Distributed and Multiphase Inference in Theory and Practice: Principles, Modeling, and Computation for High-Throughput Science

A dissertation presented

by

Alexander W. Blocker

to

The Department of Statistics

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the subject of

Statistics

Harvard University

Cambridge, Massachusetts

April 2013

Dissertation Advisors: Professors Xiao-Li Meng, Edoardo Airoldi Alexander W. Blocker

Distributed and Multiphase Inference in Theory and Practice: Principles, Modeling, and Computation for High-Throughput Science

Abstract

The rise of high-throughput scientific experimentation and data collection has introduced new classes of statistical and computational challenges. The technologies driving this data explosion are subject to complex new forms of measurement error, requiring sophisticated statistical approaches. Simultaneously, statistical computing must adapt to larger volumes of data and new computational environments, particularly parallel and distributed settings. This dissertation presents several computational and theoretical contributions to these challenges.

In chapter 1, we consider the problem of estimating the genome-wide distribution of nucleosome positions from paired-end sequencing data. We develop a modeling approach based on nonparametric templates that controls for variability due to enzymatic digestion. We use this to construct a calibrated Bayesian method to detect local concentrations of nucleosome positions. Inference is carried out via a distributed HMC algorithm that scales linearly in complexity with the length of the genome being analyzed. We provide MPI-based implementations of the proposed methods, stand-alone and on Amazon EC2, which can provide inferences on an entire S. cerevisiae genome in less than 1 hour on EC2.

We then present a method for absolute quantitation from LC-MS/MS proteomics experiments in chapter 2. We present a Bayesian model for the non-ignorable missing data mechanism induced by this technology, which includes an unusual combination of censoring and truncation. We provide a scalable MCMC sampler for inference in this setting, enabling full-proteome analyses using cluster computing environments. A set of simulation studies and actual experiments demonstrate this approach's validity and utility.

We close in chapter 3 by proposing a theoretical framework for the analysis of preprocessing under the banner of multiphase inference. Preprocessing forms an oft-neglected foundation for a wide range of statistical and scientific analyses. We provide some initial theoretical foundations for this area, including distributed preprocessing, building upon previous work in multiple imputation. We demonstrate that multiphase inferences can, in some cases, even surpass standard single-phase estimators in efficiency and robustness. Our work suggests several paths for further research into the statistical principles underlying preprocessing.

Contents

Dedicated to Paula.

Author List

The authors for Chapter 1 are Alexander W Blocker and Edoardo M Airoldi.

The authors for Chapter 2 are Alexander W Blocker, Eric J Solís, and Edoardo M Airoldi.

The authors for Chapter 3 are Alexander W Blocker and Xiao-Li Meng.

Acknowledgments

I would like to thank my advisors Professor Xiao-Li Meng and Professor Edoardo Airoldi first and foremost. Both have been incredibly supportive mentors and collaborators throughout my graduate studies. I have been continually amazed by the depth and breadth of Xiao-Li's knowledge and scholarship, and it has been a wonderful privilege to work with him over the past five years. I will endeavor to keep seeking out the subtleties of statistics, as he has shown me. Edo has been an invaluable guide to the worlds of networks, machine learning, and computational biology, as well as a coauthor, a collaborator, and a friend.

I would also like to thank the entire faculty of the Department of Statistics for their support and wisdom. They created a wonderful intellectual environment both inside and outside of the classroom that I was very lucky be a part of for a few years. The courses I took with Joe Blitzstein, Carl Morris, Jun Liu, and Don Rubin opened my eyes to the world of statistics beyond my original training in econometrics. Everyone I have taught with over the past five years has also helped me to grow and take on challenges I never expected to, for which I thank them. I would also like to particularly thank Jun Liu for becoming a reader for this dissertation late in the process.

Collaborative research has been at the heart of my graduate career, and I would like to extend my thanks everyone I have worked with. Xu Zhou and Erin O'Shea have provided invaluable scientific guidance and perspective on my nucleosome research. On my proteomics research, Eric Solís, David Allan Drummond, and Bogdan Budnik have been excellent guides to the technically challenging world of mass spectrometry. Throughout my time at Harvard, the CHASC group has provided a stimulating research environment and valuable feedback.

My colleagues have been a source of advice and inspiration throughout my time at Harvard. I thank all of you for your patience as we worked through theoretical and computational problems and for all of the wonderful times we had throughput my studies at Harvard.

I would also like to thank my mother Kelley Weaver for her support through everything. Last but certainly not least, thank you to my wonderful fiancée Paula Griffin. This dissertation would not have been possible without her patience and support.

1

Template-based estimation of genome-wide nucleosome positioning via distributed HMC

1.1 Introduction

The organization of genetic material within cells plays a major role in the regulation of biological activities. In the cell, DNA is wrapped around histone proteins to form nucleosomes, which constitute the smallest units of such organization. DNA must be accessible for transcription to occur, thus the presence of nucleosomes physically constrains regu-

lation. High-throughput sequencing technology produces indirect noisy evidence about the positions of nucleosomes across an entire genome, with an unprecedented resolution. In this paper, we develop methods to provide accurate, reproducible estimates of nucleosome positions across a genome from high-throughput sequencing data, enabling the investigation of fine-grained structure in nucleosome positioning and its regulatory role.

We consider high-throughput sequencing data derived from micrococcal nuclease digestion (Tirosh, 2012). Briefly, this technique involves linking histone proteins to the target DNA wrapped around them, digesting the remaining DNA using an enzyme, then digesting the histone proteins to make the target DNA accessible for further processing (e.g., see Tsankov et al., 2010). A gel is used to select DNA fragments with an approximate length of 150 base pairs—the length of DNA wrapped around each nucleosome. These fragments are amplified via PCR and sequenced (Albert et al., 2007a). The resulting sequences, or reads, are aligned to a reference genome for the organism of interest using standard software (Bowtie; Langmead et al., 2009). The data consists of the number of read centers that align to each base pair along the genome. We analyze data obtained with paired-end sequencing technology; that is, each DNA fragment is sequenced simultaneously from both ends, and the two reads are recorded as a pair. This technology provides the length of each fragment, in addition to its location, following alignment of the paired reads. Figure 1.1 illustrates some example data.

1.1.1 Related work

The positioning of nucleosomes along the genome was first studied with tiling microarrays (Yuan et al., 2005; Segal et al., 2006; Lee et al., 2007). High-throughput sequencing data allows for the analysis nucleosome positioning in any organism, and overcomes many technical limitations of tiling microarrays (Jansen and Verstrepen, 2011). The first wave

of studies using high-throughput sequencing to infer nucleosome positioning used single-end sequencing technology (Albert et al., 2007b; Shivaswamy et al., 2008; Tsankov et al., 2010). More recent studies have used paired-end sequencing technology (Gkikopoulos et al., 2011).

The statistical approach to identifying nucleosome positions form tiling microarray data consisted largely of hidden Markov models and their variants (Gupta, 2007; Yuan and Liu, 2008; Yassour et al., 2008; Sun et al., 2009b; Mitra and Gupta, 2011), with some mixture model approaches also in use (Sun et al., 2009a). Most analyses of sequencing data have adopted Parzen-window based estimators, which convolve the observed read counts within a window, extract local maxima, and perform subsequent computations based on taking these maxima as nucleosome positions. Variants of this technique include the use of multiple windows (Weiner et al., 2010), frequency-based filtering using fast Fourier transformation (FFT) (Flores and Orozco, 2011), and a Kolmogorov-Smirnov based method for detecting differences in nucleosome positioning between samples (Fu et al., 2012). Others have adapted HMMs to this class of data (Cairns et al., 2011). Model-based analyses of sequencing data have focused on mixture models (Polishko et al., 2012; Rashid et al., 2011; Zhang et al., 2012). Recent work combines a new biochemical protocol with a Bayesian deconvolution method (Brogaard et al., 2012); however, their inference

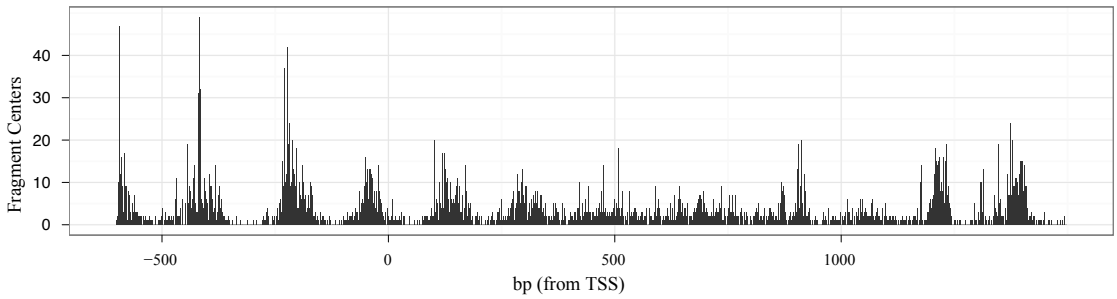

Figure 1.1: Example data for yeast gene PHO5.

procedure targets different estimands.

Methodology relevant to the problem we consider has been developed to infer transcription-factor (TF) binding sites from high-throughput ChIP-seq data (Park, 2009). Analyses of ChIP-seq data often combine variants of Parzen window estimation (Schwartzman et al., 2011), a Poisson model for sequence counts (Zhang et al., 2008), and detection methods for peak finding (Pepke et al., 2009). From a statistical perspective, a key feature of ChIP-seq data is that the TF binding sites are non-overlapping. This allows for independence assumptions in models for ChIP-seq data (Barski and Zhao, 2009) that cannot be defended in models of nucleosomes, which are likely to overlap when cell populations are sequenced.

1.1.2 Contributions of this article

We develop a *template-based approach* for estimating the genome-wide distribution of nucleosome positions from paired-end sequencing data. This approach uses information on fragment lengths provided by paired-end sequencing to estimate the amount of variation due to enzymatic digestion in each lane of sequencing data. Using this information, we posit a model that captures both the variation of read positions due to enzymatic digestion and the variation due other sources of experimental error, in Section 1.2. This model incorporates a hierarchical structure within discrete segments of the genome to provide local regularization. We also introduce a set of novel estimands that provide interpretable summaries of the genome-wide distribution of nucleosome positions.

We develop a parallel Hamiltonian Markov Chain Monte Carlo sampler to draw from the posterior distribution of the quantities of interest under our model, in Section 1.3. This sampler is highly amenable to distributed computation and scales linearly with the length of the genome being analyzed. We provide a non-parametric estimator of the distribution of digestion errors and propose a segmentation algorithm that splits the genome

in regions of similar coverage, respecting biological features. We introduce a calibrated Bayesian method with frequentist error guarantees, to detect local concentrations of nucleosome positions.

We demonstrate the proposed methods on real and simulated data in Section 1.4, assessing the accuracy and reproducibility of the inferences. We also compare the performance of our methods to the popular Parzen-window and read-based estimators.

1.2 Model

Here we develop a model for paired-end reads, obtained using Solexa high-throughput sequencing technology. The data consist of integer counts y_k of the fragment centers observed at each base pair k along an N-base pair long chromosome, together with the corresponding fragment lengths l_j for each of the M observed fragments, which provide information about how far apart the paired reads are.

The proposed model consists of two distinct components: an observation model $p(\vec{y}|\vec{\beta})$, which provides the distribution of the observed read counts given the underlying distribution of nucleosome positions $\vec{\beta}$, and a positioning model $p(\vec{\beta}|\vec{\mu}, \vec{\sigma}^2)$, which describes the structure of the nucleosome position distribution. Given a segmentation function, $s : \{1 \dots N\} \to \{1 \dots S\}$, which maps the N base pair locations to S regions in which coefficients β_k can be assumed to be identically distributed, we posit

$$y_k|\lambda_k \sim \text{Poisson}\,(\lambda_k) \tag{1.1}$$

$$\vec{\lambda}_{(N\times 1)} \equiv X_{(N\times(N-2\lfloor \ell_o/2 \rfloor))}\, \vec{\beta}_{((N-2\lfloor \ell_o/2 \rfloor)\times 1)}, \tag{1.2}$$

$$\beta_k > 0 \text{ for } k = \lfloor \ell_o/2 \rfloor + 1 \dots N - \lfloor \ell_o/2 \rfloor$$

$$\log \beta_k \sim \text{Normal}\,(\mu_{s_k}, \sigma^2_{s_k}) \tag{1.3}$$

where X specifies the contribution of a nucleosome positioned at base pair k to the expected number of reads at base pair m due to digestion variability, and $s(k)$ is denoted as s_k for compactness. (The construction of the matrix X is detailed in Section 1.2.1.) The log-likelihood for the proposed model is as follows, subject to the positivity constraint on $\vec{\beta}$,

$$\begin{aligned} \log p(\vec{y}|\vec{\theta},\vec{\mu},\vec{\sigma}^2) &= -\sum_k \vec{x}_k^T\vec{\beta} + \sum_k y_k \log\left(\vec{x}_k^T\vec{\beta}\right) \\ &\quad -\frac{1}{2}\sum_k \log\sigma^2_{s_k} - \frac{1}{2}\sum_k \frac{(\theta_k-\mu_{s_k})^2}{\sigma^2_{s_k}} + \text{const.} \end{aligned} \tag{1.4}$$

To complete the model specifications, we place priors on μ_s and σ_s^2. We use independent conjugate priors for σ_s^2, assuming $\sigma_s^2 \sim \text{InvGamma}(\alpha_o, \gamma_o)$. Our priors for μ_s are fully conjugate and independent across segments; we assume $p(\mu_s|\sigma_s^2) \sim N(\mu_o, \frac{\sigma_s^2}{n_s\tau_o})$ where n_s is the length of segment s. These stabilize our inferences and reflect vague prior information on the distribution of $\vec{\beta}$. This is particularly true for our prior on σ_s^2, which regulates the uniformity of nucleosome positioning. Their form also allows for efficient computation, as outlined in Section 1.3. We analyze the sensitivity of the inferences to the choice of (μ_o, τ_o, α_o, γ_o) in Section 1.4.

The proposed model depends upon two technical constructs: digestion-variability templates and a segmentation of the DNA sequence. We discuss them further in the next two subsections, before introducing the estimands of interest in Section 1.2.3.

1.2.1 Digestion variability template

A template summarizes variation due to enzymatic digestion in a single lane of sequencing and it is used to build the X matrix in Equation 1.2. We cover the paired-end case here and discuss extensions to single-end sequencing in Section 1.5. Consider a simple model for

the variability of enzymatic digestion. We denote the length of each fragment j as ℓ_j and assume

$$\ell_j = \ell_o + e_{1j} + e_{2j}, \quad e_{1j}, e_{2j} \sim IID\ . \tag{1.5}$$

ℓ_o is the base length of each fragment and the $e_{\cdot j}$ terms are the digestion errors at each end of the fragment. We assume these errors are bounded and symmetric between the ends of each fragment; physically, this means that the enzyme has the same propensity towards over-or under-digestion at each end. Under this model, each fragment's center varies about its nucleosome's true center according to the distribution of $d_j \equiv \frac{1}{2}(e_{1j} - e_{2j})$. Our template is this distribution, expressed in vector form and transformed to account for the random rounding of fragment centers to integer positions. Hence,

$$t_k = P(d_j = k) + \frac{1}{2}\left(P(d_j = k - \frac{1}{2}) + P(d_j = k + \frac{1}{2})\right) \tag{1.6}$$

for $k = -w, \ldots, w$, yielding a vector $\vec{t}$ of length $2w+1$. We estimate the template from the empirical fragment length distribution corresponding to a lane of paired-end sequencing data, as detailed in Section 1.3.1. Example exact and approximate templates for the same data are shown in Figure 1.2.

The matrix X in Equation 1.2 is constructed using a template $\vec{t}$ by leveraging an equivalence between a realistic data-generating process and the marginal specification given in Equations 1.1–1.2. Briefly, an explicit model would combine a Poisson distribution for the unobservable number of reads that are generated from a given nucleosome location, with a multinomial distribution that controls the offsets of the observed read centers from the center of that nucleosome, which is where they would all be observed in the absence of digestion variability. This Poisson-multinomial structure for the observed reads is marginally equivalent to the more convenient Poisson GLM with an identity link func-

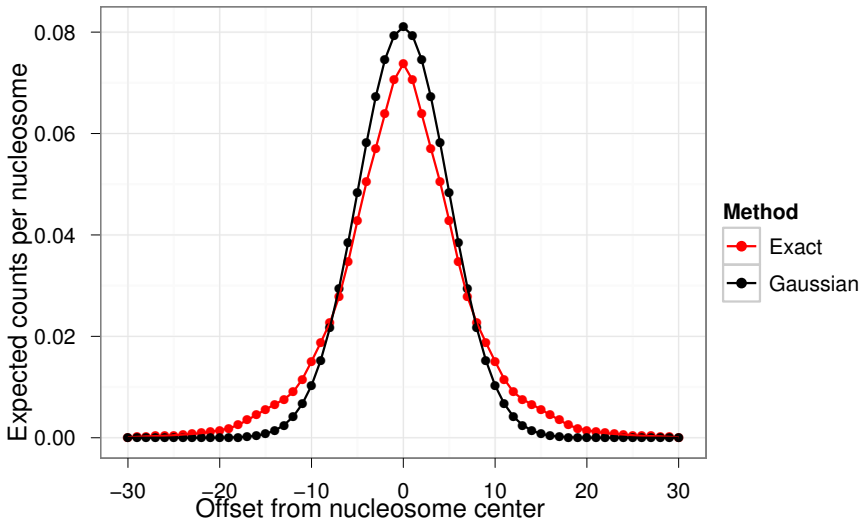

Figure 1.2: Example templates for yeast growing in high-phosphate.

tion specified in Equations 1.1–1.2.

In detail, denote the length of the sequence of interest N, and the width of the template $2w + 1$ as above. Then we can define the digestion (or basis) matrix X as an $(N \times N - 2\lfloor \ell_o/2 \rfloor)$ matrix where each row corresponds to a shifted version of the template. The matrix X is fully specified as follows,

$$X = \begin{pmatrix} \vec{t} & & & & \\ & \vec{t} & & & \\ & & \ddots & & \\ & & & \vec{t} & \\ & & & & \vec{t} \end{pmatrix} \tag{1.7}$$

Using the $(N - 2\lfloor \ell_o/2 \rfloor)$-dimensional constrained vector of coefficients $\vec{\beta} \geq 0$, we obtain an N-dimensional vector of expected counts $\vec{\lambda}$ using Equation 1.2. Each coefficient β_k provides the number of fragments we expect to sequence from nucleosomes centered at

position k. Analogously, $\vec{\lambda}$ provides the number of fragment centers we expect to observe at each position. Formally, $\vec{\lambda}$ is a convolution of $\vec{\beta}$ with $\vec{t}$. This structure models the effect of digestion variability on the observations.

Digestion variability affects the statistical properties of our data in two important ways under this model. First, the expected counts of fragment centers are convolved with the digestion variability template. This reduces the concentration of counts at each nucleosome position, obscuring the true center of the nucleosome. Second, as digestion variability convolves the expected fragment counts over a broader stretch of the genome, the expected number of counts at each base pair decreases, driving down the signal-to-noise ratio. This phenomenon is not unique to Poisson noise, but it is particularly acute in this setting because the signal-to-noise ratio of a Poisson random variable is equal to its expectation. The combination of these effects makes inferring nucleosome positions very challenging in this setting, even in high-coverage experiments. The resulting combination of "vertical" noise (from Poisson-lognormal variation) and "horizontal" convolution across the sequence (from digestion variability) creates a challenging deconvolution problem.

1.2.2 Segmentation

Our segmentation of the DNA sequence accounts for variation in occupancy, coverage, and structure. The goal is to split chromosomes into local regions where the IID assumption on the coefficients β_k appear sensible. The segmentation function s defined above must fulfill a monotonicity condition, $s(k+1) - s(k) \in \{0, 1\}$, so that segments are indexed contiguously and in strictly increasing order. An example segmentation of yeast chromosome I is shown in Figure 1.3.

Statistically, the segmentation enables local regularization in the estimation of $\vec{\beta}$. These coefficients are weakly identified in a model specified by Equations 1.1–1.2 alone. Such a

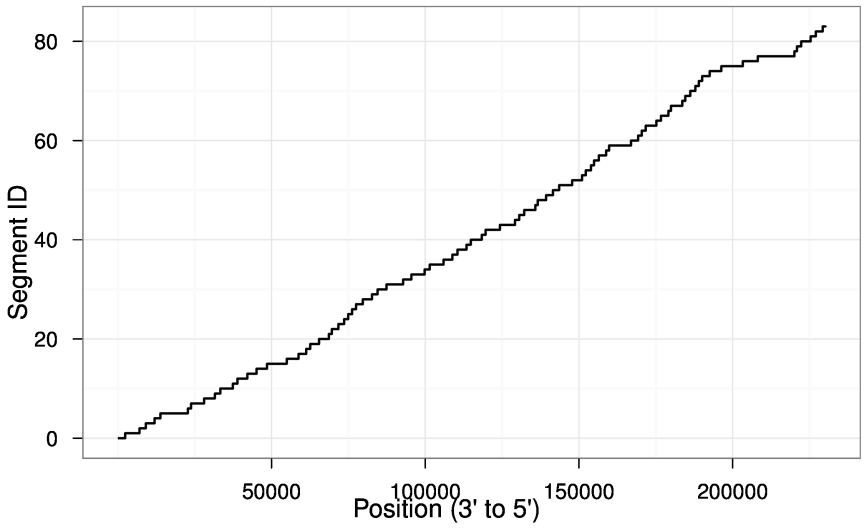

Figure 1.3: Example segmentation of yeast chromosome I. See Section 1.3 for estimation details.

model would involve $N-\ell_o$ parameters and N observations, and the Hessian matrix for the implied log-likelihood of $\vec{\beta}$ would be $H = -X^T W X$, where $W = \text{diag}\left(y_1/\vec{\lambda}_1^2, \ldots, y_n/\vec{\lambda}_n^2\right)$. This is negative-definite if $\vec{y}$ contains no all-zero subvector of length $2w+1$ or more; otherwise, it is only negative semi-definite. Furthermore, H is typically very ill-conditioned due to the convolution structure of X. Estimates of β_k from this model would be extremely unstable. We regularize the estimates of β_k by modeling the distribution of nucleosome positions with Equation 1.3. In this complete model, we pool information locally within each chromosome, as β_j is independent of β_k if $s_j \neq s_k$, where s_k is the segment to which base pair k belongs.

Segments divide each chromosome into local stretches over which a consistent distribution of nucleosome positions is plausible. We posit a log-normal distribution for the magnitudes of the coefficients β_k. The idea is that most locations on the sequence are expected have a very low concentration of nucleosome positions. Such locations correspond to small, non-zero values of β_k. A few locations have a relatively high concentration of nu-

cleosomes across a population of cells. These are the positions of interest, corresponding to large values of β_k. The log-Normal distribution captures such behavior: it allows the majority of values in $\vec{\beta}$ to concentrate around a low baseline rate with a few values many orders of magnitude larger than the baseline. The parameters μ_{s_k} and $\sigma^2_{s_k}$ control this baseline and the prevalence of extreme values in $\vec{\beta}$, providing us with a flexible, parsimonious way to regularize our estimation and provide more reliable inferences.

The segmentation also provides a way to control the bias-variance trade-off of our regularization. Using a large number of short segments results in low bias, as they can capture sequence features at a fine scale; however, this also leads to greater uncertainty, as more parameters are introduced and less observations are available for regularization within each segment. Using a smaller number of longer segments produces the opposite effect. We discuss a strategy to managing this trade-off in Section 1.3.2.

1.2.3 Estimands

We can express the scientific estimands of interest as functions of $\vec{\beta}$. The parameter $\vec{\beta}$ itself is of interest, as it captures the pattern of nucleosome positioning across each chromosome. However, $\vec{\beta}$ is high-dimensional and unsuitable for human interpretation. The posterior expectation, standard deviation, and quantiles of $\vec{\beta}$ are useful for visualization and exploratory analysis. Below, we develop several more refined estimands to quantify the structure of the nucleosome position distribution. These new estimands fall into two broad categories: (1) local measures of concentration, and (2) cluster-level summaries of structure.

The first family of estimands aims to quantify the relative concentration of nucleosome centers within a local window. Formally, for each base pair location k in $\vec{\beta}$, we consider the

ratio

$$C_{p,l}(k) = \frac{\sum_{i=-p}^{p} \beta_{k+i}}{\sum_{i=-l}^{l} \beta_{l+i}}, \tag{1.8}$$

where $2l + 1$ is the width of a local window and $2p + 1$ is the width of the region of interest. We typically choose $l = 73$, yielding a local window of width 147. For $l \leq 73$, the structure of $\vec{\beta}$ within $2l + 1$ bp windows can be taken as measure of the distribution of nucleosome positions across the population of cells. Physically, a nucleosome consists of 147bp of DNA wrapped around histone proteins, so, within a single cell, nucleosomes must be spaced by at least 147bp. As a result, each cell can contribute at most one nucleosome center within a window of width 147bp or less. Thus, the relative magnitudes of the entries of $\vec{\beta}$ within such a window reflect only the distribution of nucleosome positions across cells, not the arrangement of multiple nucleosomes within any individual cell.

Choosing $p = 0$ yields a measure of relative concentration at each base pair in the chromosome. However, choosing $p > 0$ is generally preferred to account for biological variation in nucleosome positions. These estimands come with a useful baseline. Assuming a uniform local distribution of nucleosome positions across cells in the population would imply $C_{p,l}(k) = \frac{(2p+1)}{(2l+1)}$. Deviations from this baseline provide a normalized measure of local concentration. We present strategies for the detection of local nucleosome concentrations based on these estimands in Section 1.3.4.

The second family of estimands provides summaries of small clusters of nucleosome positions. By definition, these estimands rely on a procedure to identify local clusters, such as Parzen window filtering, applied to the estimated vector of $\vec{\beta}$ coefficients. Because of this, these estimands may inherit issues from this clustering procedures; however, they are useful for comparative analysis and can capture interesting patterns. We define the estimand $\vec{\kappa}$ to be the cluster centers obtained by running the selected clustering method on $\vec{\beta}$. $\vec{\kappa}$ is a cluster-level estimand itself, but it is primarily of interest as a means to obtain

summaries of $\vec{\beta}$ within individual clusters. We consider measures of structure, localization, and sparsity within the cluster, defined as the normalized entropy, mean absolute deviation, and quantiles of the entries of $\vec{\beta}$, taken as an unnormalized discrete distribution over the base pairs in the cluster. Formally, considering cluster $\vec{\beta}_{[i:j]}$ and defining $p_{[i,j]}(k) = \beta_k / \sum_{m=i}^{j} \beta_m$, our localization, structure, and sparsity measures are defined as

$$L_{i,j} = 1 - \frac{4}{j-i+1} \sum_{k=i}^{j} p_{[i,j]}(k)|k - m_{i,j}|, \; m_{i,j} = \sum_{k=i}^{j} k p_{[i,j]}(k) \tag{1.9}$$

$$S_{i,j} = 1 + \frac{1}{\log(j-i+1)} \sum_{k=i}^{j} \log p_{[i,j]}(k) \tag{1.10}$$

$$R_{i,j,q} = 1 - \frac{n_{i,j,q} - 1}{q(j-i+1)}, \; n_{i,j,q} = \min \left(n : \sum_{k=i}^{i+n} \tilde{p}_{[i,j]}(k) \right) , \tag{1.11}$$

respectively, where $\tilde{p}_{[i,j]}(k)$ is $p_{[i,j]}(k)$ sorted in descending order. All measures are normalized so $L_{i,j} = S_{i,j} = R_{i,j,q} = 0$ if $\vec{\beta}_{[i:j]}$ is constant and $L_{i,j} = S_{i,j} = R_{i,j,q} = 1$ if $\vec{\beta}_{[i:j]}$ contains only one non-zero entry.

Using the methods described in Section 1.3, we can obtain draws from the posterior distribution of all of these estimands. This allows us to cleanly separate the modeling of the measurement process and broad properties of nucleosome positioning from the features of interest.

1.3 Inference, estimation, and computation

To extract useful inferences from the model of Section 1.2, we must address three sets of unknown quantities: the digestion template $\vec{t}$, the segmentation of each chromosome s, and the parameters and latent variables of the positioning model, $\vec{\beta}$, $\vec{\mu}$, and $\vec{\sigma}^2$. The parameter $\vec{\beta}$ and quantities derived from it are of the greatest scientific interest, as they correspond directly to the chromatin structure. However, before inferring $\vec{\beta}$, we address

$\vec{t}$ and s.

To estimate the template t from paired-end sequencing data, we develop a non-parametric method, in Section 1.3.1. We develop a simple algorithm to segment each chromosome into non-overlapping segments with useful biological and statistical properties, in Section 1.3.2. Using the estimated template t and segmentation s, we turn to model-based inference for $(\vec{\beta}, \vec{\mu}, \vec{\sigma}^2)$. We build a parallel MCMC algorithm that can efficiently sample from the joint posterior of these parameters, in Section 1.3.3. By combining the conditional independence structure of our model with distributed computation, we are able to handle datasets where $\vec{\beta}$ contains millions of entries.

An approximate EM algorithm is also provided in the online supplement as an optional initialization step for this sampling. The EM approach provides a computationally-efficient way to obtain rough estimates of these parameters, but the joint posterior distribution of $(\vec{\beta}, \vec{\mu}, \vec{\sigma}^2)$ has a complex multimodal structure that EM is ill-equipped to address. Implementation details are given in the online supplement.

Finally, we calibrate the frequentist operating characteristics of our Bayesian estimators using a permutation null hypothesis, detailed in Section 1.3.4. This ensures that our conclusions are valid both as Bayesian posterior probability assessments and under frequentist criteria. We focus on controlling the false discovery rate (FDR) for the detection of local structure in the distribution of nucleosome positions.

1.3.1 Template estimation

Recall from Section 1.2.1 that we model the length of each fragment as $l_j = l_o + e_{1,j} + e_{2,j}$. We assume that $e_{1,j}$ and $e_{2,j}$ (the digestion errors) are independent and identically distributed, and l_o is fixed at 147, which is the known length of DNA wrapped around a single nucleosome. We show how $e_{1,j}$ and $e_{2,j}$ relate to each fragment in Figure 1.4. Along the genome, the distributions of digestion errors at the ends of each fragment are mirror

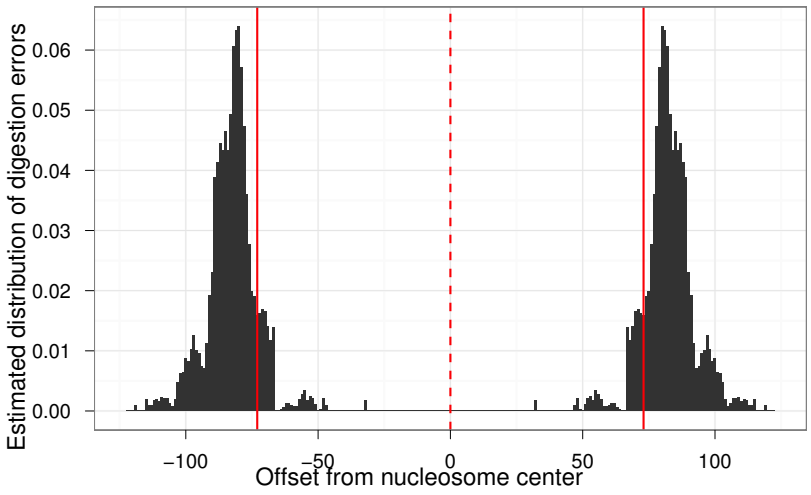

Figure 1.4: Estimated digestion error distributions vs. offset from nucleosome center; vertical lines at $\pm l_o/2$ (solid) and nucleosome center (dashed).

images of each other, so positive values imply that some DNA not bound to a nucleosome is under-digested, so $l_j > l_o$.

To setup our estimation problem, we define two probability distributions,

$$p(i) = Pr\left(l_j = i\right) \tag{1.12}$$

$$q(i) = Pr\left(e_{1,j} = i\right). \tag{1.13}$$

Physically, $l_j \geq 0$, which implies $e_{1,j}, e_{2,j} \geq -\lfloor \frac{l_o}{2} \rfloor$. Analogously, if the longest observed fragment length is l_{max}, we have $Pr\left(l_j > l_{max}\right) = 0$. We require $l_j \leq l_{max}$, which imples $e_{1,j}, e_{2,j} \leq l_{max} - l_o + \lfloor \frac{l_o}{2} \rfloor$. Thus, we can write

$$p(i) = \sum_{k=-\lfloor \frac{l_o}{2} \rfloor}^{l_{max}-l_o+\lfloor \frac{l_o}{2} \rfloor} q(k)q(i-l_o-k). \tag{1.14}$$

The resulting log-likelihood for the observed fragment lengths is

$$\ell(q) = \sum_{j=1}^{M} \log p(l_j) \; . \tag{1.15}$$

We maximize this numerically, using a multivariate logit transformation on the values of $q(k)$ to avoid bounded optimization. Using the L-BFGS-B algorithm (Zhu et al., 1997) on a laptop with a Core i5 processor and 8GB of RAM, this maximization requires approximately 40 seconds for a typical experiment. This computation scales only with the number of unique fragment lengths observed, so it cannot become bottleneck for this method.

We obtain the template distribution t from q via a convolution sum and linear transformation. Recall from Section 1.2.1 that t is the distribution of $\frac{e_1 - e_2}{2}$, restricted to the integers via random rounding. We first obtain the distribution of $e_1 - e_2$ via

$$u(i) = P(e_1 - e_2 = i) = \sum_{k=-\lfloor \frac{l_0}{2} \rfloor}^{l_{max} - l_0 + \lfloor \frac{l_0}{2} \rfloor} q(k) q(k-i) \; . \tag{1.16}$$

We finally transform the distribution $u(i)$ to the desired template $t(i)$ by accounting for random rounding, as

$$t(k) = \frac{1}{2} u(2k-1) + u(2k) + \frac{1}{2} u(2k+1) \; . \tag{1.17}$$

Thus, the estimated template accurately reflects both the variation due to enzymatic digestion and the details of our preprocessing. We use this estimated template to build the design matrix X in the observation model, as discussed in Section 1.2, and for the simulation study discussed in Section 1.4.1.

1.3.2 Segmentation algorithm

We estimate the segmentation function $s : \{1 \dots N\} \to \{1 \dots S\}$ by leveraging the biological structure of each chromosome. We begin by enumerating all open reading frames (ORFs) and intergenic regions on a given chromosome. Merging overlapping ORFs into single segments yields a starting set of contiguous, non-overlapping segments. Many of these segments are too short to provide useful local regularization. To increase the segmentation's utility, we merge neighboring segments until all segments exceed a minimal length (800bp for the purposes of the analysis in Section 1.4).

We iteratively merge the most similar short segments until the resulting segmentation fulfills the given minimum length constraint. We measure similarity using the coverage within each segment, defined as

$$c_i = \frac{1}{n_i} \sum_{k\,:\,s(k)=i} y_k \,, \tag{1.18}$$

where n_i is the length of segment i. Algorithm 1 provides pseudocode for this procedure.

Given Minimum segment length M; initial segmentation;
Calculate $\{n_i\}$ and $\{c_i\}$;
while $\min_i n_i < M$**:**
  Clear minimal difference in coverages d_m and index i_m;
  `/* Find the best merge among short segments */`
  for $i \,:\, n_i < M$**:**
    Compute $d_i = \min(|c_i - c_{i-1}|, |c_i - c_{i+1}|)$;
    if $d_i < d_m$**:**
      Update $d_m = d_i$ and $i_m = i$;
  `/* Execute best merge */`
  Merge segment i_m with neighbor having nearest coverage;
  Update $\{n_i\}$ and $\{c_i\}$;
fintq
return *Segmentation s*

Algorithm 1: Segmentation algorithm

At the conclusion of Algorithm 1, we obtain a segmentation for which each segment has enough observations to provide useful local regularization. The boundaries of each segment also align with biologically-meaningful features, as every step in the above procedure maintains segment boundaries as a subset of ORF boundaries. This estimated segmentation is fixed and used in all subsequent inference.

1.3.3 Posterior sampler

The MCMC sampler consists of two alternating updates. At each iteration r, our algorithm

1. Draws $(\vec{\mu}^{(r)}, \vec{\sigma}^{2\,(r)})|\vec{\beta}^{(r-1)}$ directly, then
2. Updates $\vec{\beta}^{(r)}|(\vec{\mu}^{(r)}, \vec{\sigma}^{2\,(r)})$ via a distributed HMC step.

The first update is straightforward as we can directly sample from the conditional posterior of $(\vec{\mu}^{(t)}, \vec{\sigma}^{2\,(t)})$. This is a standard conjugate normal update, given the log-normal hierarchical structure, and operates independently across segments. We give details in the online supplement.

The second update is computationally challenging. The chromosomes of S. cerevisiae range in length from 230,218 to 1,531,933 base pairs, so the $\vec{\beta}$ vectors are very high-dimensional. In some of the experiments discussed in Section 1.4, we work with simulated chromosomes with over 3.85 million base pairs. The conditional posterior of $\vec{\beta}^{(t)}|(\vec{\mu}^{(t)}, \vec{\sigma}^{2\,(t)})$ is not part of any standard family, so we turn to Hamiltonian Monte Carlo (HMC). The dimensionality of $\vec{\beta}$ makes a single HMC update for the entire vector both computationally infeasible and numerically unstable. To enable fast, statistically-efficient computation, we take advantage of the conditional independence structure of this conditional posterior.

Subvectors of $\vec{\beta}$ separated by at least $2w$ entries are conditionally independent given $(\vec{\mu}^{(t)}, \vec{\sigma}^{2\,(t)})$ and the entries of $\vec{\beta}$ between them. Consider the subvectors $\vec{\beta}_{[j_1:j_2]}$ and $\vec{\beta}_{[k_1:k_2]}$,

with $j_1 < j_2 < k_1 < k_2$. The elements of $\vec{\beta}_{[j_1:j_2]}$ affect only $\vec{\lambda}_{[j_1-w:j_2+w]}$, and the elements of $\vec{\beta}_{[k_1:k_2]}$ affect only $\vec{\lambda}_{[k_1-w:k_2+w]}$. Hence, if $k_1 > j_2 + 2w$, then $\vec{\beta}_{[j_1:j_2]}$ and $\vec{\beta}_{[k_1:k_2]}$ are conditionally independent given $\vec{\mu}$ and $\vec{\sigma}^2$.

We take advantage of this conditional independence to construct a distributed set of HMC updates. We first fix the length of each subvector that will be updated via a single HMC step to $B > 4w$. Next, consider two partitions of $\vec{\beta}$ into subvectors. The first starts at the beginning of $\vec{\beta}$ and proceeds forward with subvectors of length at most B separated by $2w$, yielding

$$\vec{\beta}_{[1:B]}, \vec{\beta}_{[B+2w+1:2B+2w]}, \ldots, \vec{\beta}_{[n_b(B+2w)+1:N]} \cdot$$

The second begins at the $B/2$th entry of $\vec{\beta}$ and again proceeds forward in subvectors of length at most B, as

$$\vec{\beta}_{[B/2+1:3B/2]}, \vec{\beta}_{[3B/2+2w+1:5B/2+2w]}, \ldots, \vec{\beta}_{[n_b(B+2w)B/2+1:N]} \cdot$$

Within each partition, the subvectors are conditionally independent, and, in combination, these partitions include all entries of $\vec{\beta}$.

Within each iteration of our sampler, we cycle through each of these partitions, updating each subvector of $\vec{\beta}$ with one HMC step. As each subvector within each parition is conditionally independent, we can execute all HMC steps in parallel for each partition. This allows us to distribute the computational burden over hundreds of cores, providing fast scalable inference. Each of these distributed HMC steps is, on its own, relatively standard. However, they are much faster than expected, as the log-conditional posterior's value and gradient can both be computed via a convolution, lowering the computational cost per core to $O(B \log B)$ with the fast Fourier transform. In particular, all matrix-vector products involving the X matrix can be computed as convolutions with the template vector $\vec{t}$

instead, reducing the complexity of such products from $O(B^2)$ to $O(B \log B)$. Details of distributed algorithm, computational infrastructure, and tuning of the HMC are given in the online supplement. A Python implementation of the sampler is available on GitHub, `www.github.com/awblocker/cplate`.

1.3.4 Detection and calibration

Recall from Section 1.2.3 that we quantify local concentrations of nucleosomes using the estimand $C_{p,l}(k)$, which defines local concentrations as small regions of the chromosome that contain a density of nucleosomes greater than that we would expect under a uniform distribution of nucleosomes across cells in our population, p/l.

We can estimate $P(C_{p,l}(k) > (2p+1)/(2l+1) \mid \vec{y})$ for each base pair k using the MCMC sampler described in Section 1.3. However, we require greater security in our detection results than the Bayesian approach alone can provide. To quantify the operating characteristics of our procedure and provide frequentist guarantees on its performance, we turn to a permutation null.

Our null hypothesis is that $\vec{y}$ consists of a set of multinomial draws. Under this null, the entries of $\vec{y}$ within each segment i are drawn from a multinomial distribution with equal probability assigned to each base pair within the segment and $n = \sum_{k:s(k)=i} y_k$. This null hypothesis rests on the same idea as Fisher's exact test: we condition on the marginal distribution of the data and consider all independent permutations of the observations. We approximate this null distribution by repeatedly randomly permuting the observed reads within each segment.

We then run our MCMC sampler on each such draw from the null, using the template and segmentation estimated from the observed data. From the sampler's output, we obtain an estimate of the distribution of $P(C_{p,l}(k) > (2p+1)/(2l+1) | \vec{y})$ over positions k under the null. We compare this to the distribution of posterior probabilities for the ob-

served data and set a detection threshold to control the FDR using the method of Storey and Tibshirani (2003). For example, with the datasets analyzed in Section 1.4, we have typically found that a threshold of approximately 0.8 on $P(C_{p,l}(k) > (2p+1)/(2l+1)|\vec{y})$ yields a FDR of 5% or less. This approach provides a secure detection procedure with both Bayesian and frequentist interpretations.

1.4 Results

We demonstrate the proposed methods on real and simulated data. High-throughput sequencing data were collected on S. cerevisiae cell populations growing in a high-phosphate medium. The data consist of two lanes of sequencing, referred to as technical replicates, on each of two separate samples with different enzymatic digestion, referred to as biological replicates. Analyses with the proposed methods are highly reproducible, as we show in Section 1.4.4, and provide new insights on the fine-grained structure of nucleosome positioning. The biological relevance of these substantive findings is detailed elsewhere (Zhou et al., 2012).

The simulation studies aim to demonstrate the utility of the estimands introduced in Section 1.2.3 used in combination with the proposed deconvolution approach to inference, as well as the scalability of our methods. In Section 1.4.1, we describe the design of the simulation studies. We simulate high-throughput sequencing data with different coverage, on genes with primary and alternative nucleosome positions, with different degrees of variation throughout the population. In Section 1.4.2, we discuss the efficiency and scalability of inference via parallel Hamiltonian Markov Chain Monte Carlo sampling. In Section 1.4.3, we compare the performance of the proposed method to that of a Parzen-window estimator followed by greedy search (the standard in the field; Albert et al., 2007a; Shivaswamy et al., 2008; Tsankov et al., 2010; Tirosh, 2012) for estimating measures of

structure, quantifying power and error in estimating the locations of clusters of nucleosome positions. In Section 1.4.3, we assess the performance of the proposed method for estimating measures of local concentration, quantifying power and error in estimating the locations of distinct primary and alternative positions. In both Sections 1.4.3 and 1.4.3, we use design-based analysis of variance (ANOVA) to quantify the relative contributions to estimation errors of coverage, distance between primary and alternative positions and their relative frequencies across the cell population. We also use logistic regression to analyze the sensitivity of power to the three experimental factors we consider. In Section 1.4.4, we assess the reproducibility of our inferences for cluster-level summaries of nucleosome positioning (Section 1.4.4) and the locations of local concentrations (Section 1.4.4). We also compare the reproducibility of our cluster-level inferences to those obtained from Parzen-window methods and read-based estimators of the cluster-level estimands.

To perform inference throughout this section, we set $\mu_0 = 0$, $\tau_0 = 1/10$, $\alpha_0 = 7$, and $\gamma_0 = 10$. These values were chosen to be weakly-informative on the basis of prior biological information. These values of α_0 and γ_0 imply that there is a 99% prior probability that 0.2–13% of base pairs have β_k greater than or equal to 10 times their median. We found little sensitivity of our inferences to these choices of parameter values, using data from chromosome I. For instance, sweeping τ_0 over two orders of magnitude (0.01–1) showed little effect on inferences, as did similar changes to (α_0, γ_0).

1.4.1 Experimental design

To assess the performance of the proposed methods, we generated artificial chromosomes using the classical principles of experimental design. These artificial chromosomes consist of a series of genes, each containing a set of nucleosome positions. We fix the length of each gene to 3501bp, consisting of a 1000bp promoter region, a 2500bp coding region, and a 1bp transcription start site (TSS).

We designed a simulation with three factors, varied at the gene level: coverage (the expected number of reads per gene), the spacing between primary nucleosome positions and alternative positions (which we refer to as offset), and the relative magnitudes of primary and alternative positions. Coverage had 10 levels, spanning the 5th to 95th percentile observed gene-level coverages in increments of 10%. Alternative position spacing had 10 levels, spanning from 0bp (no alternative positions) to 45bp in increments of 5bp. Alternative position magnitude had 11 levels, spanning from 0 (no alternative positions) to 1 (alternative positions of the same magnitude as primary positions) in increments of 0.1. Thus, the effective magnitude of the primary position relative to the alternative positions ranged from 1 to $\frac{1}{3}$. We used a full factorial design on these three factors, yielding 1100 distinct treatments for each of 10 simulated chromosomes. We then constructed a realistic distribution of nucleosome positions within each artificial gene. Using one of our high-phosphate data sets, we first identified clusters of nucleosome positions using the standard Parzen window method. We indexed these clusters by their ordering within each actual gene, considering 1000bp before TSS to the end of each ORF, and computed the proportion of reads within the ORF observed within each such cluster. Finally, we averaged over the positions and proportions of these clusters by their order from their TSS, obtaining the average offset from the TSS and relative occupancy of the first, second, third, etc. clusters before and after the TSS. Figure 1.5 provides an illustration of both coefficients, β_k, and read counts y_k, for one gene.

To generate our artificial dataset, we followed a modified version of the generative process outlined in Section 1.2. For each gene, we first drew coefficients for its subset of $\vec{\beta}$ from an upper-truncated log-normal distributed with parameters estimated from those regions with similar coverage. These are our "background" positions. Then, we set the entries of $\vec{\beta}$ corresponding to the gene's primary and alternative positions deterministically.

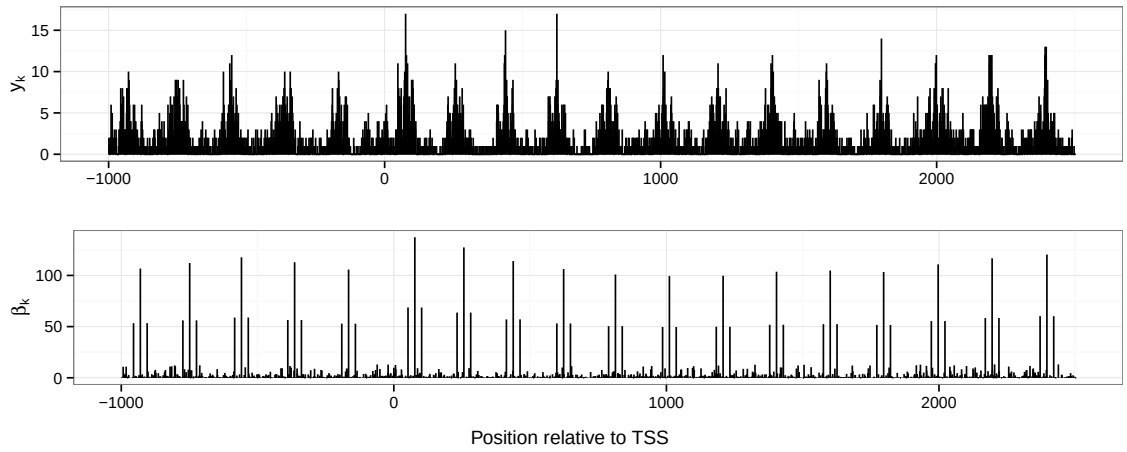

Figure 1.5: Illustration of one simulated gene: 0.55 quantile of coverage, with alternative position magnitude of 0.5, and alternative positions at $\pm$25bp from each primary position. Read counts y_k (top panel), and coefficients β_k (bottom panel).

The sum of the coefficients for these positions was fixed to the remaining total occupancy of the gene, less the sum of the background positions. Their relative magnitudes were determined by the design described above, with two alternative positions placed symmetrically around each primary position at the designated spacings. Thus, for a given level of coverage, the expected number of reads within each cluster was fixed, but its distribution across primary and alternative positions varies.

We convolved these $\vec{\beta}$ vectors with the template estimated from the experimental data to obtain vectors of expected read counts $\vec{\lambda}$. Finally, we generated $\vec{y} \sim$ iid Poisson$(\vec{\lambda})$ to obtain simulated read counts. This entire procedure was repeated for each replicate, yielding 10 artificial chromosomes of length 3,851,100bp each.

These simulations are inspired by our generative model, but they do not follow it's structure to the letter. The distribution of background coefficients and the effects of digestion agree between our model and our simulations. However, we introduce much more structure into the locations of nucleosome concentration via the deterministic placement of primary and alternative positions. This simultaneously provides a stringent test of our

methods and decreases the amount of residual variation across our experimental replicates. As a "sanity check" on this design, simulated read counts were shown side-by-side with matched actual read counts to experienced biologists in this field. They could not reliably distinguish between the simulated and actual data. We provide further algorithmic details for this procedure and supporting figures in the online supplement.

1.4.2 Parallel HMC performance

The parallel Hamiltonian Monte Carlo sampler performed well on both real and simulated datasets, based on standard MCMC diagnostics. For the actual and simulated datasets used in Section 1.4, we ran 2,000 iterations, discarding the first 200 as burn-in. This yielded 1,800 draws for $\vec{\beta}$, $\vec{\mu}$, and $\vec{\sigma}^2$. The mean effective sample size for the elements of $\vec{\theta} = \log \vec{\beta}$ in the real dataset was 1573, with 99% of the coefficients having effective sample sizes between 304 and 2057. For the simulated dataset, the mean was 1675 with 99% between 520 and 2011. Gelman-Rubin diagnostics based a set of MCMC runs with dispersed initializations on the smallest chromosome (I) showed multivariate potential scale reductions of 1.05 or less.

Our sampler proved extremely scalable. Using 144 cores on the Harvard Odyssey cluster and setting $B = 2000$, each simulated chromosome required 1.83 seconds per iteration for a total runtime of approximately 1 hour. The smallest S. cerevisiae chromosome (I) required 0.136 seconds per iteration, while the largest (IV) required 0.699 second per iteration, yielding total runtimes of 4.5 and 23.3 minutes, respectively. Running the entire S. cerevisiae genome required approximately 3.24 hours. The sampler was also run on an Amazon EC2 cluster with 512 cores, processing the same genome in under an hour.

