boldsymbol{\theta},g} h(\boldsymbol{\theta},g) = \sum_i \frac{1}{2s_i^2}(y_i-\theta_i)^2 - l_{\mathrm{NM}}(z_{g,s_i^2}(\theta_i);g,s_i^2) - \frac{(\theta_i - z_{g,s_i^2}(\theta_i))^2}{2s_i^2} - \frac{1}{2}\log 2\pi s_i^2, \tag{2.33}$$

where $z_{g,s_i^2}(\theta_i) = S^{-1}_{g,s_i^2}(\theta_i)$.

The partial derivative of $h(\boldsymbol{\theta},g)$ with respect to θ_i is

$$\begin{aligned}\frac{\partial h(\boldsymbol{\theta},g)}{\partial \theta_i} &= \frac{1}{s_i^2}(\theta_i - y_i) + \frac{1}{s_i^2}(S^{-1}_{g,s_i^2}(\theta_i) - \theta_i) \\ &= \frac{1}{s_i^2}(S^{-1}_{g,s_i^2}(\theta_i) - y_i).\end{aligned} \tag{2.34}$$

So the optimal θ_i is $S_{g,s_i^2}(y_i)$, which again is not surprising.

We ran a simple simulation example to illustrate the two approaches, and compare them with ash. We set $n = 200$, and $b_1 = b_2, ..., = b_{180} = 0$, and $b_{181} =, ..., b_{200} = 10$. All s_i^2 are set to be 1. All of the methods are using ash prior, and are provided the same prior variances σ_k^2 for $k = 0, 1, 2, ..., K$, which are obtained using `ebnm:::get_ashr_grid` function. We used function `ash` from `R` package `ashr` for fitting ash model. For inversion and compound methods, the initial value of π_k is $1/K$. The initial value of $\boldsymbol{\theta}$ in inversion method is $\mathbf{0}_n$.

All three methods give identical log-likelihood at -408.2915, and almost identical posterior mean, as well as fitted prior $\hat{\boldsymbol{\pi}}$. Figure 2.2 shows the posterior mean from the three methods. Though the results of the three methods are very similar, `ashr` runs the fastest and is the most stable method while the inversion and compound methods are slower.

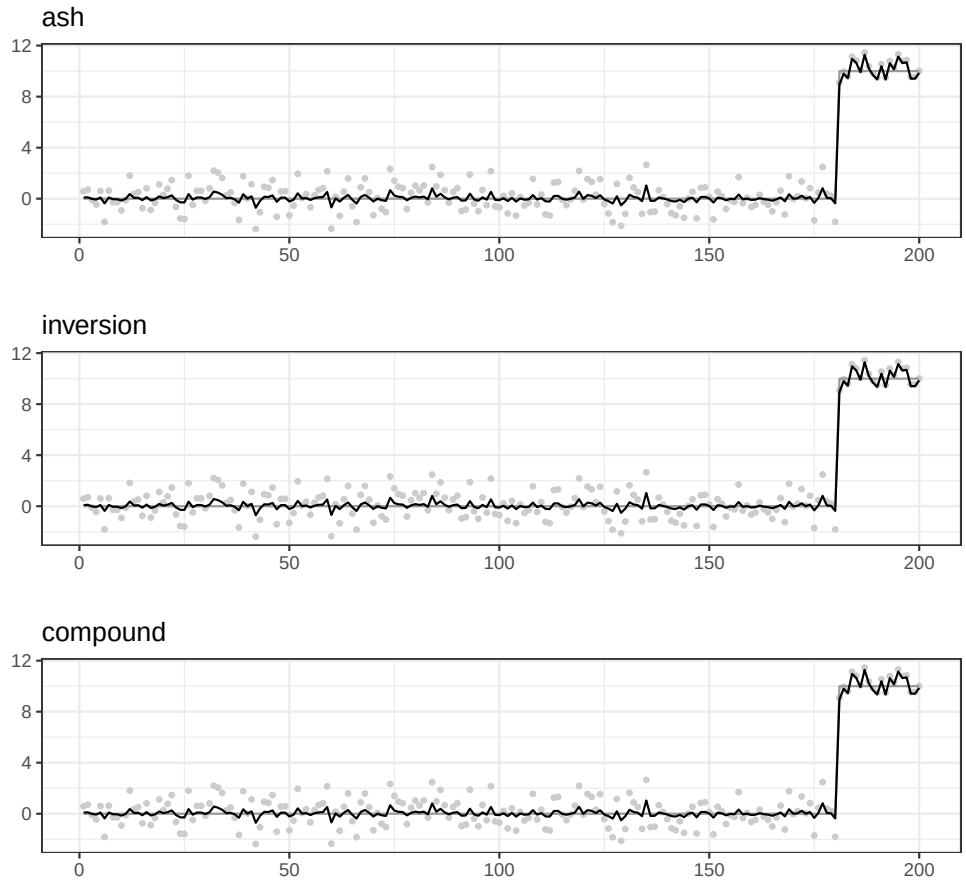


Figure 2.2: Posterior mean from three EBNM methods on simulated data. The grey dots are samples, grey line is the true mean parameter, and the black line is the posterior mean.

2.4.5 Extension to generalized linear model

We extend the penalty-based formulation of empirical Bayes mean problem to regression analysis. In particular, we focus on solving empirical Bayes generalized linear model (GLM). Consider the exponential family distributions,

$$f(y_i; \eta_i, \phi) = \exp\left(\frac{y_i\eta_i - b(\eta_i)}{\phi} + c(y_i, \phi)\right), \tag{2.35}$$

where η_i is the natural parameter and ϕ is the dispersion parameter. With canonical link function, the natural parameter η_i equates to the linear predictor, $\eta_i = \boldsymbol{x}_i^T\boldsymbol{\beta}$, where $\boldsymbol{x}_i$ is a

length p vector and $\boldsymbol{\beta}$ is the length p coefficients. The prior on β_j is

$$\beta_j \overset{iid}{\sim} g(\cdot). \tag{2.36}$$

A mean-field variational inference approach assumes the posterior factorizes as $q(\boldsymbol{\beta}) = \prod_j q_{\beta_j}(\cdot)$ and the evidence lower bound (ELBO) is then

$$F(q, g) = \mathbb{E} \sum_i l(\eta_i) - \sum_j D_{KL}(q_{\beta_j} || g), \tag{2.37}$$

where $l(\eta_i) = (y_i \eta_i - b(\eta_i))/\phi$ and $\eta_i = \boldsymbol{x}_i^T \boldsymbol{\beta}$. Denote the posterior mean of $\boldsymbol{\beta}$ as $\bar{\boldsymbol{\beta}}$, and posterior variance of β_j as $V_{q_{\beta_j}}$. The corresponding posterior mean of η_i is then $\bar{\eta}_i = \boldsymbol{x}_i^T \bar{\boldsymbol{\beta}}$. To formulate the penalty-based ELBO, we take a second order Taylor series expansion of $l(\eta_i)$ around $\bar{\eta}_i$

$$l(\eta_i) \approx \tilde{l}(\eta_i) = (y_i \eta_i - b(\bar{\eta}_i) - b'(\bar{\eta}_i)(\eta_i - \bar{\eta}_i) - \frac{1}{2} b''(\bar{\eta}_i)(\eta_i - \bar{\eta}_i)^2)/\phi. \tag{2.38}$$

Replacing $l(\eta_i)$ with $\tilde{l}(\eta_i)$ in (2.37), the approximated ELBO is then

$$\begin{aligned} \tilde{F}(q, g) &= \mathbb{E} \sum_i \tilde{l}(\eta_i) - \sum_j D_{KL}(q_{\beta_j} || g) \\ &= \sum_i l(\boldsymbol{x}_i^T \bar{\boldsymbol{\beta}}) - \sum_j \left(\frac{1}{2 s_j^2(\bar{\boldsymbol{\beta}})} V_{q_{\beta_j}} + D_{KL}(q_{\beta_j} || g) \right), \end{aligned} \tag{2.39}$$

where $1/s_j^2(\bar{\boldsymbol{\beta}}) = \sum_i b''(\boldsymbol{x}_i^T \bar{\boldsymbol{\beta}}) x_{ij}^2/\phi$. Note that $s_j^2(\bar{\boldsymbol{\beta}})$ is a function of the posterior mean vector $\bar{\boldsymbol{\beta}}$.

Following the similar argument in section 2.4.1, we solve the following optimization to get

(approximated) posterior mean $\boldsymbol{\beta}$, and estimated prior $\hat{g}$,

$$\min_{\boldsymbol{\theta},g} -\sum_i l(\boldsymbol{x}_i^T\boldsymbol{\theta}) + \sum_j r_j(\boldsymbol{\theta}, g), \tag{2.40}$$

where

$$r_j(\boldsymbol{\theta}, g) = \min_{q:E_q=\boldsymbol{\theta}} \frac{1}{2s_j^2(\boldsymbol{\theta})} V_{q_{\beta_j}} + D_{KL}(q_{\beta_j}||g). \tag{2.41}$$

2.5 Simulation

We compare the proposed methods on estimating the Poisson mean parameter under different simulation settings. For comparisons, three additional methods are included - gamma prior, mixture of exponential prior and Poisson ash with mixture of uniform distributions (Lu [2018]). The models with the gamma prior and the mixture of exponential prior are described in `https://zihao12.github.io/ebpmf_demo/ebpm.pdf`. The performance of each method is measured by mean squared error (MSE) relative to maximum likelihood estimation (MLE) for estimating λ and MSE for estimating $\log(\lambda)$. For methods that are developed for log link function, we do not fix the prior mode and instead allow it to be estimated from the data. The methods in the comparisons are: the model with Gaussian prior and posterior (GG), the model with Gaussian mixture prior and posterior (GMGM), the Pólya-Gamma augmentation approach (nb_pg), Poisson ash using identity link (ash_pois_identity), ebpm with mixture of exponential distribution as prior (ebpm_exp_mixture), ebpm with gamma prior(ebpm_gamma), the model with $\log(1+\exp(\cdot))$ link function (log1exp), penalized ebpm compound approach (penalty_compound) and inversion approach (penalty_inversion), as well as the splitting method (split).

We generate $n = 1000$ samples from $y_j \sim \text{Poisson}(\lambda_j)$, with λ_j generated under the following different data-generating distributions

a. $\lambda_j \sim \text{Exp}(1)$.

b. $\lambda_j \sim \text{Exp}(0.1)$.

c. $\lambda_j \sim \pi_0 \delta_0 + \pi_1 \,\text{Exp}(0.1)$.

We also compare the methods when generating data using a log-link. We generate $n = 1000$ samples from $y_j \sim \text{Poisson}(\exp(\mu_j))$, and μ_j are generated under the following different data-generating distributions

d. $\mu_j \sim N(0, 2)$.

e. $\mu_j \sim 0.8\delta_0 + 0.2N(0, 2)$.

f. $\mu_j \sim 0.8\delta_5 + 0.2N(5, 2)$.

For each prior, we simulate 30 datasets, and plot the mean run time(log2 base, seconds) and the MSE for each method. Figure 2.3 displays the results of simulation a, b, and c, where the mean parameters are sampled from an exponential distribution or a mixture of point-mass and exponential distribution. Since the link function for data generation is identity, we expect methods that directly work with the identity link function to generally perform better. Indeed, the two ebpm methods are positioned at the bottom left corner of all plots, indicating that they are the fastest and most accurate in terms of MSE. All other methods, except nb_pg and log1exp, perform similarly in terms of the estimation accuracy. The two penalized methods and GMGM are very close in the plots. This is not surprising given that they are different methods for solving the same model – log link with ash prior. The penalized vebpm with inversion approach seems to be slightly faster compared to the other two methods. The Pólya-Gamma and log1exp methods are not very stable across different simulation settings. For the Pólya-Gamma augmentation method, we set $r = 100$, and altering r can have influence on the convergence. For log1exp method,

it uses a constant variance that depends on the maximum observation for pseudo-Gaussian data. So it is unstable and the convergence is usually very slow. The splitting method falls in the middle in terms of run time and accuracy, providing almost the same estimation as the GG method, because the algorithm converges to a local optimum where $g(\cdot)$ for b_i is a point-mass, resulting in an identical model as GG.

Figure 2.4 presents the results of simulation d, e, and f, where the log of mean parameters are sampled from a normal or a mixture of normal distribution. In simulation d and e, the prior mode is 0, and we observe that the ebpm with gamma prior performs much worse in terms of mean parameter estimation, especially in the simulation e. This is likely because a single gamma distribution is not sufficiently flexible. While the splitting method and GG remain almost the same in simulation d, the splitting method slightly outperforms them when the natural parameters are sampled from mixture distributions. In simulation f, the prior mode is 5. Methods that cannot adapt to the prior mode generally struggle to provide accurate estimate of the mean parameters. The two penalized methods and the GMGM are again closely matched and have the smallest MSE. The Poisson ash method consistently performs well and runs quickly across different simulation settings. The Pólya-Gamma augmentation and log1exp approaches are again unstable, possibly for the same reasons mentioned earlier.

2.6 Discussion

In this chapter, we examined and compared variational empirical Bayes methods for solving EB Poisson mean problems, incorporating different link functions and priors. We focused on addressing the Poisson mean problem by leveraging existing empirical Bayes normal mean methods. Additionally, we proposed a novel penalty formulation for the mean inference problem and presented two approaches for solving the optimization problems.

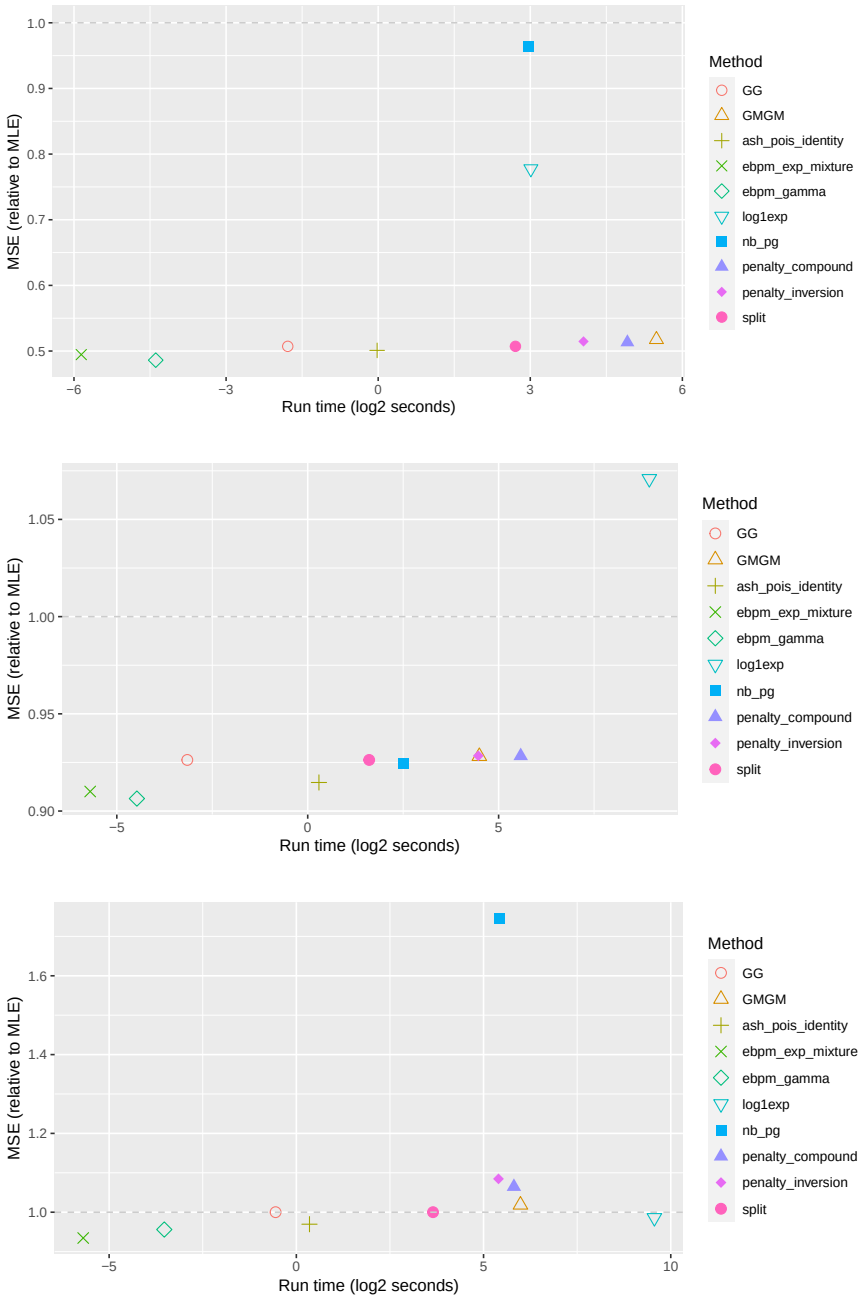


Figure 2.3: Run time (log2) and MSE (mean parameter, relative to MLE) in simulation study of VEBPM. Three plots correspond to simulation a, b, and c.

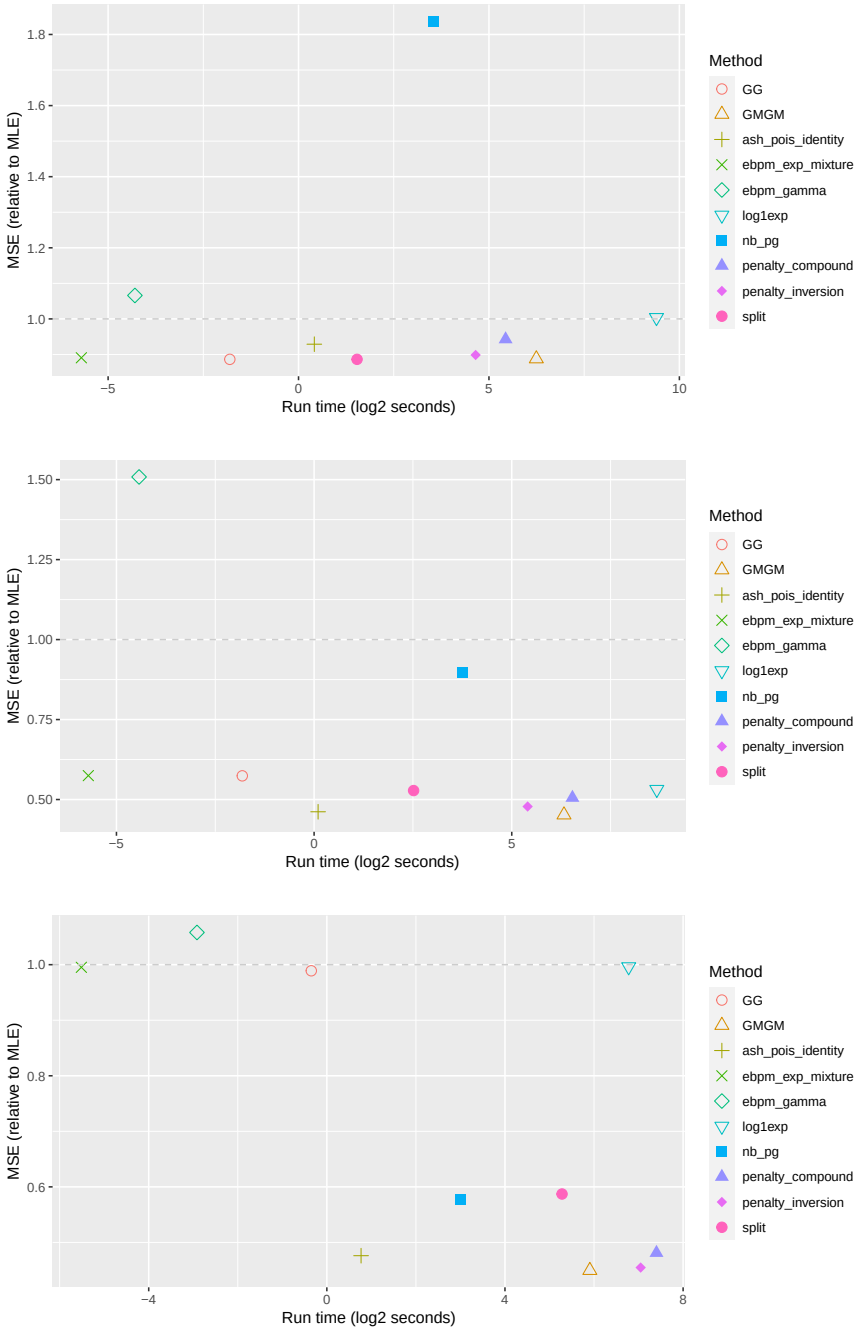


Figure 2.4: Run time (log2) and MSE (mean parameter, relative to MLE) in simulation study of VEBPM. Three plots correspond to simulation d, e, and f.

Although the two ebpm methods are fast and perform well in specific simulation settings, their priors are not adequately flexible, especially when the prior mode is not at 0. Moreover, these methods are tailored specifically to the Poisson mean problem, and generalizing them to more complex and potentially useful models, such as regression or matrix factorization, is non-trivial.

The splitting method, among those that leverage existing EBNM methods, performs the best. Although it may not be the most accurate method for mean estimation, it offers the advantage of simplicity in terms of algorithm and software development and can be easily extended to other models. Using the splitting approach, we extended EBMF to the Poisson distribution in Chapter 4.

Overall, the penalty-based methods, GMGM, and Poisson ash consistently outperform other methods due to the flexibility of their models and priors. In particular, the ash prior allows adaptation to the unknown mode and scale of the data. The penalty-based methods are a promising framework that can be extended to regression problems, as they can utilize existing optimization solvers and avoid coordinate updates typically found in mean-field variational inference algorithms.

Code availability

All the Poisson mean algorithms studied in this chapter have been implemented in `R` package `vebpm`, available at `https://github.com/DongyueXie/vebpm`

Code for plotting the EBNM penalty form is at `https://github.com/DongyueXie/gsmash/blob/main/analysis/normal_mean_penalty.Rmd`, and for comparing ash and penalty-based EBNM is available at `https://dongyuexie.github.io/gsmash/normal_mean_penalized_optimization.html`.

Code for running simulations is at `https://github.com/DongyueXie/gsmash/blob/main/code/poisson_mean/simu_thesis.R`, and for generating plots is at `https://github.com/DongyueXie/gsmash/blob/main/analysis/vebpm_simu_thesis.Rmd`.

CHAPTER 3

SMOOTHING SEQUENCING COUNT DATA

3.1 Introduction

Sequencing data along the genome motivates the development of spatial/smoothing models for count data. For example, Figure 3.1 shows the RNA-seq data of the $FTH1$ gene in adipose tissue from GTEx. The regions with high counts are the exons, where the expression is expected to be roughly constant and high. In contrast, low count regions are introns, where the expression level is mostly very low. Therefore, the underlying true expression is expected to be spatially-structured, and one may interested in recovering it using a statistical model. Nonparametric regression is a natural choice, but most methods are developed based on Gaussian likelihood while RNA-seq data are typically counts. One way to address this issue is to transform the count data (e.g. square root transformation) and then apply methods developed for Gaussian data. Alternatively, we can apply nonparametric regression methods developed for count data directly. For example, Xing et al. [2021] introduced smoothing via adaptive shrinkage (smash), a flexible empirical Bayes (EB) method. The method is based on wavelet denoising and can deal with Gaussian (smash-Gaussian) and Poisson (smash-Poisson) sequences. However, existing methods for Poisson sequence smoothing do not output smooth estimates of the true expression level when applied to RNA-seq data, mainly because base-specific effects introduce extra variations. See Figure 3.2 for an example. In this work, we generalize smash to handle over-dispersed count data.

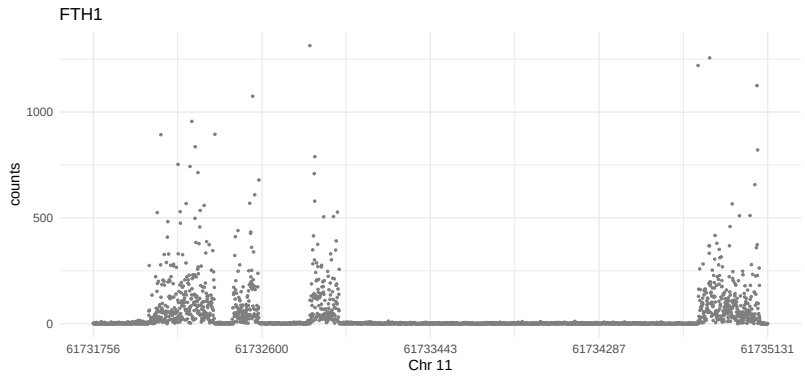


Figure 3.1: Scatter plot of gene *FTH1* RNA-seq data. The gene encodes the heavy subunit of ferritin, an iron storage protein.

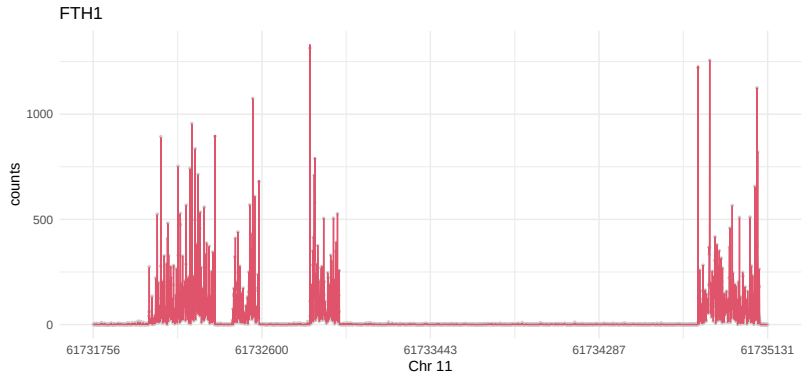


Figure 3.2: Smash-Poisson fit to the gene *FTH1* RNA-seq data. The fitted curve exhibits a significant lack of smoothness.

3.2 Model

To recover the underlying smooth signal in sequencing data, we introduce a nugget effect to account for the extra variations

$$\begin{aligned} & x_i|\lambda_i \sim \text{Poisson}(\lambda_i), i = 1, 2, ..., n, \\ & h(\lambda_i) = \mu_i + \epsilon_i, \\ & \boldsymbol{\mu} \sim g_{\text{smooth}}(\cdot) \in \mathcal{G}_{\text{smooth}}, \\ & \epsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2), \end{aligned} \tag{3.1}$$

where $h(\cdot)$ is the link function (e.g. log), $g_{\text{smooth}}(\cdot)$ is a smoothness-inducing prior so that $\boldsymbol{\mu} = (\mu_1, ..., \mu_n)$ is spatially-structured and ϵ_i are i.i.d Gaussian distributed random variables sometimes known as the nugget effect. The nugget effect originates from geostatistics (Carrasco [2010]) and it is used to account for the variation between two closely spaced samples.

It turns out that fitting the model (3.1) via empirical Bayes approach is nontrivial, mainly due to the Poisson likelihood and smoothness constraints (priors). We explore two methods to fit this model. The first method involves approximating the Poisson likelihood by a Gaussian one and we call it the likelihood expansion approach. The advantages of the method are that it transforms the problem to the well-studied Gaussian smoothing and it allows us to use Gaussian smoothing methods. The second method applies the splitting variational inference introducing in Chapter 4 to solve the smoothing problem (3.1).

Before we introduce the detailed methods, we first define the following EB procedures.

Definition 3.2.1. *An EB normal smoothing (EBNS) procedure defines a mapping from* $(\boldsymbol{y}, \boldsymbol{s})$ *to* $(\hat{g}_{smooth}, q)$*, under the model*

$$y_i|\mu_i \sim N(\mu_i, s_i^2),$$
$$\boldsymbol{\mu} \sim g_{smooth}(\cdot),$$

where $\boldsymbol{y}$ *is a vector of observations,* $\boldsymbol{s}$ *is a vector of standard errors,* g_{smooth} *is a smoothness-inducing prior,* $\hat{g}_{smooth}$ *is the estimated prior, and* q *is the posterior. This mapping is denoted as*

$$(\hat{g}_{smooth}, q) = EBNS(\boldsymbol{y}, \boldsymbol{s}).$$

When s_i^2 *is unknown, and is to be estimated within the procedure, we denote the procedure as* $(\hat{g}_{smooth}, \hat{\boldsymbol{s}}, q) = EBNS(\boldsymbol{y})$.

We highlight the connection between EBNS and empirical Bayes normal mean (EBNM, Willwerscheid and Stephens [2021]) problem. In EBNM, the prior is usually specified as a univariate one, $\mu_i \overset{iid}{\sim} g(\cdot)$. While in EBNS, the prior is multivariate and structured.

Definition 3.2.2. *An EB Poisson smoothing (EBPS) procedure defines a mapping from* $\boldsymbol{y}$ *to* $(\hat{g}_{smooth}, q)$*, under the model*

$$y_i|\lambda_i \sim \text{Poisson}(\lambda_i),$$
$$\boldsymbol{\lambda} \sim g_{smooth}(\cdot),$$

where $\boldsymbol{y}$ *is a vector of observations,* $\hat{g}_{smooth}$ *is the estimated prior,* g_{smooth} *is a smoothness-inducing prior, and* q *is the posterior. This mapping is denoted as*

$$(\hat{g}_{smooth}, q) = EBPS(\boldsymbol{y}).$$

Definition 3.2.3. *A variational Bayes Poisson mean (VBPM) procedure defines a mapping from* $(\boldsymbol{y}, g)$ *to* q*, under the model*

$$y_i|\mu_i \sim \text{Poisson}(\exp(\mu_i)),$$
$$\mu_i \sim g(\cdot).$$

where $\boldsymbol{y}$ *is a vector of observations,* g *is the (known) prior, and* q *is the posterior. This mapping is denoted as*

$$q = VBPM(\boldsymbol{y}, g).$$

One example of the smoothness-inducing prior g_{smooth} in EBNS problem is the wavelet prior (see Appendix B.1). Choosing wavelet prior results in an empirical Bayes wavelet denoising model. Wavelet denoising has shown its success in removing noises from signals in many

applications (Donoho [1995], Donoho and Johnstone [1998], Coifman and Donoho [1995]). A typical wavelet denoising procedure involves decomposing the signal into wavelet coefficients, shrinking the coefficients, and then reconstructing the smoothed signal. Among the wavelet smoothing methods, smash (Xing et al. [2021]) is a flexible smoothing method based on discrete wavelet transformation (DWT) and empirical Bayes shrinkage.

3.3 Likelihood expansion approach

Consider a Poisson distributed random variable $x \sim \text{Poisson}(\lambda)$, and let $\eta := \log \lambda$, then the log-likelihood is $l(\eta; x) = x\eta - \exp(\eta)$. Note that η is the canonical parameter of Poisson distribution. A second order Taylor series expansion of $l(\eta; x)$ around $\tilde{\eta}$ gives

$$l(\eta; x) \approx \bar{l}(\eta; x, \tilde{\eta}) = l(\tilde{\eta}; x) + (x - \exp(\tilde{\eta}))(\eta - \tilde{\eta}) - \frac{\exp(\tilde{\eta})}{2}(\eta - \tilde{\eta})^2. \tag{3.2}$$

The approximated log-likelihood $\bar{l}(\eta; x, \tilde{\eta})$ is quadratic in η and after completing the square, we have

$$\bar{l}(\eta; x, \tilde{\eta}) = -\frac{1}{2\exp(-\tilde{\eta})}\left(\tilde{\eta} + \frac{x - \exp(\tilde{\eta})}{\exp(\tilde{\eta})} - \eta\right)^2 + \text{const}, \tag{3.3}$$

This suggests that the pseudo-variable $y := \tilde{\eta} + \frac{x - \exp(\tilde{\eta})}{\exp(\tilde{\eta})}$ admits a Gaussian likelihood with mean η and variance $\exp(-\tilde{\eta})$ approximately. The log-likelihood $l(\eta; x)$ and $\bar{l}(\eta; x, \tilde{\eta})$ are equivalent when $\eta = \tilde{\eta}$.

We now need to choose $\tilde{\eta}$. One natural choice is $\tilde{\eta} = \log x$, where x is the maximum likelihood estimate of λ. However we have to decide what to do when $x = 0$ and in practice, the proportion of zeros could be large. Another choice is to apply an empirical Bayes procedure to get an estimate of λ using the posterior mean. For example, we can apply Poisson-ash (Lu [2018]) to all observations and calculate the posterior mean. Here, we choose $\tilde{\eta}$ to be $\log x$ if x is nonzero and log of the posterior mean obtained from the Poisson-ash (the prior

on λ is unimodal at 0) if x is 0.

Given the pseudo-data y_i, the task is to estimate σ^2 and $\boldsymbol{\mu}$ in the model $y_i \sim N(\mu_i, \sigma^2 + s_i^2)$, where $s_i^2 = \exp(-\tilde{\eta})$ is known, and $\boldsymbol{\mu}$ is spatially-structured. Suppose $\boldsymbol{\mu}$ is known, we can use maximum likelihood estimation method for estimating σ^2. The log-likelihood of σ^2 is

$$l(\sigma^2; \boldsymbol{y}) = \sum_{i=1}^{n} l_i(\sigma^2; \boldsymbol{y}) = \sum_{i=1}^{n} -\frac{1}{2}\log(\sigma^2 + s_i^2) - \frac{1}{2}\frac{(y_i - \mu_i)^2}{\sigma^2 + s_i^2}. \tag{3.4}$$

The estimated σ^2 is obtained by solving

$$l'(\sigma^2; \boldsymbol{y}) = \sum_i \left(\frac{1}{\sigma^2 + s_i^2} - \frac{(y_i - \mu_i)^2}{(\sigma^2 + s_i^2)^2} \right) = 0, \tag{3.5}$$

which can be solved numerically. In practice, $\boldsymbol{\mu}$ is unknown and we can use the following two-step algorithm to estimate $\boldsymbol{\mu}$ and σ^2:

1. estimate $\boldsymbol{\mu}$ given σ^2 and $\boldsymbol{s}^2$, by applying an EBNS procedure to $\boldsymbol{y}$;

2. estimate σ^2 given $\boldsymbol{\mu}$ and $\boldsymbol{s}^2$, by solving (3.5).

Remark. *In real data, we observe that nugget effects are more obvious in larger counts. And for low-count regions, we expect there are less information about the nugget effect. So in the step 2, when solving* (3.5)*, we can use the top* 30% *largest counts. Specifically, let the index set of the top* 30% *largest counts be* $\mathcal{L}$*, then* $\hat{\sigma}^2 = \arg\max_{\sigma^2} \sum_{i \in \mathcal{L}} l_i(\sigma^2; \boldsymbol{y})$.

For the initialization of σ^2, we use a second order difference-based method presented in equation (3) of Brown and Levine [2007], which can be further traced back to Gasser et al. [1986]. The estimator is

$$\hat{\sigma}^2 = \frac{2}{3(n-2)} \sum_{i=1}^{n-2} \left(\frac{1}{2} y_i - y_{i+1} + \frac{1}{2} y_{i+2} \right)^2. \tag{3.6}$$

Algorithm 3 Splitting smoothing for count data

Input: Count vector $\boldsymbol{x}$.
Init: $\mathbb{E}\, b_i = \log(\sum_i x_i/n)$ for $i = 1, 2, ..., n$.
repeat
  Update q_{μ_i} by VBPM($\boldsymbol{x}, g(\cdot; \bar{\boldsymbol{b}}, \sigma^2)$), where $\bar{\boldsymbol{b}} = \mathbb{E}\, \boldsymbol{b}$;
  Update $q_{\boldsymbol{b}}, g_{\boldsymbol{b}}$ by EBNS($\bar{\boldsymbol{\mu}}, \sigma$), where $\bar{\boldsymbol{\mu}} = \mathbb{E}\, \boldsymbol{\mu}$;
  Update $\sigma^2 \leftarrow \mathbb{E} \sum_i (\mu_i - b_i)^2/n$.
until Converged
Output: Estimated priors $\hat{g}_{\boldsymbol{b}}$, $\hat{\sigma}^2$, and fitted posterior $q_{\boldsymbol{\mu}}$ and $q_{\boldsymbol{b}}$.

3.4 Variational splitting method

We use the splitting variational inference method introducing in Chapter 4 for solving the model (3.1). We re-write the model in the splitting form as

$$x_i|\mu_i \sim \text{Poisson}(\exp(\mu_i)), \tag{3.7a}$$

$$\mu_i|b_i \sim N(b_i, \sigma^2), \tag{3.7b}$$

$$\boldsymbol{b} \sim g_{\boldsymbol{b},\text{smooth}}(\cdot). \tag{3.7c}$$

The splitting method involves solving

(i) (3.7a) and (3.7b), a VBPM problem defined in Definition 3.2.3;

(ii) (3.7b) and (3.7c), an EBNS problem.

For step (i), the VBPM problem is solved using a convex optimization algorithm, and is discussed in detail in Section 4.3.1. In step (ii), we choose the empirical Bayes wavelet smoothing method studied in Johnstone and Silverman [2005] and Xing et al. [2021]. Specifically we use smash but use the discrete wavelet transformation (DWT) instead of Non-Decimated Wavelet Transform (NDWT) because the latter does not have an explicit objective function. The splitting variational inference algorithm is given in Algorithm 3.

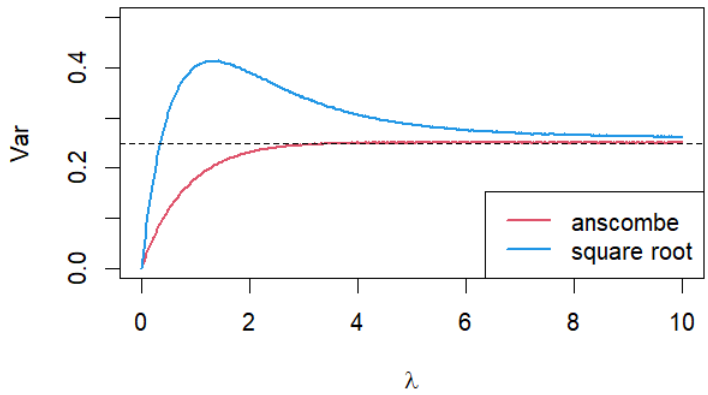


Figure 3.3: Comparison of two VST transformations.

3.5 Other methods

3.5.1 Variance stabilizing transformation

Variance stabilizing transformation (VST) transforms a random variable such that it has a constant variance. For a Poisson distributed random variable $x \sim \text{Poisson}(\lambda)$, the commonly used transformation is taking the square root of x plus a constant c, $y = \sqrt{x+c}$. Bartlett [1936] suggested the square root transformation $y = \sqrt{x}$, with $\mathbb{E}(y) \approx \sqrt{\lambda}$ and Anscombe [1948] suggested $y = \sqrt{x+3/8}$, with $\mathbb{E}(y) \approx \sqrt{\lambda+3/8}$. Both transformed variables have approximate constant variance, $\text{var}(y) \approx \frac{1}{4}$. Specifically, Anscombe [1948] showed that asymptotically

$$\begin{aligned} \mathbb{E}(y) &= \sqrt{\lambda+c} - \frac{1}{8\lambda^{1/2}} + O(\lambda^{-3/2}), \\ \text{var}(y) &= \frac{1}{4} + O(1/\lambda^2). \end{aligned} \tag{3.8}$$

In addition, when λ is large, the transformed variable is approximately normal distributed. After the VST, we are working with $y_i = \sqrt{x_i+c} \sim N(\sqrt{\lambda_i+c}, \frac{1}{4}))$. Similarly, to account for over-dispersion, we assume $\sqrt{\lambda_i+c}$ is drawn from a normal distribution with mean μ_i and variance σ^2, where $\boldsymbol{\mu} = (\mu_1, .., \mu_n)$ is smooth. Then we can apply Gaussian smoothing

method to the model $y \sim N(\mu, \sigma^2 + \frac{1}{4})$ and get an estimate of $\boldsymbol{\mu}$. One advantage of VST is that we avoid the complications associated with the computation of the log(0). But when the λ is small, the actual variance of transformed variable is smaller than $\frac{1}{4}$ and the distribution is not approximately normal distributed. Since the transformed sequence is homogeneous, we can apply the difference-based estimator (3.6) directly to the sequence $\boldsymbol{y}$ to get variance estimate (the total variance $\tilde{\sigma}^2 = \sigma^2 + 1/4$), then apply Gaussian smoothing method to $\boldsymbol{y}$ (with standard error $\sqrt{\hat{\tilde{\sigma}}^2}$). This method assumes a different link function other than the log-link, and we include it here due to its simplicity.

3.5.2 A two-step procedure

In the model (3.1), if $\boldsymbol{\lambda}$ were known, we can simply run an EBNS procedure such as smash on $\log(\boldsymbol{\lambda})$ with homogeneous variance. This idea naturally leads to the following two-step procedures:

1. run EBPS($\boldsymbol{x}$) to get $\hat{\boldsymbol{\lambda}}$;
2. run EBNS($\log \hat{\boldsymbol{\lambda}}$) to get $\hat{\boldsymbol{\mu}}$.

In the second step, when using smash as the EBNS procedure, we can apply smash-Gaussian with homogeneous or heteroskedastic variance assumptions. With the homogeneous assumption, the variance is estimated using the estimator (3.6). With heteroskedastic variance assumption, the variances are estimated within `smash.gaus` function in `R` package `smashr`. The smash-Gaussian with heteroskedastic variance assumes the variances are also spatially-structured. This assumption is satisfied because the pseudo-data from the two-step procedure are supposed to have constant variance. Despite the method appearing somewhat ad hoc, it is very simple and it performs well in practice, as we will see below.

3.6 Simulation

We compared the methods described in this chapter for the log-link function under various simulation settings. We considered six different types of smooth signals, namely clipped blocks, simple blocks, angles, bursts, spikes, and heavy sine waves (Donoho and Johnstone [1994]). The simple blocks functions are designed to simulate the exon and intron regions, while the other functions are commonly used in nonparametric regressions (Antoniadis et al. [2001], Donoho and Johnstone [1994]). We set $n = 1024$ and used two different signal-to-noise (SNR) ratios, 1 and 3, where the SNR is defined as $\text{var}(\boldsymbol{\mu})/\sigma^2$ in model (3.1). In addition, we varied the max-mean-count sizes by choosing three different values: 5, 10, and 100, where the max-mean-count size is defined as the maximum value of the smooth signal $\exp(\boldsymbol{\mu})$. The min-mean-count is fixed at 0.01. For each combination of settings, we simulated 30 datasets. We report the run time of each method and evaluate the performance using the rooted mean squared error (RMSE), which is defined as $\sqrt{\sum_i (\lambda_i - \hat{\lambda}_i)^2}$, where $\lambda_i = \exp(\mu_i)$.

For the likelihood expansion method, we include methods that use all observations (lik_exp) and use top 30% largest observations for estimating the nugget effect (lik_exp_top). For the two-step procedure, we apply smash-Gaussian with homogeneous (two_step) and heteroskedastic (two_step_hetero) variance assumptions in step 2. For the splitting method, we include results for using both DWT and NDWT as wavelet smoothing methods. The NDWT is included for fair comparison because all other methods use NDWT. We also tried two different initialization methods (of $\mathbb{E}\,\mu_i$ and $\mathbb{E}\,b_i$) for the splitting method - smash-Poisson and VGA.

Figure 3.4 and Figure 3.5 show the results when the SNR is 1, and the max-mean-count size is 5 and 100. For the two-step method, after obtaining the pseudo-data, running smash-Gaussian with heteroskedastic variance gives equally good or better signal estimation than assuming homogeneous variance in most cases, though it is a bit slower in terms of run

time. Thus, for the following analysis, we will refer to the methods as their heteroskedastic version. The performance of the two-step procedure is in the middle among all methods, both in terms of run time and RMSE.

The likelihood expansion approach appears to produce over-smoothed curves when counts are small (Figure 3.6a and Figure 3.7a). Additionally, using only the top largest counts for estimating the nugget effect performs poorly in small-count scenarios. However, when counts are larger, the opposite is observed. For instance, when the count size is small and the nugget effect is small (such as in the case of SNR = 3 and max-mean-count size = 5, as shown in Figure B.2), smash-Poisson can accurately estimate the smooth signal, even though it does not account for the extra variation. As the nugget effect or count size increases, smash-Poisson produces less smoothed estimates, resulting in a larger RMSE.

The DWT version of the splitting method uses Haar wavelet which gives piecewise-constant signal estimation. Therefore, in the simple blocks simulation, the DWT method gives the lowest RMSE, while the NDWT method has better estimation accuracy in other simulation settings. We focus on the NDWT version of the splitting method. The two initialization methods for splitting do not result in significantly different estimations, but the VGA initialization is usually faster. When the max-mean-count size is 5, the splitting method has a larger RMSE when estimating burst and spike signals. In general, the splitting method performs better than the likelihood expansion method when counts are small, but it can be worse than the two-step procedure in estimating certain signals, such as burst or spike. For larger counts, the splitting method works equally well or better than the likelihood expansion method as well as the two-step procedure. In Figure 3.6b, the splitting and likelihood expansion approaches give very similar results.

