ABSTRACT

Computational Bayesian Methods Applied to Complex Problems in Bio and Astro Statistics

Chris Elrod, Ph.D.

Co-Chairpersons: James D. Stamey, Ph.D
Matthew Hejduk, Ph.D.

In this dissertation we apply computational Bayesian methods to three distinct problems. In the first chapter, we address the issue of unrealistic covariance matrices used to estimate collision probabilities. We model covariance matrices with a Bayesian Normal-Inverse-Wishart model, which we fit with Gibbs sampling. In the second chapter, we are interested in determining the sample sizes necessary to achieve a particular interval width and establish non-inferiority in the analysis of prevalences using two fallible tests. To this end, we use a third order asymptotic approximation. In the third chapter, we wish to synthesize evidence across multiple domains in measurements taken longitudinally across time, featuring a substantial amount of structurally missing data, and fit the model with Hamiltonian Monte Carlo in a simulation to analyze how estimates of a parameter of interest change across sample sizes.

Computational Bayesian Methods Applied to Complex Problems in Bio and Astro Statistics

by

Chris Elrod, B.S., M.S.

A Dissertation

Approved by the Department of Statistical Science

James D. Stamey, Ph.D., Chairperson

Submitted to the Graduate Faculty of
Baylor University in Partial Fulfillment of the
Requirements for the Degree
of
Doctor of Philosophy

Approved by the Dissertation Committee

James Stamey, Ph.D., Co-Chairperson

Matthew Hejduk, Ph.D., Co-Chairperson

John Seaman, Ph.D.

Dean Young, Ph.D.

David Ryden, Ph.D.

Accepted by the Graduate School
December 2019

J. Larry Lyon, Ph.D., Dean

Page bearing signatures is kept on file in the Graduate School.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

Thank you to Dr. Stamey, Dr. Hejduk, and Dr. Seaman for all their help, guidance, and contributions throughout the dissertation process, and for the opportunities to explore numerous avenues of Bayesian computation.

DEDICATION

I would like to dedicate this dissertation to my family; to the memories of my grandparents and my father who lost theirs to dementia, and to my mom and brother who gave me the support I needed. Additionally, I would like to dedicate this to B Cavello who, despite my lack of technical or mathematical background, encouraged me to change directions and pursue my interest in Bayesian statistics with a graduate degree.

CHAPTER ONE

Satellite Collision Probability Given Non-Realistic Covariance

1.1 Problem Introduction

About 22,000 objects presently exist in Earth orbit, but only about 1,000 of these objects are active satellites—the remaining objects are a combination of "dead" satellites that are no longer functional, spent rocket bodies left over from launch activities, and debris objects, most of the latter resulting either from tests of anti-satellite weapons (such as a well-known Chinese test in 2007 that produced 3,500 pieces of trackable debris) or from collisions between satellites (such as the 2009 collision between a commercial and a Russian satellite, which produced over 2,000 trackable pieces). Defunct spacecraft and debris pieces often stay in orbit for decades, if not centuries, posing navigation and collision risks for active satellites. While in the earlier days of space operations it was correctly maintained that the "big sky" of space rendered concerns of collisions between spacecraft so unlikely that they could be safely neglected, the much more crowded conditions of the current orbital environment, in addition to the presence of actual confirmed on-orbit collisions, have changed the posture substantially: it is now commonplace for satellite operators to perform conjunction assessment activities to determine when their spacecraft will be in close proximity to other on-orbit objects and to take mitigation actions should one of these conjunctions pose an unacceptably high risk of collision (Newman et al., 2014).

The US Air Force (USAF) maintains a space "catalogue," which is an enumeration of all of the tracked objects in orbit. By operating a number of worldwide radar and optical sensor sites that track space objects, they are able both to discover new space objects and to take tracking data on existing objects; these data are then fed

into an orbital dynamics model that produces for each object both a state estimate (position and velocity of the object at a given moment in time) and a statement of estimation uncertainty, which is usually rendered as a covariance matrix of the estimated state variables. Using this same dynamical model, satellite positions and velocities, and the accompanying uncertainty, can be propagated forward temporally to time points of interest. As a routine task, the USAF propagates forward the positions of the protected US government active satellites (called "primary objects") and those of all of the remaining objects in the space catalogue (called "secondary objects") to determine which pairs of primaries and secondaries will come into close proximity of one other in the near future; objects that will pass closely to each other are identified as "conjunctors," and certain orbital products are generated for them, namely the states − position and velocity vectors − and associated covariances of the two objects at their time of closest approach (TCA). These products allow the principal collision risk assessment metric, the probability of collision (P_c), to be calculated and thus the likelihood of collision to be assessed.