1.4.3 Power analysis

Using simulated chromosomes, we compare the performance of the proposed method to that of a Parzen-window estimator for estimating locations of clusters of nucleosome positions, and assess the performance of the proposed methods for detecting and estimating the locations of both primary and alternative positions. For both analyses, we use ANOVA to quantify the relative contributions to estimation errors of coverage, distance between primary and secondary positions and their relative frequencies across the cell population. We complement these with logistic regression to analyze the sensitivity of power to these three experimental factors.

The ground truth consists of the primary and alternative positions generated in Section 1.4.1, along with their coefficients. Recall that the output of the calibrated detection procedure, detailed in Section 1.3.4, is a series of positions where high local concentrations have been detected, and the output of the cluster-based estimands is a series of cluster centers. To assess performance in estimating cluster positions, we match inferred cluster centers to ground truth cluster centers. Similarly, to assess performance of local concentration measures, we match detected local concentrations all ground truth positions, primary and alternative. Finding the nearest estimated position for each ground truth position yields measures of power, as we can measure the distance from each true position to the nearest inferred one. Large distances imply low sensitivity, and vice versa. Conversely, finding the nearest ground truth position for each inferred position yields measures of accuracy. If inferred positions are far from the true ones, we would consider the results unreliable.

The analyses below are based on summaries of these matched distances; we compute mean and median absolute errors, and we tabulate the proportion of true positions matched to an inferred positions within a fixed number of base pairs. The first set of

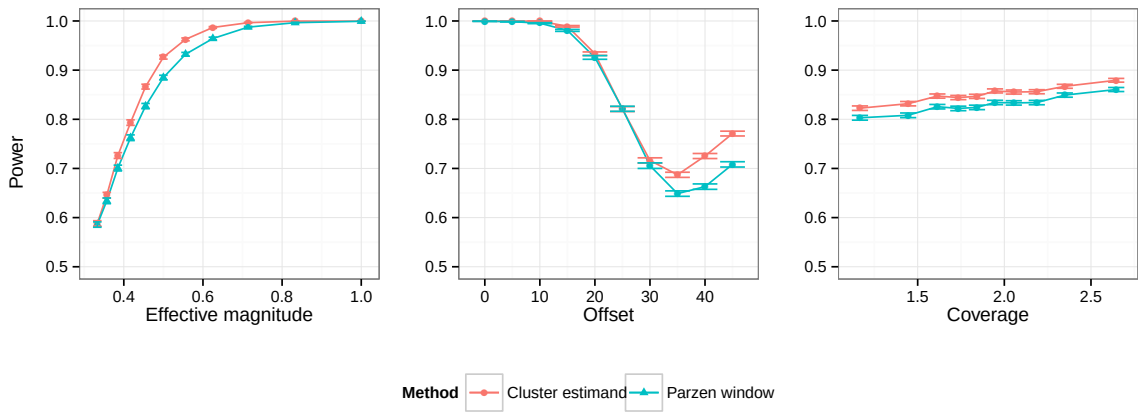

Figure 1.6: Power vs. effective magnitude (left), alternative position offset (center), and coverage (right) for Parzen window and cluster estimand methods

quantities summarize distributions of errors in estimated positions, while the second is directly interpretable as a measure of power.

Clusters

Detection of a cluster was defined as a best-match distance of less than 5bp between the inferred and true cluster center (κ_m). Figure 1.6 summarizes our key findings, showing the relative power of each method against the effective magnitude of the primary position, the offset of the alternative positions, and gene-level coverage. Tables 1.1 provides the results of a design-based ANOVA of the mean absolute errors of estimated cluster locations, by gene, and Table 1.2 provides the results of a logistic regression of power on the design factors.

For estimating cluster locations, the proposed method dominates the Parzen-window estimator both in terms of power, with average difference of 2.1%, and mean absolute error (not shown) across all conditions. Power ranges from approximately 12% to 100% over all factor combinations in our experiments for both methods, while mean absolute position errors range from approximately 0.1 to 60bp. Our method provided an average

Table 1.1: Analysis of variance of absolute errors in cluster centers for cluster estimand and Parzen window methods. All factors and interactions were statistically significant with $p < 0.0001$.

		Cluster Estimand		Parzen Window	
	Df	Sum Sq	F value	Sum Sq	F value
Coverage	9	1928.94	198.73	753.24	79.04
Offset	9	19422.58	2001.01	10789.06	1132.10
Magnitude	9	10764.55	1109.02	6768.51	710.22
Coverage:Offset	81	275.41	3.15	264.68	3.09
Coverage:Magnitude	81	175.19	2.01	142.80	1.66
Offset:Magnitude	72	23535.40	303.09	19172.33	251.47
Coverage:Offset:Magnitude	648	950.25	1.36	947.80	1.38
Residuals	10090	10881.93		10684.31	

power of 85%, while the Parzen window method's average power was 83%. Power shows a strong dependence on the local distribution of nucleosome positions; the accuracy in identifying the cluster centers of primary positions is reduced by the presence of stable alternative positions. The spacing between primary and alternative positions also affects power substantially, with power diminishing by approximately 30% as the offset increase from 0 to 35bp. Power increases slightly for both methods at offsets of 40 and 45bp. The relative performance of the proposed method is largest for offsets over 30bp, with a difference in power of 7% at 45bp. Power shows little marginal dependence upon local coverage with only a 6% change in power over the range of coverages for both methods.

The ANOVA and logistic regression analyses support these observations and provide further insights into the role of interactions between the design factors. ANOVA results in Table 1.1 indicate that alternative position offset, effective magnitude, and their interaction account for the vast majority of variation in absolute position errors (approximately 75% of total variation and 94% of explained variation) for both methods. Logistic regression results in Table 1.2 suggest that the power of the proposed method and of the Parzen window estimator respond similarly to the experimental factors. The marginal effects of

Table 1.2: Logistic regression of power on design factors as continuous variables cluster estimand and Parzen Window method. Are regressors are normalized to have range $[0, 1]$.

	Cluster Estimand		Parzen Window	
	Estimate	z value	Estimate	z value
(Intercept)	3.3461	54.15	3.4660	56.74
Coverage	0.6069	5.20	0.7498	6.54
Offset	-5.3115	-58.00	-5.4596	-60.92
Effective Magnitude	2.7211	10.76	2.3296	10.82
Coverage·Offset	-0.4217	-2.48	-0.6071	-3.66
Coverage·Effective Magnitude	1.7927	3.20	0.9037	2.02
Offset·Effective Magnitude	8.9842	22.27	6.9932	21.15
Coverage·Offset·Effective Magnitude	2.1476	2.52	1.8962	2.84

offset and effective magnitude are strongly negative and positive, respectively, but the offset-effective magnitude interaction effect is overwhelmingly large and positive. Coverage has a weak marginal effect on power, but it enters more strongly in the interaction with effective magnitude and in the three-way interaction.

Taken together, these results demonstrate that the proposed method offers improved performance relative to the standard method in the field, for estimating cluster locations. However, the proposed method offers the greatest benefits for exploring local concentration in the distribution of nucleosome positions, an estimand that the Parzen-window estimator cannot reliably infer.

Local concentrations

We next examine the ability of the proposed method to detect local concentrations in the distribution of nucleosome positions, a quantification of small-scale structure. We focus on detecting small regions of excess local concentration using the $C_{p,l}(k)$ estimand, defined in Equation 1.8. For this analysis, we fix $l = 73$ and $p = 3$, and used the calibrated detection procedure described in Section 1.3.4 with a maximum FDR of 5%.[1] For primary

[1]We reduce any contiguous sequences of detections to their mean position for interpretability. This is conservative in terms of FDR control and provides a more stringent test of the proposed methodology.

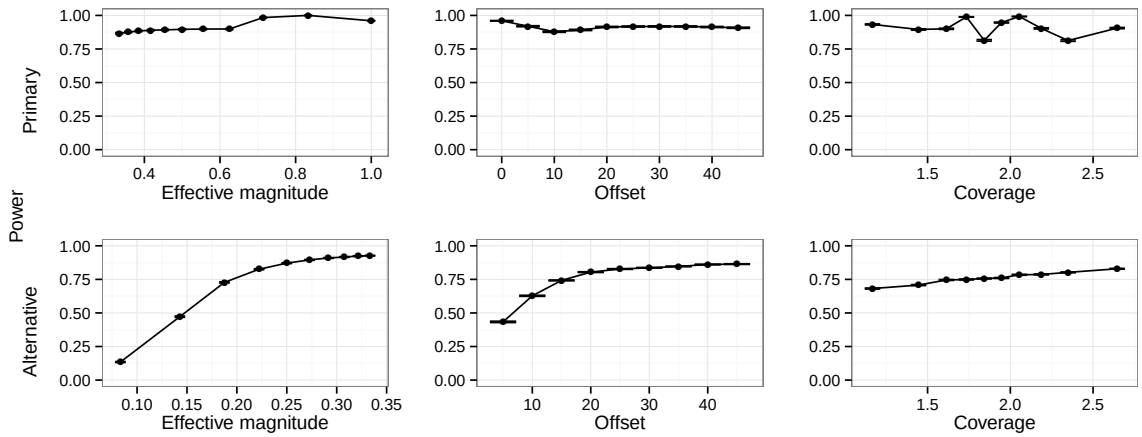

Figure 1.7: Power vs. effective magnitude (left), alternative position offset (center), and coverage (right) for detection of primary and alternative positions $\pm$3bp

positions, we declare a successful detection if the best-match distance is less than 5bp between a detected position and the true primary position. For alternative positions, we declare a successful detection if the best-match distance between a detected position and the true alternative position is less than 1/2 of the alternative position's distance from its primary position. Figure 1.7 summarizes our results for primary and alternative positions, showing the power of the proposed method against the effective magnitude of the primary position, the offset of the alternative positions, and gene-level coverage. Tables 1.3 provides the results of a design-based ANOVA of the mean absolute errors of estimated cluster locations, by gene, and Table 1.4 provides the results of a logistic regression of power on the design factors.

Power ranges from approximately 64% to 100% for primary positions and from approximately 2% to 100% for alternative positions across all combinations of factors, while mean absolute position errors range from 0.389 to 6.61bp and from 2.17 to 41.8bp, respectively. The sensitivity of power and absolute position errors to the experimental factors differs between primary and alternative positions.

Table 1.3: Analysis of variance of absolute position errors for the detection of primary and alternative positions using local concentration estimands. * indicates that a factor was statistically significant with $p < 0.0001$. Remaining factors had p-values larger than 0.95.

	Primary Positions			Alternative Positions		
	Df	Sum Sq	F value	Df	Sum Sq	F value
Coverage	9	1453	63*	9	11136	1826*
Offset	9	5838	253.20*	8	34615	6385*
Magnitude	9	116497	5053*	9	74826	12267*
Coverage:Offset	81	148	0.7	72	5331	110*
Coverage:Magnitude	81	1292	6*	81	4532	83*
Offset:Magnitude	72	2964	16*	72	154557	3167*
Coverage:Offset:Magnitude	648	1044	0.6	648	3968.16	9.04*
Residuals	10090	25848		8100	5490	

For primary positions, the power increases as the effective magnitude of the primary position increases and decreases as the offset to the alternative position increases. The ANOVA results in Table 1.3 suggest that the majority of variation in absolute estimation errors for primary positions (75% of total and 90% of explained) is driven by the relative magnitude of primary and alternative positions. Coverage plays a minor role in the variation of these errors, even when including all of its interactions. The logistic regression results in Table 1.4 tell a similar story, with effective magnitude of the primary position and its interaction with the offset to the alternative position showing a strong positive effect on power. Other effects are considerably smaller.

For alternative positions, the power increases as effective magnitude of the primary position, the offset to the alternative position, and the coverage increase. The ANOVA results in Table 1.3 show that the majority of the variation in absolute estimation errors for alternative positions is accounted for by the offset-magnitude interaction (52% of total, 53% of explained), with the marginal contributions of magnitude, offset, and coverage accounting for most of the remaining variation (41% of total, 42% of explained).

The logistic regression results in Table 1.4 support these findings and shed more light on

Table 1.4: Logistic regression of power on design factors as continuous variables for primary and alternative positions. Are regressors are normalized to have range $[0, 1]$.

	Primary Positions		Alternative Positions	
	Estimate	z value	Estimate	z value
(Intercept)	1.7018	2.13	-1.6631	-44.64
Coverage	-0.0456	2.33	0.4591	7.12
Offset	0.4778	0.82	-1.5089	-20.65
Effective Magnitude	2.0866	14.07	2.1086	38.37
Coverage·Offset	-0.5585	-1.42	1.0953	8.64
Coverage·Effective Magnitude	-0.7448	-5.31	-0.3608	-3.75
Offset·Effective Magnitude	0.9623	2.62	7.8374	55.50
Coverage·Offset·Effective Magnitude	-0.3217	-0.95	8.7109	31.84

the drivers of power for primary and alternative positions. The marginal effect of effective magnitude of the primary position is similar for primary and alternative positions, but the offset-effective magnitude and three-way interactions are far stronger for alternative positions than they are for primary positions. Coverage also has a pronounced effect on power for alternative positions, both marginally and through the interaction terms.

Taken together, these results demonstrate that the proposed method can detect local concentrations in the distribution of nucleosome positions across a broad range of realistic conditions. We can reliably detect and estimate the locations primary positions with average power over 90% and average absolute position errors of only 2.1bp. Although alternative positions are more difficult to detect, the proposed method provides reliable inferences about their positions as well, yielding an average power of 76% and mean absolute position errors of 6.0bp. We discuss the implications of these capabilities for biological analyses in Section 1.5.

1.4.4 Reproducibility analysis

We compared the reproducibility of estimates of cluster-level properties from the proposed method to those from a Parzen-window estimator, and assessed the reproducibil-

ity of estimated local concentration locations from the proposed method. For this comparison, we used measurements from two distinct samples (biological replicates, indexed by i), each of which was sequenced twice (technical replicates, indexed by j). This design yields four data sets, H_{ij} for $i, j = 1, 2$, which allow for two comparisons within biological replicates (i.e., H_{11} versus H_{12}, and H_{21} versus H_{22}), and four comparisons across biological replicates. Biological replicates have different levels of enzymatic digestion, allowing us to directly assess gains in robustness from estimation of the digestion variability template, introduced in Section 1.2.1.

We examine the reproducibility of inferences on cluster-level and local concentration estimands in Sections 1.4.4 and 1.4.4, respectively. For these analyses, we first matched inferred positions within pairs of replicates. We then took the union of all matched positions, within each pair of replicates, as a basis for subsequent analyses; for instance, to estimate the distribution of distances between matched positions, and the correlations of inferred measures associated with each position. The same matching procedure was used for inferences on both cluster-level properties and local concentrations. We present detailed results for each class of estimand below.

Clusters

We assessed the reproducibility of estimated cluster positions and cluster-level summaries from both our method and the standard Parzen-window technique. For the former, all estimates are posterior means of the estimands specified in Equations 1.9–1.11 ($L_{i,j}$, $S_{i,j}$, and $R_{i,j,q}$) using a window of $\pm$73bp around each estimated cluster center. We set $q = 0.9$ for the sparsity estimand. For the latter, we estimated cluster-level properties using the observed read counts $\vec{y}$ directly to obtain estimates of the localization, sparsity, and structure indices described in Section 1.2.3 for the clusters identified by the Parzen-window method. In addition to matching inferred positions between replicates for each method,

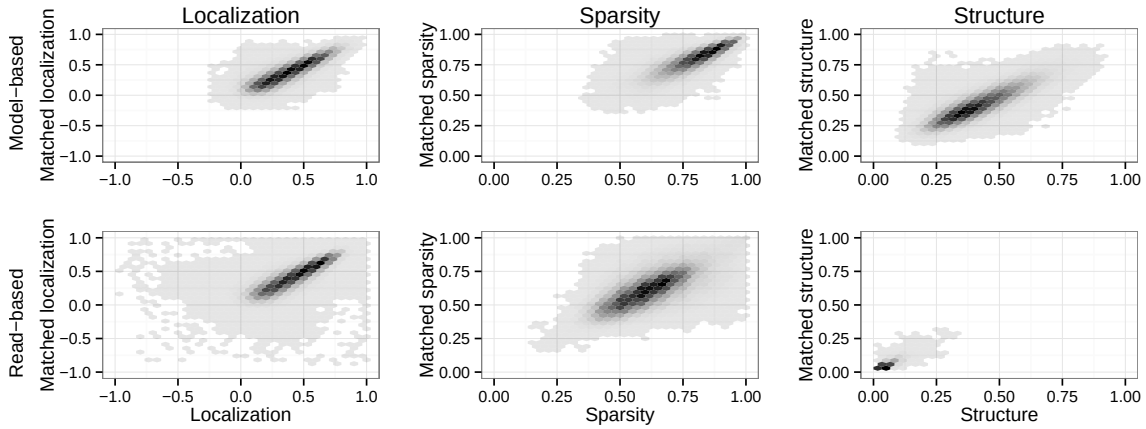

Figure 1.8: Joint distributions of local, structure, and sparsity indices for matched clusters between biological replicates for model-based (top) and Parzen window/read-based (bottom).

as described above, we also matched inferred positions between methods within each replicate to asses the comparability of estimates obtained by the different methods. Our results are summarized in Figure 1.8.

Inferred cluster positions were highly reproducible with a mean best-match distance of 15.72 ± 0.14bp and median best-match distance of 4bp, between biological replicates, and of 14.30 ± 0.2bp and 3bp, respectively, between technical replicates. With the proposed method, 90% of clusters were matched within 44bp across biological replicates, and within 35bp across technical replicates. These results are comparable to the those obtained with a Parzen-window estimator, which achieves mean and median best-match distances of 15.24 ± 0.14bp and 4bp, between biological replicates, and of 13.98 ± 0.19bp and 8bp, respectively, between technical replicates. Inferred cluster positions were also consistent across methods, within each replicate, with mean and median best-match distances of 3.11 ± 0.07bp and 1bp. Across methods, 90% of inferred cluster positions were matched within 2bp and 95% were matched within 3bp.

Cluster-level properties, however, showed significant differences between the model-

based and Parzen-window estimates, both in terms of reproducibility and comparability, as Figure 1.8 shows. The model-based estimator of the localization estimand L showed the greatest reproducibility with an R^2 of 0.765 $\pm$ 0.002 between matched clusters for biological replicates (0.799 $\pm$ 0.002 for technical replicates), performing better than the read-based estimates which had R^2's of 0.713 $\pm$ 0.003 and 0.745 $\pm$ 0.005, respectively. The model-based estimator of the structure estimand S was close behind with R^2's of 0.749 $\pm$ 0.002 and 0.795 $\pm$ 0.002 for biological and technical replicates. However, the read-based estimator of S fared considerably worse with R^2's of only 0.664 $\pm$ 0.003 and 0.698 $\pm$ 0.004, respectively. The model-based estimator of the sparsity estimand R showed the largest gap in reproducibility between model-based and read-based estimators, with R^2's of 0.720 $\pm$ 0.002 and 0.736 $\pm$ 0.002 for the model-based method (between biological and technical replicates) and R^2's of only 0.403 $\pm$ 0.007 and 0.526 $\pm$ 0.005 for the read-based estimator, respectively.

Localization (L) was also the most comparable feature between the model-based and read-based estimators with a Spearman correlation of 0.950$\pm$0.001 within replicates. This can be seen graphically in the leftmost panels of Figure 1.8: the read-based localization index is noisier than the model-based one, but their distributions appear otherwise comparable. The structure index (S) was moderately comparable between the model-based and read-based estimators with a Spearman correlation of 0.784 $\pm$ 0.001. The magnitudes of these estimators are less comparable than the correlation suggests, with the model-based estimator spanning nearly 3 times the range of the read-based one. The sparsity index (R) was barely comparable between estimators, as one would expect from the middle panels of Figure 1.8. Its Spearman correlation was only 0.218 $\pm$ 0.003, and the read-based estimator spanned a far wider range of values than the model-based one. These differences arise because the model-based and read-based estimators are actually estimating different quantities. Read-based estimators are estimating properties of both the experimental er-

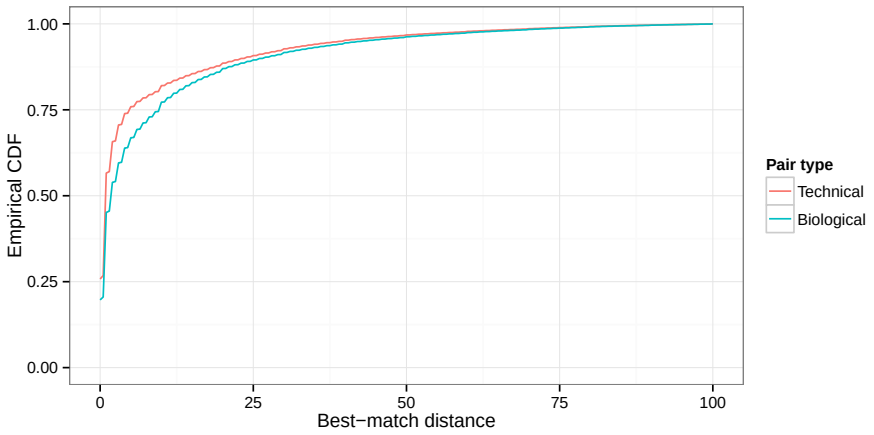

Figure 1.9: Empirical CDFs of best-match distances between detected local concentrations for technical (red) and biological (blue) replicates.

rors and the distribution of positions, whereas the model-based estimators are targeting only the underlying distribution of nucleosome positions.

These results show that the proposed methods provide reproducible inferences about the local structure of nucleosome positions across variation from biological and technical sources, including explicit changes to the degree of enzymatic digestion. They significantly outperform standard Parzen-window and read-based estimators in this regard and provide a richer, more accurate view of the true distribution of nucleosome positions.

Local concentrations

The locations of detected local concentrations (based on $C_{3,147}(k)$) are highly reproducible across both biological and technical replicates. These results are summarized in Figure 1.9, where we compare the distributions of best-match distances between biological and technical replicates.

We observe higher reproducibility between technical replicates than between biological replicates, with median best-match distances of 1bp and 2bp, respectively. For biological replicates, 75% of positions were matched within 10bp, 80% were matched within 15bp,

and 90% were matched within 33bp. For technical replicates, the corresponding quantiles were 6bp, 11bp, and 30bp. These results demonstrate the reliability of the proposed method in analyzing high-throughput sequencing data, and provide confidence that the small-scale details of nucleosome positioning identified by the proposed method represent real biological structure.

1.5 Concluding Remarks

We have presented an approach to modeling and making inferences about the genome-wide distribution of nucleosome positions from paired-end sequencing data. The results presented in Section 1.4 demonstrate the utility of the proposed methods for biological analyses, particularly the reproducibility of its inferences across experimental conditions. Below, we expand on several broader points that have informed the development of these methods, including the lack of utility of single-cell constraints in the analyses of measurements on cell populations, the relationship between estimands of interest and the performance gains stemming from model-based inferences on them, and the role of distributed computing in inference with massive, high-dimensional data.

1.5.1 Modeling

We explicitly choose not to include prior information on nucleosome spacing in this model. Previous work has used the empirically-observed 150-200bp spacing between nucleosomes within individual cells to constrain inferences on nucleosome positions (e.g., see Shivaswamy et al., 2008; Yuan et al., 2005). In the presence of alternative nucleosome positioning and chromatin dynamics, however, constraints on spacing that hold on a single-cell level need not hold after aggregation across a population of cells, which is where measurements are taken. With sequencing coverage on the order of 10–100, only

a tiny fraction of the cells in the population contribute to the observed data within each small region of the genome. The probability of observing even two reads from the same cell within, for instance, a single ORF is minuscule. As a result, single-cell constraints provide few constraints on the range of probable observations in high-throughput sequencing experiments. Thus, the proposed model does not use information on expected separation among nucleosomes along the sequence to constrain the inferred nucleosome positions. Instead, we opt for a simpler hierarchical structure within each segment, modeling locally shared distributions of nucleosome localization.

The proposed method uses information about the fragment lengths between by pairs of reads that is provided by paired-end sequencing technology to infer the effects of enzymatic digestion on the measurements $\vec{y}$. Many studies, however, use single-end sequencing technology, which does not provide fragment lengths. In related work (Zhou et al., 2012), we have developed an approach to estimate the digestion variability template, $\vec{t}$, for single-end data using an alternative source of fragment-length information; Bioanalyzer technology (e.g., Mueller, 2000). The model and inference presented in Sections 1.2 and 1.3 can be adapted to this single-end context with an appropriate modification of the template $\vec{t}$ and of the digestion matrix X.

1.5.2 Estimands

In defining the estimands of biological interest, we aimed at separating properties of the distribution of observed reads, which include the effects of enzymatic digestion, PCR, and sequencing, from the distribution of nucleosome positions, which is the true target of biological investigations. Existing estimators defined directly as functions of the read counts confound these distributions, impairing reproducibility of the analysis and ultimately their utility for scientific exploration. In the model introduced in Section 1.2, the distribution of nucleosome positions corresponds to the $\vec{\beta}$ vector, while the template $\vec{t}$

and the remaining error structure capture other sources of experimental variation. The estimands presented in Section 1.2.3 are functions only of the true underlying $\vec{\beta}$, and are thus unaffected by variation due to the experimental process, at least in principle. Below, we discuss two subtle points on the construction of these estimands.

First, there is a key distinction between cluster-based estimands such as $L_{i,j}$, $S_{i,j}$, and $R_{i,j,q}$ and other summaries of local structure, such as those based on $C_{p,l}(k)$. Cluster-based estimands capture the properties of the distribution of nucleosome positions within small regions identified by a clustering algorithm. These measured depend on the particular definition of "cluster" and on the clustering method used. The sensitivity of these estimands to these choices is problematic, in practice, and fine-grained structure is lost in the reduction of data to clusters. However, this reduction can simplify subsequent interpretation. In contrast, estimands such as the local concentration index $C_{p,l}(k)$ summarize the local structure of the nucleosome position distribution without relying on a clustering criterion. They lead to reproducible analyses and can be relied upon for scientific discovery of small-scale features. We believe that these classes of estimands are most useful in combination, providing complementary views on the distribution of nucleosome positions.

Second, we have found that the magnitude of the performance gains stemming from model-based inferences depends strongly on the estimand of interest. For instance, the proposed method outperforms read-based estimators in terms of power and error when targeting our cluster-level localization estimands L, but the difference in reproducibility is not overwhelming. However, the increase in reproducibility one can expect from the proposed method is substantial when targeting structure and sparsity measures, S and R. This reflects the greater sensitivity of read-based estimators of structure and sparsity to observation noise. In addition, as we have shown in Section 1.4.3 and 1.4.4, the proposed method can provide reproducible inferences about individual local features less than 10bp wide, when inference on properties of the distribution of nucleosome positions at such a

fine resolution has been so far considered infeasible with read-based estimators. More generally, the more sensitive an estimand is to observation noise, the greater the performance gains expected from using the proposed method.

Our results suggest that careful probabilistic modeling of the core sources of experimental variation can enable new types of scientific inferences.

1.5.3 Inference

The use of distributed computing was essential for our method, as it allowed us to sample from the marginal posterior of $\vec{\beta}$ in only minutes per chromosome. We leveraged the conditional independence structure of our model to create an efficient, scalable distributed MCMC sampler. This structure stems from the finite length of the digestion variability template $\vec{t}$. As the template is $2w + 1$ wide, subvectors of $\vec{\beta}$ separated by at least $2w$ base pairs are conditionally independent *a posteriori* given $(\vec{\mu}, \vec{\sigma}^2)$. Thus we can update collections of such subvectors independently across hundreds of processors. The communication costs involved in this procedure are low, as only each subvector (padded by w entries on each end) and the relevant entries of $(\vec{\mu}, \vec{\sigma}^2)$ are needed for each update.

To update each subvector of $\vec{\beta}$ in our MCMC, we use a simple HMC step. Because of the convolution structure of X, computation of the conditional posterior and its gradient for B-entries-long subvectors of $\vec{\beta}$ scale as $O(B \log B)$. For a fixed block size B, adding processors in proportion to the length of the chromosome being analyzed maintains constant runtime. In addition, the proposed method has constant runtime with respect to the number of fragments observed, omitting alignment. As shown in Section 1.4.2, this scalable inference strategy leads to a high-quality sampler.

We propose a combination of Bayesian and frequentist techniques for the detection of local concentrations of nucleosomes. We use the local concentration estimands, $C_{p,l}(k)$, to define the structure of interest. We then use draws from the parallel HMC sampler to esti-

mate the posterior probabilities of these estimands exceeding their expected value under a locally uniform distribution of nucleosome positions. Instead of using these estimated posterior probabilities to make inferences directly, we calibrate them using frequentist multiple testing techniques (Storey and Tibshirani, 2003). Instead of relying upon the model to provide a null distribution for such calibration, we adopt a data-dependent permutation null in the spirit of Fisher's exact test. The calibration step provides guarantees on the behavior of the detection procedure under a permutation null and transforms our Bayesian posterior probabilities to the more standard, interpretable scale of FDRs and q-values.

The pragmatic approach to detection described above is one step in our broader approach to the analysis of nucleosome positioning. First, we used a probability model to build statistics that directly target the scientific estimands of interest, and performed inference with MCMC. Second, we used permutations to define a reference distribution based on the observed data and the segmentation, and detect local concentrations of nucleosome positions. Third, we evaluated the power, accuracy, and reproducibility of inferences from our method using biologically-motivated simulations, and technical and biological replicates. Each step in this strategy reflects less reliance upon modeling assumptions and a greater emphasis on external validity. The success of this strategy is reflected in the empirical results and simulation studies presented in Section 1.4. We obtain accurate, reproducible, scalable inferences about the genome-wide distribution of nucleosome positions with well-studied operating characteristics, providing new capabilities to this area of biology.

2

Absolute protein quantitation: Inference with non-ignorable missing data in high throughput proteomics

2.1 Introduction

Proteins are the leading actors in cellular processes, making them a prime target for biological investigation. Understanding their absolute concentrations within a population of

cells can elucidate the molecular dynamics that regulate their functions (Ishihama et al., 2005) and open a broad range of biological processes to deeper investigation. However, measuring the concentration of many proteins in a single experiment has been difficult (Ghaemmagham et al., 2003). As a result, many investigations have relied upon gene expression as a proxy for the concentration of these proteins (Franks et al., 2013). In recent years a new field of high-throughput proteomics has emerged, in which biologists and biochemists have begun exploring the next level of biological complexity with Fourier transform mass spectrometers (Scigelova and Makarov, 2006; Scigelova et al., 2011). High-throughput mass spectrometry can deliver a fine-grained view of cellular activity at an unprecedented scale and precision. In principle, this new technology enables the direct estimation of absolute protein concentrations from the relative intensities of protein fragments in a biological sample.

In practice, the measurement process implemented by mass spectrometers introduces complex systematic biases that must be accounted for to obtain valid estimates of absolute protein concentrations. Proteins are large macromolecules, consisting of intricately-folded sequences of amino acids. They are enzymatically digested into fragments, which are amenable to analysis by mass spectrometry (Thakur et al., 2011). Protein fragments are generally referred to as *peptides*; throughout this paper, fragments with the same sequence but distinct characteristics (e.g., charge states and chemical modifications) are considered *distinct* peptides. These peptides are analyzed by two separate mass spectrometers to produce quantitative summaries (Steen and Mann, 2004). Because of sample complexity and instrument limitations, however, not all of the peptides can be analyzed in the second-stage of mass spectrometry to have their sequence identified. The instrument is programmed to select the most abundant peptides in the first stage of mass spectrometry for sequencing, at any given time. This multi-stage measurement process results in a systematic bias towards observing peptides from the sample's most abundant proteins.

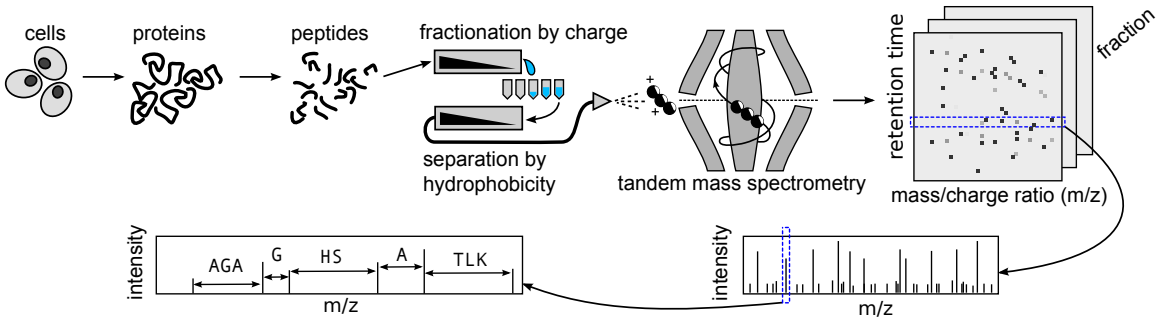

Figure 2.1: Overview of the measurement process in LC-MS/MS proteomics. Clockwise from top left: Cells are lysed to extract proteins, which are broken down into fragments by a protease. These fragments are then fractionated by charge and separated by hydrophobicity in the liquid chromatography (LC) stage before being sent through tandem mass spectrometry (MS/MS). This yields the spectrographic intensity (shown by shaded dots) and mass/charge ratio for each fragment retention time. A subset of these fragments are selected for further processing at each time, allowing for identification.

In this paper, we present and evaluate a statistical technique to correct these biases, providing reliable estimates of absolute protein abundances from mass spectrometry experiments.

Figure 2.1 illustrates the experimental process in detail. Starting from the top left, a culture of approximately 10^8 cells are opened (lysed) to extract their proteins, which are then digested with an enzyme (protease) into fragments (peptides). These peptides are then separated based on their hydrophobicity (via high-performance liquid chromatography). As peptides reach the spray tip gradually, they are given an electrical charge and fly into the mass spectrometer. All of these steps simply transform the long, complex mass of proteins from our cell culture into a well-separated set of simpler molecules that the mass spectrometers can process.

Fragments that are ionized within the same short time window are analyzed together by the first of two mass spectrometers (MS1). At this stage, fragments with the same sequences of amino acids are present in a number of different states as a result of ion charges, *in vivo* post-translational modifications and *in vitro* chemical modifications from

the sample preparation (Michalski et al., 2011). This step results in raw measurements $\xi(t, r)$ each of which quantifies the intensity[1] corresponding to ionized fragments with mass-to-charge ratio r, analyzed within a time window indexed by t. These intensities are proxies for relative abundance of fragment in the mass spectrometer within each time window. The rightmost sections of Figure 2.1 illustrate the output of this process, with each fraction yielding a set of intensities $\xi(t, r)$ for each (retention) time t and mass-to-charge ratio r. The lower right panel of this figure shows one slice through this dataset corresponding to a particular time t. A subset of the fragments analyzed by the first mass spectrometers is selected, based on their intensities, to be broken down into yet smaller fragment components by collision with a gas. The products of these collisions are then analyzed by a second mass spectrometer (MS2). This step results in an additional set of intensity measurements for these components, each of which has a distinct mass-to-charge ratio and is associated with its original fragment's mass-to-charge ratio by the instrument. The final (bottom left) panel of Figure 2.1 shows the intensity spectrum obtained by MS2 for the components of the fragments with mass-to-charge ratio r selected from MS1. The amino acid sequence of these fragments can often be reliably determined by analyzing differences in the mass-to-charge ratio between adjacent peaks in the MS2 spectrum.

The statistical problem of interest is to estimate the absolute concentrations of proteins in a sample from the observed peptide-level intensities $\{\xi(t, r) : t \in T, r \in R\}$. However, to tackle this problem, we must first associate these raw intensities with peptides and condense them into a more manageable set of summaries. This preprocessing falls under the heading of identification, a well-studied problem in MS/MS proteomics. In

[1] This intensity corresponds to the magnitude of a Fourier coefficient associated with peptides of mass-to-charge ratio r (Scigelova and Makarov, 2006). Modern mass spectrometers generally measure the amplitudes at which ionized protein fragments oscillate along an electrode over time. The Fourier transformation $\xi(f)$ of the amplitude time series from a mixture of ionized fragments provides the power associated with each frequency f. Each frequency f is associated with fragments of a particular mass-to-charge ratio r according to $f = C/r$, where C encodes instrument geometry and settings, yielding $\xi(t, r(f)) = \xi(t, r)$.

the identification task, the intensity and mass-to-charge data from MS2 are used to detect which peptides are present in the sample, in terms of their most likely sequence, charge, and possible chemical and post-translational modifications. In a typical run, hundreds of thousands of unique peptides are simultaneously detected. The observed spectrum of each detected peptide is then compared to the theoretical spectra of the all peptides that would be generated by digesting the proteins in the sample with a specific enzyme. For example, the enzyme trypsin digests the amino acids arginine (R) and lysine (K), so each protein is expected to be fragmented into peptides by removing all arginines and lysines from its amino acid sequence. Currently, several well-established methods for peptide/protein identification exist in the literature (Cox and Mann, 2008; Perkins et al., 1999; Eng et al., 1994); we generally use the MaxQuant software of Cox and Mann (2008).

Identification yields a set of integrated log-intensities y_{ikl}^{obs}, each of which is associated with peptide k from protein i, and l indexes distinct modifications and charge states. These integrated log-intensities are known to be approximately linearly related to the log of the number of molecules of that peptide present in the mass spectrometer, making them an excellent basis for absolute quantitation (Old et al., 2005; Scigelova et al., 2011). Formally, each y_{ikl} is the result of two operations: a mapping between identified fragment and a given set of values for $(t, r) \in \Delta_{ikl}$, and the integration of $\xi(t, r)$ over these values. The mapping serves to associate each observed $\xi(t, r)$ with a given peptide state ikl. The raw intensities $\xi(t, r)$ are then integrated over a small window in both retention time and mass-to-charge ratio. The former accounts for the fact that any given peptide ikl is typically observed across adjacent time windows, while the latter accounts for minor variation in r for each fragment originating from the presence of multiple isotopes and other factors. Thus, all

of our inferences are based on

$$y_{ikl}^{obs} = \log_{10}\left(\int\int_{(t,r)\in\Delta_{ikl}} \xi(t,r)\mathrm{d}r\mathrm{d}t\right). \tag{2.1}$$

With these quantities in-hand, we can refine the statement of our statistical problem: using the collection of integrated log-intensities y_{ikl}^{obs} and known properties of the observed proteins, estimate the log-absolute concentrations of proteins in our sample ζ_i. We present a probabilistic model for these data in Section 2.2, with particular details of the core estimand ζ_i provided in Section 2.2.2.

2.1.1 Related work

There are two threads of literature on protein quantitation: relative and absolute quantitation. In relative quantitation, the estimand of interest is the ratio of a given protein concentration in two distinct samples (generally different experimental conditions). In contrast, absolute quantitation requires the ability to estimate the concentration of all proteins in a single sample, relative to one another—these quantities together with the total amount of protein measured in the sample lead to the estimand of interest in this work, $\{\zeta_i : i \in I\}$, as we discuss in Section 2.2.2.

While there has been much work on the problem of peptide identification (e.g., Cox and Mann, 2008; Perkins et al., 1999; Eng et al., 1994), less progress has been made on the quantitation problem (Makawita and Diamandis, 2010). Despite the relatively strong correlation between the amount of each protein in a sample, the observed intensities, and the number of its peptides identified by the mass spectrometer (Old et al., 2005), heterogeneity in this relationship between labs, samples, and even peptides from a common protein has hampered the development of robust quantitation methods (Tabb et al., 2010; Bell et al., 2009). To control for these sources of variation, the popular methods for

relative quantitation rely on physically labeling each sample being compared, analyzing them simultaneously, and estimating the relative abundance protein-by-protein based on summaries of peptide intensity ratios (Ong et al., 2002). Existing experimental methods for estimating absolute abundance are quite intricate. For each protein of interest, the experimenter must synthesize a set of synthetic, standard peptides. They then compare the observed intensities of peptides from the sample to the intensities of the corresponding standard peptides, which are introduced at known concentrations (Gerber et al., 2003). These approaches have limited throughput, are experimentally complex, and are limited to quantitation of a preselected set of proteins.

Motivated by the limitations of experimental quantitation methodologies, there has been a growing interest in computational methods for relative and absolute quantitation. The simplest of these are based on spectral counting, either in the form of ratio-based estimates for relative quantitation (Liu et al., 2004), or using rescaled counts for absolute quantitation (Ishihama et al., 2005). More recently, a semi-supervised absolute quantitation method called APEX has been developed, which uses a large training dataset to learn how differences in physiochemical properties affect the probability of peptide identification, independent of abundance (Lu et al., 2006). It then uses these estimated probabilities of identification to construct a weighted estimator of protein abundance. Another class of methods uses peptide intensities, which have a wider dynamic range than spectral counts, for quantitation, the most common of which are based on simple summaries of observed intensities, such as their median (de Godoy et al., 2008; Silva, 2005). Most recently, a few attempts have been made to combine intensity and identification information, for example using principle component analysis to combine these features across samples (e.g., Dicker et al., 2010). Finally, the most sophisticated techniques, which motivated our own work, are model-based methods for relative quantitation that account for missing data at the peptide level (Karpievitch et al., 2009; Luo et al., 2009). However, by

focusing on peptide-level missingness, these methods fail to account for the true amount of missing data, since each peptide is generally found in a number of different states, due to biological and chemical modifications (Michalski et al., 2011).

2.1.2 Contributions of this article

In this paper, we introduce a statistical model that combines a hierarchical intensity model with an observation model for intensity-based censoring, in Section 2.2. This combination accounts for key aspects of the biology and of the data acquisition process, including the fact that peptides are observed in multiple charge and modification states, and that there are at least two missing data mechanisms that compound throughout any LC-MS/MS instrument.

Our approach is novel in a number of ways: (1) the focus is on absolute, rather than on relative quantitation; (2) while most existing approaches to quantitation involve simple summary statistics after identification, we posit and estimate a realistic censoring mechanisms that results in non-ignorable missing data; (3) the model explains the variability in the intensities of peptides observed in multiple charge and modification states, rather than the aggregated intensities for individual peptides. We find that accounting for non-ignorable missing data helps reduce the selection bias induced by the measurement process. In addition, modeling intensity-based censoring at the level of peptide states helps transfer information from abundant to rare proteins, since the number of states we expect to observe any given peptide in is independent of abundance. As we show in Sections 2.4 and 2.5, these aspects of the model allows robust, accurate estimation of absolute protein abundances in complex samples with concentrations spanning many orders of magnitude.

We provide efficient inference for this model, in Section 2.3, by leveraging a combination of Halley's method, adaptive Gauss-Hermite quadrature, and explicit envelopes for sampling the number of censored peptide states and the censored intensities. In Section

2.4, we explore the frequentist coverage of interval estimates, and we compare the performance of the proposed method with existing methods for estimating absolute protein concentrations, as we systematically vary the abundance and the length of the target proteins. In Section 2.5, we explore the extent to which key assumptions of our model hold in practice, and we analyze estimates based on three biological samples, processed with different LC-MS/MS settings, in which a set of proteins with known concentration was introduced for validation purposes.

2.2 Model

We develop a probabilistic model for the measurements, y_{ikl}^{obs}, the output of an LC-MS/MS experiment combined with identification analysis with standard software. The observed data consists of: the observed state-level intensities y_{ikl}^{obs}, indexed by protein i, peptide k, and charge state l, and the observed counts of observed states by peptides s_{ik}^{obs}. We also know the number of possible peptides per protein m_i, for a given enzyme used for digestion, which is independent of the sample in theory but requires careful preprocessing in practice.

We are interested in inferring the abundance of each protein i within the given sample. This abundance is a monotone function of the parameter μ_i as detailed in Section 2.2.2.

We posit the following model,

$$\gamma_{ik} \mid \mu_i, \tau_i^2 \sim \text{Normal}(\mu_i, \tau_i^2),\ i = 1, \ldots, n,\ k = 1, \ldots, m_i \tag{2.2}$$

$$s_{ik} \mid \lambda, r \sim 1 + \text{ Negative-Binomial } (1 - \lambda, r),\ k = 1, \ldots, m_i \tag{2.3}$$

$$y_{ikl} | \gamma_{ik}, \sigma_i^2 \sim \text{Normal}(\gamma_{ik}, \sigma_i^2),\ l = 1, \ldots, s_{ik} \tag{2.4}$$

$$R_{ikl} | \pi^{rnd}, s_{ik} \sim \text{Bernoulli}(1 - \pi^{rnd}),\ l = 1, \ldots, s_{ik} \tag{2.5}$$

$$I_{ikl} | y_{ikl}, \vec{\eta}, s_{ik}, R_{ikl} = 1 \sim \text{Bernoulli}(1 - g(y_{ikl}; \vec{\eta})),\ l = 1, \ldots, s_{ik} \tag{2.6}$$

$$O_{ikl} = R_{ikl} \cdot I_{ikl} \tag{2.7}$$

$$Y_{obs} = \{y_{ikl} : O_{ikl} = 1\} \tag{2.8}$$

$$Y_{mis} = \{y_{ikl} : O_{ikl} = 0\} \tag{2.9}$$

This model consists of two pieces: the distribution of the complete data $Y_{com} = (Y_{obs}, Y_{mis})$ given the parameters, and the distribution of the observed data given the complete data and parameters. Starting with the complete data for protein i, this model specifies that each peptide gets a mean intensity γ_{ik} distributed around the protein-mean μ_i. No prior distribution on μ_i is assumed. Similarly, each state-level intensity y_{ikl} is distributed around the peptide-level intensity. The variances of these distributions are τ_i^2 and σ_i^2 are drawn from inverse-Gamma distributions.

$$1/\tau_i^2 \sim \text{Gamma}(\alpha_\tau, \beta_\tau) \tag{2.10}$$

$$1/\sigma_i^2 \sim \text{Gamma}(\alpha_\sigma, \beta_\sigma) \tag{2.11}$$

This is a straightforward Normal hierarchical model except for one complication. The number of states per peptide s_{ik} is drawn from a shifted negative binomial distribution. This distribution is fixed across proteins and peptides, reflecting the physical indepen-

dence between the number of possible charge states and a protein's abundance. This invariance plays a crucial role in our inference, as we discuss in Sections 2.3 and 2.6. Hence, we have

$$P(Y_{com}|\vec{\mu},\vec{\sigma}^2,\vec{\tau}^2,r,\lambda) = \prod_{ik}\left[\frac{1}{\tau_i}\phi\left(\frac{\gamma_{ik}-\mu_i}{\tau_i}\right)\binom{s_{ik}+r-2}{s_{ik}-1}\lambda^r(1-\lambda)^{s_{ik}-1}\cdot \prod_{l=1}^{s_{ik}}\left[\frac{1}{\sigma_i}\phi\left(\frac{y_{ikl}-\gamma_{ik}}{\sigma_i}\right)\right]\right], \tag{2.12}$$

where $\phi(z)$ is defined as $\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}z^2\right)$.

2.2.1 Missing data mechanism

The missing data mechanism operates at the state level and is characterized using two random variables. The first is a random censoring indicator, $R_{ikl} \sim \text{Bernoulli}(1-\pi^{rnd})$, which accounts for censoring due to factors other than abundance. The second is an intensity censoring indicator, $I_{ikl} \sim \text{Bernoulli}(1-g(y_{ikl};\vec{\eta}))$, which accounts for the intensity-dependent censoring. We assume I_{ikl} is drawn only if $R_{ikl} = 1$, as shown in Figure 2.2. Hence, we have

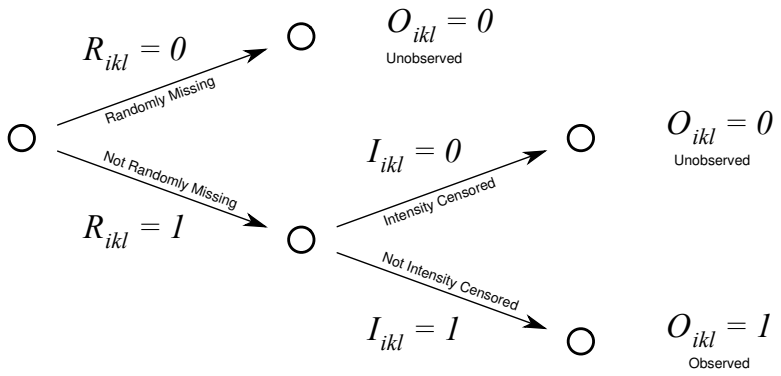

Figure 2.2: Possible missingness indicator values.

$$P(\vec{I}, \vec{R} | Y_{com}, \eta, \pi^{rnd}) = \prod_{ikl} (\pi^{rnd})^{1-R_{ikl}} (1-\pi^{rnd})^{R_{ikl}} (1-g(y_{ikl}; \eta))^{I_{ikl}R_{ikl}} g(y_{ikl}; \eta)^{(1-I_{ikl})R_{ikl}}, \quad (2.13)$$

which implies a corresponding distribution on the (redundant) variables $\vec{O} \equiv \vec{I} \circ \vec{R}$, where $\circ$ denotes element-wise product, which indicate whether each intensity *ikl* is observed. In particular, the marginal probability of observing a given peptide state given y_{ikl} and all other parameters is then

$$P(O_{ikl} = 1 | y_{ikl}, \vec{\Theta}) = (1 - \pi^{rnd})(1 - g(y_{ikl}; \vec{\eta})). \quad (2.14)$$

In combination with (2.12), these assumptions imply that

$$s_{ik}^{obs} | \vec{\gamma}, \vec{\sigma}^2, \vec{s} \sim \text{Binomial}(s_{ik}, (1 - \pi^{rnd})(1 - \pi_{ik}^{int})) \quad (2.15)$$

are conditionally independent across peptides, where $\pi_{ik}^{int} = \int_{\mathbb{R}} g(t; \eta) \frac{1}{\sigma_i} \phi \left(\frac{t - \gamma_{ik}}{\sigma_i} \right) dt$. From this, we see how the division of the missing data mechanism into random and intensity-based censoring adds flexibility to our model, allowing the probability of missingness to

asymptote to a value lower than 1 as intensity increases. However, it is important to note that this separation is largely conceptual, not physical. The random and intensity-based censoring mechanisms corresponds only roughly to the early and late stages of the LC-MS/MS process, respectively.

From a theoretical perspective, this missing data mechanism induces a stochastic dominance relationship between the distribution Y_{obs} and Y_{com}, as Theorem 1 establishes:

Theorem 1. *Suppose $X \sim F_X(x)$; that is, X has a cumulative distribution function that can be represented as a Riemann-Stieltjes integral over $\mathbb{R}$. Let $Z|X = x \sim \text{Bern}(g(x))$, where $g(x)$ is strictly increasing on $\mathbb{R}$ from 0 to 1. Then, the posterior distribution of X given $Z = 1$ stochastically dominates the original distribution of X.*

This result establishes that observed intensities will be biased upwards relative to the complete ones. We use this result for model checking in Section 2.5.2. A proof is provided in Section 2.8.