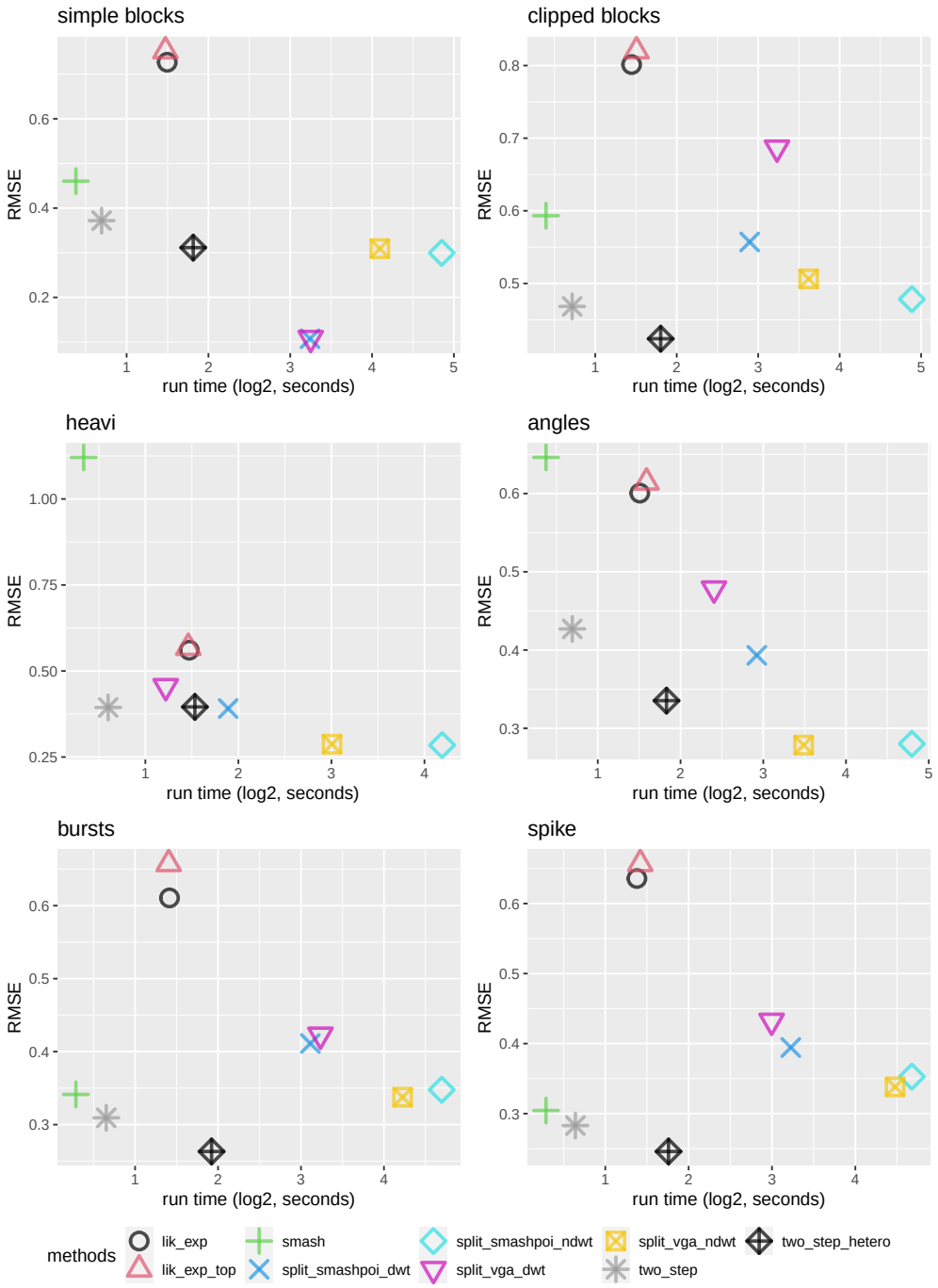


Figure 3.4: Plots of run time (log2 seconds) and RMSE when SNR = 1 and max-mean-count size = 5, in the simulation study of smoothing count data.

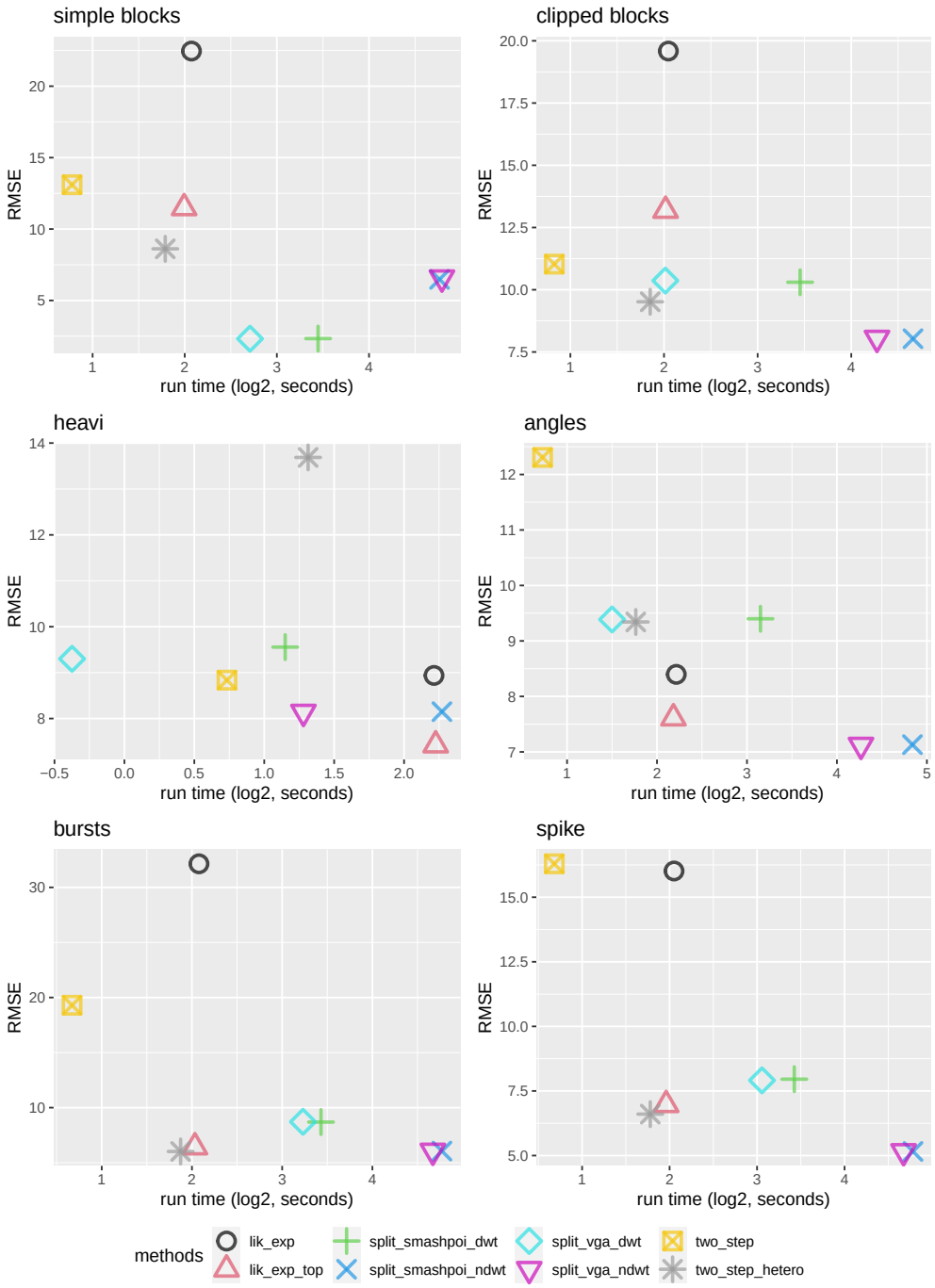
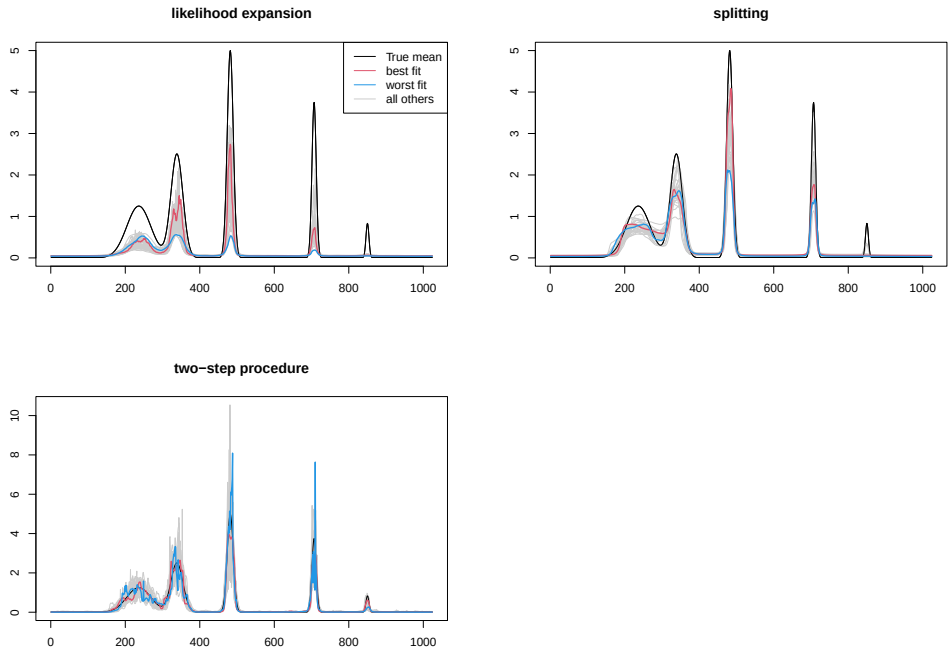
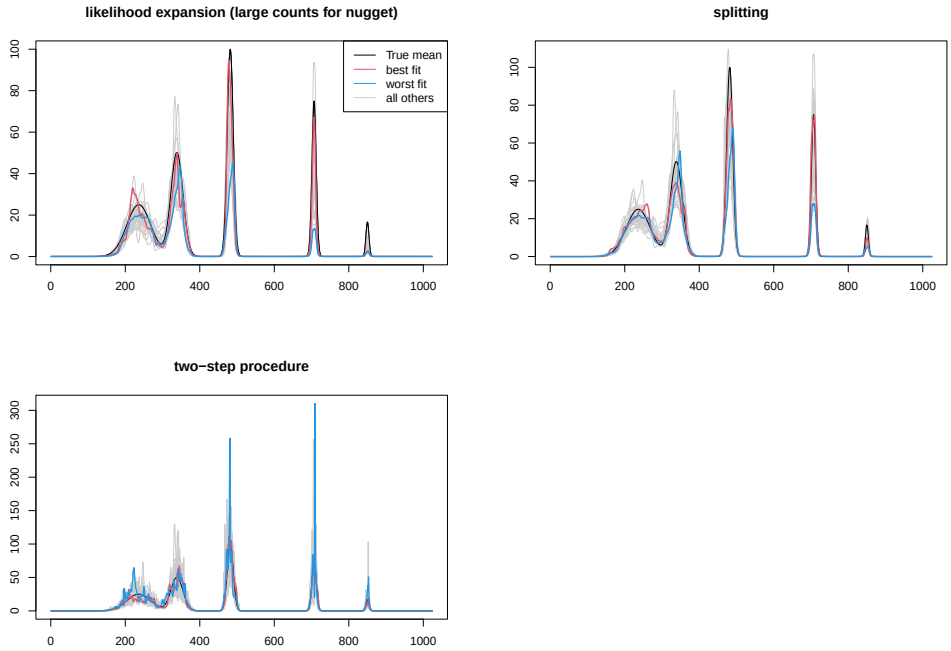


Figure 3.5: Plots of run time (log2 seconds) and RMSE when SNR = 1 and max-mean-count size = 100, in the simulation study of smoothing count data.

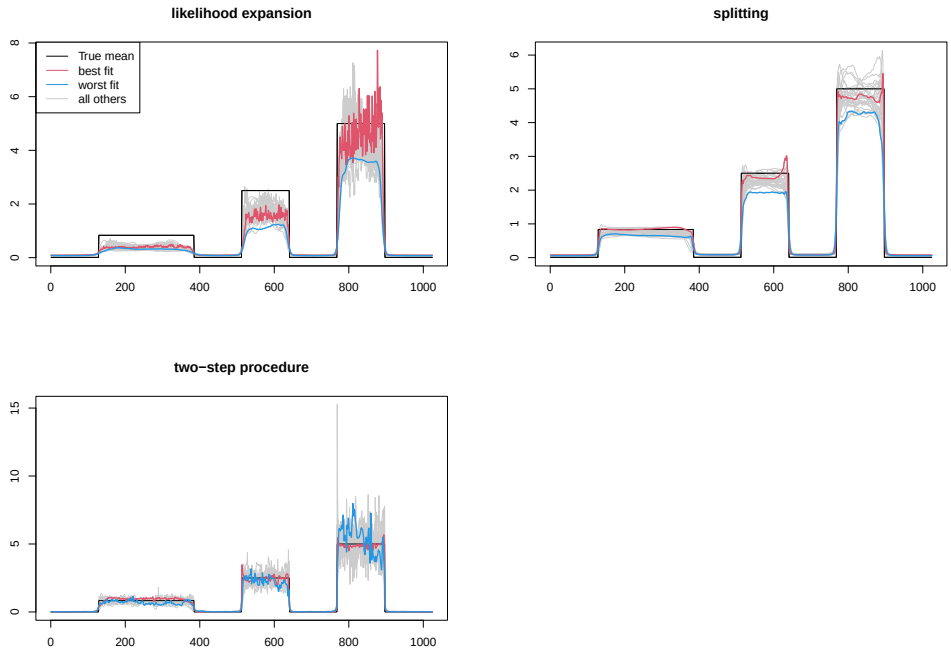


(a) max-mean-count size = 5

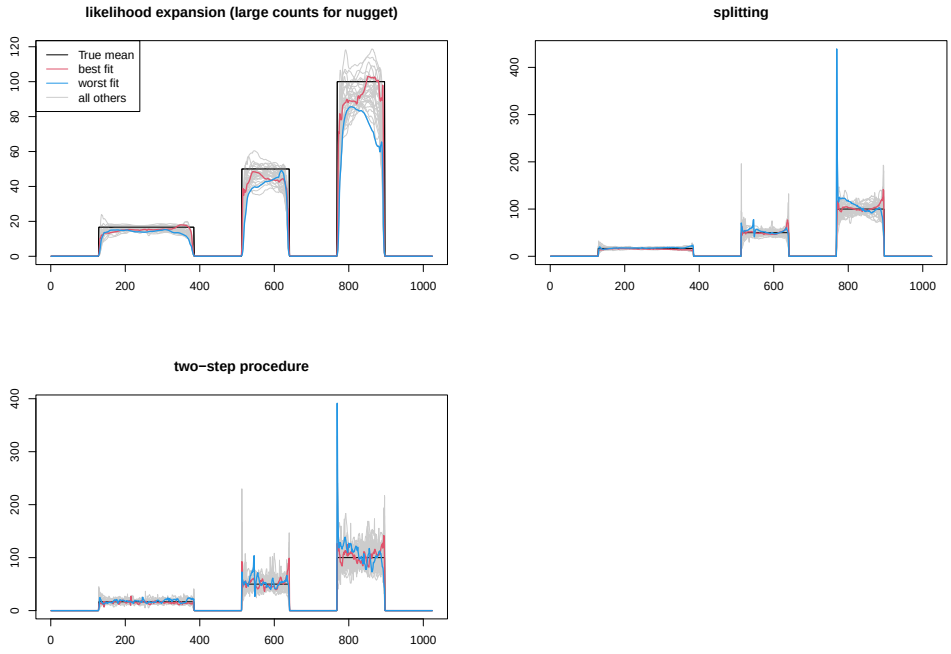


(b) max-mean-count size = 100

Figure 3.6: Plots of estimated spike function. SNR = 1. The black line the true mean, the red line is the fit with minimum RMSE, and the blue line is the fit with largest RMSE.



(a) max-mean-count size = 5



(b) max-mean-count size = 100

Figure 3.7: Plots of estimated simple block function. SNR = 1. The black line the true mean, the red line is the fit with minimum RMSE, and the blue line is the fit with largest RMSE.

3.7 Smooth sequencing data

In this section we apply splitting, likelihood expansion, two-step procedure, and VST to two sequencing count data – RNA-seq and ChIP-seq.

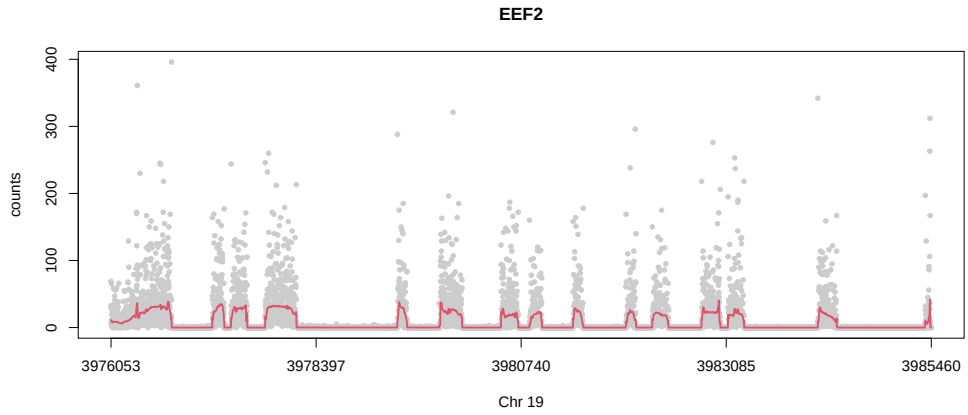
3.7.1 Smooth RNA-seq data from GTEx

We consider per base pair data for three genes from GTEx. Each sample is from an individual donor and a specific tissue. For the analysis in this section, we focus on samples collected from adipose tissue and donor SRR1069097. The three genes are *EEF2*, *FTH1* and *FTL*. They have high expressions and the max-count sizes are large, ranging from hundreds to thousands.

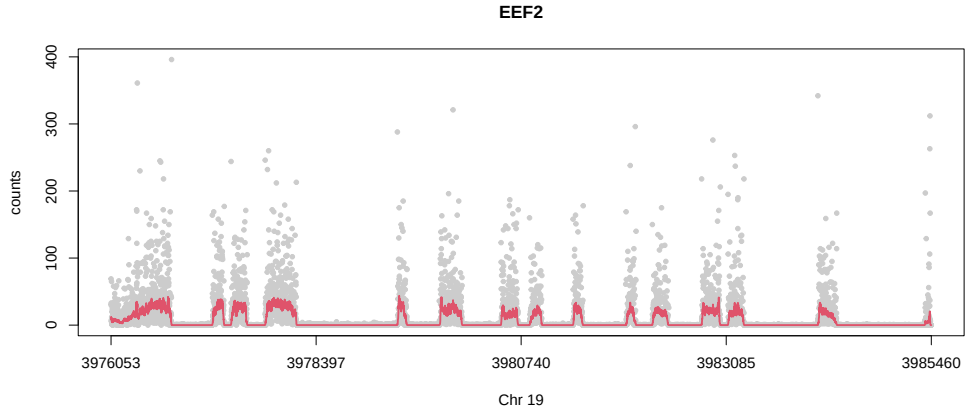
The gene *EEF2* is a protein coding gene on Chromosome 19 in humans and it has 15 exons. In Figure 3.8, we show the recovered expression level by three methods that can potentially work better for large counts. All the methods are able to capture the exon regions, and the splitting method gives the most visually appealing curve. The estimated nugget effect by splitting method is $\hat{\sigma}^2 = 1.11$. The likelihood expansion and two-step methods give slightly more ragged estimation than the splitting method, especially in the exon regions. Figure B.9 shows that the VST method has trouble removing the extra variation, so the estimated curve is not smooth at all. The results shown in Figure B.11 and B.10 suggest that for the protein coding genes *FTH1* (Chr11, 4 exons) and *FTL* (Chr19, 4 exons), the conclusions are similar, with the splitting method giving the most visually appealing recovered gene expression level.

3.7.2 Smooth ChIP-seq data

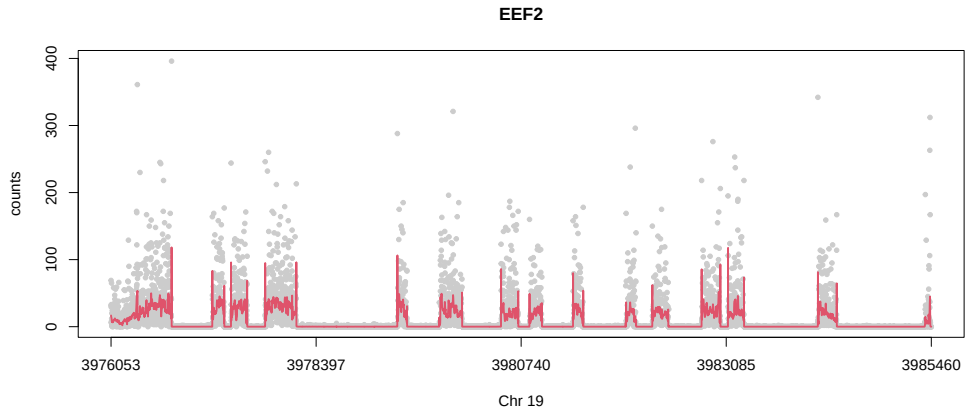
ChIP sequencing is a powerful method for identifying genome-wide transcription factors binding sites. The sites are usually identified by peak-calling methods. The read counts are



(a) Splitting method.



(b) Likelihood expansion method (use top 30% largest counts for nugget estimation).



(c) Two-step procedure (heteroskedastic variance).

Figure 3.8: Recovered expression level of gene *EEF2* by different smoothing methods.

a measure of the abundance of DNA fragments and can be used to infer the binding locations of the transcription factors. Typically, regions with higher read counts are considered to have a stronger binding signal, and are more likely to be functional elements involved in gene regulation or other cellular processes (Park [2009]). CTCF is a transcription factor that plays important roles in gene regulation and the three-dimensional organization of chromatin in three dimensions. The ChIP-seq data of CTCF from HeLa cells are from Broad Institute (`http://hgdownload.cse.ucsc.edu/goldenpath/hg19/encodeDCC/wgEncodeBroadHistone/`). The counts are much smaller than the RNA-seq counts and almost all of the counts are smaller than 10. The count is also very sparse - more than 80% of counts are 0. In this dataset, there are two biological replicates and each has data from forward and the reverse strands. We consider the CTCF binding site region Chr1: $110074895 - 110075320$, which is of length 425 bp and has been used as an example of MACE (Wang et al. [2014], `https://chipexo.sourceforge.net/`).

Figure 3.9 shows the smoothed ChIP-seq sequencing data of the forward strand. The splitting and VST methods give very similar estimation of the binding profile of CTCF. Clearly, there is an enriched region. The likelihood expansion method almost missed the enriched region, and can potentially lead to false negative. Both smash-Poisson and the two-step methods give less smoothed estimation than the splitting method.

3.8 Discussion

In this chapter, we studied the model (3.1) for smoothing count sequence with possible extra variation. Specifically we proposed a variance stabilizing transformation method that takes square root of the count data, and three methods for the log-link function model (3.1). We note that in model (3.1), any empirical Bayes nonparametric regression method can be applied. There is a rich literature on nonparametric regression, and some well-known meth-

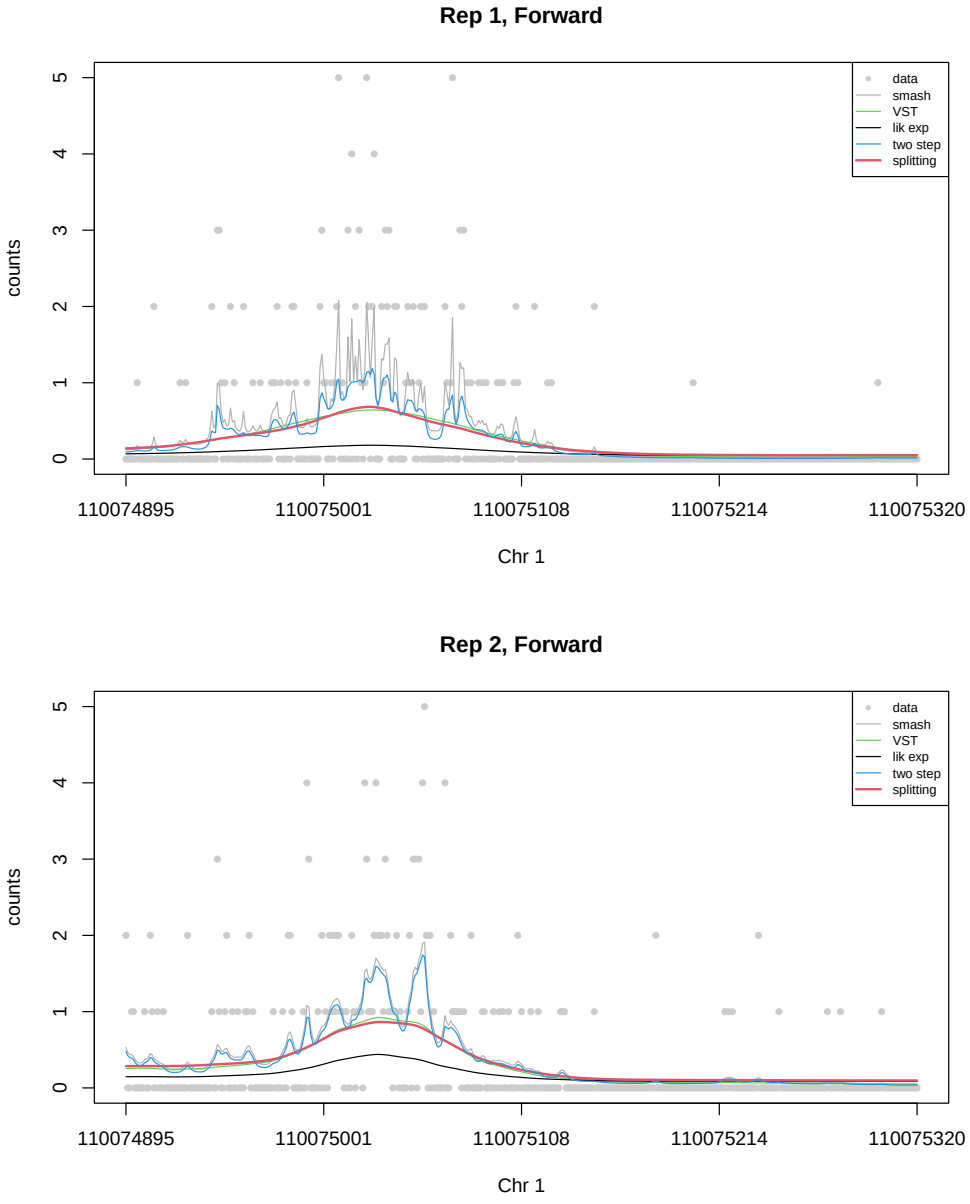


Figure 3.9: Smooth ChIP-seq data. Replicate 1 and 2, forward strand.

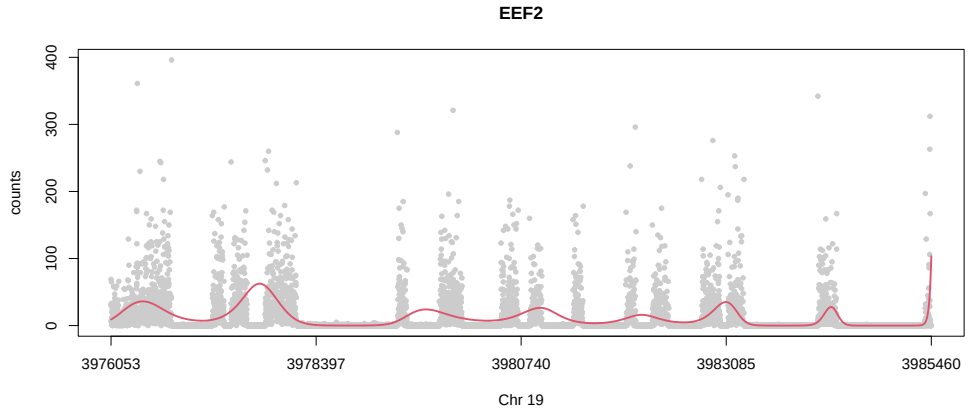
ods include local polynomials, kernels, splines, trend filtering and wavelets. Here we choose the empirical Bayes wavelet smoothing method because it is fast and spatially-adaptive (Donoho and Johnstone [1998]). Existing method for smoothing count data while accounting for over-dispersion usually takes a generalized additive model (GAM) approach (Hastie [2017], Stasinopoulos and Rigby [2008]) and usually uses non-locally-adaptive smoothing methods. Moreover, tuning the smoothing parameter requires extra efforts, especially the dimension of the basis used to represent the smooth function. This limits the application to the methods to sequencing data. As an illustration, we fit a penalized spline with negative binomial distribution using `gam` function from `R` package `mgcv` (Wood [2011]) to the gene EEF2 RNA-seq data. The results are shown in Figure 3.10.

The simulation study suggests that there's no one dominating method in all the settings, and we should probably select methods based on the count size and the variation level. We have shown that in general when the noise level is small and counts are small, simply applying smash-Poisson can give satisfactory results. In the ChIP-seq example, the VST and splitting method give very similar results. In numerical examples we have seen that the splitting method might miss small smooth spikes. For larger counts, the splitting method should be the preferred one, as shown in simulations and the RNA-seq examples. To some extent, the splitting method is a more rigorous solution for the log-link model (3.1), than the likelihood expansion and the two-step procedure. Because it has a clear objective function and is derived explicitly following a variational inference approach.

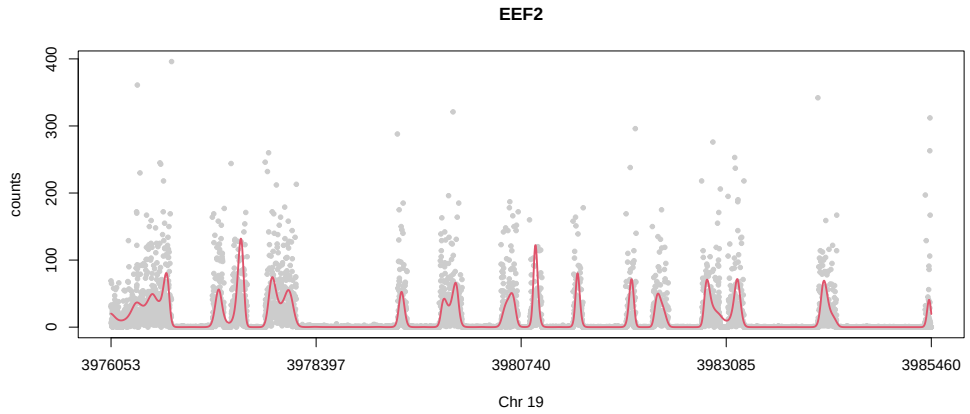
Code and data availability

All the Poisson smoothing methods have been implemented in the `R` package `smashrgen`, available at `https://github.com/DongyueXie/smashrgen`.

Code for running the simulations is at `https://github.com/DongyueXie/gsmash/blob/ma`



(a) $K = 20$



(b) $K = 100$

Figure 3.10: Recovered expression level of gene *EEF2* by fitting GAM. P-spline, negative Binomial distribution, and varying number (K) of basis functions.

in/code/poisson_smooth/simu_thesis.R, and for producing the simulation plots as well as real data analysis results is at https://github.com/DongyueXie/gsmash/blob/main/analysis/poisson_smoothing_benchmark.Rmd.

The GTEx data are available at https://github.com/DongyueXie/gsmash/tree/main/data/CoverageCounts, and the ChIP-seq data are at https://github.com/DongyueXie/smash-gen/tree/master/data/chipseq_examples.

CHAPTER 4

A SPLITTING VARIATIONAL INFERENCE APPROACH FOR NON-GAUSSIAN DATA

4.1 Introduction

Single cell RNA sequencing (scRNA-seq, Tang et al. [2009], Jovic et al. [2022], Brennecke et al. [2013], Cao et al. [2017], Klein et al. [2015]) is a technique used in genomics to analyze the gene expression profile of individual cells. In a typical statistical analysis of scRNA-seq, the starting point is a gene expression count matrix where rows correspond to cells and columns correspond to genes. Matrix factorization has been applied to learn the low-dimensional structure of the gene expression matrix (Stein-O'Brien et al. [2018], Feng et al. [2020], Xiang et al. [2021], Townes et al. [2019], Duren et al. [2018], DeBruine et al. [2021], Kotliar et al. [2019]). The low dimension structures are represented by two matrices, typically referred to as loadings and factors. The factors represent gene expression programs and loadings tell the cell membership in each gene expression program.

There are several existing methods for factorizing scRNA-seq matrices, which vary in their modelling assumptions regarding distributions, link functions, and constraints on the low-dimensional structure (Argelaguet et al. [2018], Carbonetto et al. [2023], Townes et al. [2019], Levitin et al. [2019], Sun et al. [2019], Risso et al. [2018], Lopez et al. [2018], Sarkar and Stephens [2021]). The two most commonly used distributions for scRNA-seq count data are Poisson and Negative Binomial. Both methods can naturally model count data, with the latter also accounting for over-dispersion. The Poisson measurement model is typically preferred for scRNA-seq data from a theoretical standpoint (Sarkar and Stephens [2021]). One of the most relevant methods to our interest is GLM-PCA (Townes et al. [2019]). The model extends PCA to the exponential family distribution and can perform dimension

reduction for count data directly. On the other hand, methods developed primarily for Gaussian and/or continuous data can also be applied. The current practice for applying a Gaussian model is to log-transform the counts after the addition of a pseudo-count to deal with zeros (Ahlmann-Eltze and Huber [2023], Amezquita et al. [2020], Love et al. [2014], Borella et al. [2022]).

Imposing constraints on loadings and factors can lead to more interpretable structures. For example, sparse matrix factorization assumes sparsity on loadings and/or factors, leading to a parsimonious representation (Wang and Stephens [2021], Witten et al. [2009], Zou et al. [2006]). Non-negative matrix factorization constrains the low dimensional structures to be non-negative, which can result in part-based decomposition (Lee and Seung [1999], Carbonetto et al. [2021]). Among all the existing methods, EBMF (Wang and Stephens [2021]) provides a flexible framework for imposing constraints on low dimensional structures. The method utilizes an empirical Bayes approach to learn the amount of shrinkage from data, requiring minimum tuning. Together with variational inference (Blei et al. [2017]), EBMF is modular and allows the addition of factors one-by-one, followed by backfitting for refinement. However, the method was originally developed for Gaussian likelihood, and extending it to a non-Gaussian distribution is non-trivial, especially for Poisson distribution with a log-link.

In this work, we propose a new variational inference method for non-Gaussian data. The method enables us to apply well-developed Gaussian empirical Bayes methods for inference on non-Gaussian models. The algorithm is modular, alternating between a step that handles the count data and a step that fits the Gaussian model. This results in a general empirical Bayes method for non-Gaussian factor analysis, allowing for various prior families on both loadings and factors. It can handle different assumptions about the latent structure, such as non-negativity and sparsity of factors.

This chapter is organized as follows. In the next section we introduce the general method and in section 3, we develop corresponding methods for Poisson and Binomial data. In section 4, we discuss the empirical Bayes Poisson matrix factorization model. And in the last two sections, numerical studies are performed to show the benefits of directly modelling the count using Poisson distribution, and how the new variational inference method gives flexible and accurate latent structure recovery.

4.2 Method

We start with a simple univariate model,

$$\begin{aligned} &y_i|\mu_i \sim \mathcal{D}(h(\mu_i)), i = 1, 2, ..., n, \\ &\boldsymbol{\mu} \sim f(\cdot), \end{aligned} \tag{4.1}$$

where $\mathcal{D}(\cdot)$ is a distribution in exponential family with canonical parameter μ_i and link function $h^{-1}(\cdot)$. Though the model is simple, it is very general and can include mean inference, regression and matrix factorization problems. Table 4.1 summarizes the different choices of $f(\cdot)$ that lead to different models. The empirical Bayes normal mean (EBNM) problem for false discovery rate control has been studied in Stephens [2017] (adaptive shrinkage, ash), and Willwerscheid and Stephens [2021] provides a thorough study of the EBNM problem with a wide range of priors. Kim et al. [2022] proposed Mr.ASH, a sparse regression model with ash prior, and Xing et al. [2021] extended the empirical Bayes selection of wavelet threshold (Johnstone and Silverman [2005]) to allow for the ash prior. Empirical Bayes matrix factorization (Wang and Stephens [2021]) is a flexible framework for sparse factor analysis. Urbut et al. [2019] proposed multivariate adaptive shrinkage (mash) by extending the ash prior to the multivariate normal mean problem.

A typical empirical Bayes procedure estimates the prior $f(\cdot)$ by maximizing the marginal

Table 4.1: Choices of prior in model 4.1.

Prior	Model	Software
$\mu_i \overset{i.i.d}{\sim} g(\cdot)$	univariate mean inference	`ashr,ebnm`
$\mu_i = \boldsymbol{x}_i^T\boldsymbol{\beta}, \boldsymbol{\beta} \sim g(\cdot)$	regression	`smashr, Mr.ASH`
$\mu_{ij} = \sum_k l_{ik}f_{jk}, l_k \sim g_{l_k}(\cdot), f_k \sim g_{f_k}(\cdot)$	matrix factorization	`flashr, flashier`
$\boldsymbol{\mu} \sim \sum_k \pi_k N(\boldsymbol{0}, \boldsymbol{U}_k)$	multivariate mean inference	`mashr`

likelihood $p(\boldsymbol{y}|f)$, then compute the posterior $p(\mu_i|y_i, \hat{f})$. All the empirical Bayes Gaussian methods in table 4.1 are well studied and the corresponding software are well developed, though they are not necessarily straightforward. For example, the EBMF model relies on variational inference to estimate the prior and compute the posterior. To extend these methods to non-Gaussian data, a natural idea is to take advantage of Gaussian likelihood-based methods. In general, if we could find a quadratic lower bound for the log-likelihood of a distribution, we would be able to perform inference of a non-Gaussian model by iteratively solving a Gaussian model. However, for Poisson distribution with log-link function, it is impossible to find such a quadratic lower-bound for the log-likelihood. On the other hand, from a modelling perspective, we prefer the log-link function because it gives multiplicative effects, and the gene expression programs (factors) can be roughly interpreted as log-fold changes.

Inspired by the splitting technique used in the alternating direction method of multipliers (ADMM, Boyd et al. [2011]), we introduce a latent splitting variable b, such that

$$\begin{aligned} y_i|\mu_i &\sim \mathcal{D}(h(\mu_i)), \\ \mu_i|b_i &\sim N(b_i, \sigma^2), \\ \boldsymbol{b} &\sim g(\cdot). \end{aligned} \tag{4.2}$$

For model (4.2), an empirical Bayes approach estimates the prior g and hyper-parameter σ^2 then compute posterior $p(\boldsymbol{\mu}, \boldsymbol{b}|\boldsymbol{y}, \hat{\sigma}^2, \hat{g})$. However both steps could be non-trivial depending

on the choice of prior g. To address this challenge, we adopt a variational approach and assume that the posterior factorizes as

$$\begin{aligned} q(\boldsymbol{\mu}, \boldsymbol{b}) &= \prod_i q_{\mu_i}(\mu_i) q_{\boldsymbol{b}}(\boldsymbol{b}), \\ q_{\mu_i} &\in \mathcal{Q}_\mu, \\ q_{\boldsymbol{b}} &\in \mathcal{Q}_b, \end{aligned} \tag{4.3}$$

where $\mathcal{Q}_\mu, \mathcal{Q}_b$ are pre-specified families of distributions. The evidence lower bound is

$$F(q, g; \sigma^2) = \sum_i \mathbb{E} \log \frac{p(y_i|h(\mu_i))}{q_{\mu_i}(\mu_i)} + \sum_i \mathbb{E} \log p(\mu_i|b_i, \sigma^2) + \mathbb{E} \log \frac{g(\boldsymbol{b})}{q_{\boldsymbol{b}}(\boldsymbol{b})}. \tag{4.4}$$

A coordinate-ascent algorithm iterates among the updates of $q_{\boldsymbol{\mu}}$, $\{q_{\boldsymbol{b}}, g(\cdot)\}$, and σ^2. Given $q_{\boldsymbol{b}}, g(\cdot), \sigma^2$, the objective function of $q_{\mu_i}, i = 1, 2, ..., n$ is

$$F(q_{\mu_i}) = \sum_i \mathbb{E} \log p(y_i|h(\mu_i)) + \sum_i \mathbb{E} \log p(\mu_i|\bar{b}_i, \sigma^2) - \sum_i \mathbb{E} \log q_{\mu_i}(\mu_i), \tag{4.5}$$

where $\bar{b}_i = \mathbb{E}_q(b_i)$, the posterior mean of b_i. Essentially, we are solving n independent variational Bayes mean inference problems, each with the likelihood $p(y_i|h(\mu_i))$, the prior $\mu_i \sim N(\bar{b}_i, \sigma^2)$, and the posterior q_{μ_i}. We will discuss this sub-problem in more details with a specific choice of $q_{\boldsymbol{\mu}_i}$. Given $q_{\boldsymbol{\mu}}, \sigma^2$, the objective function of $q_{\boldsymbol{b}}, g$ is

$$F(q_{\boldsymbol{b}}, g) = \sum_i \mathbb{E} \log N(\bar{\mu}_i; b_i, \sigma^2) + \mathbb{E} \log \frac{g(\boldsymbol{b})}{q_b(\boldsymbol{b})}, \tag{4.6}$$

where $\bar{\mu}_i = \mathbb{E}_q(\mu_i)$, the posterior mean of μ_i. The objective function corresponds to the evidence lower bound of a Gaussian model with observations $\bar{\mu}_i$, prior distribution $g(\boldsymbol{b})$, and posterior $q_{\boldsymbol{b}}$. Again different choices of prior $g(\cdot)$ lead to different models (see table 4.1).

Algorithm 4 Splitting Variational Inference for non-Gaussian data

1: **Input:** $\boldsymbol{y}$
2: **Init:** $q_{\boldsymbol{b}}, \sigma^2$
3: **repeat**
4: Given $q_{\boldsymbol{b}}$ and σ^2, update $q_{\mu_i}, i = 1, 2, ..., n$, by solving a variational Bayes Poisson mean (VBPM) problem (4.5) with prior mean $\bar{b}_i$ and prior variance σ^2
5: Given $q_{\mu_i}, i = 1, 2, ..., n$ and σ^2, update $q_{\boldsymbol{b}}, g(\cdot)$ by solving an empirical Bayes normal (EBN) model (4.6)
6: Given $q_{\boldsymbol{\mu}}$ and $q_{\boldsymbol{b}}$, update σ^2 as $\sigma^2 = \sum_i \mathbb{E}_q(\mu_i - b_i)^2/n$
7: **until** Converged
8: **Output:** Estimated prior $\hat{g}$, variance $\hat{\sigma}^2$ and fitted posteriors $q_{\boldsymbol{b}}, q_{\boldsymbol{\mu}}$.

The Gaussian distribution $\mu_i|b_i \sim N(b_i, \sigma^2)$ acts as a bridge between two parts of the algorithm 4: a variational Bayes mean problem and an empirical Bayes model for Gaussian data. Each sub-problem is simpler than the original problem, but together they solve a more complicated one.

Remark. *The conditional distribution of μ given b can be interpreted in two ways. Firstly, we introduced it as a "device" when comparing it to model (4.1) for algorithm development. Alternatively, we can regard σ^2 as an over-dispersion parameter, which is particularly useful in count data analysis. When y follows a Poisson distribution, model (4.2) becomes a Poisson-LogNormal (PLN, Aitchison and Ho [1989], Chiquet et al. [2021]) model. In the PLN model, $y|\mu \sim \text{Poisson}(\exp \mu), \mu \sim N(b, \sigma^2)$, the expectation of y is $\mathbb{E}\, y = \exp(b + \sigma^2/2)$, and the variance is $\text{Var}(y) = \mathbb{E}\, y(1 + (\exp(\sigma^2) - 1)\, \mathbb{E}\, y)$. Clearly the variance is larger than the expectation. as long as σ^2 is not 0. In either case, we refer to this approach as a splitting approach because by introducing the latent variable b, we split the objective function into two parts. The algorithm then iterates between solving two simpler problems. The splitting technique is also widely used in other methods such as ADMM and proximal algorithms (Polson et al. [2015]).*

Obviously, when the variance σ^2 is 0, the splitting model (4.2) is equivalent to the original

model (4.1). In the splitting algorithm the update of σ^2 is given by the average of the second moment of $\mu_i - b_i$. When σ^2 converges to 0 in practice, the following Lemma 4.2.1 gives the posterior distributions and optimal g at convergence.

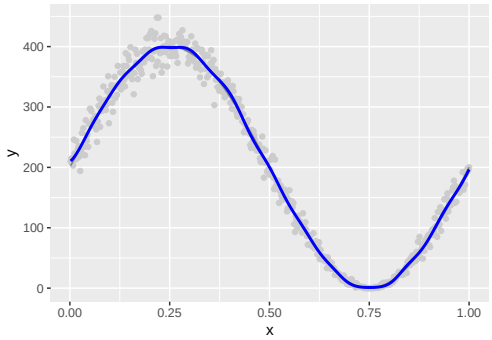
Lemma 4.2.1. *In the Algorithm 4, $\sigma^2 \to 0$ if and only if $q_{\mu_i} \to \delta_{\bar{\theta}_i}$, $q_{b_i} \to \delta_{\bar{\theta}_i}$ for all i, where q_{b_i} is the marginal posterior density of b_i and $\bar{\theta}_i$ is the posterior mean of μ_i and b_i. When $\sigma^2 \to 0$, $\hat{g}$ is the maximum likelihood estimator (MLE) derived from $\bar{\theta}_i$, for $i = 1, 2, ..., n$, i.e. $\hat{g} = \arg\max_{g \in \mathcal{G}} \log g(\bar{\theta}_i)$.*

Proof. If $\sigma^2 \to 0$, then $\mathbb{E}(\mu_i - b_i)^2 \to 0$ for each i. Since μ_i and b_i are independent a posteriori, we have $\mathbb{E}(\mu_i - b_i)^2 = \mathrm{Var}(\mu_i) + \mathrm{Var}(b_i) + (\mathbb{E}\,\mu_i - \mathbb{E}\,b_i)^2$. Given that $\mathbb{E}(\mu_i - b_i)^2 \to 0$, each term on the right side of the equation must also converge to 0. This implies that $\mathrm{Var}(\mu_i)$ and $\mathrm{Var}(b_i)$ both converge to 0, and $\mathbb{E}\,\mu_i$ converges to $\mathbb{E}\,b_i$. Thus, q_{μ_i} and q_{b_i} converge to the same point-mass $\delta_{\bar{\theta}_i}$, and the point-mass is the posterior mean. The "if" direction is also obvious. Since the prior $g(\cdot)$ is estimated by maximizing marginal log-likelihood under a normal likelihood, and the variance $\sigma^2 \to 0$, the $\hat{g}$ is the MLE derived from $\bar{\theta}_i$. □

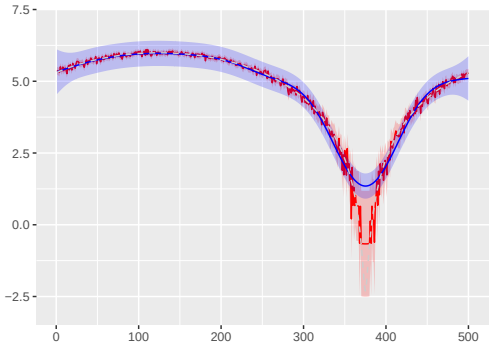
We illustrate the behaviour of q_{μ_i} and q_{b_i} when $\sigma^2 \to 0$ in a Poisson multivariate mean example. Consider the model

$$\begin{aligned} y_i &\sim \mathrm{Poisson}(\exp(b_i)), \\ \boldsymbol{b} &\sim N(\theta \mathbf{1}, \Sigma(\tau^2, l)), \end{aligned} \tag{4.7}$$

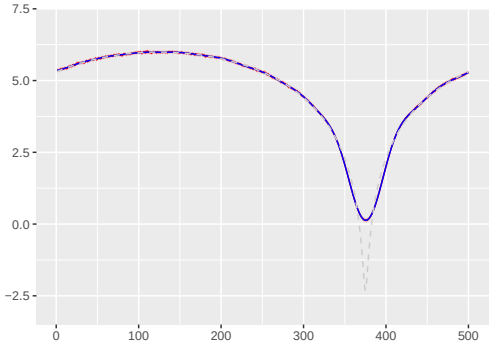
where θ is the unknown prior mean, and $\Sigma(\tau^2, l)$ is the square exponential kernel covariance matrix defined as $\Sigma_{ij} = \tau^2 \exp(-(x_i - x_j)^2/(2l^2))$. The x_i in the kernel matrix is the location of y_i and we assume observations are on grids with equal spacing, ranging from 0 to 1. The τ^2 and l are unknown parameters that control the smoothness of the curve $\boldsymbol{b}$. In this simulation example, we set $n = 500$, $x_i = i/n$, and $\exp(\boldsymbol{b}) = (\sin(2\pi \boldsymbol{x}) - \min(\sin(2\pi \boldsymbol{x}))) \times c + 0.1$. The



(a) Fitted Poisson Gaussian process. The blue line is the recovered curve, the posterior mean of $\exp(\boldsymbol{b})$.