While the process described above is robust in design, like most engineering processes it is only as reliable as the quality of its input data; and the input datum drawing the most scrutiny is the covariance matrix. In generating the covariance matrix, the orbit determination process presumes that all systematic errors have been fully solved for, leaving only Gaussian noise; thus, the assembled covariance merely provides variances and cross-correlation terms for the solved-for parameters, implicitly assuming that their errors follow a multivariate Gaussian distribution. While this assumption is probably a tolerable one for summarizing the errors in the orbit determination fit (although there are also several ways in which assumption can fall short even here), it is more questionable in propagation. Covariance matrices are typically propagated forward in time through the pre- and post-multiplication by a state-transition matrix, which is a linearized form of the partial derivatives that

project each state variable forward in time. While this approach has the advantage of simplicity, it does introduce certain problems. First, since the orbital dynamics are actually non-linear, the linearizations used to construct the state-transition matrix will have finite lifetimes; so linear propagation past a certain point in time will produce divergence between the true uncertainty volume and that represented by the linearized dynamics. Second, there is a mismatch between the secular state error growth, which follows the actual curvilinear orbital trajectory, and the covariance error rendered in Cartesian coordinates, which is a rectilinear system; this creates a disjunction between the Cartesian position covariance ellipsoid, which at any instant is oriented as a tangent to the orbit trajectory, and the actual error volume, which follows the trajectory and is thus "banana" shaped. A rendering of the orbital parameters in a curvilinear space, such as classical orbital elements, is one way to improve the situation; but the main collision risk assessment computation, the probability of collision, requires satellite state representation in a manner that separates the position and velocity components and therefore cannot be calculated in element space. While there do exist certain remediation techniques (e.g., Sabol et al. (2010)), these require Monte Carlo approaches, which are often computationally intensive (Hall et al., 2018). So it is important to determine the level of "realism" of covariances at propagation time points in the coordinate systems that are used for certain calculations and, if possible, to incorporate the results of such evaluations into the computation of the key collision risk assessment parameters. The thrust of the present work is to perform these realism evaluations and develop methodologies that allow an informed use of any "irrealism" data to inform P_c calculations.

Our first activity is to discuss the P_c calculation itself. There are a number of different ways to proceed in performing this calculation; we thus give an explanation of the theory behind the calculation, a brief survey of different approaches that have

been used, and the development of a highly-efficient calculation for the purposes of the present study.

Next we will explain the "covariance realism" dataset that will be used as the basis for the study's algorithm development and testing. Precision data on satellite positions are compared to predicted data and their associated covariances in order to assess the adequacy of the covariances to model actual position prediction errors. Examining these data identifies two difficulties: there is substantial autocorrelation among the evaluated position adequacy data, and the actual position error data do not conform to the expected trivariate normal distribution and thus cannot be represented by simple, unaltered covariances.

The first developed algorithm focuses on characterizing and remediating the autocorrelation in the position error dataset. Four different techniques to address this problem are explored, with an eye not only to the remediation but, perhaps even more strongly, to computational efficiency; for the dataset used in the study is extremely large, and the datasets used in the operational application of the algorithm will be even larger. One particular algorithm is selected and results presented.

The second developed algorithm considers how to address the non-Gaussian behavior of the position errors. We consider three algorithms, each of which may be interpreted as representing the error distributions as mixtures of zero-mean trivariate normal distributions. These afford us with the computational ease of working with normal distributions, while providing flexibility in fitting error distributions. We assess their performance by performing five-fold cross validation and assessing the one-sample Cramér-von Mises statistic of the test data compared to the fitted distributions.

Finally, we analyze a data set of conjunction events, both to demonstrate how corrected datasets would be used operationally to calculate improved P_c values and to assess the degree to which these methods change nominal P_c estimates.

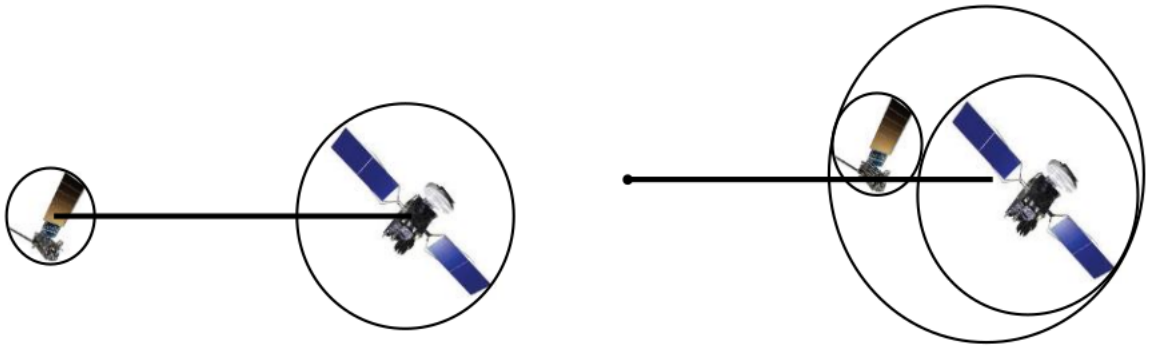

Figure 1.1. Satellite Hard-Body Radii. Primary (larger) and secondary (smaller) satellites each with circumscribing spheres, both encased by a singe circumscribing "supersphere." Figure from Hejduk and Johnson (2016).

1.2 Calculating Probability of Collision

There are different ways of calculating the P_c, and they divide most strongly into two strains: Monte Carlo approaches, which perform draws on position error statements, perturb satellite positions accordingly, and determine based on these perturbations the likelihood of the two satellites' passing within a specified close distance; and analytical approaches, which attempt a direct integration of uncertainty volume overlap with satellite combined exposed areas. Each of these approaches will be described in greater detail below.