2.2.2 Estimands

The μ_i parameters are the primary target of the inference; however, they are not directly interpretable as absolute measures of protein abundance. To provide absolute measures, we must convert μ_i from the log-intensity scale to the scale of protein abundances. We define an estimand ζ_i for this purpose,

$$\zeta_i = \log_{10}\left(\frac{T \times 10^{\mu_i}}{\sum_{i=1}^{n} 10^{\mu_i}}\right) = \mu_i + \log_{10}(T) - \log_{10}\left(\sum_{i=1}^{n} 10^{\mu_i}\right), \tag{2.16}$$

where T is the total amount of proteins in the sample of interest. The key feature of this estimand is normalization by $\sum_{i=1}^{n} 10^{\mu_i}$, which provides the core conversion from log-intensities to the log-abundance scale (up to an additive constant). The total protein

Table 2.1: Prior Distributions.

$$
\begin{aligned}
\log(\alpha_\sigma) &\sim N(\mu_{o\alpha_\sigma}, v_{o\alpha_\sigma}) \\
\beta_\sigma &\sim Gamma(\alpha_{o\sigma}, \beta_{o\sigma}) \\
\log(\alpha_\tau) &\sim N(\mu_{o\alpha_\tau}, v_{o\alpha_\tau}) \\
\beta_\tau &\sim Gamma(\alpha_{o\tau}, \beta_{o\tau}) \\
\pi^{rnd} &\sim Beta(\alpha_{o\pi}, \beta_{o\pi}) \\
\lambda &\sim Beta(\alpha_{o\lambda}, \beta_{o\lambda}) \\
\log(r) &\sim N(\mu_{or}, v_{or})
\end{aligned}
$$

amount T serves primarily as a rescaling factor for interpretability, which converts estimates from log-proportion of proteins to log-molecules per cell or log-femtomoles. We assume that T is known and fixed. While this assumption is often challenged in practice, it is crucial to neither the validity nor the utility of our estimates. Modeling T is also a possibility.

2.3 Inference and estimation

We develop an efficient Monte Carlo Markov Chain algorithm to perform inference with the proposed model on proteome-wide data sets with hundreds of thousands of peptide states. We design a Metropolis-within-Gibbs algorithm, alternating between updates for the missing data and parameters. The algorithm consists of the following steps within each iteration:

1. Draw the censored peptide latent variables, $\mathbf{M}|\mathbf{Y^{obs}}, \vec{\Theta}$.

   (a) Draw the number of peptide states, $\vec{s}|\mathbf{Y^{obs}}, \vec{\Theta}$, using rejection sampling.

   (b) Draw the random censoring indicators, $\vec{W}|\vec{s}, \mathbf{Y^{obs}}, \vec{\Theta}$.

   (c) Draw the censored intensities, $\mathbf{Y^{mis}}|\vec{W}, \vec{s}, \vec{Y^{obs}}, \vec{\Theta}$, using rejection sampling.

2. Update the parameters $\vec{\Theta}$ given the complete data $(\vec{Y}^{\text{obs}}, \vec{M})$.

The updates of $\vec{\Theta}|\vec{Y^{obs}}, M$ are of a standard form. In Section 2.3.1, we provide the details of the exact update of $\vec{M}$ given $\vec{\Theta}$. We leave further details of our inference strategy are contained in the Appendix, which includes the complete specification the updates for $\vec{\Theta}$ given $\vec{M}$. Table 2.1 provides the prior distributions used in our inference; we provide the specific parameter values used in Appendix B.

2.3.1 Draw the censored peptide latent variables, $\mathbf{M}|\vec{\Theta}$.

Drawing the missing data $\mathbf{M} = \{\mathbf{Y}^{mis}, \mathbf{s}, \mathbf{R}\}$ is the most challenging component of the inference. The dimensionality of the missing data $\mathbf{M}$ is not fixed across iterations, so standard Metropolis-Hastings techniques are not enough. Reversible Jump Metropolis-Hastings (RJMH) methods provide one option, which we originally explored, but they proved too inefficient and fragile. Instead, we develop a partially marginalized update than draws from the exact conditional distribution of $\vec{M}$ given $(\vec{Y}^{obs}, \vec{\Theta})$. We implement this exact draw using a "triangular" dependence structure, starting with the easiest draws to marginalize:

$$\begin{aligned} p(\vec{M} \mid \vec{Y}^{obs}, \vec{\Theta}) \propto & p(\vec{s} \mid \vec{s}^{obs}, \vec{Y}^{obs}, \vec{\Theta}) \\ & \times p(\vec{R} \mid \vec{s}, \vec{Y}^{obs}, \vec{\Theta}) \\ & \times p(\vec{Y}^{mis} \mid \vec{R}, \vec{s}, \vec{Y}^{obs}, \vec{\Theta}) \end{aligned} \tag{2.17}$$

Using efficient numerical integration techniques (such as Gauss-Hermite quadrature) and exact sampling methods (involving explicit envelopes for rejection samplers), we generate exact draws from the joint posterior of the missing data using the above sequence of conditional distributions. Details of each of these draws is given in the following sections. Section 2.3.1 details the steps required to compute and sample $\vec{s}$ from $p(\vec{s} \mid \vec{s}^{obs}, \vec{Y}^{obs}, \vec{\Theta})$. Section 2.3.1 covers $p(\vec{R} \mid \vec{s}, \vec{Y}^{obs}, \vec{\Theta})$. Section 2.3.1 details the exact sampling strategy for

$p(\vec{Y}^{mis} \mid \vec{R}, \vec{s}, \vec{Y}^{obs}, \vec{\Theta})$.

Drawing from $p(\vec{s} \mid \vec{Y}^{obs}, \vec{\Theta})$

We derive the posterior of s_{ik} given $(\vec{Y}^{obs}, \vec{\Theta})$ by iteratively marginalizing over the remaining components of $\vec{M}$. For the derivations in this section, we define the number of unobserved states for peptide k of protein i as $s_{ik}^{mis} \equiv s_{ik} - s_{ik}^{obs}$; drawing $s_{ik}|s_{ik}^{obs}, Y^{obs}, \vec{\Theta}$ is then equivalent to drawing $s_{ik}^{mis}|s_{ik}^{obs}, Y^{obs}, \vec{\Theta}$. First, we note that the conditional posterior distribution of $\vec{M}$ factors by both protein and peptide, so it suffices to consider a single s_{ik}. This yields

$$p(s_{ik} \mid \vec{y}_{ik}^{obs}, s_{ik}^{obs}, \vec{\Theta}) \propto p(\vec{y}_{ik}^{obs} \mid s_{ik}, \vec{\Theta}) \cdot p(s_{ik} \mid \vec{\Theta}) \tag{2.18}$$

$$= \left[\int_{\mathbb{R}^{s_{ik}^{mis}}} p(\vec{y}_{ik}^{obs}, \vec{y}_{ik}^{mis} \mid s_{ik}, \vec{\Theta}) \, \mathrm{d}\vec{y}_{ik}^{mis} \right] \cdot p(s_{ik} \mid \vec{\Theta}) \tag{2.19}$$

$$\propto \begin{pmatrix} s_{ik}^{obs} + s_{ik}^{mis} \\ s_{ik}^{obs} \end{pmatrix} \cdot \left\{ \prod_{l=s_{ik}^{obs}+1}^{s_{ik}^{obs}+s_{ik}^{mis}} \int_{\mathbb{R}} \left[\sum_{R_{ikl}, I_{ikl}} p(y_{ikl}^{mis}, R_{ikl}, I_{ikl} \mid s_{ik}, \vec{\Theta}) \right] \mathrm{d}y_{ikl}^{mis} \right\}$$

$$\cdot \, p(s_{ik} \mid \vec{\Theta}), \tag{2.20}$$

where the last relationship follows by conditioning on $\vec{Y}^{obs}$ and expansion over the missingness indicators I_{ikl} and R_{ikl}. The combinatorial term enters the above expression due to the varying dimensionality of our missing data.

We then focus on $p(y_{ikl}^{mis}, R_{ikl}, I_{ikl} \mid s_{ik}, \vec{\Theta})$, obtaining

$$\sum_{R_{ikl}, I_{ikl}} p(y_{ikl}^{mis}, R_{ikl}, I_{ikl} \mid s_{ik}, \vec{\Theta}) = p(y_{ikl}^{mis} \mid s_{ik}, \vec{\Theta}) \cdot \Big(p(R_{ikl} = 0 \mid y_{ikl}^{mis}, s_{ik}, \vec{\Theta}) + p(R_{ikl} = 1 \mid y_{ikl}^{mis}, s_{ik}, \vec{\Theta}) \cdot p(I_{ikl} = 0 \mid R_{ikl} = 0, y_{ikl}^{mis}, s_{ik}, \vec{\Theta}) \Big) \tag{2.21}$$

$$= \frac{1}{\sigma_i} \phi \left(\frac{y_{ikl}^{mis} - \gamma_{ik}}{\sigma_i} \right) \cdot \Big(\pi^{rnd} + (1 - \pi^{rnd}) g(y_{ikl}^{mis}, \vec{\eta}) \Big). \tag{2.22}$$

Integrating this expression over y_{ikl}^{mis} yields

$$\int_{\mathbb{R}} p(y_{ikl}^{mis}, R_{ikl}, I_{ikl} \mid s_{ik}, \vec{\Theta}) \, \mathrm{d}y_{ikl}^{mis} = \pi^{rnd} + (1 - \pi^{rnd}) \int_{\mathbb{R}} \frac{1}{\sigma_i} \phi \left(\frac{y_{ikl}^{mis} - \gamma_{ik}}{\sigma_i} \right) g(y_{ikl}^{mis}, \vec{\eta}) \, \mathrm{d}y_{ikl}^{mis} \tag{2.23}$$

$$= \pi^{rnd} + (1 - \pi^{rnd}) \pi_{ik}^{int}, \tag{2.24}$$

where we define

$$\pi_{ik}^{int} = \int_{\mathbb{R}} \frac{1}{\sigma_i} \phi \left(\frac{y_{ikl}^{mis} - \gamma_{ik}}{\sigma_i} \right) g(y_{ikl}^{mis}, \vec{\eta}) \, \mathrm{d}y_{ikl}^{mis} \tag{2.25}$$

Substituting π_{ik}^{int} from (2.25) into (2.23) yields a simple form for the conditional posterior PMF of s_{ik}:

$$\begin{aligned} p(s_{ik} \mid \vec{y}_{ik}^{obs}, \vec{\Theta}) &\propto \begin{pmatrix} s_{ik}^{obs} + s_{ik}^{mis} \\ s_{ik}^{obs} \end{pmatrix} \cdot \Big(\pi^{rnd} + (1 - \pi^{rnd}) \cdot \pi_{ik}^{int} \Big)^{s_{ik}^{mis}} \cdot p(s_{ik} \mid \vec{\Theta}) \\ &\propto (s_{ik}^{obs} + s_{ik}^{mis}) \cdot NegativeBinomial(s_{ik}^{mis} \mid 1 - p_{ik}^{*}, \, s_{ik}^{obs} + r - 1) \end{aligned} \tag{2.26}$$

where $p_{ik}^{*} = (1 - \lambda)(\pi^{rnd} + \pi_{ik}^{IC}(1 - \pi^{rnd}))$. The conditional posterior given by (2.26) deviates from a Negative Binomial PMF only due to the constraint that $s_{ik} \geq 1$.

In order to draw from the posterior of s_{ik} exactly, we develop a rejection sampler using

a $NegativeBinomial(1 - p^*_{ik}, s^{obs}_{ik} + r)$ as the proposal distribution. We structure this as a draw from the conditional posterior of s^{mis}_{ik} for computational convenience and clarity of notation. If $s^{obs}_{ik} = 0$, $s^{mis}_{ik} - 1 \sim NegativeBinomial(1 - p^*_{ik}, r)$ exactly, so we consider only the $s^{obs}_{ik} > 0$ case. To construct this, we obtain a target-proposal ratio of

$$\frac{p(s^{mis}_{ik}|\vec{Y^{obs}}, \vec{\Theta})}{NegativeBinomial(1 - p^*_{ik}, s^{obs}_{ik} + r)} = \frac{c^*(s^{obs}_{ik} + s^{mis}_{ik})}{s^{obs}_{ik} + s^{mis}_{ik} + r - 1},$$

where c^* is a constant that is not a function of s^{mis}_{ik}. This ratio is bounded by

$$\frac{c^*(s^{obs}_{ik} + s^{mis}_{ik})}{s^{obs}_{ik} + s^{mis}_{ik} + r - 1} \leq \begin{cases} c^* & \text{if } r \geq 1 \\ \frac{c^* s^{obs}_{ik}}{s^{obs}_{ik} + r - 1} & \text{if } 0 < r < 1 \end{cases}.$$

Optimizing this with respect to c^* yields the following acceptance probabilities:

$$p(accept|X) = \begin{cases} \frac{(s^{obs}_{ik} + s^{mis}_{ik})}{(s^{obs}_{ik} + s^{mis}_{ik} + r - 1)} & \text{if } r \geq 1 \\ \frac{(s^{obs}_{ik} + s^{mis}_{ik})}{s^{obs}_{ik} + s^{mis}_{ik} + r - 1} \times \frac{s^{obs}_{ik} + r - 1}{s^{obs}_{ik}} & \text{if } 0 < r < 1 \end{cases}.$$

However, the integral required to compute π^{int}_{ik} is not generally available in closed form. We develop a simple, accurate numerical integration strategy based on adaptive Gauss-Hermite quadrature (Liu and Pierce, 1994). We first find the mode of the integrand $\hat{y}^{mis}_{ikl}$ and its logarithm's second derivative at the mode $\hat{v}^{mis}_{ik} \equiv \frac{1}{\sigma_i^2} - \frac{\partial^2}{\partial y^{mis\,2}_{ikl}} \log\left[g(y^{mis}_{ikl}, \vec{\eta})\right]\Big|_{y^{mis}_{ikl} = \hat{y}^{mis}_{ikl}}$. Since even this mode is not available in closed form, we use a vectorized version of Halley's method to efficiently approximate it. Using this information, we then approximate π^{int}_{ik} using Gauss-Hermite quadrature, with the nodes shifted and scaled based on $\hat{y}^{mis}_{ikl}$ and $\hat{v}^{mis}_{ik}$, yielding $\hat{\pi}^{int}_{ik}$. For moderate values of η_1, only a small number of nodes (10 or less) are typically required for accuracy to machine precision. We summarize this strategy in Figure 2.3.

The algorithm for drawing s_{ik}^{mis} is given in Algorithm 2. See Figure 2.4 for sample draws compared to the true density and the proposal negative binomial density.

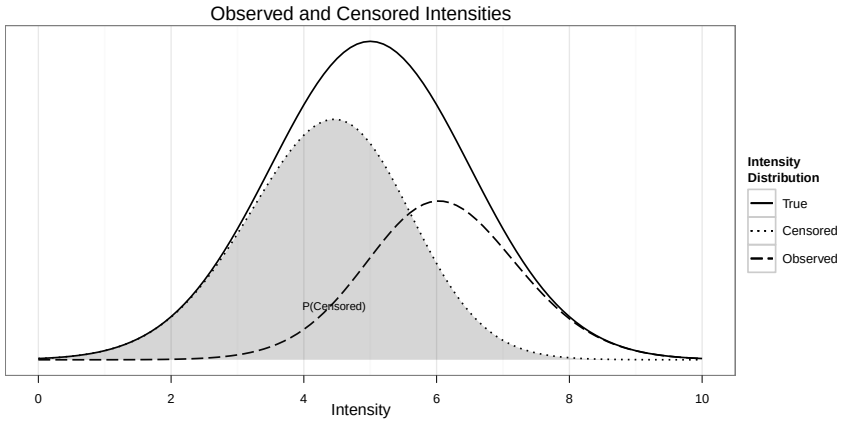

Figure 2.3: Comparisons of original, censored, and observed intensity distributions.

1. Draw $X \sim NegativeBinomial(1 - \hat{p}_{ik}^{*}, s_{ik}^{obs} + r)$, with
   $\hat{p}_{ik}^{*} = (1 - \lambda)\left[\pi^{rnd} + (1 - \pi^{rnd})\hat{\pi}_{ik}^{IC}\right]$ and $U \sim Uniform(0, 1)$.
2. Accept $s_{ik}^{mis} = X$, if $U \leq \begin{cases} \frac{(s_{ik}^{obs} + s_{ik}^{mis})}{(s_{ik}^{obs} + s_{ik}^{mis} + r - 1)} & \text{if } r \geq 1 \\ \frac{(s_{ik}^{obs} + s_{ik}^{mis})}{s_{ik}^{obs} + s_{ik}^{mis} + r - 1} \times \frac{s_{ik}^{obs} + r - 1}{s_{ik}^{obs}} & \text{if } 0 < r < 1 \end{cases}$.
3. Return to 1 otherwise.

Algorithm 2: s_{ik} Rejection Sampler

Drawing from $p(\vec{R} \mid \vec{s}, \vec{Y}^{obs}, \vec{\Theta})$

After drawing the number of states per peptide $s_i k$, we draw the latent random censoring indicators for each censored peptide, R_{ikl}^{mis}. Because random censoring occurs before intensity-based censoring, if a peptide was randomly censored ($R_{ikl} = 0$), then $p(I_{ikl} = 1 | R_{ikl} = 0, \vec{\Theta}) = 0$ and $p(O_{ikl} = 1 | R_{ikl} = 0, \vec{\Theta}) = 0$ as outlined in Figure 2.2. We then ob-

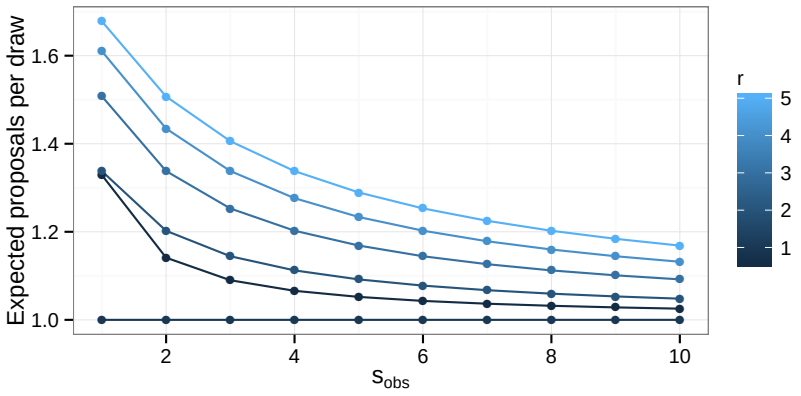

Figure 2.4: Expected iterations per accepted draw for s_{ik}^{mis} rejection sampler for $\lambda = 0.1$, $\pi^{rnd} = 0.1$, and $\pi_{ik}^{IC} = .5$, with r ranging from 0.5 to 5, s_{ik}^{obs} ranging from 1 to 10.

tain the posterior probability that a peptide is randomly censored given that it is missing via a straightforward application of Bayes Theorem:

$$\begin{aligned} p(R_{ikl} = 1 | O_{ikl} = 0, \vec{\Theta}, \vec{Y}^{obs}) &= 1 - \frac{p(O_{ikl} = 0 | R_{ikl} = 0, \vec{\Theta}, \vec{Y}^{obs}) p(R_{ikl} = 0 | \vec{\Theta}, \vec{Y}^{obs})}{\sum_{r_{ikl}=0}^{1} p(O_{ikl} = 0 | R_{ikl} = r_{ikl}, \vec{\Theta}) p(R_{ikl} = r_{ikl} | \vec{\Theta}, \vec{Y}^{obs})} \\ &= 1 - \frac{\pi^{rnd}}{\pi^{rnd} + (1 - \pi^{rnd})\pi_{ik}^{int}}, \end{aligned} \tag{2.27}$$

where π_{ikl}^{int} is defined as in (2.25). Plugging in our numerical approximation for π_{ikl}^{int}, $\hat{\pi}_{ikl}^{int}$, we draw

$$R_{ikl} \mid \vec{Y}^{obs}, \vec{\Theta}, s_{ik} \sim Bernoulli\left(1 - \frac{\pi^{rnd}}{\pi^{rnd} + (1 - \pi^{rnd})\hat{\pi}_{ik}^{int}}\right). \tag{2.28}$$

Drawing from $p(\vec{Y}^{mis} \mid \vec{R}, \vec{s}, \vec{Y}^{obs}, \vec{\Theta})$

The final step in drawing the missing data is to impute the unobserved intensities by drawing each intensity from its full conditional posterior distribution. The full conditional

posterior is given by

$$p(y_{ikl}^{mis}|s_{ik}, R_{ik}, \vec{\Theta}) \quad \propto \quad \begin{cases} \phi\left(\frac{y_{ikl}-\gamma_{ik}}{\sigma_i}\right) & \text{if } R_{ikl} = 0 \text{ and } O_{ikl} = 0 \\ \phi\left(\frac{y_{ikl}-\gamma_{ik}}{\sigma_i}\right) g(y_{ikl}, \vec{\eta}) & \text{if } R_{ikl} = 1 \text{ and } O_{ikl} = 0 \end{cases}.$$

For randomly censored states, where $R_{ikl} = 0$, the missingness mechanism is ignorable given R_{ikl}. Hence, we simply draw y_{ikl}^{mis} from its unconditional distribution,

$$p(y_{ikl}^{mis}|s_{ik}, I_{ik} = 0, R_{ikl} = 0, \vec{\Theta}) \quad \sim \quad Normal(\gamma_{ik}, \sigma_i^2).$$

The posterior distribution for intensity-censored states ($R_{ikl} = 1$, $I_{ikl} = 0$) is given by

$$p(y_{ikl}^{mis}|O_{ikl} = 0, R_{ikl} = 1, \vec{\Theta}) \quad = \quad (\pi_{ik}^{int})^{-1} \frac{1}{\sigma_i} \phi\left(\frac{y_{ikl}^{mis} - \gamma_{ik}}{\sigma_i}\right) g(y_{ikl}^{mis}, \vec{\eta}). \qquad (2.29)$$

As this is not a standard distribution, we draw from it using a rejection sampler. We use information from the adaptive quadrature of Section 2.3.1 to construct an efficient proposal distribution with little additional computation. Specifically, we propose from a location-scale transformation of a t_ν-distribution as

$$Y_{ikl}^{*} \quad \sim \quad \hat{y}_{ikl}^{mis} + \hat{\sigma}_{ik}^{mis} \sqrt{\frac{\nu - 2}{\nu}} t_\nu, \qquad (2.30)$$

which has expectation $\hat{y}_{ik}^{mis}$ and variance $\hat{\sigma}_{ik}^{mis})^2$ for $\nu > 2$.

The rejection sampling algorithm requires bounding the ratio of the target density to the proposal density. For this sampler, the given ratio has two local maxima, as shown in Figure 2.5 B and C. The global maximum of the ratio can be either the first or third root of the log-ratio's derivative depending upon the particular relationship between $\vec{\eta}$, σ_i^2, and $\gamma_i k$. To reliably find the global maximum of the target-proposal density ratio, we use a

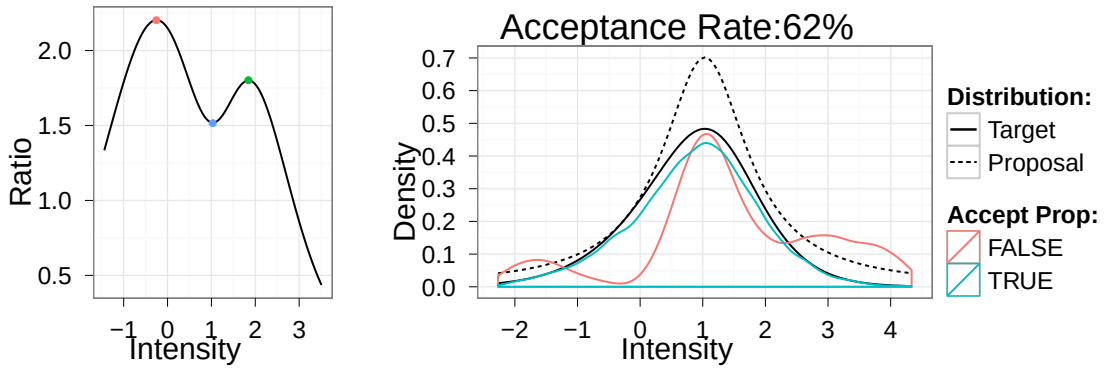

Figure 2.5: Censored Intensity Rejection Sampler Example

pair of vectorized bisection searches to find the roots of the derivative of the log-ratio in two ranges: $(-\infty, \hat{y}_{ik}^{mis})$ and $(\hat{y}_{ik}^{mis}, \infty)$. Once these roots are obtained, we simply select the one corresponding to the larger local maximum to compute acceptance probabilities. A graphical example of the rejection sampler is shown in Figure 2.5 D, and the sampling algorithm is detailed in Algorithm 3.

2.4 Simulation studies

To evaluate the performance of our method and the stability of the algorithm presented in Section 2.3, we tested our method on a set of simulated MS-MS experiments. We describe the design of these simulations in detail in Section 2.4.1. We cover the computational performance and validation of our MCMC sampler in Section 2.4.2. In Section 2.4.3, we evaluate the frequentist properties of our Bayesian estimates, including the coverage of our posterior intervals (2.4.3) and the performance of the proposed method relative to standard estimators used for this class of experiments (2.4.3).

Target density: $f^{mis}(y_{ikl}|\eta_0, \eta_1, \gamma_{ik}, \sigma_i^2) = (\pi_{ik}^{int})^{-1} \frac{1}{\sigma_i} \phi\left(\frac{y_{ikl}-\gamma_{ik}}{\sigma_i}\right) g(y_{ikl}, \vec{\eta})$.

Proposal density: $f^*(y_{ikl}^*|\nu, \tilde{\mu}_{ikl}, \tilde{\sigma}_{ikl}) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)} \left(1 + \frac{(y_{ikl}^*-\tilde{\mu}_{ikl})^2}{\nu\tilde{\sigma}_{ikl}^2}\right)^{-\left(\frac{\nu+1}{2}\right)}$.

1. Compute z_{ikl}:

   (a) Using bisection, find the roots of of the first derivative of the log density ratio,
   $z_{ikl}^{(1)} \equiv \underset{y_{ikl}\in[-\infty,\tilde{\mu}_{ikl})}{\arg\max} \log\left[\frac{f^{mis}(y_{ikl}|\eta_0,\eta_1,\gamma_{ik},\sigma_i^2)}{f^*(y_{ikl}|\nu,\tilde{\mu}_{ikl},\tilde{\sigma}_{ikl})}\right]$ and
   $z_{ikl}^{(2)} \equiv \underset{y_{ikl}\in(\tilde{\mu}_{ikl},\infty]}{\arg\max} \log\left[\frac{f^{mis}(y_{ikl}|\eta_0,\eta_1,\gamma_{ik},\sigma_i^2)}{f^*(y_{ikl}|\nu,\tilde{\mu}_{ikl},\tilde{\sigma}_{ikl})}\right]$.

   (b) Determine the maximum of the density ratios,
   $c^* \equiv \max\left(\frac{f^{mis}(z_{ikl}^{(1)}|\eta_0,\eta_{int},\gamma_{ik},\sigma_i^2)}{f^*(z_{ikl}^{(1)}|\nu,\tilde{\mu}_{ikl},\tilde{\sigma}_{ikl})}, \frac{f^{mis}(z_{ikl}^{(2)}|\eta_0,\eta_1,\gamma_{ik},\sigma_i^2)}{f^*(z_{ikl}^{(2)}|\nu,\tilde{\mu}_{ikl},\tilde{\sigma}_{ikl})}\right)$.

2. Generate $X \sim g(y_{ikl}^*|\nu, \tilde{\mu}_{ikl}, \tilde{\sigma}_{ikl})$, $U \sim Unif(0, 1)$.

3. Accept $y_{ikl} = X$, if $U \leq \frac{f^{mis}(X|\eta_0,\eta_{int},\gamma_{ik},\sigma_i^2)}{c^* f^*(X|\nu,\tilde{\mu}_{ikl},\tilde{\sigma}_{ikl})}$.

4. Else, return to 2.

Algorithm 3: y_{ikl}^{mis} rejection sampler algorithm

Table 2.2: Parameter settings for simulation studies.

Gradient	α_τ	β_τ	α_σ	β_σ	π^{rnd}	η_0	η_1	λ	r
90m	6.3	2.1	8.5	3.5	0.20	16	-3.0	0.84	1.9
180m	5.3	1.3	7.0	4.1	0.21	14	-2.7	0.79	1.9
360m	6.2	1.8	7.0	2.5	0.30	20	-4.3	0.62	1.6

2.4.1 Design of experiments

We simulated a set of complex biological samples, spanning a broad range of abundances and protein-specific properties. For each such sample we simulate 54 proteins with abundances spanning 6 orders magnitude, $\mu_i = 3, \ldots, 8$, yielding 9 proteins per abundance level. Within each abundance level, each simulated protein consists of $m_{ik} = 20, 25 \ldots, 60$ peptides. With these properties fixed across replicates, we simulated 1,200 experiments from the model described in Section 2.2, using a probit link function for $g(\cdot, \vec{\eta})$. These consisted for 400 replicates for each of the parameter settings provided in Table 2.2, each of which was based on estimates from an experiment with the given gradient length using the Sigma UPS2 standard.

2.4.2 MCMC performance and validation

Our MCMC sampler produced high-quality draws from the target posterior at a low computational cost. Running 3,000 iterations for each replicate required an average of 267 seconds (0.088 seconds per iteration) with a standard deviation of 17.1 seconds. Of these 3,000 iterations, we discarded the first 1,000 as burn in. The mean effective sample sizes for each top-level parameter based on the remaining 2,000 draws are provided in Table 2.3. For μ_i and ζ_i, we also include the mean standard deviation of effective sample sizes across proteins. We note that all mean effective sample sizes are greater than 100, indicat-

Table 2.3: Average effective sample sizes by parameter. $\pm$ for μ_i and ζ_i indicates the average standard deviation of effective sample sizes across proteins.

Gradient	ζ_i	η	λ	μ_i	π^{rnd}	r	β_σ	β_τ	α_σ	α_τ
90m	341±188	287	773	331±191	191	1160	272	217	303	242
180m	256±143	285	823	250±148	175	1197	337	176	379	191
360m	421±208	179	1013	411 ±215	275	1338	346	222	392	250

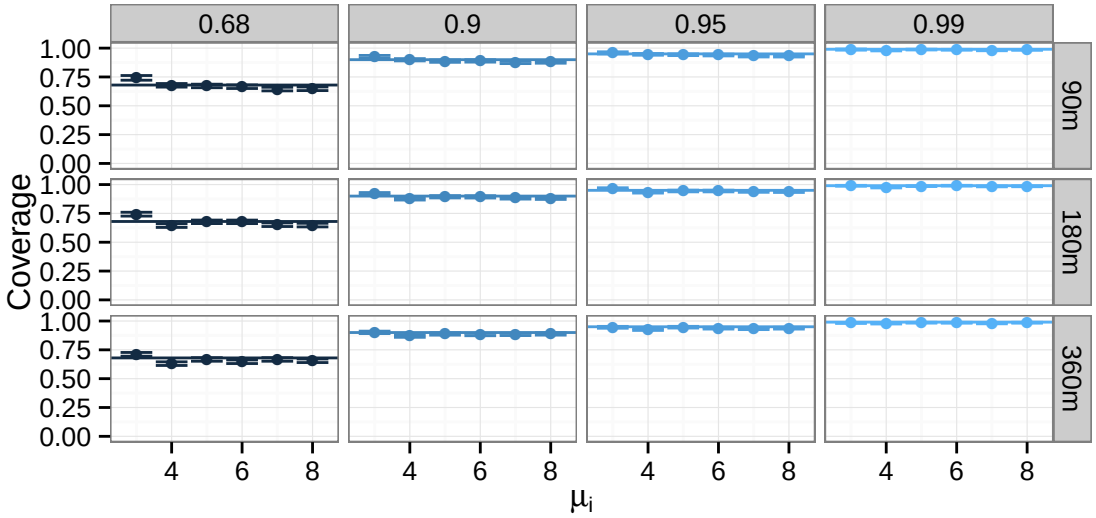

Figure 2.6: Coverage of HPD intervals vs. μ_i by simulated gradient length and Bayesian level.

ing that even 2,000 iterations are sufficient to obtain Monte Carlo an order of magnitude below the posterior standard deviation of each parameter.

2.4.3 Frequentist evaluations

Coverage analysis

We first used our simulated replicates to evaluate the frequency coverage of our posterior intervals, focusing on those for ζ_i. The results of these evaluations are summarized in Figure 2.6; corresponding results for coverage vs. m_i are similar and provided in Appendix B. We find that our posterior intervals are well-calibrated for the regimes of interest. Across

all three simulated gradient lengths, we obtain 67% coverage for our 68% posterior intervals, 89% mean coverage for our 90% posterior intervals, 94% mean coverage for our 95% posterior intervals, and 98% mean coverage for our 99% posterior intervals. These demonstrate that, despite the complexity of our Bayesian method, it can provide inferences with frequentist guarantees.

Comparison with exiting methods

We compared the performance of our model-based abundance estimates with several standard methods for absolute protein quantitation. Many of these methods were originally developed for relative quantitation and are converted to absolute measures by appropriate rescaling. All methods used in this comparison are of the form

$$\hat{\zeta}_i = \log_{10}\left(\hat{T}\frac{10^{\hat{\mu}_i}}{\sum_i 10^{\hat{\mu}_i}}\right). \tag{2.31}$$

We compare the performance of our method to two classes of common methods: intensity-based estimators and count-based estimators. We set $\hat{T} = \sum_i 10^{\mu_i}$ for all of these estimators. Among intensity-based estimators, we consider simple summed intensities (2.32) and median observed intensity (2.33) (e.g., Cox and Mann, 2008). We also evaluate two variants based on estimators from the spectral counting literature: mean observed intensity (2.34) and adjusted mean observed intensity (2.35)

$$\begin{aligned}
\hat{\mu}_i^{si} &= \log_{10}\left(\sum_{kl} 10^{y_{ikl}^{obs}}\right), && (2.32)\\
\hat{\mu}_i^{med} &= \mathrm{Med}_{kl}(y_{ikl}), && (2.33)\\
\hat{\mu}_i^{mi} &= \log_{10}\left(\sum_{kl} 10^{y_{ikl}^{obs}}/m_i^{obs}\right), && (2.34)\\
\hat{\mu}_i^{ami} &= \log_{10}\left(\sum_{kl} 10^{y_{ikl}^{obs}}/\sum_k s_{ik}^{obs}\right). && (2.35)
\end{aligned}$$

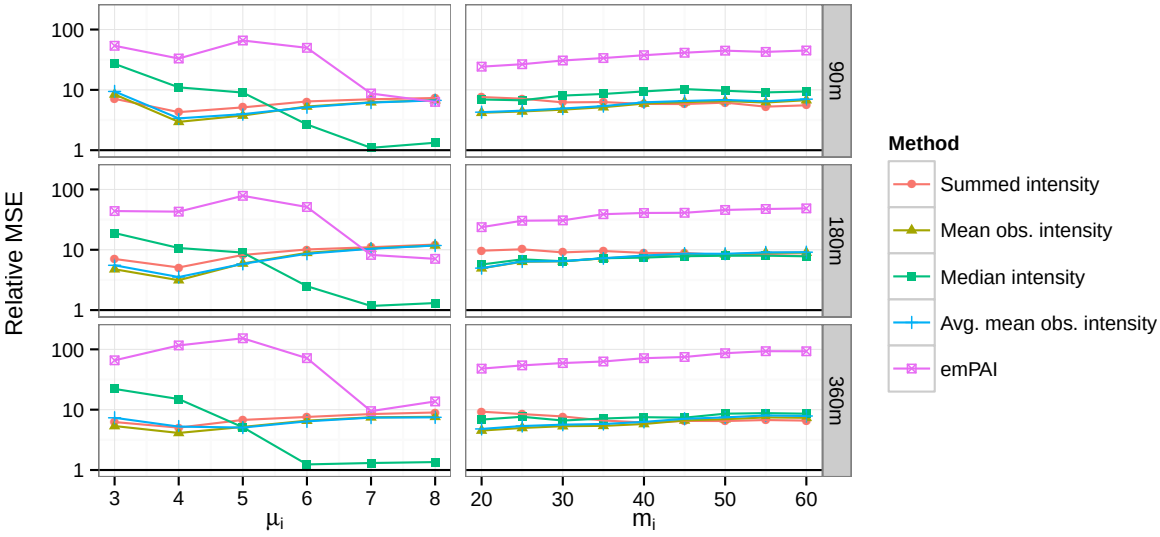

Figure 2.7: Relative efficiency of standard methods ($MSE_{Method}/MSE_{Model-based}$) vs. μ_i and m_i by block (left and right columns, respectively). Relative MSE axis is logarithmic. Solid black line at 1 corresponds to the model-based method's efficiency, by definition.

In the above, m_i is the (known) number of possible peptides that are generated by digesting protein i using a particular enzyme.

Another common approach, known as spectral counting, disregards the intensity measurements and uses counts of the observed states associated with each protein. A range of such methods have been developed relative quantitation, including basic spectral counting (e.g., Liu et al., 2004), average spectral counts (e.g., Weiss et al., 2010), and the proportion of peptides identified (PPI) (Rappsilber et al., 2002). However, these are not directly suitable for absolute quantitation due to a lack of normalization. Ishihama et al. (2005) defined an exponentially-modified protein abundance index (emPAI) to address these issues:

$$\hat{\mu}_i^{emPAI} = \log_{10}\left(10^{\sum_k \mathbb{I}(s_{ik}^{obs}>0)/m_i} - 1\right). \tag{2.36}$$

This provides a commonly-used representative of the count-based class of methods.

We summarize the results of these comparisons in Figures 2.7 and 2.8. Complete tab-

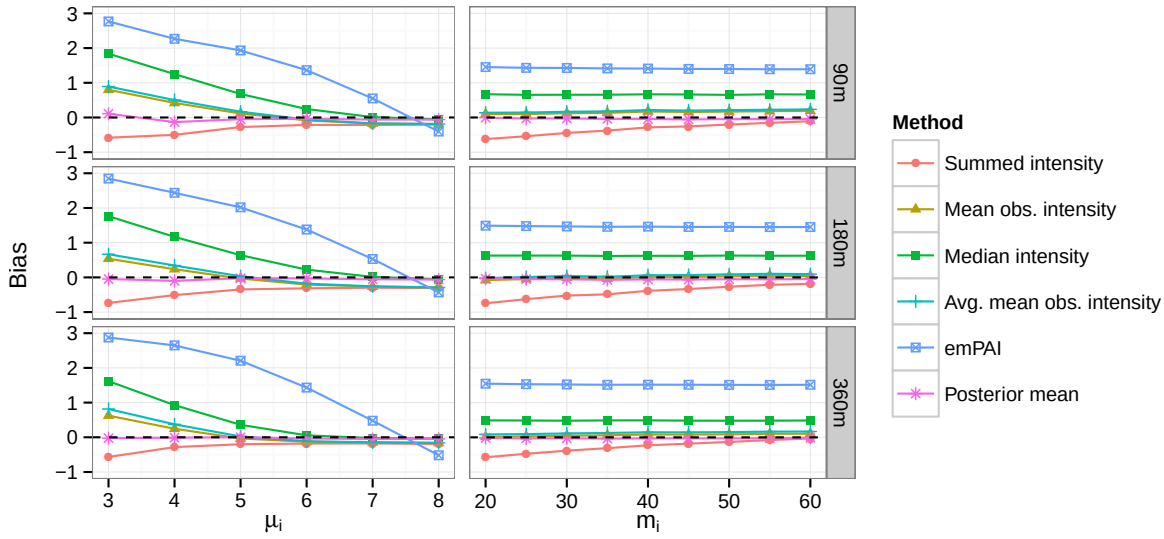

Figure 2.8: Bias of $\hat{\zeta}_i$ for standard methods and posterior mean from model vs. μ_i and m_i by gradient length (left and right columns, respectively).

ular results are available in Appendix B. From Figure 2.7, we see that our method reduces the overall MSE of abundance estimates by a factor of 5–10 relative to intensity-based estimates and by a factor of 6–150 relative to emPAI. Most intensity-based estimators exhibit consistent efficiency relative to ours, although the median intensity estimator's efficiency improves to nearly 1 as abundance increases. However, this estimator's efficiency is quite poor for low-abundance proteins with approximately 20 times the model-based estimator's MSE. The efficiency of emPAI increases by an order of magnitude over the range of simulated abundances and decreases slightly with protein length m_i, demonstrating the value of intensity information in low-abundance regimes. By carefully combining count and intensity information, we obtain consistent reductions in MSE relative to both classes of methods for all values of μ_i and m_i.

Turning to bias, we see that the model-based estimator provides lower bias than both classes of standard methods. Summed intensity shows a slight negative bias (between -0.7 and 0), while mean and average mean observed intensity show a positive bias of sim-

ilar magnitude. The median intensity estimator shows a far larger positive bias (nearly 2 orders of magnitude) at low abundances. The biases of all four intensity-based estimators diminish as abundance increases. The emPAI estimator shows a large positive bias that also decreases in magnitude with abundance. The bias of the summed intensity estimator diminishes from approximately -0.7 to 0 as protein length increases; the biases of our other estimators exhibits little sensitivity to protein length. Again, we see that our model-based approach enables us to combine count and intensity information into estimators that outperform both classes of standard methods.

2.5 Empirical results

Having established our method's properties on simulated data, we turn to actual experimental data. We use a quantitative protein standard, described in Section 2.5.1, for these experiments. This standard provides known abundances for evaluation with a realistic set of proteins. In Section 2.5.2, we use these experiments to check our model assumptions and demonstrate their implications for real data. In Section 2.5.3, we assess the comparative performance of the proposed methods with respect to the standard methods presented in Section 2.4.3.

2.5.1 Data

We conducted a set of LC/MS-MS experiments using the Universal Proteome Standard 2 (UPS2), a constructed biological sample that contains 48 human proteins. These proteins are present in six concentrations, ranging from .5fM to 50,000 fM, with eight known proteins at each concentration. These proteins are selected to span a broad range of physical properties (size and hydrophobicity) within each concentration, removing a potential confounding factor while reflecting the actual variation among proteins in typical sam-

ples. The concentration specified for each protein in the standard was confirmed by the manufacturer using multiple methods, so we are comfortable taking the concentrations reported by the manufacturer as ground truth for subsequent analyses. With its high dynamic range, the standard provides a realistic and stringent test of our method.

We ran five LC/MS-MS experiments using this standard at three gradient lengths, 90, 180, and 360 minutes, with two, two, and one replicates per gradient length, respectively. Each gradient length implies a different set of parameters for the missing data mechanism. Longer gradients allow for the observation of a greater variety MS2 spectra and is expected to decrease the degree of intensity-based missingness. However, longer gradients can also reduce the dynamic range of observed intensities and require substantially greater effort from the experimenter. Our experiments allow us to explore these tradeoffs in as we evaluate our method. The 90 minute gradients reflect the degree of censoring typically observed in analyses of more complex mixtures, while the 180 and 360 minute gradients provide a "sanity check" of our method's behavior in a setting with less missing data.

2.5.2 Exploratory analysis and model checking

Using the experimental data, we examined the distributions of intensities, identifications, and states to check the assumptions of our model. Figure 2.9 summarizes these results. Panel A of this figure shows the observed distribution of intensities by concentration. It is immediately clear from these histograms that the number of intensities observed and the median observed intensity increases strongly with the concentration. The distributions of observed intensities are also right-skewed for each concentration level, which is consistent with the combination of intensity-dependent censoring and an underlying log-normal distribution of intensities.

Panels B, C, and D show the relationship between concentration and identification, me-

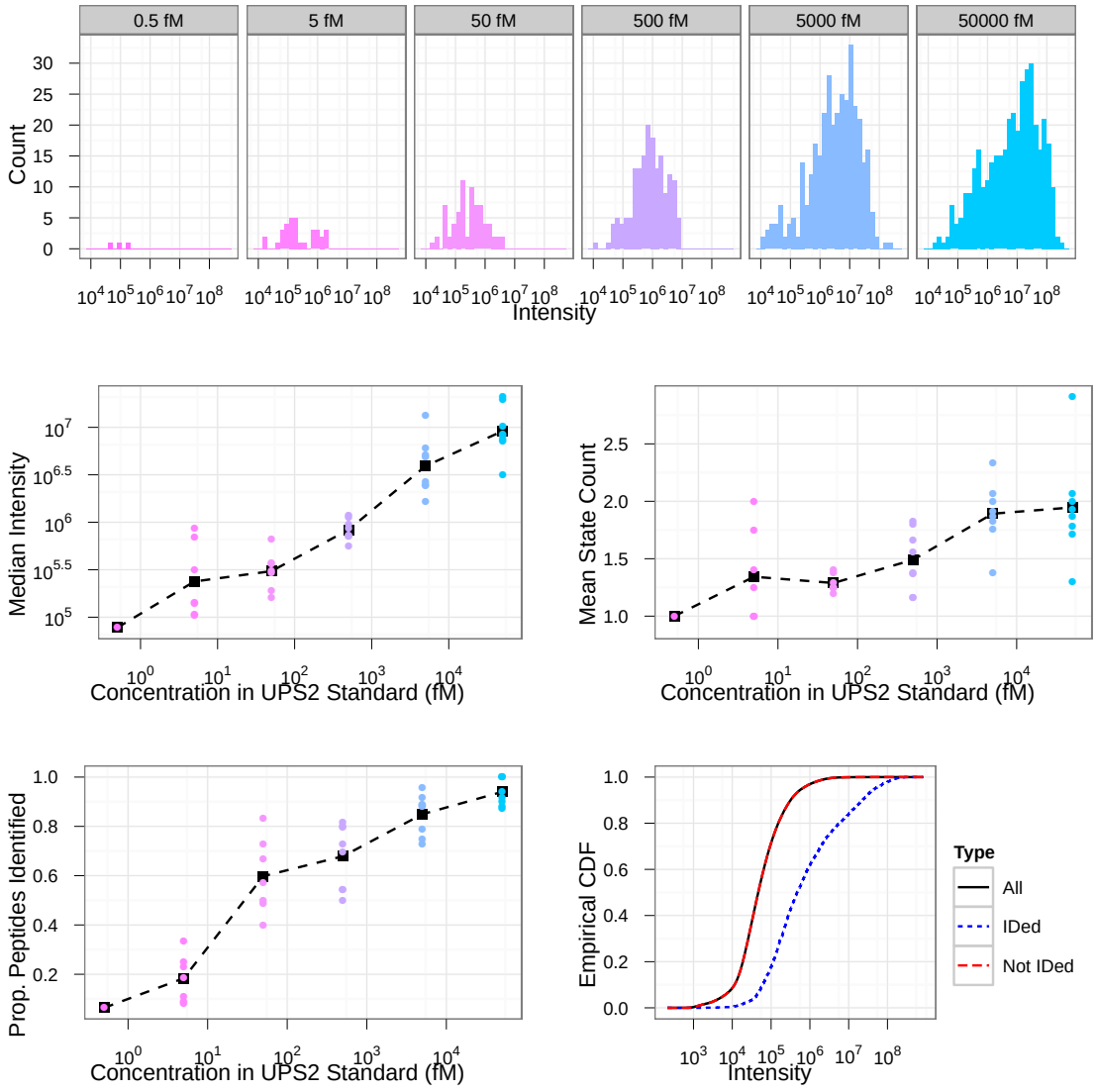

Figure 2.9: Exploratory data analysis using the UPS2 quantitative protein standard.

dian intensity, and s_{ik}^{obs} in greater detail. The median observed intensity by protein increases with the protein's concentration, although the former has far less dynamic range than the latter (2 orders of magnitude vs. 5). Intensity-based censoring at the state level explains this discrepancy, as the lowest-intensity ions are preferentially removed from the sample. We also compared the number of states that peptides were detected in across protein concentrations. The physical process that generates distinct peptide states is known to be independent of protein or peptide abundance, so the number of *possible* peptide states must be invariant to the peptide's concentration. Since it is not possible to determine the number of possible states per for peptide *a priori*, we model this value as an IID random variable in our model. However, we observe a strong relationship between the number of observed peptide states and concentration in the experimental data. This supports the assumption of intensity-dependent state-level censoring in our model.

The proportion of peptides identified per protein ranges nearly 1 for the most abundant proteins to approximately 5% for the least abundant, while the This is mostly attributable to selection for abundant ions within the MS-MS equipment, as panel E demonstrates. By design, the subset of peptides selected for fragmentation is determined by the instrument's software in real time as the sample is analyzed. While we focus on the subset of peptides that are both detected by the mass spectrometer and successfully identified, nearly all of the peptides in the sample can be quantified in terms of their total intensity[2]. This allows us to investigate the intensity-dependence of the peptide identification process in detail and validate our model assumptions. Comparing the distributions of identified and unidentified state-level intensities in Panel E shows that (1) the distribution of identified

[2]Software is able to quantify peptides that were not identified is due to the both mass accuracy of the mass spectrometer and the effective separation of the sample by chromatography. The software used to process the experimental data (Cox and Mann, 2008) is able to identify and integrate over all unique profiles in the m/z $\times$ intensity $\times$ retention time space, as shown in Figure 2.1 C. This allows quantitation of the majority of peptides; however, since the majority of these are not identified, there is no way to map them to a specific protein. Hence, we exclude them from our model-based inference.

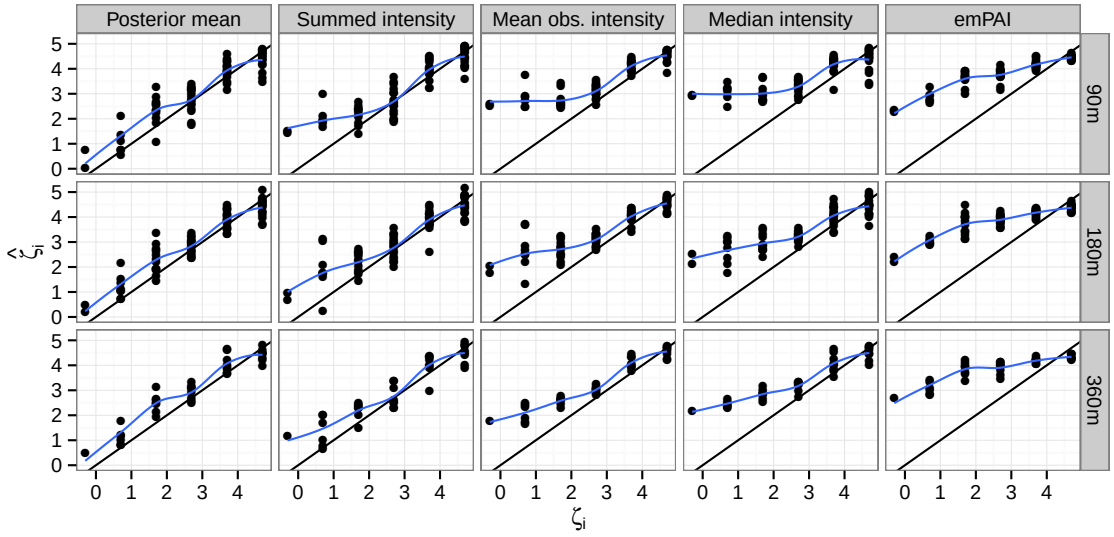

Figure 2.10: Inferred protein abundances from experiments with UPS2 standard.

Table 2.4: Mean squared errors ± standard errors from experiments with UPS2 standard.

	Gradient length		
Method	90m	180m	360m
Posterior mean	0.41 ± 0.11	0.34 ± 0.07	0.37 ± 0.06
Summed intensity	1.06 ± 0.12	0.76 ± 0.14	0.65 ± 0.07
Mean obs. intensity	2.45 ± 0.16	1.82 ± 0.21	1.26 ± 0.08
Median intensity	3.18 ± 0.13	2.26 ± 0.21	1.87 ± 0.06
emPAI	2.81 ± 0.10	3.03 ± 0.12	3.55 ± 0.13

intensities stochastically dominates that of the unidentified intensities and (2) there is substantial overlap between these distributions. (1) is consistent with Theorem 1, while (2) is consistent with the assumption of a stochastic censoring mechanism.