(b) Point-wise 95% posterior credible interval of μ_i, b_i at iteration 1.



(c) Point-wise 95% posterior credible interval of μ_i, b_i at convergence.

Figure 4.1: Splitting variational inference on Poisson Gaussian process. Figure (a) shows the fitted curve. Figure (b) and (c) show the posterior mean and variance of μ_i, b_i at iteration 1 and at convergence respectively. The blue region corresponds to q_{b_i}, and the red region corresponds to q_{μ_i}. The dashed grey line is the true $\boldsymbol{b}$ for simulating the data.

constant c is for controlling the size of the maximum count and we set $c = 200$ such that the maximum mean count is around 400. Note that the minimum mean count is 0.1.

Wt fit the splitting algorithm to the observations $\boldsymbol{y}$, and initialize each $\bar{b}_i$ to be $\log(\boldsymbol{y}^T\mathbf{1}/n)$. Figure 4.1a shows that the splitting algorithm successfully recovers the underlying smooth signal. The initial value of σ^2 is 2.56, and Figure 4.1b shows the point-wise posterior 95% credible interval of μ_i, b_i after the first iteration. Apparently, at the very first iteration, the posterior of μ_i and b_i vary substantially. The σ^2 eventually converges to 0.0014 and Figure 4.1c shows the point-wise posterior 95% credible interval at convergence, when the the posterior distributions of both μ_i, b_i concentrate around a single point.

4.2.1 The objective function of q_b

When σ^2 is greater than 0, there are two equivalent views of the splitting model. In the first case, if we keep the likelihood $p(y_i|\mu_i)$ and integrate out the variable $\boldsymbol{b}$, we obtain an induced prior on $\boldsymbol{\mu}$ as $p(\boldsymbol{\mu}; g, \sigma^2) = \int p(\boldsymbol{\mu}|\boldsymbol{b}, \sigma^2)g(\boldsymbol{b})d\boldsymbol{b}$. On the other hand, if we keep the prior $\boldsymbol{b} \sim g(\cdot)$ and integrate out the variable $\boldsymbol{\mu}$, we obtain an induced likelihood on $\boldsymbol{y}$. This is particularly useful when we are dealing with over-dispersed count data, and the parameter σ^2 can account for the extra variation in the data. In this section, we develop theory on the connection between the mean field posterior $q(\boldsymbol{b}, \boldsymbol{\mu}) = q_{\boldsymbol{b}}q_{\boldsymbol{\mu}}$ in the splitting model and the posterior we would obtain if we directly solve the induced models.

We consider the induced model by integrating out the variable μ. WLOG, we focus on univariate case and drop the subscript i. The marginal distribution of y, by marginalizing out μ is

$$p(y|b, \sigma^2) = \int p(y|\mu)N(\mu; b, \sigma^2)d\mu. \tag{4.8}$$

One of the primary goal is to estimate $g(\cdot)$ and compute posterior of b. The most straightforward approach is to directly work with the model where y follows a distribution with

density $p(y|b,\sigma^2)$, and the prior on b is $g(\cdot)$. However as we have mentioned this is in general non-trivial. The splitting approach together with the mean-field approach provide a much easier model fitting procedure. Let the posterior of b be q_b, then the ELBO in the induced model is

$$\tilde{F}(q_b, g; \sigma^2) = \mathbb{E} \log p(y|b,\sigma^2) + \mathbb{E} \log \frac{g(b)}{q_b}. \tag{4.9}$$

On the other hand, the profiled ELBO for q_b, g, obtained by maxing q_μ out in (4.4) is defined as

$$F(q_b, g; \sigma^2) = \max_{q_\mu} F(q_\mu, q_b; g, \sigma^2). \tag{4.10}$$

We show in Theorem 4.2.3 that the profiled ELBO is a lower bound of the marginal ELBO (4.9). Before introducing the theorem, we first present the following lemma.

Lemma 4.2.2. *The second order derivative of* $\log p(y|b,\sigma^2) = \log \int p(y|\mu) N(\mu; b, \sigma^2) d\mu$ *with respect to* b *is lower bounded by* $-1/\sigma^2$.

Proof. Denote $f(b) := p(y|b,\sigma^2)$, the second derivative of $\log f(b)$ is

$$\frac{d^2 \log f(b)}{db^2} = \frac{f''(b)}{f(b)} - \left(\frac{f'(b)}{f(b)}\right)^2, \tag{4.11}$$

where

$$\begin{aligned} f'(b) &= \frac{1}{\sigma^2} f(b) \int p(\mu|y,b,\sigma^2) \mu d\mu - \frac{b}{\sigma^2} f(b) = \frac{1}{\sigma^2} f(b)(\mathbb{E}\,\mu - b), \\ f''(b) &= \frac{1}{(\sigma^2)^2} f(b)(\mathbb{E}\,\mu^2 - b\,\mathbb{E}\,\mu) - \frac{1}{\sigma^2} f(b) - \frac{b}{\sigma^2} f'(b) \\ &= f(b) \left(\frac{1}{(\sigma^2)^2} \mathbb{E}\,\mu^2 - \frac{2b}{(\sigma^2)^2} \mathbb{E}\,\mu - \frac{1}{\sigma^2} + \frac{b^2}{(\sigma^2)^2} \right), \end{aligned} \tag{4.12}$$

and the expectation is under $p(\mu|y, b, \sigma^2)$. Then

$$\begin{aligned}\frac{d^2 \log f(b)}{db^2} &= \left(\frac{1}{(\sigma^2)^2}\,\mathbb{E}\,\mu^2 - \frac{2b}{(\sigma^2)^2}\,\mathbb{E}\,\mu - \frac{1}{\sigma^2} + \frac{b^2}{(\sigma^2)^2}\right) - \frac{1}{(\sigma^2)^2}((\mathbb{E}\,\mu)^2 - 2b\,\mathbb{E}\,\mu + b^2) \\ &= -\frac{1}{\sigma^2} + \frac{1}{(\sigma^2)^2}(\mathbb{E}\,\mu^2 - (\mathbb{E}\,\mu)^2) \\ &\geq -\frac{1}{\sigma^2}.\end{aligned} \tag{4.13}$$

□

Theorem 4.2.3. *The profiled ELBO $F(q_b, g; \sigma^2)$ is a lower bound of the marginal ELBO $\tilde{F}(q_b, g; \sigma^2)$.*

Proof. The ELBO $F(q_\mu, q_b; g, \sigma^2)$ for the splitting model (4.2) is

$$F(q_\mu, q_b; g, \sigma^2) = \mathbb{E} \log p(y|\mu) + \mathbb{E} \log \frac{N(\mu; \bar{b}, \sigma^2)}{q_\mu} + \mathbb{E} \log \frac{g(b)}{q_b} - \frac{V_{q_b}}{2\sigma^2}, \tag{4.14}$$

where $\bar{b} = \mathbb{E}\,b$ and $V_{q_b} = \mathbb{E}(b - \bar{b})$. Then the profiled ELBO for q_b, g is

$$\begin{aligned}F(q_b, g; \sigma^2) &= \max_{q_\mu} F(q_\mu, q_b, g; \sigma^2) \\ &= \log p(y|\bar{b}, \sigma^2) + \mathbb{E} \log \frac{g(b)}{q_b} - \frac{V_{q_b}}{2\sigma^2}.\end{aligned} \tag{4.15}$$

The ELBO $F(q_\mu, q_b, g; \sigma^2)$ reaches its maximum over q_μ at $q_\mu^* = p(\mu|y, \bar{b}, \sigma^2)$. For the marginal ELBO $\tilde{F}(q_b, g; \sigma^2)$, a second order Taylor series expansion of $\log p(y|b, \sigma^2)$ around

$\bar{b}$ gives

$$\begin{aligned}
\tilde{F}(q_b, g; \sigma^2) &= \mathbb{E} \log p(y|b, \sigma^2) + \mathbb{E} \log \frac{g(b)}{q_b} \\
&= \log p(y|\bar{b}, \sigma^2) + \frac{1}{2} \left(\frac{d^2 \log p(y|b, \sigma^2)}{db^2} \right) \Big|_{b=\tilde{b}} V_{q_b} + \mathbb{E} \log \frac{g(b)}{q_b} \\
&\geq \log p(y|\bar{b}, \sigma^2) - \frac{1}{2\sigma^2} V_{q_b} + \mathbb{E} \log \frac{g(b)}{q_b} \\
&= F(q_b, g; \sigma^2).
\end{aligned} \tag{4.16}$$

where $\tilde{b}$ is between $\bar{b}$ and b. The inequality is due to Lemma 4.2.2. □

The following lemma connects the maximization of $F(q_b, b; \sigma^2)$ with the penalty-based formulation of empirical Bayes problem introduced in Section 2.4.1.

Lemma 4.2.4. *Let*

$$\bar{F}(\bar{b}, g; \sigma^2) = \max_{q: \mathbb{E}_q = \bar{b}} \left(\log p(y| \mathbb{E}_q b, \sigma^2) - (\frac{1}{2\sigma^2} V_q + KL(q_b || g)) \right),$$

then it can be written in penalty-based form as

$$\bar{F}(\bar{b}, g; \sigma^2) = \log p(y|\bar{b}, \sigma^2) - r_g(\bar{b}; \sigma^2),$$

where $r_g(\bar{b}; \sigma^2)$ *is defined in* (2.20).

Proof. The maximization of objective function $F(q_b, g; \sigma^2)$ can be written as

$$\begin{aligned}
&\max_{q_b, g} F(q_b, g; \sigma^2) \\
=&\max_{\bar{b}, g} \max_{q: \mathbb{E}_q = \bar{b}} \left(\log p(y | \, \mathbb{E}_q \, b, \sigma^2) - (\frac{1}{2\sigma^2} V_q + KL(q_b || g)) \right) \\
=&\max_{\bar{b}, g} \left(\log p(y | \bar{b}, \sigma^2) - \min_{q: \mathbb{E}_q = \bar{b}} \left(\frac{1}{2\sigma^2} V_q + KL(q_b || g) \right) \right) \\
:=&\max_{\bar{b}, g} \bar{F}(\bar{b}, g; \sigma^2).
\end{aligned} \tag{4.17}$$

Recall the term $r_g(\bar{b}; \sigma^2) = \min_{q: \mathbb{E}_q = \bar{b}} \left(\frac{1}{2\sigma^2} V_q + KL(q_b || g) \right)$ is the normal mean penalty form and is evaluated as

$$\begin{aligned}
r_g(\bar{b}; \sigma^2) &= -\log p(z(\bar{b}; g, \sigma^2); g, \sigma^2) - \frac{(z(\bar{b}; g, \sigma^2) - \bar{b})^2}{2\sigma^2} - \frac{1}{2} \log 2\pi\sigma^2 \\
&= -\log p(z(\bar{b}; g, \sigma^2); g, \sigma^2) + \log N(z(\bar{b}; g, \sigma^2); \bar{b}, \sigma^2).
\end{aligned} \tag{4.18}$$

where $\bar{b} = S_{g,\sigma^2}(z)$ and $S_{g,\sigma^2}(\cdot)$ is the posterior mean operator in a normal mean problem, $z(\bar{b}; g, \sigma^2) = S^{-1}_{g,\sigma^2}(\bar{b})$, and $p(z; g, \sigma^2) = \int g(b) N(z; b, \sigma^2) db$. Thus the new objective function over $\bar{b}, g$ is

$$\begin{aligned}
\bar{F}(\bar{b}, g; \sigma^2) &= \log p(y | \bar{b}, \sigma^2) - r_g(\bar{b}; \sigma^2) \\
&= \log p(y | \bar{b}, \sigma^2) - \log p(z(\bar{b}; g, \sigma^2); g, \sigma^2) + \log N(z(\bar{b}; g, \sigma^2); \bar{b}, \sigma^2).
\end{aligned} \tag{4.19}$$

□

To study the behaviour of q_b when $\sigma^2 \to 0$, we consider the profiled ELBO of q_b, g, σ^2 in (4.15). Unless q_b is a point mass ($V_{q_b} = 0$), the ELBO $F(q_b, g, \sigma^2)$ goes to $-\infty$ as $\sigma^2 \to 0$, because of the term $-\frac{V_{q_b}}{\sigma^2}$. So the ELBO will not be optimized at $\sigma^2 = 0$ unless it is also optimized at $V_{q_b} = 0$. If q_b is a point mass at $\bar{b}$ (WLOG, let $\bar{b} = 0$), then the ELBO is $-\infty$

unless g also has some mass at 0. In general, if q_b is a point mass, the σ^2 will be optimized at:

$$\arg\max_{\sigma^2} \log p(y|0, \sigma^2). \tag{4.20}$$

To get a clearer understanding of the above argument. Let's consider a simple normal model, for $i = 1, 2, ..., n$,

$$\begin{aligned} y_i|\mu_i &\sim N(\mu_i, s^2), \\ \mu_i|b_i &\sim N(b_i, \sigma^2), \\ b_i &\overset{i.i.d}{\sim} N(0, \phi^2), \end{aligned} \tag{4.21}$$

where s^2 is known, and σ^2, ϕ^2 are unknown. The profiled ELBO for $q_{b_i}, \phi^2, \sigma^2$ is

$$\begin{aligned} F(q_{\boldsymbol{b}}, \phi^2, \sigma^2) &= \sum_i \mathbb{E} \log N(y_i; \bar{b}_i, s^2 + \sigma^2) + \sum_i \mathbb{E} \log \frac{N(b_i; 0, \phi^2)}{N(b_i; \bar{b}_i, V_{b_i})} - \frac{1}{2\sigma^2} \sum_i V_{b_i} \\ &= -\frac{n}{2} \log(s^2 + \sigma^2) - \frac{1}{2(s^2 + \sigma^2)} \sum_i (y_i - \bar{b}_i)^2 \\ &\quad - \frac{n}{2} \log \phi^2 - \frac{1}{2\phi^2} \sum_i (\bar{b}_i^2 + V_{b_i}) + \frac{1}{2} \sum_i \log V_{b_i} - \frac{1}{2\sigma^2} \sum_i V_{b_i}. \end{aligned} \tag{4.22}$$

The optimal $\bar{b}_i, V_{b_i}, \phi^2$ obtained by setting the corresponding partial derivatives to 0 are

$$\bar{b}_i = \frac{\phi^2}{\phi^2 + s^2 + \sigma^2} y_i, \tag{4.23}$$

$$V_{b_i} = \frac{\phi^2 \sigma^2}{\phi^2 + \sigma^2}, \tag{4.24}$$

$$\phi^2 = \frac{1}{n} \sum_i (\bar{b}_i^2 + V_{b_i}). \tag{4.25}$$

The optimal σ^2 is the solution to the following root-finding problem:

$$-\frac{n}{s^2+\sigma^2}+\frac{1}{(s^2+\sigma^2)^2}\sum_i(y_i-\bar{b}_i)^2+\frac{1}{(\sigma^2)^2}\sum_i V_{b_i}=0. \tag{4.26}$$

Thus, σ^2 is optimized at 0 if the following two conditions are true: $V_{b_i}=0$ for all i, and $\frac{1}{n}\sum_i(y_i-\bar{b}_i)^2-s^2\leq 0$. The first condition is due to the last term in the LFS of (4.26), and the second condition is obtained by solving for σ^2 after setting $V_{b_i}=0$.

Furthermore, if σ^2 is optimized at 0, then the $\phi^2=0$ which is obtained by solving for ϕ^2 in (4.25), and the fact that any $\phi^2>0$ leads to the ELBO being $-\infty$. Thus we have $\bar{b}_i=0, V_{b_i}=0, \phi^2=0$ and $\frac{1}{n}\sum_i y_i^2\leq s^2$ if σ^2 is optimized at 0.

4.3 Variational Gaussian posterior approximation

In this section we study the first sub-problem in the splitting algorithm 4 - variational Gaussian posterior approximation (VGA) for Poisson and Binomial distribution. We have not specified the form of q_{μ_i}, and in fact it can be flexible. In this work, we choose a Gaussian distribution for q_{μ_i}, specifically $q_{\mu_i}=N(\mu_i;\bar{\mu}_i,v_i)$, due to the Gaussian prior on μ_i (given b_i) and the computational simplifications this choice offers. While the updates for $\bar{\mu}_i$ and v_i do not typically have explicit solutions with the chosen q_{μ_i}, we will demonstrate here that they can be efficiently optimized using convex optimization algorithms.

The VGA model is defined as

$$\begin{aligned} y|\mu &\sim \mathcal{D}(h(\mu)),\\ \mu &\sim N(\bar{b},\sigma^2), \end{aligned} \tag{4.27}$$

where $\bar{b}$ and σ^2 are known. The posterior distribution of μ is assumed to follow a Gaussian

distribution $q_\mu = N(\mu; \bar{\mu}, v)$. The aim is to perform variational inference on the posterior distribution q_μ. As we solve n independent variational Gaussian approximation (VGA) problems in the splitting variational inference, we omit the subscript i in this section and focus on the univariate case.

4.3.1 Poisson distribution

We study the posterior inference of μ in the model

$$\begin{aligned} y|\mu &\sim \text{Poisson}(s\exp(\mu)), \\ \mu &\sim N(\bar{b}, \sigma^2), \end{aligned} \tag{4.28}$$

where $s > 0$ is a known scaling scalar. The posterior mean and variance can be obtained by solving the following optimization problem:

$$\begin{aligned} (\bar{\mu}^*, v^*) &= \arg\max_{\bar{\mu}, v} F(\bar{\mu}, v) \\ &= \mathbb{E}_{q_\mu} \log p(y, \mu) - \mathbb{E}_{q_\mu} \log q_\mu \\ &= \mathbb{E}_{q_\mu} \log p(y|\mu) + \mathbb{E}_{q_\mu} \log N(\mu; \bar{b}, \sigma^2) - \mathbb{E}_{q_\mu} \log N(\mu; \bar{\mu}, v) \\ &= \mathbb{E}_{q_\mu}(y\mu - se^\mu) - \mathbb{E}_{q_\mu} \frac{(\mu - \bar{b})^2}{2\sigma^2} - \mathbb{E}_{q_\mu}(-\frac{1}{2}\log v - \frac{(\mu - \bar{\mu})^2}{2v}) + \text{const} \\ &= y\bar{\mu} - se^{\bar{\mu} + v/2} - \frac{\bar{\mu}^2 + v - 2\bar{\mu}\bar{b}}{2\sigma^2} + \frac{1}{2}\log v + \text{const}, \end{aligned} \tag{4.29}$$

where we have used $\mathbb{E}\, e^\mu = e^{\bar{\mu} + v/2}$, which is the mean of the log-Normal distribution. Solving the optimization problem turns out to be simple because $F(\bar{\mu}, v)$ is a concave function (as shown in Lemma 4.3.1). Therefore, the general results and algorithms for convex optimization can be applied. For example, one can use Newton's method or quasi-Newton method to solve the optimization problem.

Lemma 4.3.1. *Minimizing $-F(\bar{\mu}, v)$ is a convex problem.*

Proof. The gradients of function (4.29)(omitting s) are

$$\begin{aligned}\frac{\partial F}{\partial \bar{\mu}} &= y - e^{\bar{\mu}+v/2} - (\bar{\mu} - \bar{b})/\sigma^2, \\ \frac{\partial F}{\partial v} &= -\frac{1}{2}e^{\bar{\mu}+v/2} - \frac{1}{2\sigma^2} + \frac{1}{2v},\end{aligned} \tag{4.30}$$

and $\boldsymbol{H}$ is the 2 by 2 Hessian matrix with elements

$$\begin{aligned}\frac{\partial^2 F}{\partial^2 \bar{\mu}} &= -e^{\bar{\mu}+v/2} - 1/\sigma^2, \\ \frac{\partial^2 F}{\partial \bar{\mu} \partial v} &= -\frac{1}{2}e^{\bar{\mu}+v/2}, \\ \frac{\partial^2 F}{\partial^2 v} &= -\frac{1}{2}e^{\bar{\mu}+v/2} - \frac{1}{2v^2}.\end{aligned} \tag{4.31}$$

The Hessian matrix $\boldsymbol{H}$ is negative definite because it is strictly diagonally dominant and the diagonal elements are negative. □

Meanwhile, according to the following lemma, it is possible to express the optimal $\bar{\mu}$ as a function of optimal v (and vice versa), so that the optimization problem can be reduced to a univariate problem.

Lemma 4.3.2. *Let* $(\bar{\mu}^*, v^*) = \arg\max_{\bar{\mu},v} F(\bar{\mu}, v)$*, then the following holds,*

$$\begin{aligned}\bar{\mu}^* &= \sigma^2 y + \bar{b} + 1 - \frac{\sigma^2}{v^*}, \\ v^* &= \frac{\sigma^2}{\sigma^2 y - \bar{\mu}^* + \bar{b} + 1}.\end{aligned} \tag{4.32}$$

Proof. Setting both gradients in 4.30 to 0 and taking their difference, we have

$$y - \frac{\bar{\mu}^* - \bar{b}}{\sigma^2} + \frac{1}{\sigma^2} - \frac{1}{v}^* = 0. \tag{4.33}$$

□

Based on Lemma 4.3.2, we can reduce the two-dimensional optimization problem (4.29) to a univariate one. Let $v = \frac{\sigma^2}{\sigma^2 y - \bar{\mu} + \bar{b} + 1}$ then the root finding equation for $\bar{\mu}$ is

$$h(\bar{\mu}) = y - se^{\bar{\mu} + \frac{1}{2}\frac{\sigma^2}{\sigma^2 y - \bar{\mu} + \bar{b} + 1}} - \frac{\bar{\mu}}{\sigma^2} + \frac{\bar{b}}{\sigma^2} = 0. \tag{4.34}$$

Lemma 4.3.3. *The root of $h(\bar{\mu}) = 0$ exists and is unique. Moreover $\bar{\mu}^* < \bar{b} + \sigma^2 y$.*

Proof. Since the exponential term is positive, an upper bound on $\bar{\mu}^*$ is $\bar{\mu}^* < \bar{b} + \sigma^2 y$. The function $h(\bar{\mu})$ is a monotonically decreasing function and $\lim_{\bar{\mu} \to -\infty} h(\bar{\mu}) \to +\infty$, $h(\bar{b} + \sigma^2 y) < 0$, so there's a guaranteed solution to the root finding problem. □

Similarly, the root finding equation for v is

$$h_v(v) = \frac{1}{v} - \frac{1}{\sigma^2} - se^{\sigma^2 y + \bar{b} + 1 - \frac{\sigma^2}{v} + \frac{v}{2}} = 0. \tag{4.35}$$

Lemma 4.3.4. *The root of $h_v(v) = 0$ exists and is unique. Moreover $v^* < \sigma^2$.*

Proof. Since the exponential term is positive, an upper bound on v^* is $v^* < \sigma^2$. The function $h_v(v)$ is a monotonically decreasing function and $h_v(0) = +\infty$, $h_v(\sigma^2) < 0$, so there's a guaranteed solution to the root finding problem. □

Although the original optimization problem (4.29) is bivariate, in practice we can solve for either $\bar{\mu}^*$ or v^* using univariate optimization solver, then obtain another one according to Lemma 4.3.2. We have found that Newton's method is more suitable for solving $h(\bar{\mu}) = 0$ due to its fast convergence and the explicit form of h'. On the other hand, the bisection

method is more reliable for solving $h_v(v) = 0$ because the natural interval for searching the root is $(0, \sigma^2)$.

Remark. *When solving $h(\bar{\mu}) = 0$, to avoid the large exponential term, we take log of both sides,*

$$\begin{aligned} & y - \frac{\bar{\mu}}{\sigma^2} + \frac{\bar{b}}{\sigma^2} = s e^{\bar{\mu} + \frac{1}{2}\frac{\sigma^2}{\sigma^2 y - \bar{\mu} + \bar{b} + 1}} \\ \iff & \tilde{h}(\bar{\mu}) = \bar{\mu} + \frac{\sigma^2}{2(\sigma^2 y + \bar{b} - \bar{\mu} + 1)} - \log(\sigma^2 y + \bar{b} - \bar{\mu}) + \log s\sigma^2 = 0 \end{aligned} \tag{4.36}$$

The derivative of $\tilde{h}(\bar{\mu})$ is

$$\tilde{h}'(\bar{\mu}) = 1 + \frac{\sigma^2}{2(\sigma^2 y + \bar{b} - \bar{\mu} + 1)^2} + \frac{1}{\sigma^2 y + \bar{b} - \bar{\mu}}. \tag{4.37}$$

4.3.2 Binomial distribution

We consider the sub-problem of VGA of Binomial data, with binary data as a special case. The model is

$$\begin{aligned} & y|\mu \sim \text{Binom}(n, \sigma(\mu)), \\ & \mu \sim N(\bar{b}, \sigma^2), \end{aligned} \tag{4.38}$$

where $\sigma(\cdot)$ is the sigmoid function, $\bar{b}$ and σ^2 are known prior mean and variance respectively. The evidence lower bound is

$$F(\bar{\mu}, v) = y\bar{\mu} - n\,\mathbb{E}_q \log(1 + \exp(\mu)) - \frac{\bar{\mu}^2 + v - 2\bar{\mu}\bar{b}}{2\sigma^2} + \frac{1}{2}\log v + \text{const.} \tag{4.39}$$

There is no close form for expected $\log(1+\exp(\mu))$. To evaluate $\mathbb{E}_q \log(1+\exp(\mu))$, we consider a Gauss-Hermite quadrature approximation as

$$\begin{aligned}\mathbb{E}\log(1+\exp(\mu)) &= \int \frac{1}{\sqrt{\pi}} \exp(-\mu^2)\log(1+\exp(\sqrt{2v}\mu+\bar{\mu}))d\mu \\ &\approx \frac{1}{\sqrt{\pi}} \sum_j w_j \log(1+\exp(\sqrt{2v}\mu_j+\bar{\mu})),\end{aligned} \tag{4.40}$$

where w_j, μ_j are pre-selected fixed sampling points. We have found that usually 10 points are enough to give accurate numerical values.

We note that the VGA problem for binomial data is more computationally demanding than the Poisson one due to the numerical approximation required for computing the expectation of $\log(1+\exp(\mu))$. However, it is still relatively straightforward to optimize for $\bar{\mu}$ and v using Newton's method.

4.4 Empirical Bayes Poisson matrix factorization

We return to the matrix factorization example for scRNA-seq data. We are interested in modelling scRNA-seq count data using Poisson distribution with a log-link function, while developing a flexible model and algorithm for different assumptions on the low dimensional structure. GLM-PCA is one of the model that factorizes UMI count data with a log-link function. More broadly, there are several existing methods that extend PCA to exponential family distributions, for example Poisson-PCA (Kenney et al. [2021]), generalizsed PCA (Landgraf and Lee [2020]), PLNPCA (Chiquet et al. [2018]) and ePCA (Liu et al. [2018]). These methods are able to factorize count matrices and can be potentially applied to scRNA-seq data.

4.4.1 Review of empirical Bayes matrix factorization

Wang and Stephens [2021] introduced empirical Bayes matrix factorization (EBMF), a flexible framework that allows for a wide range of prior families and enables different levels of sparsity to be exhibited by each component of the matrix factorization. The EBMF model is

$$
\begin{aligned}
Y &= \sum_k \boldsymbol{l}_k \boldsymbol{f}_k^T + E, && (4.41)\\
l_{k1}, ..., l_{kn} &\sim g_{l_k}, g_{l_k} \in \mathcal{G}_l, && (4.42)\\
f_{k1}, ..., f_{kp} &\sim g_{f_k}, g_{f_k} \in \mathcal{G}_f, && (4.43)\\
E_{ij} &\sim N(0, 1/\tau_{ij}). && (4.44)
\end{aligned}
$$

The priors on $\boldsymbol{l}_k$ and $\boldsymbol{f}_k$ can be chosen to reflect the data structure and modeling assumptions. For example, to induce sparsity, one may use a mixture of point-mass at 0 and a normal distribution with mean 0 (point-normal) or point-Laplace prior; to constrain the loadings and/or factors to be non-negative, one may use an exponential distribution prior. Fitting the EBMF model is done by a combination of variational inference and empirical Bayes approach, and it has been shown that the variational updates can be further reduced to empirical Bayes normal mean (EBNM) problems. The model fitting algorithm involves two major steps - greedy and back-fitting. In the greedy stage, the algorithm adds one factor at each step until the new factor is not able to increase the objective function. Then the back-fitting procedure prunes the factors by iteratively refitting each factor given the rest.

4.4.2 Model

We consider a model that combines the benefits of GLM-PCA and EBMF. In particular, we model the count data using a Poisson distribution with a log link function and introduce

priors on the loadings and factors. Consider the empirical Bayes Poisson matrix factorization (EBPMF) model

$$\begin{aligned} y_{ij} &\sim \text{Poisson}(s_i \exp(\mu_{ij})), \\ \mu_{ij}|\boldsymbol{l}_{i.}, \boldsymbol{f}_{j.} &\sim N\left(f_{j0} + \sum_k l_{ik} f_{jk}, \sigma_{ij}^2\right), \\ \boldsymbol{l}_k &\sim g_{\boldsymbol{l}_k}(\cdot), k = 1, 2, ..., K, \\ \boldsymbol{f}_k &\sim g_{\boldsymbol{f}_k}(\cdot), k = 0, 1, 2, ..., K. \end{aligned} \tag{4.45}$$

The s_i is a non-negative scaling scalar. In scRNA-seq UMI count data, s_i is usually chosen to reflect the cell size, and is fixed at $s_i = \sum_j y_{ij}$. The variance σ_{ij}^2 can be specified as constant across i, j, row-specific or column-specific. The f_{j0} is the baseline expression level of gene j, against which any changes are measured. This baseline is estimated from data and is unconstrained. We choose a non-sparse prior (such as normal prior) for f_{j0}. If changes can be positive or negative, then the baseline might be close to the mean or median expression; if changes are constrained to be positive, then the background is a "low" expression level. Each factor $\boldsymbol{f}_k$ is a gene expression program (GEP), and each loading $\boldsymbol{l}_k$ gives the membership of each GEP for the cells.

We restrict the posterior to have the form

$$\begin{aligned} q(\boldsymbol{\mu}, \boldsymbol{L}, \boldsymbol{F}) &= q_{\boldsymbol{\mu}} q_{\boldsymbol{L}} q_{\boldsymbol{F}} \\ &= \prod_{i,j} q_{\mu_{ij}} \prod_k q_{\boldsymbol{l}_k}(\boldsymbol{l}_k) \prod_k q_{\boldsymbol{f}_k}(\boldsymbol{f}_k) \\ &= \prod_{i,j} N(\mu_{ij}; \bar{\mu}_{ij}, v_{ij}) \prod_k q_{\boldsymbol{l}_k}(\boldsymbol{l}_k) \prod_k q_{\boldsymbol{f}_k}(\boldsymbol{f}_k). \end{aligned} \tag{4.46}$$

The splitting variational algorithm follows the general Algorithm 4, and is provided in Algorithm 5.

Algorithm 5 Splitting Variational Inference for EBPMF

1: **Input:** Count matrix $\boldsymbol{Y} \in \mathbb{N}_0^{n \times p}$
2: **Init:** $\bar{\boldsymbol{L}} = \boldsymbol{0}, \bar{\boldsymbol{F}} = \boldsymbol{0}$ (except $\boldsymbol{f}_0 = \log(Y^T \boldsymbol{1}/\boldsymbol{s})$)
3: **repeat**
4: Update $\bar{\mu}_{ij}, v_{ij}$ by solving variational Gaussian posterior approximation on data y_{ij}, s_i, with known prior mean $(\bar{\boldsymbol{L}}\bar{\boldsymbol{F}}^T)_{ij}$, and known (except for the first iteration) prior variance $\boldsymbol{\sigma}^2$, for $i = 1, 2, ..., n$ and $j = 1, 2, ..., p$.
5: Update $q_{\boldsymbol{L}}, g_{\boldsymbol{L}}, q_{\boldsymbol{F}}, g_{\boldsymbol{F}}$ by fitting EBMF (greedy Kmax = 1 and backfitting iteration = 1) on matrix $\boldsymbol{M} = [\bar{\mu}_{ij}]$ with known residual variance $\boldsymbol{\sigma}^2$.
6: Update σ^2 as $\sigma^2 = \sum_{ij} \mathbb{E}_q(\mu_{ij} - b_{ij})^2/(np)$.
7: **until** Converged
8: **Output:** Estimated priors $\hat{g}_{\boldsymbol{L}}, \hat{g}_{\boldsymbol{F}}$, estimated variance $\hat{\boldsymbol{\sigma}}^2$ and fitted posteriors $q_{\boldsymbol{\mu}}, q_{\boldsymbol{L}}, q_{\boldsymbol{F}}$

4.5 Numerical examples

In this section, we evaluate the performance of the EBPMF model through a simple simulation example and a simulation based on a real scRNA-seq dataset. We also compare the method with GLM-PCA and log transformation + EBNMF. Since GLM-PCA only allows unconstrained low-dimensional structures, we do not include it when comparing models with non-negative loadings and factors.

4.5.1 A simple example

We assess the EBPMF model in a simple simulation example, alongside GLM-PCA, log-transformation + EBMF, and Poisson-PCA. We set $N = 100$, $p = 300$, $K = 3$, and independently draw factors f_{jk} for $j = 1, 2, ..., p$, $k = 1, 2, ..., K$ from a standard normal distribution. The first 20 elements of loading $\boldsymbol{l}_1$ are independently sampled from $1 + \text{Unif}(0.5, 1)$, and all the remaining elements are 0. For the second loading, $l_{2,21:40}$ are independently sampled from $-2 + \text{Unif}(0.5, 1)$, and all the remaining elements are 0. Similarly, for the third loading, $l_{3,41:60}$ are independently sampled from $3 + \text{Unif}(0.5, 1)$, and all the remaining elements are 0. This structure defines three groups (clusters) of samples for the first 60 samples. The remaining 40 samples have all-zero loadings and belong to the fourth group. Observations y_{ij} are

sampled from a Poisson distribution according to model (4.45) (by setting $f_{j0} = 0, s_i = 1$). We consider two simulation settings when $\sigma_{ij}^2 = 0$ and $\sigma_{ij}^2 = 1$.

We fit EBPMF and EBMF using a mixture of point-mass and normal prior (point-normal prior) on both loadings and factors, and setting the variance type to be constant across i and j. For EBMF, we consider two log-transformations on the count data, $\log(1 + y_{ij})$ and $\log(0.1 + y_{ij})$. For all methods except EBPMF, we set K to be the true value, while for EBPMF, we set Kmax = 10 and let the model choose the optimal K. In this simulation example, EBPMF successfully selects the correct K. Figure 4.2 displays the true and estimated loading matrices when $\sigma_{ij}^2 = 0$. Notably, the EBPMF method recovers the true structure, and the sparsity-inducing prior shrinks the loadings of group 4 towards 0. Different log-transformations can lead to varying results when applying EBMF. In this simulation, the estimated loadings using the $\log(0.1 + y_{ij})$ transformation are closer to the true values. Although GLM-PCA and Poisson-PCA can detect the non-zero loadings for the first three groups, their interpretations of the loadings may differ from the true ones due to the lack of sparsity constraints. When $\sigma_{ij}^2 = 1$, accounting for the extra variation and including the sparsity constraint are crucial for recovering the true low-dimensional structure (Figure C.1).

In Figure 4.3, we show the the sequences of $\hat{\sigma}^2$ from the splitting variational algorithm in both simulations when $\sigma_{ij}^2 = 0$ and 1 for all i, j. It takes many more iterations for the algorithm to converge when the true σ^2 is 0. In the Figure 4.3a, the $\hat{\sigma}^2$ is 0.039 at iteration 100, and in the Figure 4.3b, the $\hat{\sigma}^2$ is 0.98 at iteration 29. This difference in the number of iterations indicates that the algorithm requires additional iterations to converge when the true σ^2 is 0 or close to 0. Despite this sensitivity, the EBPMF method demonstrates robust performance in recovering the true low-dimensional structure across various simulation settings.

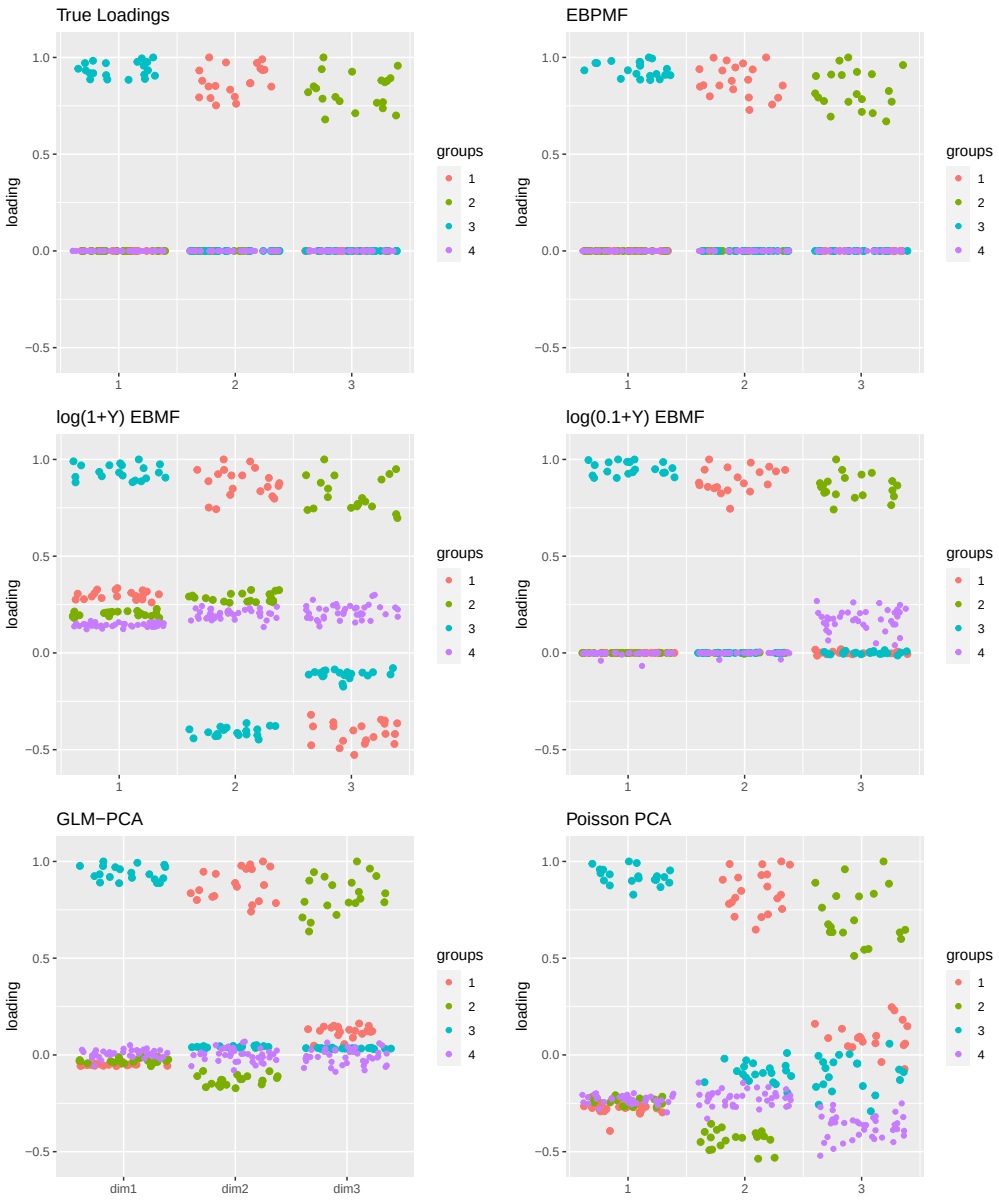
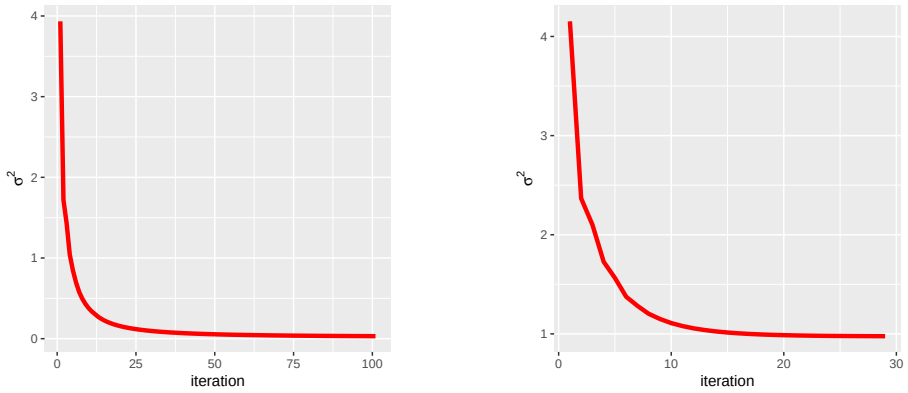


Figure 4.2: Plot of the loading matrices in simulation example of EBPMF. $N = 100, p = 300, K = 3, \sigma_{ij}^2 = 0$. The signs of loadings are flipped so that the largest element of each loading is positive, and scaled to be 1 for visualization purpose. In each plot, each column is a loading, and colors of dots indicate groups.



(a) True $\sigma^2_{ij} = 0$.

(b) True $\sigma^2_{ij} = 1$.

Figure 4.3: The sequence of $\hat{\sigma}^2_{ij}$ from the splitting variational algorithm when fitting EBPMF in the simple simulation examples.

4.5.2 Simulation based on scRNA-seq data from Zheng et al. [2017]

In this study, we conducted a more realistic simulation by fitting the EBPMF model on a subset of a public available scRNA-seq dataset and then generating data from the fitted model. The data consists of a selection of PBMC (Donor A) scRNA-seq data described in Zheng et al. [2017], comprising unique molecular identifier (UMI) counts for 16,791 genes in 3,774 cells. The data is included in the R package `fastTopics` (Carbonetto et al. [2021], Dey et al. [2017]). In the pre-processing phase, genes expressed in fewer than 10 cells were removed, with no further filtering or pre-processing conducted. The final dataset includes 3,774 cells and 11,487 genes, with approximately 94% of the matrix entries being 0. Five main cell types – B cell, CD14+, CD34+, NK cell, and T cell, and 10 sub-cell types are present in the data and are sorted by FACS (as shown in Supplementary Figure 6. in Zheng et al. [2017]).

We fitted the EBPMF model with sparse and non-negative priors (a mixture of point-mass and exponential distribution, in short point-exponential) on both loadings and factors. The method identified 8 factors (in addition to the background factors), with the cell membership in the 8 factors illustrated in the top structure plot in Figure 4.4. Subsequently, data were

generated from the fitted EBPMF model. We also compared the EBPMF model fit with the results from log-transformation + EBNMF and log-transformation + NMF (squared loss). The log-transformation used was $\tilde{y}_{ij} = \log\left(1 + \frac{\text{median}(s_i)}{0.5}\frac{y_{ij}}{s_i}\right)$, where s_i is the cell size of cell i. This transformation is derived from $\log\left(\frac{y_{ij}}{s_i} + \frac{0.5}{\text{median}(s_i)}\right)$, while preserving data sparsity.

The EBPMF method accurately recovers the number of factors, and the structure plot of the estimated cell membership closely resembles the true one. The EBNMF method on log-transformed data also manages to recover the main structure, although the loadings are less sparse and exhibit some spikes, likely due to the transformation. Comparing the NMF model fit at the bottom of Figure 4.4 with EBPMF, it is clear that EBPMF's sparsity constraint results in more interpretable components than those provided by the standard NMF fits..

4.6 Real data results

We analyze two scRNA-seq UMI count data using EBPMF with point-exponential priors on both loadings and factors. The first dataset is the full PBMC purified data from Zheng et al. [2017] with more than 90,000 cells. The second dataset is from Montoro et al. [2018], where more than 7000 trachea epithelial cells from mice were sequenced by droplet-based 3' scRNA-seq.