1.2.1 P_c Calculation Methods

There are a number of analytic methods to calculate the P_c (Alfano, 2005; Chan, 2008; Foster and Estes, 1992; Patera, 2001), and all of them rely on the following simplifying assumptions (Hall et al., 2017) that allow the dimensionality to be reduced and a straightforward calculation to be performed: 1) the collision duration is short enough that satellite motion can be considered rectilinear during the encounter; 2) the collision duration is short enough that the two satellite covariance matrices can be presumed to be invariant during the encounter; and 3) the state errors for the two satellites can be considered independent, so the joint covariance can

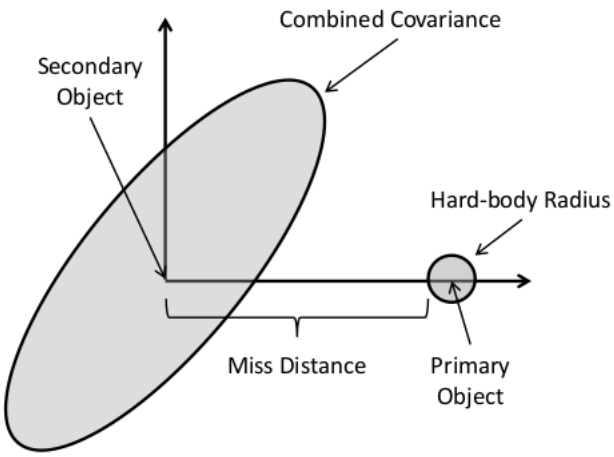

Figure 1.2. Conjunction situation in conjunction plane. Figure taken from Hejduk (2019).

be produced by a simple summation of the two covariance matrices. These three situations abide for most satellite conjunctions, so any analytical approach that requires them will in fact be relevant for most encounters. For two objects in conjunction, the uncertainty in each object's position is represented by a position covariance ellipsoid placed at the position given by the object's estimated mean state. Because it is the relative position and uncertainty that will determine whether a collision will occur, one proceeds by generating these relative quantities: the estimated positions are differenced and the relative uncertainty created by summing the position covariances. This is because if $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, then their difference $\mathbf{xy} = \mathbf{w} \sim \mathcal{N}(\boldsymbol{\mu}_1\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2)$. That is, the covariance matrix of the difference equals the sum of the individual covariance matrices.

A similar approach can be followed for space object size. One way of defining a satellite's size is to circumscribe it with a sphere and use the radius of that sphere as a size indicator; if this is done for the primary and secondary object, two circumscribing spheres are produced. A collision—or if one prefers, a close-proximity event—can be said to occur if these two spheres were to overlap at all at the time of closest approach.

A more efficient test for such a condition is to create a "supersphere" that encompasses the two touching primary and secondary object spheres, as shown in Figure 1.1.

The next step is to recognize that, given the assumptions stated previously, any collision that may occur will take place in the "conjunction plane," which is the plane normal to the relative velocity vector; the marginal probability associated with the component of the uncertainty volume normal to this plane, because the path of passage of the primary is presumed to be rectilinear in motion and thus infinitely straight, will approach unity and thus can be removed from consideration (Alfano, 2005; Chan, 2008). Furthermore, since the combined covariance, according to assumption, is static, one can place the entire conjunction situation into this plane (called the "conjunction plane") and thus pursue a two-dimensional evaluation of the conjunction situation. A plot showing this rendering (called, unsurprisingly, a "conjunction plane plot") is given in Figure 1.2; one observes the combined covariance centered at the origin (becoming an ellipse in projection), the miss vector at TCA placed upon the x-axis, and the satellites' combined size, given as the projected circle of the supersphere, placed at the right end of the miss vector. The probability of collision is thus calculated analytically as the portion of the combined uncertainty's probability density that falls within the HBR projected circle; it is these situations that will result in a miss vector that will penetrate this circle.

1.2.2 Chebyshev Integrator for Analytic P_c Calculation

We will develop here a particular optimized methodology for computing P_c, which follows the general approach of Alfano (2005) but avails itself of certain important efficiency improvements. To begin and establish consistent notation, assuming the uncertainties in the satellites position are trivariate normal, let $\mathbf{p}$ and $\mathbf{s}$ be the

positions of the primary and secondary, respectively, with distributions:

$$\mathbf{p} \sim \mathcal{N}\left(\boldsymbol{\mu}_p, \mathbf{C}_p\right)$$
$$\mathbf{s} \sim \mathcal{N}\left(\boldsymbol{\mu}_s, \mathbf{C}_s\right).$$

Then the distribution of distance between the centers of both satellites is

$$\boldsymbol{\delta} \sim \mathcal{N}\left(\boldsymbol{\mu}_p - \boldsymbol{\mu}_s, \mathbf{C}_p + \mathbf{C}_s\right).$$

We refer to the propagated covariance by the alpha numerical character C rather than a Greek letter to emphasize its treatment as provided data, rather than a parameter. Following the approach outlined in the previous section, the probability of collision integral, with the HBR sphere centered at $\mathbf{x}_0$, the probability of collision is:

$$P_c = \int_{-HBR}^{HBR} \int_{x_0-\sqrt{HBR^2-y^2}}^{x_0+\sqrt{HBR^2-y^2}} \frac{1}{2\pi\sqrt{|\mathbf{C}|}} \exp\left\{-\frac{1}{2}\begin{bmatrix} x & y \end{bmatrix} \mathbf{C}^{-1} \begin{bmatrix} x \\ y \end{bmatrix}\right\} dx\ dy.$$