2.5.3 Comparison of empirical results

We used the experiments described in Section 2.5.1 to evaluate our method's performance against the standard methods described in Section 2.4.3. We summarize these results in Figure 2.10 and Table 2.4 and provide complete tables of results in Appendix B. We restrict

our attention here to the best-performing subset of the estimators presented in Section 2.4.3. We see from Table 2.4 that our method's performance is particularly strong on short gradients, providing a factor of 2.6 reduction in MSE relative to the best standard estimator (summed intensity) for the 90 minute gradients and a factor of 2.2 reduction for the 180 minute gradients. On the longest gradient (360 minutes), our method's MSE is a factor of 1.8 lower than than that of the best standard estimator. This behavior matches our expectations based on the relevant properties of the underlying experimental processes. Detailed modeling of the missing data mechanism yields the largest gains when the sample is least separated and there are fewer possible spectra—in the shortest gradients. In the longest gradient, intensity-based missingness diminishes in importance relative to other factors.

From Figure 2.10, we note that our method's improvements are greatest for low-abundance proteins. All estimators considered provide reasonably good estimates of ζ_i for the highest-abundance proteins. However, most of the standard estimators exhibit a substantial positive bias (1-2 orders of magnitude) for the lowest-abundance proteins, particularly on short gradients. In contrast, our method provides consistently accurate estimates for low-abundance proteins across all gradient lengths. We discuss the implications of this capability for biological applications in Section 2.6.

2.6 Concluding remarks

We have presented a model-based approach to label-free absolute quantitation in LC/MS-MS proteomics experiments. As the results of Sections 2.4 and 2.5 show, accounting for non-ignorable missing data in the proteomics context improves estimates of absolute abundances. Our method also provides well-calibrated measures of uncertainty, as the results of Sections 2.4.3 and 2.5.3 show. Below, we provide some broader context and

particular insights on this class of methods and problems, focusing on the role of non-ignorable missing data in complex experiments, efficient computation with missing data of variable dimension, and the role of absolute quantitation in biological analyses. We also discussion extensions of the methods presented to the semi-supervised setting and distributed computational environments.

2.6.1 Modeling and inference

The core of our model for LC-MS/MS proteomics data is the structure of our non-ignorable missing data mechanism. We explicitly account for the dependence of censoring probabilities on unobserved peptide intensities at the level of individual peptide states. This creates an additional complication, as the number of possible states per peptide is unknown. Thus, we must account for truncation at the state level in addition to censoring at the peptide level. However, we do know that the distribution of then number possible peptide states is invariant to the peptide's concentration. Incorporating this information into our model allows us to extract information from the number of states observed per peptide (s_{ik}^{obs}).

Working with state-level data provides fine-grained measures of missingness, which provide a great deal of information on low-abundance proteins when used properly. For high-abundance proteins, the vast majority of peptide states are observed, providing the majority of the information on the underlying distribution of the number of states per peptides. Our model uses this information to improve inferences on the abundances of low-abundance proteins, leveraging all of the available information from their intensity and state information. Extracting all of this information requires a sophisticated approach to inference and computation, however.

The presence of state-level censoring with an unknown number of states per peptide makes inference under our missing data model particularly challenging. The dimension of

our missing data $\vec{M}$ varies across draws, requiring more specialized computational strategies than standard Metropolis-Hastings updates. We implemented an exact marginal draw from the conditional distribution of $\vec{M}$ given $\vec{\Theta}$ using a combination of unidimensional numerical integration and efficient rejection sampling, as detailed in Section 2.3.1. The reductions in autocorrelation provided by this marginal update more than offset the computational costs of rejection sampling and numerical approximation. This allows for efficient inference in the presence of missing data of unknown dimension, as the results of Section 2.4 demonstrate. We believe that this technique can provide effective computational techniques for a range of non-ignorable missing data problems originating from complex experimental techniques.

2.6.2 Applications

Absolute quantitation provides a different view on biological processes than relative quantitation, enabling the study of new phenomena. Whereas relative quantitation can only measure the protein-by-protein differences in abundances across experiments, absolute quantitation provides estimates of the abundances of different proteins within a single population of cells. Such information is somewhat interesting for high-abundance proteins, but it promises the greatest insights from the study of low-abundance ones.

Reliable, accurate quantitation of low-abundance proteins also makes LC/MS-MS methods applicable to new classes of biological investigations. Two of the most promising areas are the study of mistranslation and transcription factors. Both involve proteins which appear at low concentrations but can have an outsized biological impact. In the case of mistranslation, the cell must spend a great deal of energy handling defective proteins; the rate of mistranslation represents the largest determinant of these costs. Transcription factors regulate the conversion of DNA into mRNA, providing one of the primary mechanisms of control and feedback for cellular functions. Our method allows

for reliable estimates of these proteins' concentrations within large populations of cells, opening new areas for high-throughput research.

2.6.3 Extensions

We are developing two extensions of the proposed methodology with a focus on larger, shallower datasets. In relatively shallow datasets, few proteins are observed and, due to intensity-based censoring, the dynamic range of observed intensities is compressed. This makes estimation of the censoring parameters (η and π^{rnd}) from observed data difficult, reducing the precision of our estimates. The quantitative protein standard described in Section 2.5.1 provides a potential solution to this problem. Such a standard can be spiked into the biological sample of interest and used as a calibration source. A semi-supervised version of the model presented in Section 2.2 can use the known properties of this standard to enable reliable inferences about the censoring mechanism in shallow samples. As an additional benefit, this supervision could simplify the conversion of our estimates between the intensity and abundance scales.

To tackle larger datasets, including full-proteome quantitation for complex organisms, we are also developing a distributed variant of the MCMC algorithm presented here. The goal is to scale the sampler across computational clusters with minimal communication within iterations. The conditional independence structure of the model presented in Section 2.2 makes such distribution both appealing and feasible.

We see a rich range of opportunities for statistical research in this area. The field of MS-MS proteomics is young, and it can provide a unique view on the details of biological processes. Extracting this technology's full value requires care, deep subject-matter knowledge, and sophisticated statistical techniques, but the rewards are great. We hope that more statisticians will enter this field and continue to build strong methodological foundations to advance biological research.

2.7 Acknowledgments

This work was partially supported by the National Science Foundation under grants no. DMS-0907009 and no. IIS-1017967, by the National Institute of Health under grant no. R01 GM-096193, all to Harvard University. Additional funding was provided by the Harvard Medical School's Milton Fund.

2.8 Proof of Theorem 1

Proof. Define Y as a random variable with a distribution given by

$$\mathrm{d}F_Y(y) = \frac{1}{w} g(y) \mathrm{d}F_x(y)$$

where $w \equiv \int_{\mathbb{R}} g(t)\mathrm{d}F_x(t) \equiv P(Z = 1)$. We assume we are in a non-trivial case, so $w \in (0, 1)$. Then, define $h \equiv g^{-1}(w)$. Consider the difference $D(t) \equiv F_X(t) - F_Y(t)$. This difference is equivalent to

$$D(t) = \int_{\mathbb{R}} \left(1 - \frac{1}{w} g(x)\right) \mathrm{d}F_X(x)$$

We now consider two cases:

(1) If $t \leq h$, then we have $g(x) \leq w$ for all $x \in (-\infty, h])$ by monotonicity. Thus, over the same range, $\left(1 - \frac{1}{w} g(x)\right) \geq 0$, so $D(t) \geq 0$.

(2) If $t > h$, then

$$D(t) = \int_{-\infty}^{h} \left(1 - \frac{1}{w} g(x)\right) \mathrm{d}F_X(x) + \int_{h}^{t} \left(1 - \frac{1}{w} g(x)\right) \mathrm{d}F_X(x)$$

Now, we have $D(h) \geq 0$ from case (1) and we know $D(h) \to 0$ as $t \to \infty$ as it is

the difference of CDFs. By monotonicity, for $x \geq h$, $g(x) \geq w$, so $1 - \frac{1}{w}g(x) \leq 0$ for $x \in [h, t]$. Thus, as $\mathrm{d}F_X(x) \geq 0$, we know that $\int_h^t \left(1 - \frac{1}{w}g(x)\right) \mathrm{d}F_X(x)$ is (weakly) monotone decreasing in t and ≤ 0. Thus, $D(t)$ increases monotonically to $D(h)$, then decreases monotonically to zero from h as $t \to \infty$.

Therefore, for any value of t, we have shown that $D(t) \geq 0$. Thus, Y stochastically dominates X.

□

3

The promise and perils of preprocessing: building foundations for multiphase inference

3.1 Summary

Preprocessing forms an oft-neglected foundation for a wide range of statistical and scientific analyses. However, it is rife with subtleties and pitfalls. Decisions made in preprocessing constrain all later analyses and are typically irreversible. Hence, data analysis becomes a collaborative endeavor by all parties involved in data collection, preprocessing

and curation, and downstream inference. Even if each party has done its best given the information and resources available to them, the final result may still fall short of the best possible in the traditional single-phase inference framework. This is particularly relevant as we enter the era of "big data". The technologies driving this data explosion are subject to complex new forms of measurement error. Simultaneously, we are accumulating increasingly massive databases of scientific analyses. As a result, preprocessing has become more vital (and potentially more dangerous) than ever before.

We propose a theoretical framework for the analysis of preprocessing under the banner of multiphase inference. We provide some initial theoretical foundations for this area, including distributed preprocessing, building upon previous work in multiple imputation. We motivate this foundation with two problems from biology and astrophysics, illustrating multiphase pitfalls and potential solutions. These examples also emphasize the motivations behind multiphase analyses—both practical and theoretical. We demonstrate that multiphase inferences can, in some cases, even surpass standard single-phase estimators in efficiency and robustness. Our work suggests several rich paths for further research into the statistical principles underlying preprocessing. To tackle our increasingly complex and massive data, we must ensure that our inferences are built upon solid inputs and sound principles. Principled investigation of preprocessing is thus a vital direction for statistical research.

3.2 What is multiphase inference?

3.2.1 Defining multiphase problems

Preprocessing and the analysis of preprocessed data are ubiquitous components of statistical analysis, but their treatment has often been informal. We aim to develop a theory that provides a set of formal statistical principles for such problems under the banner of

multiphase inference. The term "multiphase" refers to settings in which inferences are obtained through the application of multiple procedures in sequence, with each procedure taking the output of the previous phase as its input. This encompasses settings such as multiple imputation (MI, Rubin, 1987) and extends to other situations. In a multiphase setting, information can be passed between phases in an arbitrary form; it need not consist of (independent) draws from a posterior predictive distribution, as is typical with multiple imputation. Moreover, the analysis procedure for subsequent phases is not constrained to a particular recipe, such as Rubin's MI combining rules (Rubin, 1987).

The practice of multiphase inference is currently widespread in applied statistics. It is widely used as an analysis technique within single publications—any paper that uses a "pipeline" to obtain its final inputs or clusters estimates from a previous analysis provides an example. Furthermore, projects in astronomy, biology, ecology, and social sciences (to name a small sampling) increasingly focus on building databases for future analyses as a primary objective. These projects must decide what levels of preprocessing to apply to their data and what additional information to provide to their users. Providing all of the original data clearly allows the most flexibility in subsequent analyses. In practice, the journey from raw data to a complete model is typically too intricate and problematic for the majority of users, who instead choose to use preprocessed output.

Unfortunately, decisions made at this stage can be quite treacherous. Preprocessing is typically irreversible, necessitating assumptions about both the observation mechanisms and future analyses. These assumptions constrain all subsequent analyses. Consequently, improper processing can cause a disproportionate amount of damage to a whole body of statistical results. However, preprocessing can be a powerful tool. It alleviates complexity for downstream researchers, allowing them to deal with smaller inputs and (hopefully) less intricate models. This can provide large mental and computational savings.

Two examples of such trade-offs come from NASA and high-throughput biology. When

NASA satellites collect readings, the raw data are usually massive. These raw data are referred to as the "Level 0" data (Evans et al., 2006). The Level 0 data are rarely used directly for scientific analyses. Instead, they are processed to Levels 1, 2, and 3, each of which involves a greater degree of reduction and adjustment. Level 2 is typically the point at which the processing becomes irreversible. Braverman et al. (2012) provide an excellent illustration of this process for the Atmospheric Infrared Sounder (AIRS) experiment. This processing can be quite controversial within the astronomical community. Several upcoming projects, such as the Advanced Technology Solar Telescope (ATST) will not be able to retain the Level 0 or Level 1 data (Davey, 2012). This inability to obtain raw data and increased dependence on preprocessing has transformed low-level technical issues of calibration and reduction into a pressing concern.

High-throughput biology faces similar challenges. Whereas reproducibility is much needed (e.g., Ioannidis and Khoury, 2011), sharing raw datasets is difficult because of their sizes. The situation within each analysis is similar. Confronted with an overwhelming onslaught of raw data, extensive preprocessing has become crucial and ubiquitous. Complex models for genomic, proteomic, and transcriptomic data are usually built upon these heavily-processed inputs. This has made the intricate details of observation models and the corresponding preprocessing steps the groundwork for entire fields.

To many statisticians, this setting presents something of a conundrum. After all, the ideal inference and prediction will generally use a complete correctly-specified model encompassing the underlying process of interest and all observation processes. Then, why are we interested in multiphase? We focus on settings where there is a natural separation of knowledge between analysts, which translates into a separation of effort. The first analyst(s) involved in preprocessing often have better knowledge of the observation model than those performing subsequent analyses. For example, the first analyst may have detailed knowledge of the structure of experimental errors, the equipment used, or the par-

ticulars of various protocols. This knowledge may not be easy to encapsulate for later analysts—the relevant information may be too large or complex, or the methods required to exploit this information in subsequent analyses may be prohibitively intricate. Hence, the practical objective in such settings is to enable the best possible inference given the constraints imposed and provide an account of the trade-offs and dangers involved. To borrow the phrasing of Meng and Romero (2003) and Rubin (1996), we aim for achievable practical efficiency rather than a theoretical efficiency that is practically unattainable.

Multiphase inference currently represents a serious gap between statistical theory and practice. We typically delineate between the informal work of preprocessing and feature engineering and formal, theoretically-motivated work of estimation, testing, and so forth. However, the former fundamentally constrains what the latter can accomplish. As a result, we believe that it represents a great challenge and opportunity to build new statistical foundations to inform statistical practice.

3.2.2 Practical motivations

We present two examples that show both the impetus for and perils of undertaking multiphase analyses in place of inference with a complete, joint model. The first concerns microarrays, which allow the analysis of thousands of genes in parallel. We focus on expression microarrays, which measure the level of gene expression in populations of cells based upon the concentration of RNA from different genes. These are typically used to study changes in gene expression between different experimental conditions.

In such studies, the estimand of interest is typically the log-fold change in gene expression between conditions. However, the raw data consist only of intensity measurements for each probe on the array, which are grouped by gene along with some form of controls. These intensities are subject to several forms of observation noise, including additive background variation and additional forms of interprobe and interchip variation

(typically modeled as multiplicative noise). To deal with these forms of observation noise, a wide range of background correction and normalization strategies have been developed (for a sampling, see Tusher et al., 2001; Quackenbush, 2002; Affymetrix, 2002; Irizarry et al., 2003; McGee and Chen, 2006; Ritchie et al., 2007; Xie et al., 2009). Later analyses then focus on the scientific question of interest without, for the most part, addressing the underlying details of the observation mechanisms.

Background correction is a particularly crucial step in this process, as it is typically the point at which the analysis moves from the original intensity scale to the log-transformed scale. As a result, it can have a large effect on subsequent inferences about log-fold changes, especially for genes with low expression levels in one condition (Smyth, 2005; Irizarry et al., 2006). One common method (MAS5), provided by one microarray manufacturer, uses a combination of background subtraction and truncation at a fixed lower threshold for this task (Affymetrix, 2002). Other more sophisticated techniques use explicit probability models for this de-convolution. A model with normally-distributed background variation and exponentially distributed expression levels has proven to be the most popular in this field (McGee and Chen, 2006; Xie et al., 2009).

Unfortunately, even the most sophisticated available techniques pass only point estimates onto downstream analyses. This necessitates ad-hoc screening and corrections in subsequent analyses, especially when searching for significant changes in expression (e.g., Tusher et al., 2001). Retaining more information from the preprocessing phases of these analyses would allow for better, simpler inference techniques with greater power and fewer hacks. The motivation behind the current approach is quite understandable: scientific investigators want to focus on their processes of interest without becoming entangled in the low-level details of observation mechanisms. Nevertheless, this separation can clearly compromise the validity of their results.

The role of preprocessing in microarray studies extends well beyond background cor-

rection. Normalization of expression levels across arrays, screening for data corruption, and other transformations preceding formal analysis are standard. Each technique can dramatically affect downstream analyses. For instance, quantile normalization equates quantiles of expression distributions between arrays, removing a considerable amount of information. This mutes systematic errors (Bolstad et al., 2003), but it can seriously compromise analyses in certain contexts (e.g., miRNA studies).

Another example of multiphase inference can be found in the estimation of correlations based upon indirect measurements. This appears in many fields, but astrophysics provides one recent and striking case. The relationships between the dust's density, spectral properties, and temperature are of interest in studies of star-forming dust clouds. These characteristics shed light on the mechanisms underlying star formation and other astronomical processes. Several studies (e.g., Dupac et al., 2003; Désert et al., 2008; Anderson et al., 2010; Paradis et al., 2010) have investigated these relationships, finding negative correlations between the dust's temperature and spectral index. This finding is counter to previous astrophysical theory, but it has generated many alternative explanations.

Such investigations may, however, be chasing a phantasm. These correlations have been estimated by simply correlating point estimates of the relevant quantities (temperature T and spectral index β) based on a single set of underlying observations. As a result, they may conflate properties of this estimation procedure with the underlying physical mechanisms of interest. This has been noted in the field by Shetty et al. (2009), but the scientific debate on this topic continues. Kelly et al. (2012) provide a particularly strong argument, using a cohesive hierarchical Bayesian approach, that improper multiphase analyses have been a pervasive issue in this setting. Improper preprocessing led to incorrect, negative estimates of the correlation between temperature and spectral index, according to Kelly et al. (2012). These incorrect estimates even appeared statistically significant with narrow confidence intervals based on standard methods. On a broader level, this case again

demonstrates some of the dangers of multiphase analyses when they are not carried out properly. Those analyzing this data followed an intuitive strategy: estimate what we want to work with (T and β), then use it to estimate the relationship of interest. Unfortunately, such intuition is not a recipe for valid statistical inference.

3.2.3 Related work

Multiphase inference has wide-ranging connections to both the theoretical and applied literatures. It is intimately related to previous work on multiple imputation and missing data (Rubin, 1976, 1987; Meng, 1994; Rubin, 1996; Meng and Romero, 2003; Xie and Meng, 2012). In general, the problem of multiphase inference can be formulated as one of missing data. However, in the multiphase setting, missingness arises from the preprocessing choices made, not a probabilistic response mechanism. Thus, we can leverage the mathematical and computational methods of this literature, but many of its conceptual tools need to be modified. Multiple imputation addresses many of the same issues as multiphase inference and is indeed a special case of the latter. Concepts such as congeniality between imputation and analysis models and self-efficiency (Meng, 1994) have natural analogues and roles to play in the analysis of multiphase inference problems.

Multiphase inference is also tightly connected to work on the comparison of experiments and approximate sufficiency, going back to Blackwell (1951, 1953) and continuing through Le Cam (1964) and Goel and DeGroot (1979), among others. This literature has addressed the relationship between decision properties and the probabilistic structure of experiments, the relationship between different notions of statistical information, and notions of approximate sufficiency—all of these are quite relevant for the study of multiphase inference. We view the multiphase setting as an extension of this work to address a broader range of real-world problems, as we will discuss in Section 3.3.3.

The literature on Bayesian combinations of experts also informs our thinking on multi-

phase procedures. Kadane (1993) provides an excellent review of the field, while Lindley et al. (1979) provides the core formalisms of interest for the multiphase setting. Overall, this literature has focused on obtaining coherent (or otherwise favorable) decision rules when combining information from multiple Bayesian agents, in the form of multiple posterior distributions. We view this as a best-case scenario, focusing our theoretical development towards the mechanics of passing information between phases. We also focus on the sequential nature of multiphase settings and the challenges this brings for both preprocessors and downstream analysts, in contrast to the more "parallel" or simultaneous focus of the literature mentioned above.

There are also fascinating links between multiphase inference and the signal processing literature. There has been extensive research on the design of quantizers and other compression systems; see for example Gray and Neuhoff (1998). Such work is often focused on practical questions, but it has also yielded some remarkable theory. In particular, the work of Nguyen et al. (2009) on the relationship between surrogate loss functions in quantizer design and f-divergences suggests possible ways to develop and analyze a wide class of multiphase procedures, as we shall discuss in Section 3.5.2.

3.3 Multiphase logic and concepts for preprocessing

3.3.1 A model for two phases

To formalize the notion of multiphase inference, we begin with a formal model for two-phase settings. The first phase consists of the data generation, collection, and preprocessing, while the second phase consists of inference using the output from the first phase. We will call the first-phase agent the "preprocessor" and the second-phase agent the "downstream analyst". The preprocessor observes the raw data Y. This is a noisy realization of X, variables of interest that are not directly obtainable from a given experiment, e.g., gene

expression from sequencing data, or stellar intensity from telescopic observations.

We assume that the joint density of X and Y with respect to product measure $\mu_X \times \mu_Y$ can be factored as

$$p_{Y,X}(Y, X \mid \theta, \xi) = p_Y(Y \mid X, \theta, \xi) \cdot p_X(X \mid \theta, \xi) = p_Y(Y \mid X, \xi) \cdot p_X(X \mid \theta) \,. \tag{3.1}$$

Here, p_X encapsulates the underlying process of interest and p_Y encapsulates the observation process. We assume that θ is of fixed dimension in all asymptotic settings. In practice, the preprocessor should be able to postulate a reasonable "observation model" $p_Y(Y \mid X, \xi)$, but will not always know the true "scientific model" $p_X(X \mid \theta)$. This is analogous to the MI setting, where the imputer does not know the form of the final analysis.

Using this model, the preprocessor provides the downstream analyst with some output $T = T(Y, U)$, where U is a (possibly stochastic) additional input. When $T(Y, U)$ is stochastic (e.g., an MCMC output), the conditional distribution $p_T(T|Y)$ is its theoretical description instead of its functional form. However, for simplicity, we will present our results when T is a deterministic function of Y only, but many results generalize easily. Given such T, downstream analysts can carry out their inference procedures. Figure 3.1 depicts our general model setup.

This model incorporates several restrictions. First, it is Markovian with respect to Y, X, and θ; Y is conditionally independent of θ given X (and ξ). Second, the parameters governing the observation process (ξ) and those governing the scientific process (θ) are distinct. In Bayesian settings, we further assume that ξ and θ are independent *a priori*. The parameters ξ are nuisance from the perspective of all involved; the downstream analyst wants to draw inferences about X and θ, and the preprocessor wants to pass forward information that will be useful for said inferences. If downstream inferences are Bayesian with respect

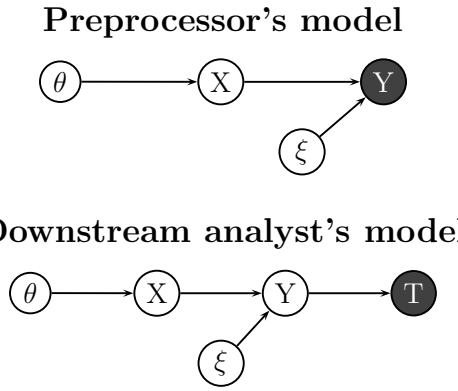

Figure 3.1: Graphical diagram of our generic two-phase setting. The preprocessor observes Y from the original data generating process and outputs T, with X as missing data. The downstream analyst observes the preprocessor's output T and has both X and Y missing.

to ξ, then $p_Y(Y|X) = \int p_Y(Y|X,\xi)\pi_\xi(\xi)\mathrm{d}\mu_\xi(\xi)$ (which holds under (3.1)) is sufficient for all inference under the given model and prior. Hence, this conditional density is frequently of interest in our theoretical development, as is the corresponding marginalized model $p_{X,Y}(Y,X|\theta) = \int p_Y(Y|X,\xi)p_X(X|\theta)\pi_\xi(\xi)\mathrm{d}\mu_\xi(\xi)$. We will compare results obtained with a fixed prior to those obtained in a more general setting to better understand the effects of nuisance parameters in multiphase inference.

These restrictions are somewhat similar to those underlying Rubin's (1976) definition of "missing at random"; however, we do not have missing data mechanism (MDM) in this setting *per se*. The distinction between missing and observed data (X and Y) is fixed by the structure of our model. In place of MDM, we have two imposed patterns of missingness: one for the data-generating process, and one for the inference process. The first is $p_Y(Y \mid X, \xi)$, which creates a noisy version of the desired scientific variables. Here, X can be considered the missing data and Y the observed. For the inference process, the downstream analyst observes T in place of Y but desires inference for θ based upon $p_X(X|\theta)$. Hence, Y and X are both missing for the downstream analyst. Neither pattern is entirely intrinsic to the problem—both are fixed by choice. The selection of scientific variables X for a given marginal likelihood $p_Y(Y|\theta,\xi) = \int p_Y(Y|X,\xi)p_X(X|\theta)\mathrm{d}\mu_X(X)$ is a modeling

decision. The selection of preprocessing $T(Y)$ is a design decision. This contrasts with the typical missing data setting, where MDM is forced upon the analyst by nature. With multiphase problems, we seek to design and evaluate engineered missingness. Thus the investigation of multiphase inference requires tools and ideas from design, inference, and computation in addition to the established theory of missing data.

3.3.2 Defining multiphase procedures

With this model in place, we turn to formally defining multiphase procedures. This is more subtle than it initially appears. In the MI setting, we focus on complete-data procedures for the downstream analyst's estimation and do not restrict the dependence structure between missing data and observations. In contrast, we restrict the dependence structure as in (3.1), but place far fewer constraints on the analysts' procedures. Here, we focus our definitions and discussion on the two phase case of a single preprocessor and downstream analyst. This provides the formal structure to describe the interface between any two phases in a chain of multiphase analyses.

In our multiphase setting, downstream analysts need not have any complete-data procedure in the sense of one for inferring θ from X and Y; indeed, they need not formally have one based only upon X for inferring θ. We require only that they have a set of procedures for their desired inference using the quantities provided from earlier phases as inputs (T), not necessarily using direct observations of X or Y. Such situations are common in practice, as methods are often built around properties of preprocessed data such as smoothness or sparsity that need not hold for the actual values of X.

For the preprocessor, the input is Y and the output is T. Here T could consist of a vector of means with corresponding standard errors, or, for discrete Y, T could consist of carefully selected cross-tabulations. In general, T clearly needs to be related to X to capture inferential information, but its actual form is influenced by practical constraints (e.g., ag-

gregation to lower than desired resolutions due to data storage capacity).

For the downstream analyst, the input is T and the output is an inference for θ. This analyst can obviously adapt. For example, suppose $\theta = E(X_i)$ for each entry i of X. If the preprocessor provides $T_0 = \hat{X}$, the analyst may simply use an unweighted mean to estimate θ. If the preprocessor instead gives the analyst $T_1 = (\hat{X}, S)$, where S contains standard errors, the latter could instead use a weighted mean to estimate θ. This adaptation extends to an arbitrary number of possible inputs T_k, each of which corresponds to a set of constraints facing the preprocessor.

To formalize this notion of adaptation, we first define an index set C with one entry for each such set of constraints. This maps between forms of input provided by the preprocessor and estimators selected by the downstream analyst. In this way, C captures the downstream analyst's knowledge of previous processing and the underlying probability model. Thus, this index set plays an central role in the definition of multiphase inference problems, far beyond that of a mere mathematical formality; it regulates the amount of mutual knowledge shared between the preprocessor and the downstream analyst.

Now, we turn to the estimators themselves. We start with point estimation as a foundation for a broader class of problems. Testing begins with estimating rejection regions, interval estimation with estimating coverage, classification with estimating class membership, and prediction with estimating future observations and, frequently, intermediate parameters. The framework we present therefore provides tools that can be adapted for more than estimation theory. We define multiphase estimation procedures as follows:

Definition 1. *A* multiphase estimation procedure $\mathcal{P}$ *is a set of estimators* $\{\hat{\theta}_k(T_k) : k \in C\}$ *indexed by the set C, where* T_k *corresponds to the output of the kth first-phase method; that is,* $\mathcal{P}$ *is a family of estimators with different inputs.*

When clear, we will drop the subscripts k and index the estimators in $\mathcal{P}$ by their in-

puts. This definition provides enough flexibility to capture many practical issues with multiphase inference, and it can be iterated to define procedures for analyses involving a longer sequence of preprocessors and analysts. It also encompasses the definition of a missing data procedure used by Meng (1994). Such procedures cannot, of course, be arbitrarily constructed if they are to deliver results with general validity. Hence, having defined these procedures, we will cull many of them from consideration in Section 3.3.3.

The obvious choice of our estimand, suggested by our notation thus far, is the parameter for the scientific model, θ. This is very amenable to mathematical analysis and relevant to many investigations. Hence, it forms the basis for our results in Section 3.4. However, for multiphase analyses, other classes of estimands may prove more useful in practice. In particular, functions of X, future scientific variables X_{rep}, or future observations Y_{rep} may be of interest. Prediction of such quantities is a natural focus in the multiphase setting because such statements are meaningful to both the preprocessor and downstream analyst. Such estimands naturally encompass a broad range of statistical problems including prediction, classification, and clustering. However, there is often a lack of mutual knowledge about $p_X(X \mid \theta)$, so the preprocessor cannot expect to "target" estimation of θ in general, as we shall discuss in Section 3.5.

3.3.3 When is more better?

It is not automatic for multiphase estimation procedures to produce better results as the first phase provides more information. To obtain a sensible context for theoretical development, we must regulate the way that the downstream analyst adapts to different inputs. For instance, they should obtain better results (in some sense) when provided with higher-resolution information. This carries over from the MI setting (Meng, 1994; Meng and Romero, 2003; Meng and Xie, 2013; Xie and Meng, 2012), where notions such as self-efficiency are useful for regulating the downstream analyst's procedures. We define

a similar property for multiphase estimation procedures, but without restricting ourselves to the missing data setting. Specifically, let $T_1 \preceq T_2$ indicate T_1 is a deterministic function of T_2. In practice, T_1 could be a subvector, aggregation, or other summary of T_2.

Definition 2 (Risk monotonicity). *A multiphase estimation procedure $\mathcal{P}$ is* risk monotone *with respect to a loss function L if, for all pairs of outputs T_1, T_2, $T_1 \preceq T_2$ implies $R(\hat{\theta}_2(T_2), L) \leq R(\hat{\theta}_1(T_1), L)$.*

An asymptotic analogue of risk monotonicity is defined as would be expected, scaling the relevant risks at an appropriate rate to obtain nontrivial limits. This is a natural starting point for regulating multiphase estimation procedures; stronger notions may be required for certain theoretical results. Note that this definition does not require that "higher-quality" inputs necessarily lead to lower risk estimators. Risk monotonicity requires only that estimators based upon a larger set of inputs perform no worse than those with strictly less information (in a deterministic sense). However, risk monotonicity is actually quite tight in another sense. It requires that additional information cannot be misused by the downstream analyst, imposing a strong constraint on mutual knowledge. For an example, consider the case of unweighted and weighted means. To obtain better results when presented with standard errors, the downstream analyst must know that they are being given (the correct) standard errors and to weight by inverse variances.

This definition is related to the comparison of experiments, as explored by Blackwell (1951, 1953), but diverges on a fundamental level. Our ordering of experiments, based on deterministic functions, is more stringent than that of Blackwell (1953), but they are related. Indeed, our $\preceq$ relation implies that of Blackwell (1953). In the latter work, an experiment α is defined as more informative than experiment β, denoted $\alpha \supset \beta$, if all losses attainable from β are also attainable from α. This relation is also implied if α is sufficient for β. Our stringency stems from our broader objectives in the multiphase setting. From a

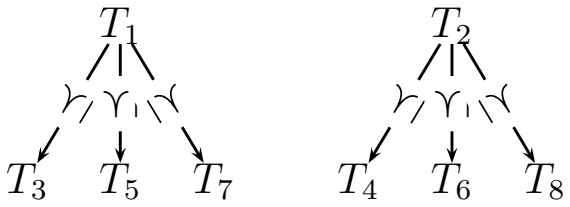

Figure 3.2: Illustration of risk-monotone "basis" construction. In this case, T_1 and T_2 form the basis set of statistics. Each of these has three descendants (T_3, T_5, T_7 from T_1 and T_4, T_6, T_8 from T_2). These descendants are deterministic functions of their parent, but they are not deterministic functions of any other basis statistics. Given correctly-specified models for T_1 and T_2, a risk monotone procedure can be constructed for all statistics $(T_1, \ldots, T_8)$ shown here as described in the text.

decision-theoretic perspective, the partial ordering of experiments investigated by Blackwell and others deal with which risks are attainable given pairs of experiments, allowing for arbitrary decision procedures. In contrast, our criterion restricts procedures based on whether such risks are actually attained, with respect to a particular loss function. This is because, in the multiphase setting, it is not generally realistic to expect downstream analysts to be capable of obtaining optimal estimators for all forms of preprocessing.

The conceptually-simplest way to generate such a procedure is to begin with a complete probability model for $p_Y(Y|\theta)$. Under traditional asymptotic regimes, all procedures consisting of Bayes estimators based upon such a model will (with full knowledge of the transformations involved in each T_k and a fixed prior) be risk monotone. The same is true asymptotically under the same regimes (for squared-error loss) for procedures consisting of MLEs under a fixed model. Under some other asymptotic regimes, however, these principles of estimation do not guarantee risk-monotonicity; we explore this further in Section 3.4.2. But such techniques are not the only way to generate risk monotone procedures from probability models. This is analogous to self-efficiency, which can be achieved by procedures that are neither Bayesian nor MLE (Meng, 1994; Xie and Meng, 2012).

A risk monotone procedure can be generated from any set of probability models for

distinct inputs that "span" the space of possible inputs. Suppose that an analyst has a set of probability models, all correctly specified, for $p_{T_b}(T_b|\theta)$, where b ranges over a subset B of the relevant index set C. We also assume that this analyst has a prior distribution $\pi_b(\theta)$ for each such basis models. These priors need not agree between models; the analyst can build a risk-monotone procedure from an inconsistent set of prior beliefs. Suppose that the inputs $\{T_b : b \in B\}$ are not deterministic functions of each other and all other inputs can be generated as nontrivial deterministic transformations of one of these inputs. Formally, we require $T_b \not\preceq T_c$ for all distinct $b, c \in B$ and, for each $k \in C$ there exists a unique $b \in B$ such that $T_k \preceq T_b$ (each output is uniquely descended from a single T_b), as illustrated in Figure 3.2. This set can form a basis, in a sense, for the given procedure.

Using the given probability models with a single loss function and set of priors (potentially different for each model), the analyst can derive a Bayes rule under each model. For each $b \in B$, we require $\hat{\theta}(T_b)$ to be an appropriate Bayes rule on said model. As $T_k = g_k(T_b)$ for some function g_k, we then have the implied $p_{T_k}(T_k|\theta) = \int_{t:g_k(t)=T_k} p_{T_b}(t|\theta)\mathrm{d}t$, yielding the Bayes rule for estimating θ based on T_k, which is no less risky than $\hat{\theta}(T_b)$. The requirement that each output T_k derives from a unique T_b means that each basis component T_b has a unique line of descendants. Within each line, each descendant is comparable to only a single T_b in the sense of deterministic dependence. Between these lines, such comparisons are not possible. This ensures the overall risk-monotonicity.

Biology provides an illustration of such bases. A wide array of methodological approaches have been used to analyze high-throughput gene expression data. One approach, builds upon order and rank statistics (Geman et al., 2004; Tan et al., 2005; Geman, 2012). Another common approach uses differences in gene expression between conditions or experiments, often aggregating over pathways, replicates, and so forth. Each class of methods is based upon a different form of preprocessing: ranks transformations for the former, normalization and aggregation for the latter. Taking procedures based on rank

statistics and aggregate differences in expression as a basis, we can consider constructing a risk-monotone procedure as above. Thus, the given formulation can bring together apparently disparate methods as a first step in analyzing their multiphase properties.

Such constructions are, unfortunately, not sufficient to generate all possible risk monotone procedures. Obtaining more general conditions and constructions for risk monotone procedures is a topic for further work.

3.3.4 Revisiting our examples and probing our boundaries

By casting the examples in Section 3.2.2 into the formal structure just established, we can clarify the practical role of each mathematical component and see how to map theoretical results into applied guidance. We also provide an example that illustrates the boundaries of the framework's utility, and another that demonstrates its formal limits. These provide perspective on the trade-offs made in formalizing the multiphase inference problem.

The case of microarray preprocessing presented previously fits quite nicely into the model of Section 3.3.1. There, Y corresponds to the observed probe-level intensities, X corresponds to the true expression level for each gene under each condition, and θ corresponds to the parameters governing the organism's patterns of gene expression. In the microarray setting, p_Y would characterize the relationship between expression levels and observed intensities, governed by ξ. These nuisance parameters could include chip-level offsets, properties of any additive background, and the magnitudes of other sources of variation. The assumptions of a Markovian dependence structure and distinct parameters for each part of the model appear quite reasonable in this case, as (1) the observation Y can only (physically) depend upon the sample preparation, experimental protocol, and RNA concentrations in the sample and (2) the distributions p_X and p_Y capture physically distinct portions of the experiment. Background correction, normalization, and the reduction of observations to log-fold changes are common examples of preprocessing $T(Y)$.

As discussed previously, estimands based upon X may be of greater scientific interest than those based upon θ. For instance, we may want to know whether gene expression changed between two treatments in a particular experiment (a statement about X) than whether a parameter regulating the overall patterns of gene expression takes on a particular value.

For the astrophysical example, the fit is similarly tidy. The raw astronomical observations correspond to Y, the true temperature, density, and spectral properties of each part of the dust cloud become X, and the parameters governing the relationship between these quantities (e.g., their correlation) form θ. The p_Y distribution governs the physical observation process, controlled by ξ. This process typically includes the instruments' response to astronomical signals, atmospheric distortions, and other earthbound phenomena. As before, the conditional independence of θ and Y given X and ξ is sensible based upon the problem structure, as is the separation of θ and ξ. Here X corresponds to signals emitted billions or trillions of miles from Earth, whereas the observation process occurs within ground- or space-based telescopes. Hence, any non-Markovian effects are quite implausible. Preprocessing $T(Y)$ corresponds to the (point) estimates of temperature, density, and spectral properties from simple models of Y given X and ξ.

The multiphase framework encompasses a broad range of settings, but it does not shed additional light on all of them. If T is a many-to-one transformation of Y, then our framework implies that the preprocessor and downstream analyst face structurally different inference (and missing data) problems. This is the essence of multiphase inference, in our view. Settings where $p_Y(Y|X,\xi)$ is degenerate or T is a one-to-one function of Y are boundary cases where our multiphase interpretation and framework add little.

For a concrete example of these cases, consider a time-to-failure experiment, with the times of failure $W_i \sim$ i.i.d Expo(θ), $i = 1, \ldots, n$. Now, suppose that the experimenters actually ran the experiment in m equally-sized batches. They observe each batch only until its first failure; that is, they observe and report $Y_b = \min\{W_i : i \text{ in batch } b\}$ for each batch

b. Subsequent analysts have access only to $T = (Y_1, \ldots, Y_b)$. This seems to be a case of preprocessing, but it actually resides at the very edge of our framework.

We could take the complete observations to be X and the batch minima to be Y. This would satisfy our Markov constraint, with a singular, and hence deterministic, observation process $p_Y(Y|X)$ simply selecting a particular order statistic within each batch. However, $T(Y)$ is one-to-one; the preprocessor observes only the order statistics, as does the downstream analyst. There is no separation of inference between phases; the same quantities are observed and missing to both the preprocessor and the downstream analyst. Squeezing this case into the multiphase framework is technically valid but unproductive.

The framework we present is not, however, completely generic. Consider a chemical experiment involving a set of reactions. The underlying parameters θ describe the chemical properties driving the reactions, X are the actual states of the reaction, and Y are the (indirectly) measured outputs of the reactions. The measurement process for these experiments, as described by $p_Y(Y|X, \xi)$, could easily violate the structure of our model in this case. For instance, the same chemical parameters could affect both the measurement and reaction processes, violating the assumed separation of θ and ξ.

Even careful preprocessing in such a setting can create a fundamental incoherence. Suppose the downstream analysis will be Bayesian, so the preprocessor provides the conditional density of Y as a function of X, $p_Y(Y|X)$, for the observed Y. If θ and ξ share components, and the preprocessor uses their prior on ξ to create $p_Y(Y|X)$, the conditional density need not be sufficient for θ under the downstream analyst's model. Because the downstream analyst's prior on θ need not be compatible with the preprocessor's prior on ξ, inferences based on the preprocessor's $p_Y(Y|X)$ can be seriously flawed in this setting. Hence, we exclude such cases from our investigation for the time being.

Thinking Bayesianly, our model (3.1) obviously does not exclude the possibility that the downstream analyst has more knowledge about θ than the preprocessor in the form

of a prior on θ. However, *prior* information means that it is based on studies that do not overlap with the current one. Probabilistically speaking, this means that our model permits the downstream analyst to formally incorporate another data set Z, as long as Z is conditionally independent of the scientific variables X and observations Y given (θ, ξ) or θ. For example, the downstream analyst could observe completely separate experiments pertaining to the same underlying process governed by θ or the outcomes of separate calibration pertaining to ξ, but not additional replicates governed by the same realization of X. In a biological setting, this means that the downstream analyst could have access to results from samples not available to the preprocessor (e.g., biological replicates), possibly using the same equipment; however, they could not have access to additional analyses of the same biological sample (e.g., technical replicates), as a single biological sample would typically correspond to a single realization of X.

These examples remind us that our multiphase setting does not encompass all of statistical inference. This is quite a relief to us. Our work aims to open new directions for statistical research, but it cannot possibly address every problem under the sun!

3.3.5 Constraints will set your theory free

Multiphase theory hinges on procedural constraints. Consider, for example, finding the optimal multiphase estimation procedure in terms of the final estimator's Bayes risk. Without stringent procedural constraints, the result is trivial: compute the appropriate Bayes estimator using the distribution of T given θ. Similarly, the optimal preprocessing T will, without tight constraints, simply compute an optimal estimator using Y and pass it forward. Note that both of these cases respect risk-monotonicity to the letter; it is not sufficiently tight to enable interesting, relevant theory. More constraints, based upon careful consideration of applied problems, are clearly required.

This is not altogether bad news. We need only look to the history of multiple imputation

to see how rich theory can arise from stringent, pragmatic constraints. Multiple imputation forms a narrow subset of multiphase procedures: X corresponds to the complete data (Y_{com}, in MI notation), Y corresponds to the observed data Y_{obs} and missing data indicator R, and T usually consists of posterior predictive draws of the missing data together with the observed data. The Markovian property depicted in Figure 3.1 holds when the parameter (ξ) for the missing data mechanism $p(R|Y_{com}, \xi)$ is distinct from the parameter of interest (θ) in $p(Y_{com}|\theta)$, which is a common assumption in practice. The second-phase procedure is then restricted to repeatedly applying a complete-data procedure and combining the results. These constraints were originally imposed for practical reasons—in particular, to make the resulting procedure feasible with existing software. However, they have opened the door to deep theoretical investigations.

In that spirit, we consider two types of practically-motivated constraints for multiphase inference: restrictions on the downstream analyst's procedure and restrictions on the preprocessor's methods. These constraints are intended to work in concert with coherence conditions (e.g., risk monotonicity), not in isolation, to enable meaningful theory.

Constraints on the downstream analyst are intended to reflect practical limitations of their analytic capacity. Examples include restricting the downstream analyst to narrow classes of estimators (e.g., linear functions of preprocessed inputs), to specific principles of estimation (e.g., MLEs), or to special cases of a method we can reasonably assume the downstream analyst could handle, such as a complete-data estimator $\hat{\theta}(X)$, available from software with appropriate inputs. Estimators derived from nested families of models are often suitable for this purpose. For example, whereas $\hat{\theta}(X)$ may involve only an ordinary regression, the computation of $\hat{\theta}(T)$ may require a weighted least-squares regression.

Another constraint on the downstream analyst pertains to nuisance parameters. Such constraints are of great practical and theoretical interest, as we believe that the preprocessor will typically have better knowledge and statistical resources available to address

nuisance parameters than the downstream analyst. An extreme but realistic case of this is to assume that the downstream analyst cannot address nuisance parameters at all. As we shall discuss in Section 3.4, this would force the preprocessor to either marginalize over the nuisance parameters, find a pivot with respect to them, or trust the downstream analyst to use a method robust to the problematic parameters.

Turning to the preprocessor, we consider restricting either the form of the preprocessor's output or the mechanics of their methods. In the simplest case of the former, we could require that T consist of the posterior mean ($\hat{X}$) and posterior covariance (V) of the unknown X under the preprocessor's model. A richer, but still realistic, class of output would be finite-dimensional real or integer vectors. Restricting output to such a class would prevent the preprocessor from passing arbitrary functions onto the downstream analyst. This leads naturally to the investigation of (finite-dimensional) approximations to the preprocessor's conditional density, aggregation, and other such techniques.

On the mechanical side, we can restrict either the particulars of the preprocessor's methods or their broader properties. Examples of the former include particular computational approximations to the likelihood function or restrictions to particular principles of inference (e.g., summaries of the likelihood or posterior distribution of X given (Y, ξ)). Such can focus our inquiries to specific, feasible methods of interest or reflect the core statistical principles we believe the preprocessor should take into account. In a different vein, we can require that preprocessor's procedures be distributable across multiple researchers, each with their own experiments and scientific variables of interest. Such settings are of interest for both the accumulation of scientific results for later use and for the development of distributed statistical computation. This leads to preprocessing based upon factored "working" models for X, as we explore further in Section 3.4.1. Nuisance parameters play an important role in these constraints, narrowing the class of feasible methods (e.g., marginalization over such parameters may be exceedingly difficult) and

largely determining the extent to which preprocessing can be distributed. We explore these issues in more detail throughout Section 3.4.

3.4 A few theoretical cornerstones

We now present a few steps towards a theory of multiphase inference. In this, we endeavor to address three basic questions: (1) how can we determine what to retain, (2) what limits the performance of multiphase procedures, and (3) what are some minimal requirements for being an ideal preprocessor? We find insight into the first question from the language of classical sufficiency. We leverage and specialize results from the missing-data literature to address the second. For the third question, we turn to the tools of decision theory.

3.4.1 Determining what to retain

Suppose we have a group of researchers, each with their own experiments. They want to preprocess their data to reduce storage requirements, ease subsequent analyses, and (potentially) provide robustness to measurement errors. This group is keenly aware of the perils of preprocessing and want to ensure that the output they provide will be maximally useful for later analyses. Their question is, "Which statistics should we retain?"

If each of these researchers was conducting the final analysis themselves, using only their own data, they would be in a single-phase setting. The optimal strategy then is to keep a minimal sufficient statistic for each researcher's model. Similarly, if the final analysis were planned and agreed upon among all researchers, we would again have a single-phase setting, and it is optimal to retain the sufficient statistics for the agreed-upon model. We use the term *optimal* here because it achieves maximal data reduction without losing information about the parameters of interest. Such lossless compression–in the general sense of avoiding statistical redundancy–is often impractical, but it provides

a useful theoretical gold standard.

In the multiphase setting, especially with multiple researchers in the first phases, achieving optimal preprocessing is far more complicated even in theory. If $T(Y)$ is the output of the *entire* preprocessing phase, then in order to retain all information we must require $T(Y)$ to be a sufficient statistics for $\{\theta, \xi\}$ under model (3.1); that is,

$$L(\theta, \xi | T(Y)) = L(\theta, \xi | Y), \tag{3.2}$$

where L denotes a likelihood function; or at least in the (marginal) Bayesian sense,

$$P(\theta | T(Y)) = P(\theta | Y), \tag{3.3}$$

where $P(\theta|D)$ is the posterior of θ given data D with the likelihood given by (3.1). Note that (3.2) implies (3.3), and (3.3) is useful when the downstream analyst wants only a Bayesian inference of θ. In either case the construction of the sufficient statistic generally depends on the joint model for Y as implied by (3.1), requiring more knowledge than individual researchers typically possess.

Often, however, it is reasonable to assume the following conditional independence. Let $\{Y_i, X_i, \xi_i\}$ be the specification of $\{Y, X, \xi\}$ for researcher i $(= 1, \ldots, r)$, where $\{Y_1, \ldots, Y_r\}$ forms a *partition* of Y. We then assume that

$$p_Y(Y|X, \xi) = \prod_{i=1}^{r} p_{Y_i}(Y_i | X_i, \xi_i) \, . \tag{3.4}$$

Note in the above definition implicitly we also assume the baseline measure μ_Y is a product measure $\prod_{i=1}^{r} \mu_{Y_i}$, such as Lebesgue measure. The assumption (3.4) holds, for example, in microarray applications, when different labs provide conditionally-independent observations of probe-level intensities. The preceding discussion suggests that this assumption is

necessary for ensuring (3.2) or even (3.3), but obviously it is far from sufficient because it says nothing about the model on X.

It is reasonable—or at least more logical than not—to assume each researcher has the best knowledge to specify his/her own observation model $p_{Y_i}(Y_i|X_i,\xi_i)$ $(i=1,\ldots,r)$. But, for the scientific model $p_X(X|\theta)$ used by the downstream analyst, the best we can hope is that each researcher has *a working model* $\tilde{p}_X(X_i|g_i(\eta))$ that is in some way related to $p_X(X|\theta)$. The notation $g_i(\eta)$ reflects our hope to construct a common working parameter η that can ultimately be *linked* to the scientific parameter θ.

Given this working model, the ith researcher can obtain the corresponding (minimal) sufficient statistic $T_i(Y)$ for $\{g_i(\eta),\xi_i\}$ with respect to

$$\tilde{p}_X(Y_i|g_i(\eta),\xi_i)=\int p_Y(Y_i|X_i,\xi_i)\tilde{p}_X(X_i|g_i(\eta))\mathrm{d}\mu_{X_i}(X_i),\quad i=1,\ldots,r. \tag{3.5}$$

When one has a prior $\pi_{\xi_i}(\xi_i)$ for ξ_i, one could alternately decide to retain the (Bayesian) sufficient statistic $T_i^B(Y_i)$ with respect to the model

$$\tilde{p}_Y(Y_i|g_i(\eta))=\int\int p_Y(Y_i|X_i,\xi_i)\tilde{p}_X(X_i|g_i(\eta))\pi_{\xi_i}(\xi_i)\mathrm{d}\mu_{X_i}(X_i)\mathrm{d}\mu_{\xi}(\xi_i)\,. \tag{3.6}$$

Our central interest here is to determine when the collection $T(Y)=\{T_i(Y_i):i=1,\ldots,r\}$ will satisfy (3.2) and when $T^B(Y)=\{T_i^B(Y_i):i=1,\ldots,r\}$ will satisfy (3.3). This turns out to be an exceedingly difficult problem if we seek a necessary and sufficient condition for *when* this occurs. However, it is not difficult to identify sufficient conditions that can provide useful practical guidelines. We proceed by first considering cases where $\{X_1,\ldots,X_r\}$ forms a partition of X. Compared to the assumption on partitioning Y, this assumption is less likely to hold in practice because different researchers can share common parts of X's or even the entire scientific variable X. However, as we shall demonstrate

shortly, we can extend our results formally to all models for X, as long as we are willing to put tight restrictions on the allowed class of working models. Specifically, the following condition describes a class of working models that are ideal because they permit separate preprocessing yet retain joint information. Note again that an implicit assumption here is that the baseline measure μ_X is a product measure $\prod_{i=1}^{r} \mu_{X_i}$.