4.6.1 PBMC purified data from Zheng et al. [2017]

We applied the EBPMF model to the purified PBMC scRNA-seq data from Zheng et al. [2017]. Cell types are provided by the authors, and are sorted by FACS. Thus, they are treated as true cell type labels. The dataset contains 94,655 cells from five main cell types. We selected a set of 3,000 most variable genes using the `devianceFeatureSelection` func-

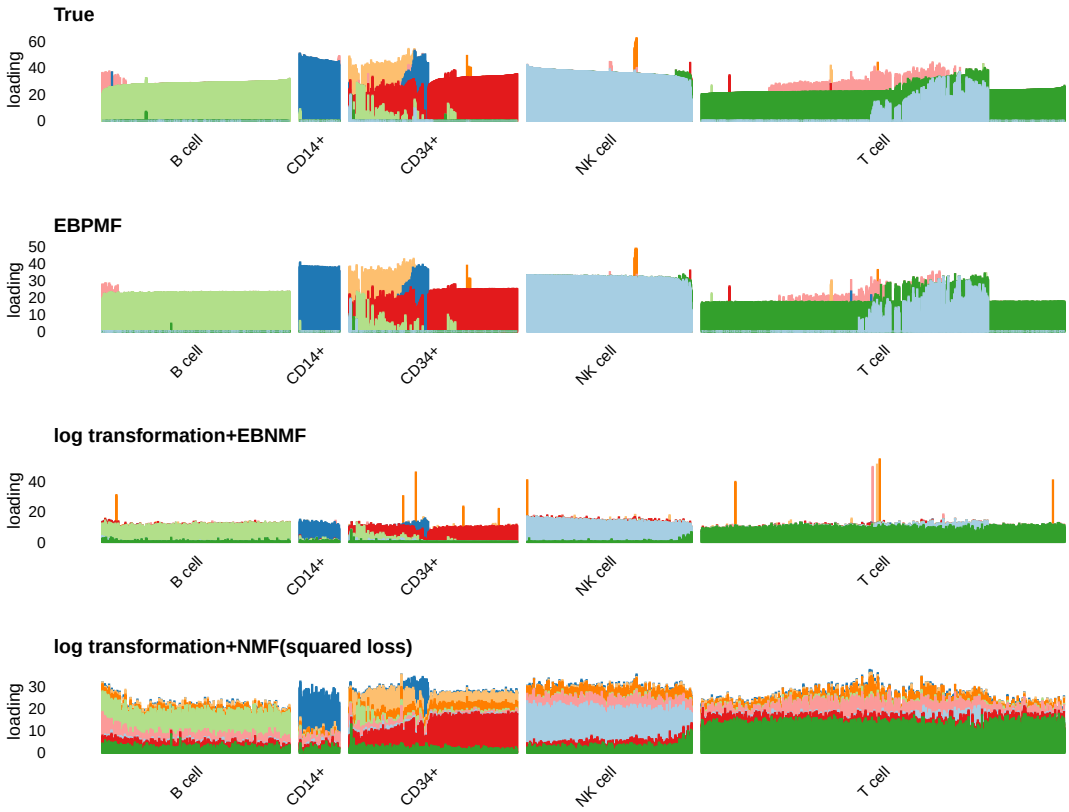


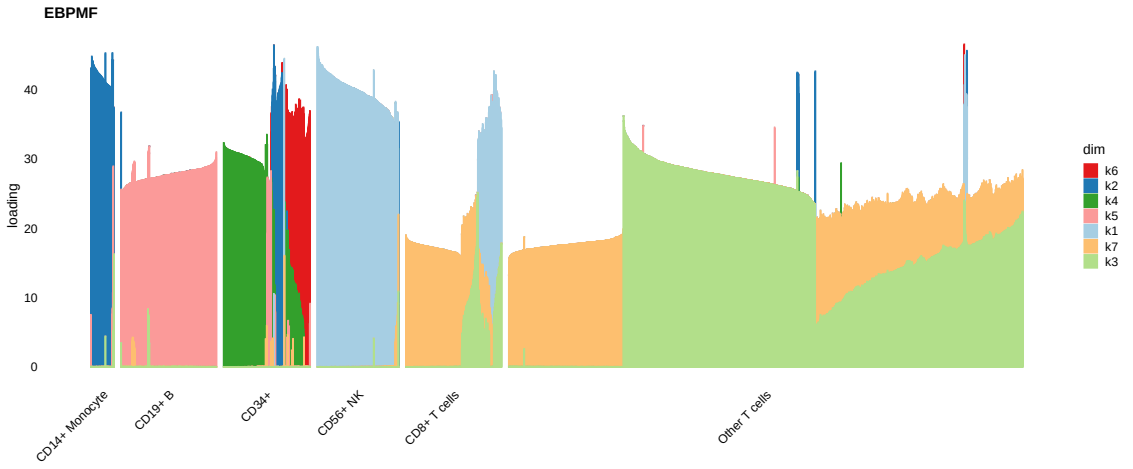
Figure 4.4: Structure plots of cell membership (loading matrix) in simulated PBMC data.

tion in the R package `scry` (Street et al. [2021]). Afterward, we removed any remaining genes for which transcripts appeared in fewer than 10 cells. The EBPMF model, with sparse and non-negative priors, identified seven factors (in addition to the baseline one), and the cell membership is displayed as a structure plot in Figure 4.5a.

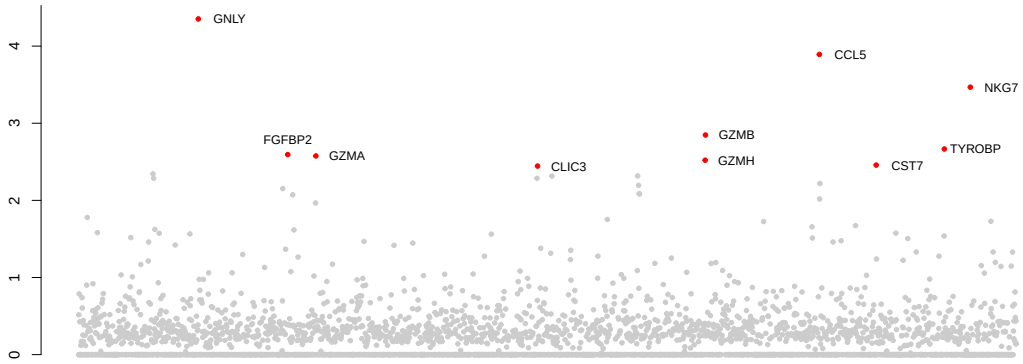
Clearly, many EBPMF factors correspond to one or more cell types: factor 1 is associated with NK cells and a portion of CD8+ T cells, factor 2 is specific to CD14+ monocyte cells, factors 3 and 7 are present in T cells, factors 4 and 6 appear in CD34+ cells, and factor 5 is specific to B cells. An interesting observation is the shared factor 1 by NK cells and some CD8+ T cells, with the factor being more specific to the former cell type. Indeed, NK cells are part of the innate immune system that can respond quickly to pathological challenges. While CD8+ T cells perform specific roles in mediating adaptive immune responses, they sometimes exhibit "NK-like" properties (Pereira et al. [2020]). This relationship is clearly captured by the EBPMF model fit. Figure 4.5b presents the scatterplot of factor 1, and we label the top 10 genes with the highest expression program in this factor. Since the EBPMF model includes a baseline factor and employs a sparsity-inducing prior on factors, the genes with the highest expression program in each factor are key driving genes for the underlying biological processes. The genes appearing in the figure, such as *GNLY, GZMB, NKG7*, are clearly related to the immune system, as they encode proteins primarily found in cytotoxic T cells and natural killer cells.

4.6.2 Trachea epithelial cells scRNA-seq data

We re-examine the scRNA-seq UMI count data from Montoro et al. [2018], which consists of 7,193 trachea epithelial cells sequenced using droplet-based 3' scRNA-seq. The original analysis classified cells into seven distinct clusters, which were annotated post hoc based on the expression of known marker genes. These clusters were mapped to abundant cell types, such as basal, club, and ciliated cells, as well as rare cell types, including tuft, neuroendocrine,



(a) Structure plot of the cell membership in each gene expression programs.



(b) Scatter plot of factor 1. Top 10 genes are labelled in red.

Figure 4.5: EBPMF fit on the PBMC purified scRNA-seq data from Zheng et al. [2017].

and goblet cells. Note that the cell type labels come from clusters not FACS. The publicly available dataset (GEO accession GSE103354) contains gene expression profiles for 18,388 genes across 7,193 cells. Around 90.7% of the entries in the data matrix are 0.

We fit the EBPMF model using point-exponential priors on both loadings and factors, and we set the variances to be gene-specific. The model identifies a total of 14 factors. Figure 4.6a displays the structure plot of cell membership for all cell types and the top 7 gene expression programs (GEPs). It is evident that most cells exhibit mixed membership in different GEPs, while some factors are specific to certain cell types. For instance, basal cells primarily have membership in GEP 2, ciliated cells in GEP 1, and tuft cells in GEP 3. GEP 6 is specific to neuroendocrine cells. Rare cell types have fewer samples, so to obtain a clearer view of their membership, we zoom in on the structure plot for these cells in Figure 4.6b. It is apparent that GEP 11 is specific to goblet cells, and most ionocyte cells have unique membership in GEP 9.

For GEPs that are specific to a cell type, we are interested in identifying the corresponding key driving genes. To do this, we plot the mean gene expression (in log space) across all cells versus the gene expression program (factor). The top genes appearing in the plot are then the key driving genes for the corresponding GEP. Figure 4.7 displays the plot for GEP 1, and the known cell marker genes provided in Montoro et al. [2018] are labeled in black. Clearly, these marker genes are all genes with high expression programs in GEP 1. Specifically, *Cdhr3* is a risk gene for asthma that encodes a rhinovirus receptor and has been linked to severe childhood asthma exacerbations. We also checked the other genes with high expression programs, and they match most of the top marker genes provided in Supplemental Table 1 in Montoro et al. [2018], such as *AU040972*, *Ccdc153*, *Tmem212*, and *Dynlrb2*.

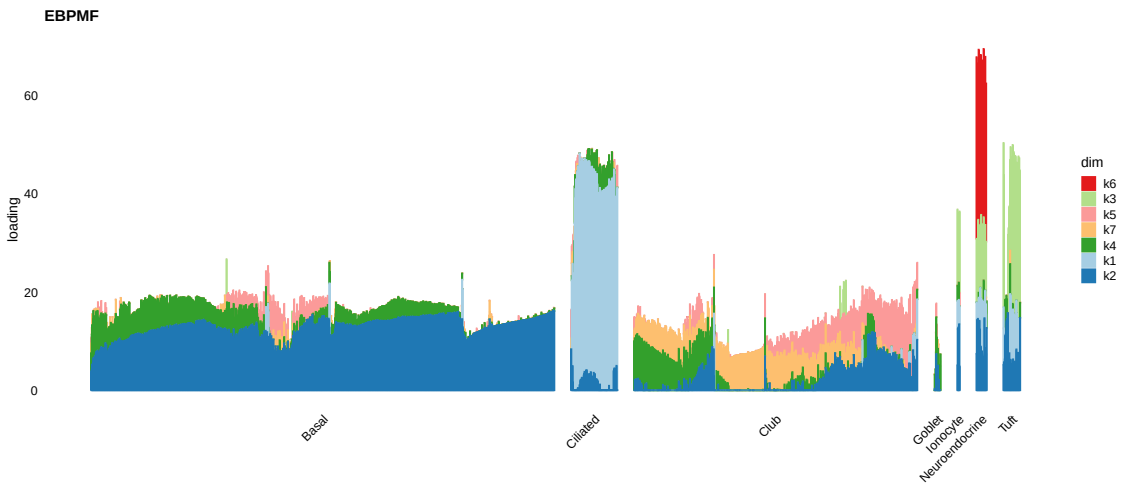
One of the main findings in Montoro et al. [2018] is the detection of a rare cell type, the ionocyte. Figure 4.8 displays the key driving genes for GEP 9, a factor in which most ionocyte

cells have a unique membership. Known cell type marker genes provided in Montoro et al. [2018] (in the Figure 5c and Extended Data Fig. 1d) are labeled in black, while genes appearing at the top of the plot are labeled in red. The latter also match the top marker genes identified in a post hoc analysis in Montoro et al. [2018]. A considerable number of genes highlighted by the authors as crucial to ionocyte function, such as *Cftr*, *Foxi1*, and *Ascl3*, appear more prominently in this plot. In particular, as pointed out by the authors, pulmonary ionocytes express the cystic fibrosis transmembrane conductance regulator (*Cftr*). Although ionocytes make up only approximately 0.4% of mouse cells profiled through single-cell RNA sequencing, they express over 54% of all identified *Cftr* transcripts.

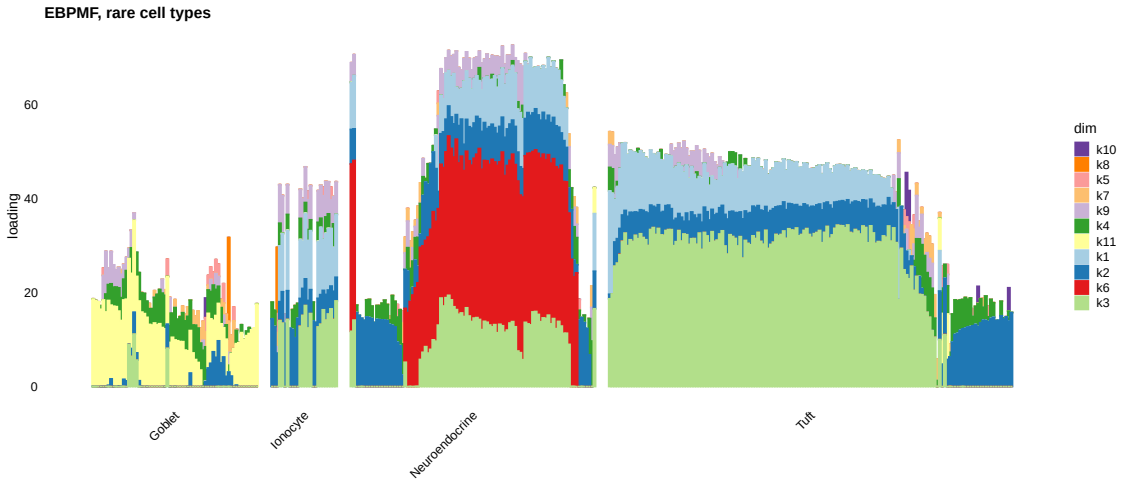
Figure 4.9 shows the marker genes for each sub-cell type of goblet cells. They all appear at the top of the plot and are more prominently displayed in this context compared to a list ordered by GEP. In addition, the plots for GEP 6 (neuroendocrine cells), GEP 3 (tuft cells), and GEP 2 (basal cells) are shown in Figure C.2. These plots confirm that the EBPMF method successfully identifies cell memberships and the key driving genes for each GEP.

4.7 Discussion

In this study, we proposed a novel variational inference approach for non-Gaussian data and developed the empirical Bayes Poisson matrix factorization method for scRNA-seq data analysis. Our main contribution lies in introducing a modular algorithm that applies well-established Gaussian empirical Bayes methods to non-Gaussian models. The EBPMF method allows for different prior families on both loadings and factors, as well as different assumptions about the latent structure, including non-negativity and sparsity of factors. This flexibility enables more accurate and interpretable recovery of latent gene expression patterns and their corresponding cell memberships.



(a) Structure plot of the cell membership in each of top 7 gene expression programs, all cell types.



(b) "Zoom-in" structure plot of the cell membership in each of 11 gene expression programs, rare cell types only.

Figure 4.6: Structure plot of the cell membership, results from EBPMF fit on Montoro et al. [2018] data. Cell types are are annotated post hoc by the authors.

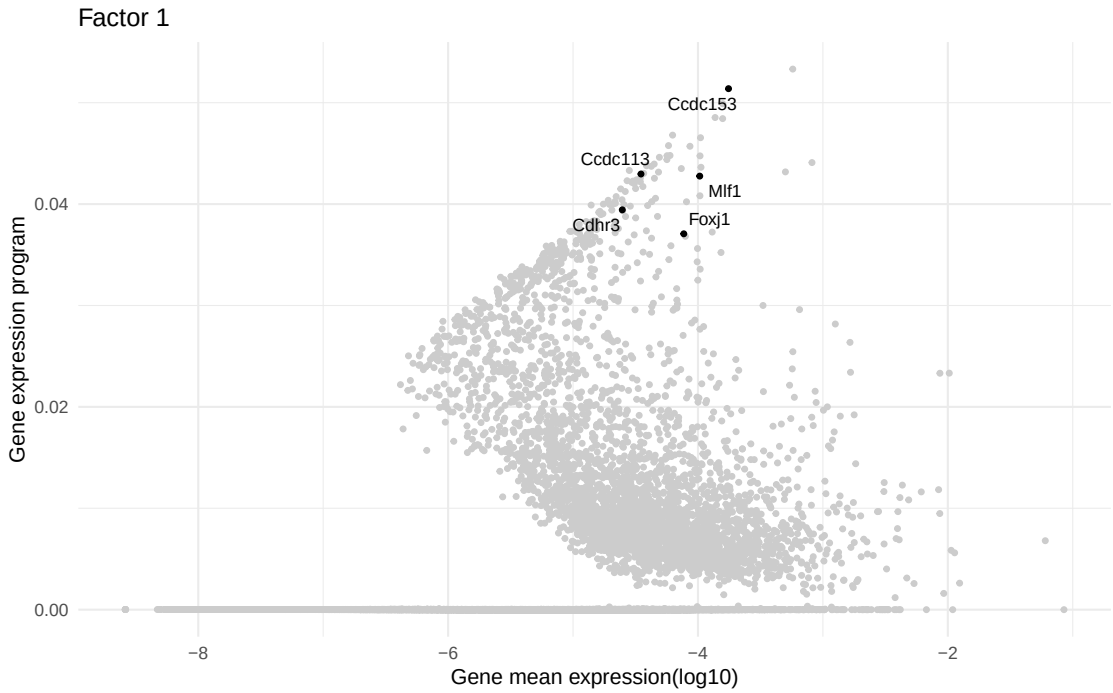


Figure 4.7: Factor 1, ciliated cells.

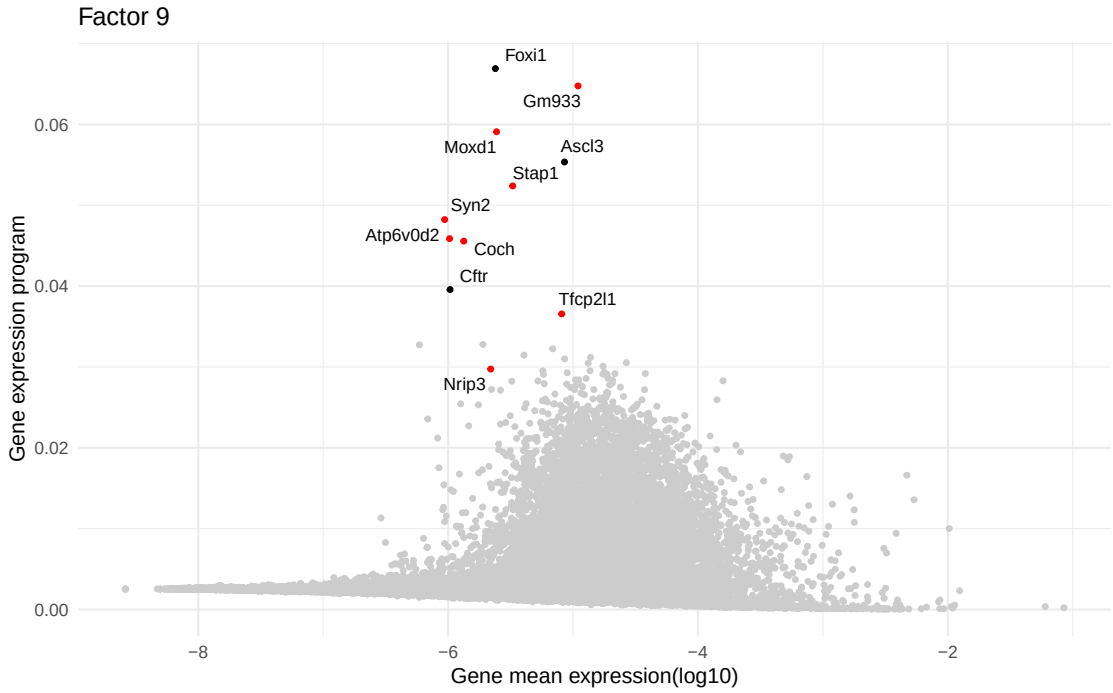


Figure 4.8: Factor 9, ionocyte cells. Known marker genes provided in Montoro et al. [2018] (from Figure 5c and Extended Data Fig. 1d) are labelled in black. Genes showing on top of the plot (that also match the marker genes detected in Montoro et al. [2018]) are labelled in red.

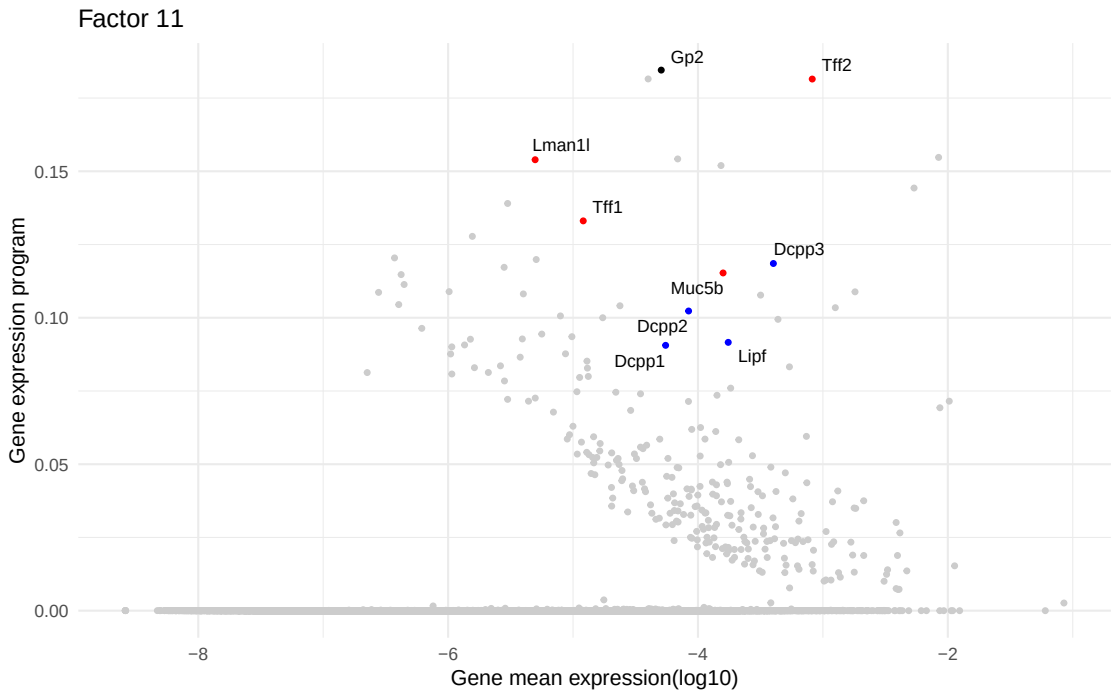


Figure 4.9: Factor 11, goblet cells. Known marker gene *Gp2* provided in Montoro et al. [2018] is labelled in black. Genes showing on top of the plot that are marker genes of goblet-1 cells are labelled in red and of goblet-2 cells are labelled in blue.

Simulation studies and real scRNA-seq data analysis demonstrate the superior performance of the EBPMF method in terms of latent structure recovery, data adaptation, and downstream analysis for scRNA-seq data such as key driving gene identification. The method enables a deeper understanding of the complex biological processes governing cellular function, which has the potential to drive further advances in the study of cellular heterogeneity and its implications for health and disease.

Our novel variational inference method can be extended to other types of non-Gaussian data, such as binomial data, and other statistical models, such as regression analysis. In the context of regression, developing a variational inference method for sparse generalized linear regression can be challenging. However, the splitting variational inference method makes it straightforward to handle various sparsity inducing priors on regression coefficients. Overall, the versatility and modularity of our splitting variational inference method make it suitable for application to various combinations of data types and models.

Code and data availability

The EBPMF method is implemented in the `R` package `ebpmf`, available at `https://github.com/DongyueXie/ebpmf`.

Code for producing the plots for the Poisson Gaussian process is at `https://github.com/DongyueXie/gsmash/blob/main/analysis/pois_gp_split.Rmd`. Code for the simulation and real data analysis is at `https://github.com/DongyueXie/ebpmf-paper`.

The scRNA-seq PBMC data used in the simulation is from `R` package `fastTopics`. The scRNA-seq PBMC purified data is downloaded from 10x Genomics website. A detailed preparation guide can be found at `https://stephenslab.github.io/single-cell-topics/prepare_purified_pbmc.html`.

The droplet scRNA-seq UMI count data from Montoro et al. [2018] is available from GEO with accession number GSE103354. A detailed preparation guide can be found at `https://stephenslab.github.io/single-cell-topics/prepare_droplet.html`.

CHAPTER 5

DEVELOPING AND EXTENDING EMPIRICAL BAYES POISSON NON-NEGATIVE MATRIX FACTORIZATION

5.1 Introduction

Traditional clustering methods partition samples into distinct subgroups, and each sample belongs to one of the groups. Topic models, also known as grade of membership (GoM) models, generalize clustering models by allowing each sample to have partial membership in multiple clusters or factors (Pritchard et al. [2000], Blei et al. [2003]). The model has been applied to learn and visualize the latent structure of RNA-seq data (Dey et al. [2017]). Gene expressions can be regarded as the outcome of interactions of several interrelated biological processes and topic model assigns partial memberships to each sample in multiple biological clusters.

In this chapter, we study two novel models that enhance the vanilla topic model or the non-negative matrix factorization (Lee and Seung [1999], Carbonetto et al. [2021]). The first model allows the loadings (membership) and/or the factors (biological process) to be smooth so that it potentially fits better to the data that exhibits spatial structures. For example, sequencing-based assays that measure expression or other traits along the genome. The second model employs a recently proposed transformation on the count matrix called biwhitening (Landa et al. [2022]), that enables us to use existing flexible non-negative matrix factorization methods developed for Gaussian models, such as EBNMF (Willwerscheid [2022]). We show using numerical examples that both models can improve the latent structure recovery and improve the interpretability of loadings and factors.

5.2 Smoothed Poisson non-negative matrix factorization

In Xing [2016], a smoothed version of grade of membership model, smoothed GoM, was proposed to factorize RNA-seq data measured at fine resolution - per base pair along the genome. Since the read counts are measured along the genome, we expect the underlying signal to have spatial structure. The method combines a smoothing step with an EM update of factors. It was found empirically that the method improves the accuracy of latent structure recovery when compared to the standard topic model.

Though the smoothed GoM has been shown the potential to work in practice, it lacks an explicit objective function. This is due to the "plug-in" nature of the smoothing step - there's no theoretical verification of it. It is unclear whether the algorithm corresponds to a specific model. In this chapter, we develop smoothed Poisson non-negative matrix factorization (SPNMF) model, that allows for spatially-structured loadings and/or factors, and derive a variational empirical Bayes algorithm for model inference. Though the original topic model has a multinomial likelihood, Carbonetto et al. [2021] showed that one can potentially utilize algorithms for non-negative matrix factorization (NMF) to improve parameter estimation in topic models.

In the following sections, we first review a basically framework for empirical Bayes Poisson matrix factorization with identity link function, then generalize the framework to allow more flexible priors.

5.2.1 Empirical Bayes Poisson non-negative matrix factorization

The Poisson matrix factorization (PMF) model with identity link function is

$$x_{ij} \sim \text{Poisson}\left(\sum_k l_{ik} f_{jk}\right), \tag{5.1}$$

where $l_{ik} \geq 0$, and $f_{jk} \geq 0$. A Bayesian version of the model has priors on $\boldsymbol{L}$ and $\boldsymbol{F}$, as

$$\boldsymbol{l}_k \sim g_{\boldsymbol{l}_k}(\cdot), \boldsymbol{f}_k \sim g_{\boldsymbol{f}_k}(\cdot), \tag{5.2}$$

where each prior has support on the non-negative real line. We briefly introduce the Bayesian Poisson matrix factorization framework originally studied in Cemgil [2009], but also allow the priors $g_{\boldsymbol{l}_k}(\cdot), g_{\boldsymbol{f}_k}(\cdot)$ to be estimated using the variational approach. The variational empirical Bayes version was studied by Zihao Wang in his unpublished work (`https://zihao12.github.io/ebpmf_demo/ebpmf.pdf`). In these existing work, typical choices of $g(\cdot)$ are gamma distribution due to the conjugacy. For example, $l_{ik} \overset{iid}{\sim} \text{Gamma}(a_k, b_k)$. Here we describe a more general model that allows multivariate prior on the loadings and/or factors. Thus we can assume smoothness-inducing priors. Choosing the smoothness-inducing prior to be the multiscale prior introduced in Appendix D.4 is analogous to a model-based version of the smoothed GoM introduced in Xing [2016]. We first introduce the following procedures as they will be used in the variational inference algorithm.

Definition 5.2.1. *An EB Poisson mean (EBPM) procedure defines a mapping from $\boldsymbol{y}, \boldsymbol{s}$ to $(\hat{g}, q)$, under the model*

$$y_i|\lambda_i \sim \text{Poisson}(s_i\lambda_i),$$
$$\boldsymbol{\lambda} \sim g(\cdot) \in \mathcal{G}.$$

where $\boldsymbol{y}$ is a vector of observations, s_i is a knwon scaling scalar, $\hat{g}$ is the estimated prior, and q is the posterior. This mapping is denoted as

$$(\hat{g}, q) = EBPM_{\mathcal{G}}(\boldsymbol{y}, \boldsymbol{s}).$$

We note that when choosing a smoothness-inducing prior for $\boldsymbol{\lambda}$ in EBPM, i.e. $\mathcal{G}_{\text{smooth}}$,

the procedure becomes an EB Poisson smoothing procedure. We denote such procedure as $(\hat{g}_{\text{smooth}}, q) = EBPM_{\mathcal{G}_{\text{smooth}}}(\boldsymbol{y}, \boldsymbol{s})$.

Rank-1 Model

For simplicity we study the rank-1 model first, by setting $k = 1$ in (5.1). In matrix form, we may write the model as $\boldsymbol{Z} \sim \text{Poisson}(\boldsymbol{l}\boldsymbol{f}^T)$ where $\boldsymbol{l} = (l_1, l_2, ..., l_N)^T$ and $\boldsymbol{f} = (f_1, f_2, ..., f_p)^T$, $\boldsymbol{l} \sim g_{\boldsymbol{l}}(\cdot) \in \mathcal{G}_{\boldsymbol{l}}$, $\boldsymbol{f} \sim g_{\boldsymbol{f}}(\cdot) \in \mathcal{G}_{\boldsymbol{f}}$.

An empirical Bayes procedure estimates priors via $\hat{g}_{\boldsymbol{l}}, \hat{g}_{\boldsymbol{f}} = \arg\max_g \log p(\boldsymbol{Z}|g_{\boldsymbol{l}}, g_{\boldsymbol{r}})$, then calculates posterior of $\boldsymbol{l}, \boldsymbol{f}$. We use mean field variational method to approximate the posterior of $\boldsymbol{l}$ and $\boldsymbol{f}$. Assume a variational family that the distributions factorize as $q(\boldsymbol{l}, \boldsymbol{f}) = q_{\boldsymbol{l}}(\boldsymbol{l})q_{\boldsymbol{f}}(\boldsymbol{f})$, the evidence lower bound is

$$F(q, g) = \mathbb{E}_q \log p(\boldsymbol{Z}|\boldsymbol{l}, \boldsymbol{f}) + \mathbb{E}_q \log \frac{g_{\boldsymbol{l}}(\boldsymbol{l})}{q_{\boldsymbol{l}}(\boldsymbol{l})} + \mathbb{E}_q \log \frac{g_{\boldsymbol{f}}(\boldsymbol{f})}{q_{\boldsymbol{f}}(\boldsymbol{f})}. \tag{5.3}$$

Given $q_{\boldsymbol{f}}, g_{\boldsymbol{f}}$, the objective function with respect to $q_{\boldsymbol{l}}, g_{\boldsymbol{l}}$ is

$$F(q_{\boldsymbol{l}}, g_{\boldsymbol{l}}) = \mathbb{E}_{q_{\boldsymbol{l}}} \sum_i \left(\sum_j z_{ij} \log l_i - l_i \sum_j \bar{f}_j \right) - D_{KL}(q_{\boldsymbol{l}}||g_{\boldsymbol{l}}), \tag{5.4}$$

We recognize that this is the objective function EBPM problem. Thus the update of $q_{\boldsymbol{l}}, g_{\boldsymbol{l}}$ can be obtained by $\text{EBPM}_{\mathcal{G}_{\boldsymbol{l}}}(\boldsymbol{Z}\boldsymbol{1}_p, \sum_j \bar{f}_j)$, where $\bar{f}_j = \mathbb{E}_q f_j$. Given $q_{\boldsymbol{l}}, g_{\boldsymbol{l}}$, the updates of $q_{\boldsymbol{f}}, g_{\boldsymbol{f}}$ are similar to the ones in updating $q_{\boldsymbol{l}}, g_{\boldsymbol{l}}$. The corresponding EBPM problem is $\text{EBPM}_{\mathcal{G}_{\boldsymbol{f}}}(\boldsymbol{Z}^T\boldsymbol{1}_N, \sum_i \bar{l}_i)$, where $\bar{l}_i = \mathbb{E}_q l_i$. The algorithm for fitting rank-1 model is given in Algorithm 6.

Algorithm 6 Rank-1 EBPMF (identity link)

1: **Input:** Count matrix $\boldsymbol{Z}$.
2: **Init:** $q_{\boldsymbol{l}}, q_{\boldsymbol{f}}, g_{\boldsymbol{l}}, g_{\boldsymbol{f}}$.
3: **repeat**
4: $\quad (q_{\boldsymbol{l}}, \hat{g}_{\boldsymbol{l}}) = \text{EBPM}_{\mathcal{G}_{\boldsymbol{l}}}(\boldsymbol{Z}\mathbf{1}_p, \sum_j \bar{f}_j)$;
5: $\quad (q_{\boldsymbol{f}}, \hat{g}_{\boldsymbol{f}}) = \text{EBPM}_{\mathcal{G}_{\boldsymbol{f}}}(\boldsymbol{Z}^T\mathbf{1}_N, \sum_i \bar{l}_i)$.
6: **until** Converged
7: **Output:** $q_{\boldsymbol{l}}, q_{\boldsymbol{f}}, \hat{g}_{\boldsymbol{l}}, \hat{g}_{\boldsymbol{f}}$

Rank-K Model

The extension to rank-K model is straightforward, based on the fact that sum of independent Poisson random variables are Poisson random variables. The rank-K model can be written as

$$\begin{aligned} \boldsymbol{X}|\boldsymbol{Z}_1, ..., \boldsymbol{Z}_K &= \sum_{k=1}^{K} \boldsymbol{Z}_k, \\ \boldsymbol{Z}_k|\boldsymbol{l}_k, \boldsymbol{f}_k &\sim \text{Poisson}(\boldsymbol{l}_k \boldsymbol{f}_k^T), \\ \boldsymbol{f}_k &\sim g_{\boldsymbol{f}_k}(\cdot), \\ \boldsymbol{l}_k &\sim g_{\boldsymbol{l}_k}(\cdot), \end{aligned} \tag{5.5}$$

where we have introduced latent variables $\boldsymbol{Z}_k$, each of which follows a rank-1 model.

Again we use a variational inference approach and assume the posterior factorizes as

$$\begin{aligned} q(\boldsymbol{Z}, \boldsymbol{L}, \boldsymbol{F}) &= q_Z(\boldsymbol{Z}) q_L(\boldsymbol{L}) q_F(\boldsymbol{F}) \\ &= \prod_{i,j} q_{z_{ij}}(\boldsymbol{z}_{ij}) \prod_k q_{\boldsymbol{l}_k} q_{\boldsymbol{f}_k}. \end{aligned} \tag{5.6}$$

Given $q_Z(\cdot)$, the objective function for $q_{\boldsymbol{l}_k}(\cdot), g_{\boldsymbol{l}_k}(\cdot), q_{\boldsymbol{f}_k}(\cdot), g_{\boldsymbol{f}_k}(\cdot)$ for $k = 1, 2, ..., K$ can be

Algorithm 7 EBPMF (identity link)

1: **Input:** Count matrix $\boldsymbol{X}$, rank K.
2: **Init:** q_L, q_F, g_L, g_F, q_Z.
3: **repeat**
4: Update $q_{\boldsymbol{l}_k}, q_{\boldsymbol{f}_k}, g_{\boldsymbol{l}_k}, g_{\boldsymbol{f}_k}$ by running Algorithm 6 with observation $\bar{\boldsymbol{Z}}_k$ for $k = 1, 2, ..., K$;
5: Update $\bar{\boldsymbol{Z}}_k$ according to (D.4).
6: **until** Converged
7: **Output:** $\hat{q}_L, \hat{q}_F, \hat{g}_L, \hat{g}_F, \hat{q}_Z$

written as a summation of K sub-objective functions

$$\begin{aligned} F(\cdot) &= \sum_k F_k(\cdot), \\ &= \sum_k \mathbb{E} \log p(\bar{\boldsymbol{Z}}_k | \boldsymbol{l}_k, \boldsymbol{f}_k) + \mathbb{E} \log \frac{g_{\boldsymbol{l}_k}(\boldsymbol{l}_k)}{q_{\boldsymbol{l}_k}(\boldsymbol{l}_k)} + \mathbb{E} \log \frac{g_{\boldsymbol{f}_k}(\boldsymbol{f}_k)}{q_{\boldsymbol{f}_k}(\boldsymbol{f}_k)}. \end{aligned} \tag{5.7}$$

where $F_k(\cdot)$ is the objective function for a specific k, and $\bar{\boldsymbol{Z}}_k := \mathbb{E}\, \boldsymbol{Z}_k$. One can easily verify that maximizing $F_k(\cdot)$ is equivalent to solving a rank-1 problem with $\bar{\boldsymbol{Z}}_k$ as the observations. The variational inference algorithm is given in Algorithm 7.

5.2.2 Smoothed Poisson non-negative matrix factorization

The smoothed Poisson non-negative matrix factorization (SPNMF) is a special case of the general model (5.5), and we assume smoothness-inducing priors on loadings and/or factors, $\boldsymbol{f}_k \sim g_{\boldsymbol{f}_k,\text{smooth}}(\cdot)$, and $\boldsymbol{l}_k \sim g_{\boldsymbol{l}_k,\text{smooth}}(\cdot)$. To fit the SPNMF model, we only need to use an EB Poisson smoothing procedure in algorithm 6. Specifically, the procedures are $\text{EBPM}_{\mathcal{G}_{\boldsymbol{l},\text{smooth}}}$ and $\text{EBPM}_{\mathcal{G}_{\boldsymbol{f},\text{smooth}}}$.

The extra variation in the count data might lead to non-smooth posterior mean of loadings and/or factors, even if we assume the smoothness-inducing priors. In this case, we can take advantage of the model 3.1 introduced in Chapter 3 to obtain smooth results. We focus on smoothing factors, as each sample is usually collected along spatial coordinates. Specifically,

consider the following rank-1 model:

$$\begin{aligned} z_{ij}|l_i, f_j &\sim \text{Poisson}(l_i f_j), \\ \boldsymbol{l} &\sim g_{\boldsymbol{l}}(\cdot), \\ f_j &= \exp(\mu_j), \\ \mu_j|b_j &\sim N(b_j, \sigma^2), \\ \boldsymbol{b} &\sim g_{\boldsymbol{b},\text{smooth}}(\cdot), \end{aligned} \tag{5.8}$$

for $i = 1, 2, ..., N$, and $j = 1, 2, ..., p$. In Chapter 3, we have introduced the nugget effect and splitting algorithm for smoothing over-dispersed count data. The formulation of $\boldsymbol{f}$ here is exactly the same, but in a matrix factorization context. We will see below that within the EBPMF framework as in Algorithm 7, the splitting algorithm for smoothing is involved.

Assume the posterior factorizes as $q(\boldsymbol{l}, \boldsymbol{\mu}, \boldsymbol{b}) = q_{\boldsymbol{l}} q_{\boldsymbol{\mu}} q_{\boldsymbol{b}} = q_{\boldsymbol{l}} q_{\boldsymbol{b}} \prod_j q_{\mu_j}$, then the evidence lower bound for the model is

$$\begin{aligned} F(q, g, \sigma^2) = \sum_{i,j} \mathbb{E}\left(z_{ij}(\log l_i + \mu_j) - l_i \exp(\mu_j)\right) \\ + \mathbb{E}\log\frac{g_{\boldsymbol{l}}}{q_{\boldsymbol{l}}} + \sum_j \mathbb{E}\log\frac{N(\mu_j|b_j, \sigma^2)}{q_{\mu_j}} + \mathbb{E}\log\frac{g_{\boldsymbol{b}}}{q_{\boldsymbol{b}}}. \end{aligned} \tag{5.9}$$

Given $g_{\boldsymbol{l}}, q_{\boldsymbol{l}}$, the update of $q_{\mu_j}, g_{\boldsymbol{b}}, q_{\boldsymbol{b}}, \sigma^2$ is given by Algorithm 3 in Chapter 3, with observations $\sum_i z_{ij}$, for $j = 1, 2, ..., p$, and scaling factor $\sum_i \bar{l}_i$. For a general rank-K SPNMF model, it can be also reduced to solve K rank-1 problems iteratively, similar to Algorithm 7.

5.2.3 An illustrative example

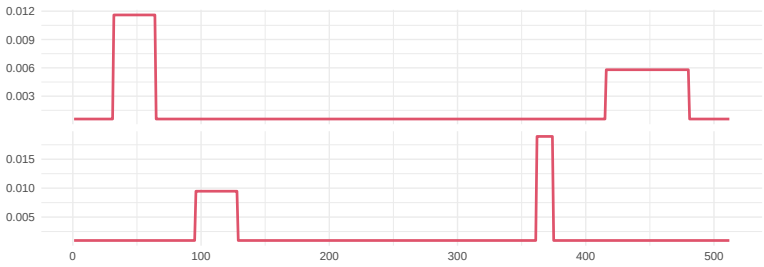
To illustrate the benefits of using SPNMF over vanilla NMF when the true factors are smooth, we consider a simple example with $n = 45, K = 2$, and $p = 512$. The loadings l_{ik} are drawn

i.i.d. from a mixture of exponential distributions: $\frac{1}{3}\,\mathrm{Exp}(10) + \frac{1}{3}\,\mathrm{Exp}(5) + \frac{1}{3}\,\mathrm{Exp}(1)$. The factors are step functions, with base level 0.5, shown in Figure 5.1a. In the first simulation, we set the largest step value of the factors to be 10 and the smaller step size to be 5. As shown in Figure 5.1, both vanilla NMF and SPNMF are able to recover the true factor structures because of the relatively strong signals in the factors. In the second simulation, we set the largest step value of the factors to be 5 and the smaller step size to be 1. In Figure 5.2, vanilla NMF was not able to give a clear recovery of the factors because the smaller steps seem to be mixed with the variations in the data. On the other hand, SPNMF is able to recover the true factor structures. Figure 5.2c clearly visualizes the smooth pattern of factors. This suggests that when the true underlying factors are smooth, incorporating the smoothness in the model can give more accurate structure recovery, especially when some of the signals in certain regions are relatively low.

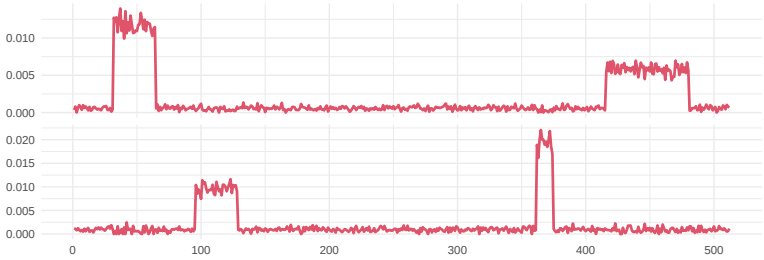
5.2.4 Application to GTEx RNA-seq data

We apply the SPNMF model to bulk RNA-seq data from GTEx. In a previous study, Dey et al. [2017] applied the topic model to GTEx RNA-seq V6 data on gene-level expression and visualized the sample memberships by structure plots. In this study, we analyze RNA-seq data at the base pair resolution within a gene region. Gene regions containing both exons and introns exhibit spatially-structured underlying gene expression. This organization also allows for complex regulation of gene expression, for example, through alternative splicing. Thus, by incorporating the smoothness constraint on the factor, we hope to identify potential alternative splicing patterns of a gene. Specifically, we look at three genes that exhibit differentially expressed patterns between muscle and brain tissues.

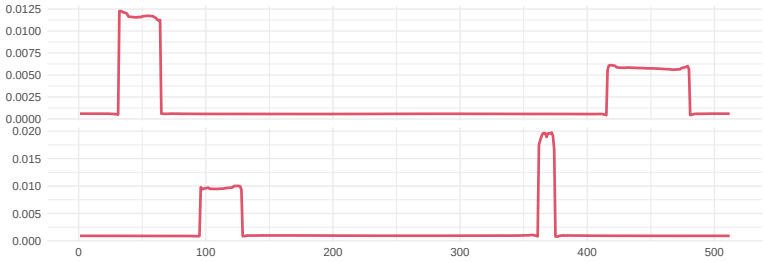
Figure 5.3a shows the membership of each sample in each of the three factors of gene *PKM*. Samples are grouped by tissues, as shown on the left sidebar. Clearly, the muscle skeletal tissues have much less membership in factor 3 compared to other tissues. The factor 3 and



(a) True factors.

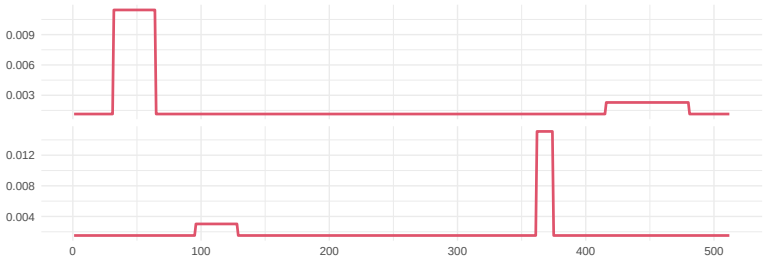


(b) Fitted factors from vanilla NMF.



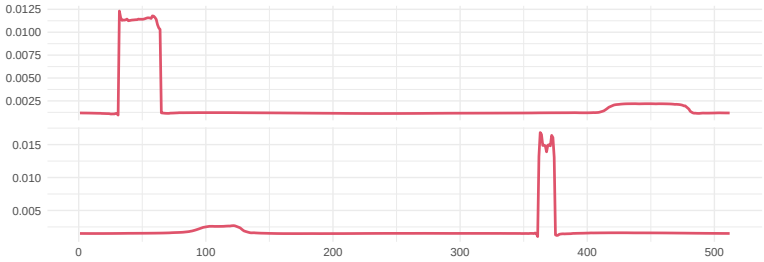
(c) Fitted factors from SPNMF.

Figure 5.1: An illustrative example of SPNMF. The shorter step size is 5, a half of the larger one.



(a) True factors.

(b) Fitted factors from vanilla NMF.



(c) Fitted factors from SPNMF.

Figure 5.2: An illustrative example of SPNMF. The shorter step size is 1, which is one fifth of the larger one.