Our strategy for performing this integration is to reduce it to a one dimensional integral. To do this, we first perform a change of variables from x, y to w, z, where w, z each have independent standard normal distributions. Let $\mathbf{U}$ be the inverse of the Cholesky factor of $\mathbf{C}^{-1}$, so that $\mathbf{U}$ is an upper triangular matrix such that $\mathbf{U} = \begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix}$ and $\mathbf{U}\mathbf{U}^\top = \mathbf{C}$. Now, let $\mathbf{U}^{-1}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} w \\ z \end{bmatrix}$, so that

$$x = U_{11}w + U_{12}z$$
$$y = U_{22}z$$
$$z = \frac{y}{U_{22}}$$
$$w = \frac{x - U_{12}z}{U_{11}}$$
$$= \frac{x}{U_{11}} - \frac{U_{12}y}{U_{11}U_{22}}.$$

Performing the change of variables, the quadratic form in the exponential of the bivariate normal reduces as follows:

$$\begin{aligned}
\begin{bmatrix} x & y \end{bmatrix} \mathbf{C}^{-1} \begin{bmatrix} x \\ y \end{bmatrix} &= \left(\mathbf{U} \begin{bmatrix} w \\ z \end{bmatrix} \right)^{\top} \left(\mathbf{U}\mathbf{U}^{\top} \right)^{-1} \mathbf{U} \begin{bmatrix} w \\ z \end{bmatrix} \\
&= \begin{bmatrix} w & z \end{bmatrix} \mathbf{U}^{\top}\mathbf{U}^{-\top}\mathbf{U}^{-1}\mathbf{U} \begin{bmatrix} w \\ z \end{bmatrix} \\
&= \begin{bmatrix} w & z \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix} \\
&= w^2 + z^2.
\end{aligned}$$

Additionally, the determinant of the Jacobian of the transformation is simply $|\mathbf{U}| = \sqrt{|\mathbf{C}|}$, so that our new multivariate normal to integrate over is:

$$f(w, z) = \frac{1}{2\pi} \exp\left\{ -\frac{w^2 + z^2}{2} \right\}$$

That is, w and z have independent standard normal distributions.

Now, we must derive our new limits of integration. We must integrate over the circle of radius HBR, centered at x_0. That is, it is over all values such that $\sqrt{(x - x_0)^2 + y^2} < HBR$. Putting these bounds in terms of w and z, we have that the boundaries are

$$\begin{aligned}
HBR^2 &= (x - x_0)^2 + y^2 \\
HBR^2 &= (U_{11}w + U_{12}z - x_0)^2 + (U_{22}z)^2,
\end{aligned}$$

with the integration region being all values of w and z such that the right hand side is less than or equal to HBR^2, the left hand side.

Because the extreme bounds on y are $\pm HBR$, the bounds on z will be $\pm\frac{HBR}{U_{22}}$. The bounds on w conditioned on z can be found using the quadratic formula:

$$\begin{aligned}
0 &= U_{11}^2 w^2 + U_{12}^2 z^2 + x_0^2 + U_{22}^2 z^2 + 2U_{11}U_{12}wz - 2U_{11}wx_0 - 2U_{12}zx_0 - HBR^2 \\
&= \left(U_{11}^2\right) w^2 + \left(2U_{11}U_{12}z - 2U_{11}x_0\right) w + \left(U_{12}^2 z^2 + x_0^2 + U_{22}^2 z^2 - 2U_{12}zx_0 - HBR^2\right),
\end{aligned}$$

solving for w, we find

$$\begin{aligned}
r &= \left(2U_{11}U_{12}z - 2U_{11}x_0\right)^2 - 4U_{11}^2\left(U_{12}^2 z^2 + x_0^2 + U_{22}^2 z^2 - 2U_{12}zx_0 - HBR^2\right) \\
w &= \frac{2U_{11}x_0 - 2U_{11}U_{12}z \pm \sqrt{r}}{2U_{11}^2} \\
&= \frac{x_0 - U_{12}z \pm \sqrt{\left(U_{12}z - x_0\right)^2 - \left(U_{12}^2 z^2 + x_0^2 + U_{22}^2 z^2 - 2U_{12}zx_0 - HBR^2\right)}}{U_{11}} \\
&= \frac{x_0 - U_{12}z \pm \sqrt{\left(U_{12}z - x_0\right)^2 - \left(U_{12}^2 z^2 + x_0^2 + U_{22}^2 z^2 - 2U_{12}zx_0 - HBR^2\right)}}{U_{11}} \\
&= \frac{x_0 - U_{12}z \pm \sqrt{U_{12}^2 z^2 - 2U_{12}zx_0 + x_0^2 - U_{12}^2 z^2 - x_0^2 - U_{22}^2 z^2 + 2U_{12}zx_0 + HBR^2}}{U_{11}} \\
&= \frac{x_0 - U_{12}z \pm \sqrt{HBR^2 - U_{22}^2 z^2}}{U_{11}}.
\end{aligned}$$

The standard normal cdf $\Phi(x) = \frac{1}{2}\left[2 - \texttt{erfc}\left(\frac{x}{\sqrt{2}}\right)\right]$. We use the complementary error function, $\texttt{erfc}(x) = 1 - \texttt{erf}(x)$ in place of the error function, because `erfc` is numerically accurate for values far from the mean, while `erf` is numerically accurate for values close to the mean. For example, using standard $IEEE$ double precision arithmetic, $\texttt{erf}(6) = 1.0$, while $\texttt{erfc}\,(6) = 2.1519736712498916 \times 10^{-17}$. In fact, arguments as high as 27, corresponding to distances $27\sqrt{2} \approx 38$ standard deviations from the mean, are still evaluated without rounding to zero. Because we are interested in accurately estimating what are often very small probabilities of collision where parts of the integration region may be more than $6\sqrt{2} \approx 8.5$ standard deviations from the mean, `erfc` is preferred.