Definition 3 (Distributed Separability Condition (DSC)). *A set of working models* $\{\tilde{p}_X(X_i|g_i(\eta)) : i = 1, \ldots, r\}$ *is said to satisfy the* distributed separability condition *with respect to* $p_X(X|\theta)$ *if there exists a probability measure* $p_\eta(\eta|\theta)$ *such that*

$$p_X(X|\theta) = \int_\eta \left[\prod_{i=1}^{r} \tilde{p}_X(X_i|g_i(\eta)) \right] \mathrm{d}p_\eta(\eta|\theta). \tag{3.7}$$

Theorem 2. *Under the assumptions (3.4) and (3.7), we have*

(1) The collection of individual sufficient statistics from (3.5), that is, $T(Y) = \{T_i(Y_i), i = 1, \ldots, r\}$*, is jointly sufficient for* $\{\theta, \xi\}$ *in the sense that (3.2) holds.*

(2) Under the additional assumption that $\{\xi_1, \ldots, \xi_r\}$ *forms a partition of* ξ *and* $\pi(\xi)\mathrm{d}\mu_\xi = \prod_{i=1}^{r} \pi_{\xi_i}(\xi_i)\mathrm{d}\mu_{\xi_i}$*, both* $T(Y)$ *corresponding to (3.5) and* $T^B(Y)$ *corresponding to (3.6) are Bayesianly sufficient for* θ *in the sense that (3.3) holds.*

Proof. By the sufficiency of T_i for $(g_i(\eta), \xi_i)$, we can write

$$\int_{X_i} p_Y(Y_i|X_i, \xi_i)\tilde{p}_X(X_i|g_i(\eta))\mathrm{d}\mu_{X_i}(X_i) = \tilde{p}_Y(Y_i|g_i(\eta), \xi_i) = h_i(Y_i)f_i(T_i; g_i(\eta), \xi_i). \tag{3.8}$$

This implies that,

$$
\begin{aligned}
p_Y(Y|\theta,\xi) &= \int_X p_Y(Y|X,\xi)p_X(X|\theta)\mathrm{d}\mu_X(X) \\
[\text{by (3.4) and (3.7)}] &= \int_X \left[\prod_{i=1}^{r} p_Y(Y_i|X_i,\xi_i)\right] \cdot \\
&\qquad \left[\int_\eta \left[\prod_{i=1}^{r} \tilde{p}_X(X_i|g_i(\eta))\right] \mathrm{d}p_\eta(\eta|\theta)\right] \mathrm{d}\mu_X(X) \\
[\text{by factorization of } \mu_X] &= \int_\eta \prod_{i=1}^{r} \left[\int_{X_i} p_Y(Y_i|X_i,\xi_i)\tilde{p}_X(X_i|g_i(\eta))\mathrm{d}\mu_{X_i}(X_i)\right] \mathrm{d}p_\eta(\eta|\theta) \\
[\text{by (3.8)}] &= \left[\prod_{i=1}^{r} h_i(Y_i)\right] \left[\int_\eta \prod_{i=1}^{r} f_i(T_i;g_i(\eta),\xi_i)\mathrm{d}p_\eta(\eta|\theta)\right].
\end{aligned}
$$

This establishes (1) by the factorization theorem. Assertion (2) is easily established via an analogous argument, by integrating all the expressions above with respect to $\pi(\xi)\mathrm{d}\mu(\xi) = \prod_{i=1}^{r} \pi_{\xi_i}(\xi_i)\mathrm{d}\mu_{\xi_i}(\xi_i)$. □

We emphasize that DSC does not require individual researchers to model their parts of X in the same way as the downstream analyst would, which would make it an essentially tautological condition. Rather, it requires that individual researchers understand their own problems and how they can fit into the broader analysis hierarchically. This means that the working model for each X_i $(i = 1,\ldots,r)$ can be more saturated than the downstream analyst's model for the same part of X.

Consider a simple case with $r = 1$, where the preprocessor correctly assumes the multivariate normality for X but is unaware that its covariance actually has a block structure or is unwilling to impose such a restriction to allow for more flexible downstream analyses. Clearly any sufficient statistic under the unstructured multivariate model is also sufficient for any (nested) structured ones. The price paid here is failing to achieve the greatest possible sufficient reduction of the data, but this sacrifice may be necessary to ensure

the broader validity of downstream analyses. For example, even if downstream analysts adopt a block-structured covariance, they may still want to perform a model checking, which would not be possible if all they are given is a *minimal* sufficient statistic for the model to be checked.

Knowledge suitable for specifying a saturated model is more attainable than complete knowledge of $p_X(X|\theta)$, although ensuring common knowledge of its (potential) hierarchical structure still requires some coordination among the researchers. Each of them could independently determine for which classes of scientific models their working model satisfies the DSC. However, without knowledge of the partition of X across researchers and the overarching model(s) of interest, their evaluations need not provide any useful consensus. This suggests the necessity of some general communications and a practical guideline for distributed preprocessing, even when we have chosen a wise division of labors that permits DSC to hold.

Formally, DSC is similar in flavor to de Finetti's theorem, but it does not require the components of the factorized working model to be exchangeable. DSC, however, is by no means necessary (even under (3.2)), as an example in Section 3.4.4 will demonstrate. Its limits stem from "unparameterized" dependence—dependence between X_i's that is not controlled by θ. When such dependence is present, statistics can exist that are sufficient for both η and θ without the working model satisfying DSC.

However, a simple necessary condition for distributed sufficiency is available. Unsurprisingly, it links the joint sufficiency of $T(Y) = \{T_i(Y_i) : i = 1, \ldots, r\}$ under $p_Y(Y|\theta)$ to the joint sufficiency of $S(X) = \{S_i(X_i), i = 1, \ldots, r\}$ under the scientific model $p_X(X|\theta)$, where $S_i(X_i)$ is any sufficient statistic for the working model $\tilde{p}_X(X_i|g_i(\eta)), i = 1, \ldots, r$.

Theorem 3. *If, for all observation models satisfying (3.4), the collections of individual sufficient statistics from (3.5) $T(Y) = \{T_i(Y_i), i = 1, \ldots, r\}$ are jointly sufficient for $\{\theta, \xi\}$ in the sense*

that (3.2) holds, then any collection of individual sufficient statistics under $\{\tilde{p}_X(X_i|g_i(\eta)), i = 1, \ldots, r\}$*, that is,* $S(X)$*, must be sufficient for* θ *under* $p_X(X|\theta)$*.*

The proof of this condition emerges easily by considering the trivial observation model $p_Y(Y_i|X_i, \xi) = \delta_{\{Y_i = X_i\}}$, where δ_A is the indicator function of set A. Theorem 3 holds even if we require the observation model to be nontrivial, as the case of $p_Y(Y_i|X_i, \xi) \propto \delta_{\{Y_i \in B_\varepsilon[X_i]\}}$ for arbitrary ε-neighborhoods of X_i demonstrates. The result says that if we want distributed preprocessing to provide a lossless compression regardless of the actual form of the observation model, then even under the conditional independence assumption (3.4), we must require the individual working models to *collectively* preserve sufficiency under the scientific model. Note that preserving sufficiency for a model is a much weaker requirement than preserving the model itself. Indeed, two models can have very different model spaces yet share the same *form* of sufficient statistics, as seen with i.i.d. Poisson(μ) and $N(\mu, 1)$ models, both yielding the sample average as a complete sufficient statistic.

Although we find this sufficiency-preserving condition quite informative about the limits of lossless distributed preprocessing, it is not a sufficient condition. As a counterexample, consider $Y_{ij}|X_i \sim N(\mu_i, \sigma_{ij}^2)$ independent for $i = 1, \ldots, n$, $j = 1, \ldots, m$, where $X_i \equiv (\mu_i, \sigma_{i1}^2, \ldots, \sigma_{im}^2)$. For the true model, we assume $p_X(X|\theta)$ as follows: $\mu_i|\theta \sim N(\theta, 1)$, $\sigma_{ij}^2 \sim 1/{\chi_1}^2$, and all variables are mutually independent. For the working model, we take $\tilde{p}_X(X|\eta)$ as follows: $\mu_i|\eta_i \sim N(\eta_i, 1)$ independently, and $\sigma_{ij}^2 = 1$ with probability 1 for all i, j. Obviously $S = (\mu_1, \ldots, \mu_n)$ is a sufficient statistic for both $\tilde{p}_X(X|\eta)$ and $p_X(X|\theta)$ because of their normality. Because S is *minimally* sufficient for η, this implies that any sufficient statistic for $\tilde{p}_X(X|\eta)$ must be sufficient for $p_X(X|\theta)$, therefore the sufficiency preserving condition holds.

However, the collection of the complete sufficient statistics $T_i = \sum_j y_{ij}/m, i = 1, \ldots, r$ for η under $p_Y(Y|\eta)$ is not sufficient for θ under $p_Y(Y|\theta)$ because the latter is no longer an

exponential family. The trouble is caused by the failure of the working models to capture additional flexibility in the scientific model that is not controlled by its parameter θ. Therefore, obtaining a condition that is both necessary and sufficient for lossless compression via distributed preprocessing is a challenging task. Such a condition appears substantially more intricate than those presented in Theorems 2 and 3 and may therefore be less useful as an applied guideline. Below we discuss a few further subtleties.

Likelihood sufficiency verses Bayesian efficiency. Although Theorem 2 covers both likelihood and Bayesian cases, it is important to note a subtle distinction between their general implications. In the likelihood setting (3.2), we achieve lossless compression for all downstream analyses targeting (θ, ξ). This allows the downstream analyst to obtain inferences that are robust to the preprocessor's beliefs about ξ, and they are free to revise their inferences if new information about ξ becomes available. But, the downstream analyst must address the nuisance parameter ξ from the preprocessing step, a task a downstream analyst may not be able or willing to handle.

In contrast, the downstream analyst need not worry about ξ in the Bayesian setting (3.3). However, this is archived at the cost of robustness. All downstream analyses are potentially affected by the preprocessors' beliefs about ξ. Furthermore, because $T^B(Y)$ is required only to be sufficient for θ, it may not carry any information for a downstream analyst to check the preprocessor's assumptions about ξ. Fortunately, as it is generally logical to expect the preprocessor to have better knowledge addressing ξ than the downstream analyst, such robustness may not be a serious concern from a practical perspective. Theoretically, the trade-off between robustness and convenience is not clear-cut; they can coincide for other types of preprocessing, as seen in Section 3.4.2 below.

Deterministic dependencies among X_i's. As discussed earlier, (conditional) dependencies among the observation variables Y_i across different i's will generally rule out the

possibility of achieving lossless compression by collecting individual sufficient statistics. This points to the importance of appropriate separation of labors when designing distributed preprocessing. In contrast, dependencies among X_i's are permitted, at the expense of redundancy in sufficient statistics. We first consider deterministic dependencies, and for simplicity, take $r = 2$ and constrain attention to the case of sufficiency for θ. Suppose we have X_1 and X_2 forming a partition of X, with a working model $\tilde{p}_X(X|\vec{\eta}) = \tilde{p}_{X_1}(X_1|\eta_1)\tilde{p}_{X_2}(X_2|\eta_2)$ that satisfied the DSC for some $p_\eta(\eta|\theta)$. Imagine we need to add a common variable Z to both X_1 and X_2 that is conditionally independent of $\{X_1, X_2\}$ given θ and has density $p_Z(Z|\theta)$, with the remaining model unchanged. However, the two researchers are unaware of the sharing of Z, so they set up $X_1' = \{X_1, Z_1\}$ and $X_2' = \{X_2, Z_2\}$, with $p_{X_1'}(X_1'|\eta_1')$ and $\tilde{p}_{X_2'}(X_2'|\eta_2')$ as their respective working models.

At the first sight this seems to be a hopeless situation for applying the DSC condition, because $X' = \{X_1', X_2'\} = \{X_1, Z_1, X_2, Z_2\}$ does not correspond to the scientific variable $X = \{X_1, X_2, Z\}$ of interest. However, we notice that if we can force $Z_1 = Z_2 = Z$ in X', then we can recover X. This forcing is not a mere mathematical trick. Rather, it reflects an extreme yet practical strategy when researchers are unsure whether they share some components of their X_i's with others. The strategy is simply to retain statistics sufficient for the entire part that they may *suspect* to be common, which in this case means that both researchers will retain statistics sufficient for the Z_i's $(i = 1, 2)$ in their entirety. Mathematically, this corresponds to letting $\tilde{p}_{X_i'}(X_i'|\eta_i') = \tilde{p}_{X_i}(X_i|\eta_i)\delta_{\{Z_i=\zeta_i\}}$, where $\eta_i' = \{\eta_i, \zeta_i\}$. It is then easy to verify that DSC holds, if we take $p_\eta'(\eta'|\theta) = p_\eta(\eta|\theta)p_Z(\zeta_1|\theta)\delta_{\{\zeta_1=\zeta_2\}}$, where $\eta' = \{\eta, \zeta_1, \zeta_2\}$. This is because when $Z_1 \neq Z_2$, both sides of (3.7) are zero. When $Z_1 = Z_2 = Z$, we have

(adopting integration over δ functions)

$$\begin{aligned}\int_{\eta'}\left[\prod_{i=1}^{2}\tilde{p}_{X_i'}(X_i'|\eta_i')\right]\mathrm{d}p_\eta'(\eta'|\theta) &= \int_\eta\int_{\zeta_1}\left[\prod_{i=1}^{2}\tilde{p}_{X_i}(X_i|\eta_i)\delta_{\{Z=\zeta_i\}}\right]\mathrm{d}p_\eta(\eta|\theta)\delta_{\{\zeta_1=\zeta_2\}}\mathrm{d}p_Z(\zeta_1|\theta)\\ &= \left[\int_\eta\prod_{i=1}^{2}\tilde{p}_{X_i}(X_i|\eta_i)\mathrm{d}p_\eta(\eta|\theta)\right]\int_{\zeta_1}\delta_{\{\zeta_1=Z\}}\mathrm{d}p_Z(\zeta_1|\theta)\\ &= p_X(X_1,X_2|\theta)p_Z(Z|\theta) = p_X(X|\theta).\end{aligned}$$

This technique of expanding η to include shared parts of the X allows the DSC and Theorem 2 to be applied to all models $p_X(X|\theta)$, not only those with with distinct X_i's. However, this construction also restricts working models to those with deterministic relationships between parts of η and each X_i.

The derivation above demonstrates both the broader applications of DSC as a theoretical condition and its restrictive nature as a practical guideline. Retaining sufficient statistics for both Z_1 and Z_2 can create redundancy. If each preprocessor observes Z without noise, then only one of them actually needs to retain and report their observation of Z. However, if each observes Z with independent noise, then both of their observations are required to obtain a sufficient statistic for θ. The noise-free case also provides a straightforward counterexample to the necessity of DSC. Assuming both preprocessors observe Z directly, as long as one of the copies of Z is retained via the use of the saturated δ density, the other copy can be modeled in any way— and hence can be made to violate DSC— without affecting their joint sufficiency for θ.

Regardless of the dependencies among the X_i's, there is always a safe option open to the preprocessors for data reduction: retain T_i sufficient for (X_i, ξ_i) under $p_Y(Y_i|X_i, \xi_i)$. This will preserve sufficiency for θ under any scientific model $p_X(X|\theta)$:

Theorem 4. *If $p_Y(Y|X,\xi)$ is correctly specified and satisfies (3.4), then any collection of individual sufficient statistics $\{T_i : i = 1,\ldots,r\}$ with each T_i sufficient for (X_i,ξ_i) is jointly sufficient*

for (θ, ξ) in the sense of (3.2) for all $p_X(X|\theta)$.

Proof. By the factorization theorem, we have $p_Y(Y_i|X_i, \xi_i) = h_i(Y_i)f_i(T_i; X_i, \xi_i)$ for any i. Hence, by (3.4), $p_Y(Y|\theta) = [\prod_{i=1}^r h_i(Y_i)] \int_X [\prod_{i=1}^r p_T(T_i|X_i, \xi_i)] \, p_X(X|\theta) \mathrm{d}\mu_X(X)$. Therefore $\{T_i : i = 1, \ldots, r\}$ is sufficient for θ, by the factorization theorem for sufficiency. □

Theorem 4 provides a universal, safe strategy for sufficient preprocessing and a lower bound on the compression attainable from distributed sufficient preprocessing. As all minimal sufficient statistics for θ are functions of any sufficient statistic for (X, ξ), retaining minimal sufficient statistics for each (X_i, ξ_i) results in less compression than any approach properly using knowledge of $p_X(X|\theta)$. However, the compression achieved relative to retaining Y itself may still be significant. Minimal sufficient statistics for θ provide an upper bound on the attainable degree of compression by the same argument. Achieving this compression generally requires that each preprocessor knows the true scientific model $p_X(X|\theta)$. Between these bounds, the DSC (3.7) shows a trade-off between the generality of preprocessing (with respect to different scientific models) and the compression achieved: the smaller the set of scientific models for which a given working model satisfies (3.7), the greater the potential compression from its sufficient statistics.

Stochastic dependencies among X_i's. More generally, stochastic dependence among X_i's reduces compression and increases redundancy in distributed preprocessing. These costs are particularly acute when elements of θ control dependence among X_i's, as seen in the

following example where

$$X = (X_1, X_2)^\top \sim N_{4D}\left(\theta_1 \vec{1}_{4D}, \begin{pmatrix} 1 & 0 & 0 & \theta_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \theta_2 & 0 & 0 & 1 \end{pmatrix} \otimes I_D\right) \text{ for } D > 1,$$

$$Y_i = (Y_{i1}, Y_{i2})^\top \mid X_i \sim N_{2D}(X_i, I_{2D}) \text{ independently for } i = 1, 2.$$

Here $\vec{1}_{4D}$ is a column vector with $4D$ 1's as its components, and $\otimes$ is the usual Kronecker product. If θ_2 is known, then each researcher can reduce their observations Y_i to a scalar statistic $Y_i^\top \vec{1}_{2D}$ and preserve sufficiency for θ_1. If θ_2 is unknown, then each researcher must retain all of Y_{ii} (but not Y_{ij} for $i \neq j$) in addition to these sums to ensure sufficiency for $\theta = (\theta_1, \theta_2)$, because the minimal sufficient statistic for (θ_1, θ_2) requires the computation of $Y_{11}^\top Y_{22}$. Thus, the cost of dependence here is D additional pieces of information per preprocessor. Dependence among the X_i's forces the preprocessors to retain enough information to properly combine their individual contributions in the final analysis, down-weighting redundant information. This is true even if they are interested only in efficient estimation of θ_1, leading to less reduction of their raw data and less compression from preprocessing than the independent case.

Practical perspective. From this investigation, we see that it is generally not enough for each researcher involved in preprocessing to reduce data based on even a correctly-specified model for their problem at hand. We instead need to look to other models that include each experimenter's data hierarchically, explicitly considering higher-level structure and relationships. However, significant reductions of the data are still possible despite these limitations. Each T_i need not be sufficient for each X_i, nor must T be sufficient for X overall. This often implies that much less data need to be retained and shared than

retaining sufficient statistics for each X_i would demand. For instance, if a working model with $X_i|\eta_i \sim N(\mu_i, \Sigma_i)$ satisfies the DSC for a given model $p_X(X|\theta)$ and $Y_{ij}|X_i, \xi_i \sim N(X_i, \xi_i)$, then only means and covariance matrices of Y_{ij} within each experiment i need to be retained.

The discussions above demonstrate the importance of involving downstream analysts in the design of preprocessing techniques. Their knowledge of $p_X(X|\theta)$ is extremely useful in determining what compression is appropriate, even if said knowledge is imperfect. Constraining the scientific model to a broad class may be enough to guarantee effective preprocessing. For example, suppose we fix a working model and consider all scientific models that can be expressed as (3.7) by varying the choices of $p_\eta(\eta|\theta)$. This yields a very broad class of hierarchical scientific models for downstream analysts to evaluate, while permitting effective distributed preprocessing based on the given working model.

Practically, we see two paths to distributed preprocessing: coordination and caution. Coordination refers to the downstream analyst evaluating and guiding the design of preprocessing as needed. Such guidance can guarantee that preprocessed outputs will be as compact and useful as possible. However, it is not always feasible. It may be possible to specify preprocessing in detail in some industrial and purely computational settings. Accomplishing the same in academic research or for any research conducted over time is an impractical goal. Without such overall coordination, caution is needed. It is not generally possible to maintain sufficiency for θ without knowledge of the possible models $p_X(X|\theta)$ unless the retained summaries are sufficient for X itself. Preprocessors should therefore proceed cautiously, carefully considering which scientific models they effectively exclude through their preprocessing choices. This is analogous to the oft-repeated guidance to include as many covariates and interactions as possible in imputation models (Meng, 1994; Meng and Romero, 2003).

3.4.2 Doing the best with what you get

Having considered the lossless preprocessing, we now turn to more realistic but less clear-cut situations. We consider a less careful preprocessor and a sophisticated downstream analyst. The preprocessor selects an output T, which may discard much information in Y but nevertheless preserves the identifiability of θ, and the downstream analyst knows enough to make the best of whatever output they are given. That is, the index set C completely and accurately captures all relevant preprocessing methods $T = \{T_i : i = 1, \ldots, r\}$. This does not completely capture all the practical constraints discussed in Section 3.3. However, it is important to establish an upper bound on multiphase performance before incorporating such issues. This upper bound is on the Fisher information, and hence a lower bound on the asymptotic variances of estimators $\hat{\theta}$ of θ. As we will see, nuisance parameters (ξ) play a crucial role in these investigations.

When using a lossy compression, an obvious question is how much information is lost compared to a lossless compression. This question has a standard asymptotic answer when the downstream analyst adopts an MLE or Bayes estimator, so long as nuisance parameters behave appropriately (as will be discussed shortly). If the downstream analyst adopts some other procedures, such as an estimating equation, then there is no guarantee that the procedure based on Y is more efficient than the one based on T. That is, one can actually obtain a more efficient estimator with less data when one is not using *probabilistically principled* methods, as discussed in detail in Meng and Xie (2013).

Therefore, as a first step in our theoretical investigations, we will focus on MLEs; the results also apply to Bayesian estimators under the usual regularity conditions to guarantee the asymptotic equivalence between MLEs and Bayesian estimators. Specifically, let $(\hat{\theta}(Y), \hat{\xi}(Y))$ and $(\hat{\theta}(T), \hat{\xi}(T))$ be the MLEs of (θ, ξ) based respectively on Y and T under model (3.1). We place standard regularity conditions for the joint likelihood of (θ, ξ),

assuming bounded third derivatives of the log-likelihood, common supports of the observation distributions with respect to (θ, ξ), full rank for all information matrices at the true parameter value (θ_o, ξ_o), and the existence of an open subset of the parameter space that contains (θ_o, ξ_o). These conditions imply the first and second Bartlett identities.

However, the most crucial assumption here is a sufficient accumulation of information, indexed by an *information size* N_Y, to constrain the behavior of remainder terms in quadratic approximations of the relevant score functions. Independent identically distributed observations and fixed-dimensional parameters would satisfy this requirement, in which case N_Y is simply the data size of Y, but weaker conditions can suffice (for an overview, see Lehmann and Casella, 1998). In general, this assumption requires that the dimension of both θ and ξ are bounded as we accumulate more data, preventing the type of phenomenon revealed in Neyman and Scott (1948). For multiphase inferences, cases where these dimensions are unbounded are common (at least in theory) and represent interesting settings where preprocessing can actually improve asymptotic efficiency, as we discuss shortly.

To eliminate the nuisance parameter ξ, we work with the observed Fisher information matrices based on the profile likelihoods for θ, denoted by I_Y and I_T respectively. Let F be the limit of $I_Y^{-1}(I_Y - I_T)$, the so-called *fraction of missing information* (see Meng and Rubin, 1991), as $N_Y \to \infty$. The proof of the following result follows the standard asymptotic arguments for MLEs, with the small twist of applying them to profile likelihoods instead of full likelihoods. (We can also invoke the more general arguments based on decomposing estimating equations, as given in Xie and Meng (2012).)

Theorem 5. *Under the conditions given above, we have asymptotically as* $N_Y \to \infty$,

$$\mathrm{Var}\left(\hat{\theta}(T) - \hat{\theta}(Y)\right)\left[\mathrm{Var}\left(\hat{\theta}(T)\right)\right]^{-1} \to F, \tag{3.9}$$

and

$$\mathrm{Var}\left(\hat{\theta}(Y)\right)\left[\mathrm{Var}\left(\hat{\theta}(T)\right)\right]^{-1} \to I - F. \tag{3.10}$$

This establishes the central role of the fraction of missing information F in determining the asymptotic efficiency of multiphase procedures under the usual asymptotic regime. As mentioned above, this is an ideal-case bound on multiphase performance, and it is based on the usual squared-error loss; both the asymptotic regime and amount of knowledge held by the downstream analyst are optimistic. We explore these issues below, focusing on (1) mutual knowledge and alternative definitions of efficiency, (2) the role of reparameterization, (3) asymptotic regimes and multiphase efficiency, and (4) the issue of robustness in multiphase inference.

Mutual knowledge and efficiency. In practice, downstream analysts are unlikely to have complete knowledge of p_Y. Therefore, even if they were given the entire Y, they would not be able to produce the optimal estimator $\hat{\theta}(Y)$, making the F value given by Theorem 5 an unrealistic yardstick. Nevertheless, Theorem 5 suggests a direction for a more realistic standard.

The classical theory of estimation focuses on losses of the form $L(\hat{\theta}, \theta_0)$, where θ_0 denotes the truth. Risk based on this type of loss, given by $R(\hat{\theta}, \theta_0) = E[L(\hat{\theta}, \theta_0)]$, is a raw measure of performance, using the truth as a baseline. An alternative is regret, the difference between the risk of a given estimator and an ideal estimator $\hat{\theta}^*$; that is, $R(\hat{\theta}, \theta_0) - R(\hat{\theta}^*, \theta_0)$. Regret is popular in the learning theory community and forms the basis for oracle inequalities. It provides a more adaptive baseline for comparison than raw risk, but we can push further. Consider evaluating loss with respect to an estimator rather than

the truth. For mean-squared error, this yields

$$R\left(\hat{\theta}(T), \hat{\theta}(Y)\right) = E\left[\left(\hat{\theta}(T) - \hat{\theta}(Y)\right)^{\top}\left(\hat{\theta}(T) - \hat{\theta}(Y)\right)\right]. \tag{3.11}$$

Can this provide a better baseline, and what are its properties?

For MLEs, $R(\hat{\theta}(T), \hat{\theta}(Y))$ behaves the same (asymptotically) as additive regret because Theorem 5 implies that, as $N_Y \to \infty$ under the classical asymptotic regime,

$$R(\hat{\theta}(T), \hat{\theta}(Y)) = \mathrm{Var}(\hat{\theta}(T) - \hat{\theta}(Y)) = \mathrm{Var}(\hat{\theta}(T)) - \mathrm{Var}(\hat{\theta}(Y)) = R(\hat{\theta}(T), \theta_0) - R(\hat{\theta}(Y), \theta_0). \tag{3.12}$$

For inefficient estimators, (3.12) does not hold in general because $\hat{\theta}(T) - \hat{\theta}(Y)$ is no longer guaranteed to be asymptotically uncorrelated with $\hat{\theta}(Y)$. In such cases, this is precisely the reason $\hat{\theta}(T)$ can be more efficient than $\hat{\theta}(Y)$ or, more generally, there exists a constant $\lambda \neq 0$ such that $\lambda\hat{\theta}(T) + (1-\lambda)\hat{\theta}(Y)$ is (asymptotically) more efficient than $\hat{\theta}(Y)$. In the terminology of Meng (1994), the estimation procedure $\hat{\theta}(\cdot)$ is not *self-efficient* if (3.12) does not hold, viewing Y as the complete data Y_{com} and T as the observed data Y_{obs}. Indeed, if $R(\hat{\theta}(T), \theta_0) < R(\hat{\theta}(Y), \theta_0)$, $R(\hat{\theta}(T), \hat{\theta}(Y))$ may actually be *larger* for a *better* $\hat{\theta}(T)$ because of the inappropriate baseline $\hat{\theta}(Y)$; it is a measure of difference, not dominance, in such cases. Hence, some care is needed in interpreting this measure.

Therefore, we can view (3.11) as a generalization of the usual notion of regret, or the relative regret if we divide it by $R(\hat{\theta}(Y), \theta_0)$. This generalization is appealing for the study of preprocessing: we are evaluating the estimator based on preprocessed data directly against what could be done with the complete raw data, sample by sample, and we no longer need to impose the restriction that the downstream analysts must carry out the most efficient estimation under a model that captures the actual preprocessing. This direction is closely related to the idea of strong efficiency from Xie and Meng (2012) and

Meng and Xie (2013), which generalizes the idea of asymptotic decorrelation beyond the simple (but instructive) setting covered here. Such ideas from the theory of missing data provide a strong underpinning for the study of multiphase inference and preprocessing.

Reparameterization. Theorem 5 also emphasizes the range of effects that preprocessing can have, even in ideal cases. Consider the role that F plays under different transformations of θ. Although the eigenvalues of F are invariant under one-to-one transformations of the parameters, submatrices of F can change substantially. Formally, if $\theta = (\theta_1, \theta_2)$ is transformed to $\omega = (\omega_1, \omega_2) = (g_1(\theta_1, \theta_2), g_2(\theta_1, \theta_2))$, then the fraction of missing information for ω_1 can be very different from that for θ_1. These changes mean that changes in parameterization can reallocate the fractions of missing information among resulting subparameters in unexpected—and sometimes very unpleasant—ways. This is true even for linear transformations; a given preprocessing technique can preserve efficiency for θ_1 and θ_2 individually while performing poorly for $\theta_1 - \theta_2$. Such issues have arisen in, for instance, the work of Xie and Meng (2012) when attempting to characterize the behavior of multiple imputation estimators under uncongeniality.

Asymptotic regimes and multiphase efficiency. On a fundamental level, Theorem 5 is a negative result for preprocessing, at least for MLEs. Reducing the data from Y to T can only hinder the downstream analyst. Formally, this means that $I_T \leq I_Y$ (asymptotically) in the sense that $I_Y - I_T$ is positive semi-definite. As a result, $\hat{\theta}(Y)$ will dominate $\hat{\theta}(T)$ in asymptotic variance for any preprocessing T. Thus, the only justification for preprocessing appears to be pragmatic; if the downstream analyst could not make use of p_Y for efficient inference or such knowledge could not be effectively transmitted, preprocessing provides a feasible way to obtain the inferences of interest. However, this conclusion depends crucially on the assumed behavior of the nuisance parameter ξ.

The usual asymptotic regime is not realistic for many multiphase settings, particularly with regards to ξ. In many problems of interest, $\dim(\xi)/N_Y$ does not tend to zero as N_Y increases, preventing sufficient accumulation of information on the nuisance parameter ξ. A typical regime of this type would accumulate observations Y_i from individual experiments i, each of which brings its own nuisance parameter ξ_i. Such a process could describe the accumulation of data from microarrays, for instance, with each experiment corresponding to a chip with its own observation parameters, or the growth of astronomical datasets with time-varying calibration. In such a regime, preprocessing can have much more dramatic effects on asymptotic efficiency.

In the presence of nuisance parameters, inference based on T can be more robust and even more efficient than inference based on Y. It is well-known that the MLE can be inefficient and even inconsistent in regimes where $\dim(\xi) \to \infty$ (going back to at least Neyman and Scott, 1948). Bayesian methods provide no panacea either. Marginalization over the nuisance parameter ξ is appealing, but resulting inferences are typically sensitive to the prior on ξ, even asymptotically. In many cases (such as the canonical Neyman-Scott problem), only a minimal set of priors provide even consistent Bayes estimators. Careful preprocessing can, however, enable principled inference in such regimes.

Such phenomena stand in stark contrast to the theory of multiple imputation. In that theory, complete data inferences are typically assumed to be valid (in the sense of self-efficiency, for example). Thus, under traditional missing data mechanisms, the observed data (corresponding to T) cannot provide better inferences than Y. This is not necessarily true in multiphase settings. If the downstream analyst is constrained to particular principles of inference (e.g., MLE or Bayes), then estimators based on T can provide lower asymptotic variance than those based on Y. This occurs, in part, because the mechanisms generating Y and T from X are less restricted in the multiphase setting compared to the traditional missing-data framework. Principled inferences based on X would, in the mul-

tiphase setting, generally dominate those based on either Y or T. However, such a relationship need not hold between Y and T without restrictions on the behavior of ξ. We emphasize that this does not contradict the general call in Meng and Xie (2013) to follow the probabilistically-principled methods (such as MLE and Bayes recipes) to prevent violations of self-efficiency, precisely because the well-established principles of single-phase inference may need to be "re-principled" before they can be equally effective in the far more complicated multiphase setting.

Robustness and nuisance parameters. In the simplest case, if a T can be found such that it is a pivot with respect to ξ and remains dependent upon θ, then sensitivity to the behavior of ξ can be eliminated by preprocessing. In such cases, an MLE or Bayes rule based on T can dominate that based on Y even asymptotically. One such example would be providing z-statistics from each of a set of experiments to the downstream analyst. This clearly limits the range of feasible downstream inferences. With these z-statistics, detection of signals via multiple testing (e.g., Benjamini and Hochberg, 1995) would be straightforward, but efficient combination of information across experiments could be difficult. This is a ubiquitous trade-off of preprocessing: reductions that remove nuisance parameters and improve robustness necessarily reduce the amount of information available from the data. These trade-offs must be considered carefully when designing preprocessing techniques—universal utility is unattainable without the original data.

A more subtle case involves the selection of T as a "partial pivot". In some settings, there exists a decomposition of ξ as (ξ_1, ξ_2) such that $\dim(\xi_1) < D$ for some fixed D and all N_Y, and the distribution of T is free of ξ_2 for all values of ξ_1. Many normalization techniques used in the microarray application of Section 3.2.2 can be interpreted in this light. These methods attempt to reduce the unbounded set of experiment-specific nuisance parameters affecting T to a bounded, manageable size.

For example, suppose each processor i observes $y_{ij} \sim N(\beta_0 + \beta_{1i}x_j, \sigma^2)$, $j = 1, \ldots, m$. The downstream analyst wants to estimate β_0, considering $\{\beta_{1i} : i = 1, \ldots, n\}$ and σ^2 as nuisance parameters. In our previous notation, we have $\theta = \beta_0$ and $\xi = (\sigma^2, \beta_{11}, \ldots, \beta_{1n})$. Suppose each preprocessor reduces her data to $T_i = \frac{1}{m}\sum_{j=1}^{m}(y_{ij} - \hat{\beta}_{1i}x_j)$, where $\hat{\beta}_{1i}$ is the OLS estimator of β_{1i} based on $\{y_{ij} : j = 1, \ldots, m\}$. The distribution of each T_i depends on σ^2 but is free of β_{1i}. Hence, $T = \{T_i : i = 1, \ldots, n\}$ is a partial pivot as defined above, with $\xi_1 = \sigma^2$ and $\xi_2 = \{\beta_{1i} : i = 1, \ldots, n\}$.

Such pivoting techniques can allow $\hat{\theta}(T)$ to possess favorable properties even when $\hat{\theta}(Y)$ is inconsistent or grossly inefficient. As mentioned before, this kind of careful preprocessing can dominate Bayesian procedures in the presence of nuisance parameters when $\dim(\xi)$ can grow with N_y. In these regimes, informative priors on ξ can affect inferences even asymptotically. However, reducing Y to T so only the ξ_1-part of ξ is relevant for T's distribution allows information to accumulate on ξ_1, making inferences far more robust to the preprocessor's beliefs about ξ.

These techniques share a common conceptual framework: invariance. Invariance has a rich history in the Bayesian literature, primarily as a motivation for the construction of noninformative or reference priors (e.g., Jeffreys, 1946; Hartigan, 1964; Geisser and Eddy, 1979; Berger and Bernardo, 1992; Kass and Wasserman, 1996). It is fundamental to the pivotal methods discussed above and arises in the theory of partial likelihood (Cox, 1975). We see invariance as a core principle of preprocessing, although its application is somewhat different from most Bayesian settings. We are interested in finding functions of the data that are invariant to subsets of the parameter, not priors invariant to reparameterization. For instance, the rank statistics that form the basis for Cox's proportional hazards regression in the absence of censoring (1972) can be obtained by requiring a statistic invariant to monotone transformations of time. Indeed, Cox's regression based on rank statistics can be viewed as an excellent example of eliminating an infinite dimen-

sional nuisance parameter, i.e., the baseline hazard, via preprocessing, which retains only the rank statistics. The relationship between invariance in preprocessing, modeling, and prior formulation is a rich direction for further investigation.

An interesting practical question arises from this discussion of robustness: how realistic is it to assume efficient inference with preprocessed data? This may seem unrealistic as preprocessing is frequently used to simplify problems so common methods can be applied. However, preprocessing can make many assumptions more appropriate. For example, aggregation can make normality assumptions more realistic, normalization can eliminate nuisance parameters, and discretization greatly reduces reliance on parametric distributional assumptions altogether. It may therefore be more appropriate to assume that efficient estimators are generally used with preprocessed data than with raw data.

The results and examples explored here show that preprocessing is a complex topic in even large-sample settings. It appears formally futile (but practically useful) in standard asymptotic regimes. Under other realistic asymptotic regimes, preprocessing emerges as a powerful tool for addressing nuisance parameters and improving the robustness of inferences. Having established some of the formal motivation and trade-offs for preprocessing, we discuss further extensions of these ideas into more difficult settings in Section 3.5.2.

3.4.3 Giving all that you can

In some cases, effective preprocessing techniques are quite apparent. If $p_Y(Y \mid X, \xi)$ forms an exponential family with parameter X or (X, ξ), then we have a straightforward procedure: retain a minimal sufficient statistic. To be precise, we mean that one of the following

factorizations holds for a sufficient statistic $T(Y)$ of bounded dimension:

$$\begin{aligned} p_Y(Y \mid X, \xi) &= g(Y) \exp\left(T(Y)^\top f(X, \xi) + h(X, \xi)\right); \\ p_Y(Y \mid X, \xi) &= g(Y; \xi) \exp\left(T(Y)^\top f(X) + h(X)\right) \ . \end{aligned}$$

Retaining this sufficient statistic will lead to a lossless compression, assuming that the first-phase model is correct. Unfortunately, such nice cases are rare. Even the Bayesian approach offers little reprieve. Integrating $p_Y(Y \mid X, \xi)$ with respect to a prior $\pi_\xi(\xi)$ typically removes the observation model from the exponential family—consider, for instance, a normal model with unknown variance becoming a t distribution.

If $\log p_Y(Y \mid X)$ is approximately quadratic as a function of X, then retaining its mode and curvature would seem to provide much of the information available from the data to downstream analysts. However, such intuition can be treacherous. If a downstream analyst is combining inferences from a set of experiments, each of which yielded an approximately quadratic likelihood, the individual approximations may not be enough to provide efficient inferences. Approximations that hold near the mode of each experiment's likelihood need not hold away from these modes—including at the mode of the joint likelihood from all experiments. Thus, remainder terms can accumulate in the combination of such approximations, degrading the final inference on θ. Furthermore, the requirement that $\log p_Y(Y \mid X)$ be approximately quadratic in X is quite stringent. To justify such approximations, we must either appeal to asymptotic results from likelihood theory or confine our attention to a narrow class of observation models $p_Y(Y \mid X)$. Unfortunately, asymptotic theory is often an inappropriate justification in multiphase settings, because X grows in dimension with Y in many asymptotic regimes of interest, so there is no general reason to expect information to accumulate on X. These issues are of particular concern as such quadratic approximations are a standard implicit justification for passing point estimates

with standard errors onto downstream analysts.

Moving away from these cases, solutions become less apparent. No processing (short of passing the entire likelihood function) will preserve all information from the sample when sufficient statistics of bounded dimension do not exist. However, multiphase approaches can still possess favorable properties in such settings.

We begin by considering a stubborn downstream analyst—she has her method and will not consider anything else. For example, this analyst could be dead set on using linear discriminant analysis or ANOVA. The preprocessor has only one way to affect her results: carefully designing a particular T given to the downstream analyst. Such a setting is extreme. We are saying that the downstream analyst will charge ahead with a given estimator regardless of her input with neither reflection nor judgment. We investigate this setting because it maximizes the preprocessor's burden in terms of her contribution to the final estimate's quality. Formally, we consider a fixed second-stage estimator $\hat{\theta}(T)$; that is, the form of its input T and the function producing $\hat{\theta}$ are fixed, but the mechanism actually used to generate T is not. T could be, for example, a vector of fixed dimension.

As we discuss below, admissible designs for the first-phase with a fixed second-phase method are given by a (generalized) Bayes rule. This uses the known portion of the model $p_Y(Y \mid X, \xi)$ to construct inputs for the second stage and assumes that any prior the preprocessor has on ξ is equivalent to what a downstream analyst would have used in the preprocessor's position. Formally, this describes all rules that are admissible among the class of procedures using a given second-stage method, following from previous complete class results in statistical decision theory (e.g., Berger, 1985; Farrell, 1968).

Admissibility. Assume that the second-stage procedure $\hat{\theta}(T)$ is fixed as discussed above and we are operating under the model (3.1). Further assume that the preprocessor's prior on ξ is the only such prior used in all Bayes rule constructions. For $T \in \mathbb{R}^d$,

consider a smooth, strictly convex loss function L. Then, under appropriate regularity conditions (e.g., Berger, 1985; Farrell, 1968), if $\hat{\theta}(T)$ is a smooth function of T, then all admissible procedures for generating T are Bayes or generalized Bayes rules with respect to the risk $R(\hat{\theta}(T), \theta_o)$. The same holds when T is restricted to a finite set.

This guideline follows directly from conventional complete class results in decision theory. We omit technical details here, focusing instead on the guideline's implications. However, a sketch of its proof proceeds along the following lines.

There are two ways to approach this argument: intermediate loss and geometry. The intermediate loss approach uses an intermediate loss function $\tilde{L}(T, \theta_o) = L(\hat{\theta}(T), \theta_o)$. This $\tilde{L}$ is the loss facing the preprocessor given a fixed downstream procedure $\hat{\theta}(T)$. If $\tilde{L}$ is well-behaved, in the sense of satisfying standard conditions (strict convexity, or a finite parameter space, and so on), then the proof is complete from previous results for real T. Similarly, if T is restricted to a finite discrete set, then we face a classical multiple decision problem and can apply previous results to $\tilde{L}(T, \theta_o)$. These straightforward arguments cover a wide range of realistic cases, as Berger (1985) has shown. Otherwise, we must turn to a more intricate geometric argument. Broadly, this construction uses a convex hull of risks generated by attainable rules.

This guideline has direct bearing upon the development of inputs for machine learning algorithms, typically known as *feature engineering*. Given an algorithm that uses a fixed set of inputs, it implies that using a correctly-specified observation model to design these inputs is necessary to obtain admissible inferences. Thus, it is conceptually similar to "Rao-Blackwellization" over part of a probability model.

However, several major caveats apply to this result. First, on a practical level, deriving such Bayes rules is quite difficult for most settings of interest. Second, and more worryingly, this result's scope is actually quite limited. As we discussed in Section 3.4.2, even

Bayesian estimators can be inconsistent in realistic multiphase regimes. However, these estimators are still admissible, as they cannot be dominated in risk for particular values of the nuisance parameters ξ. Admissibility therefore is a minimal requirement; without it, the procedure can be improved uniformly, but with it, it can still behave badly in many ways. Finally, there is the problem of robustness. An optimal input for one downstream estimator $\hat{\theta}_1(T)$ may be a terrible input for another estimator $\hat{\theta}_2(T)$, even if $\hat{\theta}_1$ and $\hat{\theta}_2$ take the same form of inputs. Such considerations are central to many real-world applications of preprocessing, as researchers aim to construct databases for a broad array of later analyses. However, this result does show that engineering inputs for downstream analyses using Bayesian observation models can improve overall inferences. How to best go about this in practice is a rich area for further work.

3.4.4 Counterexamples and conundrums

As befits first steps, we are left with a few loose ends and puzzles. Starting with the DSC condition (3.7) of Section 3.4.1, we provide a simple counterexample to its necessity.

Suppose we have $Y_1, Y_2, X_1, X_2 \in \mathbb{R}^n$. Let $Y_i|X_i \sim N(X_i, I)$ independent of each other. Now, let $X_1 = \theta Z_1$, $Z_1 \sim N(0, I)$, $X_2 = \theta \,\mathrm{abs}(Z_2) \circ \mathrm{sign}(X_1)$, where $Z_2 \sim N(0, I)$, $Z_2 \perp\!\!\!\perp Z_1$, $\mathrm{sign}(X_1)$ is a vector of signs $(-1, 0, \text{or } 1)$ for X_1, $\mathrm{abs}()$ denotes the element-wise absolute value, and $\circ$ denotes the Hadamard product. We fix $\theta > 0$.

As our working model, we posit that $X_i|\eta \sim N(0, \eta_i I)$ independently. Then, we clearly have $(Y_1^\top Y_1, Y_2^\top Y_2) = (T_1, T_2)$ as a sufficient statistic for both η and θ. However, the DSC does not hold for this working model. We cannot write the actual joint distribution of X as a marginalization of $\tilde{p}_X(X|\eta)$ with respect to some distribution over η in such a way that (T_1, T_2) is sufficient for η. To enforce $\mathrm{sign}(X_1) = \mathrm{sign}(X_2)$ under the working model, any such model must use η to share this information.

For this example, we can obtain a stronger result: no factored working model $\tilde{p}_X(X|\eta)$

exists such that (1) $Y_i^\top Y_i$ is sufficient for $g_i(\eta)$ under $\tilde{p}_Y(Y_i|g_i(\eta))$ and (2) the DSC holds. For contradiction, assume such a working model exists. Under this working model, Y_i is conditionally independent of $g_i(\eta)$ given $Y_i^\top Y_i$, so we can write $\tilde{p}_Y(Y_i|g_i(\eta)) = \tilde{p}_Y(Y_i|Y_i^\top Y_i)h_i(Y_i^\top Y_i; g_i(\eta))$. As the DSC holds for this working model, we have

$$p_Y(Y|\theta) = \left[\prod_{i=1}^{2} \tilde{p}_Y(Y_i|Y_i^\top Y_i)\right] \int_\eta \left[\prod_{i=1}^{2} h_i(Y_i^\top Y_i; g_i(\eta))\right] p_\eta(\mathrm{d}\eta|\theta).$$

Hence, we must have Y_1 conditionally independent of Y_2 given $(Y_1^\top Y_1, Y_2^\top Y_2)$. However, this conditional independence does not hold under the true model. Hence, the given working model cannot both satisfy the DSC and have $Y_i^\top Y_i$ sufficient for each $g_i(\eta)$.

The issue here is unparameterized dependence, as mentioned in Section 3.4.1. The X's have a dependence structure that is not captured by θ. Thus, requiring that a working model preserves sufficiency for θ does not ensure that it has enough flexibility to capture the true distribution of Y. A weaker condition than the DSC (3.7) that is necessary and sufficient to ensure that all sufficient statistics for η are sufficient for θ may be possible.

From Sections 3.4.2 and 3.4.3, we are left with puzzles rather than counterexamples. As mentioned previously, many optimality results are trivial without sufficient constraints. For instance, minimizing risk or maximizing Fisher information naively yield uninteresting (and impractical) multiphase strategies: have the preprocessor compute optimal estimators, then pass them downstream. Overly tight constraints bring their own issues. Restricting downstream procedures to excessively narrow classes (e.g., point estimates with standard errors) limits the applied utility of resulting theory and yields little insight on the overall landscape of multiphase inference. Striking the correct balance with these constraints is a core challenge for the theory of multiphase inference and will require a combination of computational, engineering, and statistical insights.

3.5 From the past to the future

As we discussed in Sections 3.3 and 3.4, we have a deep well of questions that motivate further research on multiphase inference. These range from the extremely applied (e.g., enhancing preprocessing in astrophysical systems) to the deeply theoretical (e.g., bounding multiphase performance in the presence of nuisance parameters and computational constraints). We outline a few directions for this research below.

But, before we look forward, we take a moment to look back and place multiphase inference within the context of broader historical debates. Such "navel gazing" helps us to understand the connections and implications of the theory of multiphase inference.

3.5.1 Historical context

On a historical note, the study of multiphase inference touches the long-running debate over the role of decision theory in statistics. One side of this debate, championed by Wald and Lehmann (among others), has argued that decision theory lies at the core of statistical inference. Risk-minimizing estimators and, more generally, optimal decision rules play a central role in their narrative. Even subjectivists such as Savage and de Finetti have embraced the decision theoretic formulation to a large extent. Other eminent statisticians have objected to such a focus on decisions. As noted by Savage (1976), Fisher in particular vehemently rejected the decision theoretic formulation of statistical inference. One interpretation of Fisher's objections is that he considered decision theory useful for eventual economic decision-making, but not for the growth of scientific knowledge.

We believe that the study of multiphase inference brings a unifying perspective to this debate. Fisher's distinction between intermediate processing and final decisions is fundamental to the problem of multiphase inference. However, we also view decision theory as a vital theoretical tool for the study of multiphase inference. Passing only risk-minimizing

point estimators to later analysts is clearly not a recipe for valid inference. The key is to consider the use of previously generated results explicitly in the final decision problem. In the study of multiphase inference, we do so by focusing on the separation of knowledge and objectives between agents. Such separation between preprocessing and downstream inference maps nicely to Fisher's distinction between building scientific knowledge and reaching actionable decisions.

Thus, we interpret Fisher's line of objections to decision-theoretic statistics as, in part, a rejection of adopting a myopic single-phase perspective in multiphase settings. We certainly do not believe that our work will bring closure to such an intense historical debate. However, we do see multiphase inference as an important bridge between these competing schools of thought.

3.5.2 Where can multiphase inference go from here?

We see a wide range of open questions in multiphase inference. Can more systematic ways to leverage the potential of preprocessing be developed? Is it possible to create a mathematical "warning system," alerting practitioners when their inferences from preprocessed data are subject to severe degradation and showing where additional forms of preprocessing are required? And, can multiphase inference inform developments in distributed statistical computation and massive-data inference (as outlined below in Section 3.5.3)? All of these problems call for a shared collection of statistical principles, theory, and methods. Below, we outline a few directions for the development of these tools for multiphase inference.