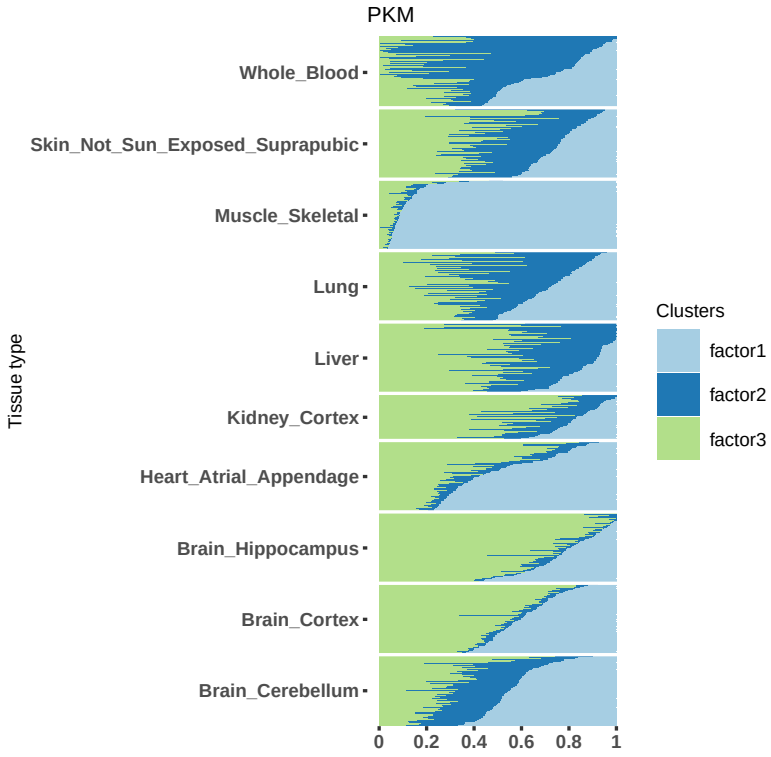
factor 1 mainly differ in the presence of the third and fourth exon. The presence of the fourth exon is related to muscle functionality. Similar contrasts between muscle and brain tissues can also be found in gene *RTN2* and *NDUFA3*, and the results are shown in Figure 5.4 and Figure 5.5. Interestingly, factor 2 in Figure 5.3b shows a potential intron retention pattern between the second and third exons. Similar potential intron retentions are also observed in factor 2, as shown in Figure 5.4b, and in factor 3, as depicted in Figure 5.5b.

5.3 Biwhitening EBNMF

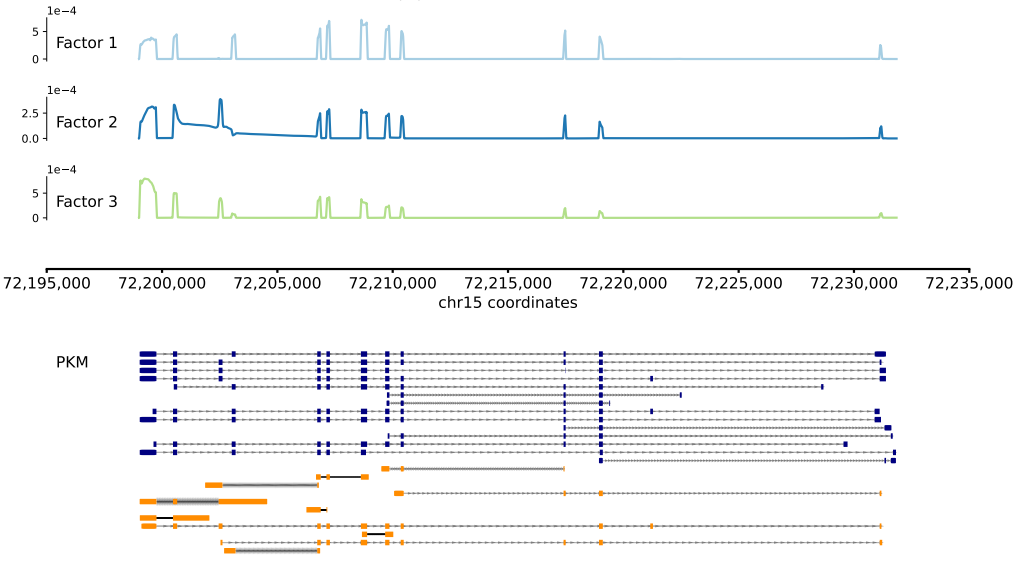
Non-negative matrix factorization (NMF), or Poisson matrix factorization (PMF), constrains the low-dimensional structures to be non-negative. It has been shown that NMF produces part-based decomposition, thus improving interpretability (Lee and Seung [1999]). Recent work on NMF has extended the constraints to allow sparsity in latent factors (Hoyer [2004]). Commonly used penalties for sparsity include the l_1 penalty, with a tuning parameter controlling the level of sparsity. However, tuning the parameter itself is non-trivial, and in real applications, it is more flexible to assume different sparsity levels for each factor, requiring multiple tuning parameters. This, in turn, introduces further difficulties in parameter tuning.

Empirical Bayes matrix factorization (EBMF, Wang and Stephens [2021]) is a flexible framework for sparse matrix factorization. It allows a wide range of priors on both loadings and factors, such as sparse, non-negative, and spatial priors. We refer to the EBMF with non-negative (or sparse and non-negative) priors as EBNMF (Willwerscheid [2022]). However the method was originally developed for Gaussian likelihood, and direct extension to Poisson likelihood is non-trivial. This is mainly due to the non-quadratic nature of the Poisson log-likelihood, as well as the identity link function used in the model.

In this work, we combine the biwhitening method proposed by Landa et al. [2022], and the

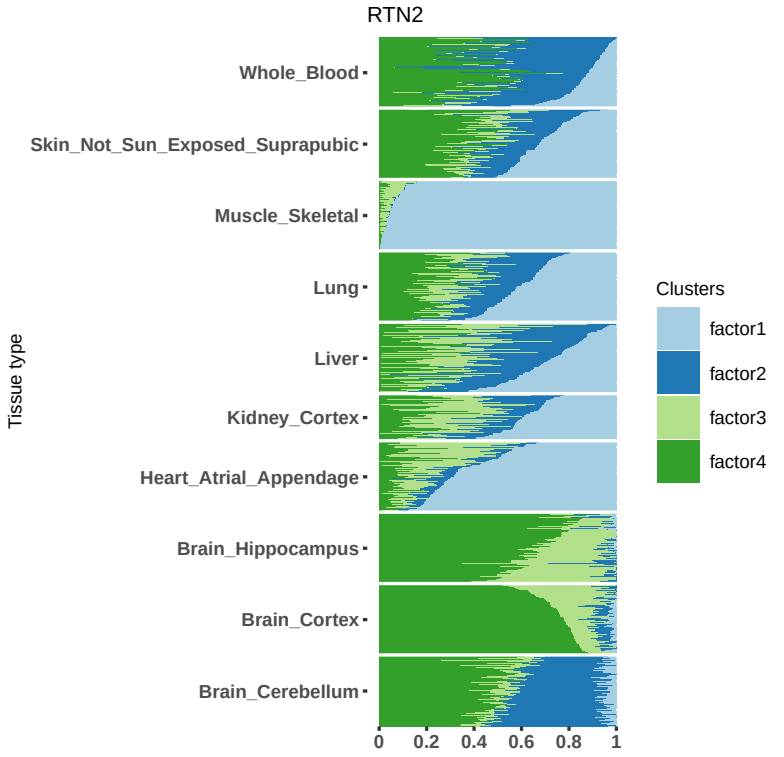


(a) Structure plot.

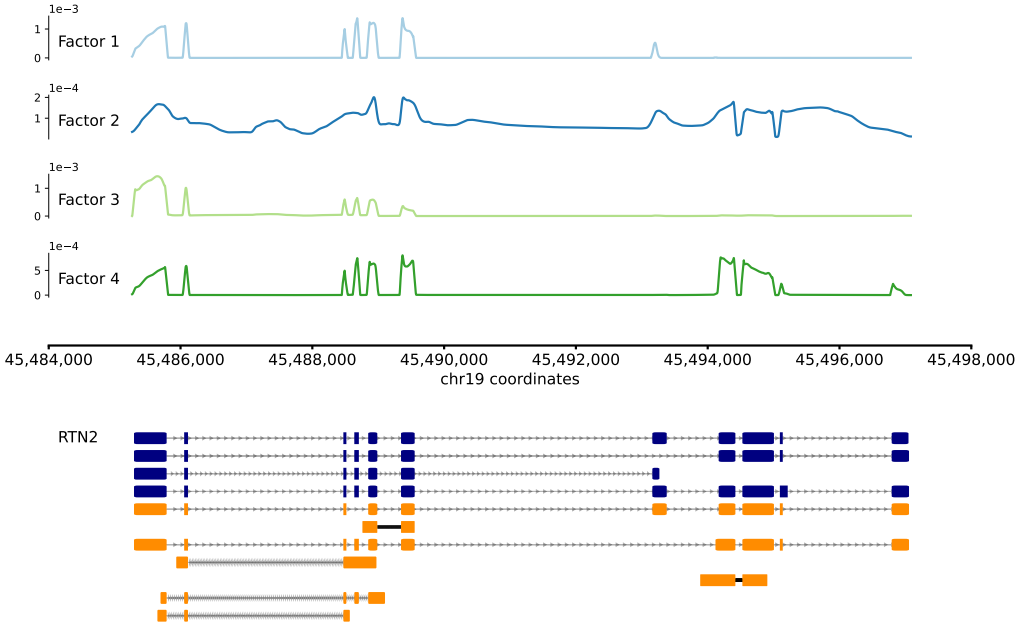


(b) Factors. The bottom part shows the annotated isoform of the gene.

Figure 5.3: SPNMF fit on gene *PKM*, $K = 3$.

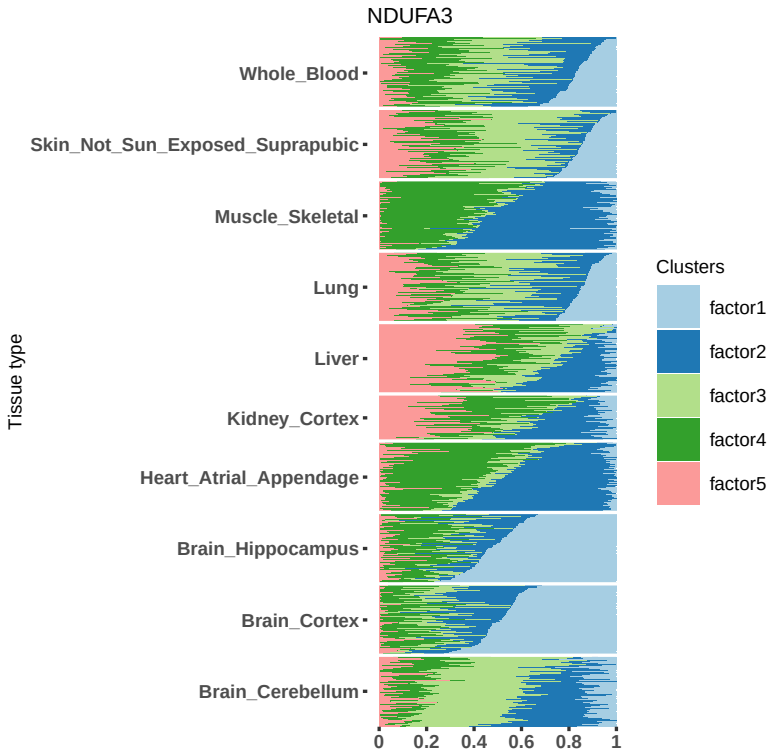


(a) Structure plot.

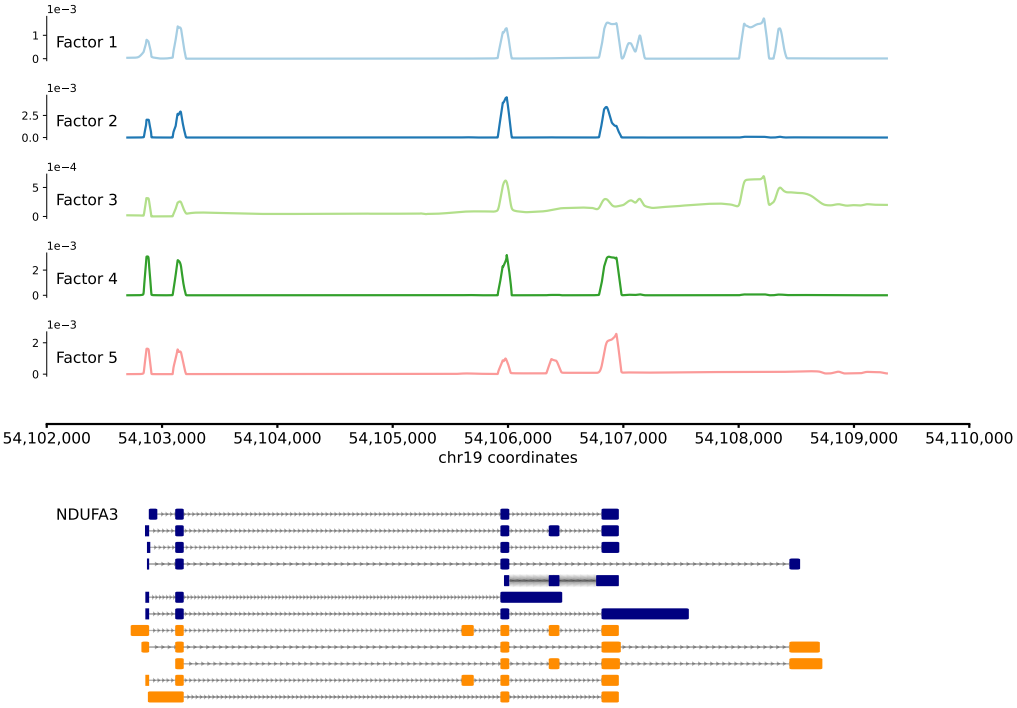


(b) Factors. The bottom part shows the annotated isoform of the gene.

Figure 5.4: SPNMF fit on gene *RTN2*, $K = 4$.



(a) Structure plot.



(b) Factors. The bottom part shows the annotated isoform of the gene.

Figure 5.5: SPNMF fit on gene *NDUFA3*, $K = 5$.

EBMF to develop a flexible framework for sparse Poisson matrix factorization (with identity link function).

5.3.1 Biwhitening

The method biwhitening was original proposed by Landa et al. [2022] to reveal the rank of a Poisson matrix, or matrix of other distributions that satisfy a quadratic relation between the mean and variance. The method involves scaling the rows and columns of the count matrix so that the corresponding noise spectrum agrees with the Marchenko-Pastur law. The scaling factors can be estimated directly from the observations using the Sinkhorn-Knopp algorithm.

Consider the model for a random matrix $Y \in \mathbb{R}^{n \times p}$,

$$Y = X + E, \tag{5.10}$$

where $\text{rank}(X) = r$, E is a noise matrix with mean 0. When the noise variables are heteroskedastic, the rows and columns are scaled such that the noise components are roughly homogeneous. Specifically, let $\boldsymbol{u}$ and $\boldsymbol{v}$ be positive vectors, then the scaled matrix is given as

$$\tilde{Y} = D(\boldsymbol{u}) Y D(\boldsymbol{v}) = \tilde{X} + \tilde{E}, \tag{5.11}$$

where $D(\boldsymbol{u}) = \text{diag}(\boldsymbol{u})$ and $D(\boldsymbol{v}) = \text{diag}(\boldsymbol{v})$. The scaling preserves the rank of matrix X, and the scaled noise random variables $\tilde{e}_{ij}$ are still independent with mean 0. Though the variances of $\tilde{e}_{ij}$ are still not all equal, the scaling procedure can ensure that the mean variance in each row and each column to be 1. Specifically for Poisson matrix, we find $\boldsymbol{u}$ and $\boldsymbol{v}$ such that $\sum_{i=1}^{n} u_i^2 X_{ij} v_j^2 = n$, and $\sum_{j=1}^{p} u_i^2 X_{ij} v_j^2 = p$. The Sinkhorn-Knopp algorithm for finding such scaling vectors are given in Algorithm 2.1 in Landa et al. [2022].

The biwhitening method can be further generalized to the family of distributions that satisfies a quadratic relation between the mean and the variance. For example, negative binomial and gamma distribution. Refer to Algorithm 5.1 in Landa et al. [2022] for finding the optimal mean-variance relationship, and the scaling vectors.

The distribution of transformed error $\tilde{e}_{ij}$ is not guaranteed to be Gaussian theoretically, but its entries have mean 0 and roughly homogeneous variances (the mean variance in each row and each column is 1). To have a better understanding of the error distributions, we have performed empirical study to show that the distribution of $\tilde{e}_{ij}$ is close to Gaussian, when the mean parameter is large. We use the simulation setting in SM4.2. for producing Figures 2 and 3 in Landa et al. [2022], with $n = 50, p = 100$. The mean matrix X is generated by independent sampling from Unif$(1, 2)$, then multiplied the resulting matrix from left and right by diagonal matrices whose diagonal entries are sampled independently from exp(Unif$(-2, 2)$). The entries of matrix Y are sampled from Poisson distribution with mean X. We repeat the sampling of Y 1000 times for a given X, and each time the biwhitening transformation is applied to Y.

In Figure 5.6, we plot the of Poisson variance (entries of X), and the noise variance (estimated using 1000 repetitions) after biwhitening, as well as the histogram of transformed noise variances. We can see that after biwhitening, the variances are more homogeneous and concentrate around 1. In Figure 5.7, we plot the histograms of $\tilde{y}_{ij}$, with different mean parameter x_{ij}. The normal density curves with sample mean and variance are overlaid on the histogram. When the original mean parameter x is greater than 3, $\tilde{y}$ is approximately normal distributed; while when the mean parameter is smaller, there are many zero's in the corresponding entries hence in the $\tilde{y}$.

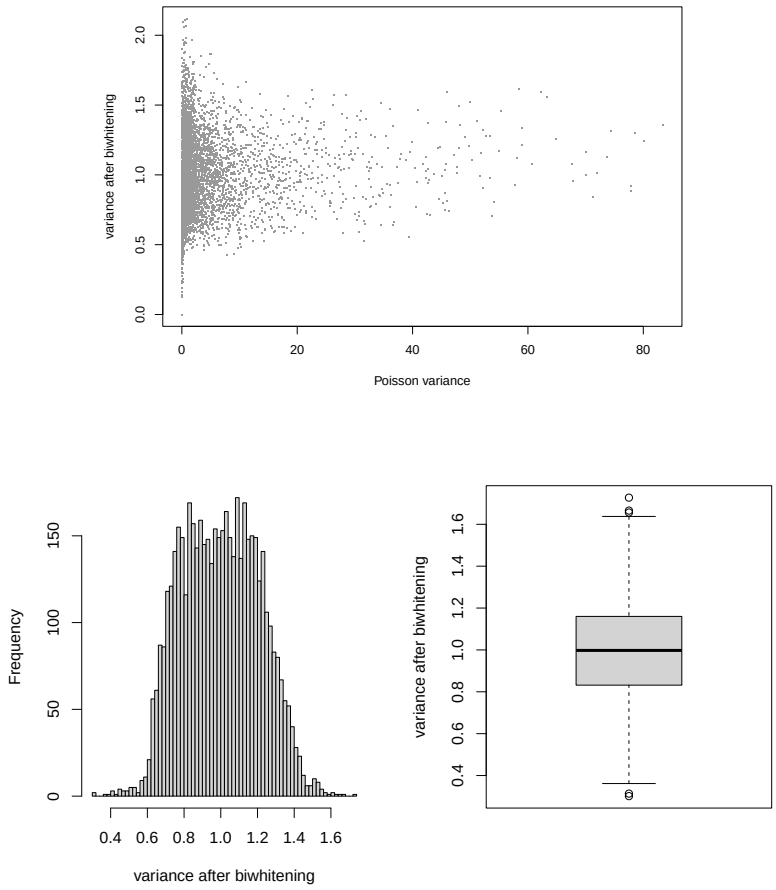


Figure 5.6: Biwhitening Poisson matrix. Upper: scatter plot of Poisson variance (entries of X), and the noise variance (estimated using 1000 repetitions) after biwhitening. Lower: histogram and boxplot of biwhitened noise variances

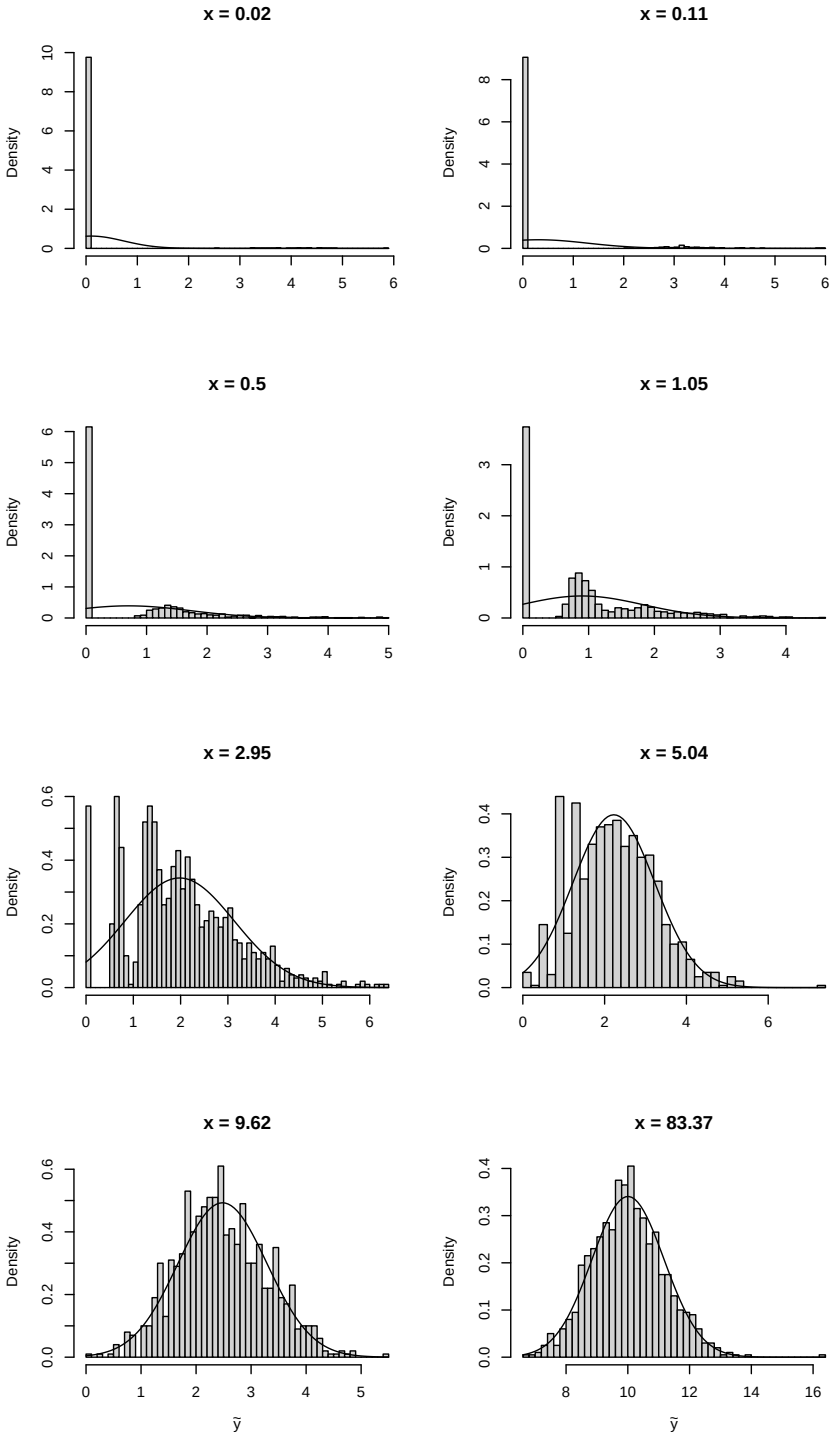


Figure 5.7: Histograms of $\tilde{y}_{ij}$, the transformed Poisson entries by biwhitening, with different mean parameter x_{ij}.

5.3.2 Biwhitening EBNMF

The biwhitening scales rows and columns such that the mean variance in each row and each column is 1. After the biwhitening transformation, we can then apply EBNMF to the matrix $\tilde{Y}$, to recover the non-negative(and/or sparse) loadings and factors. Specifically, the original Bayesian Poisson matrix factorization model is

$$\begin{aligned} y_{ij} &\sim \text{Poisson}\left(\sum_k l_{ik} f_{jk}\right), \\ \boldsymbol{l}_k &\sim g_{\boldsymbol{l}_k}(\cdot), \\ \boldsymbol{f}_k &\sim g_{\boldsymbol{f}_k}(\cdot). \end{aligned} \tag{5.12}$$

For Poisson distribution, the noise variables e_{ij} in the signal plus noise model (5.10) are centered Poisson with variance $\sum_k l_{ik} f_{jk}$.

Applying the biwhitening to the matrix Y, we work with the model

$$\begin{aligned} \tilde{Y} &= D(\hat{\boldsymbol{u}}) L F^T D(\hat{\boldsymbol{v}}) + \tilde{E}, \\ \tilde{e}_{ij} &\sim N(0, \sigma_{ij}^2), \\ \boldsymbol{l}_k &\sim g_{\boldsymbol{l}_k}(\cdot), \\ \boldsymbol{f}_k &\sim g_{\boldsymbol{f}_k}(\cdot), \end{aligned} \tag{5.13}$$

where $\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}$ are obtained from the Sinkhorn-Knopp algorithm. We can regard the biwhitening transformation as a variance stabilizing procedure, while it preserves the rank of the signal, and more importantly preserves the identity link function. Then we can apply EBNMF to the matrix $\tilde{Y}$, with row and column scaling factors $\hat{u}$, and $\hat{v}$. Biwhitening also preserves the 0's in the original data matrix so we can exploit the data sparsity for faster algorithms. Note that the model (5.13) is an extended version of EBMF by allowing the known row and column scaling matrices. The algorithm for the scaled EBMF is given in Appendix D.3. The

extension to allow smoothed loadings and/or factors are also straightforward, by combining the scaled EBMF with smoothness-inducing priors, as shown in Appendix D.2.

The model (5.13) can be further written as

$$
\begin{aligned}
&Y = LF^T + \mathcal{E}, \\
&\epsilon_{ij} \sim N(0, \sigma_{ij}^2/(\hat{u}_i^2 \hat{v}_j^2)), \\
&\boldsymbol{l}_k \sim g_{\boldsymbol{l}_k}(\cdot), \\
&\boldsymbol{f}_k \sim g_{\boldsymbol{f}_k}(\cdot),
\end{aligned}
\tag{5.14}
$$

where $\mathcal{E} = D^{-1}(\hat{\boldsymbol{u}})\tilde{E}D^{-1}(\hat{\boldsymbol{v}})$. In this model, the matrix Y is the original count matrix, while the errors are scaled by the $\hat{\boldsymbol{u}}$ and $\hat{\boldsymbol{v}}$. This formulation is simpler in that it involves merely altering the error structures based on a variance-stabilizing procedure, such as biwhitening. More broadly, one could propose different error structures. For instance, one might scale the error term with an estimate of the mean of Y.

5.3.3 Numerical examples

We illustrate the benefits of biwhitening EBNMF on a simple simulated example, and compare it with three other NMF models. The first one is the Poisson matrix factorization model studied in Carbonetto et al. [2021], and we use the function `fit_poisson_nmf` in R package `fastTopics`. The `fit_poisson_nmf` function does not make sparsity assumption on loadings and factors. The second NMF model is studied in DeBruine et al. [2021], and we use the function `nmf` in R package `RcppML`. The function has an option to regularize loadings and factors by l_1 penalty but only allows a single one for all loadings and another one for all factors. The last NMF implementation is due to Gaujoux and Seoighe [2010], and we use the function `nmf` in R package `NMF`. Specifically the algorithm "snmf/l" solves the sparse NMF model proposed in Kim and Park [2007]. The model uses l_1 penalty for inducing spar-

sity: $\min_{L,F} 1/2(||Y - LF^T||_F^2 + \eta||F||_F^2 + \beta(\sum_i ||L[i,:]||_1^2))$ s.t. $L \geq 0, F \geq 0$. However the function does not support l_1 penalization on both loadings and factors, and difference loss function than the quadratic loss. To distinguish the two NMF functions, we denote them as `RcppML::nmf` and `NMF::nmf`.

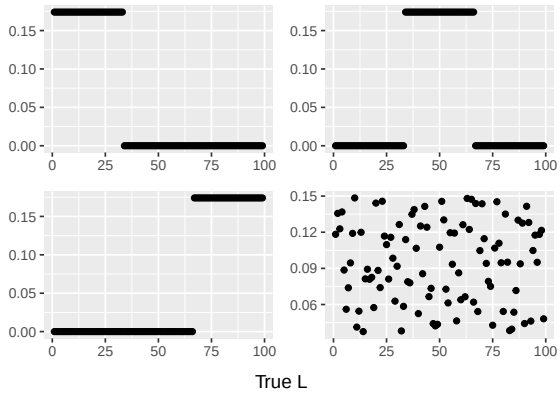
We generate the count matrix as follows. The matrix dimension is set to be $n = 99, p = 300$, and there are $K = 4$ latent factors. The first three loadings are sparse, with non-zero elements $\boldsymbol{l}_{1,1:33}, \boldsymbol{l}_{2,34:66}, \boldsymbol{l}_{3,67:99}$ all being 1's and all other elements are 0's. The elements of the fourth loading are sampled from $1+\alpha \times \text{Unif}(0,1)$, where α is a scalar that controls the proportion of variance explained (PVE) of the fourth factor. For the factors, the first three are sparse, with non-zero elements $\boldsymbol{f}_{1,1:100}, \boldsymbol{f}_{2,101:200}, \boldsymbol{f}_{3,201:300}$ sampled from $1+10 \times \text{Unif}(0,1)$, and all other elements are 0's. The elements of fourth factor are sampled from $1 + \alpha \times \text{Unif}(0,1)$.

For the two NMF models with sparsity constraints, we ran each model 10 times with different initialization seed, and keep the one with smallest squared error. For `RcppML::nmf`, we set the both l_1 penalty parameters for L and F to be 0.99 (the allowed range is $[0, 1)$). For `NMF::nmf`, we set the l_1 penalty for loading to be $\beta = 0.5$. When running the function, out of the 10 runs, 8 of them are not converged, and the warning message says "Too many restarts due to too big beta value". We set $\alpha = 3$. After the model fittings, we scale the factors such that each factor $\boldsymbol{f}_k$ has norm 1, and multiply the corresponding scaling scalar for each factor to each $\boldsymbol{l}_k$.

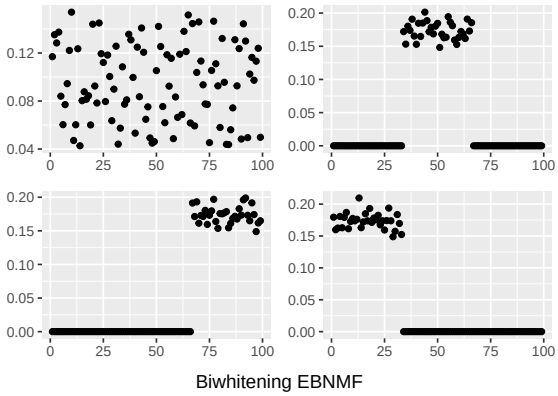
Figure 5.8 shows the true and estimated loadings. The estimated L and F from Poisson matrix factorization are not sparse because there's no such penalization in `fit_poisson_nmf`. The biwhitening EBNMF is able to recover the sparse loadings and the sparsity patterns match the true ones. It also stops adding factors at $K = 4$ thus also selects the true K automatically. For `RcppML::nmf`,though we have set the l_1 penalty to be near the upper bound of the allowed range, the estimated L and F from `RcppML::nmf` are not sparse. For

`NMF::nmf`, it gives sparse loadings with $\beta = 0.5$. However setting β to be smaller will not give sparse estimates. Thus the choice of β is important but the package does not have built-in method for selecting it. The next Figure 5.9 shows the true and estimated factors. Only the biwhitening EBNMF method was able to recover sparse factors, whereas all other methods failed, regardless of whether a penalty was applied or not. We also repeat the experiment with $\alpha = 5$, and similar results are observed.

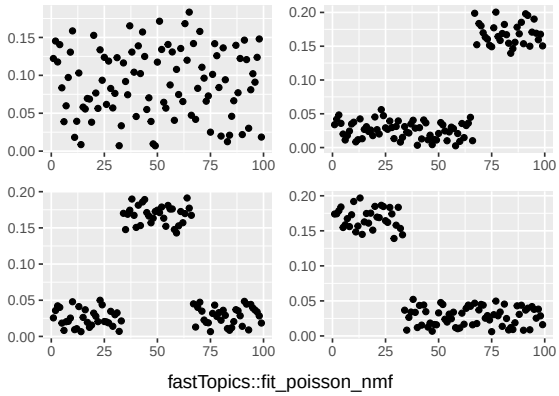
In the second numerical study, we revisit the SPNMF example outlined in section 5.2.3. The data generating process remains the same as in the previous section, but we fit the biwhitening EBNMF model with a point exponential prior on loadings and a wavelet prior on factors. It is important to note that the wavelet prior is a smoothness-inducing prior, but it does not guarantee non-negativity of the factors. For comparison purposes, we have included results where both the loadings and factors have point exponential priors. We opt for the NDWT version of the "smash" method to smooth the factors when running simulations with larger signals, as the NDWT method generally offers a better fit. While this method lacks a clear objective function, we found in simulations with larger signals that the ELBO typically increases after every step in greedy and back-fitting in EBMF. However, this was not the case with simulations involving smaller signals, where we ultimately chose to use the DWT method to smooth factors. In Figure 5.10, the biwhitening EBNMF with a wavelet prior on factors successfully recovers the first factor. However, for the second factor, the estimated factor displays two "lower" negative regions that correspond to the first factor. This is likely due to the unconstrained smoothing of the factors, which does not guarantee non-negativity. We observe that the biwhitening EBNMF with sparse non-negative priors successfully recovers the true factors' patterns, albeit less smoothly. Similar observations can be seen in Figure 5.11, where the smoothing helps visualize the small jumps in the factors.



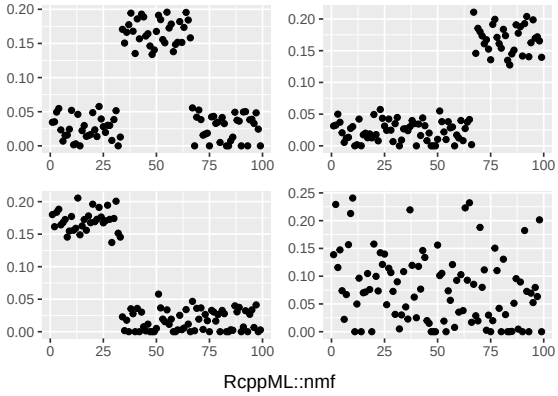
(a) True loadings



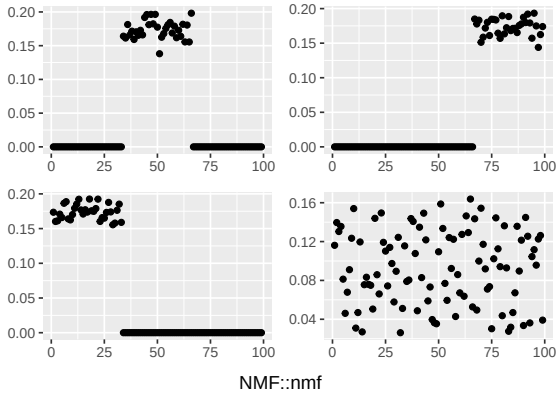
(b) Biwhitening EBNMF



(c) PMF

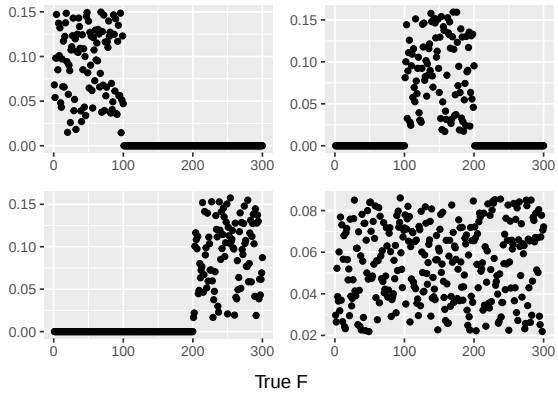


(d) Sparse NMF from RcppML package

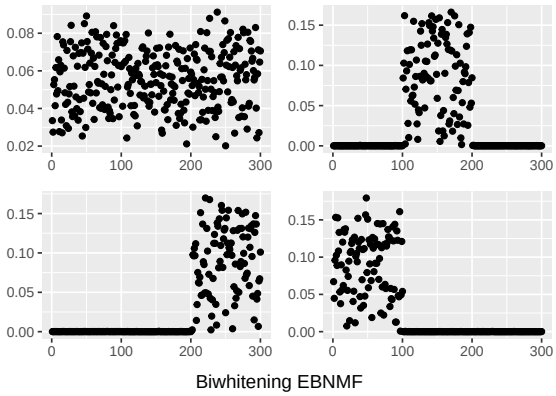


(e) Sparse NMF from NMF package

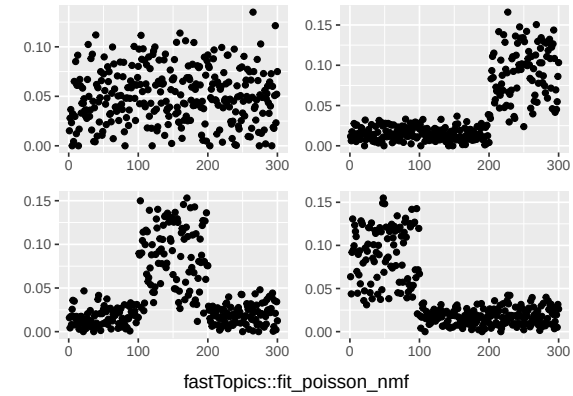
Figure 5.8: Simulation example of biwhitening EBNMF. Plot of estimated loadings from comparing methods.



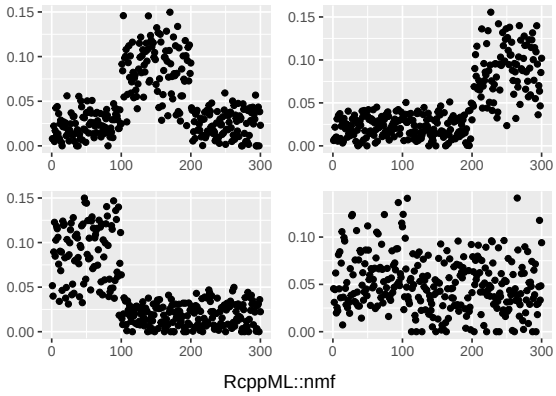
(a) True factors



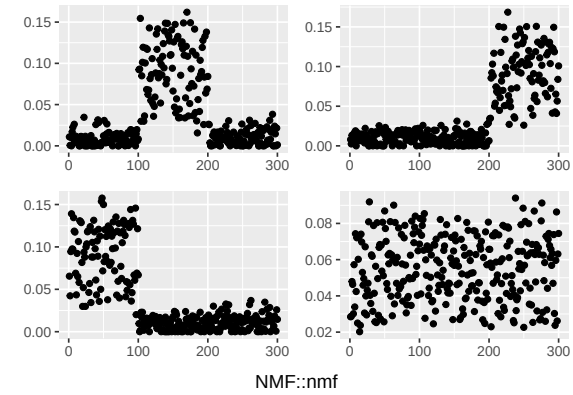
(b) Biwhitening EBNMF



(c) PMF



(d) Sparse NMF from RcppML package



(e) Sparse NMF from NMF package

Figure 5.9: Simulation example of biwhitening EBNMF. Plot of estimated factors from comparing methods.

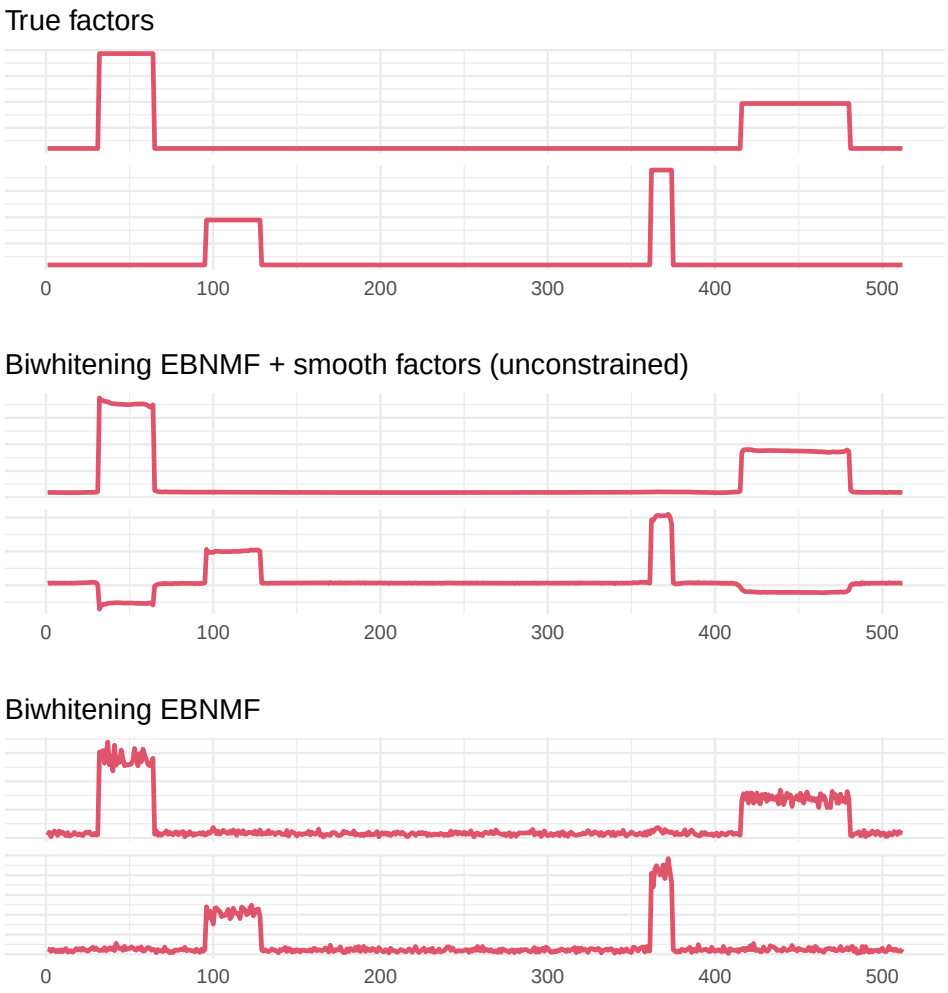


Figure 5.10: Simulation example of biwhitening EBNMF with smooth factors (unconstrained). In the true factor, the shorter step size is 5, a half of the larger one.

Figure 5.11: Simulation example of biwhitening EBNMF with smooth factors (unconstrained). In the true factor, the shorter step size is 1, one fifth of the larger one.

5.3.4 GTEx data analysis

We compared the biwhitening EBNMF with Poisson NMF fit on GTEx V8 data (dbGaP Accession phs000424.v8.p2). The dataset was downloaded from GTEx portal, and we analyzed a subset of genes studied in Dey et al. [2017] and they are available at `http://stephenslab.github.io/count-clustering/project/utilities/gene_names_all_gtex.txt`. The final data matrix has $15,153$ gene expressions on $17,382$ samples. The samples are from 30 main tissues (and 54 tissues with finer classification).

We fit Poisson NMF with $K = 20$ using `fit_poisson_nmf` function, and biwhitening EBNMF with Kmax = 20. The current implementation of biwhitening EBNMF does not support back-fitting so the results are from greedy fitting only. For visualization, we use the structure plot in Dey et al. [2017]. Originally the structure plot is for the loading (membership) matrix in a multinomial topic model. Here we show the structure plot for the loading matrix, after scaling each factor to have unit norm. For biwhitening EBNMF, we remove the first factor because it corresponds to a mean factor. Figure 5.12 shows the structure plots of estimated loadings by biwhitening EBNMF, and Poisson NMF. For tissues that clearly have their corresponding factor, for example muscle and pancreas, both methods are able to highlight the factor. The biwhitening EBNMF seems to find more mixture components. For example, there are mixture of different factors for cultured fibroblasts, and whole blood found by biwhitening EBNMF while the Poisson NMF finds one dominating factor for these two tissues. For some tissues, due to sparsity constraint, the biwhitening EBNMF is able to give a clearer visualization of the membership. For example the artery tissue.

We further ran the models on brain tissue samples only, with 15,147 gene expression in 2,642 samples. There are total 13 tissues from brain. We fit topic model with $K = 6$ (as done in Dey et al. [2017]), and biwhitening EBNMF with Kmax = 6. Again we removed the mean factor from biwhitening EBNMF for visualization. Overall the mixture profiles learned by

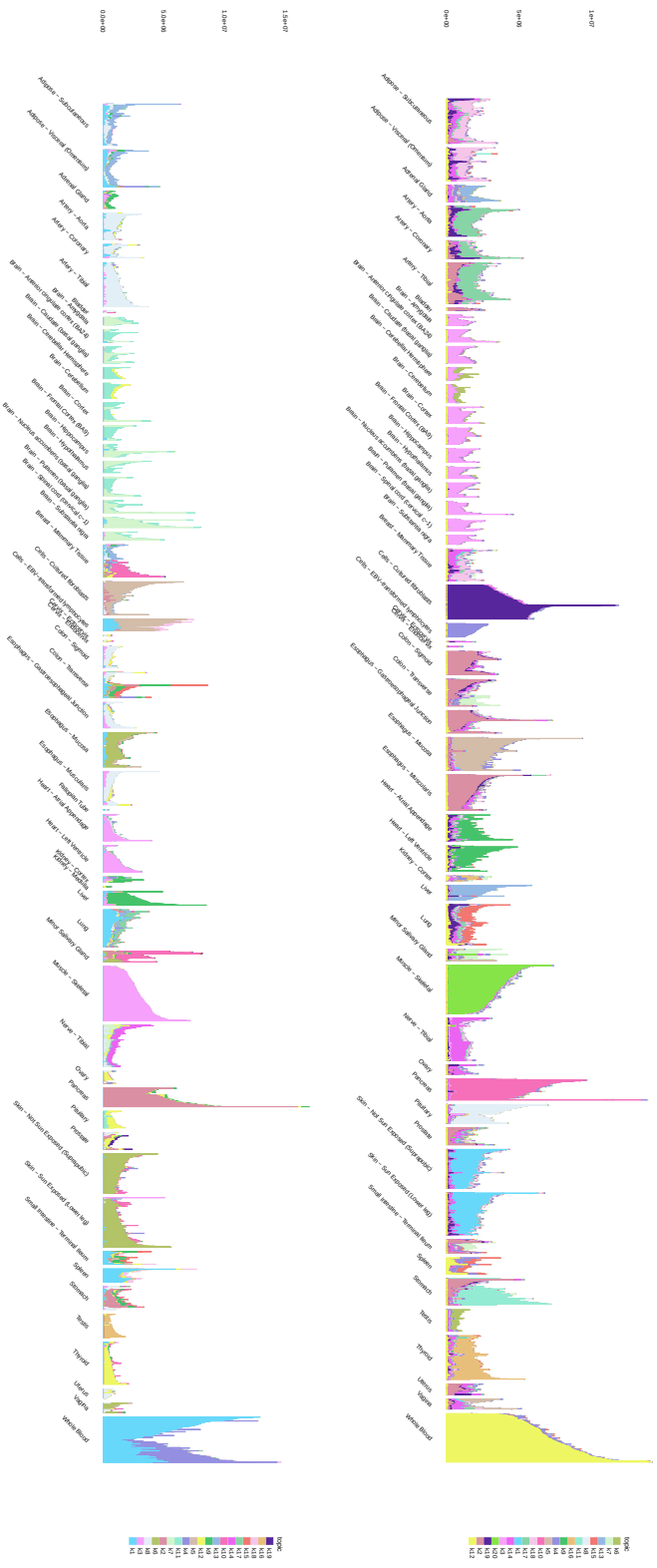


Figure 5.12: Structure plot of estimated loadings by biwhitening EBNMF (left), and Poisson NMF (right) on GTEx V8 data.