Proceeding with the integration, we have

$$
\begin{aligned}
P_c &= \frac{1}{2\pi}\int_{-\frac{HBR}{U_{22}}}^{\frac{HBR}{U_{22}}}\int_{\frac{x_0-U_{12}z-\sqrt{HBR^2-U_{22}^2z^2}}{U_{11}}}^{\frac{x_0-U_{12}z+\sqrt{HBR^2-U_{22}^2z^2}}{U_{11}}} \exp\left\{-\frac{w^2+z^2}{2}\right\} dw\ dz \\
&= \frac{1}{\sqrt{2\pi}}\int_{-\frac{HBR}{U_{22}}}^{\frac{HBR}{U_{22}}} \exp\left\{-\frac{z^2}{2}\right\} \int_{\frac{x_0-U_{12}z-\sqrt{HBR^2-U_{22}^2z^2}}{U_{11}}}^{\frac{x_0-U_{12}z+\sqrt{HBR^2-U_{22}^2z^2}}{U_{11}}} \frac{1}{\sqrt{2\pi}}\exp\left\{-\frac{w^2}{2}\right\} dw\ dz \\
&= \frac{1}{\sqrt{2\pi}}\int_{-\frac{HBR}{U_{22}}}^{\frac{HBR}{U_{22}}} \exp\left\{-\frac{z^2}{2}\right\} \left[\mathbf{\Phi}\left(\frac{x_0-U_{12}z+\sqrt{HBR^2-U_{22}^2z^2}}{U_{11}}\right)\right. \\
&\qquad \left. -\mathbf{\Phi}\left(\frac{x_0-U_{12}z-\sqrt{HBR^2-U_{22}^2z^2}}{U_{11}}\right)\right] dz \\
&= \frac{1}{\sqrt{8\pi}}\int_{-\frac{HBR}{U_{22}}}^{\frac{HBR}{U_{22}}} \exp\left\{-\frac{z^2}{2}\right\} \left[\mathrm{erf}\left(\frac{x_0-U_{12}z+\sqrt{HBR^2-U_{22}^2z^2}}{\sqrt{2}U_{11}}\right)\right. \\
&\qquad \left. -\mathrm{erf}\left(\frac{x_0-U_{12}z-\sqrt{HBR^2-U_{22}^2z^2}}{\sqrt{2}U_{11}}\right)\right] dz \\
&= \frac{1}{\sqrt{8\pi}}\int_{-\frac{HBR}{U_{22}}}^{\frac{HBR}{U_{22}}} \exp\left\{-\frac{z^2}{2}\right\} \left[\mathrm{erfc}\left(\frac{x_0-U_{12}z-\sqrt{HBR^2-U_{22}^2z^2}}{\sqrt{2}U_{11}}\right)\right. \\
&\qquad \left. -\mathrm{erfc}\left(\frac{x_0-U_{12}z+\sqrt{HBR^2-U_{22}^2z^2}}{\sqrt{2}U_{11}}\right)\right] dz.
\end{aligned}
$$

We can use scaled and weight-adjusted Gaussian-Chebyshev Quadrature of the second kind to numerically integrate this function, approximating the integral as

$$
\begin{aligned}
P_c \approx \sum_{i=1}^{N} w_i \exp\left\{-\frac{\left(\frac{HBRn_i}{U_{22}}\right)^2}{2}\right\} &\left[\mathrm{erfc}\left(\frac{x_0-U_{12}\frac{HBRn_i}{U_{22}}-\sqrt{HBR^2-U_{22}^2\left(\frac{HBRn_i}{U_{22}}\right)^2}}{\sqrt{2}U_{11}}\right)\right. \\
&\left. -\mathrm{erfc}\left(\frac{x_0-U_{12}\left(\frac{HBRn_i}{U_{22}}\right)+\sqrt{HBR^2-U_{22}^2\left(\frac{HBRn_i}{U_{22}}\right)^2}}{\sqrt{2}U_{11}}\right)\right]
\end{aligned}
$$

$$= \sum_{i=1+N/2}^{N} w_i \exp\left\{-\frac{\left(\frac{HBRn_i}{U_{22}}\right)^2}{2}\right\}\left[\text{erfc}\left(\frac{x_0 - U_{12}\left(\frac{HBRn_i}{U_{22}}\right) - HBR\sqrt{1-n_i^2}}{\sqrt{2}U_{11}}\right)\right.$$
$$+\text{erfc}\left(\frac{x_0 + U_{12}\left(\frac{HBRn_i}{U_{22}}\right) - HBR\sqrt{1-n_i^2}}{\sqrt{2}U_{11}}\right)$$
$$-\text{erfc}\left(\frac{x_0 - U_{12}\left(\frac{HBRn_i}{U_{22}}\right) + HBR\sqrt{1-n_i^2}}{\sqrt{2}U_{11}}\right)$$
$$\left.-\text{erfc}\left(\frac{x_0 + U_{12}\left(\frac{HBRn_i}{U_{22}}\right) + HBR\sqrt{1-n_i^2}}{\sqrt{2}U_{11}}\right)\right]$$