Passing information. The mechanics of passing information between phases constitute a major direction for further research. One approach leverages the fact that the likelihood function itself is always a minimal sufficient statistic. Thus, a set of (computation-

ally) efficient approximations to the likelihood function $L(X, \xi; Y)$ for (X, ξ) could provide the foundation for a wide range of multiphase methods. Many probabilistic inference techniques for the downstream model (e.g., MCMC samplers) would be quite straightforward to use given such an approximation. The study of such multiphase approximations also offers great dividends for distributed statistical computation, as discussed below. We believe these approximations are promising direction for general-purpose preprocessing. However, there are stumbling blocks.

First, nuisance parameters remain an issue. We want to harness and understand the robustness benefits offered by preprocessing, but likelihood techniques themselves offer little guidance in this direction. Even the work of Cox (1975) on partial likelihood focuses on the details of estimation once the likelihood has been partitioned. We would like to identify the set of formal principles underlying techniques such as partial pivoting (to mute the effect of infinite-dimensional nuisance parameters), building a more rigorous understanding of the role of preprocessing in providing robust inferences. As discussed in Section 3.4.2, invariance relationships may be a useful focus for such investigations, guiding both Bayesian and algorithmic developments.

Second, we must consider the burden placed on downstream analysts by our choice of approximation. Probabilistic, model-based techniques can integrate such information with little additional development. However, it would be difficult for a downstream analyst accustomed to, say, standard regression methods to make use of a complex emulator for the likelihood function. The burden may be substantial for even sophisticated analysts. For instance, it could require a significant amount of effort and computational sophistication to obtain estimates of X from such an approximation, and estimates of X are often of interest to downstream analysts in addition to estimates of θ.

Bounding errors and trade-offs. With these trade-offs in mind and through the formal analysis of widely-applicable multiphase techniques, we can begin to establish bounds on the error properties of such techniques in a broad range of problems under realistic constraints (in both technical and human terms). More general constraints, for instance, can take the form of upper bounds on the regret attainable with a fixed amount of information passed from preprocessor to downstream analyst for fixed classes of scientific models. Extensions to nonparametric downstream methods would have both practical and theoretical implications. In cases where the observation model is well-specified but the scientific model is less clearly defined, multiphase techniques can provide a useful alternative to computationally-expensive semi-parametric techniques. Fusing principled preprocessing with flexible downstream inference may provide an interesting way to incorporate model-based subject-matter knowledge while effectively managing the bias-variance trade-off.

Links to multiple imputation. The directions discussed above share a conceptual, if not technical, history with the development of congeniality (Meng, 1994). Both the study of congeniality in MI and our study of multiphase inference seek to bound and measure the amount of degradation in inferences that can occur when agents attempt (imperfectly) to combine information and inferences. Despite these similarities, the treatment of nuisance parameters are rather different. Nuisance parameters lie at the very heart of multiphase inference, defining many of its core issues and techniques. For MI, the typical approaches have been to integrate them out in a Bayesian analysis (e.g., Rubin, 1996) or assume that the final analyst will handle them (e.g., Nielsen, 2003). Recent work by Xie and Meng (2012) has shed new light on the role of nuisance parameters in MI, but the results are largely negative, demonstrating that nuisance parameters are often a stumbling block for practical MI inference. Understanding the role of preprocessing in addressing nuisance

parameters, providing robust analyses, and effectively distributing statistical inference represent further challenges beyond those pursued with MI. Therefore, much remains to be done in the study of multiphase inference, both theoretical and methodological.

3.5.3 How does multiphase inference inform computation?

We also see multiphase inference as a source for computational techniques, drawing inspiration from the history of MI. MI was initially developed as a strategy for handling missing data in public data releases. However, because MI separates the task of dealing with incomplete data from the task of making inferences, its use spread. It has frequently been used as a practical tool for dealing with missing-data problems where the joint inference of missing data and model parameters would impose excessive modeling or computational burdens. That is, increasingly the MI inference is carried out from imputation through analysis by a single analyst or research group. This is feasible as a computational strategy only because the error properties and conditions necessary for the validity of MI are relatively well-understood (e.g., Meng, 1994; Xie and Meng, 2012).

Multiphase methods can similarly guide the development of efficient, statistically-valid computational strategies. Once we have a theory showing the trade-offs and pitfalls of multiphase methods, we will be equipped to develop them into general computational techniques. In particular, our experience suggests that models with a high degree of conditional independence (e.g., exchangeable distributions for X) can often provide useful inputs for multiphase inferences, even when the true overall model has a greater degree of stochastic structure. The conditional independence structure of such models allows for highly parallel computation with first-phase procedures, providing huge computational gains on modern distributed systems compared to methods based on the joint model.

For example, in Blocker and Protopapas (2012), a factored model was used to preprocess a massive collection of irregularly-sampled astronomical time series. The model was so-

phisticated enough to account for complex observation noise, yet its independence structure allowed for efficient parallelization of the necessary computation. Its output was then combined and used for population-level analyses. Just as Markov chain Monte-Carlo (MCMC) has produced a windfall of tools for approximate high-dimensional integration (see Brooks et al., 2010, for many examples), we believe that this type of principled preprocessing, with further theoretical underpinnings, has the potential to become a core tool for the statistical analysis of massive datasets.

Acknowledgments

We would like to acknowledge support from the Arthur P. Dempster Award and partial financial support from the NSF. We would also like to thank Arthur P. Dempster and Stephen Blyth for their generous feedback. This work developed from the inaugural winning submission for said award. We also thank David van Dyk, Brandon Kelly, Nathan Stein, Alex D'Amour, and Edo Airoldi for valuable discussions and feedback, and Steven Finch for proofreading. Finally, we would like to thank our reviewers for their thorough and thoughtful comments, which have significantly enhanced this paper.

Bibliography

Affymetrix, I. (2002), "Statistical Algorithms Description Document," Affymetrix, Inc., Santa Clara, CA. Available at http://media.affymetrix.com/support/technical/ whitepapers/sadd_whitepaper.pdf. (Accessed April, 2013.).

Albert, I., Mavrich, T. N., Tomsho, L. P., Qi, J., Zanton, S. J., Schuster, S. C., and Pugh, B. F. (2007a), "Translational and rotational settings of H2A.Z nucleosomes across the Saccharomyces cerevisiae genome." *Nature*, 446, 572–6.

— (2007b), "Translational and rotational settings of H2A.Z nucleosomes across the Saccharomyces cerevisiae genome." *Nature*, 446, 572–6.

Anderson, L. D., Zavagno, A., Rodón, J. A., Russeil, D., Abergel, A., Ade, P., André, P., Arab, H., Baluteau, J.-P., Bernard, J.-P., Blagrave, K., Bontemps, S., Boulanger, F., Cohen, M., Compiègne, M., Cox, P., Dartois, E., Davis, G., Emery, R., Fulton, T., Gry, C., Habart, E., Huang, M., Joblin, C., Jones, S. C., Kirk, J. M., Lagache, G., Lim, T., Madden, S., Makiwa, G., Martin, P., Miville-Deschênes, M.-A., Molinari, S., Moseley, H., Motte, F., Naylor, D. A., Okumura, K., Pinheiro Gonçalves, D., Polehampton, E., Saraceno, P., Sauvage, M., Sidher, S., Spencer, L., Swinyard, B., Ward-Thompson, D., and White, G. J. (2010),

"The physical properties of the dust in the RCW 120 H ii region as seen by Herschel," *Astronomy and Astrophysics*, 518, L99.

Barski, A. and Zhao, K. (2009), "Genomic location analysis by ChIP-Seq," *Journal of Cellular Biochemistry*, 107, 11–18.

Bell, A. W., Deutsch, E. W., Au, C. E., Kearney, R. E., Beavis, R., Sechi, S., Nilsson, T., Bergeron, J. J. M., Beardslee, T. A., Chappell, T., Meredith, G., Sheffield, P., Gray, P., Hajivandi, M., Pope, M., Predki, P., Kullolli, M., Hincapie, M., Hancock, W. S., Jia, W., Song, L., Li, L., Wei, J., Yang, B., Wang, J., Ying, W., Zhang, Y., Cai, Y., Qian, X., He, F., Meyer, H. E., Stephan, C., Eisenacher, M., Marcus, K., Langenfeld, E., May, C., Carr, S. A., Ahmad, R., Zhu, W., Smith, J. W., Hanash, S. M., Struthers, J. J., Wang, H., Zhang, Q., An, Y., Goldman, R., Carlsohn, E., van der Post, S., Hung, K. E., Sarracino, D. A., Parker, K., Krastins, B., Kucherlapati, R., Bourassa, S., Poirier, G. G., Kapp, E., Patsiouras, H., Moritz, R., Simpson, R., Houle, B., LaBoissiere, S., Metalnikov, P., Nguyen, V., Pawson, T., Wong, C. C. L., Cociorva, D., Yates III, J. R., Ellison, M. J., Lopez-Campistrous, A., Semchuk, P., Wang, Y., Ping, P., Elia, G., Dunn, M. J., Wynne, K., Walker, A. K., Strahler, J. R., Andrews, P. C., Hood, B. L., Bigbee, W. L., Conrads, T. P., Smith, D., Borchers, C. H., Lajoie, G. A., Bendall, S. C., Speicher, K. D., Speicher, D. W., Fujimoto, M., Nakamura, K., Paik, Y.-K., Cho, S. Y., Kwon, M.-S., Lee, H.-J., Jeong, S.-K., Chung, A. S., Miller, C. A., Grimm, R., Williams, K., Dorschel, C., Falkner, J. A., Martens, L., and no, J. A. V. i. (2009), "A HUPO test sample study reveals common problems in mass spectrometry—based proteomics," *Nat Meth*, 6, 423–430.

Benjamini, Y. and Hochberg, Y. (1995), "Controlling the false discovery rate: a practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B.*, 57, 289–300.

Berger, J. O. (1985), *Statistical Decision Theory and Bayesian Analysis*, Springer, 2nd ed.

Berger, J. O. and Bernardo, J. M. (1992), "On the development of reference priors," *Bayesian Statistics*, 4, 35–60.

Blackwell, D. (1951), "Comparison of experiments," *Proceeding of 2nd Berkeley Symposium on Probability and Statistics*, 1, 93–102.

— (1953), "Equivalent comparisons of experiments," *Annals of Statistics*, 24(2), 265–272.

Blocker, A. W. and Protopapas, P. (2012), "Semi-parametric robust event detection for massive time-domain databases," in *Statistical Challenges in Modern Astronomy V*, eds. Feigelson, E. D. and Babu, G. J., New York, NY: Springer, vol. 902 of *Lecture Notes in Statistics*, pp. 177–187.

Bolstad, B. M., Irizarry, R. A., Astrand, M., and Speed, T. P. (2003), "A comparison of normalization methods for high density oligonucleotide array data based on variance and bias," *Bioinformatics*, 19, 185–193.

Braverman, A. J., Fetzer, E. J., Kahn, B. H., Manning, E. M., Robert, B., and Teixeira, J. P. (2012), "Infrared sounder massive dataset analysis for NASA's Atmospheric Infrared Sounder," *Technometrics*, 54, 1–15.

Brogaard, K., Xi, L., Wang, J.-P., and Widom, J. (2012), "A map of nucleosome positions in yeast at base-pair resolution." *Nature*, 486, 496–501.

Brooks, S., Gelman, A., Jones, G. L., and Meng, X.-L. (eds.) (2010), *Handbook of Markov Chain Monte Carlo: Methods and Applications*, Taylor & Francis.

Cairns, J., Spyrou, C., Stark, R., Smith, M. L., Lynch, A. G., and Tavaré, S. (2011), "BayesPeak–an R package for analysing ChIP-seq data." *Bioinformatics (Oxford, England)*, 27, 713–4.

Cox, D. (1975), "Partial likelihood," *Biometrika*, 62, 269–276.

Cox, D. R. (1972), "Regression models and life-tables," *Journal of the Royal Statistical Society. Series B (Methodological)*, 34, 187–220.

Cox, J. and Mann, M. (2008), "MaxQuant enables high peptide identification rates, individualized ppb-range mass accuracies and proteome-wide protein quantification," *Nat Biotechnol*.

Davey, A. (2012), "Massive data streams," *Presented at SolarStat 2012*.

de Godoy, L. M. F., Olsen, J. V., Cox, J., Nielsen, M. L., Hubner, N. C., Fröhlich, F., Walther, T. C., and Mann, M. (2008), "Comprehensive mass-spectrometry-based proteome quantification of haploid versus diploid yeast," *Nature*.

Désert, F.-X., Macías-Pérez, J. F., Mayet, F., Giardino, G., Renault, C., Aumont, J., Benoît, A., Bernard, J.-P., Ponthieu, N., and Tristram, M. (2008), "Submillimetre point sources from the Archeops experiment: very cold clumps in the Galactic plane," *Astronomy and Astrophysics*, 481, 411–421.

Dicker, L., Lin, X., and Ivanov, A. R. (2010), "Increased power for the analysis of label-free LC-MS/MS proteomics data by combining spectral counts and peptide peak attributes." *Mol Cell Proteomics*.

Dupac, X., Bernard, J.-P., Boudet, N., Giard, M., Lamarre, J.-M., M□ny, C., Pajot, F., Ristorcelli, I., Serra, G., Stepnik, B., and Torre, J.-P. (2003), "Inverse temperature dependence of the dust submillimeter spectral index," *Astronomy and Astrophysics*, 404, L11–L15.

Eng, J. K., McCormack, A. L., and Yates, J. R. (1994), "An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database," *J Am Soc Mass Spectrom*, 5, 976–989.

Evans, I., Cresitello-Dittmar, M., Doe, S., Evans, J., Fabbiano, G., Germain, G., Glotfelty, K., Plummer, D., and Zografou, P. (2006), "The Chandra X-ray Observatory data processing system," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, vol. 6270 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*.

Farrell, R. H. (1968), "On a necessary and sufficient condition for admissibility of estimators when strictly convex loss is used," *Annals of Mathematical Statistics*, 39, 23–28.

Flores, O. and Orozco, M. (2011), "nucleR: a package for non-parametric nucleosome positioning." *Bioinformatics (Oxford, England)*, 27, 2149–50.

Franks, A. M., Csardi, G., Choi, D. S., Drummond, D. A., and Airoldi, E. M. (2013), "Estimating a structured covariance matrix from multi-lab measurements in high-throughput biology," Under review.

Fu, K., Tang, Q., Feng, J., Liu, X. S., and Zhang, Y. (2012), "DiNuP: a systematic approach to identify regions of differential nucleosome positioning." *Bioinformatics (Oxford, England)*, 28, 1965–71.

Geisser, S. and Eddy, W. (1979), "A predictive approach to model selection," *Journal of the American Statistical Association*, 74, 153–160.

Geman, D. (2012), "Order statistics and gene regulation," *Medallion Lecture at Joint Statistical Meetings*.

Geman, D., D'Avignon, C., Naiman, D. Q., and Winslow, R. L. (2004), "Classifying gene expression profiles from pairwise mRNA comparisons," *Statistical Applications in Genetics and Molecular Biology*, 3.

Gerber, S. A., Rush, J., Stemman, O., Kirschner, M. W., and Gygi, S. P. (2003), "Absolute quantification of proteins and phosphoproteins from cell lysates by tandem MS." *Proc Natl Acad Sci USA*, 100, 6940–6945.

Ghaemmagham, S., O'Shea, E., and Weissman, J. (2003), "Global analysis of protein expression in yeast," *Nature*.

Gkikopoulos, T., Schofield, P., Singh, V., Pinskaya, M., Mellor, J., Smolle, M., Workman, J. L., Barton, G. J., and Owen-Hughes, T. (2011), "A Role for Snf2-Related Nucleosome-Spacing Enzymes in Genome-Wide Nucleosome Organization," *Science*, 333, 1758–1760.

Goel, P. and DeGroot, M. (1979), "Comparisons of experiments and information measures," *Annals of Statistics*, 7(2), 1066–1077.

Gray, R. M. and Neuhoff, D. L. (1998), "Quantization," *IEEE Transactions on Information Theory*, 44, 2325–2383.

Gupta, M. (2007), "Generalized hierarchical markov models for the discovery of length-constrained sequence features from genome tiling arrays." *Biometrics*, 63, 797–805.

Hartigan, J. (1964), "Invariant prior distributions," *Annals of Mathematical Statistics*, 35, 836–845.

Ioannidis, J. P. A. and Khoury, M. J. (2011), "Improving validation practices in "omics" research." *Science*, 334, 1230–1232.

Irizarry, R. A., Hobbs, B., Beazer-barclay, Y. D., Antonellis, K. J., Scherf, U., and Speed, T. P. (2003), "Exploration, normalization, and summaries of high density oligonucleotide array probe level data," *Biostatistics*, 4, 249–264.

Irizarry, R. A., Wu, Z., and Jaffee, H. A. (2006), "Comparison of Affymetrix GeneChip expression measures." *Bioinformatics*, 22, 789–794.

Ishihama, Y., Oda, Y., Tabata, T., Sato, T., Nagasu, T., Rappsilber, J., and Mann, M. (2005), "Exponentially modified protein abundance index (emPAI) for estimation of absolute protein amount in proteomics by the number of sequenced peptides per protein." *Mol Cell Proteomics*.

Jansen, A. and Verstrepen, K. J. (2011), "Nucleosome Positioning in Saccharomyces cerevisiae." *Microbiology and molecular biology reviews : MMBR*, 75, 301–20.

Jeffreys, H. (1946), "An invariant form for the prior probability in estimation problems," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 186, 453–461.

Kadane, J. B. (1993), "Several Bayesians: A review," *Test*, 2, 1–32.

Karpievitch, Y., Stanley, J., Taverner, T., Huang, J., Adkins, J., Ansong, C., Heffron, F., Metz, T., Qian, W., and Yoon, H. (2009), "A statistical framework for protein quantitation in bottom-up MS-based proteomics," *BIOINFORMATICS*.

Kass, R. E. and Wasserman, L. (1996), "The selection of prior distributions by formal rules," *Journal of the American Statistical Association*, 91, 1343–1370.

Kelly, B. C., Shetty, R., Stutz, A. M., Kauffmann, J., Goodman, A. a., and Launhardt, R. (2012), "Dust spectral energy distributions in the era of Herschel and Planck : a hierarchical Bayesian-fitting technique," *The Astrophysical Journal*, 752, 55.

Langmead, B., Trapnell, C., Pop, M., and Salzberg, S. L. (2009), "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome," *Genome Biology*, 10, R25.

Le Cam, L. (1964), "Sufficiency and approximate sufficiency," *Annals of Mathematical Statistics*, 35, 1419–1455.

Lee, W., Tillo, D., Bray, N., Morse, R. H., Davis, R. W., Hughes, T. R., and Nislow, C. (2007), "A high-resolution atlas of nucleosome occupancy in yeast." *Nature genetics*, 39, 1235–44.

Lehmann, E. and Casella, G. (1998), *Theory of Point Estimation*, Springer, 2nd ed.

Lindley, D., Tversky, A., and Brown, R. (1979), "On the reconciliation of probability assessments," *Journal of the Royal Statistical Society. Series A.*, 142, 146–180.

Liu, H., Sadygov, R. G., and Yates, J. R. (2004), "A model for random sampling and estimation of relative protein abundance in shotgun proteomics." *Anal Chem*.

Liu, Q. and Pierce, D. (1994), "A note on Gauss-Hermite quadrature," *Biometrika*, 81, 624–629.

Lu, P., Vogel, C., Wang, R., Yao, X., and Marcotte, E. (2006), "Absolute protein expression profiling estimates the relative contributions of transcriptional and translational regulation," *Nat Biotechnol*.

Luo, R., Colangelo, C., and Sessa, W. (2009), "Bayesian Analysis of iTRAQ Data with Nonrandom Missingness: Identification of Differentially Expressed Proteins," *Statistics in Biosciences*.

Makawita, S. and Diamandis, E. P. (2010), "The bottleneck in the cancer biomarker pipeline and protein quantification through mass spectrometry-based approaches: current strategies for candidate verification." *Clin. Chem.*

McGee, M. and Chen, Z. (2006), "Parameter estimation for the exponential-normal convolution model for background correction of Affymetrix GeneChip data," *Statistical Applications in Genetics and Molecular Biology*, 5, Article 24.

Meng, X.-L. (1994), "Multiple-imputation inferences with uncongenial sources of input (with discussion)," *Statistical Science*, 9, 538–558.

Meng, X.-L. and Romero, M. (2003), "Discussion: Efficiency and self-efficiency with multiple imputation inference," *International Statistical Review*, 71, 607–618.

Meng, X.-L. and Rubin, D. B. (1991), "Using EM to obtain asymptotic variance-covariance matrices: the SEM algorithm," *Journal of the American Statistical Association*, 86, 899–909.

Meng, X.-L. and Xie, X. (2013), "I got more data, my model is more refined, but my estimator is getting worse! Am I just dumb?" *Econometric Reviews (special issue on Bayesian Inference and Information Theoretic Methods: In Memory of Arnold Zellner)*, to appear.

Michalski, A., Cox, J., and Mann, M. (2011), "More than 100,000 Detectable Peptide Species Elute in Single Shotgun Proteomics Runs but the Majority is Inaccessible to Data-Dependent LC–MS/MS," *J Proteome Res*, 10, 1785–1793.

Mitra, R. and Gupta, M. (2011), "A continuous-index Bayesian hidden Markov model for prediction of nucleosome positioning in genomic DNA." *Biostatistics (Oxford, England)*, 12, 462–77.

Mueller, O. (2000), "High precision restriction fragment sizing with the Agilent 2100 Bioanalyzer Application Note," .

Nash, S. (2000), "A survey of truncated-Newton methods," *Journal of Computational and Applied Mathematics*, 124, 45–59.

Neal, R. (2010), "MCMC using Hamiltonian dynamics," *Handbook of Markov Chain Monte Carlo*, 54, 113–162.

Neyman, J. and Scott, E. L. (1948), "Consistent estimates based on partially consistent observations," *Econometrica*, 16, 1–32.

Nguyen, X., Wainwright, M. J., and Jordan, M. I. (2009), "On surrogate loss functions and *f*-divergences," *The Annals of Statistics*, 37, 876–904.

Nielsen, S. F. (2003), "Proper and improper multiple imputation," *International Statistical Review*, 71, 593–607.

Old, W. M., Meyer-Arendt, K., Aveline-Wolf, L., Pierce, K. G., Mendoza, A., Sevinsky, J. R., Resing, K. A., and Ahn, N. G. (2005), "Comparison of label-free methods for quantifying human proteins by shotgun proteomics." *Mol Cell Proteomics*.

Ong, S.-E., Blagoev, B., Kratchmarova, I., Kristensen, D. B., Steen, H., Pandey, A., and Mann, M. (2002), "Stable isotope labeling by amino acids in cell culture, SILAC, as a simple and accurate approach to expression proteomics." *Mol Cell Proteomics*.

Paradis, D., Veneziani, M., Noriega-Crespo, A., Paladini, R., Piacentini, F., Bernard, J. P., de Bernardis, P., Calzoletti, L., Faustini, F., Martin, P., Masi, S., Montier, L., Natoli, P., Ristorcelli, I., Thompson, M. A., Traficante, A., and Molinari, S. (2010), "Variations of the spectral index of dust emissivity from Hi-GAL observations of the Galactic plane," *Astronomy and Astrophysics*, 520, L8.

Park, P. J. (2009), "ChIP-Seq: Advantages and challenges of a maturing technology," *Nature Reviews Genetics Genet.*, 10, 669–680.

Pepke, S., Wold, B., and Mortazavi, A. (2009), "Computation for ChIP-seq and RNA-seq studies." *Nature methods*, 6, S22–32.

Perkins, D. N., Pappin, D. J., Creasy, D. M., and Cottrell, J. S. (1999), "Probability-based protein identification by searching sequence databases using mass spectrometry data." *Electrophoresis*, 20, 3551–3567.

Polishko, A., Ponts, N., Le Roch, K. G., and Lonardi, S. (2012), "NORMAL: accurate nucleosome positioning using a modified Gaussian mixture model." *Bioinformatics (Oxford, England)*, 28, 242–9.

Quackenbush, J. (2002), "Microarray data normalization and transformation." *Nature Genetics*, 32 Suppl, 496–501.

Rappsilber, J., Ryder, U., Lamond, A. I., and Mann, M. (2002), "Large-Scale Proteomic Analysis of the Human Spliceosome," *Genome Research*, 12, 1231–1245.

Rashid, N. U., Giresi, P. G., Ibrahim, J. G., Sun, W., and Lieb, J. D. (2011), "ZINBA integrates local covariates with DNA-seq data to identify broad and narrow regions of enrichment, even within amplified genomic regions." *Genome biology*, 12, R67.

Ritchie, M. E., Silver, J., Oshlack, A., Holmes, M., Diyagama, D., Holloway, A., and Smyth, G. K. (2007), "A comparison of background correction methods for two-colour microarrays." *Bioinformatics*, 23, 2700–2707.

Rubin, D. (1976), "Inference and missing data," *Biometrika*, 63, 581–592.

Rubin, D. B. (1987), *Multiple Imputation for Nonresponse in Surveys*, Wiley.

— (1996), "Multiple imputation after 18+ years," *Journal of the American Statistical Association*, 91, 473–489.

Savage, L. (1976), "On rereading RA Fisher," *Annals of Statistics*, 4, 441–500.

Schwartzman, A., Jaffe, A., Gavrilov, Y., and Meyer, C. A. (2011), "Multiple Testing of Local Maxima for Detection of Peaks in ChIP-Seq Data," .

Scigelova, M., Hornshaw, M., Giannakopulos, A., and Makarov, A. (2011), "Fourier transform mass spectrometry." *Mol Cell Proteomics*.

Scigelova, M. and Makarov, A. (2006), "Orbitrap mass analyzer-overview and applications in proteomics," *Proteomics*.

Segal, E., Fondufe-Mittendorf, Y., Chen, L., Thastrom, A., Field, Y., Moore, I. K., Wang, J. Z., and Widom, J. (2006), "A Genomic Code for Nucleosome Positioning," *Nature*, 42, 772–778.

Shetty, R., Kauffmann, J., Schnee, S., Goodman, A. A., and Ercolano, B. (2009), "The effect of line-of-sight temperature variation and noise on dust continuum observations," *The Astrophysical Journal*, 696, 2234–2251.

Shivaswamy, S., Bhinge, A., Zhao, Y., Jones, S., Hirst, M., and Iyer, V. R. (2008), "Dynamic remodeling of individual nucleosomes across a eukaryotic genome in response to transcriptional perturbation." *PLoS biology*, 6, e65.

Silva, J. C. (2005), "Absolute Quantification of Proteins by LCMSE: A Virtue of Parallel ms Acquisition," *Mol Cell Proteomics*, 5, 144–156.

Smyth, G. K. (2005), "Limma : Linear Models for Microarray Data," in *Bioinformatics and Computational Biology Solutions using R and Bioconductor*, eds. Gentelman, R., Carey, V., Dudoit, S., Irizarry, R., and Huber, W., Springer, no. 2005, pp. 397–420.

Steen, H. and Mann, M. (2004), "The ABC's (and XYZ's) of peptide sequencing," *Nat Rev Mol Cell Biol*.

Storey, J. and Tibshirani, R. (2003), "Statistical significance for genomewide studies," *Proceedings of the National Academy of Sciences of the United States of America*, 100, 9440.

Sun, W., Buck, M. J., Patel, M., and Davis, I. J. (2009a), "Improved ChIP-chip analysis by a mixture model approach." *BMC bioinformatics*, 10, 173.

Sun, W., Xie, W., Xu, F., Grunstein, M., and Li, K.-C. (2009b), "Dissecting nucleosome free regions by a segmental semi-Markov model." *PloS one*, 4, e4721.

Tabb, D. L., Vega-Montoto, L., Rudnick, P. A., Variyath, A. M., Ham, A.-J. L., Bunk, D. M., Kilpatrick, L. E., Billheimer, D. D., Blackman, R. K., Cardasis, H. L., Carr, S. A., Clauser, K. R., Jaffe, J. D., Kowalski, K. A., Neubert, T. A., Regnier, F. E., Schilling, B., Tegeler, T. J., Wang, M., Wang, P., Whiteaker, J. R., Zimmerman, L. J., Fisher, S. J., Gibson, B. W., Kinsinger, C. R., Mesri, M., Rodriguez, H., Stein, S. E., Tempst, P., Paulovich, A. G., Liebler, D. C., and Spiegelman, C. (2010), "Repeatability and Reproducibility in Proteomic Identifications by Liquid Chromatography–Tandem Mass Spectrometry," *J Proteome Res*.

Tan, A. C., Naiman, D. Q., Xu, L., Winslow, R. L., and Geman, D. (2005), "Simple decision rules for classifying human cancers from gene expression profiles." *Bioinformatics*, 21, 3896–3904.

Thakur, S. S., Geiger, T., Chatterjee, B., Bandilla, P., Froehlich, F., Cox, J., and Mann, M. (2011), "Deep and highly sensitive proteome coverage by LC-MS-MS without prefractionation." *Mol Cell Proteomics*.

Tirosh, I. (2012), "Computational analysis of nucleosome positioning." *Methods in molecular biology (Clifton, N.J.)*, 833, 443–9.

Tsankov, A. M., Thompson, D. A., Socha, A., Regev, A., and Rando, O. J. (2010), "The role of nucleosome positioning in the evolution of gene regulation." *PLoS biology*, 8, e1000414.

Tusher, V. G., Tibshirani, R., and Chu, G. (2001), "Significance analysis of microarrays applied to the ionizing radiation response." *Proceedings of the National Academy of Sciences of the United States of America*, 98, 5116–5121.

van Dyk, D., Connors, A., Esch, D., Freeman, P., Kang, H., Karovska, M., Kashyap, V., Siemiginowska, A., and Zezas, A. (2006), "Deconvolution in high-energy astrophysics: Science, instrumentation, and methods," *Bayesian Analysis*, 1, 189–236.

Weiner, A., Hughes, A., Yassour, M., Rando, O. J., and Friedman, N. (2010), "High-resolution nucleosome mapping reveals transcription-dependent promoter packaging." *Genome research*, 20, 90–100.

Weiss, M., Schrimpf, S., Hengartner, M. O., Lercher, M. J., and von Mering, C. (2010), "Shotgun proteomics data from multiple organisms reveals remarkable quantitative conservation of the eukaryotic core proteome." *Proteomics*.

Xie, X. and Meng, X.-L. (2012), "Exploring multi-party inferences: what happens when there are three uncongenial models involved?" Submitted.

Xie, Y., Wang, X., and Story, M. (2009), "Statistical methods of background correction for Illumina BeadArray data." *Bioinformatics*, 25, 751–757.

Yassour, M., Kaplan, T., Jaimovich, A., and Friedman, N. (2008), "Nucleosome Positioning from Tiling Microarray Data," *Bioinformatics 24*, 24, i139–146.

Yuan, G.-C. and Liu, J. S. (2008), "Genomic sequence is highly predictive of local nucleosome depletion." *PLoS computational biology*, 4, e13.

Yuan, G.-C., Liu, Y.-J., Dion, M. F., Slack, M. D., Wu, L. F., Altschuler, S. J., and Rando, O. J. (2005), "Genome-scale identification of nucleosome positions in S. cerevisiae." *Science*, 309, 626–30.

Zhang, X., Robertson, G., Woo, S., Hoffman, B. G., and Gottardo, R. (2012), "Probabilistic inference for nucleosome positioning with MNase-based or sonicated short-read data." *PloS one*, 7, e32095.

Zhang, Y., Liu, T., Meyer, C. a., Eeckhoute, J., Johnson, D. S., Bernstein, B. E., Nusbaum, C., Myers, R. M., Brown, M., Li, W., and Liu, X. S. (2008), "Model-based analysis of ChIP-Seq (MACS)." *Genome biology*, 9, R137.

Zhou, X., Blocker, A. W., Airoldi, E. M., and O'Shea, E. K. (2012), "A genome-wide analysis of chromatine remodeling after phosphate starvation," Manuscript.

Zhu, C., Byrd, R. H., Lu, P., and Nocedal, J. (1997), "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization," *ACM Trans. Math. Softw.*, 23, 550–560.

A

Supplemental algorithms and figures for "Template-based estimation of genome-wide nucleosome positioning via distributed HMC"

A.1 Algorithmic details of inference

A.1.1 Distributed HMC sampler

Recall that the model specified in Section 2 is:

$$y_k|\lambda_k \sim \text{Poisson}\left(\lambda_k\right) \tag{A.1}$$

$$\lambda_{(N\times 1)} \equiv X_{(N\times(N-\ell_o))}\,\beta_{((N-\ell_o)\times 1)}, \tag{A.2}$$

$$\beta_k > 0 \text{ for } k = \lfloor \ell_o/2 \rfloor + 1 \ldots N - \lfloor \ell_o/2 \rfloor$$

$$\log \beta_k \sim \text{Normal}\left(\mu_{s_k}, \sigma^2_{s_k}\right) \tag{A.3}$$

given a segmentation function $s : \{1 \ldots N\} \to \{1 \ldots S\}$, which maps the N base pair locations to S regions in which coefficients β_k can be assumed to be identically distributed. X specifies the contribution of a nucleosome positioned at base pair k to the expected number of reads at base pair m due to digestion variability, and $s(k)$ is denoted as s_k for

compactness. This specification is completed with independent priors on each (μ_s, σ_s^2):

$$\sigma_s^2 \sim \text{InvGamma}(\alpha_o, \gamma_o)\,, \tag{A.4}$$

$$\mu_s|\sigma_s^2 \sim N(\mu_o, \frac{\sigma_s^2}{n_s \tau_o})\,, \tag{A.5}$$

where n_s is the length of segment s.

Our MCMC sampler alternates between two conditional updates:

1. Draw $(\vec{\mu}^{(r)}, \vec{\sigma}^{2\,(r)})|\vec{\beta}^{(r-1)}$ directly, then
2. Update $\vec{\beta}^{(r)}|(\vec{\mu}^{(r)}, \vec{\sigma}^{2\,(r)})$ via a distributed HMC step.

The former is a standard conjugate draw, while the latter is done via a distributed version of the standard Hamiltonian Monte Carlo (HMC) routine.

Hyperparameter updates

In detail, step 1 consists of the following draws for each (μ_s, σ_s^2), defining $\vec{\theta} = \log \vec{\beta}$:

$$\begin{aligned}\sigma_s^{2\,(r)}|\vec{\beta}^{(r-1)} \sim \mathit{InvGamma}\Bigg(&\frac{n_s}{2} + \alpha_o,\ \frac{1}{2}\sum_{k:s_k=s}(\theta_k^{(r-1)} - \bar{\theta}_s^{(r-1)})^2 \\ &+ \frac{\tau_o n_s}{2(1+\tau_o)}(\bar{\theta}_s^{(r-1)} - \mu_o)^2 + \gamma_o\Bigg)\,,\end{aligned} \tag{A.6}$$

$$\mu_s^{(r)}|\sigma_s^{2\,(r)}, \vec{\beta}^{(r-1)} \sim N\left(\frac{\bar{\theta}_s^{(r-1)} + \tau_o \mu_o}{1+\tau_o}, \frac{\sigma_s^{2\,(r)}}{n_s(1+\tau_o)}\right)\,, \tag{A.7}$$

where $\bar{\theta}_s^{(r-1)} = \frac{1}{n_s}\sum_{k:s_k=s}\theta_k^{(r-1)}$. These are standard conjugate updates and have computational and memory complexity $O(N)$.

Distributed HMC update for $\vec{\beta}$

The draws in step 2 proceed in two stages, using two partitions of $\vec{\beta}$. The first starts at the beginning of $\vec{\beta}$ and proceeds forward with subvectors of length at most B separated by $2w$, yielding

$$D_1 = \vec{\beta}_{[1:B]}, \vec{\beta}_{[B+2w+1:2B+2w]}, \ldots, \vec{\beta}_{[n_b(B+2w)+1:N]} \,, \tag{A.8}$$

$$D_2 = \vec{\beta}_{[B/2+1:3B/2]}, \vec{\beta}_{[3B/2+2w+1:5B/2+2w]}, \ldots, \vec{\beta}_{[n_b(B+2w)B/2+1:N]} \,. \tag{A.9}$$

The subvectors within each partition are conditionally-independent given the $(\vec{\mu}, \vec{\sigma}^2)$ and the entries of $\vec{\beta}$ separating them. Hence, we can update them in parallel across multiple processors. The basic structure of these updates follows Algorithm 4.

The individual, worker-level HMC updates are done on $\vec{\theta} = \log \vec{\beta}$ and follow the standard leapfrog-based HMC procedure outlined in Neal (2010). To compute these HMC updates, we require the log-posterior density of each subvector of $\vec{\theta}$ and its gradient. First, define $\vec{\lambda} = X\vec{\beta}$, $\vec{m} = (\mu_{s_k} \text{ for } k = 1, \ldots, N)^\top$, and $\vec{v} = (\sigma^2_{s_k} \text{ for } k = 1, \ldots, N)^\top$. Also, for vectors of equal dimension, let $/$ denote entrywise division and $**$ denote entrywise powers. Then,

$$\begin{aligned} \log p(\vec{\theta}|\vec{\mu}, \vec{\sigma}^2) = &-\vec{1}^\top X\vec{\beta} + \sum_k y_k \log\left(\vec{x}_k^T\vec{\beta}\right) \\ &- \frac{1}{2}\sum_k \frac{(\theta_k - \mu_{s_k})^2}{\sigma^2_{s_k}} + \text{const} \,, \end{aligned} \tag{A.10}$$

$$\nabla_{\vec{\theta}} \log p(\vec{\theta}|\vec{\mu}, \vec{\sigma}^2) = -\text{diag}\left(\vec{\beta}\right) X^\top(\vec{1} - \vec{y}/\vec{\lambda}) - (\vec{\theta} - \vec{m})/\vec{v} \,, \tag{A.11}$$

$$\begin{aligned} \nabla_{\vec{\theta}}\nabla_{\vec{\theta}^\top} \log p(\vec{\theta}|\vec{\mu}, \vec{\sigma}^2) = &-\text{diag}\left(\vec{\beta}\right) X'WX\text{diag}\left(\vec{\beta}\right) \\ &- \text{diag}\left(\vec{\beta}\right) X'(\vec{1} - \vec{y}/\vec{\lambda}) - \text{diag}\left(\vec{1}/\vec{\sigma}^2\right) \,, \end{aligned} \tag{A.12}$$

Distributed HMC update

```
/* Send conditioning information                                    */
Broadcast μ⃗^(t−1), and σ⃗²^(t−1) to all workers ;
for offset in (0, B/2):
    /* Send first round of jobs to workers                          */
    for w in range(min(nWorkers, nBlocks)):
        start = max(0, (w − 1)(B + 2w) − 2w + offset) + 1;
        end = min(N, w(B + 2w) + 2w + offset);
        Send θ⃗[start : end] to worker process w with work tag attached ;
    if len(startVec) < nWorkers:
        Pause remaining workers ;
    /* Collect results                                              */
    Set nComplete = 0, nStarted = min(nWorkers, nBlocks)
    while nComplete < nBlocks:
        Receive result θ⃗[start : end] from arbitrary worker with tag b_1 ;
        Incorporate result into working copy of θ⃗^(t) ;
        nComplete++ ;
        if nStarted < nBlocks:
            /* Send additional jobs as needed                       */
            w = nStarted + 1;
            start = max(0, (w − 1)(B + 2w) − 2w + offset) + 1;
            end = min(N, w(B + 2w) + 2w + offset);
            Send θ⃗[start : end] to last completed worker process with work tag attached;
            nStarted++;
    fintq
```

Algorithm 4: Distributed HMC update

where $W = \text{diag}\left(\vec{y}/\vec{\lambda}^{**2}\right)$. Due to the convolution structure of X, all matrix-vector products involving X and $X^\top$ can be reduced to convolutions of vectors with the template vector $\vec{t}$. This also enables efficient computation of the Hessian's diagonal, as

$$\vec{\lambda} = X\vec{\beta} = (\vec{0}^\top_{\lfloor \ell_0/2 \rfloor}\ \vec{\beta}^\top\ \vec{0}^\top_{\lfloor \ell_0/2 \rfloor})^\top * \vec{t} \tag{A.13}$$

$$X^\top(\vec{1} - y/\vec{\lambda}) = (\vec{1} - y/\vec{\lambda}) * \vec{t}\,, \tag{A.14}$$

$$\text{diag}\,(X'WX) = (\vec{y}/\vec{\lambda}^{**2}) * \vec{t}^{**2}\,. \tag{A.15}$$

This reduces the computational complexity of these evaluations to $O(B \log B)$ for each update of each subvector of $\vec{\beta}$. Our block-level HMC steps are detailed in Algorithm 5. In practice, we fix $\epsilon_{\min} = 0.001$, $\epsilon_{\max} = 0.1$, and $L = 100$. We also use a fixed diagonal mass matrix, although the algorithm can accommodate estimating it at every iteration if needed. However, to maintain the $O(B \log B)$ scaling of our algorithm's complexity with block size, M must remain diagonal. If non-diagonal M is used and/or estimated, our HMC update instead scales as $O(B^2)$. In either case, the overall algorithm scales $O(N)$ for given a fixed block size B. Memory requirements are $O(N)$ for the master process running the hyperparameter draws and coordinating the distributed HMC updates. Each worker process requires $O(B \log B)$ memory to run the distributed HMC updates for diagonal M, while using a non-diagonal mass matrix M requires $O(B^2)$ memory per worker.

A.1.2 Approximate EM algorithm

We develop an approximate EM algorithm, based on a Gaussian approximation of the conditional posterior of $\vec{\theta}$, as to obtain starting values for the MCMC sampler given in A.1.1. It provides a high-quality initialization for $\vec{\beta}$, $\vec{\mu}$, and $\vec{\sigma}^2$. Simpler initializations are possible, but obtaining high-quality initial estimates can greatly reduce the number of

Data: Trajectory length L, $\vec{\mu}$, $\vec{\sigma}^2$, $\vec{\theta}[start : end]$, ϵmin, $\epsilon_{\max}$, block start b, template $\vec{t}$, chromosome length N

`/* Subset` $\vec{\theta}[start : end]$ `to B-length subvector to update and buffers */`

$\tilde{\vec{\theta}} = \vec{\theta}[b : \min(b + B - 1, N)]$;

$\tilde{\vec{\theta}}_0 = \tilde{\vec{\theta}}$; $\underline{\vec{\theta}} = (\vec{\theta}[start : b], \vec{\theta}[\min(b + B - 1, N) : end])$;

Draw step size $\epsilon \sim \text{Unif}[\epsilon_{\min}, \epsilon_{\max}]$;

`/* Optionally estimate mass matrix from Hessian; default is identity */`

if *Estimating mass matrix*:

  Maximize log conditional posterior to obtain $\tilde{\vec{\theta}}$;

  $M = -\nabla_{\tilde{\vec{\theta}}} \nabla_{\tilde{\vec{\theta}}^\top} \log p(\tilde{\vec{\theta}} | \vec{\mu}, \vec{\sigma}^2, \underline{\vec{\theta}})$;

  if *Using diagonal mass matrix*:

    $M = \text{diag}(M)$;

else:

  $M = I_{end-start}$;

`/* Draw momentum */`

Draw $\vec{p} \sim N(\vec{0}, M)$;

$\vec{p}_0 = \vec{p}$;

`/* Run leapfrog integration */`

$\vec{p} += \epsilon \nabla_{\tilde{\vec{\theta}}} \log p(\tilde{\vec{\theta}} | \vec{\mu}, \vec{\sigma}^2, \underline{\vec{\theta}})/2$;

for i **in** `range(`L`):`

  $\tilde{\vec{\theta}} += \epsilon M^{-1} p$;

  if $i < L - 1$:

    $\vec{p} += \epsilon \nabla_{\tilde{\vec{\theta}}} \log p(\tilde{\vec{\theta}} | \vec{\mu}, \vec{\sigma}^2, \underline{\vec{\theta}})$;

$\vec{p} += \epsilon \nabla_{\tilde{\vec{\theta}}} \log p(\tilde{\vec{\theta}} | \vec{\mu}, \vec{\sigma}^2, \underline{\vec{\theta}})/2$;

`/* Metropolis-Hastings step to correct for integration errors */`

$\log r = \log p(\tilde{\vec{\theta}} | \vec{\mu}, \vec{\sigma}^2, \underline{\vec{\theta}}) - \log p(\tilde{\vec{\theta}}_0 | \vec{\mu}, \vec{\sigma}^2, \underline{\vec{\theta}}) - 1/2(\vec{p}^\top M^{-1} \vec{p} - \vec{p}_0^\top M^{-1} \vec{p}_0)$;

Draw u Unif$[0, 1]$;

if $u \leq r$:

  return $(\tilde{\vec{\theta}}, 1)$; `/* Accept update */`

else:

  return $(\tilde{\vec{\theta}}_0, 0)$; `/* Reject update */`

Algorithm 5: Worker-level HMC update

MCMC iterations required for reliable inferences.

Choice initial estimator

We use $\hat{\theta}_k = E\left[\theta_k | \vec{y}, \hat{\mu}_{s_k}, \hat{\sigma}^2_{s_k}\right]$ as an initial point estimate of θ_k. The distributed HMC sampler presented in Section A.1.1 yields information on the complete marginal posterior of $\vec{\theta}$ via simulation but, given the scale of this problem, an optimization-based approach is useful as a fast initialization method. The approximate EM algorithm described in Section A.1.2 provides both approximate marginal MAP estimates of the parameters $\vec{\mu}$ and $\vec{\sigma}^2$ and estimates of the target conditional expectations $\hat{\theta}_k$.

Approximate EM algorithm via Laplace approximation

We implement an approximate EM algorithm to provide initial estimates of $(\vec{\theta}, \vec{\mu}, \vec{\sigma})$. In the E-step, we build an approximation of the conditional posterior of $\vec{\theta}$ given $(\vec{y}, \hat{\vec{\mu}}, \hat{\vec{\sigma}}^2)$ to estimate the Q function, detailed below. The M-step updates the estimates of $\vec{\mu}$ and $\vec{\sigma}^2$ toward the marginal posterior mode of $p(\vec{\mu}, \vec{\sigma}^2 | \vec{y})$.

Approximate E-step In the E-step, the objective is to compute

$$Q_t(\vec{\mu}, \vec{\sigma}^2) = E\left[\log p(\vec{\theta}, \vec{\mu}, \vec{\sigma}^2 | \vec{y}) | \vec{y}, \vec{\mu}^{(r-1)}, \vec{\sigma}^{2\,(r-1)}\right]. \tag{A.16}$$

The log joint posterior for $(\vec{\mu}, \vec{\sigma}^2, \vec{\theta})$ is given by

$$\begin{aligned}\log p(\vec{\theta}, \vec{\mu}, \vec{\sigma}^2 | \vec{y}, \vec{s}, \tau_o) &= -\sum_k \vec{x}_k^T \beta_k + \sum_k y_k \log\left(\vec{x}_k^T \beta_k\right) \\ &\quad -\frac{1}{2}\sum_k \log \sigma^2_{s_k} - \frac{1}{2}\sum_k \frac{(\theta_k - \mu_{s_k})^2}{\sigma^2_{s_k}} \\ &\quad -\frac{1}{2}\sum_s \log\left(\frac{\sigma_s^2}{n_s \tau_o}\right) - \frac{1}{2}\sum_s \frac{(\mu_s - \mu_o)^2}{\sigma_s^2 / n_s \tau_o} \\ &\quad -\sum_s \log \sigma_s^2 + \text{const.}\end{aligned} \tag{A.17}$$

Thus, we can write the relevant portion of the expected log conditional posterior for $\vec{\theta}$ given $\{\mu_{s_k}, \sigma^2_{s_k}\}$ as

$$\begin{aligned}Q_t(\vec{\mu}, \vec{\sigma}^2) &= -\frac{1}{2}\sum_k \log \sigma^2_{s_k} - \frac{1}{2}\sum_k \frac{(\hat{\theta}_k - \mu_{s_k})^2}{\sigma^2_{s_k}} - \frac{1}{2}\sum_k \frac{\hat{V}_k}{\sigma^2_{s_k}} \\ &\quad -\frac{1}{2}\sum_s \log\left(\frac{\sigma_s^2}{n_s \tau_o}\right) - \frac{1}{2}\sum_s \frac{(\mu_s - \mu_o)^2}{\sigma_s^2 / n_s \tau_o} - \sum_s \log \sigma_s^2.\end{aligned} \tag{A.18}$$

where $\hat{\theta}_k = E[\theta_k | \vec{\mu}_{(t-1)}, \vec{\sigma}^2_{(t-1)}]$ and $\hat{V}_k = \text{Var}[\theta_k | \vec{\mu}_{(t-1)}, \vec{\sigma}^2_{(t-1)}]$.

While the conditional posterior $p(\vec{\theta} | \vec{y}, \vec{\mu}, \vec{\sigma}^2)$ is available in close form, the necessary expectations $\hat{\theta}_k$ and variances $\hat{V}_k$ are not. However, under the proposed log-Normal/Poisson model structure, the univariate conditional posteriors of θ_k given $\{\mu_{s_k}, \sigma^2_{s_k}\}$ are unimodal, log-concave, nearly symmetric, and have tails that go to zero as $\exp(-c\theta_k^2)$. Thus, these conditional posteriors are nearly Gaussian and a Laplace approximation is appropriate.

To compute the Laplace approximation, we first find the posterior mode of θ_k given $(\vec{\mu}^{(r-1)}, \vec{\sigma}^{2\,(r-1)})$. This amounts to maximizing

$$g(\vec{\theta}) = -\sum_k \vec{x}_k^T \beta_k + \sum_k y_k \log\left(\vec{x}_k^T \beta_k\right) - \frac{1}{2}\sum_k \frac{(\theta_k - \mu_{s_k})^2}{\sigma^2_{s_k}} \tag{A.19}$$

with respect to $\vec{\theta}$. This mode is not available in closed form, but the given objective function is concave and has a continuous gradient, so numerical optimization is feasible.

The Laplace approximation then consists of substituting a Gaussian distribution with mean $\vec{\hat{\theta}}_t$ equal to the mode of g, and variance $\hat{V}_k = -\text{diag}\left(H^{-1}\right)_k$ for the conditional posterior $p(\theta_k|\vec{y}, \mu_{s_k}, \sigma^2_{s_k})$.

M-step The M-step consists of maximizing $Q_t(\vec{\mu}, \vec{\sigma}^2)$ with respect to μ_{s_k} and $\sigma^2_{s_k}$. We obtain two simple closed-form solutions, summarized in Equations A.20 and A.21:

$$\hat{\mu}_s = \frac{1}{1+\tau_0}\left(\frac{1}{n_s}\sum_{k:s_k=s}\hat{\theta}_k\right) + \frac{\tau_0}{1+\tau_0}\mu_0 \tag{A.20}$$

$$\hat{\sigma}^2_s = \frac{\frac{1}{n_s}\sum_{k:s_k=s}(\hat{\theta}_k - \hat{\mu}_s)^2 + \frac{1}{n_s}\sum_{k:s_k=s}\hat{V}_k + \tau_0(\hat{\mu}_s - \mu_0)^2}{1+\tau_0+2/n_s} \tag{A.21}$$

The term $\hat{V}_k$ differentiates the M-step update of σ_s from the update obtained from joint maximization of the log-posterior. The joint mode of this log-posterior is reached at $\vec{\sigma}^2 = \vec{0}$ and $\theta_k = \mu_{s_k}\ \forall k$, as these values would allow the joint log-posterior density to increase without bound. Algorithmically, the $\hat{V}_k$ term introduced by the EM algorithm prevents $\hat{\sigma}^2$ from collapsing to 0, providing non-degenerate inferences.