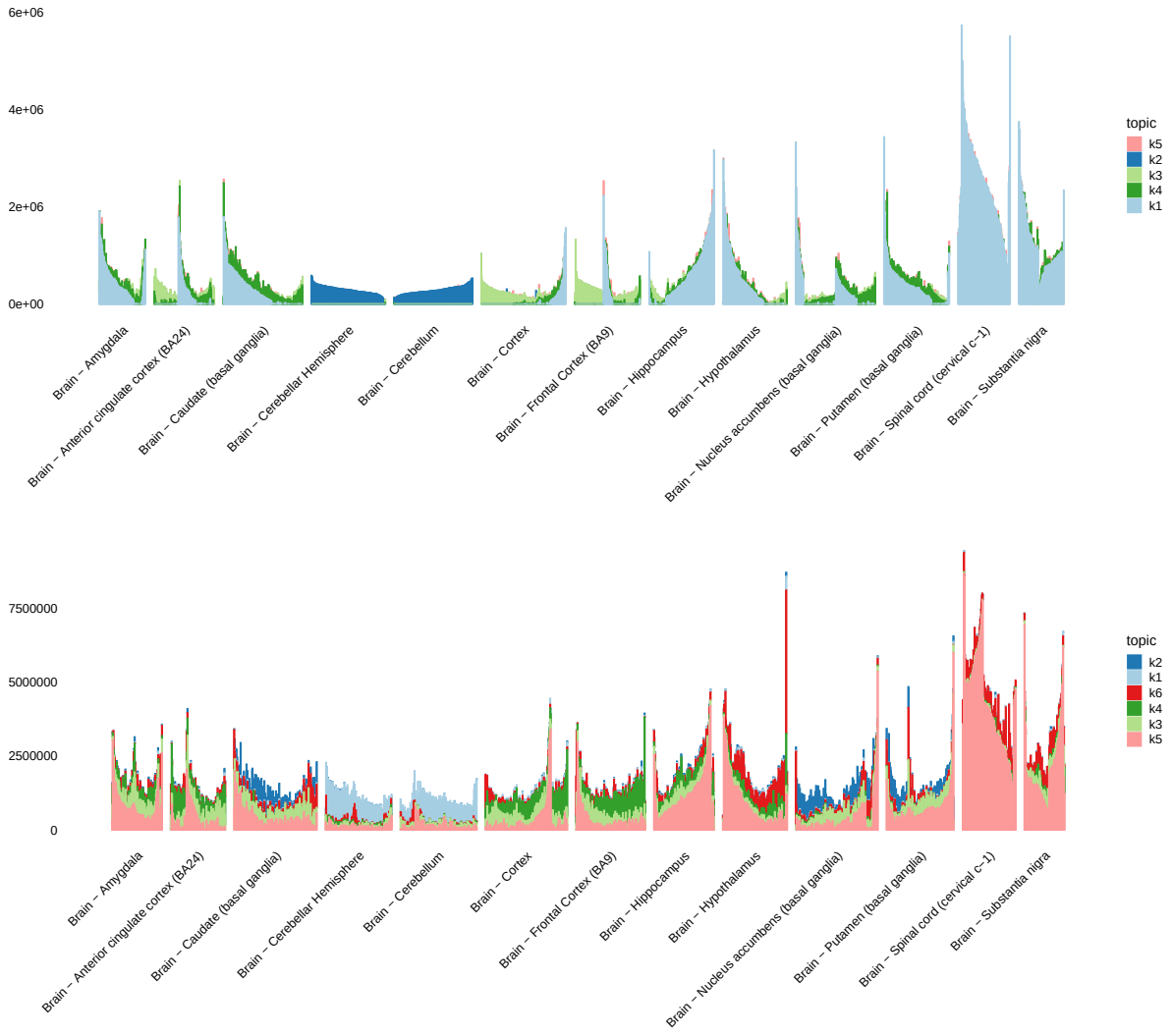


Figure 5.13: Structure plot of estimated loadings by biwhitening EBNMF (top), and Poisson NMF (bottom) on brain tissues.

the two models are similar. On the other hand, due to the sparsity constraint, we can clearly read from the top structure plot in Figure 5.13 that the factor 3 is primarily for Cerebellar Hemisphere and Cerebellum.

5.4 Discussion

In this chapter, we examined two methods for EB Poisson matrix factorization, and demonstrated how both frameworks can be extended to accommodate spatially-structured loadings

and/or factors. The first method EBPMF directly models count data, and it decomposes the matrix factorization problem into EBPM problems. The second method builds upon a recently proposed biwhitening approach for estimating the rank of a count matrix, and we have shown that it can be potentially viewed as a variance stabilizing transformation. This reduces the Poisson matrix factorization problem to a Gaussian one, allowing us to apply the EBNMF method to the transformed data. We have also illustrated the performance of these models through simulation studies and real data analysis. By incorporating sparse or smooth constraints on the latent structures, we have been able to achieve more interpretable results and more visually appealing visualizations.

We have employed the wavelet denoising method when the factors are assumed to be smooth. This method has been extensively studied and well established, and is particularly attractive due to its fast computation and local adaptability. However, it also carries certain assumptions that limit its application to matrix factorization problems. Specifically, the observations should be equally spaced, and their length must be a power of 2. When these assumptions are violated, one can employ interpolation and/or reflection. Additionally, it is typically assumed that the variance of each observation is constant, ensuring that the empirical wavelet coefficients are homogeneous and independent. In the presence of heterogeneity, the wavelet transformation leads to dependent empirical wavelet coefficients. When applying the empirical Bayes shrinkage method to empirical wavelet coefficients, a common approach is to ignore the correlations. However, all these methods for rectifying assumption violations are approximations and they modify the original objective function. Consequently, the overall ELBO in the EB matrix factorization problem may not increase after each update in the variational inference algorithm, potentially leading to issues in model selection and validation. Therefore, a future direction could involve developing other EB normal smoothing methods, such as the Gaussian process (Seeger [2004]), trend filtering (Kim et al. [2009]), and splines (Perperoglou et al. [2019]).

Code and data availability

The SPNMF method is implemented in `R` package `ebpmf` at `https://github.com/DongyueXie/ebpmf`. The biwhitening methods are implemented in `R` package, available at `https://github.com/DongyueXie/funflash`.

Code for understanding biwhitening is at `https://github.com/DongyueXie/SMF/blob/master/analysis/biwhitening.Rmd`. Code for producing the comparison between biwhitening EBNMF and other methods is at `https://github.com/DongyueXie/SMF/blob/master/analysis/biwhitening_PMF.Rmd`. Code for producing the biwhitening smoothed EBNMF is at `https://github.com/DongyueXie/SMF/blob/master/analysis/biwhitening_SPMF.Rmd`.

The GTEx data are downloaded from GTEx portal with the release version V8. A detailed preparation guide is at `https://github.com/DongyueXie/SMF/blob/master/analysis/gtex_v8.Rmd`. Code for running the GTEx analysis is at `https://github.com/DongyueXie/SMF/blob/master/code/GTEx_V8_thesis.R`, and the for producing the structure plots is at `https://github.com/DongyueXie/SMF/blob/master/analysis/biwhitening_ebnmf_gtex.Rmd`.

REFERENCES

Constantin Ahlmann-Eltze and Wolfgang Huber. Comparison of transformations for single-cell rna-seq data. *Nature Methods*, pages 1–8, 2023.

John Aitchison and CH Ho. The multivariate poisson-log normal distribution. *Biometrika*, 76(4):643–653, 1989.

Robert A Amezquita, Aaron TL Lun, Etienne Becht, Vince J Carey, Lindsay N Carpp, Ludwig Geistlinger, Federico Marini, Kevin Rue-Albrecht, Davide Risso, Charlotte Soneson, et al. Orchestrating single-cell analysis with bioconductor. *Nature methods*, 17(2):137–145, 2020.

Francis J Anscombe. The transformation of poisson, binomial and negative-binomial data. *Biometrika*, 35(3/4):246–254, 1948.

Anestis Antoniadis, Jeremie Bigot, and Theofanis Sapatinas. Wavelet estimators in nonparametric regression: a comparative simulation study. *Journal of statistical software*, 6: 1–83, 2001.

Ricard Argelaguet, Britta Velten, Damien Arnol, Sascha Dietrich, Thorsten Zenz, John C Marioni, Florian Buettner, Wolfgang Huber, and Oliver Stegle. Multi-omics factor analysis—a framework for unsupervised integration of multi-omics data sets. *Molecular systems biology*, 14(6):e8124, 2018.

Simon R Arridge, Kazufumi Ito, Bangti Jin, and Chen Zhang. Variational gaussian approximation for poisson data. *Inverse Problems*, 34(2):025005, 2018.

MSo Bartlett. The square root transformation in analysis of variance. *Supplement to the Journal of the Royal Statistical Society*, 3(1):68–78, 1936.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

Matteo Borella, Graziano Martello, Davide Risso, and Chiara Romualdi. Psinorm: a scalable normalization for single-cell rna-seq data. *Bioinformatics*, 38(1):164–172, 2022.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Philip Brennecke, Simon Anders, Jong Kyoung Kim, Aleksandra A Kołodziejczyk, Xiuwei Zhang, Valentina Proserpio, Bianka Baying, Vladimir Benes, Sarah A Teichmann, John C Marioni, et al. Accounting for technical noise in single-cell rna-seq experiments. *Nature methods*, 10(11):1093–1095, 2013.

Lawrence D Brown and Michael Levine. Variance estimation in nonparametric regression via the difference sequence method. *The Annals of Statistics*, 35(5):2219–2232, 2007.

Richard H Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on scientific computing*, 16(5):1190–1208, 1995.

Junyue Cao, Jonathan S Packer, Vijay Ramani, Darren A Cusanovich, Chau Huynh, Riza Daza, Xiaojie Qiu, Choli Lee, Scott N Furlan, Frank J Steemers, et al. Comprehensive single-cell transcriptional profiling of a multicellular organism. *Science*, 357(6352):661–667, 2017.

Peter Carbonetto, Abhishek Sarkar, Zihao Wang, and Matthew Stephens. Non-negative matrix factorization algorithms greatly improve topic model fits. *arXiv preprint arXiv:2105.13440*, 2021.

Peter Carbonetto, Kaixuan Luo, Abhishek Sarkar, Anthony Hung, Karl Tayeb, Sebastian Pott, and Matthew Stephens. Interpreting structure in sequence count data with differential expression analysis allowing for grades of membership. *bioRxiv*, pages 2023–03, 2023.

PC Carrasco. Nugget effect, artificial or natural? *Journal of the Southern African Institute of Mining and Metallurgy*, 110(6):299–305, 2010.

Ali Taylan Cemgil. Bayesian inference for nonnegative matrix factorisation models. *Computational intelligence and neuroscience*, 2009, 2009.

Julien Chiquet, Mahendra Mariadassou, and Stéphane Robin. Variational inference for probabilistic poisson pca. *The Annals of Applied Statistics*, 12(4):2674–2698, 2018.

Julien Chiquet, Mahendra Mariadassou, and Stéphane Robin. The poisson-lognormal model as a versatile framework for the joint analysis of species abundances. *Frontiers in Ecology and Evolution*, 9:588292, 2021.

Ronald R Coifman and David L Donoho. *Translation-invariant de-noising*. Springer, 1995.

Zachary J DeBruine, Karsten Melcher, and Timothy J Triche Jr. Fast and robust non-negative matrix factorization for single-cell experiments. *bioRxiv*, pages 2021–09, 2021.

Kushal K Dey, Chiaowen Joyce Hsiao, and Matthew Stephens. Visualizing the structure of rna-seq expression data using grade of membership models. *PLoS genetics*, 13(3):e1006599, 2017.

David L Donoho and Iain M Johnstone. Minimax estimation via wavelet shrinkage. *The annals of Statistics*, 26(3):879–921, 1998.

David L Donoho and Jain M Johnstone. Ideal spatial adaptation by wavelet shrinkage. *biometrika*, 81(3):425–455, 1994.

D.L. Donoho. De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627, 1995. doi:10.1109/18.382009.

Daniele Durante and Tommaso Rigon. Conditionally conjugate mean-field variational bayes for logistic models. *Statistical science*, 34(3):472–485, 2019.

Zhana Duren, Xi Chen, Mahdi Zamanighomi, Wanwen Zeng, Ansuman T Satpathy, Howard Y Chang, Yong Wang, and Wing Hung Wong. Integrative analysis of single-cell genomics data by coupled nonnegative matrix factorizations. *Proceedings of the National Academy of Sciences*, 115(30):7723–7728, 2018.

Gökcen Eraslan, Lukas M Simon, Maria Mircea, Nikola S Mueller, and Fabian J Theis. Single-cell rna-seq denoising using a deep count autoencoder. *Nature communications*, 10 (1):390, 2019.

Chao Feng, Shufen Liu, Hao Zhang, Renchu Guan, Dan Li, Fengfeng Zhou, Yanchun Liang, and Xiaoyue Feng. Dimension reduction and clustering models for single-cell rna sequencing data: a comparative study. *International journal of molecular sciences*, 21(6):2181, 2020.

Hong-Ye Gao. Wavelet shrinkage estimates for heteroscedastic regression models. In *MathSoft*. Citeseer, 1997.

Theo Gasser, Lothar Sroka, and Christine Jennen-Steinmetz. Residual variance and residual pattern in nonlinear regression. *Biometrika*, 73(3):625–633, 1986.

Renaud Gaujoux and Cathal Seoighe. A flexible r package for nonnegative matrix factorization. *BMC bioinformatics*, 11(1):1–9, 2010.

Peter Hall, JW Kay, and DM Titterington. Asymptotically optimal difference-based estimation of variance in nonparametric regression. *Biometrika*, 77(3):521–528, 1990.

Trevor J Hastie. Generalized additive models. In *Statistical models in S*, pages 249–307. Routledge, 2017.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 2013.

Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, 5(9), 2004.

Mo Huang, Jingshu Wang, Eduardo Torre, Hannah Dueck, Sydney Shaffer, Roberto Bonasio, John I Murray, Arjun Raj, Mingyao Li, and Nancy R Zhang. Saver: gene expression recovery for single-cell rna sequencing. *Nature methods*, 15(7):539–542, 2018.

Tommi S Jaakkola and Michael I Jordan. A variational approach to bayesian logistic regression models and their extensions. In *Sixth International Workshop on Artificial Intelligence and Statistics*, pages 283–294. PMLR, 1997.

Iain M. Johnstone and Bernard W. Silverman. Empirical Bayes selection of wavelet thresholds. *The Annals of Statistics*, 33(4):1700 – 1752, 2005. doi:10.1214/009053605000000345. URL https://doi.org/10.1214/009053605000000345.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37:183–233, 1999.

Dragomirka Jovic, Xue Liang, Hua Zeng, Lin Lin, Fengping Xu, and Yonglun Luo. Single-cell rna sequencing technologies and applications: A brief overview. *Clinical and Translational Medicine*, 12(3):e694, 2022.

Toby Kenney, Hong Gu, and Tianshu Huang. Poisson pca: Poisson measurement error corrected pca, with application to microbiome data. *Biometrics*, 77(4):1369–1384, 2021.

Hyunsoo Kim and Haesun Park. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 23 (12):1495–1502, 2007.

Seung-Jean Kim, Kwangmoo Koh, Stephen Boyd, and Dimitry Gorinevsky. \ell_1 trend filtering. *SIAM review*, 51(2):339–360, 2009.

Youngseok Kim, Wei Wang, Peter Carbonetto, and Matthew Stephens. A flexible empirical bayes approach to multiple linear regression and connections with penalized regression. *arXiv preprint arXiv:2208.10910*, 2022.

Allon M Klein, Linas Mazutis, Ilke Akartuna, Naren Tallapragada, Adrian Veres, Victor Li, Leonid Peshkin, David A Weitz, and Marc W Kirschner. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell*, 161(5):1187–1201, 2015.

Eric D Kolaczyk. Bayesian multiscale models for poisson processes. *Journal of the American Statistical Association*, 94(447):920–933, 1999.

Dylan Kotliar, Adrian Veres, M Aurel Nagy, Shervin Tabrizi, Eran Hodis, Douglas A Melton, and Pardis C Sabeti. Identifying gene expression programs of cell-type identity and cellular activity with single-cell rna-seq. *Elife*, 8:e43803, 2019.

Boris Landa, Thomas TCK Zhang, and Yuval Kluger. Biwhitening reveals the rank of a count matrix. *SIAM Journal on Mathematics of Data Science*, 4(4):1420–1446, 2022.

Andrew J Landgraf and Yoonkyung Lee. Dimensionality reduction for binary data through the projection of natural parameters. *Journal of Multivariate Analysis*, 180:104668, 2020.

Charity W Law, Yunshun Chen, Wei Shi, and Gordon K Smyth. voom: Precision weights unlock linear model analysis tools for rna-seq read counts. *Genome biology*, 15(2):1–17, 2014.

Amanda J Lea, Jenny Tung, and Xiang Zhou. A flexible, efficient binomial mixed model for identifying differential dna methylation in bisulfite sequencing data. *PLoS genetics*, 11 (11):e1005650, 2015.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Lihua Lei and William Fithian. Adapt: an interactive procedure for multiple testing with side information. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(4):649–679, 2018.

Lihua Lei, Aaditya Ramdas, and William Fithian. A general interactive framework for false discovery rate control under structural constraints. *Biometrika*, 108(2):253–267, 2021.

Hanna Mendes Levitin, Jinzhou Yuan, Yim Ling Cheng, Francisco JR Ruiz, Erin C Bush, Jeffrey N Bruce, Peter Canoll, Antonio Iavarone, Anna Lasorella, David M Blei, et al. De novo gene signature identification from single-cell rna-seq with hierarchical poisson factorization. *Molecular systems biology*, 15(2):e8557, 2019.

Lydia T Liu, Edgar Dobriban, and Amit Singer. e pca: high dimensional exponential family pca. *The Annals of Applied Statistics*, 12(4):2121–2150, 2018.

Yusha Liu, Peter Carbonetto, Michihiro Takahama, Adam Gruenbaum, Dongyue Xie, Nicolas Chevrier, and Matthew Stephens. A flexible model for correlated count data, with application to analysis of gene expression differences in multi-condition experiments. *arXiv preprint arXiv:2210.00697*, 2022.

Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology*, 15(12):1–21, 2014.

Meng Lu, Jianhua Z Huang, and Xiaoning Qian. Sparse exponential family principal component analysis. *Pattern recognition*, 60:681–691, 2016.

Mengyin Lu. *Generalized Adaptive Shrinkage Methods and Applications in Genomics Studies*. The University of Chicago, 2018.

Daniel T Montoro, Adam L Haber, Moshe Biton, Vladimir Vinarsky, Brian Lin, Susan E Birket, Feng Yuan, Sijia Chen, Hui Min Leung, Jorge Villoria, et al. A revised airway epithelial hierarchy includes cftr-expressing ionocytes. *Nature*, 560(7718):319–324, 2018.

Peter J Park. Chip–seq: advantages and challenges of a maturing technology. *Nature reviews genetics*, 10(10):669–680, 2009.

Branca I Pereira, Roel PH De Maeyer, Luciana P Covre, Djamel Nehar-Belaid, Alessio Lanna, Sophie Ward, Radu Marches, Emma S Chambers, Daniel CO Gomes, Natalie E Riddell, et al. Sestrins induce natural killer function in senescent-like cd8+ t cells. *Nature immunology*, 21(6):684–694, 2020.

Aris Perperoglou, Willi Sauerbrei, Michal Abrahamowicz, and Matthias Schmid. A review of spline function procedures in r. *BMC medical research methodology*, 19(1):1–16, 2019.

Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association*, 108 (504):1339–1349, 2013.

Nicholas G Polson, James G Scott, and Brandon T Willard. Proximal algorithms in statistics and machine learning. 2015.

Jonathan K Pritchard, Matthew Stephens, and Peter Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000.

Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial intelligence and statistics*, pages 814–822. PMLR, 2014.

Ho Sung Rhee and B Franklin Pugh. Comprehensive genome-wide protein-dna interactions detected at single-nucleotide resolution. *Cell*, 147(6):1408–1419, 2011.

Davide Risso, Fanny Perraudeau, Svetlana Gribkova, Sandrine Dudoit, and Jean-Philippe Vert. A general and flexible method for signal extraction from single-cell rna-seq data. *Nature communications*, 9(1):284, 2018.

Matthew E Ritchie, Belinda Phipson, DI Wu, Yifang Hu, Charity W Law, Wei Shi, and Gordon K Smyth. limma powers differential expression analyses for rna-sequencing and microarray studies. *Nucleic acids research*, 43(7):e47–e47, 2015.

Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. edger: a bioconductor package for differential expression analysis of digital gene expression data. *bioinformatics*, 26(1): 139–140, 2010.

Tim Salimans, Diederik Kingma, and Max Welling. Markov chain monte carlo and variational inference: Bridging the gap. In *International conference on machine learning*, pages 1218–1226. PMLR, 2015.

Abhishek Sarkar and Matthew Stephens. Separating measurement and expression models clarifies confusion in single-cell rna sequencing analysis. *Nature genetics*, 53(6):770–777, 2021.

Matthias Seeger. Gaussian processes for machine learning. *International journal of neural systems*, 14(02):69–106, 2004.

Matthias Seeger and Guillaume Bouchard. Fast variational bayesian inference for non-conjugate matrix factorization models. In *Artificial Intelligence and Statistics*, pages 1012–1018. PMLR, 2012.

Jens Sjölund. A tutorial on parametric variational inference. *arXiv preprint arXiv:2301.01236*, 2023.

D Mikis Stasinopoulos and Robert A Rigby. Generalized additive models for location scale and shape (gamlss) in r. *Journal of Statistical Software*, 23:1–46, 2008.

Genevieve L Stein-O'Brien, Raman Arora, Aedin C Culhane, Alexander V Favorov, Lana X Garmire, Casey S Greene, Loyal A Goff, Yifeng Li, Aloune Ngom, Michael F Ochs, et al. Enter the matrix: factorization uncovers knowledge from omics. *Trends in Genetics*, 34 (10):790–805, 2018.

Matthew Stephens. False discovery rates: a new deal. *Biostatistics*, 18(2):275–294, 2017.

Kelly Street, F. William Townes, Davide Risso, and Stephanie Hicks. *scry: Small-Count Analysis Methods for High-Dimensional Data*, 2021. URL `https://bioconductor.org/packages/scry.html`. R package version 1.6.0.

Shiquan Sun, Yabo Chen, Yang Liu, and Xuequn Shang. A fast and efficient count-based matrix factorization method for detecting cell types from single-cell rnaseq data. *BMC systems biology*, 13(2):1–8, 2019.

Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B Tuch, Asim Siddiqui, et al. mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377–382, 2009.

Klaus E Timmermann and Robert D Nowak. Multiscale modeling and estimation of poisson processes with application to photon-limited imaging. *IEEE Transactions on Information Theory*, 45(3):846–862, 1999.

F William Townes, Stephanie C Hicks, Martin J Aryee, and Rafael A Irizarry. Feature selection and dimension reduction for single-cell rna-seq based on a multinomial model. *Genome biology*, 20(1):1–16, 2019.

Sarah M Urbut, Gao Wang, Peter Carbonetto, and Matthew Stephens. Flexible statistical methods for estimating and testing effects in genomic studies with multiple conditions. *Nature genetics*, 51(1):187–195, 2019.

Catalina A Vallejos, John C Marioni, and Sylvia Richardson. Basics: Bayesian analysis of single-cell sequencing data. *PLoS computational biology*, 11(6):e1004333, 2015.

Mark A van de Wiel, Dennis E Te Beest, and Magnus M Münch. Learning from a lot: Empirical bayes for high-dimensional model-based prediction. *Scandinavian Journal of Statistics*, 46(1):2–25, 2019.

Chong Wang and David M Blei. Variational inference in nonconjugate models. *arXiv preprint arXiv:1209.4360*, 2012.

Jingshu Wang, Mo Huang, Eduardo Torre, Hannah Dueck, Sydney Shaffer, John Murray, Arjun Raj, Mingyao Li, and Nancy R Zhang. Gene expression distribution deconvolution in single-cell rna sequencing. *Proceedings of the National Academy of Sciences*, 115(28): E6437–E6446, 2018.

Liguo Wang, Junsheng Chen, Chen Wang, Liis Uusküla-Reimand, Kaifu Chen, Alejandra Medina-Rivera, Edwin J Young, Michael T Zimmermann, Huihuang Yan, Zhifu Sun, et al. Mace: model based analysis of chip-exo. *Nucleic acids research*, 42(20):e156–e156, 2014.

Wei Wang and Matthew Stephens. Empirical bayes matrix factorization. *Journal of Machine Learning Research*, 22(120):1–40, 2021.

Jason Willwerscheid. *flashier: Empirical Bayes Matrix Factorization*, 2022. R package version 0.2.34.

Jason Willwerscheid and Matthew Stephens. ebnm: An r package for solving the empirical bayes normal means problem using a variety of prior families. *arXiv preprint arXiv:2110.00152*, 2021.

Daniela M Witten, Robert Tibshirani, and Trevor Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3):515–534, 2009.

Simon N Wood. Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(1):3–36, 2011.

Ruizhi Xiang, Wencan Wang, Lei Yang, Shiyuan Wang, Chaohan Xu, and Xiaowen Chen. A comparison for dimensionality reduction methods of single-cell rna-seq data. *Frontiers in genetics*, 12:646936, 2021.

Zhengrong Xing. *Poisson multiscale methods for high-throughput sequencing data*. PhD thesis, The University of Chicago, 2016.

Zhengrong Xing, Peter Carbonetto, and Matthew Stephens. Flexible signal denoising via flexible empirical bayes shrinkage. *Journal of Machine Learning Research*, 22(93):1–28, 2021.

Grace XY Zheng, Jessica M Terry, Phillip Belgrader, Paul Ryvkin, Zachary W Bent, Ryan Wilson, Solongo B Ziraldo, Tobias D Wheeler, Geoff P McDermott, Junjie Zhu, et al. Massively parallel digital transcriptional profiling of single cells. *Nature communications*, 8(1):14049, 2017.

Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006.

APPENDIX A

DERIVATIONS OF VEBPM

A.1 Variational Gaussian posterior approximation for Poisson data

We review the variational Gaussian posterior approximation (VGA) of Poisson data. In this sub-problem, the prior distribution on μ_j is Gaussian with known mean and variance, and the posterior distribution is restricted to a Gaussian distribution. For simplicity we drop the subscript j and focus on the univariate distribution. The model is

$$\begin{aligned} y|\mu &\sim \text{Poisson}(s\exp(\mu)), \\ \mu &\sim N(\theta, \sigma^2), \end{aligned} \tag{A.1}$$

where s, θ and σ^2 are known scalars, and $s > 0$.

We restrict the posterior distribution of μ to be a Gaussian distribution, $q(\mu) = N(\mu; \bar{\mu}, v)$. The posterior mean and variance are estimated by maximizing the ELBO

$$\hat{\bar{\mu}}, \hat{v} = \arg\max_{\bar{\mu}, v} F(\bar{\mu}, v), \tag{A.2}$$

where

$$\begin{aligned} F(\bar{\mu}, v) &= \mathbb{E}\log p(y, \mu) - \mathbb{E}\log q(\mu) \\ &= y\bar{\mu} - se^{\bar{\mu}+v/2} - \frac{\bar{\mu}^2 + v - 2\bar{\mu}\theta}{2\sigma^2} + \frac{1}{2}\log v + \text{const.} \end{aligned} \tag{A.3}$$

Minimizing $-F(\bar{\mu}, v)$ is a convex problem, and can be solved efficiently with convex optimization methods. It can be further reduced to a univairate optimization problem, as shown

in Section 4.3.1.

A.2 VEBPM: ash prior and Gaussian mixture posterior ELBO

The ELBO is

$$\begin{aligned} F_{GM,GM} =& \mathbb{E}_q \sum_j (\log p(y_j|\mu_j) + \log p(\mu_j, \boldsymbol{z}_j) - \log q(\mu_j, \boldsymbol{z}_j)) \\ =& \sum_{j,k} \phi_{jk} \left(y_j \bar{\mu}_{jk} - s_j e^{\bar{\mu}_{jk} + V_{\mu_{jk}}/2} \right) \\ &+ \sum_{j,k} \phi_{jk} \left(\log \pi_k - \frac{1}{2} \log \sigma_k^2 - \frac{1}{2\sigma_k^2} (\bar{\mu}_{jk}^2 + V_{\mu_{jk}} - 2\bar{\mu}_{jk}\theta + \theta^2) \right) \\ &- \sum_{j,k} \phi_{jk} \log \phi_{jk} + \sum_{j,k} \phi_{jk} \frac{1}{2} \log V_{\mu_{jk}} \\ :=& \sum_{j,k} \phi_{jk} \Delta_{jk}. \end{aligned} \tag{A.4}$$

A.3 Seeger and Bouchard [2012] lower bound

The key property of the link function is its second derivative is bounded. Consider the model

$$\begin{aligned} &y_j|\mu_j \sim \text{Poisson}(\log(1 + \exp(\mu_j))), \\ &\mu_j \sim g(\cdot). \end{aligned} \tag{A.5}$$

The negative log-likelihood is $f(\mu) = -\log p(y|\mu) = \log(1 + e^{\mu}) - y \log\log(1 + e^{\mu})$. The function $f(\mu)$ is twice differentiable and $f''(\mu) \leq \kappa$ where $\kappa = 0.25 + 0.17 y_{max}$, and $y_{max} = \max(\boldsymbol{y})$.

By Taylor's Remainder Estimation Theorem, an upper bound of $f(\mu)$ is

$$f(\mu) \leq f(\tilde{\mu}) + f'(\tilde{\mu})(\mu - \tilde{\mu}) + \frac{\kappa}{2}(\mu - \tilde{\mu})^2 := h(\mu; \tilde{\mu}). \tag{A.6}$$

Instead of using $\log p(y|\mu)$, we can now replace it by $h(\mu; \tilde{\mu})$ in the ELBO. The parameter $\tilde{\mu}$ is a variational parameter and can be optimized together with q, g. The new optimization problem is

$$\min_{q,g,\tilde{\mu}} \mathbb{E}_q \, h(\mu; \tilde{\mu}) + D_{KL}(q||g). \tag{A.7}$$

The algorithm iterates between the following two steps until convergence:

1. Update $\tilde{\mu} = \mathbb{E}_q(\mu)$.

2. Update $\hat{q}, \hat{g} = \text{EBNM}(\tilde{\mu} - f'(\tilde{\mu})/\kappa, \sqrt{1/\kappa})$, where $f'(\mu) = \frac{e^\mu(-y+\log(1+e^\mu))}{(1+e^\mu)\log(1+e^\mu)}$.

The iterations are stopped when the increase of ELBO is smaller than a pre-specified tolerance.

A.4 Negative Binomial approximation with fixed r

Poisson distribution is a limiting distribution of negative binomial distribution. Consider a negative binomial distributed random variable

$$y_j \sim \text{NB}(r, p_j), \tag{A.8}$$

where $p(y; r, p) \propto p^y(1-p)^r$. When $r \to \infty$, $\text{NB}(r, p)$ converges to a Poisson distribution with parameter $rp/(1-p)$.

We assume a prior on the p_j as

$$\log \frac{p_j}{1-p_j} = \mu_j \sim g(\cdot). \tag{A.9}$$

Under this specification, the approximated Poisson mean parameter is $\lambda = re^{\mu}$.

In this section we examine two methods that formulate quadratic ELBO and can take advantage of EBNM results. The first method is based on the Pólya-Gamma (PG) augmentation proposed by Polson et al. [2013]. The method introduces a latent variable such that the log-likelihood function of μ is quadratic. The second method is based on the lower bound introduced by Jaakkola and Jordan [1997] (JJ), which is used for performing variational inference for logistic regression. For this method, we use the lower bound to solve the negative binomial mean problem. Durante and Rigon [2019] showed that the JJ lower bound is equivalent to the ELBO when using PG augmentation for variational inference in logistic regression. At the end of this section, we will show that the objective functions in two methods are also equivalent for Negative Binomial model.

Pólya-Gamma Augmentation

We introduce a variable $\omega_j \sim PG(y_j + r, 0)$, and according to Theorem 1 in Polson et al. [2013], we have

$$\begin{aligned} p(y; r, \mu) &= \binom{y+r-1}{y} \frac{(e^{\mu})^y}{(1+e^{\mu})^{y+r}} \\ &= \int \binom{y+r-1}{y} 2^{-(y+r)} e^{-\frac{\omega}{2}\mu^2 + \frac{y-r}{2}\mu} p(\omega) d\omega \\ &= \int p(y, \omega; r, \mu) d\omega. \end{aligned} \tag{A.10}$$

The joint distribution of y, ω, μ is

$$p(y, \omega, \mu) = p(y, \omega|\mu; r)g(\mu) = \binom{y+r-1}{y} 2^{-(y+r)} e^{-\frac{\omega}{2}\mu^2 + \frac{y-r}{2}\mu} p(\omega; y+r, 0)g(\mu). \quad \text{(A.11)}$$

We assume the posterior factorizes as

$$q = \prod_j q_{\omega_j}(\omega_j) q_{\mu_j}(\mu_j).$$

The ELBO is

$$\begin{aligned} F(q, g, r) = &\sum_j \mathbb{E}\left(-\frac{\omega_j}{2}\mu_j^2 + \frac{y_j - r}{2}\mu_j\right) + \sum_j \mathbb{E}\log\frac{p_\omega(\omega_j)}{q_{\omega_j}} + \sum_j \mathbb{E}\log\frac{p_\mu(\mu_j)}{q_{\mu_j}} \\ &+ \sum_j (\log\Gamma(y_j + r) - \log\Gamma(r) - (y_j + r)\log 2) \end{aligned} \quad \text{(A.12)}$$

We develop the updates in the CAVI algorithm below.

Update q_ω

According to Theorem 1 in Polson et al. [2013], the update of q_{ω_j} is

$$q_{\omega_j} = PG\left(y_j + r, \sqrt{\overline{\mu_j^2}}\right),$$

where $\overline{\mu_j^2} = \mathbb{E}_q\, \mu_j^2$.

Update q_μ, g_μ

The ELBO for $g(\mu), q_{\mu_j}$ is

$$\begin{aligned} F(q(\boldsymbol{\mu}), g) &= \sum_j \mathbb{E}\left(-\frac{1}{2}\bar{\omega}_j\mu_j^2 + \frac{y_j - r}{2}\mu_j\right) + \mathbb{E}\log\frac{g(\boldsymbol{\mu})}{q_{\boldsymbol{\mu}}}, \\ &= \sum_j \mathbb{E} - \frac{1}{2\bar{\omega}_j^{-1}}\left(\mu_j - \frac{y_j - r}{2\bar{\omega}_j}\right)^2 + \mathbb{E}\log\frac{g(\boldsymbol{\mu})}{q_{\boldsymbol{\mu}}} \end{aligned} \tag{A.13}$$

where $\bar{\omega}_j = \mathbb{E}_q\,\omega_j$. The objective function is the same as the one for the following EBNM problem

$$(q, \hat{g}) = \text{EBNM}\left(\frac{y_j - r}{2\bar{\omega}_j}, \sqrt{\bar{\omega}_j^{-1}}\right).$$

Calculation of the ELBO

We show that the KL divergence between q_{ω_j} and $p(\omega)$ in ELBO (A.12) can be calculated explicitly. Note that $p(\omega) = PG(\omega; b, 0), q(\omega) = PG(\omega; b, c)$ where $b = y + r, c = \sqrt{\overline{\mu^2}}$, we have

$$\mathbb{E}\log\frac{p(\omega)}{q(\omega)} = \mathbb{E}\log\frac{\cosh^{-b}(c/2)}{e^{-\frac{c^2}{2}\omega}} = \log\cosh^{-b}(c/2) + \frac{c^2}{2}\,\mathbb{E}\,\omega. \tag{A.14}$$

The KL divergence between q_μ and $g(\mu)$ can be obtained from EBNM objective as the objective of EBNM is the log marginal likelihood and has explicit form.

Optional: Update r

When solving Poisson mean problem, we may fix r to be a large number. In the context of negative binomial mean problem, we study if r can be also estimated. The objective function

$F(r)$ is

$$\begin{aligned} F(r) &= \sum_j \left(-\frac{r}{2}\bar{\mu}_j - r \log \cosh(\sqrt{\overline{\mu_j^2}}/2) + \log \binom{y_j + r - 1}{y_j} \right) - Nr \log 2 \\ &= r(\Delta - N \log 2) - N \log \Gamma(r) + \sum_j \log \Gamma(y_j + r), \end{aligned} \tag{A.15}$$

where

$$\Delta = \sum_j \left(-\frac{1}{2}\bar{\mu}_j - \log \cosh(\sqrt{\overline{\mu_j^2}}/2) \right).$$

Its derivative is

$$F'(r) = \Delta - N \log 2 - N\psi(r) + \sum_j \psi(y_j + r), \tag{A.16}$$

where $\psi(\cdot)$ is the digamma function. Thus we can use a standard optimization method to find optimal r within the CAVI algorithm.

JJ Lower Bound

Jaakkola and Jordan [1997] introduced a lower bound on the log logistic function, by first writing it as

$$-\log(1 + e^{-x}) = \frac{x}{2} - \log(\exp(x/2) + \exp(-x/2)), \tag{A.17}$$

then lower bound the latter part by a first order Taylor expansion in the variable x^2, as

$$-\log(\exp(x/2) + \exp(-x/2)) \geq -\frac{\xi}{2} - \log(1 + e^{-\xi}) - \frac{1}{2\xi} \tanh(\xi/2)(x^2 - \xi^2). \tag{A.18}$$

This lower bound is exact whenever $\xi^2 = x^2$. The log-likelihood of μ in model (A.8) can

then be lower bounded by

$$\begin{aligned} l(\boldsymbol{\mu}) =& \sum_j \log p(y_j|\mu_j) \\ =& \sum_j y_j\mu_j - \frac{y_j+r}{2}\mu_j - (y_j+r)\log(e^{\mu_j/2}+e^{-\mu_j/2}) + \log\Gamma(y_j+r) - N\log\Gamma(r) \\ \geq& \sum_j y_j\mu_j - \frac{y_j+r}{2}\mu_j + (y_j+r)(-\frac{\xi_j}{2} - \log(1+e^{-\xi_j}) - \frac{1}{4\xi_j}\tanh(\xi_j/2)(\mu_j^2-\xi_j^2)) \\ & + \log\Gamma(y_j+r) - N\log\Gamma(r) \\ :=& \tilde{l}(\boldsymbol{\mu};\boldsymbol{\xi}). \end{aligned} \tag{A.19}$$

Replacing $l(\boldsymbol{\mu})$ by $\tilde{l}(\boldsymbol{\mu};\boldsymbol{\xi})$ in the ELBO, the new objective function to be maximized is

$$F(q,g,r,\boldsymbol{\xi}) = \mathbb{E}_q \sum_j \tilde{l}(\mu_j;\xi_j) + \sum_j \mathbb{E}_q \log\frac{g(\mu_j)}{q_{\mu_j}}. \tag{A.20}$$

The variational algorithm iterates over the following two steps until convergence

1. Update $\xi_j^2 = \mathbb{E}_q(\mu_j^2)$.

2. Update $(\hat{q},\hat{g}) = \text{EBNM}((y_j-r)s_j^2/2, s_j)$, where $s_j^2 = 2\xi_j/((y_j+r)\tanh(\xi_j/2))$.

Optional: Update r

The objective function for estimating r is

$$F(r) = r\Delta + \sum_j \log\Gamma(y_r+r) - N\log\Gamma(r), \tag{A.21}$$

where

$$\Delta = \sum_j -\frac{\xi_j}{2} - \log(1+e^{-\xi_j}) - \frac{1}{4\xi_j}\tanh(\xi_j/2)(\overline{\mu_j^2} - \xi_j^2) - \frac{\bar{\mu}_j}{2}.$$

The next theorem shows the equivalence of using JJ lower bound and PG augmentation in variational inference.

Theorem A.4.1. *The objective functions using PG augmentation and JJ lower bound are the same when* $\xi_j = \sqrt{\overline{\mu_j^2}}$.

Proof. Let $\xi_j = \sqrt{\overline{\mu_j^2}}$, then

$$\begin{aligned}
F^{PG}(q,g,r) =& \sum_j \mathbb{E}\left(-\frac{\omega_j}{2}\mu_j^2 + \frac{y_j - r}{2}\mu_j\right) + \sum_j \mathbb{E}\log\frac{p_\omega(\omega_j)}{q_{\omega_j}} + \sum_j \mathbb{E}\log\frac{p_\mu(\mu_j)}{q_{\mu_j}} \\
&+ \sum_j (\log\Gamma(y_j + r) - \log\Gamma(r) - (y_j + r)\log 2) \\
\overset{\mathbb{E}_q\omega}{=}& \sum_j \mathbb{E}\left(-\frac{y_j + r}{4\xi_j}\tanh(\xi_j/2)\mu_j^2 + \frac{y_j - r}{2}\mu_j\right) \\
&+ \sum_j \left(-(y_j + r)\log\cosh(\xi_j/2) + \frac{\xi_j(y_j + r)}{4}\tanh(\xi_j/2)\right) \\
&+ \sum_j \mathbb{E}\log\frac{p_\mu(\mu_j)}{q_{\mu_j}} + \sum_j (\log\Gamma(y_j + r) - \log\Gamma(r) - (y_j + r)\log 2) \\
\overset{\cosh(x)=\frac{1+e^{-2x}}{2e^{-x}}}{=}& F^{JJ}(q,g,r,\boldsymbol{\xi}).
\end{aligned} \tag{A.22}$$

□

A.5 Derivatives in gradient-based VEBPM

A.5.1 Compound method

$$\frac{\partial h(z,s^2,g)}{\partial z} = -l'S' - l'_{\mathrm{NM}} - \frac{(S-z)(S'-1)}{s^2}$$

$$\frac{\partial h(z, s^2, g)}{\partial s^2} = -l' S' - l'_{\mathrm{NM}} - \frac{2s^2(S-z)S' - (S-z)^2}{2s^4} - \frac{1}{2s^2}$$

$$\frac{\partial h(z, s^2, g)}{\partial g} = -l' S' - l'_{\mathrm{NM}} - \frac{(S-z)S'}{s^2}$$

A.5.2 Inversion method

$$\begin{aligned}\frac{\partial h(\theta, g)}{\partial \theta} =& - l'(\theta) - \frac{\partial l_{\mathrm{NM}}(z_g(\theta); g, s^2(\theta))}{\partial \theta} \\ & - \frac{2s^2(\theta)(\theta - z_g(\theta)(1-z') - (s^2(\theta))'(\theta - z_g(\theta))^2}{2(s^2(\theta))^2} - \frac{(s^2(\theta))'}{2s^2(\theta)},\end{aligned}$$

where

$$\frac{\partial l_{\mathrm{NM}}(z_g(\theta); g, s^2(\theta))}{\partial \theta} = \frac{\partial l_{\mathrm{NM}}}{\partial z}\frac{\partial z}{\partial \theta} + \frac{\partial l_{\mathrm{NM}}}{s^2}\frac{\partial s^2(\theta)}{\theta}.$$

Taking derivative of both side w.r.t. θ of

$$\theta = S_g(z_g(\theta), s^2(\theta)) = z_g(\theta) + s^2(\theta) l'_{\mathrm{NM}}(z_g(\theta); g, s^2(\theta)),$$

gives

$$\begin{aligned}1 &= z' + (s^2(\theta))' l'_{\mathrm{NM}} + s^2(\theta)\left(l''_{\mathrm{NM}} z' + \frac{\partial l'_{\mathrm{NM}}}{\partial s^2}(s^2(\theta))'\right) \\ \Longrightarrow z' &= \frac{1 - (s^2(\theta))' l'_{\mathrm{NM}} - s^2(\theta)(s^2(\theta))'\frac{\partial l'_{\mathrm{NM}}}{\partial s^2}}{1 + s^2(\theta) l''_{\mathrm{NM}}}.\end{aligned}$$

$$\frac{\partial h(\theta, g)}{\partial g} = - \frac{\partial l_{\mathrm{NM}}(z_g(\theta); g, s^2(\theta))}{\partial g} - \frac{2(z_g(\theta) - \theta)\frac{\partial z_g(\theta)}{\partial g}}{2s^2(\theta)},$$

where

$$\frac{\partial l_{\mathrm{NM}}(z_g(\theta); g, s^2(\theta))}{\partial g} = \frac{\partial l_{\mathrm{NM}}}{\partial z}\frac{\partial z}{\partial g} + \frac{\partial l_{\mathrm{NM}}}{g}.$$

Taking derivative of both side w.r.t. g of

$$\theta = z_g(\theta) + s^2(\theta) l'_{\mathrm{NM}}(z_g(\theta); g, s^2(\theta)),$$

gives

$$\frac{\partial z}{\partial g} = -\frac{S^2(\theta)\frac{\partial l'_{\mathrm{NM}}}{\partial g}}{1 + s^2(\theta) l''_{\mathrm{NM}}}.$$

A.5.3 Derivatives when using ash prior

Let $f(z; \boldsymbol{w}, s^2) = \sum_k w_k N(z; \mu, \sigma_k^2 + s^2)$, and $l_{\mathrm{NM}}(z; \boldsymbol{w}, s^2) = \log f(z; \boldsymbol{w}, s^2)$.