where

$$w_i = \frac{\omega_i}{\sqrt{8\pi\left(1-n_i^2\right)}},$$

and n_i are the zeros of the Nth order Chebyshev polynomial, and ω_i the associated quadrature weights. The equality follows given N is even because of the symmetry of the quadrature nodes. This achieves high accuracy with few weights given reasonably smooth functions, such as this. As few as $N = 16$ allow for small relative errors on the order of 10^{-10} over a broad range of arguments. In comparing results from this calculation method to the more "standard" approach that evaluates the 2-D integral using tiled adaptive quadrature, reprocessing nearly one million actual conjunction events yielded a largest difference of $1/100^{th}$ of an order of magnitude—levels that are far from operational significance.

Our approach differs from Alfano (2005) in two chief chief ways: we use the Cholesky decomposition in place of spectral decomposition to factor the covariance matrix, and use Chebyshev quadrature following Mason and Handscomb (2002) in place of the midpoint rule for the numerical integration. Chebyshev-Gauss quadrature converges more rapdily than the midpoint rule for smooth functions, allowing us to achieve small errors with minimal computational costs. Additionally, our use of `erfc` ensures accurate P_c estimates even when they are vanishingly small.

1.3 Covariance Reaslism Test Data

To assess the ability of satellite orbit determination covariances to represent predicted state estimate errors, there are two data types that must be secured: actual state estimate errors at the propagation point(s) of interest, and associated covariances propagated to those same time points. When both of these datasets are present, it is straightforward to apply analytical procedures to determine the degree to which these covariances properly represent the actual distributions of state errors. The production of each of these datasets will be discussed in turn.

In order to generate state estimate errors for a particular satellite, one must first produce what is called a reference ephemeris for that satellite, which is an extremely precise time-history of satellite actual positions (and velocities, although the velocity portions of the states will not be examined here); it is this set of precise positions that will be compared to the positions predicted from individual vectors for the satellite in order to assess these predictions' errors. Ideally, such a reference ephemeris would be formed from external, precise data sources that have errors notably smaller than the sensor observational data that are used to form satellite state estimates generally. Such data sources do exist and form the basis for precision ephemerides for a set of satellites used to calibrate the overall orbit determination enterprise; but the number of such satellites is small (less than fifty), and we are interested in evaluating state estimates for the entire satellite catalogue, which comprises over 22,000 objects. It is therefore necessary to employ some other stratagem than using calibration reference orbits.

When satellite state estimates are updated with sensor observational data, a batch update process is employed, in which a group of past sensor position observations is collected (data falling into the update's "fit-span," which is a span of some number of days back in time from the current time) and used in an ensemble adjustment to the previous estimate's trajectory, modifying this trajectory in a manner that minimizes the variance of the residuals of the fit-span observational data to the modified orbit. Once this process is complete, the modified orbit can be propagated back in time to produce an ephemeris for the fit-span period of time, and this ephemeris will be extremely accurate because the fit-span is the

period of time for which the orbit adjustment fitting took place. If this is done for each subsequent orbit update, then a series of short ephemerides pieces can be produced, and they can be stitched together to create a continuous ephemeris for the object. One must exercise care to ensure that the different pieces of ephemeris share little to no observational data so that there is no correlation between ephemeris segments. Each segment's error can be reasonably estimated by the formation covariance that was produced from the orbit update that generated it (Hejduk, 2008).

With a reference ephemeris available, it is straightforward to calculate prediction errors for a set of vectors that was produced for a given satellite. If one is interested in, say, the accuracy of a particular vector propagated two days into the future, one finds the ephemeris point in the reference ephemeris closest to the vector epoch time plus two days and propagates the vector to that point; the propagated state can then be differenced with the ephemeris state at that point, and position errors in the three Cartesian components can be calculated. Of course, this analysis can be conducted only in the past, as the constructed reference orbit can exist for only past data; but past vectors can be propagated forward in the same manner in which they would be predicted forward from the present time into the future (one must be sure to use the space weather and other dynamical model indices that would have actually been relevant at that past time). While the position comparison is conducted in an inertial reference frame to ensure compatibility between reference ephemeris and propagated vector, it is common to transform the position differences into a relative frame, typically centered on the employed ephemeris point. This relative framework defines the **u** direction as "radial" or containing the satellite position and the center of the earth, the **v** direction as "in-track" or perpendicular to the radial component, but lying in the orbit plane (and for a circular orbit also parallel to the velocity vector), and the **w** or "cross-track" component in the direction perpendicular to the orbit plane. An entire set of vectors can be evaluated this way, at a number of different propagation times, producing a set of ordered triples of **u**, **v**, and **w** position errors (Hejduk et al., 2013).