Distributed approximate E-step

We use the conditional independence structure of $\vec{\beta}$ given $(\vec{\mu}, \vec{\sigma}^2)$ and partitions discussed in Section A.1.1 to distribute our approximate E-step across multiple processors. Given each partition of $\vec{\theta}$, we update the Laplace approximations in parallel, by finding the mode of the conditional posterior subvector-by-subvector. The overall algorithm is blockwise coordinate ascent within each approximate E-step, with each E-step consisting of iterative maximization of $\log p(\vec{\theta}|\vec{y}, \vec{\mu}, \vec{\sigma}^2)$ using different subsets of $\vec{\theta}$ (corresponding to different blocks) in each iteration. This converges to the maximum of $g(\vec{\theta})$ given $\vec{\mu}^{(r-1)}$ and $\vec{\sigma}^{2\,(r-1)}$.

The approximate E-step considers each block in each of the four configurations, during one iteration. More details are give in Section A.1.2. Within each block $m_1 : m_2$, we maximize $\log p(\vec{\theta}_{m_1:m_2} | \vec{y}, \vec{\mu}, \vec{\sigma}^2, \vec{\theta}_{-(m_1:m_2)})$ numerically via L-BFGS-B or a truncated Newton algorithm (Zhu et al., 1997; Nash, 2000); the latter is typically more efficient in this application. We carry out this maximization directly, avoiding the data augmentation typically used in additive Poisson models of this type (e.g., van Dyk et al., 2006). Such data augmentation would require storing and computing at least $(2w + 1)N$ additional variables, provide slower convergence, and slow the overall computation substantially. By controlling the size of the blocks, we can keep the scale of each optimization problem small enough that direct numerical maximization of these conditional posteriors is not a limiting factor for the algorithm (less than 100ms, typically).

We compute the Hessian of the conditional log-posterior $\log p(\vec{\theta} | \vec{y}, \vec{\mu}, \vec{\sigma}^2)$ after each complete scan through the partitions, completing the approximate E-step and providing the information necessary for our M-step. The Hessian is sparse, but its inversion is computationally-intensive even with modern sparse-matrix solvers. Thus, we typically use a diagonal approximation to the Hessian. The diagonal approximation works well in our setting, even though one would expect the strong local dependence generated by the digestion matrix to produce a Hessian with large off-diagonal elements. However, due to the use of exchangeable local regularization, the Hessian is typically diagonally-dominant. The diagonal approximation is quite accurate; we observed few differences to 2-3 significant digits in comparisons of the estimated Hessian and its inverse on small portions of the genome (single ORFs with promoters, approximately 1,000 to 10,000bp in length). We lay out the overall structure of this approximate EM algorithm in Algorithm 6.

Outline of Approximate EM Algorithm

while *not converged*:

/* E-step */

for *Partition* **in** (D_1, D_2):

Update $\hat{\vec{\theta}}$ via numerical maximization of $g(\vec{\theta})$ within each subvector

Approximate Var $\left[\theta_k | \vec{y}, \mu_{s_k}, \sigma^2_{s_k}\right]$ by inverting Hessian of $p(\vec{\theta}|\vec{y}, \vec{\mu}, \vec{\sigma}^2)$

/* M-step */

Update $\vec{\mu}$ and $\vec{\sigma}^2$ as maximizers of $E\left[p(\vec{\theta}, \vec{\mu}, \vec{\sigma}^2|\vec{y}, \vec{s}, \tau_0)|\vec{\mu}_{(t-1)}, \vec{\sigma}^2_{(t-1)}, \vec{s}, \vec{\tau}_0\right]$

fintq

Algorithm 6: Approximate EM Algorithm

Algorithmic details

The algorithm outlined in Section A.1.2 can be implemented on distributed systems with MPI, using the same techniques as the MCMC algorithm presented in Section A.1.1. Due to the use of a quasi-Newton optimization algorithm within each worker's approximate E-step, its computational complexity scales $O(B^2)$ for each such update. However, it scales $O(N)$ in the length of genome given a fixed block size B. Memory requirements are $O(N)$ for the master process running the M-step and coordinating the approximate E-step and $O(B^2)$ (independent of N) for the worker processes running the approximate E-step updates. We lay out the details of this parallel approximate E-step in Algorithm 7.

We this algorithm process a chromosome with $1.5e6$ base pairs in only 11 minutes using 256 threads on Amazon EC2; a chromosome with $2.4e5$ base pairs requires only 1 minute. This method will easily scale to genomes of far greater size (e.g. mice) with this distributed structure, especially using resources such as EC2. The one-to-one substitution of time and processors possible on the cloud makes it an ideal infrastructure for running this type of method.

Parallel Implementation of Approximate E-Step

```
/* Send conditioning information                                    */
```
Broadcast $\vec{\mu}^{(t-1)}$, and $\vec{\sigma}^{2\,(t-1)}$ to all workers ;
for *offset* **in** *(0, B/2)*:
    `/* Send first round of jobs to workers                          */`
    for *w* **in** `range(`*min(nWorkers, nBlocks)*`)`:
        $start = \max(0, (w-1)(B+2w) - 2w + \text{offset}) + 1$;
        $end = \min(N, w(B+2w) + 2w + \text{offset})$;
        Send $\vec{\theta}[start : end]$ to worker process w with work tag attached ;
    if *len(startVec)* $<$ *nWorkers*:
        Pause remaining workers ;
    `/* Collect results                                              */`
    Set *nComplete* = 0, *nStarted* = min(nWorkers, nBlocks)
    while *nComplete* $<$ *nBlocks*:
        Receive result $\vec{\theta}[start : end]$ from arbitrary worker with tag b_1 ;
        Incorporate result into working copy of $\vec{\theta}^{(t)}$;
        nComplete++ ;
        if *nStarted* $<$ *nBlocks*:
            `/* Send additional jobs as needed                       */`
            $w = nStarted + 1$;
            $start = \max(0, (w-1)(B+2w) - 2w + \text{offset}) + 1$;
            $end = \min(N, w(B+2w) + 2w + \text{offset})$;
            Send $\vec{\theta}[start : end]$ to last completed worker process with work tag attached;
            nStarted++;
    fintq
`/* Compute approximate variance, if needed                         */`
Compute approximate Var $\left[\theta_k | \vec{y}, \mu_{s_k}, \sigma^2_{s_k}\right]$ using sparse Cholesky decomposition or diagonal approximation to Hessian;

Algorithm 7: Approximate E-Step

A.2 Additional figures

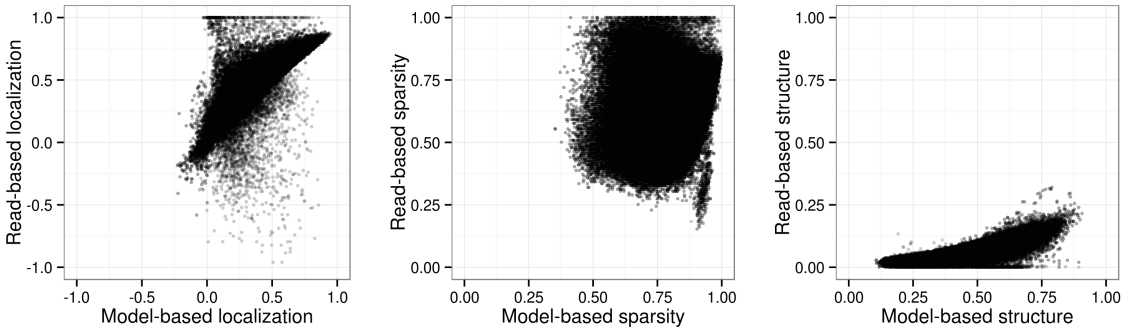

Figure A.1: Joint distributions of of model-based and read-based estimates of cluster-level properties.

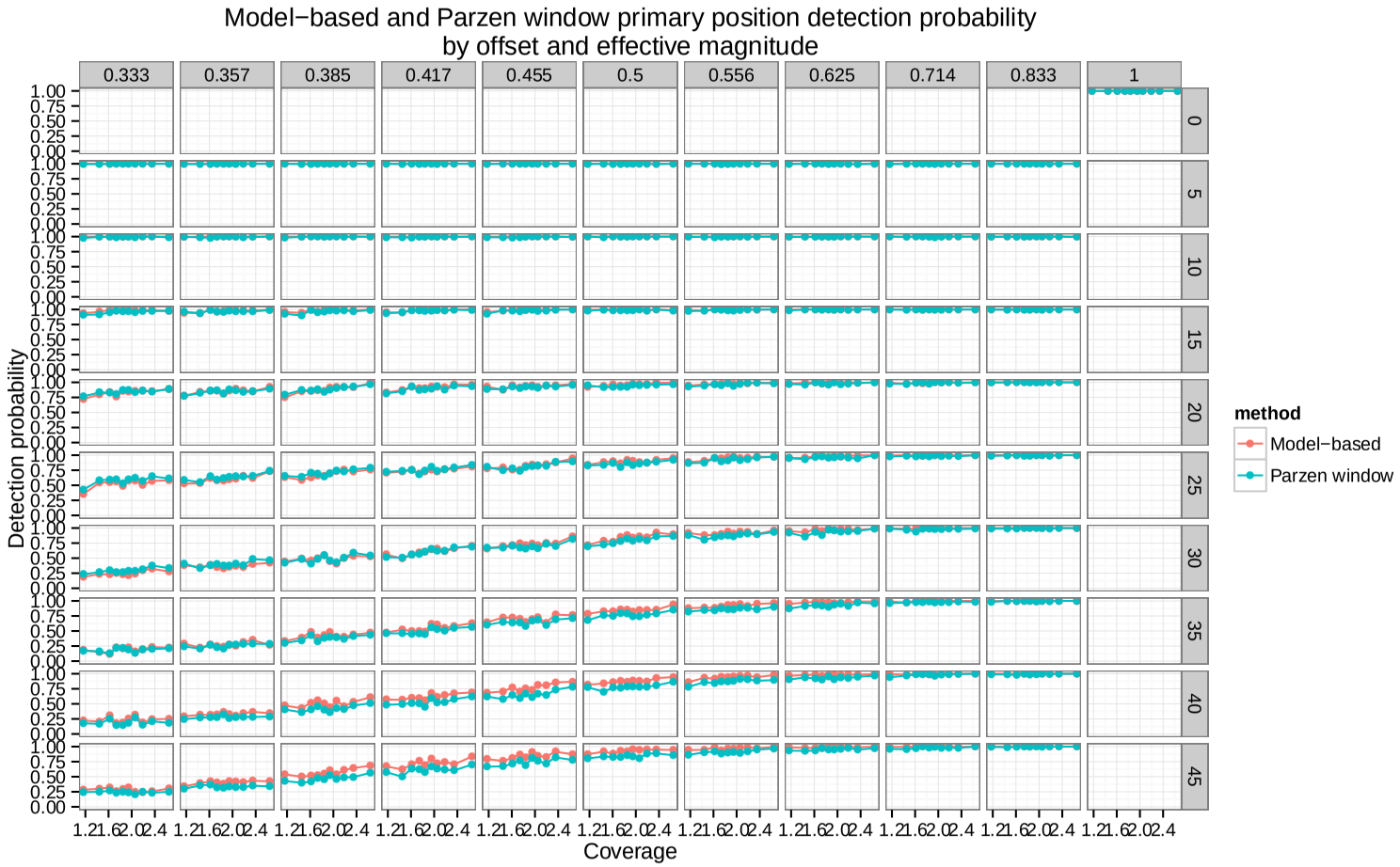

Figure A.2: Power of model-based and Parzen window methods to detect cluster centers ±5bp vs. coverage by alternative position offset (rows) and effective magnitude of primary position (columns)

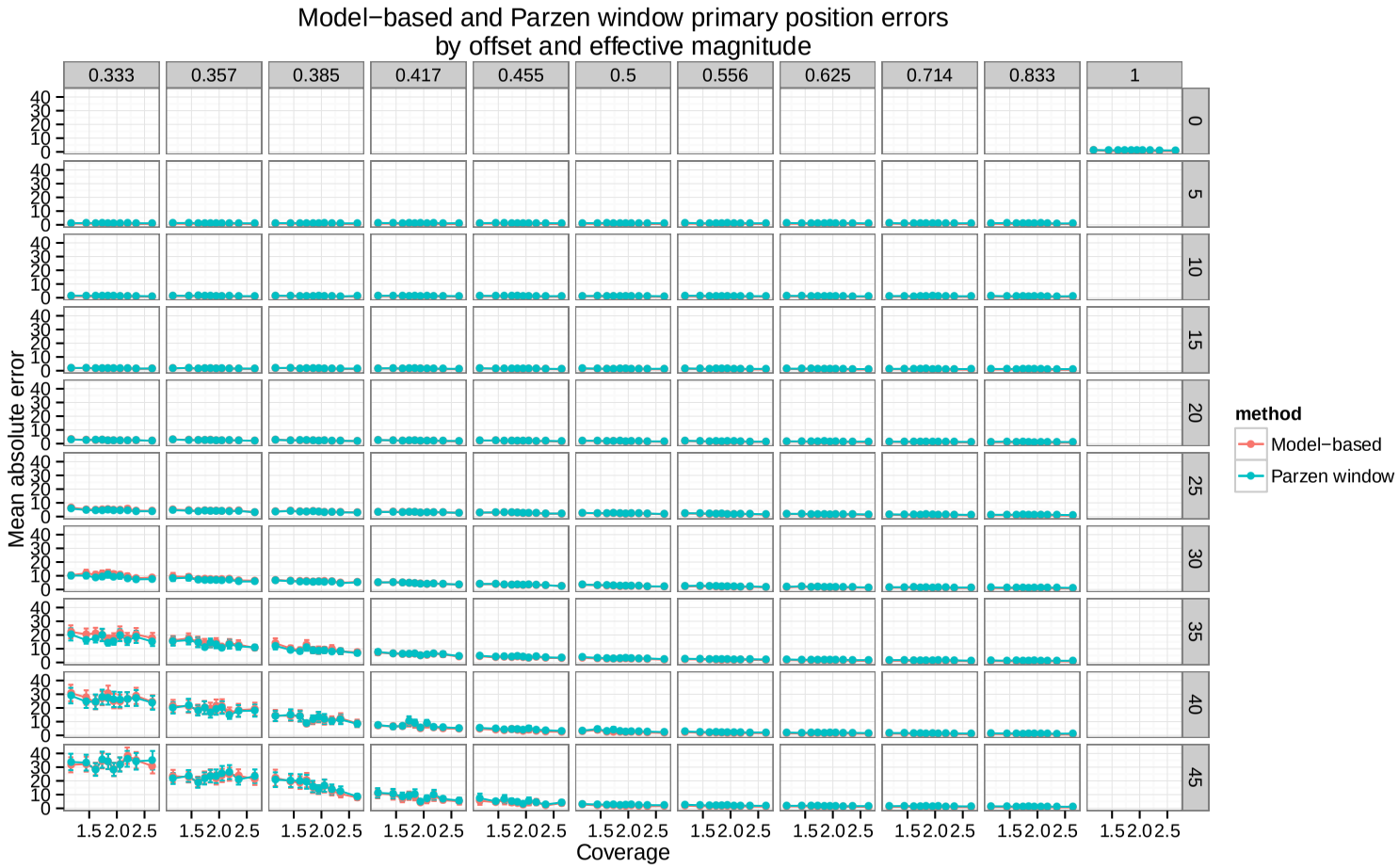

Figure A.3: Mean absolute position errors for model-based and Parzen window methods vs. coverage by alternative position offset (rows) and effective magnitude of primary position (columns)

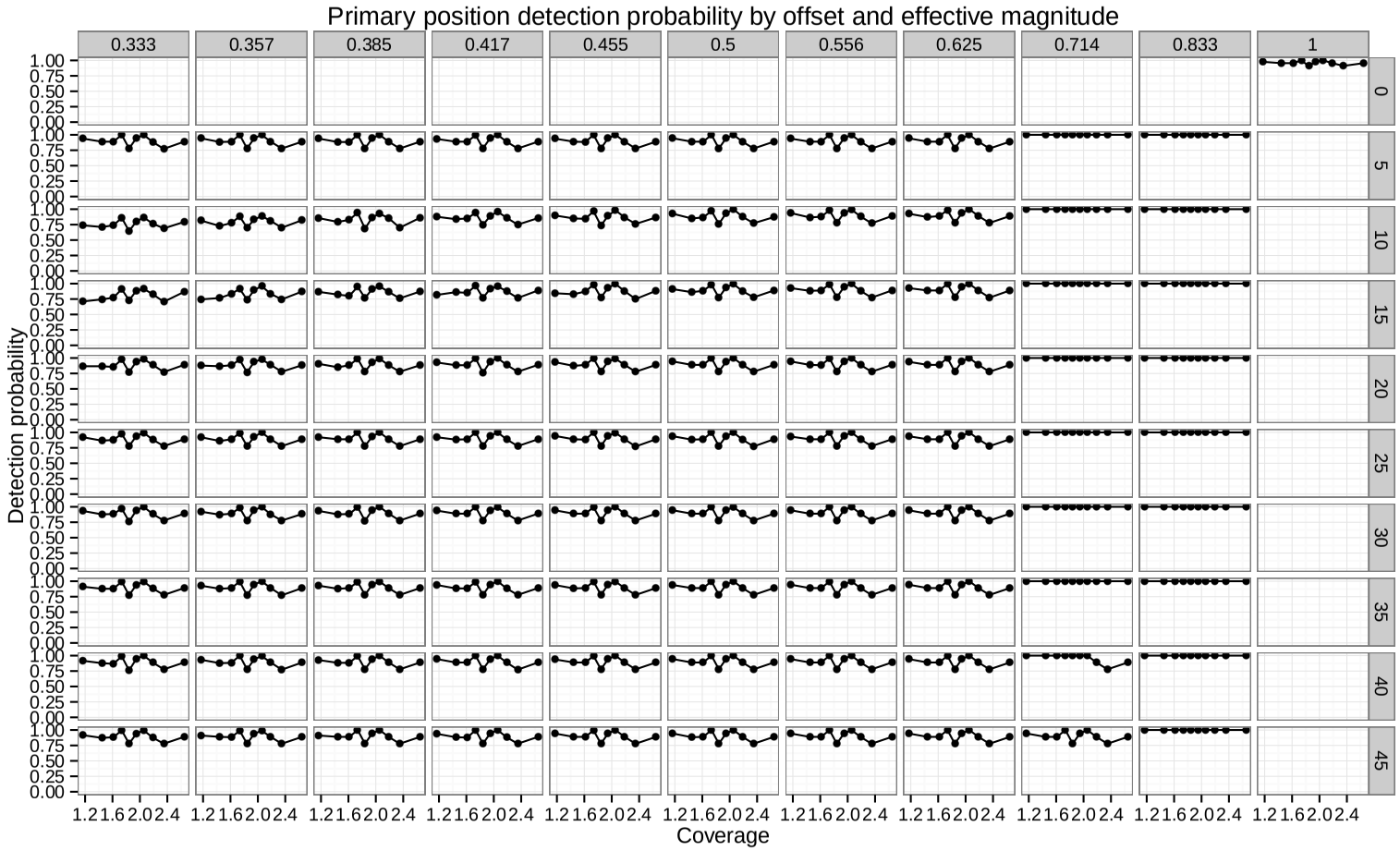

Figure A.4: Power of model-based method to detect individual primary positions ±5bp vs. coverage by alternative position offset (rows) and effective magnitude of primary position (columns)

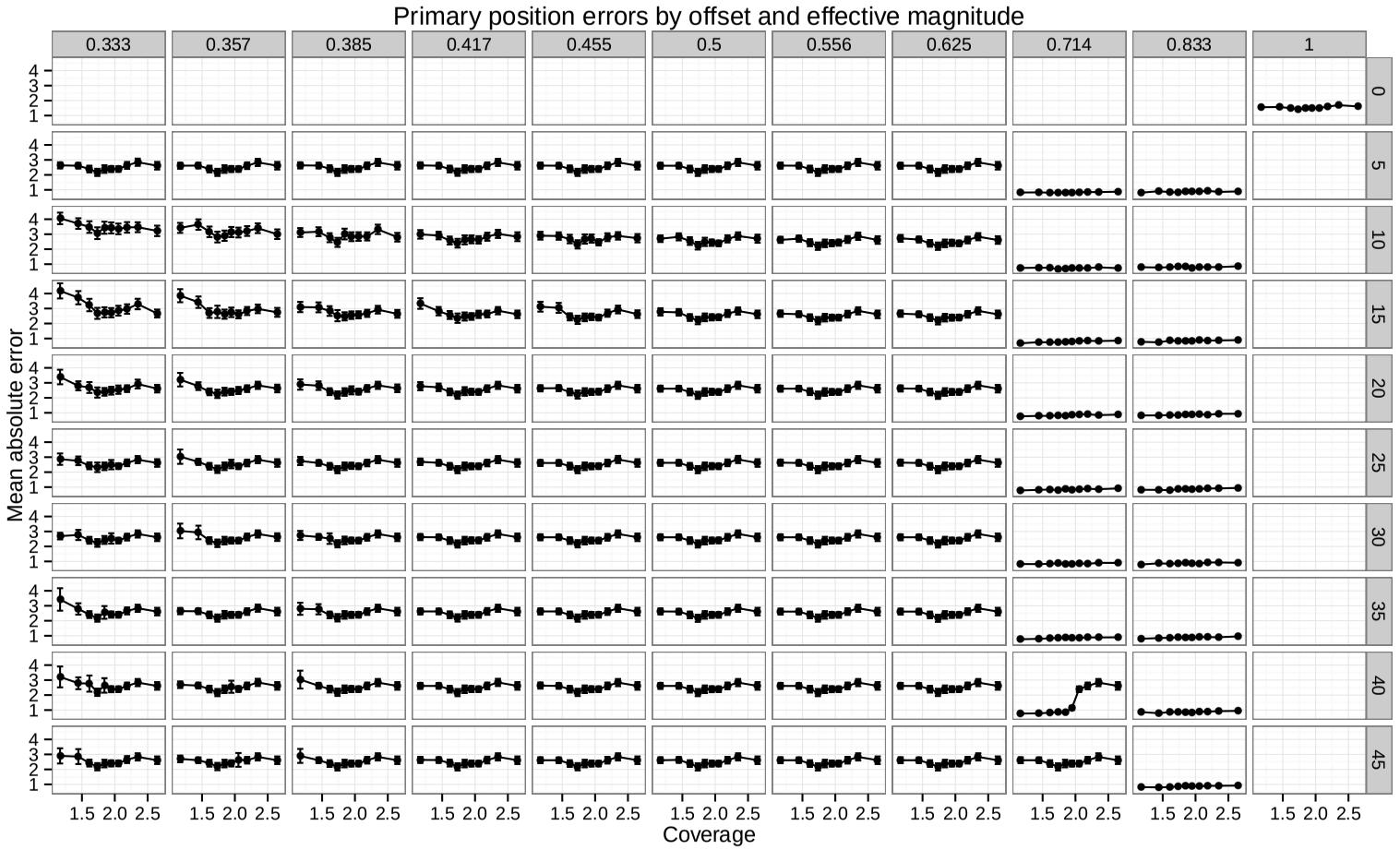

Figure A.5: Mean absolute position errors of model-based method for individual primary positions vs. coverage by alternative position offset (rows) and effective magnitude of primary position (columns)

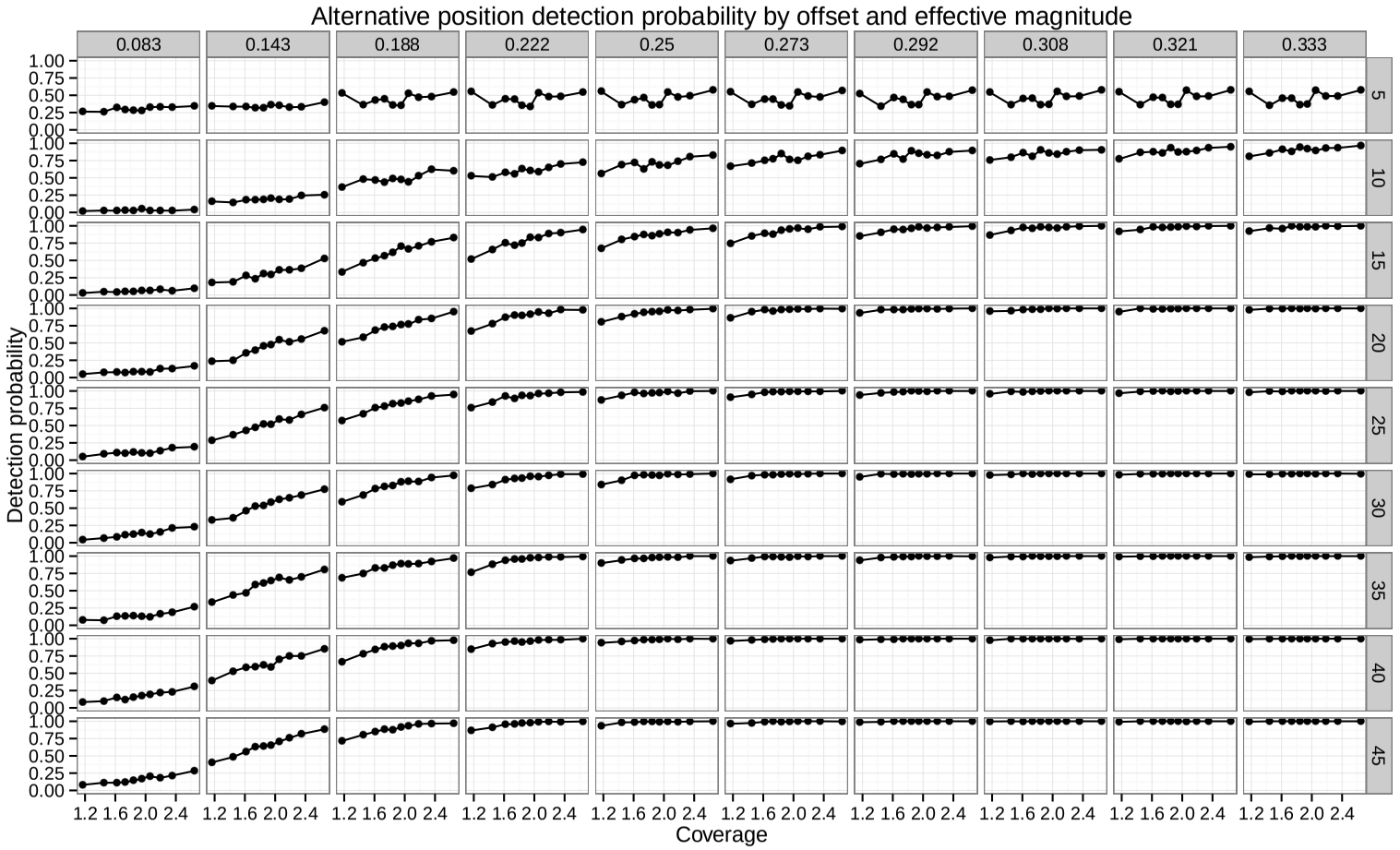

Figure A.6: Power of model-based method to detect individual alternative positions ±5bp vs. coverage by alternative position offset (rows) and effective magnitude of alternative position (columns)

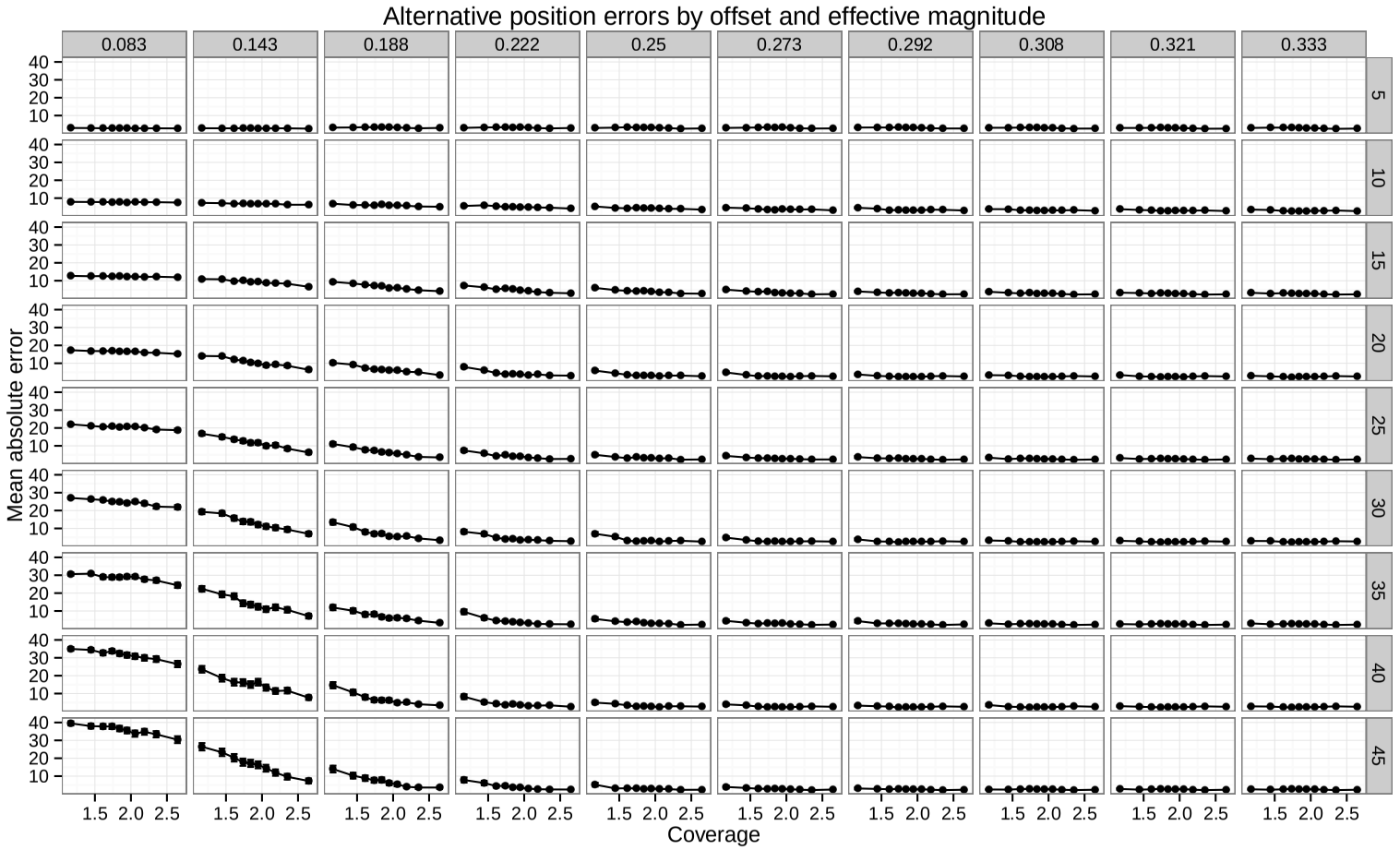

Figure A.7: Mean absolute position errors of model-based method for individual alternative positions vs. coverage by alternative position offset (rows) and effective magnitude of alternative position (columns)

B

Supplemental algorithms and figures for "Absolute protein quantitation: Inference with non-ignorable missing data in high throughput proteomics"

B.1 Prior parameter settings

The parameters of all priors used in our inference were selected to provide weakly-informative regularization. Our results were generally insensitive to perturbations in these parameter values by a factor of two or more. For α_σ and α_τ, the convolution parameters of our inverse-Gamma prior, we take $\log(\alpha_\cdot) \sim N(2.65, 1)$, which implies a prior probability of 0.95 of falling between 2 and 100. This reflects a plausible range of variation while effectively ruling out degenerate cases. For the scale parameters β_σ and β_τ, we use independent $Gamma(1, 1)$ priors. We place uniform ($Beta(1, 1)$) priors on π^{rnd} and λ and take $\log(r) \sim N(2.65, 1)$.

B.2 Details of MCMC algorithm

B.2.1 Gibbs updates for $\vec{\mu}$ and $\vec{\gamma}$

Conditional on the missing data $\vec{M}$ and all other parameters, the conditional posterior distribution of the peptide- and protein-level mean parameters $\vec{\mu}$ and $\vec{\gamma}$ are standard Normal draws. The conditional posterior of the peptide-level means is

$$h\gamma_{ik}|\vec{M},\vec{\Theta}_{-\gamma} \sim Normal\left(\frac{\frac{\mu_i}{\tau_i^2}+\frac{\bar{y}_{ik}}{\sigma_i^2/s_{ik}}}{\frac{1}{\tau_i^2}+\frac{s_{ik}}{\sigma_i^2}}, \left(\frac{1}{\tau_i^2}+\frac{s_{ik}}{\sigma_i^2}\right)^{-1}\right)$$

Similarly, the conditional posterior distribution of the protein-level mean parameters is

$$\mu_i|\vec{M},\vec{\Theta}_{-\mu} \sim Normal\left(\frac{\sum_k \gamma_{ik}}{m_i}, \frac{\tau_i^2}{m_i}\right)$$

.

B.2.2 Updates for variance parameters and hyperparameters

Gibbs updates for peptide- and state-level variance parameters $\vec{\tau}^2$ and $\vec{\sigma}^2$

The peptide and state level precisions have standard conjugate conditional posterior distributions. The conditional posterior distribution of the peptide-level precisions is

$$\frac{1}{\tau_i^2}|\vec{M},\vec{\Theta}_{-\tau^2} \sim Gamma\left(\alpha_\tau+\frac{m_i}{2}, \beta_\tau+\frac{1}{2}\sum_k(\gamma_{ik}-\mu_i)^2\right).$$

Similarly, the conditional posterior distribution for the state-level precisions is

$$\frac{1}{\sigma_i^2}|\vec{M},\vec{\Theta}_{-\sigma^2} \sim Gamma\left(\alpha_\sigma+\frac{1}{2}\sum_k s_{ik}, \beta_\sigma+\frac{1}{2}\sum_{kl}(y_{ikl}-\gamma_{ik})^2\right).$$

Metropolis-Hastings updates for variance hyperparameters

As a result of our weakly-informative log-normal priors on the shape parameters α_σ and α_τ, their conditional posterior is not available in closed-form. To handle these updates, we use conditional independence chain Metropolis-Hastings draws. Throughout the below discussion, we will focus exclusively on $(\alpha_\tau, \beta_\tau)$ as the corresponding update for $(\alpha_\sigma, \beta_\sigma)$ is of identical form.

We build our proposal based on a multivariate normal approximation to the log-transformed parameters. Specifically, we propose from a multivariate log-t distribution given by

$$(\log \alpha_\sigma^*, \log \beta_\sigma^*) \mid \vec{\sigma} \sim \begin{pmatrix} \widehat{\log \alpha_\sigma} \\ \widehat{\log \beta_\sigma} \end{pmatrix} + \sqrt{\frac{\nu}{\nu - 2}} \hat{\mathcal{I}}(\widehat{\log \alpha_\sigma}, \widehat{\log \beta_\sigma})^{-1/2} \vec{t_\nu} \tag{B.1}$$

where $\widehat{\log \alpha_\sigma}$ and $\widehat{\log \beta_\sigma}$ are the conditional posterior modes of $\log \alpha_\sigma$ and $\log \beta_\sigma$, $\hat{\mathcal{I}}(\log \alpha_\sigma, \log \beta_\sigma)$ is the negative Hessian of the log-conditional posterior evaluated at the posterior mode, and $\vec{t_\nu}$ is a vector of independent and identically distributed t_ν random variables.

To build this proposal, we must maximize $p(\log \alpha_\sigma, \log \beta_\sigma \mid \vec{\sigma})$ over $(\log \alpha_\sigma, \log \beta_\sigma)$. This requires only univariate optimization as the mode of $\log \beta_\sigma$ given $\log \alpha_\sigma$ is available in closed form:

$$\hat{\beta}_\sigma = \frac{n\alpha_\sigma + \alpha_{o\sigma}}{\sum_{i=1}^n \sigma_i^{-2}}. \tag{B.2}$$

We provide an example of this approximation's behavior in Figure B.1.

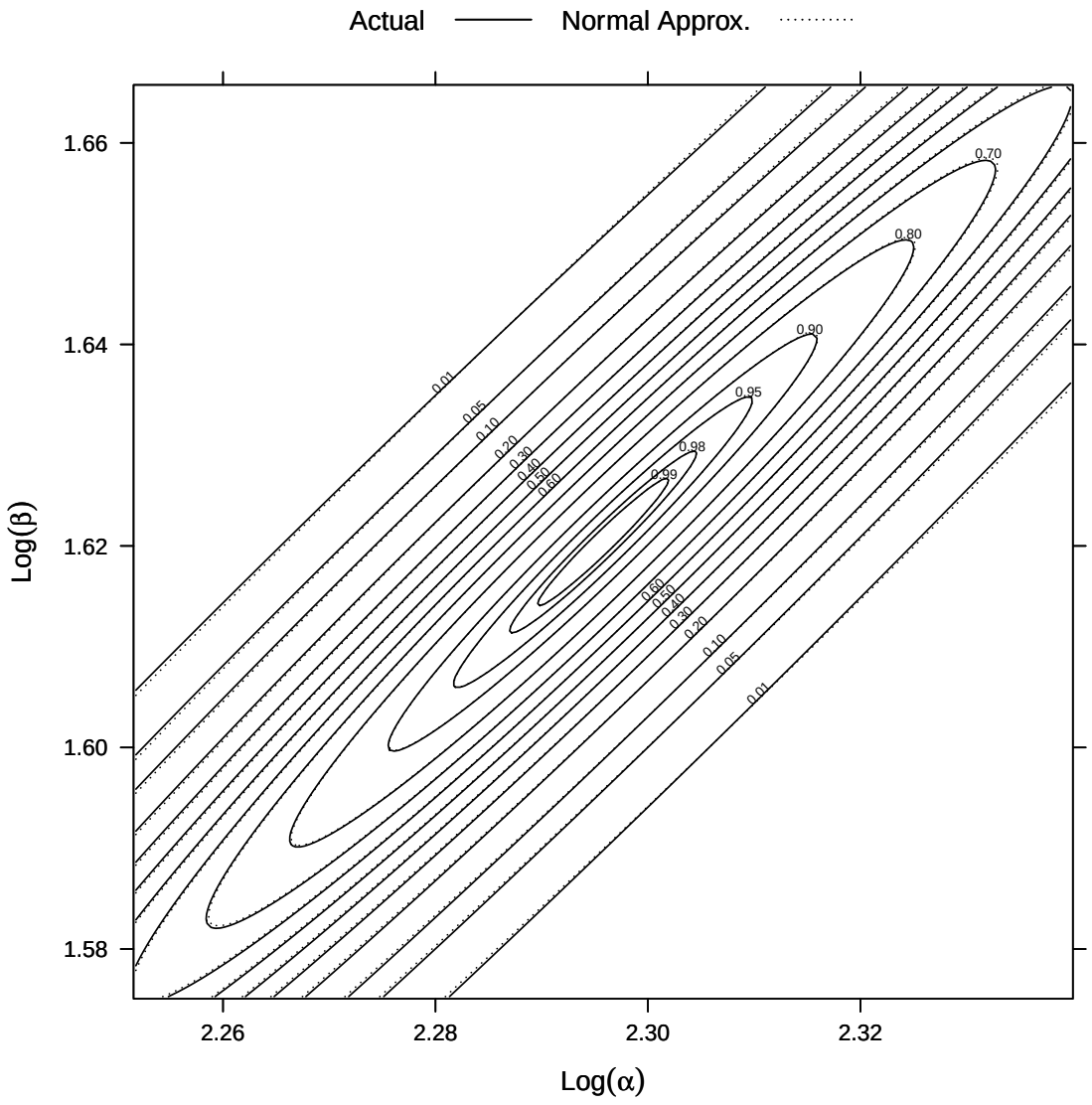

Figure B.1: Normal approximation to the conditional posterior of $(\log \alpha_\sigma, \log \beta_\sigma)$. Contours of the log-conditional posterior and proposal for $\alpha_\sigma = 10$ and $\beta_\sigma = 5$ with $n = 1000$ observations.

B.2.3 Updates for censoring model parameters

Gibbs update for π^{rnd}

The conditional posterior distribution of the random censoring probability is straightforward and conjugate, enabling a direct update:

$$\pi^{rnd}|\vec{M}, \vec{\Theta}_{-\pi^{rnd}} \sim Beta\left(\sum_{ikl} R_{ikl}, \sum_{ikl}(1 - R_{ikl})\right).$$

Metropolis-Hastings update for intensity-based censoring parameters

The parameters of the intensity-based censoring model, $\vec{\eta}$, are updated using an independence chain Metropolis-Hastings sampler. We first obtain the mode of the conditional posterior of $\vec{\eta}$, $\hat{\vec{\eta}}$, using standard GLM estimation techniques (Fisher scoring). This estimation naturally yields the expected negative Hessian of this conditional posterior $\mathcal{I}(\hat{\vec{\eta}})$. We then draw proposal as

$$\vec{\eta}^* = \hat{\vec{\eta}} + \sqrt{\frac{\nu}{\nu - 2}}\mathcal{I}(\hat{\vec{\eta}})^{-1/2}\vec{t}_\nu,$$

where $\vec{t}_\nu$ is defined as before.

B.2.4 Metropolis-Hastings update for number of states parameters (r, λ)

The conditional posterior distribution of (r, λ) is not of conjugate form regardless of the prior distribution. For our inference, we take $r \sim LogNormal(\mu_{0r}, \sigma^2_{0r})$ independent of $\lambda \sim Beta(\alpha_{0\lambda}, \beta_{0\lambda})$ a priori. We construct a proposal for these parameters by first transforming them to $(\log r, \text{logit}\,\lambda)$. We then estimate a normal approximation to the joint conditional posterior of these transformed parameters.

As in Section B.2.2, we can analytically maximize the conditional posterior over one

parameter given the other. Maximizing over $\log r$ yields

$$\widehat{\operatorname{logit}\lambda} = \frac{\sum_{ik} r + \alpha_{or}}{\sum_{ik} r + \sum_{ik}(s_{ik} - 1) + \alpha_{or} + \beta_{or}}.$$

This reduces estimation of the conditional posterior mode to univariate optimization over $\operatorname{logit} r$. Following this optimization, we compute the negative Hessian of the log-conditional posteriors $\hat{\mathcal{I}}(\widehat{\log r}, \widehat{\operatorname{logit}\lambda})$ and propose as

$$(\log r^*, \operatorname{logit}\lambda^*) \mid \vec{s} \sim \begin{pmatrix} \widehat{\log r} \\ \widehat{\operatorname{logit}\lambda} \end{pmatrix} + \sqrt{\frac{\nu}{\nu - 2}}\hat{\mathcal{I}}(\widehat{\log r}, \widehat{\operatorname{logit}\lambda})^{-1/2}\vec{t}_\nu.$$

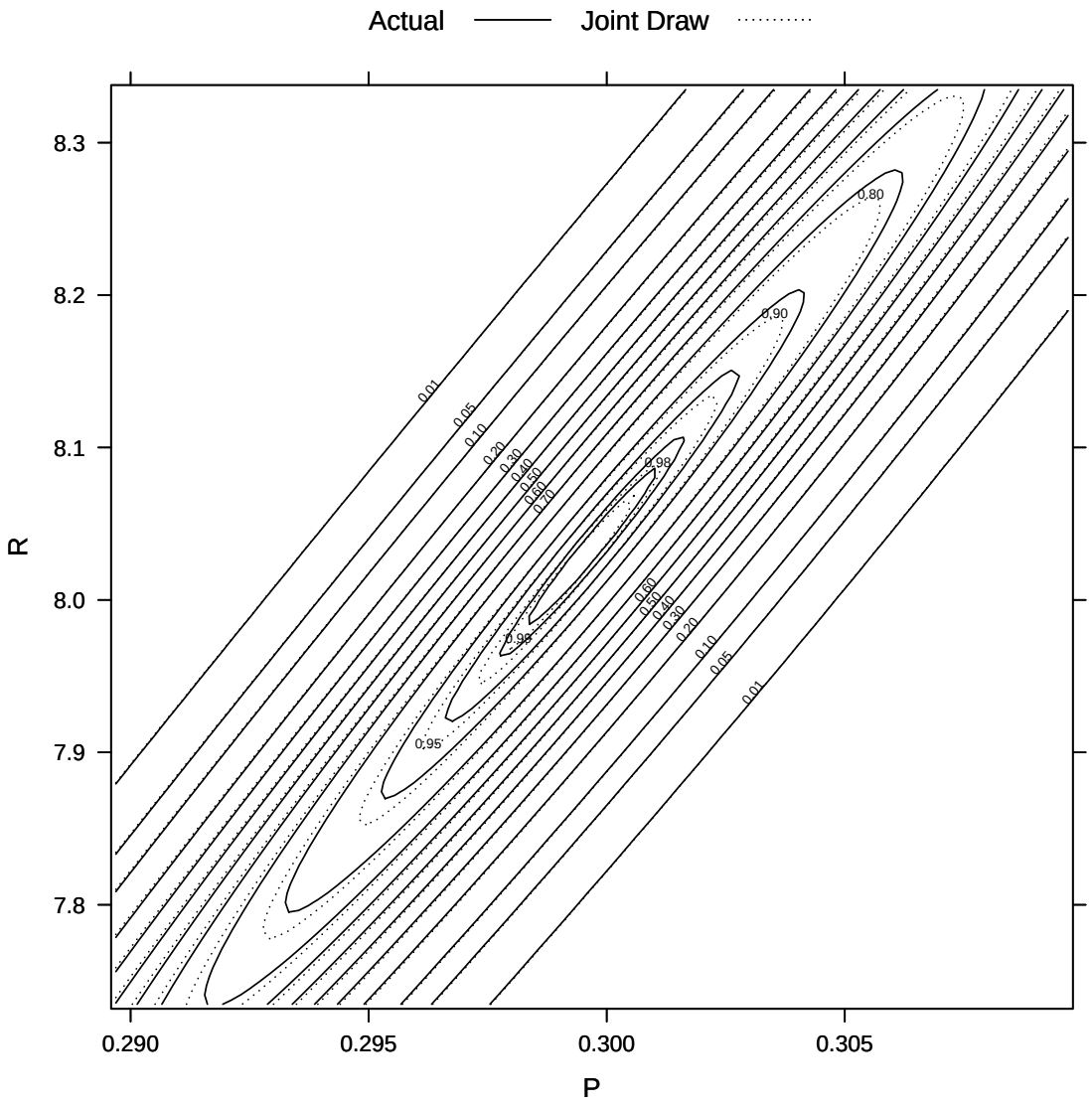

Figure B.2: Conditional posterior contours for (r, λ).

B.3 Additional simulation results

Table B.1: Simulation analysis of protein abundance estimates by ζ_i and m_i, 90m gradient. All estimates are $\pm$ one standard deviation.

Grad.	ζ_i	m_i	Posterior Mean	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
90m	3	20	3.30 ± 0.33	2.33 ± 0.92	3.78 ± 0.87	4.85 ± 0.77	3.87 ± 0.84	5.98 ± 0.16
90m	3	30	3.19 ± 0.33	2.38 ± 0.78	3.79 ± 0.73	4.84 ± 0.67	3.89 ± 0.72	5.84 ± 0.19
90m	3	40	3.09 ± 0.35	2.46 ± 0.81	3.84 ± 0.73	4.89 ± 0.64	3.94 ± 0.72	5.75 ± 0.20
90m	3	50	3.03 ± 0.34	2.40 ± 0.76	3.73 ± 0.66	4.76 ± 0.62	3.83 ± 0.65	5.68 ± 0.21
90m	3	60	2.99 ± 0.38	2.56 ± 0.85	3.86 ± 0.75	4.86 ± 0.64	3.95 ± 0.74	5.64 ± 0.24
90m	4	20	3.91 ± 0.38	3.23 ± 0.68	4.37 ± 0.60	5.29 ± 0.52	4.46 ± 0.59	6.32 ± 0.24
90m	4	30	3.87 ± 0.39	3.36 ± 0.65	4.37 ± 0.53	5.22 ± 0.45	4.45 ± 0.52	6.28 ± 0.26
90m	4	40	3.87 ± 0.38	3.57 ± 0.64	4.46 ± 0.53	5.26 ± 0.40	4.54 ± 0.51	6.27 ± 0.26
90m	4	50	3.87 ± 0.38	3.64 ± 0.58	4.45 ± 0.49	5.26 ± 0.37	4.53 ± 0.47	6.25 ± 0.25
90m	4	60	3.87 ± 0.33	3.76 ± 0.55	4.48 ± 0.46	5.24 ± 0.33	4.56 ± 0.45	6.26 ± 0.22
90m	5	20	4.92 ± 0.33	4.34 ± 0.52	5.02 ± 0.48	5.67 ± 0.29	5.07 ± 0.47	6.92 ± 0.20
90m	5	30	4.95 ± 0.28	4.61 ± 0.49	5.10 ± 0.46	5.69 ± 0.27	5.16 ± 0.45	6.93 ± 0.16
90m	5	40	4.96 ± 0.22	4.81 ± 0.50	5.17 ± 0.48	5.68 ± 0.23	5.22 ± 0.47	6.94 ± 0.13
90m	5	50	4.97 ± 0.20	4.89 ± 0.44	5.16 ± 0.42	5.69 ± 0.22	5.21 ± 0.42	6.94 ± 0.12
90m	5	60	4.98 ± 0.21	4.97 ± 0.43	5.16 ± 0.41	5.66 ± 0.19	5.21 ± 0.41	6.94 ± 0.12
90m	6	20	5.95 ± 0.28	5.48 ± 0.50	5.89 ± 0.49	6.24 ± 0.26	5.91 ± 0.48	7.36 ± 0.14
90m	6	30	5.94 ± 0.21	5.66 ± 0.45	5.89 ± 0.46	6.23 ± 0.23	5.91 ± 0.46	7.36 ± 0.11
90m	6	40	5.97 ± 0.21	5.83 ± 0.45	5.94 ± 0.45	6.25 ± 0.21	5.96 ± 0.45	7.36 ± 0.10
90m	6	50	5.97 ± 0.16	5.96 ± 0.45	5.96 ± 0.46	6.25 ± 0.18	5.98 ± 0.46	7.36 ± 0.09
90m	6	60	5.99 ± 0.16	6.04 ± 0.42	5.96 ± 0.43	6.25 ± 0.18	5.98 ± 0.43	7.37 ± 0.08
90m	7	20	6.94 ± 0.24	6.45 ± 0.47	6.76 ± 0.47	7.03 ± 0.25	6.77 ± 0.46	7.54 ± 0.10
90m	7	30	6.94 ± 0.21	6.65 ± 0.43	6.78 ± 0.43	7.01 ± 0.22	6.79 ± 0.43	7.56 ± 0.09
90m	7	40	6.92 ± 0.17	6.79 ± 0.43	6.81 ± 0.43	6.99 ± 0.19	6.81 ± 0.43	7.55 ± 0.08
90m	7	50	6.94 ± 0.16	6.96 ± 0.46	6.88 ± 0.46	7.00 ± 0.18	6.88 ± 0.46	7.55 ± 0.07
90m	7	60	6.94 ± 0.15	7.03 ± 0.42	6.87 ± 0.42	7.01 ± 0.16	6.87 ± 0.42	7.55 ± 0.07
90m	8	20	7.94 ± 0.21	7.46 ± 0.45	7.75 ± 0.42	7.95 ± 0.24	7.75 ± 0.42	7.59 ± 0.09
90m	8	30	7.93 ± 0.18	7.69 ± 0.40	7.81 ± 0.39	7.93 ± 0.21	7.81 ± 0.39	7.59 ± 0.08
90m	8	40	7.94 ± 0.15	7.83 ± 0.40	7.82 ± 0.39	7.94 ± 0.17	7.82 ± 0.39	7.60 ± 0.07
90m	8	50	7.95 ± 0.14	7.94 ± 0.38	7.83 ± 0.39	7.96 ± 0.17	7.83 ± 0.39	7.60 ± 0.06
90m	8	60	7.95 ± 0.13	8.01 ± 0.34	7.82 ± 0.36	7.96 ± 0.15	7.83 ± 0.36	7.60 ± 0.06

Table B.2: Simulation analysis of protein abundance estimates by ζ_i and m_i, 180m gradient. All estimates are $\pm$ one standard deviation.