Partial derivatives w.r.t z:

$$\frac{\partial f(z; \boldsymbol{w}, s^2)}{\partial z} = -\sum_k \frac{w_k}{\sqrt{2\pi(\sigma_k^2 + s^2)}} e^{-\frac{(z-\mu)^2}{2(\sigma_k^2+s^2)}} \frac{(z-\mu)}{\sigma_k^2 + s^2}$$

$$\frac{\partial^2 f(z; \boldsymbol{w}, s^2)}{\partial z^2} = \sum_k \frac{w_k}{\sqrt{2\pi(\sigma_k^2 + s^2)}} e^{-\frac{(z-\mu)^2}{2(\sigma_k^2+s^2)}} \left((\frac{(z-\mu)}{\sigma_k^2 + s^2})^2 - \frac{1}{\sigma_k^2 + s^2} \right)$$

$$\frac{\partial^3 f(z; \boldsymbol{w}, s^2)}{\partial z^3} = \sum_k \frac{w_k}{\sqrt{2\pi(\sigma_k^2 + s^2)}} e^{-\frac{(z-\mu)^2}{2(\sigma_k^2+s^2)}} \left(\frac{3(z-\mu)}{(\sigma_k^2 + s^2)^2} - (\frac{(z-\mu)}{\sigma_k^2 + s^2})^3 \right)$$

$$\frac{\partial l_{\mathrm{NM}}(z; \boldsymbol{w}, s^2)}{\partial z} = \frac{1}{f(z; \boldsymbol{w}, s^2)} \frac{\partial f(z; \boldsymbol{w}, s^2)}{\partial z}$$

$$\frac{\partial^2 l_{\mathrm{NM}}(z;\boldsymbol{w},s^2)}{\partial z^2} = \frac{1}{f(z;\boldsymbol{w},s^2)}\frac{\partial^2 f(z;\boldsymbol{w},s^2)}{\partial z^2} - \left(\frac{1}{f(z;\boldsymbol{w},s^2)}\frac{\partial f(z;\boldsymbol{w},s^2)}{\partial z}\right)^2$$

$$\frac{\partial^3 l_{\mathrm{NM}}(z;\boldsymbol{w},s^2)}{\partial z^3} = \frac{f'''}{f} - \frac{3f'f''}{f^2} + \frac{2(f')^3}{f^3}$$

Partial derivatives w.r.t s^2:

$$\frac{\partial f(z;\boldsymbol{w},s^2)}{\partial s^2} = \sum_k \frac{w_k}{2\sqrt{2\pi}} e^{-\frac{(z-\mu)^2}{2(\sigma_k^2+s^2)}} (\sigma_k^2+s^2)^{-5/2}((z-\mu)^2 - (\sigma_k^2+s^2))$$

$$\frac{\partial^2 f(z;\boldsymbol{w},s^2)}{\partial z \partial s^2} = \sum_k \frac{w_k(z-\mu)(3\sigma_k^2+3s^2-(z-\mu)^2)}{2\sqrt{2\pi}(\sigma_k^2+s^2)^{7/2}} e^{-\frac{(z-\mu)^2}{2(\sigma_k^2+s^2)}}$$

$$\frac{\partial l_{\mathrm{NM}}(z;\boldsymbol{w},s^2)}{\partial s^2} = \frac{1}{f(z;\boldsymbol{w},s^2)}\frac{\partial f(z;\boldsymbol{w},s^2)}{\partial s^2}$$

$$\frac{\partial^2 l_{\mathrm{NM}}(z;\boldsymbol{w},s^2)}{\partial z \partial s^2} = -\frac{1}{f^2}\frac{\partial f}{\partial s^2}\frac{\partial f}{\partial z} + \frac{1}{f}\frac{\partial^2 f}{\partial z \partial s^2}$$

Partial derivatives w.r.t a_k:

Recall $w_k = \exp(a^k)/\sum_l \exp(a_l)$.

$$\frac{\partial f}{\partial a_k} = \frac{e^{a_k}(N(z;0,\sigma_k^2+s^2)\sum_l e^{a_l} - \sum_l e^{a_l} N(z;\mu,\sigma_l^2+s^2))}{(\sum_l e^{a_l})^2}$$

$$\frac{\partial^2 f}{\partial z \partial a_k} = \frac{z e^{a_k} \left(\sum_l e^{a_l} \frac{N(z;\mu,\sigma_l^2+s^2)}{\sigma_l^2+s^2} - \frac{N(z;\mu,\sigma_k^2+s^2)}{\sigma_k^2+s^2} \sum_l e^{a_l} \right)}{(\sum_l e^{a_l})^2}$$

$$\frac{\partial l_{\text{NM}}}{\partial a_k} = \frac{1}{f} \frac{\partial f}{\partial a_k}$$

$$\frac{\partial^2 l_{\text{NM}}}{\partial z \partial a_k} = -\frac{1}{f^2} \frac{\partial f}{\partial a_k} \frac{\partial f}{\partial z} + \frac{1}{f} \frac{\partial^2 f}{\partial z \partial a_k}$$

Derivatives w.r.t μ

$$\frac{\partial f(z; \boldsymbol{w}, s^2)}{\partial \mu} = \sum_k \frac{w_k}{\sqrt{2\pi(\sigma_k^2 + s^2)}} e^{-\frac{(z-\mu)^2}{2(\sigma_k^2+s^2)}} \frac{(z-\mu)}{\sigma_k^2 + s^2}$$

$$\frac{\partial^2 f(z; \boldsymbol{w}, s^2)}{\partial z \partial \mu} = \sum_k \frac{w_k}{\sqrt{2\pi(\sigma_k^2 + s^2)}} e^{-\frac{(z-\mu)^2}{2(\sigma_k^2+s^2)}} \left((\frac{(z-\mu)}{\sigma_k^2 + s^2})^2 - \frac{1}{\sigma_k^2 + s^2} \right)$$

Derivatives of objective function:

$$\frac{\partial h}{\partial z} = e^{z+s^2 l'_{\text{NM}}}(1 + s^2 l''_{\text{NM}}) - (y - 0.5)(1 + s^2 l''_{\text{NM}}) - l'_{\text{NM}} - s^2 l'_{\text{NM}} l''_{\text{NM}}$$

$$\frac{\partial h}{\partial s^2} = (e^{z+s^2 l'_{\text{NM}}} - y + 0.5)(l'_{\text{NM}} + s^2 \frac{\partial l'_{\text{NM}}}{\partial s^2}) - \frac{\partial l_{\text{NM}}}{\partial s^2} - \frac{1}{2}(l'_{\text{NM}}(z; g, s^2))^2 - s^2 l'_{\text{NM}} \frac{\partial l'_{\text{NM}}}{\partial s^2}$$

$$\frac{\partial h}{\partial a_k} = (e^{z+s^2 l'_{\mathrm{NM}}} - y + 0.5)(s^2 \frac{\partial l'_{\mathrm{NM}}}{\partial a_k}) - \frac{\partial l_{\mathrm{NM}}}{\partial a_k} - s^2 l'_{\mathrm{NM}} \frac{\partial l'_{\mathrm{NM}}}{\partial a_k}$$

Derivatives of constraint function:

$$\frac{\partial c}{\partial z} = 1 + e^v l''_{\mathrm{NM}}$$

$$\frac{\partial c}{\partial v} = 1 + e^v (l'_{\mathrm{NM}} + \frac{\partial l'_{\mathrm{NM}}}{\partial v})$$

$$\frac{\partial c}{\partial a_k} = e^v \frac{\partial l'_{\mathrm{NM}}}{\partial a_k}$$

A.6 Additional simulation results

We show the run time and MSE of the log mean parameters here.

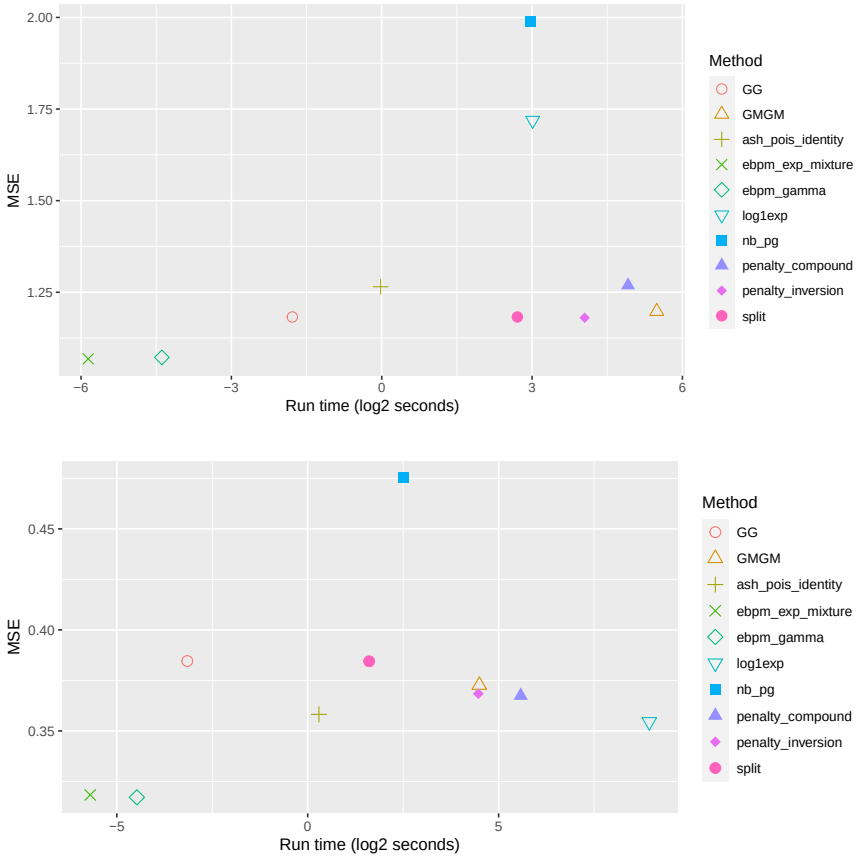


Figure A.1: Run time (log2) and MSE (log mean parameter, relative to MLE) in simulation study of VEBPM. Two plots correspond to simulation a, and b.

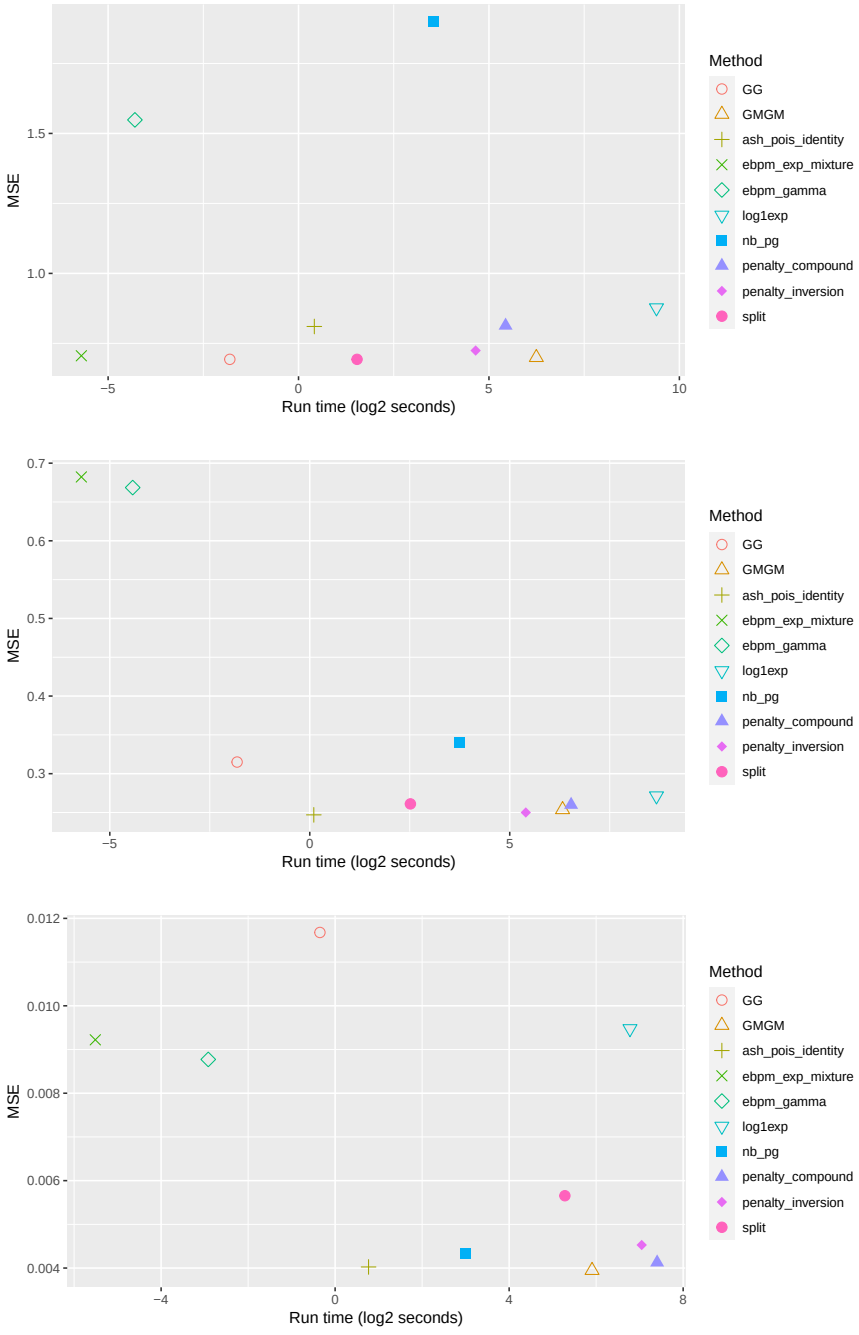


Figure A.2: Run time (log2) and MSE (log mean parameter, relative to MLE) in simulation study of VEBPM. Three plots correspond to simulation d, e, and f.

APPENDIX B

SMOOTHING

B.1 Wavelet prior

A random vector $\boldsymbol{\mu}$ of length 2^T, $T \in \mathbb{N}$ is said to have a discrete wavelet prior if

$$p(W\boldsymbol{\mu}) \propto \prod_{t=0}^{T-1} \prod_{h_t=1}^{2^t} g_t(\cdot), \tag{B.1}$$

where W is the DWT matrix, t indexes the levels of wavelet coefficients, h_t indexes the position of a wavelet coefficient at level t, and $g_t(\cdot)$ is a prior distribution of the wavelet coefficients at level t. Note that one of the element of $W\boldsymbol{\mu}$ is a scaled summation of $\boldsymbol{\mu}$ and we assume it has a flat prior. Because of the orthogonality of W, priors on wavelet coefficients imply a unique prior distribution on the random vector $\boldsymbol{\mu}$. The family of discrete wavelet prior is denoted as $\mathcal{G}_{\text{wavelet}}$.

Suppose y_i is normal distributed with mean μ_i and standard deviation s, for $i = 1, 2, ..., 2^L$, and $\boldsymbol{\mu}$ follows a discrete wavelet prior

$$y_i|\mu_i \sim N(\mu_i, s^2), \tag{B.2}$$

$$\boldsymbol{\mu} \sim g(\cdot), g \in \mathcal{G}_{\text{wavelet}}. \tag{B.3}$$

An empirical Bayes wavelet denoising (EBWD) procedure proceeds with the following two steps:

1. Estimate g by maximizing the marginal likelihood $\int p(\boldsymbol{y}|\boldsymbol{\mu})g(d\boldsymbol{\mu})$.

2. Compute the posterior distribution $p(\boldsymbol{\mu}|\boldsymbol{y}, \hat{g})$.

B.2 Additional simulation results

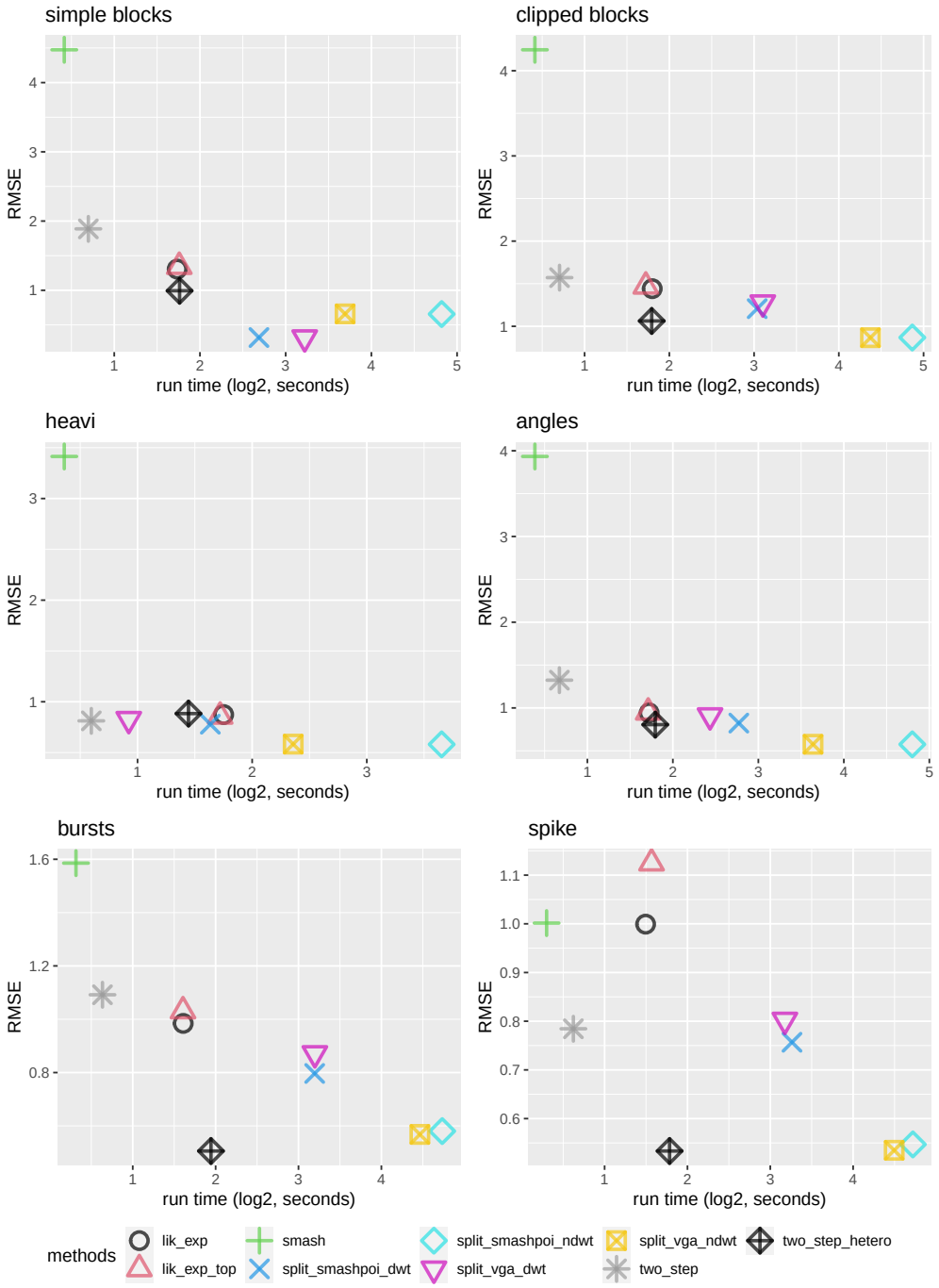


Figure B.1: Plot of run time and RMSE in simulation study of smoothing count data. SNR = 1, max-mean-count size = 10.

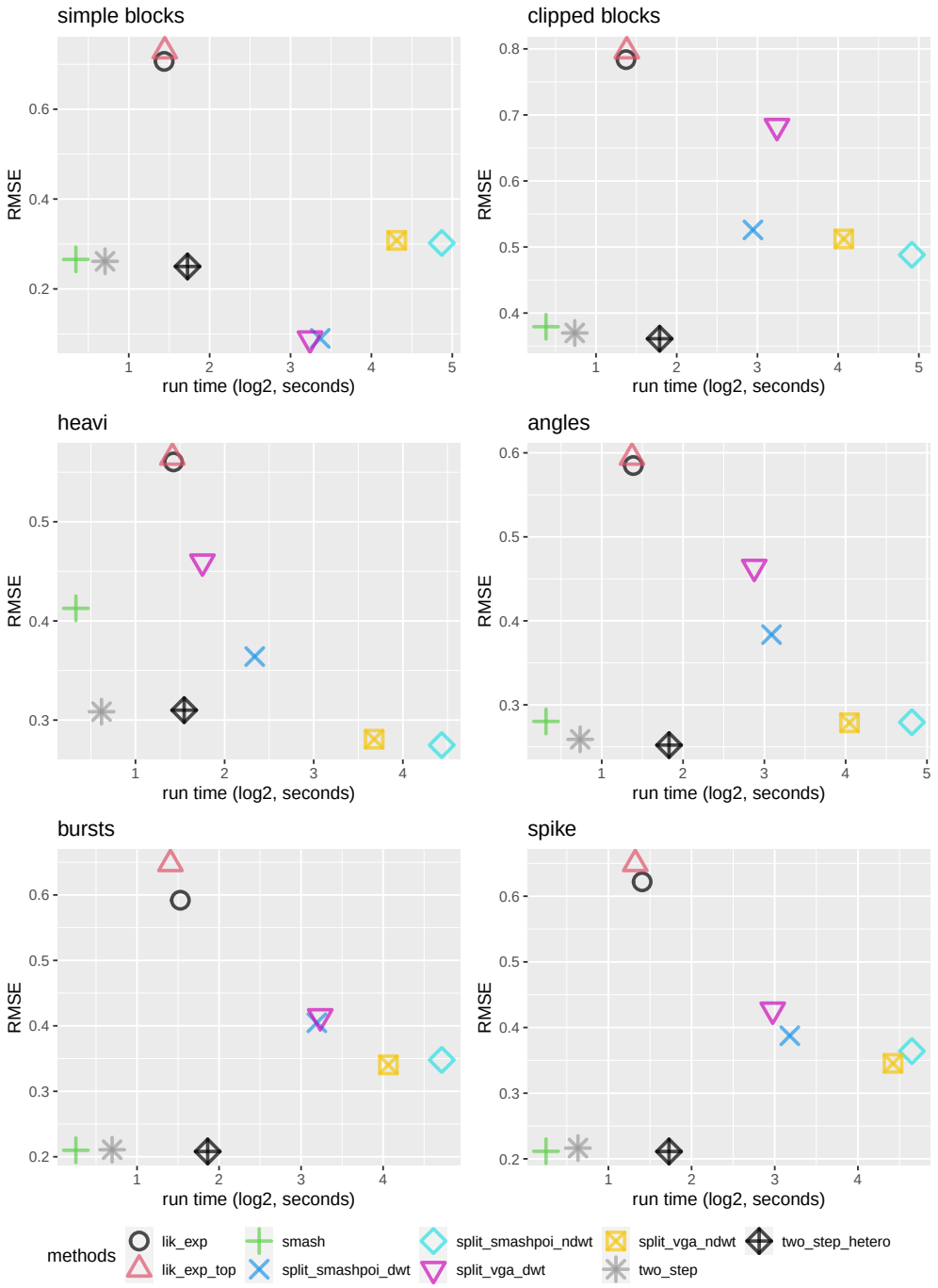


Figure B.2: Plot of run time and RMSE in simulation study of smoothing count data. SNR = 3, max-mean-count size = 5.

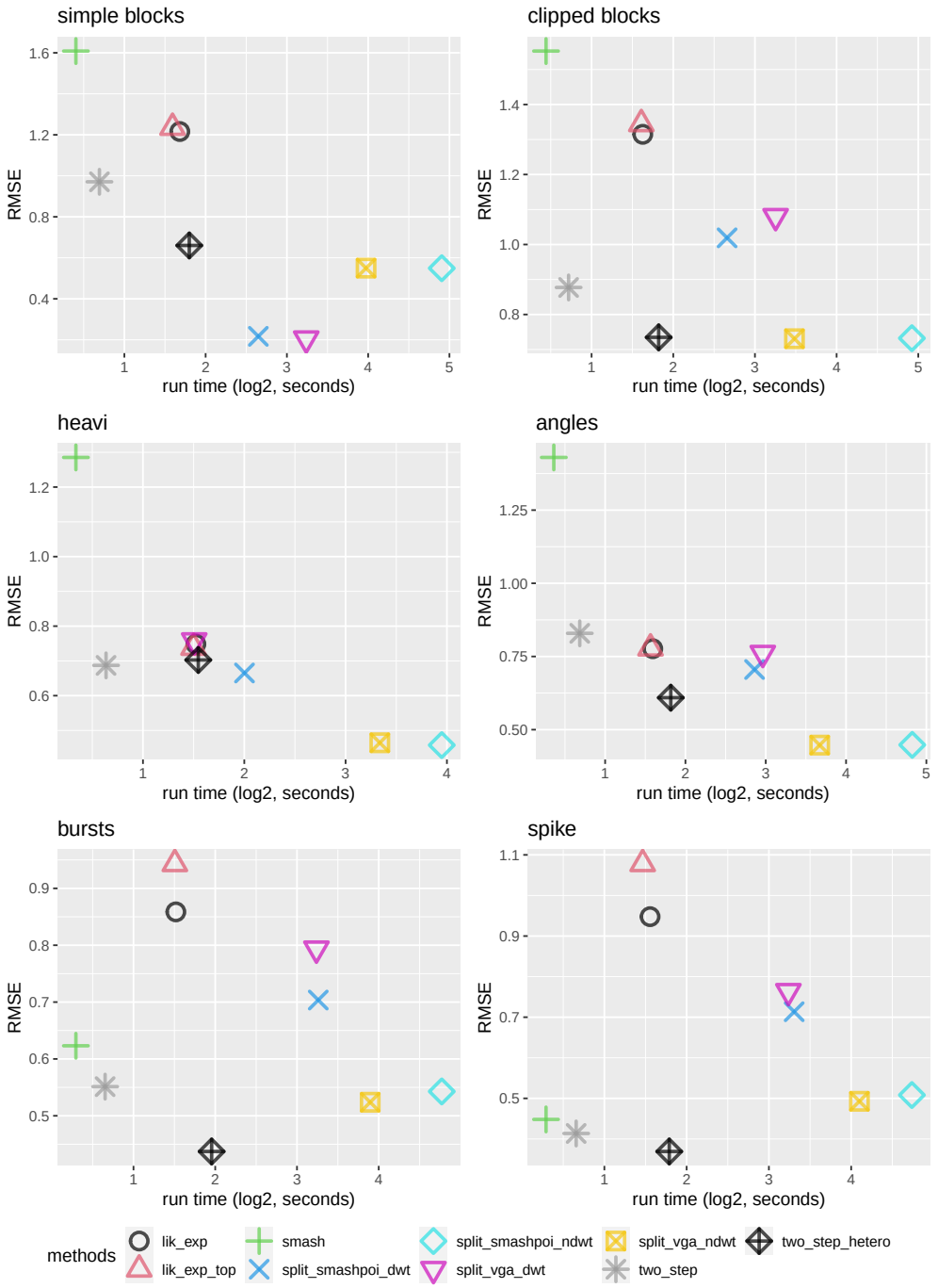


Figure B.3: Plot of run time and RMSE in simulation study of smoothing count data. SNR = 3, max-mean-count size = 10.

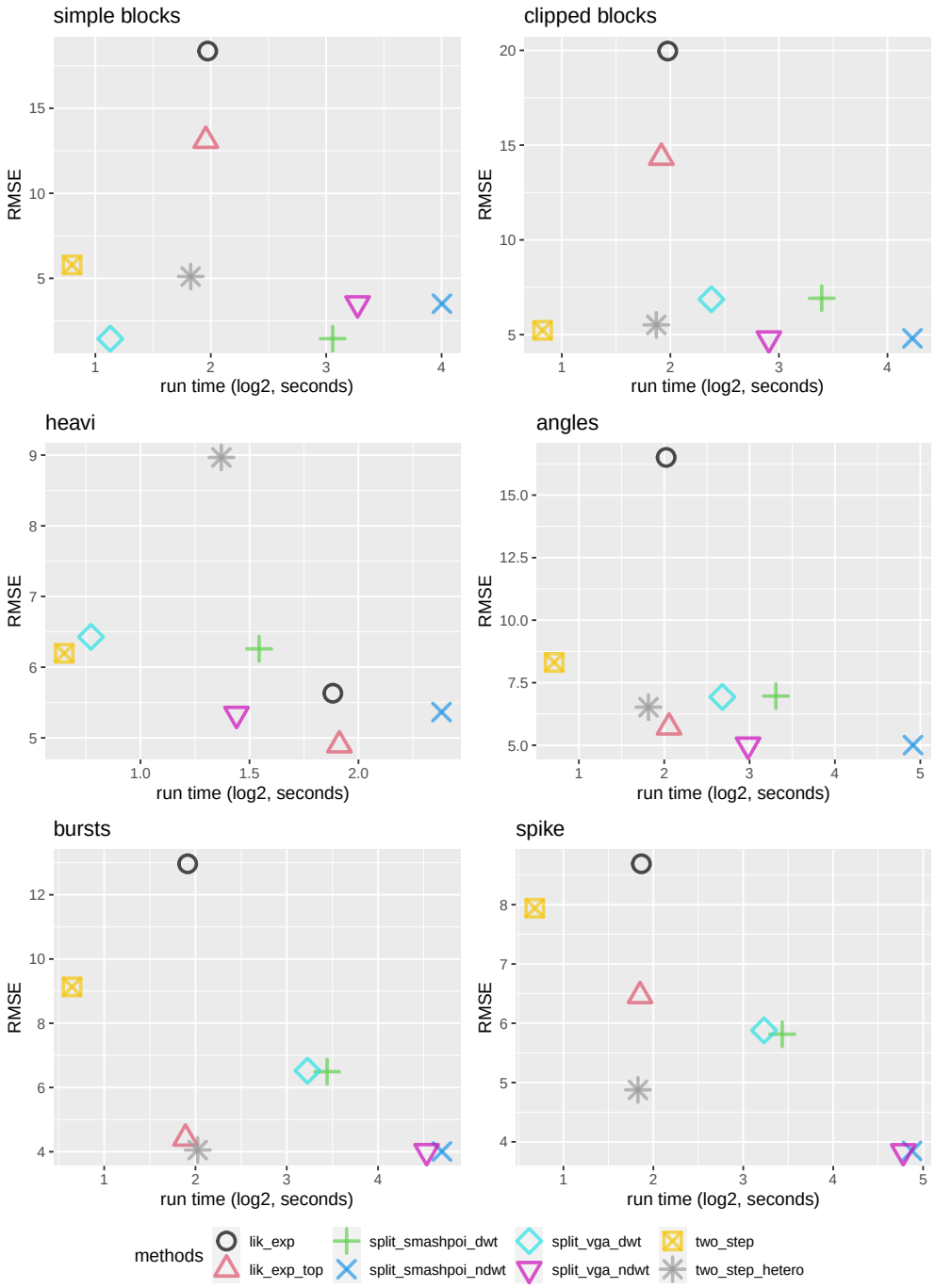
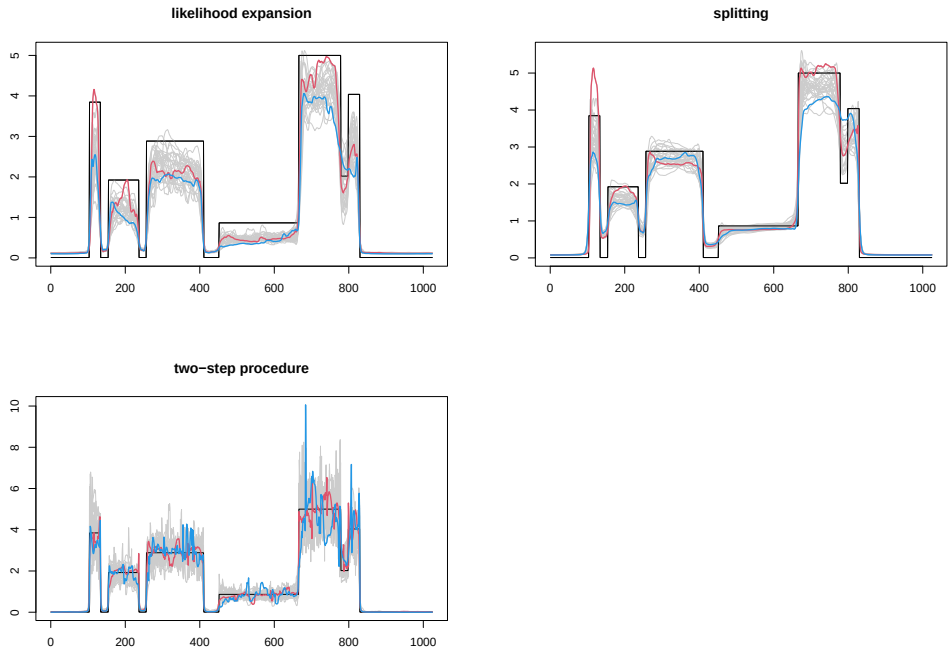
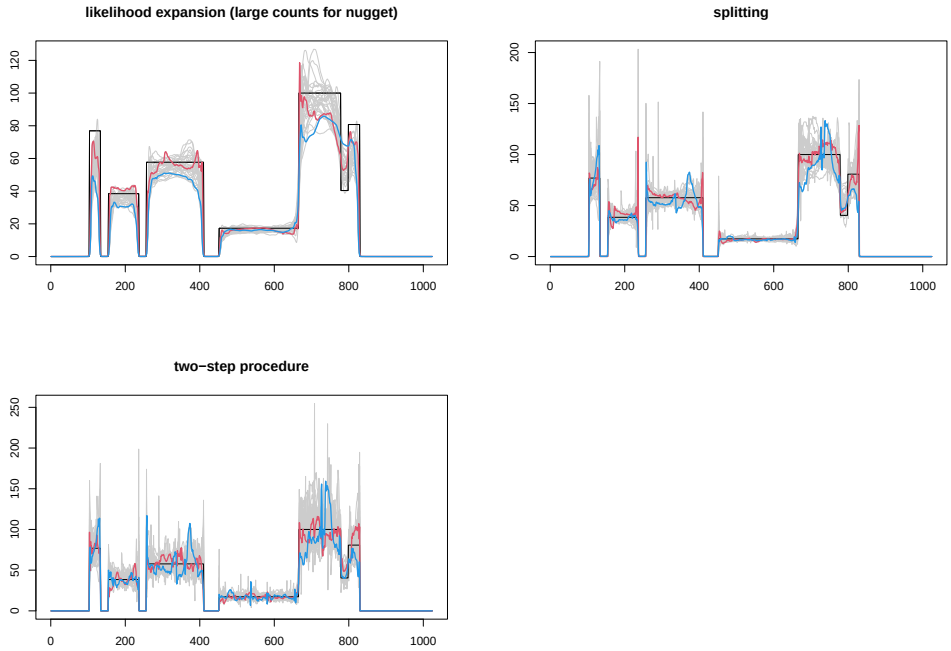


Figure B.4: Plot of run time and RMSE in simulation study of smoothing count data. SNR = 3, max-mean-count size = 100.

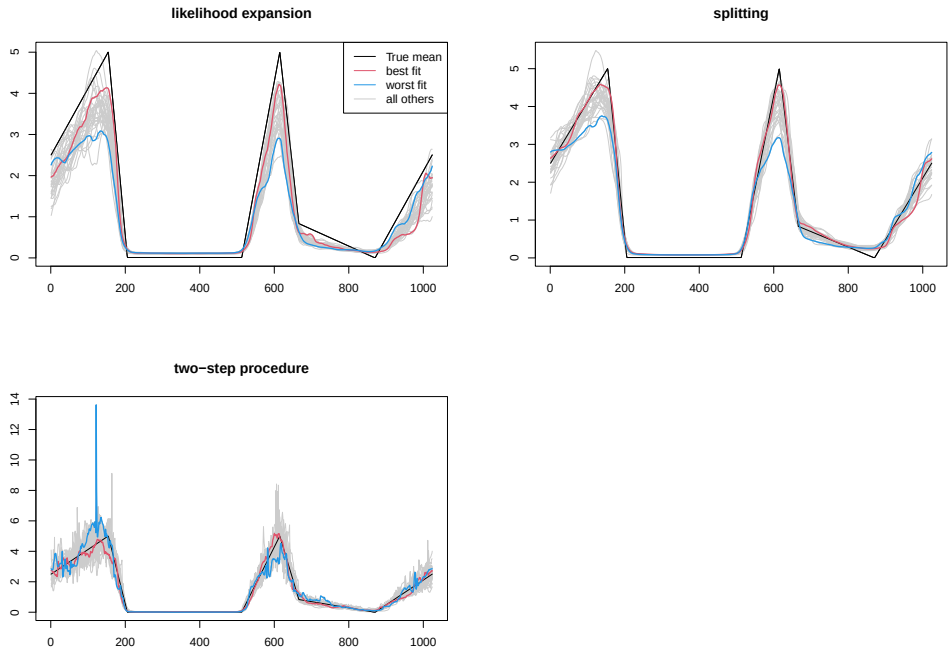


(a) max-mean-count size = 5

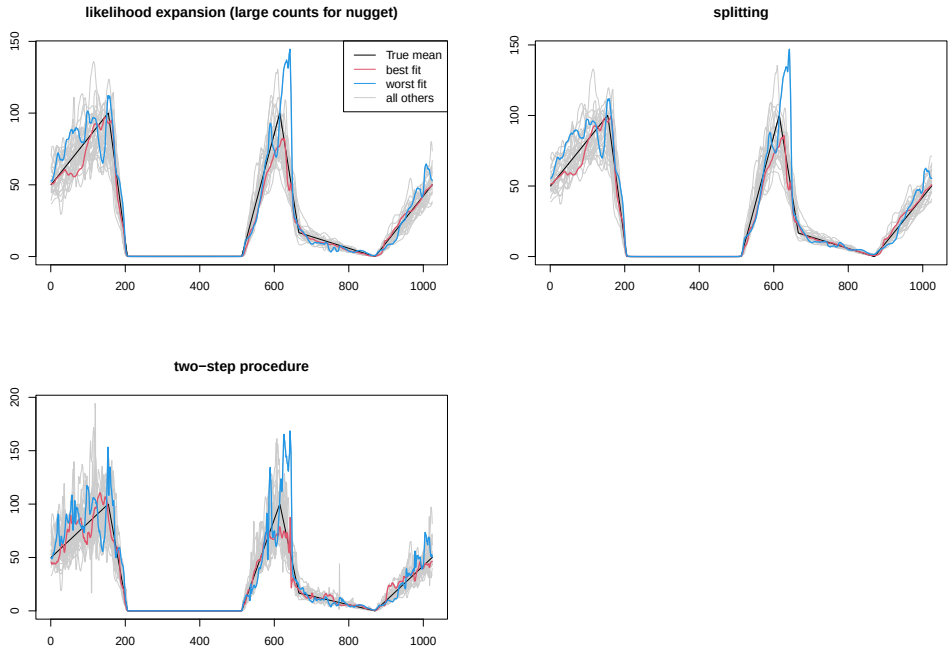


(b) max-mean-count size = 100

Figure B.5: Visualization of fitted curves in simulation study of smoothing count data. SNR = 1, clipped block function.

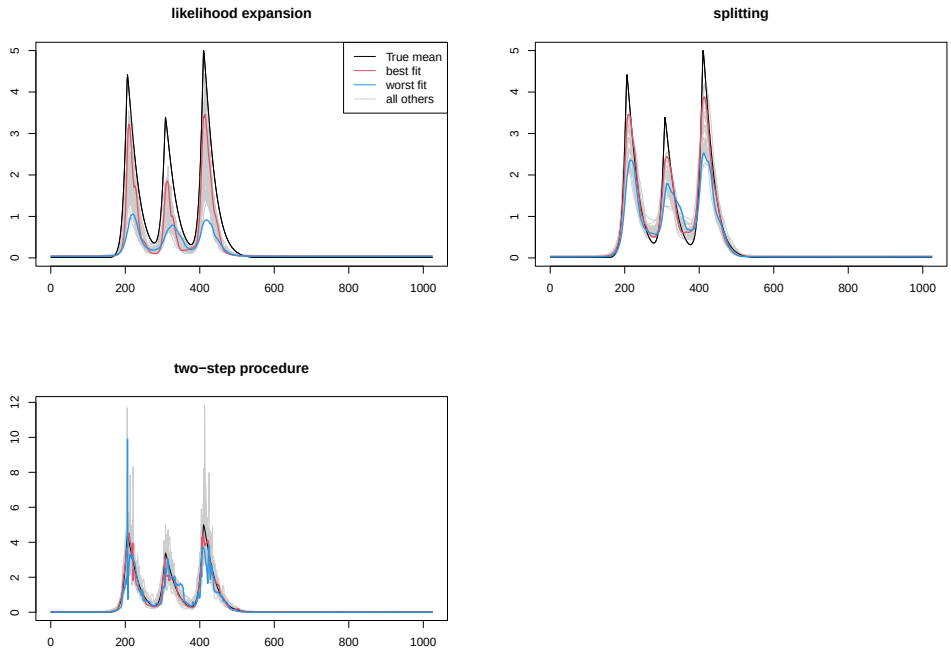


(a) max-mean-count size = 5

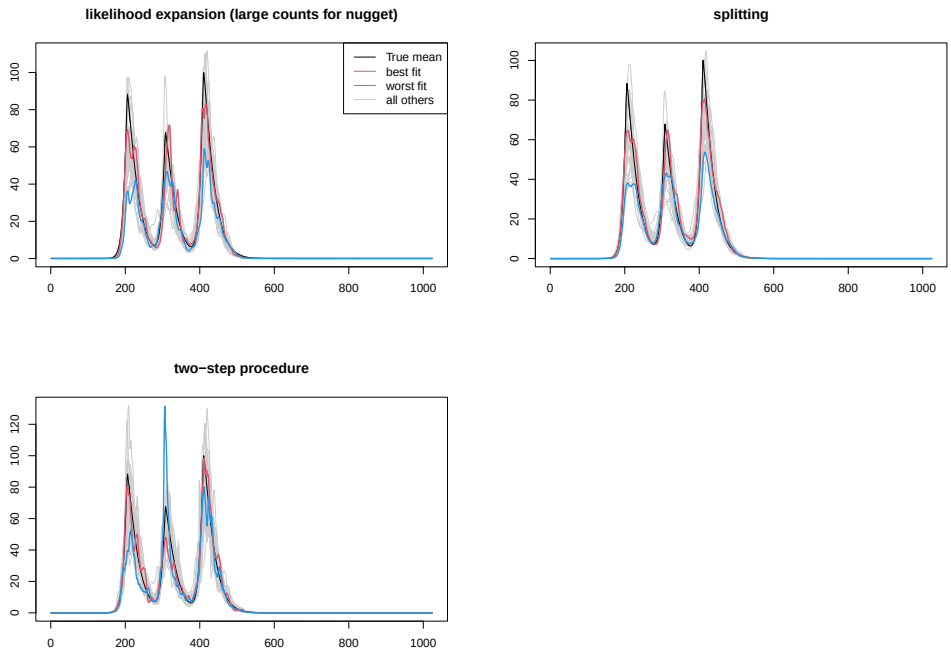


(b) max-mean-count size = 100

Figure B.6: Visualization of fitted curves in simulation study of smoothing count data. SNR = 1, angles function.

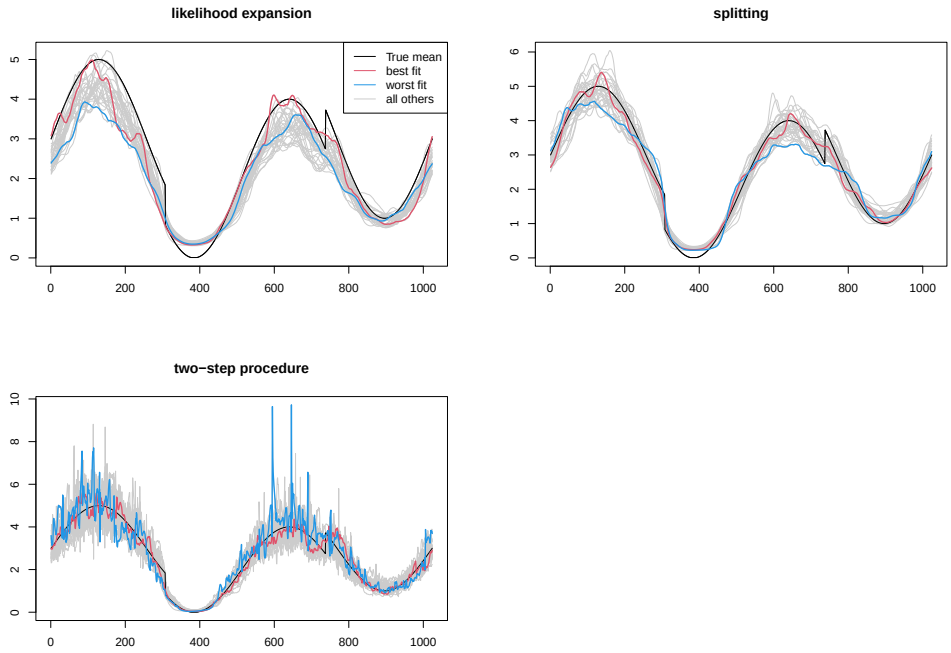


(a) max-mean-count size = 5

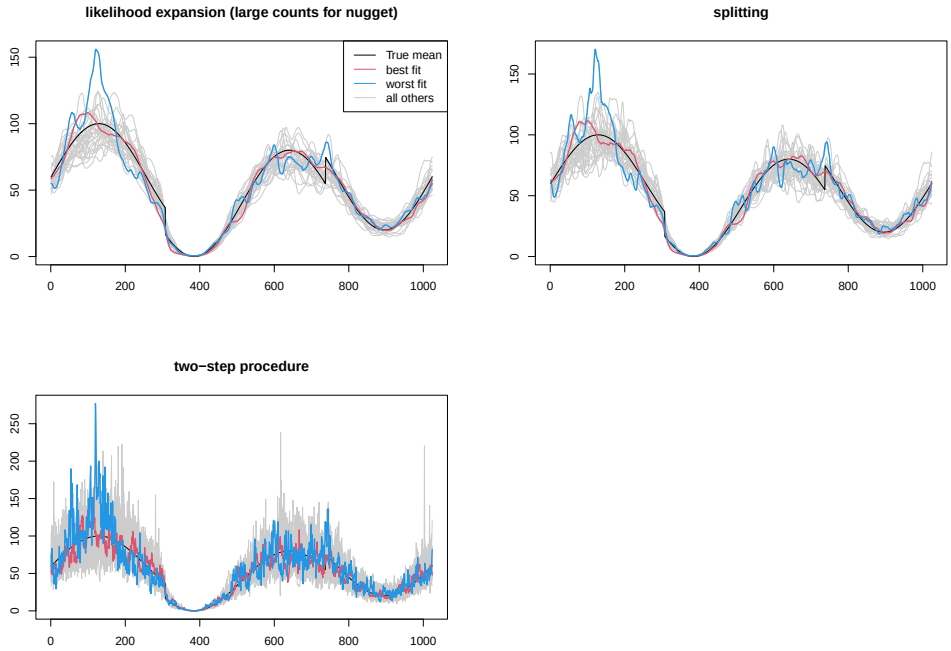


(b) max-mean-count size = 100

Figure B.7: Visualization of fitted curves in simulation study of smoothing count data. SNR = 1, bursts function.