Prediction residuals are useful in assessing the actual values of the errors encountered, but what is desired here is to determine how well or poorly the covariances associated with

the vectors represent the actual distributions of errors. For any given vector and propagation state, there is only one set of uvw residuals and one propagated covariance; and it is not possible to draw any durable conclusions from single-sample situations. However, what can be done is essentially to normalize the residuals by calculating the Mahalanobis distance: if $\mathbf{e}$ is the vector of errors so that $\mathbf{e} = \begin{bmatrix} u_n & v_n & w_n \end{bmatrix}^\top$ and $\mathbf{C}$ is the covariance, m^2 is given by $\mathbf{e}\mathbf{C}^{-1}\mathbf{e}^\top$. For a diagonal covariance, the effect is to create a sum of the squares of normalized z-variables; for situations with correlation, these effects are properly represented in the multiplication by the correlation terms. A set of m^2 values, as the squares of normal z-variables, should produce a 3-degree of freedom chi-square distribution (Zaidi and Hejduk, 2016).

The USAF routinely executes software that produces position residuals and associated propagated covariances, using the methodology described above, at a set of standard propagation states for all satellites in the satellite catalogue. These data were obtained for a very large subset of the satellite catalogue for all of 2017 and processed in order to produce m^2 histories for propagation states of interest. It is this dataset that serves as the basis for the present covariance realism examination.

To facilitate the assessment of these datasets, we standardize them using the propagated covariance matrices in a manner similar to that described earlier, but with some modifications. For a particular satellite, let N be the number of ordered triples u, b, w corresponding to this satellite (we drop the satellite subscript s to reduce notational clutter), and $\mathbf{e}_n$ and $\mathbf{C}_n$ refer to the nth set of errors and covariance matrix. Now we define $\mathbf{L}_n$ as the lower Cholesky factor of the propagated covariance matrix so that $\mathbf{C}_n = \mathbf{L}_n\mathbf{L}_n^\top$, and $\mathbf{z}_n = \mathbf{L}_n^{-1}\mathbf{e}_n$. Then we have that

$$\begin{aligned}
\mathbf{z}_n^\top\mathbf{z}_n &= \left(\mathbf{L}_n^{-1}\mathbf{e}_n\right)^\top \left(\mathbf{L}_n^{-1}\mathbf{e}_n\right) \\
&= \mathbf{e}_n^\top\mathbf{L}^{-\top}\mathbf{L}^{-1}\mathbf{e}_n \\
&= \mathbf{e}_n^\top \left(\mathbf{L}\mathbf{L}^\top\right)^{-1} \mathbf{e}_n \\
&= \mathbf{e}_n^\top\mathbf{C}^{-1}\mathbf{e}_n \\
&= m^2.
\end{aligned}$$

If $\mathbf{e}_n \sim \mathcal{N}(\mathbf{0}_3, \mathbf{C}_n)$, then $\mathbf{z}_n \sim \mathcal{N}(\mathbf{0}_3, \mathbf{I}_3)$. The statistical likelihoods of the observations are invariant to this one to one transformation. By applying the transformation, we now have N identically distributed observations, simplifying the analysis.

The Cholesky decomposition was chosen as the orthogonalizing transformation for this preprocessing step due to its simplicity and computational efficiency.

1.4 Autocorrelation in Successive Observations

The previous section discussed the methodology for determining satellite state estimation errors, including mention of an approach to eliminate correlation between subsequent ephemeris pieces that are fused into a single satellite ephemeris. While this vigilance is appropriate, one must also consider the possibility of correlation introduced by the state vectors themselves that are to be analyzed. Typically, satellite state vectors are updated three times per day (if new tracking data are received during the preceding eight-hour period), but the number of updates can be much larger if the object merits special interest. Since the typical fit-span for near-earth satellites (orbital periods less than 225 minutes) is several days, it is clear that vectors updated every eight hours will share most of the observational data used in the update; and this will introduce substantial correlation between successive updates. It is therefore necessary to test these datasets rigorously for autocorrelation and, if significant such correlation is discovered, to account for this in our modeling efforts.

1.4.1 Identifying Autocorrelation

In testing for autocorrelation, we will assume normality in defining a distribution for sample autocorrelations under the null hypothesis of no autocorrealation. While this normality assumption is not valid, it allows us to define a reference distribution for the sampling distribution of autocorrelation. Given a vector $\mathbf{z}$ of N independent and identically distributed normal observations, let $\mathbf{x}$ and $\mathbf{y}$ be two vectors of length $N-1$, such that $x_n = z_n, n = 1, \ldots, N-1$ and $y_n = z_{n+1}, n = 1, \ldots, N-1$. Then the Pearson correlation coefficient between $\mathbf{x}$ and $\mathbf{y}$ equals the lag-1 autocorrelation between elements of $\mathbf{z}$.