Grad.	ζ_i	m_i	Posterior Mean	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
180m	3	20	3.13 ± 0.42	2.08 ± 0.85	3.46 ± 0.80	4.75 ± 0.70	3.59 ± 0.79	6.02 ± 0.21
180m	3	30	2.97 ± 0.44	2.13 ± 0.89	3.47 ± 0.80	4.74 ± 0.68	3.60 ± 0.79	5.88 ± 0.23
180m	3	40	2.93 ± 0.43	2.30 ± 0.87	3.56 ± 0.76	4.77 ± 0.64	3.69 ± 0.76	5.83 ± 0.25
180m	3	50	2.91 ± 0.44	2.35 ± 0.86	3.57 ± 0.73	4.77 ± 0.60	3.69 ± 0.72	5.78 ± 0.28
180m	3	60	2.90 ± 0.45	2.45 ± 0.86	3.59 ± 0.72	4.77 ± 0.58	3.72 ± 0.72	5.77 ± 0.28
180m	4	20	3.89 ± 0.47	3.13 ± 0.77	4.15 ± 0.67	5.22 ± 0.49	4.26 ± 0.66	6.44 ± 0.29
180m	4	30	3.89 ± 0.39	3.35 ± 0.70	4.20 ± 0.62	5.15 ± 0.42	4.30 ± 0.61	6.44 ± 0.25
180m	4	40	3.91 ± 0.36	3.59 ± 0.74	4.30 ± 0.66	5.17 ± 0.38	4.41 ± 0.65	6.44 ± 0.22
180m	4	50	3.93 ± 0.33	3.65 ± 0.64	4.26 ± 0.57	5.18 ± 0.32	4.37 ± 0.57	6.44 ± 0.20
180m	4	60	3.93 ± 0.33	3.77 ± 0.62	4.30 ± 0.55	5.17 ± 0.33	4.41 ± 0.54	6.44 ± 0.20
180m	5	20	4.95 ± 0.33	4.27 ± 0.58	4.87 ± 0.55	5.63 ± 0.29	4.93 ± 0.54	7.01 ± 0.20
180m	5	30	4.97 ± 0.28	4.59 ± 0.61	5.01 ± 0.60	5.68 ± 0.27	5.07 ± 0.59	7.01 ± 0.15
180m	5	40	4.98 ± 0.22	4.70 ± 0.55	4.98 ± 0.55	5.64 ± 0.24	5.05 ± 0.54	7.02 ± 0.12
180m	5	50	4.99 ± 0.19	4.86 ± 0.57	5.04 ± 0.57	5.65 ± 0.23	5.11 ± 0.57	7.03 ± 0.11
180m	5	60	5.01 ± 0.20	4.89 ± 0.53	5.00 ± 0.53	5.65 ± 0.23	5.07 ± 0.52	7.01 ± 0.10
180m	6	20	5.96 ± 0.28	5.35 ± 0.58	5.74 ± 0.58	6.23 ± 0.28	5.77 ± 0.57	7.37 ± 0.14
180m	6	30	5.98 ± 0.22	5.61 ± 0.56	5.81 ± 0.57	6.24 ± 0.25	5.84 ± 0.56	7.39 ± 0.11
180m	6	40	5.96 ± 0.18	5.68 ± 0.50	5.75 ± 0.50	6.21 ± 0.20	5.78 ± 0.50	7.39 ± 0.09
180m	6	50	5.99 ± 0.18	5.84 ± 0.50	5.82 ± 0.50	6.23 ± 0.19	5.85 ± 0.50	7.38 ± 0.09
180m	6	60	5.99 ± 0.16	5.90 ± 0.51	5.80 ± 0.52	6.22 ± 0.17	5.83 ± 0.52	7.38 ± 0.08
180m	7	20	6.96 ± 0.23	6.36 ± 0.54	6.68 ± 0.55	7.02 ± 0.26	6.69 ± 0.55	7.53 ± 0.11
180m	7	30	6.94 ± 0.21	6.61 ± 0.57	6.75 ± 0.58	7.03 ± 0.23	6.76 ± 0.58	7.53 ± 0.08
180m	7	40	6.94 ± 0.17	6.70 ± 0.52	6.71 ± 0.54	7.01 ± 0.20	6.72 ± 0.54	7.53 ± 0.07
180m	7	50	6.94 ± 0.16	6.84 ± 0.54	6.75 ± 0.55	7.01 ± 0.18	6.75 ± 0.55	7.53 ± 0.07
180m	7	60	6.95 ± 0.15	6.95 ± 0.52	6.78 ± 0.54	7.02 ± 0.17	6.79 ± 0.54	7.53 ± 0.06
180m	8	20	7.91 ± 0.21	7.33 ± 0.56	7.63 ± 0.54	7.92 ± 0.25	7.63 ± 0.54	7.57 ± 0.09
180m	8	30	7.94 ± 0.18	7.57 ± 0.54	7.69 ± 0.53	7.95 ± 0.20	7.69 ± 0.52	7.58 ± 0.07
180m	8	40	7.94 ± 0.14	7.74 ± 0.46	7.73 ± 0.47	7.95 ± 0.18	7.73 ± 0.47	7.57 ± 0.07
180m	8	50	7.94 ± 0.14	7.86 ± 0.48	7.76 ± 0.50	7.95 ± 0.16	7.75 ± 0.50	7.57 ± 0.06
180m	8	60	7.94 ± 0.13	7.94 ± 0.43	7.75 ± 0.46	7.94 ± 0.16	7.75 ± 0.46	7.57 ± 0.05

Table B.3: Simulation analysis of protein abundance estimates by ζ_i and m_i, 360m gradient. All estimates are $\pm$ one standard deviation.

Grad.	ζ_i	m_i	Posterior Mean	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
360m	3	20	3.07 ± 0.34	2.25 ± 0.67	3.59 ± 0.58	4.63 ± 0.52	3.78 ± 0.57	6.02 ± 0.24
360m	3	30	3.00 ± 0.37	2.32 ± 0.70	3.57 ± 0.57	4.57 ± 0.44	3.76 ± 0.54	5.91 ± 0.27
360m	3	40	2.94 ± 0.37	2.45 ± 0.67	3.65 ± 0.54	4.64 ± 0.41	3.84 ± 0.52	5.84 ± 0.28
360m	3	50	2.97 ± 0.36	2.55 ± 0.68	3.65 ± 0.52	4.61 ± 0.41	3.84 ± 0.51	5.83 ± 0.28
360m	3	60	2.97 ± 0.38	2.65 ± 0.67	3.69 ± 0.51	4.62 ± 0.37	3.87 ± 0.49	5.83 ± 0.30
360m	4	20	3.97 ± 0.33	3.36 ± 0.52	4.17 ± 0.45	4.95 ± 0.30	4.30 ± 0.43	6.64 ± 0.24
360m	4	30	3.96 ± 0.28	3.59 ± 0.47	4.23 ± 0.42	4.92 ± 0.25	4.36 ± 0.40	6.63 ± 0.20
360m	4	40	4.01 ± 0.25	3.80 ± 0.49	4.29 ± 0.45	4.95 ± 0.23	4.41 ± 0.44	6.67 ± 0.18
360m	4	50	3.98 ± 0.21	3.85 ± 0.45	4.26 ± 0.41	4.92 ± 0.20	4.39 ± 0.40	6.64 ± 0.15
360m	4	60	4.03 ± 0.20	3.94 ± 0.43	4.26 ± 0.39	4.92 ± 0.21	4.39 ± 0.37	6.66 ± 0.14
360m	5	20	4.99 ± 0.24	4.46 ± 0.42	4.90 ± 0.41	5.35 ± 0.22	4.96 ± 0.41	7.21 ± 0.15
360m	5	30	4.99 ± 0.20	4.66 ± 0.41	4.93 ± 0.40	5.36 ± 0.20	4.99 ± 0.40	7.20 ± 0.12
360m	5	40	5.00 ± 0.17	4.85 ± 0.41	4.99 ± 0.41	5.36 ± 0.18	5.04 ± 0.40	7.21 ± 0.10
360m	5	50	5.00 ± 0.16	4.94 ± 0.41	4.98 ± 0.41	5.35 ± 0.16	5.04 ± 0.40	7.21 ± 0.10
360m	5	60	5.05 ± 0.14	5.05 ± 0.39	5.01 ± 0.39	5.37 ± 0.16	5.07 ± 0.38	7.21 ± 0.08
360m	6	20	5.98 ± 0.22	5.52 ± 0.47	5.85 ± 0.47	6.07 ± 0.24	5.86 ± 0.46	7.44 ± 0.11
360m	6	30	5.96 ± 0.19	5.71 ± 0.44	5.86 ± 0.44	6.06 ± 0.20	5.87 ± 0.43	7.43 ± 0.10
360m	6	40	5.96 ± 0.17	5.81 ± 0.39	5.84 ± 0.38	6.06 ± 0.16	5.85 ± 0.38	7.43 ± 0.08
360m	6	50	5.95 ± 0.15	5.94 ± 0.40	5.87 ± 0.40	6.04 ± 0.16	5.89 ± 0.40	7.43 ± 0.07
360m	6	60	5.97 ± 0.15	6.05 ± 0.41	5.90 ± 0.42	6.05 ± 0.16	5.91 ± 0.41	7.43 ± 0.07
360m	7	20	6.95 ± 0.22	6.52 ± 0.44	6.82 ± 0.43	6.98 ± 0.26	6.82 ± 0.43	7.48 ± 0.10
360m	7	30	6.96 ± 0.16	6.72 ± 0.44	6.85 ± 0.43	6.99 ± 0.19	6.86 ± 0.43	7.48 ± 0.08
360m	7	40	6.94 ± 0.15	6.85 ± 0.38	6.85 ± 0.38	6.96 ± 0.19	6.85 ± 0.38	7.48 ± 0.07
360m	7	50	6.95 ± 0.13	6.97 ± 0.41	6.88 ± 0.41	6.97 ± 0.16	6.89 ± 0.41	7.48 ± 0.07
360m	7	60	6.96 ± 0.12	7.06 ± 0.41	6.89 ± 0.41	6.98 ± 0.14	6.89 ± 0.41	7.48 ± 0.06
360m	8	20	7.94 ± 0.17	7.45 ± 0.42	7.75 ± 0.39	7.96 ± 0.21	7.76 ± 0.38	7.49 ± 0.09
360m	8	30	7.96 ± 0.15	7.71 ± 0.38	7.83 ± 0.37	7.97 ± 0.19	7.84 ± 0.36	7.48 ± 0.08
360m	8	40	7.96 ± 0.13	7.88 ± 0.36	7.88 ± 0.35	7.96 ± 0.16	7.88 ± 0.35	7.48 ± 0.07
360m	8	50	7.97 ± 0.13	7.96 ± 0.34	7.86 ± 0.35	7.97 ± 0.14	7.86 ± 0.34	7.49 ± 0.06
360m	8	60	7.96 ± 0.12	8.05 ± 0.33	7.87 ± 0.34	7.97 ± 0.14	7.88 ± 0.34	7.48 ± 0.06

Table B.4: Coverage of HPD posterior intervals for ζ_i in simulation study, 90m gradient

Grad.	ζ_i	m_i	68%	90%	95%	99%
90m	3	20	0.69	0.94	0.97	0.98
90m	3	30	0.71	0.92	0.95	0.99
90m	3	40	0.75	0.92	0.96	0.99
90m	3	50	0.79	0.93	0.95	0.98
90m	3	60	0.75	0.94	0.97	0.99
90m	4	20	0.74	0.95	0.97	0.99
90m	4	30	0.70	0.92	0.96	0.98
90m	4	40	0.67	0.88	0.93	0.98
90m	4	50	0.63	0.88	0.93	0.97
90m	4	60	0.70	0.91	0.95	0.97
90m	5	20	0.67	0.88	0.94	0.99
90m	5	30	0.63	0.86	0.92	0.99
90m	5	40	0.71	0.93	0.96	1.00
90m	5	50	0.71	0.90	0.97	0.99
90m	5	60	0.64	0.85	0.91	0.98
90m	6	20	0.63	0.86	0.93	0.99
90m	6	30	0.65	0.89	0.94	0.98
90m	6	40	0.65	0.86	0.92	0.98
90m	6	50	0.67	0.90	0.96	0.99
90m	6	60	0.67	0.91	0.95	0.98
90m	7	20	0.66	0.90	0.94	0.99
90m	7	30	0.63	0.86	0.90	0.98
90m	7	40	0.61	0.85	0.93	0.98
90m	7	50	0.65	0.88	0.94	0.99
90m	7	60	0.65	0.87	0.93	0.97
90m	8	20	0.65	0.88	0.93	0.99
90m	8	30	0.63	0.88	0.93	0.99
90m	8	40	0.65	0.89	0.94	0.98
90m	8	50	0.65	0.89	0.92	0.98
90m	8	60	0.64	0.87	0.92	0.98

Table B.5: Coverage of HPD posterior intervals for ζ_i in simulation study, 180m gradient

Grad.	ζ_i	m_i	68%	90%	95%	99%
180m	3	20	0.71	0.91	0.97	0.99
180m	3	30	0.78	0.91	0.96	0.99
180m	3	40	0.77	0.93	0.97	0.99
180m	3	50	0.72	0.92	0.96	0.99
180m	3	60	0.66	0.90	0.93	0.98
180m	4	20	0.66	0.89	0.95	0.99
180m	4	30	0.66	0.89	0.95	0.98
180m	4	40	0.68	0.88	0.93	0.97
180m	4	50	0.63	0.87	0.93	0.97
180m	4	60	0.59	0.86	0.91	0.97
180m	5	20	0.67	0.87	0.93	0.97
180m	5	30	0.68	0.90	0.94	0.98
180m	5	40	0.69	0.90	0.95	0.99
180m	5	50	0.69	0.90	0.94	0.99
180m	5	60	0.66	0.87	0.94	0.98
180m	6	20	0.65	0.88	0.94	0.98
180m	6	30	0.65	0.92	0.95	1.00
180m	6	40	0.71	0.88	0.94	0.99
180m	6	50	0.68	0.86	0.94	0.98
180m	6	60	0.71	0.89	0.93	1.00
180m	7	20	0.68	0.91	0.97	0.99
180m	7	30	0.69	0.88	0.94	0.97
180m	7	40	0.64	0.87	0.93	0.98
180m	7	50	0.65	0.87	0.92	0.97
180m	7	60	0.66	0.88	0.93	0.98
180m	8	20	0.65	0.88	0.94	0.99
180m	8	30	0.62	0.87	0.93	0.98
180m	8	40	0.68	0.90	0.94	0.98
180m	8	50	0.68	0.87	0.93	0.97
180m	8	60	0.60	0.86	0.94	0.98

Table B.6: Coverage of HPD posterior intervals for ζ_i in simulation study, 360m gradient

Grad.	ζ_i	m_i	68%	90%	95%	99%
360m	3	20	0.75	0.91	0.95	0.99
360m	3	30	0.73	0.90	0.93	0.98
360m	3	40	0.72	0.93	0.97	0.99
360m	3	50	0.68	0.89	0.94	0.98
360m	3	60	0.64	0.85	0.91	0.97
360m	4	20	0.64	0.88	0.93	0.98
360m	4	30	0.62	0.88	0.94	0.99
360m	4	40	0.62	0.85	0.91	0.97
360m	4	50	0.67	0.88	0.94	0.99
360m	4	60	0.60	0.87	0.93	0.98
360m	5	20	0.70	0.89	0.95	0.99
360m	5	30	0.69	0.91	0.94	0.99
360m	5	40	0.68	0.90	0.95	0.98
360m	5	50	0.62	0.89	0.95	0.99
360m	5	60	0.67	0.86	0.94	0.98
360m	6	20	0.65	0.89	0.95	0.99
360m	6	30	0.67	0.88	0.94	0.99
360m	6	40	0.66	0.87	0.94	0.99
360m	6	50	0.63	0.89	0.93	0.98
360m	6	60	0.59	0.86	0.92	0.96
360m	7	20	0.64	0.87	0.94	0.98
360m	7	30	0.69	0.91	0.95	0.99
360m	7	40	0.67	0.87	0.94	0.97
360m	7	50	0.68	0.89	0.93	0.98
360m	7	60	0.65	0.89	0.93	0.98
360m	8	20	0.64	0.92	0.96	0.99
360m	8	30	0.66	0.88	0.93	0.98
360m	8	40	0.66	0.91	0.94	0.98
360m	8	50	0.65	0.89	0.94	0.97
360m	8	60	0.61	0.86	0.92	0.99

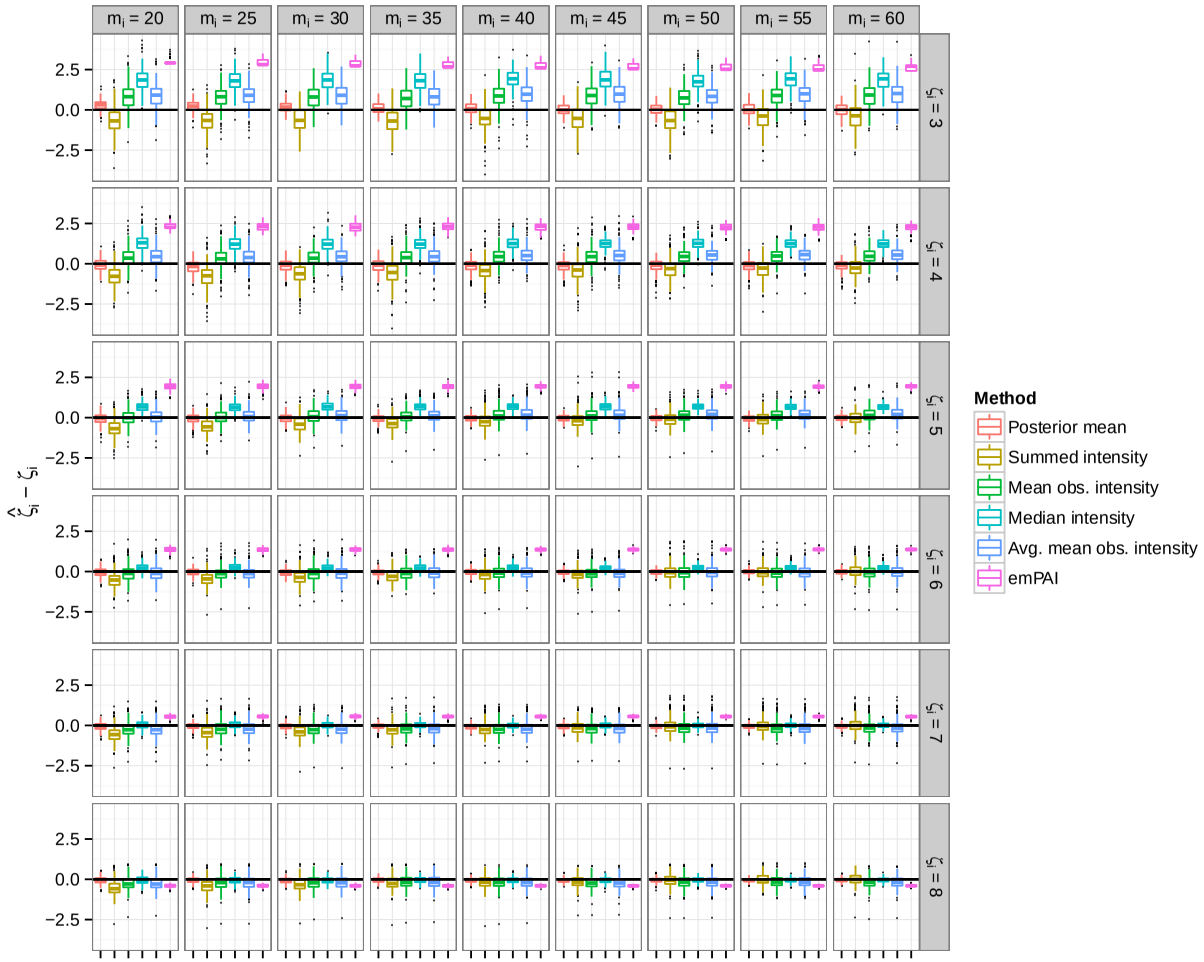

Figure B.3: Simulation results for 90m gradient. Boxplots showing the distribution of the estimates across replicates by m_i and ζ_i. Each box contains one box plot per method summarizing the fit for one combination of abundance and number of peptides. The y-axis for all plots is $\hat{\zeta}_i - \zeta$, with a black horizontal line at zero.

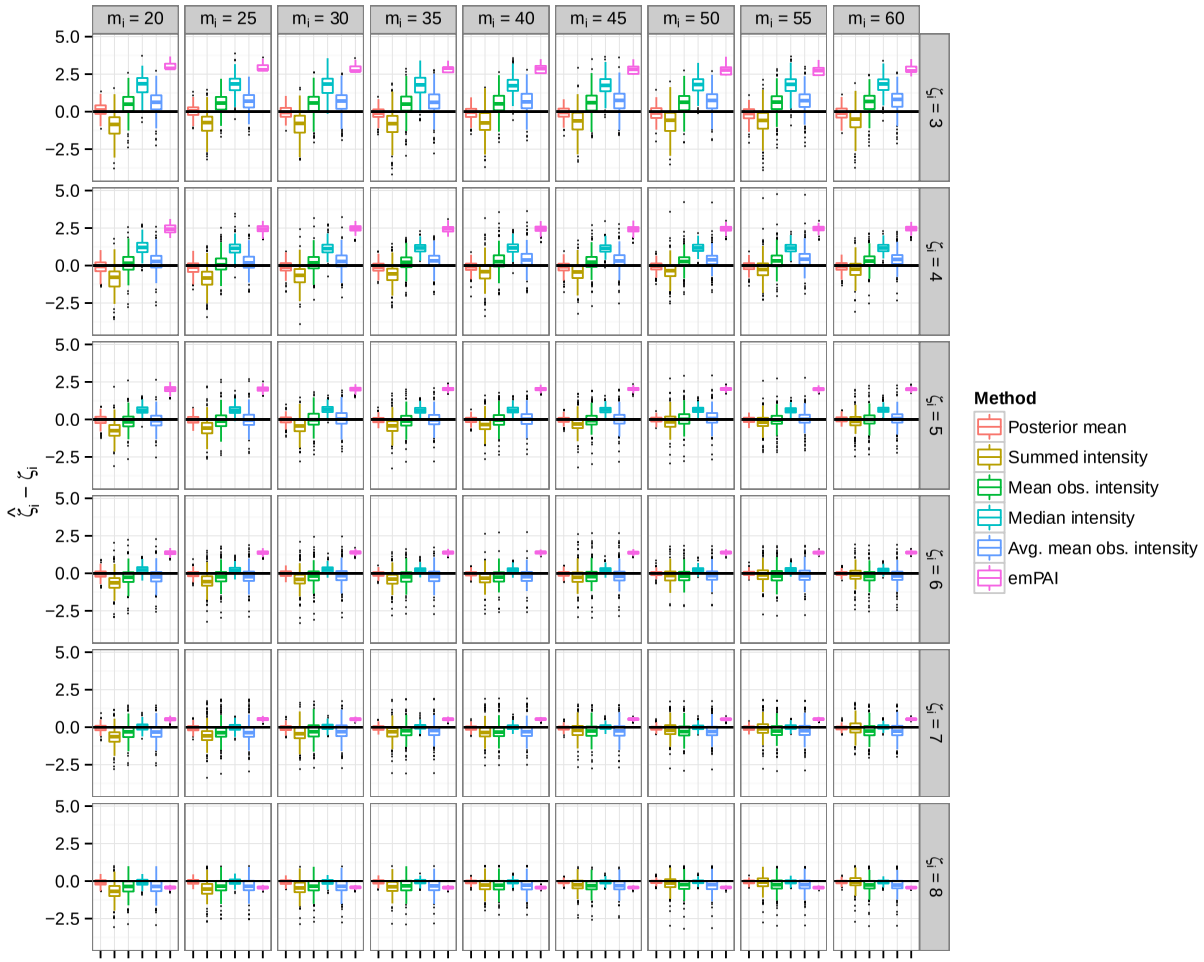

Figure B.4: Simulation results for 180m gradient. Boxplots showing the distribution of the estimates across replicates by m_i and ζ_i. Each box contains one box plot per method summarizing the fit for one combination of abundance and number of peptides. The y-axis for all plots is $\hat{\zeta}_i - \zeta$, with a black horizontal line at zero.

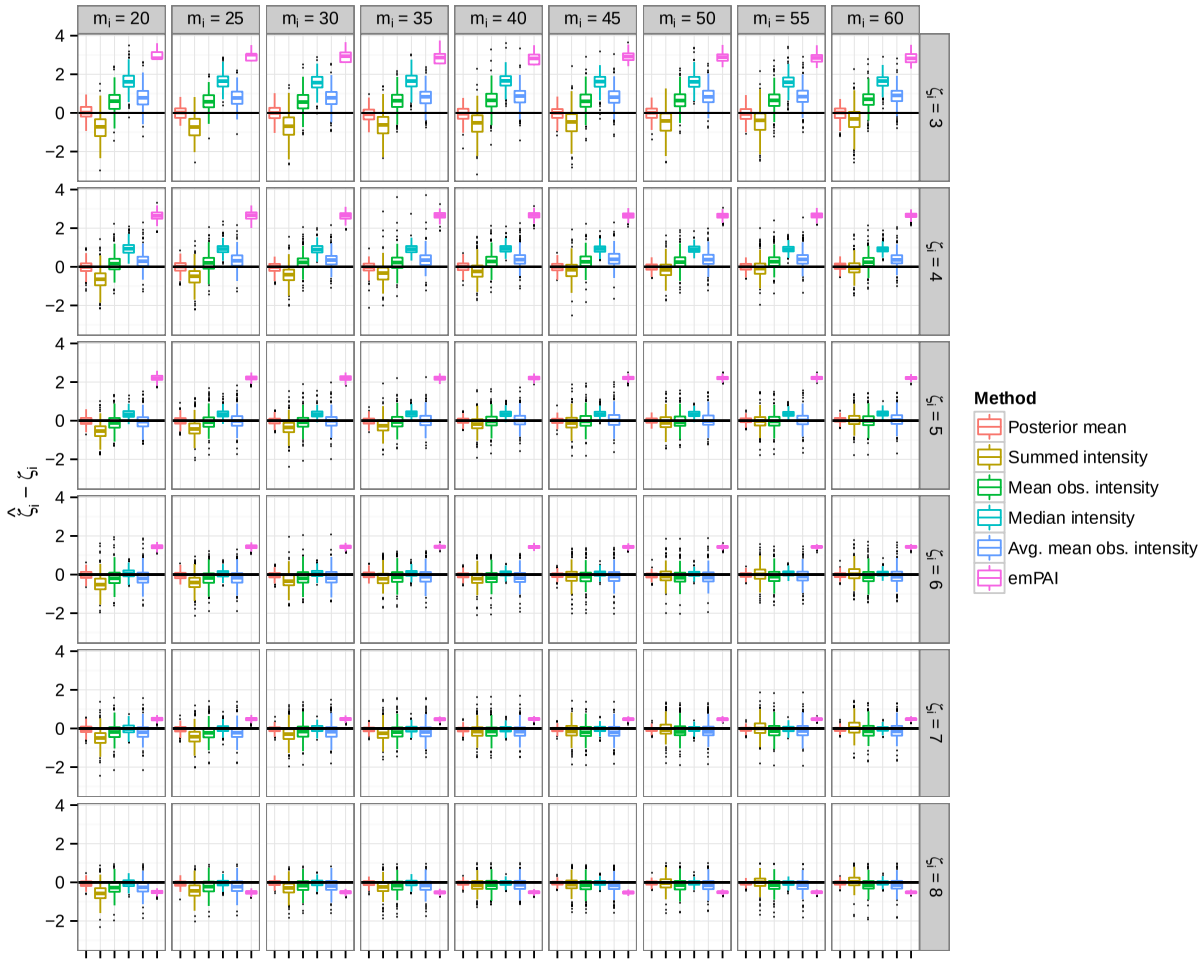

Figure B.5: Simulation results for 360m gradient. Boxplots showing the distribution of the estimates across replicates by m_i and ζ_i. Each box contains one box plot per method summarizing the fit for one combination of abundance and number of peptides. The y-axis for all plots is $\hat{\zeta}_i - \zeta$, with a black horizontal line at zero.

B.4 Additional empirical results

Table B.7: Detailed empirical results for UPS2 experiments, 90m gradients

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P02788ups	-0.30	90m	1	0.04	$(-1.26, 1.40)$	1.53	2.58	2.92	2.71	2.37
O00762ups	0.70	90m	1	1.11	$(-0.31, 2.43)$	1.85	2.90	3.24	3.03	2.96
P02787ups	0.70	90m	1	0.75	$(-0.19, 1.69)$	2.12	2.87	3.21	3.00	2.71
P06396ups	0.70	90m	1	0.55	$(-1.06, 2.56)$	3.00	3.75	3.49	3.58	2.76
O76070ups	1.70	90m	1	2.23	$(1.29, 3.20)$	1.84	2.42	2.75	2.55	3.67
P01008ups	1.70	90m	1	2.03	$(1.37, 2.61)$	2.36	2.56	2.77	2.64	3.54
P01344ups	1.70	90m	1	2.87	$(1.88, 3.95)$	2.67	3.42	3.68	3.37	3.85
P08263ups	1.70	90m	1	2.54	$(1.81, 3.17)$	2.38	2.65	2.73	2.78	3.90
P55957ups	1.70	90m	1	2.31	$(1.42, 3.33)$	2.06	2.63	2.97	2.77	3.67
P00709ups	2.70	90m	1	3.00	$(2.18, 3.86)$	2.53	2.98	3.22	3.11	4.00
P02753ups	2.70	90m	1	2.40	$(1.50, 3.37)$	2.19	2.76	3.19	2.89	3.73
P06732ups	2.70	90m	1	2.59	$(1.96, 3.30)$	3.18	3.33	3.63	3.46	3.68
P12081ups	2.70	90m	1	3.21	$(2.67, 3.74)$	3.69	3.57	3.57	3.64	3.90
P16083ups	2.70	90m	1	2.55	$(1.72, 3.39)$	3.02	3.37	3.26	3.50	3.73
P61626ups	2.70	90m	1	2.31	$(1.40, 3.18)$	1.99	2.56	2.92	2.70	3.67
P63279ups	2.70	90m	1	2.73	$(1.74, 3.68)$	2.62	3.19	2.89	3.33	3.81

Table B.7: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
Q15843ups	2.70	90m	1	1.77	(0.38, 3.32)	1.95	3.00	3.34	3.14	3.26
P00167ups	3.70	90m	1	4.58	(4.03, 5.17)	4.26	4.47	4.56	4.45	4.37
P01133ups	3.70	90m	1	4.08	(2.90, 5.34)	3.25	4.00	4.34	4.13	4.23
P02144ups	3.70	90m	1	3.46	(2.69, 4.27)	4.01	4.28	4.32	4.19	4.00
P04040ups	3.70	90m	1	4.10	(3.68, 4.52)	4.53	4.23	4.20	4.28	4.23
P15559ups	3.70	90m	1	3.69	(2.87, 4.52)	3.88	4.15	4.41	4.16	3.90
P62937ups	3.70	90m	1	3.17	(2.54, 3.81)	3.52	3.72	3.16	3.75	4.00
P63165ups	3.70	90m	1	3.41	(2.83, 4.01)	4.04	3.94	4.21	4.08	4.06
Q06830ups	3.70	90m	1	4.41	(3.92, 4.94)	4.41	4.38	4.45	4.39	4.46
P00915ups	4.70	90m	1	4.64	(4.21, 5.05)	4.79	4.69	4.61	4.67	4.55
P00918ups	4.70	90m	1	4.50	(4.05, 4.99)	4.84	4.72	4.73	4.71	4.49
P01031ups	4.70	90m	1	4.21	(3.54, 4.87)	4.26	4.53	3.95	4.54	4.62
P02768ups	4.70	90m	1	4.14	(3.78, 4.58)	4.91	4.46	4.32	4.50	4.37
P41159ups	4.70	90m	1	3.85	(3.23, 4.46)	4.13	4.28	3.82	4.27	4.50
P62988ups	4.70	90m	1	3.49	(2.79, 4.21)	3.58	3.85	3.37	3.81	4.34
P68871ups	4.70	90m	1	4.62	(4.11, 5.20)	4.67	4.77	4.66	4.74	4.42

Table B.7: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P69905ups	4.70	90m	1	4.75	(4.27, 5.29)	4.43	4.64	4.83	4.61	4.48
P02788ups	-0.30	90m	2	0.75	(−0.04, 1.38)	1.43	2.53	2.94	2.73	2.26
P01112ups	0.70	90m	2	2.10	(1.23, 2.67)	1.68	2.48	2.89	2.68	3.28
P02787ups	0.70	90m	2	1.36	(0.78, 2.00)	1.97	2.46	2.46	2.66	2.93
P06396ups	0.70	90m	2	0.74	(−0.05, 1.66)	1.90	2.70	3.11	2.90	2.65
O76070ups	1.70	90m	2	2.69	(2.13, 3.27)	2.16	2.56	2.69	2.68	3.90
P01008ups	1.70	90m	2	2.58	(2.19, 2.98)	2.41	2.55	2.74	2.66	3.58
P01344ups	1.70	90m	2	3.26	(2.47, 4.04)	2.51	3.30	3.63	3.33	3.74
P08263ups	1.70	90m	2	2.36	(1.87, 3.00)	2.26	2.66	3.05	2.86	3.67
P55957ups	1.70	90m	2	1.07	(0.07, 2.43)	1.70	2.80	3.21	3.00	2.98
P61769ups	1.70	90m	2	1.85	(0.94, 2.73)	1.39	2.48	2.90	2.68	3.15
P00709ups	2.70	90m	2	3.38	(2.63, 3.98)	2.63	3.12	3.55	3.33	3.90
P02753ups	2.70	90m	2	2.88	(2.07, 3.59)	2.30	2.79	3.25	2.99	3.81
P06732ups	2.70	90m	2	3.24	(2.87, 3.60)	3.38	3.43	3.69	3.50	3.78
P12081ups	2.70	90m	2	3.27	(2.95, 3.62)	3.47	3.31	3.52	3.45	3.92
P16083ups	2.70	90m	2	3.12	(2.67, 3.58)	3.02	3.22	3.42	3.36	3.95

Table B.7: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P61626ups	2.70	90m	2	3.14	(2.75, 3.55)	2.77	3.02	3.06	3.07	4.17
P63279ups	2.70	90m	2	2.77	(2.21, 3.59)	2.53	3.15	2.77	3.12	3.70
Q15843ups	2.70	90m	2	1.84	(0.82, 2.91)	1.86	2.96	3.37	3.16	3.15
P00167ups	3.70	90m	2	4.10	(3.61, 4.66)	3.90	4.21	4.22	4.35	4.12
P01133ups	3.70	90m	2	3.97	(3.06, 5.04)	3.24	4.04	4.45	4.24	4.12
P02144ups	3.70	90m	2	4.28	(3.91, 4.68)	4.08	4.27	4.28	4.17	4.12
P04040ups	3.70	90m	2	3.91	(3.59, 4.30)	4.46	4.19	4.10	4.24	4.16
P15559ups	3.70	90m	2	3.87	(3.34, 4.37)	3.80	3.90	4.11	4.02	4.18
P62937ups	3.70	90m	2	3.84	(3.47, 4.23)	3.81	3.90	3.89	3.93	4.18
P63165ups	3.70	90m	2	3.79	(3.44, 4.11)	4.01	3.90	3.93	3.90	4.06
Q06830ups	3.70	90m	2	4.32	(4.03, 4.68)	4.35	4.30	4.34	4.33	4.52
P00915ups	4.70	90m	2	4.74	(4.50, 5.03)	4.77	4.72	4.74	4.63	4.44
P00918ups	4.70	90m	2	4.79	(4.52, 5.04)	4.88	4.75	4.81	4.76	4.52
P01031ups	4.70	90m	2	4.43	(3.89, 5.00)	4.13	4.52	4.45	4.65	4.33
P02768ups	4.70	90m	2	4.34	(4.07, 4.60)	4.92	4.47	4.48	4.47	4.35
P41159ups	4.70	90m	2	4.37	(3.94, 4.86)	4.47	4.61	4.57	4.56	4.52

Table B.7: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P62988ups	4.70	90m	2	3.63	(3.16, 4.16)	4.03	4.22	3.42	4.18	4.52
P68871ups	4.70	90m	2	4.52	(4.19, 4.85)	4.63	4.73	4.43	4.65	4.41
P69905ups	4.70	90m	2	4.65	(4.33, 4.99)	4.41	4.61	4.61	4.50	4.52

Table B.8: Detailed empirical results for UPS2 experiments, 180m gradients

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P02788ups	-0.30	180m	1	0.47	$(-0.48, 1.40)$	0.68	1.76	2.14	1.91	2.39
O00762ups	0.70	180m	1	1.36	$(0.48, 2.27)$	1.61	2.69	2.77	2.54	2.98
P02787ups	0.70	180m	1	0.72	$(-0.41, 1.74)$	3.12	3.73	3.24	3.88	2.92
P06396ups	0.70	180m	1	1.05	$(0.05, 2.15)$	3.06	3.67	3.18	3.60	2.96
O76070ups	1.70	180m	1	2.73	$(2.12, 3.36)$	2.30	2.69	2.89	2.76	4.02
P01008ups	1.70	180m	1	1.45	$(0.60, 2.45)$	1.90	2.51	2.76	2.54	3.12
P01344ups	1.70	180m	1	3.36	$(2.52, 4.19)$	2.62	3.22	3.49	3.25	4.16
P08263ups	1.70	180m	1	1.65	$(0.80, 2.47)$	1.71	2.49	2.87	2.64	3.28
P55957ups	1.70	180m	1	2.34	$(1.44, 3.21)$	2.62	3.23	3.31	3.16	3.69
P00709ups	2.70	180m	1	2.35	$(1.39, 3.31)$	2.22	3.00	3.26	2.98	3.58
P02753ups	2.70	180m	1	2.81	$(1.92, 3.89)$	2.93	3.53	3.31	3.69	3.75
P06732ups	2.70	180m	1	2.63	$(1.94, 3.24)$	3.19	3.31	3.33	3.42	3.78
P12081ups	2.70	180m	1	3.02	$(2.66, 3.41)$	3.31	3.14	3.21	3.21	4.05
P16083ups	2.70	180m	1	2.84	$(2.27, 3.39)$	2.79	3.10	3.11	3.13	3.87
P61626ups	2.70	180m	1	2.46	$(1.76, 3.14)$	2.19	2.67	2.99	2.73	3.87
P63279ups	2.70	180m	1	2.54	$(1.22, 3.80)$	2.41	3.19	3.57	3.35	3.58

Table B.8: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
Q15843ups	2.70	180m	1	2.34	(1.23, 3.54)	2.01	2.79	3.17	2.94	3.66
P00167ups	3.70	180m	1	3.78	(2.76, 4.85)	3.68	4.16	4.52	4.31	3.94
P01133ups	3.70	180m	1	4.28	(3.27, 5.50)	3.57	4.35	4.73	4.50	4.25
P02144ups	3.70	180m	1	3.88	(2.94, 4.81)	3.70	4.09	4.24	4.16	3.90
P04040ups	3.70	180m	1	3.92	(3.58, 4.29)	4.40	4.08	3.89	4.09	4.36
P15559ups	3.70	180m	1	3.53	(2.87, 4.17)	3.84	4.07	3.77	4.03	4.02
P62937ups	3.70	180m	1	3.54	(2.92, 4.16)	3.72	3.90	3.90	3.92	4.13
P63165ups	3.70	180m	1	3.34	(2.95, 3.72)	3.73	3.58	3.41	3.60	4.23
Q06830ups	3.70	180m	1	4.11	(3.59, 4.65)	3.98	4.06	4.09	4.13	4.31
P00915ups	4.70	180m	1	5.07	(4.81, 5.37)	4.90	4.90	5.00	4.84	4.41
P00918ups	4.70	180m	1	4.59	(4.19, 5.05)	4.76	4.69	4.79	4.72	4.44
P01031ups	4.70	180m	1	3.71	(3.01, 4.40)	3.90	4.21	4.02	4.24	4.64
P02768ups	4.70	180m	1	4.50	(4.21, 4.80)	5.17	4.72	4.49	4.69	4.45
P41159ups	4.70	180m	1	3.67	(3.02, 4.34)	4.18	4.42	3.64	4.42	4.39
P62988ups	4.70	180m	1	3.96	(3.39, 4.56)	4.16	4.47	4.14	4.36	4.35
P68871ups	4.70	180m	1	4.22	(3.70, 4.77)	4.42	4.50	4.44	4.51	4.54

Table B.8: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P69905ups	4.70	180m	1	4.62	(4.18, 5.05)	4.32	4.56	4.44	4.44	4.50
P02788ups	-0.30	180m	2	0.22	(−0.55, 1.03)	0.97	2.04	2.51	2.29	2.22
P01112ups	0.70	180m	2	1.51	(0.77, 2.27)	1.75	2.52	2.99	2.77	3.23
P02787ups	0.70	180m	2	1.06	(0.57, 1.56)	1.72	2.19	2.14	2.34	2.88
P06396ups	0.70	180m	2	1.14	(0.63, 1.64)	2.10	2.47	2.37	2.64	3.04
P51965ups	0.70	180m	2	0.72	(−0.16, 1.56)	0.24	1.31	1.78	1.56	2.93
P99999ups	0.70	180m	2	2.15	(0.93, 3.31)	1.76	2.83	3.00	2.78	3.11
O76070ups	1.70	180m	2	2.20	(1.64, 2.71)	1.81	2.10	2.69	2.35	3.99
P01008ups	1.70	180m	2	1.88	(1.48, 2.31)	2.22	2.39	2.39	2.54	3.46
P01344ups	1.70	180m	2	2.93	(2.43, 3.47)	2.45	2.82	2.83	2.93	4.47
P08263ups	1.70	180m	2	2.48	(1.95, 2.97)	2.74	2.96	2.69	3.06	3.85
P10599ups	1.70	180m	2	1.94	(1.04, 2.84)	1.43	2.20	2.67	2.45	3.41
P55957ups	1.70	180m	2	2.64	(2.08, 3.21)	2.52	2.81	3.21	2.99	3.99
P61769ups	1.70	180m	2	1.91	(0.44, 3.67)	2.04	3.11	3.57	3.35	3.11
P00709ups	2.70	180m	2	3.06	(2.49, 3.63)	2.85	3.22	3.14	3.26	4.02
P02753ups	2.70	180m	2	2.93	(2.42, 3.48)	2.55	2.85	3.05	3.03	4.08

Table B.8: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P06732ups	2.70	180m	2	3.20	(2.87, 3.56)	3.31	3.23	3.24	3.28	3.90
P12081ups	2.70	180m	2	3.26	(2.98, 3.56)	3.59	3.33	3.15	3.41	3.99
P16083ups	2.70	180m	2	3.09	(2.75, 3.44)	3.01	3.00	3.29	3.15	4.24
P61626ups	2.70	180m	2	3.38	(2.94, 3.86)	3.09	3.31	3.46	3.45	4.12
P63279ups	2.70	180m	2	2.96	(2.32, 3.58)	2.65	3.12	3.39	3.19	3.85
Q15843ups	2.70	180m	2	2.64	(2.07, 3.20)	2.31	2.78	2.81	2.78	3.95
P00167ups	3.70	180m	2	4.49	(4.06, 4.96)	4.14	4.37	4.41	4.51	4.21
P01133ups	3.70	180m	2	3.34	(2.66, 4.07)	2.62	3.39	3.38	3.24	4.08
P02144ups	3.70	180m	2	4.36	(4.04, 4.69)	4.11	4.23	4.36	4.28	4.18
P04040ups	3.70	180m	2	4.04	(3.79, 4.27)	4.49	4.16	4.05	4.14	4.19
P15559ups	3.70	180m	2	4.07	(3.74, 4.42)	4.01	4.04	4.03	4.12	4.22
P62937ups	3.70	180m	2	3.83	(3.46, 4.17)	3.96	3.95	4.18	3.98	4.31
P63165ups	3.70	180m	2	3.60	(3.34, 3.87)	4.14	3.87	3.64	3.87	4.29
Q06830ups	3.70	180m	2	4.32	(3.99, 4.65)	4.24	4.26	4.42	4.32	4.22
P00915ups	4.70	180m	2	4.73	(4.46, 4.99)	4.85	4.78	4.88	4.79	4.39
P00918ups	4.70	180m	2	4.76	(4.47, 5.04)	4.80	4.76	4.86	4.78	4.20

Table B.8: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P01031ups	4.70	180m	2	4.19	(3.67, 4.71)	3.82	4.11	4.10	4.29	4.47
P02768ups	4.70	180m	2	4.41	(4.20, 4.63)	4.90	4.43	4.50	4.48	4.30
P41159ups	4.70	180m	2	4.52	(4.17, 4.88)	4.39	4.61	4.32	4.56	4.21
P62988ups	4.70	180m	2	4.08	(3.68, 4.49)	3.85	4.14	4.14	4.09	4.18
P68871ups	4.70	180m	2	4.44	(4.04, 4.81)	4.53	4.70	4.48	4.62	4.15
P69905ups	4.70	180m	2	4.66	(4.33, 5.00)	4.57	4.80	4.64	4.63	4.33

Table B.9: Detailed empirical results for UPS2 experiments, 360m gradient

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P02788ups	-0.30	360m	1	0.49	$(-0.17, 1.21)$	1.17	1.78	2.17	2.03	2.69
O00762ups	0.70	360m	1	1.19	$(0.46, 1.84)$	1.03	1.64	2.32	1.89	3.34
P01112ups	0.70	360m	1	1.76	$(1.08, 2.49)$	1.71	2.32	2.66	2.45	3.43
P02787ups	0.70	360m	1	0.80	$(0.10, 1.48)$	2.00	2.48	2.43	2.74	2.86
P06396ups	0.70	360m	1	1.02	$(0.46, 1.62)$	2.00	2.40	2.54	2.57	3.01
P09211ups	0.70	360m	1	0.80	$(-0.14, 1.79)$	0.66	1.75	2.30	2.01	2.82
P99999ups	0.70	360m	1	1.21	$(0.16, 2.22)$	0.79	1.88	2.43	2.13	3.08
O76070ups	1.70	360m	1	2.47	$(1.98, 2.96)$	2.30	2.61	2.64	2.69	3.97
P01008ups	1.70	360m	1	2.15	$(1.72, 2.57)$	2.40	2.45	2.53	2.60	3.64
P01344ups	1.70	360m	1	3.12	$(2.46, 3.73)$	2.30	2.79	3.16	2.95	4.22
P08263ups	1.70	360m	1	2.60	$(2.13, 3.06)$	2.48	2.67	2.85	2.78	3.93
P10599ups	1.70	360m	1	1.95	$(0.96, 3.05)$	1.49	2.28	2.83	2.53	3.38
P55957ups	1.70	360m	1	2.66	$(2.16, 3.14)$	2.39	2.64	2.88	2.78	4.09
P61769ups	1.70	360m	1	2.50	$(1.53, 3.37)$	2.01	2.62	3.03	2.87	3.72
P00709ups	2.70	360m	1	2.68	$(1.97, 3.39)$	2.53	2.92	3.33	3.09	4.00
P02753ups	2.70	360m	1	3.17	$(2.61, 3.74)$	2.50	2.82	3.23	3.07	4.05

Table B.9: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P06732ups	2.70	360m	1	3.32	(3.03, 3.61)	3.39	3.27	3.32	3.30	3.99
P12081ups	2.70	360m	1	2.94	(2.66, 3.21)	3.36	3.04	2.98	3.15	4.15
P16083ups	2.70	360m	1	3.11	(2.71, 3.49)	3.09	3.13	3.25	3.25	4.14
P61626ups	2.70	360m	1	3.09	(2.45, 3.80)	2.63	3.03	3.28	3.20	3.83
P63279ups	2.70	360m	1	2.59	(1.96, 3.23)	2.30	2.91	2.74	2.86	3.63
Q15843ups	2.70	360m	1	2.50	(1.65, 3.38)	2.27	3.06	3.30	3.02	3.46
P00167ups	3.70	360m	1	4.67	(4.26, 5.08)	4.02	4.34	4.66	4.46	4.05
P01133ups	3.70	360m	1	3.65	(2.96, 4.32)	2.99	3.78	3.55	3.56	4.05
P02144ups	3.70	360m	1	4.20	(3.83, 4.59)	4.34	4.47	4.24	4.45	4.16
P04040ups	3.70	360m	1	3.94	(3.72, 4.18)	4.39	4.07	4.00	4.03	4.20
P15559ups	3.70	360m	1	4.01	(3.72, 4.34)	3.97	4.02	3.86	4.02	4.20
P62937ups	3.70	360m	1	3.84	(3.49, 4.25)	3.88	3.93	4.05	3.95	4.20
P63165ups	3.70	360m	1	3.69	(3.41, 3.98)	4.20	3.94	3.76	3.94	4.27
Q06830ups	3.70	360m	1	4.61	(4.35, 4.88)	4.35	4.33	4.52	4.47	4.37
P00915ups	4.70	360m	1	4.81	(4.55, 5.06)	4.77	4.78	4.77	4.73	4.22
P00918ups	4.70	360m	1	4.51	(4.17, 4.83)	4.82	4.68	4.75	4.74	4.45

Table B.9: (continued)

Protein	ζ_i	Grad.	Rep.	$\hat{\zeta}_i$	90% HPD Int.	Sum Int.	Mean Int.	Med. Int.	AMI	emPAI
P01031ups	4.70	360m	1	4.21	(3.71, 4.73)	3.90	4.22	4.19	4.34	4.45
P02768ups	4.70	360m	1	4.31	(4.09, 4.53)	4.93	4.46	4.47	4.47	4.31
P41159ups	4.70	360m	1	4.48	(4.10, 4.88)	4.56	4.70	4.45	4.65	4.45
P62988ups	4.70	360m	1	3.99	(3.59, 4.39)	4.03	4.27	4.01	4.17	4.30
P68871ups	4.70	360m	1	4.45	(4.03, 4.89)	4.42	4.56	4.60	4.62	4.24
P69905ups	4.70	360m	1	4.61	(4.29, 4.95)	4.50	4.75	4.70	4.57	4.30