(a) max-mean-count size = 5



(b) max-mean-count size = 100

Figure B.8: Visualization of fitted curves in simulation study of smoothing count data. SNR = 1, heavi function.

B.3 Additional results for RNA-seq smoothing

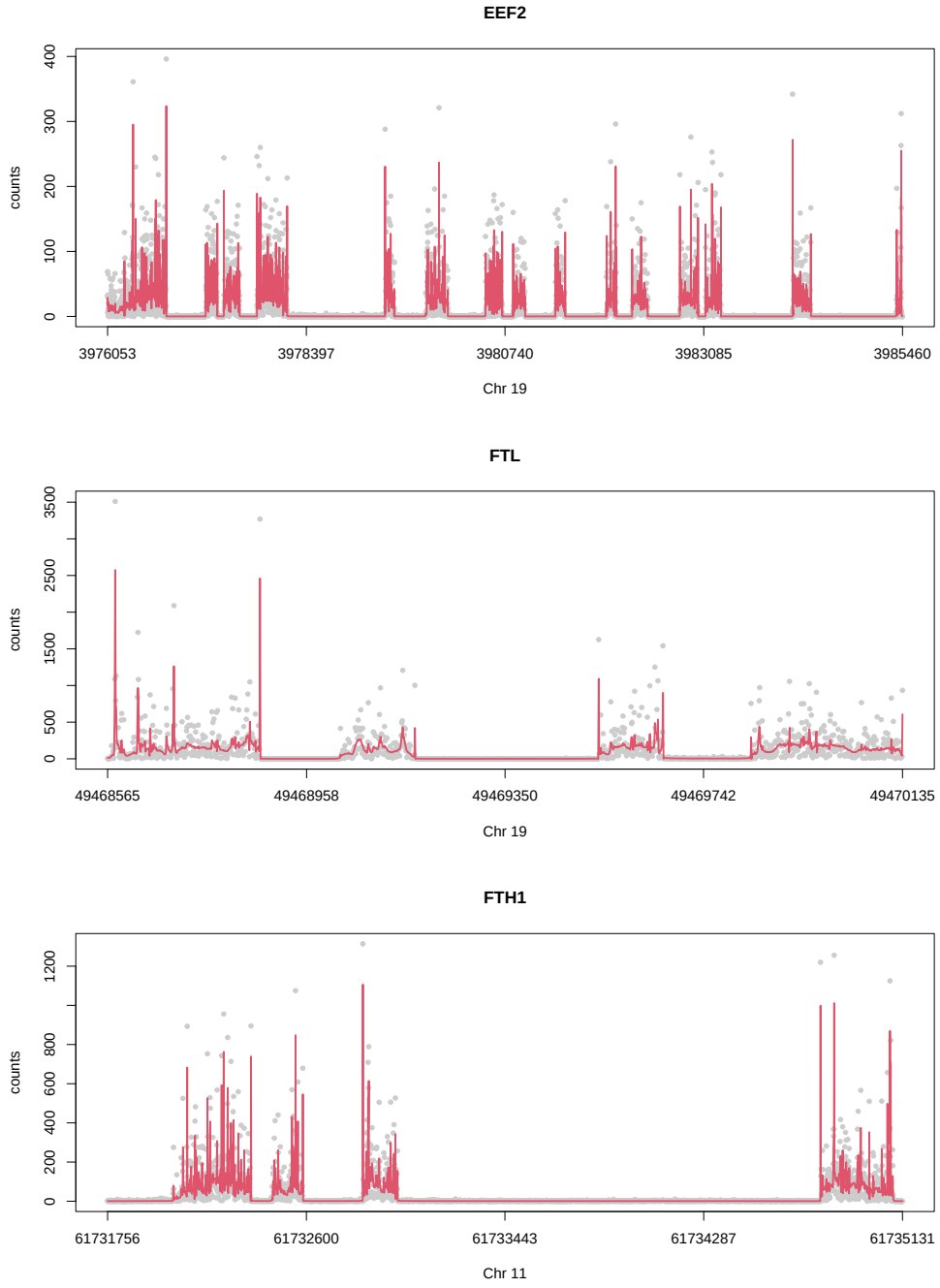
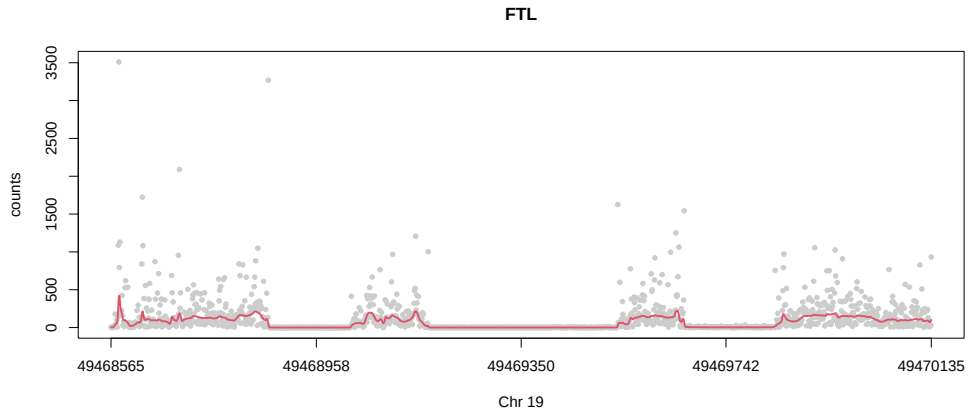
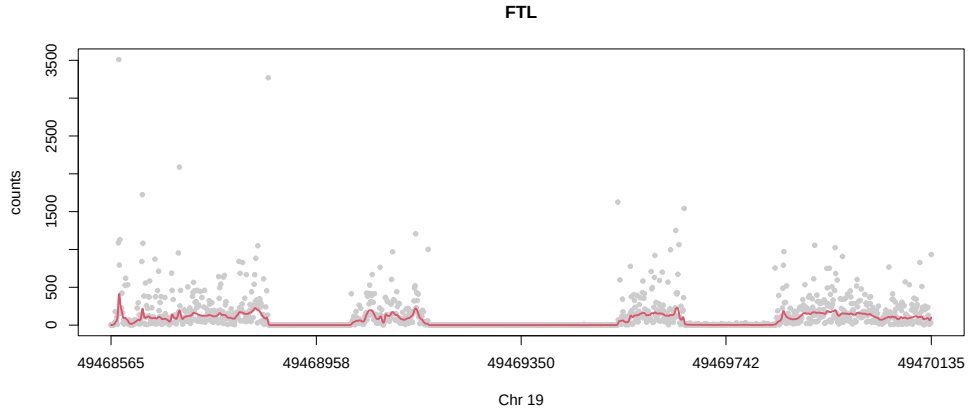


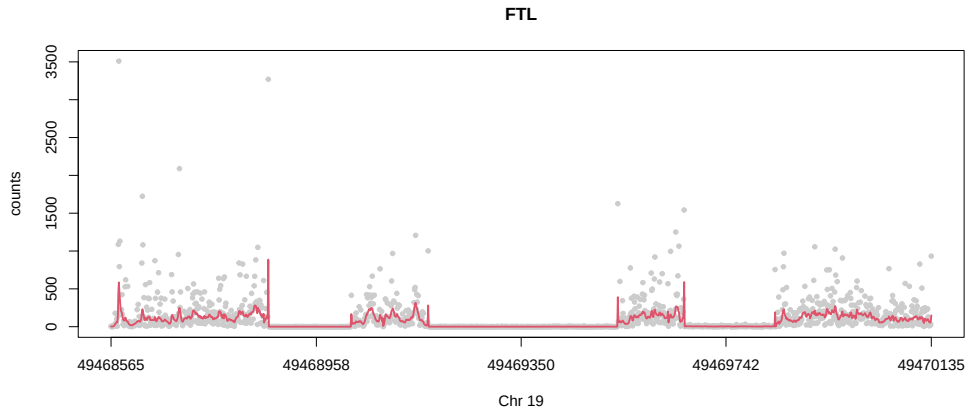
Figure B.9: Smooth RNA-seq data. VST + smash-Gaussian (heteroskedastic variance) applied to gene expression RNA-seq data.



(a) Splitting method.

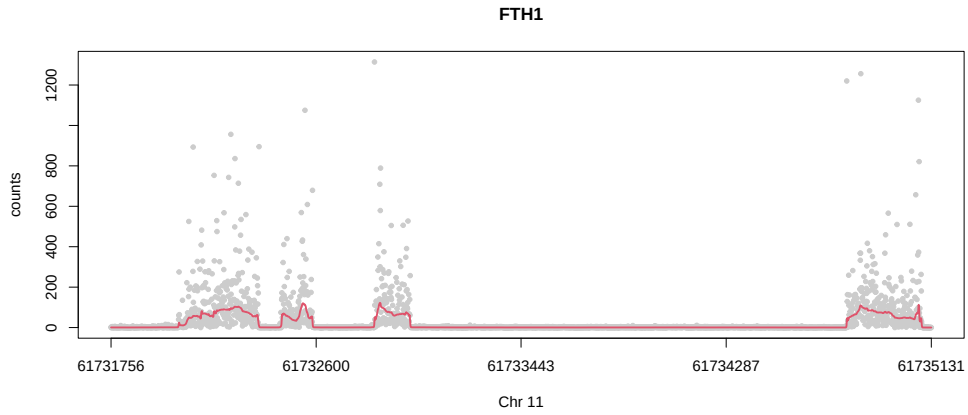


(b) Likelihood expansion method (top 30% largest counts for nugget estimation).

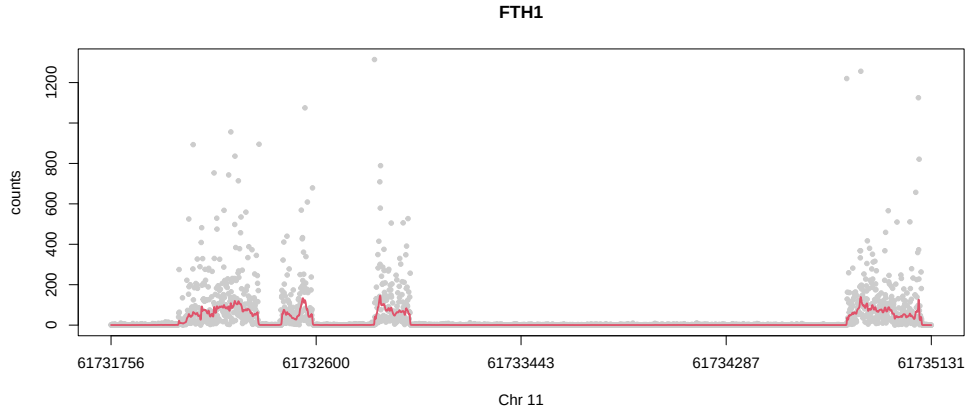


(c) Two-step method (heteroskedastic variance).

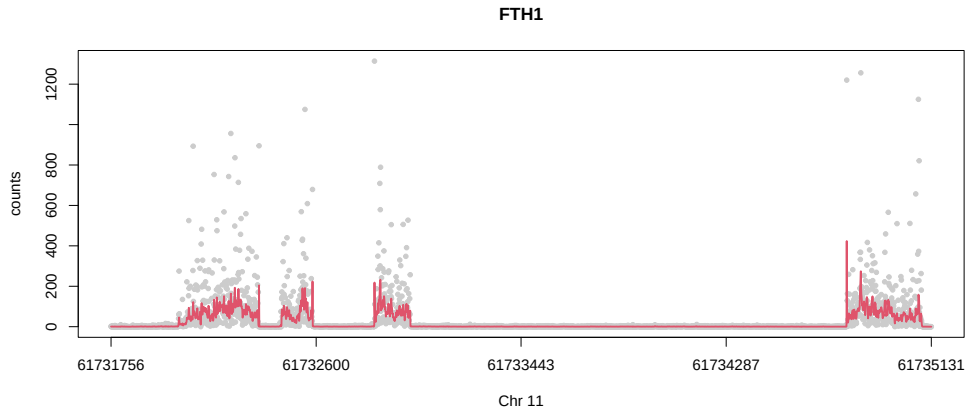
Figure B.10: Smooth RNA-seq data. Recovered expression level of FTL gene.



(a) Splitting method. $\hat{\sigma}^2 = 1.14$.



(b) Likelihood expansion method (top 30% largest counts for nugget estimation).



(c) Two-step method (heteroskedastic variance).

Figure B.11: Smooth RNA-seq data. Recovered expression level of FTH1 gene.

B.4 Additional results for ChIP-seq smoothing

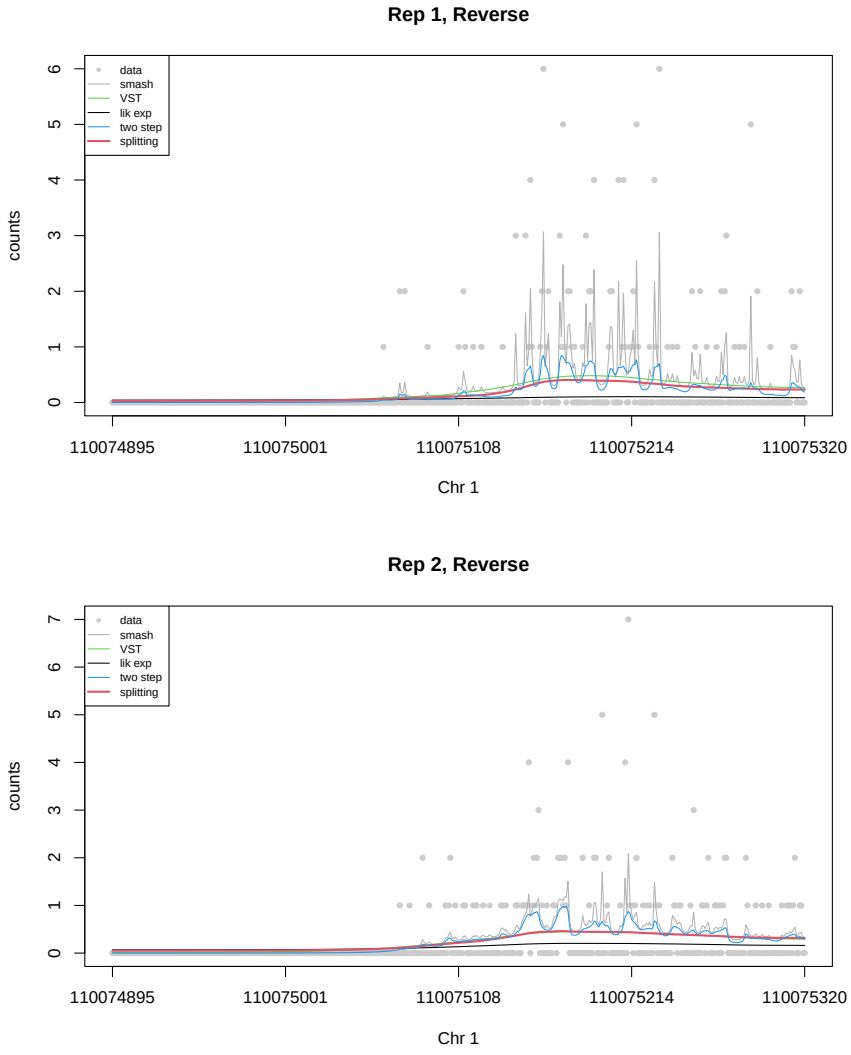


Figure B.12: Smooth ChIP-seq data. Replicate 1 and 2, reverse strand.

APPENDIX C

A SPLITTING VARIATIONAL INFERENCE APPROACH

C.1 The objective function of q_μ

We consider the induced model by integrating out b while keeps the likelihood. The marginal density (prior) of μ is

$$f(\mu; g, \sigma^2) = \int N(\mu|b, \sigma^2) g(b) db. \tag{C.1}$$

This model might be less interesting compared to the one on b. However we present the results here for completeness. The ELBO for the induced model is

$$\tilde{F}(q_\mu; g, \sigma^2) = \mathbb{E} \log \frac{p(y|\mu)}{q_\mu} + \mathbb{E} \log f(\mu; g, \sigma^2). \tag{C.2}$$

On the other hand, the profiled ELBO for $q_m u$, obtained by maxing q_b out in (4.4) is defined as

$$F(q_\mu; g, \sigma^2) = \max_{q_b} F(q_\mu, q_b; g, \sigma^2). \tag{C.3}$$

The following theorem shows that the objective function maximized by splitting approach is a lower bound of $\tilde{F}(q_\mu; \sigma^2)$.

Theorem C.1.1. *The profiled objective function* $F(q_\mu; g, \sigma^2) = \max_{q_b} F(q_\mu, q_b; \sigma^2)$ *is a lower bound of* $\tilde{F}(q_\mu; g, \sigma^2)$.

Proof. A second order Taylor series expansion of $f(\mu; g, \sigma^2)$ in $\tilde{F}(q_\mu; g, \sigma^2)$ around $\bar{\mu}$ gives

$$\begin{aligned}
\tilde{F}(q_\mu; g, \sigma^2) &= \mathbb{E} \log \frac{p(y|\mu)}{q_\mu} + \log f(\bar{\mu}; g, \sigma^2) + \frac{1}{2} \left(\frac{d^2 \log f(\mu; g, \sigma^2)}{d\mu^2} \bigg|_{\mu=\theta} \right) V_\mu, \\
&\geq \mathbb{E} \log \frac{p(y|\mu)}{q_\mu} + \log f(\bar{\mu}; g, \sigma^2) - \frac{V_\mu}{2\sigma^2} \\
&\geq \mathbb{E} \log \frac{p(y|\mu)}{q_\mu} - \frac{V_\mu}{2\sigma^2} + \max_{q_b} \mathrm{ELBO}(q_b; g, \sigma^2) \\
&= \max_{q_b} \mathbb{E} \log \frac{p(y|\mu)}{q_\mu} - \frac{V_\mu}{2\sigma^2} - \mathbb{E} \frac{(\bar{\mu} - b)^2}{2\sigma^2} - \frac{1}{2} \log 2\pi\sigma^2 + \mathbb{E} \log \frac{g(b)}{q_b} \\
&= \max_{q_b} F(q_\mu, q_b; g, \sigma^2) \\
&= F(q_\mu; g, \sigma^2),
\end{aligned}$$

where θ is between $\bar{\mu}$ and μ, and $\mathrm{ELBO}(q_b; g, \sigma^2) = \mathbb{E} \log N(\bar{\mu}; b, \sigma^2) + \mathbb{E} \log \frac{g(b)}{q_b}$ is the evidence lower bound for the empirical Bayes normal mean problem. The first inequality holds due to Lemma 4.2.2, and the second inequality is due to the definition of ELBO. □

C.2 Additional results from the simple simulation

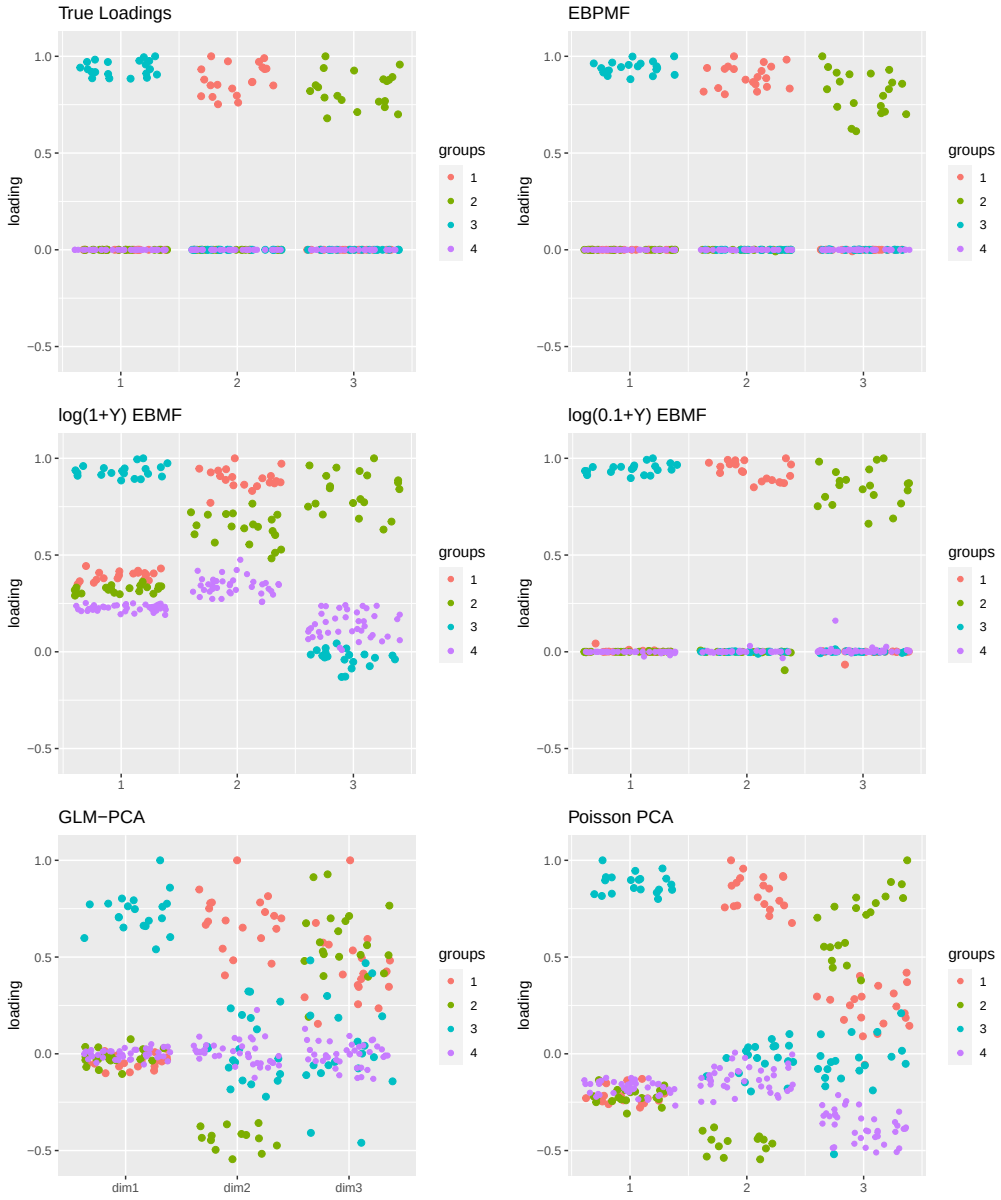
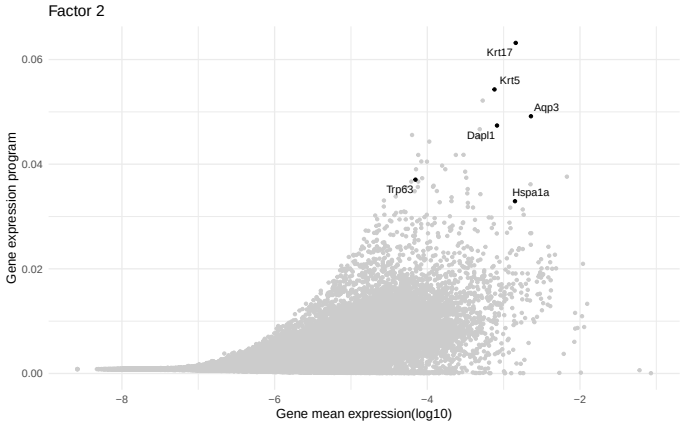
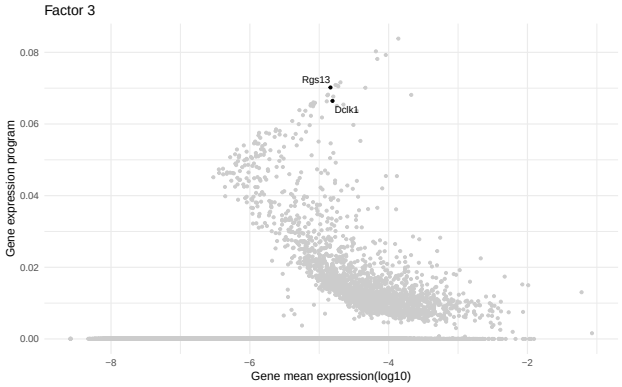


Figure C.1: Plot of the loading matrices in simulation example of EBPMF. $N = 100, p = 300, K = 3, \sigma^2_{ij} = 1$. The signs of loadings are flipped so that the largest element of each loading is positive, and scaled to be 1 for visualization purpose. In each plot, each column is a loading, and colors of dots indicate groups.

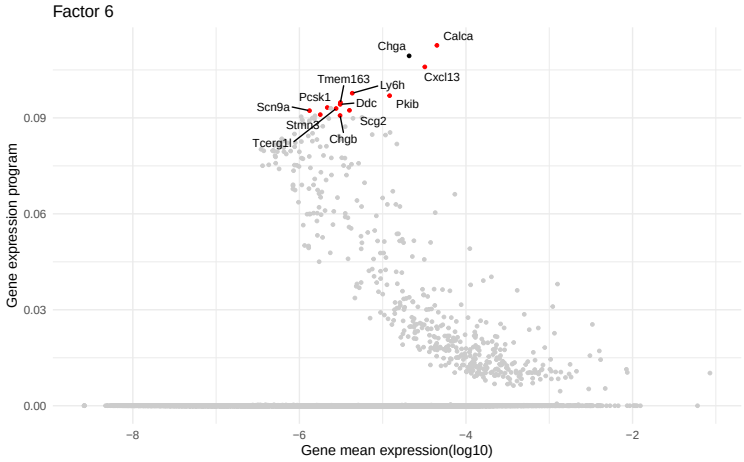
C.3 Additional GEP plots, Trachea epithelial cells



(a) Factor 2, basal cells.



(b) Factor 3, tuft cells.



(c) Factor 6, neuroendocrine cells. Genes showing on top of the plot are labelled in red, and they also match the ones detected in Montoro et al. [2018].

Figure C.2: Plot of factor 2 (basal), factor 3 (tuft) and factor 6 (neuroendocrine). Known marker genes are labelled in black.

APPENDIX D

EMPIRICAL BAYES MATRIX FACTORIZATION AND EXTENSIONS

D.1 EBPMF (identity link) rank-K model

The evidence lower bound is for rank-K EBPMF model is

$$\begin{aligned} F(q,g) &= \mathbb{E}_q \log p(\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{L},\boldsymbol{F}) - \mathbb{E}_{q_Z}\log q_Z(\boldsymbol{Z}) - D_{KL}(q_L||g_L) - D_{KL}(q_F||g_F) \\ &= \sum_{i,j}\mathbb{E}_{q_{z_{ij}}}\log\delta\left(x_{ij}-\sum_k z_{ijk}\right) + \sum_k\sum_{i,j}\mathbb{E}_q\log p(z_{ijk}|l_{ik},f_{jk}) \\ &\quad - \sum_{i,j}\mathbb{E}_{q_{z_{ij}}}\log q_{z_{ij}} - \sum_k D_{KL}(q_{\boldsymbol{l}_k}||g_{\boldsymbol{l}_k}) - \sum_k D_{KL}(q_{\boldsymbol{f}_k}||g_{\boldsymbol{f}_k}) \\ &= \sum_{i,j,k}\bar{z}_{ijk}(\overline{\log l_{ik}} + \overline{\log f_{jk}} - \log\pi_{ijk}) - \sum_{i,j,k}\bar{l}_{ik}\bar{f}_{jk} \\ &\quad - \sum_k D_{KL}(q_{\boldsymbol{l}_k}||g_{\boldsymbol{l}_k}) - \sum_k D_{KL}(q_{\boldsymbol{f}_k}||g_{\boldsymbol{f}_k}), \end{aligned} \tag{D.1}$$

where $\overline{\log l_{ik}} := \mathbb{E}\log l_{ik}$, $\overline{\log f_{jk}} := \mathbb{E}\log f_{jk}$, and $\bar{z}_{ijk} := \mathbb{E}\, z_{ijk}$.

Given q_L, q_F, conditional on the summation x_{ij}, the posterior distribution of z_{ij} is multinomial, and we have

$$q_{z_{ij}} = \text{Multinom}(\boldsymbol{z}_{ij}; x_{ij}, \boldsymbol{\pi}_{ij}), \tag{D.2}$$

where $\boldsymbol{\pi}_{ij} = (\pi_{ij1}, \pi_{ij2}, ..., \pi_{ijK})$, and

$$\pi_{ijk} = \frac{\exp(\mathbb{E}\log l_{ik} + \mathbb{E}\log f_{jk})}{\sum_{k'}\exp(\mathbb{E}\log l_{ik'} + \mathbb{E}\log f_{jk'})}. \tag{D.3}$$

Thus,

$$\bar{z}_{ijk} = x_{ij}\pi_{ijk}. \tag{D.4}$$

D.2 Extension of EBMF allowing smoothing loadings and factors

In standard factor analysis, a matrix is factorized to a product of loadings and factors. Here we consider the case where each factor is spatially-structured and we use a wavelet approach to account for the smoothness.

Consider the model

$$Y = LF^T + E, \tag{D.5}$$

where Y is an $n \times p$ matrix of observed data, L is an $n \times K$ loading matrix, F is a $p \times K$ factor matrix and E is an $n \times p$ residual matrix.

Wang and Stephens [2021] introduced empirical Bayes matrix factorization(EBMF) that assumes sparsity-inducing priors on the loadings and factors and estimates prior from the observed data. The EBMF model is

$$\begin{aligned} Y &= \sum_k \boldsymbol{l}_k \boldsymbol{f}_k^T + E, && \text{(D.6)} \\ l_{k1}, ..., l_{kn} &\sim g_{l_k}, g_{l_k} \in \mathcal{G}_l, && \text{(D.7)} \\ f_{k1}, ..., f_{kp} &\sim g_{f_k}, g_{f_k} \in \mathcal{G}_f, && \text{(D.8)} \\ E_{ij} &\sim N(0, 1/\tau_{ij}). && \text{(D.9)} \end{aligned}$$

When the rows of $\mathbb{E}\,Y$ are spatially-structured so each factor $\boldsymbol{f}_k$ is a curve, the prior g_{f_k} should properly impose the smoothness constraint. We consider the well-studied wavelet de-

noising approach. For simplicity, we assume $p = 2^T, T \in \mathbb{N}$ and $E \sim MN(0_{n\times p}, \Sigma_{n\times n}, I_{p\times p})$ where $MN(\cdot)$ is a matrix normal distribution and Σ is a diagonal matrix with elements $\sigma_i^2, i = 1, 2, ..., n$.

To induce smoothness on factors, we assume a wavelet prior (B.1) on each factor as

$$\boldsymbol{f}_k \sim g_{\boldsymbol{f}_k,\text{wavelet}}(\cdot). \tag{D.10}$$

The updates of q, g are given in algorithm 1, Wang and Stephens [2021], and they are obtained by solving an empirical Bayes normal mean (EBNM) problem. In our context, the only difference is that EBNM problem becomes the empirical Bayes wavelet denoising problem.

The objective function for updating q_{f_k}, g_{f_k} is

$$F(q_f, g_f) = \mathbb{E}_{q_{f_k}} \left(-\frac{1}{2} \sum_j (A_k f_{kj}^2 - 2B_{jk} f_{kj}) \right) + \mathbb{E}_{q_{f_k}} \log \frac{g_{f_k}}{q_{f_k}}, \tag{D.11}$$

where $A_k = \sum_i \tau_i \, \mathbb{E}_{q_l} \, l_{ik}^2$ and $B_{jk} = \sum_i \tau_i R_{ij}^k \, \mathbb{E}_{q_l} \, l_{ki}$, $R_{ij}^k = y_{ij} - \sum_{k' \neq k} \bar{l}_{k'i} \bar{f}_{k'j}$.

Due to Lemma 2 in Wang and Stephens [2021], solving the EBWD problem solves

$$\max_{q_\theta, g} F^{WD}(q_\theta, g),$$

where

$$F^{WD}(q_\theta, g) = \mathbb{E}_{q_\theta} \left(-\frac{1}{2} \sum_i (s_i^{-2} \theta_i^2 - 2 x_i s_i^{-2} \theta_i) \right) + \mathbb{E}_{q_\theta} \log \frac{g(\boldsymbol{\theta})}{q_\theta(\boldsymbol{\theta})}. \tag{D.12}$$

Thus when updating q_{f_k}, g_{f_k}, we can simply solve the EBWD problem with observations

$\frac{B_{jk}}{A_k}$ for $j = 1, 2, ..., p$ and standard deviation $A_k^{-1/2}$.

D.3 Scaled EBMF

We solve a scaled version of EBMF (Wang and Stephens [2021]). The model is

$$\begin{aligned} &Y = ALF^T B + E + \mathcal{E}, \\ &\boldsymbol{l}_k \sim g_{\boldsymbol{l}_k}, \\ &\boldsymbol{f}_k \sim g_{\boldsymbol{f}_k}, \\ &e_{ij} \sim N(0, \sigma_{ij}^2), \\ &\epsilon_{ij} \sim N(0, s_{ij}^2) \\ &A = \text{diag}(a_1, ..., a_n), \\ &B = \text{diag}(b_1, ..., b_p) \end{aligned} \tag{D.13}$$

where A, B are known diagonal scaling matrices, each entry of $\mathcal{E}$ is a random Gaussian variable with zero mean and known variance s_{ij}^2, and E is the unknown random error matrix with mean 0.

One example of the scaling factors is from the transformation of Poisson matrix by the biwhitening method introduced in Landa et al. [2022].

The objective function ELBO is

$$F(q) = \mathbb{E} \log p(Y|L, F; \boldsymbol{\sigma}^2, \boldsymbol{s}^2, A, B) + \sum_k \mathbb{E} \log \frac{g_{\boldsymbol{l}_k}}{q_{\boldsymbol{l}_k}} + \sum_k \mathbb{E} \log \frac{g_{\boldsymbol{f}_k}}{q_{\boldsymbol{f}_k}} \tag{D.14}$$

D.3.1 Update variance parameters

The objective function is

$$F(\boldsymbol{\sigma}^2) = -\sum_{i,j}\left(\log(\sigma_{ij}^2 + s_{ij}^2) + \frac{\overline{R^2}_{ij}}{\sigma_{ij}^2 + s_{ij}^2}\right), \tag{D.15}$$

where

$$\begin{aligned}\overline{R^2}_{ij} &= \mathbb{E}\left(y_{ij} - a_i b_j \sum_k l_{ki} f_{kj}\right)^2 \\ &= \left(y_{ij} - a_i b_j \sum_k \bar{l}_{ki}\bar{f}_{kj}\right)^2 + a_i^2 b_j^2 \left(\sum_k \bar{l^2}_{ki}\bar{f^2}_{kj} - \sum_k (\bar{l}_{ki})^2 (\bar{f}_{kj})^2\right)\end{aligned} \tag{D.16}$$

We can assume constant, row-specific, col-specific variances. With the appearance of s_{ij}^2, we need to solve a root-finding problem.

D.3.2 Update loadings and factors

Define $R_{ij}^k = Y_{ij} - a_i b_j \sum_{k' \neq k} \bar{l}_{k'i}\bar{f}_{k'j}$.

The objective function for updating $q_{\boldsymbol{l}_k}, g_{\boldsymbol{l}_k}$ is

$$F(q_{\boldsymbol{l}_k}, g_{\boldsymbol{l}_k}) = \mathbb{E}\left(-\frac{1}{2}\sum_i (A_{ik} l_{ki}^2 - 2B_{ik} l_{ki})\right) + \sum_k \mathbb{E}\log\frac{g_{\boldsymbol{l}_k}}{q_{\boldsymbol{l}_k}}, \tag{D.17}$$

where

$$\begin{aligned}A_{ik} &= a_i^2 \sum_j b_j^2 \bar{f^2}_{jk}/(\sigma_{ij}^2 + s_{ij}^2), \\ B_{ik} &= a_i \sum_j R_{ij}^k b_j \bar{f}_{jk}/(\sigma_{ij}^2 + s_{ij}^2).\end{aligned} \tag{D.18}$$

Based on the prior of $\boldsymbol{l}_k$, we can solve an ebnm problem with $x_i = B_{ik}/A_{ik}$ and $s_i^2 = 1/A_{ik}$.

For updating $q_{\boldsymbol{f}_k}, g_{\boldsymbol{f}_k}$, we have

$$\begin{aligned} A_{jk} &= b_j^2 \sum_i a_i^2 \bar{l^2}_{ik}/(\sigma_{ij}^2 + s_{ij}^2), \\ B_{jk} &= b_j \sum_i R_{ij}^k a_i \bar{l}_{ik}/(\sigma_{ij}^2 + s_{ij}^2). \end{aligned} \tag{D.19}$$

D.4 Empirical Bayes multiscale Poisson smoothing

For Poisson data, a multiscale decomposition, an analogy to the Haar wavelet transformation was studied in Kolaczyk [1999] and Timmermann and Nowak [1999]. Suppose we observe a Poisson sequence $x_i \sim \text{Poisson}(s\lambda_i)$, for $i = 1, 2, ..., n$, $n = 2^J$ where J is a positive integer, and s is a known positive scaling scalar. A Haar wavelet like decomposition of the sequence x_i is defined as follows. Let $j = 0, 1, ..., J-1$ denote scale and $l = 0, 1, ..., 2^j - 1$ denote location. The Poisson multiscale decomposition of a Poisson sequence is based on the following properties of Poisson distributions

$$\begin{aligned} &x_1 + x_2 \sim \text{Poisson}(s(\lambda_1 + \lambda_2)), \\ &x_1|(x_1 + x_2) \sim \text{Binom}(x_1 + x_2, \lambda_1/(\lambda_1 + \lambda_2)). \end{aligned}$$

The parameters λ_1, λ_2 are re-parameterized to $\mu = \lambda_1 + \lambda_2, p = \lambda_1/(\lambda_1 + \lambda_2)$. For a length $N = 2^J$ sequences, we can re-parameterize $\boldsymbol{\lambda}$ to $\mu = \boldsymbol{\lambda}^T \mathbf{1}$ and binomial probabilities $\boldsymbol{p} = (p_{jl})$, where $|\boldsymbol{p}| = N - 1$. Following this decomposition, we can re-write $\boldsymbol{x}$ in a recursive

dyadic partition (RDP) way as

$$\begin{aligned} x_{J,l} &:= x_{l+1}, \text{ for } l = 0, 1, ..., N-1, \\ x_{j,l} &:= x_{j+1,2l} + x_{j+1,2l+1}. \end{aligned} \tag{D.20}$$

The likelihood of sequence $\boldsymbol{x}$ is

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{\lambda}) &= \prod_{i}^{n} p(x_i|\lambda_i) \\ &= p(x_{0,0}|\mu) \times \prod_{j=0}^{J-1} \prod_{l=0}^{2^j-1} p(x_{j+1,2l}|x_{j,l}, p_{j,l}) \\ &= \text{Poisson}(x_{0,0}; \mu) \times \prod_{j=0}^{J-1} \prod_{l=0}^{2^j-1} \text{Binom}(x_{j+1,2l}; x_{j,l}, p_{j,l}). \end{aligned} \tag{D.21}$$

One can estimate a spatially-structured $\boldsymbol{\lambda}$ by shrinkage estimation of $\boldsymbol{p}$ towards 1/2, as in Kolaczyk [1999] and Timmermann and Nowak [1999]. Or equivalently, shrinkage estimation of the log odds ratio $\alpha = \log \frac{p}{1-p}$ towards 0, as in Xing et al. [2021]. Similar to the wavelet denoising for homogeneous Gaussian data, current multiscale Bayesian methods transform original sequence $\boldsymbol{x}$ to empirical multiscale coefficients $x_{j,l}$ and assume a shrinkage prior on p_{jl} for each scale. Hence for each level, we solve the following problem

$$\begin{aligned} x_{j+1,2l}|p_{j,l} &\sim \text{Binom}(x_{j,l}, p_{j,l}), \\ p_{j,l} &\sim g_j(\cdot). \end{aligned} \tag{D.22}$$

After shrinkage estimation of p_{jl} for each scale separately, an estimate of $\boldsymbol{\lambda}$ can be obtained by transforming the decomposed sequence back to original space.

The multiscale Bayesian method has been shown to work well in practice, however, it's unclear what the objective function is. In homogeneous Gaussian denoising problem, the DWT

matrix is orthogonal so the empirical Wavelet coefficients are all independent across scale and locations. While the Poisson multiscale decomposition is not an orthogonal transformation and $x_{j,l}$ are clearly not independent across scales. Thus it's unclear what the objective function the method is optimizing. Here we re-formulate the problem in a variational Bayes context and show that the method is maximizing an evidence lower bound.

D.4.1 Empirical Bayes Binomial probability

We first introduce a sub-problem called Empirical Bayes Binomial Probability (EBBP). Assume the model

$$\begin{aligned} x_i|p_i &\sim \text{Binom}(n_i, p_i) \\ p_i &\sim g(\cdot). \end{aligned} \tag{D.23}$$

Since our primary goal is to shrink p_i towards a half, we put the following mixture of beta prior (Timmermann and Nowak [1999]) on p_i with mode 1/2,

$$g(\cdot) = \pi_0 \delta(0.5) + \sum_k \pi_k \,\text{Beta}(a_k, a_k), \tag{D.24}$$

where a_k are fixed and span a large grid. The primary reason of the prior choice is its conjugacy, such that the marginal likelihood and posterior can be expressed in closed forms. An empirical Bayes procedure proceeds with the following two steps:

1. Estimate $\boldsymbol{\pi}$ by $\hat{\boldsymbol{\pi}} = \arg\max_{\boldsymbol{\pi}} \log l(\boldsymbol{x}|\boldsymbol{\pi})$ where $l(\boldsymbol{x}|\boldsymbol{\pi}) = \int p(\boldsymbol{x}|\boldsymbol{p}) g(\boldsymbol{p}) d\boldsymbol{p}$.

2. Calculate the posterior $p(p_i|y_i, n_i, \hat{\boldsymbol{\pi}})$.

The log marginal likelihood of $\boldsymbol{x}$ is

$$l(\boldsymbol{x}|\boldsymbol{\pi}) = \sum_{i=1}^{n} \log \left(\pi_0 \operatorname{Binom}(x_i|n_i, \frac{1}{2}) + \sum_{k=1}^{K} \pi_k \operatorname{Bb}(x_i|a_k, a_k, n_i) \right), \tag{D.25}$$

where $\operatorname{Bb}(x|\alpha, \beta, n)$ is the pdf of Beta-binomial distribution.

The posterior distribution is then

$$p(p_i|x_i, n_i, \hat{\pi}) = \tilde{\pi}_{i0}\delta_{\frac{1}{2}}(p_i) + \sum_{k=1}^{K} \tilde{\pi}_{ik} \operatorname{Beta}(p_i; a_k + x_i, a_k + n_i - x_i), \tag{D.26}$$

where

$$\begin{aligned} \tilde{\pi}_{i0} &= \frac{\hat{\pi}_0 \operatorname{Binom}(x_i|n_i, \frac{1}{2})}{\hat{\pi}_0 \operatorname{Binom}(x_i|n_i, \frac{1}{2}) + \sum_{k=1}^{K} \hat{\pi}_k \operatorname{Bb}(x_i|a_k, a_k, n_i)}, \\ \tilde{\pi}_{ik} &= \frac{\hat{\pi}_k \operatorname{Bb}(x_i|a_k, a_k, n_i)}{\hat{\pi}_0 \operatorname{Binom}(x_i|n_i, \frac{1}{2}) + \sum_{k=1}^{K} \hat{\pi}_k \operatorname{Bb}(x_i|a_k, a_k, n_i)}. \end{aligned} \tag{D.27}$$

For given probability densities on p_i, denoted as q_i, the objective function ELBO of EBBP problem is

$$F(q, g) = \mathbb{E}_q \sum_i \log p(x_i|p_i) - \sum_i D_{KL}(q_i(p_i)||g(p_i)), \tag{D.28}$$

where $F(q, g)$ achieves its maximum at $\hat{q}_i = p(p_i|x_i, n_i, \hat{\pi})$ and $\hat{g} = \hat{\boldsymbol{\pi}}$, and the maximum value is the marginal likelihood $l(\boldsymbol{x}|\hat{g})$. This follows from the definition of ELBO.

D.4.2 Variational Bayes inference for Poisson multiscale smoothing problem

We can write the parameters $\boldsymbol{\lambda}$ as a function of $\mu, \boldsymbol{p}$

$$\lambda_i = \mu \prod_{j=0}^{J-1} (p_{j,l(i)})^{\epsilon_{ij}} (1 - p_{j,l(i)})^{1-\epsilon_{ij}}, \tag{D.29}$$

where $\epsilon_{ij} = 1$ if the λ_i goes to the left children node at scale j, and $l(i)$ is the location of λ_i at scale j. The full model is then

$$\begin{aligned} x_i &\sim \text{Poisson}(s\lambda_i), \\ \lambda_i &= \mu \prod_{j=0}^{J-1} (p_{j,l(i)})^{\epsilon_{ij}} (1 - p_{j,l(i)})^{1-\epsilon_{ij}}, \\ p_{j,l} &\sim g_j(\cdot). \end{aligned} \tag{D.30}$$

We refer the prior on λ_i, induced by Eq.(D.29) and priors on $p_{j,l}$, as a multiscale prior. Assume the posterior distribution factorizes as

$$q(\boldsymbol{p}) = \prod_{j,l} q_{j,l}(p_{j,l}). \tag{D.31}$$

Then the ELBO is

$$\begin{aligned} F(q,g) &= \mathbb{E}_q \log p(\boldsymbol{x}|\boldsymbol{\lambda}) - D_{KL}(q(\boldsymbol{p})||g(\boldsymbol{p})) \\ &= \log p(x_{0,0}; s\mu) \\ &\quad + \sum_j \mathbb{E} \sum_l \log \text{Binom}(x_{j+1,2l}; x_{j,l}, p_{j,l}) - D_{KL}(q_{jl}(p_{j,l})||g_j(p_{j,l})) \\ &:= \log p(x_{0,0}; s\mu) + \sum_j F_j(q,g). \end{aligned} \tag{D.32}$$

The estimate of μ is $\hat{\mu} = \sum_i x_i/s$. Notice that $F_j(\cdot,\cdot)$ has the same form as the EBBP objective function (D.28). To maximize $F(q,g)$, we can perform the EBBP for each scale j, which is exactly the multiscale Bayesian method. We denote the mapping from $(\boldsymbol{x}, s)$ to $(\hat{g}, q)$ by the variational empirical Bayes Poisson multiscale smoothing method as $\text{EBMS}(\boldsymbol{x}, s) = (\hat{g}, q)$.