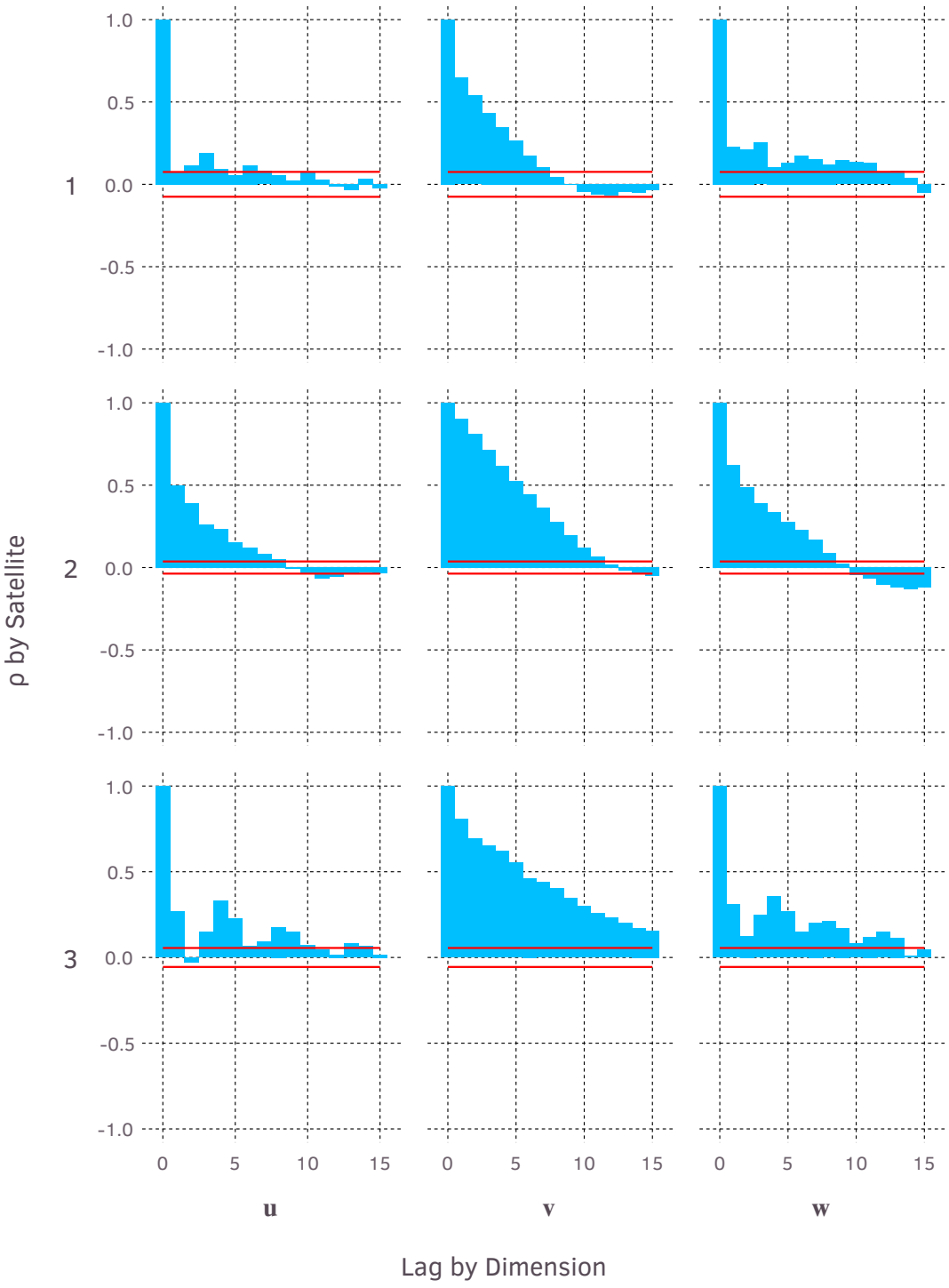

Figure 1.3. Autocorrelations for the first 15 lags for three sample satellites.

Given that the true autocorrelation is zero, this correlation coefficient has a sampling distribution according to the density (Hotelling, 1953):

$$f(r) = \frac{\left(1-r^2\right)^{\frac{N-5}{2}}}{\mathrm{B}\left(\frac{1}{2}, \frac{N-3}{2}\right)}$$
$$F(r) = \frac{r\,{}_2F_1\left(\frac{1}{2}, \frac{5}{2} - \frac{N}{2}; \frac{3}{2} | r^2\right)}{\mathrm{B}\left(\frac{1}{2}, \frac{N-3}{2}\right)}$$

where ${}_2F_1$ is the hypergeometric function:

$${}_2F_1\left(a, b; c | z\right) = \sum_{n=0}^{\infty} \frac{\left(\prod_{i=0}^{n-1}\left(a+i\right)\right)\left(\prod_{i=0}^{n-1}\left(b+i\right)\right)}{\left(\prod_{i=0}^{n-1}\left(c+i\right)\right)} \frac{z^n}{n!}$$

and B is the Beta function:

$$\mathrm{B}(a,b) = \frac{\Gamma(a)\,\Gamma(b)}{\Gamma(a+b)},$$

and a product with zero terms is taken to equal zero, that is $\prod_i = 0^0\left(x+i\right) = 1$. We used the software library Nemo (Fieker et al., 2017) to evaluate the hypergeometric function.

We use the Newton-Raphson method for calculating the 2.5% and 97.5% quantiles of the sampling distribution. We plot these alongside the first 15 autocorrelations of three example satellites (Fig. 1.3) with sample sizes of 684, 2827, and 1260, respectively. The first few autocorrelations are strongly positive, as high as 0.9 for $\mathbf{v}$ in the second satellite. Beyond a lag of 10, some of the autocorrelations become negative.

1.4.2 Candidate Autocorrelation Modeling Approaches

Our goal here is to present a family of models that will have the following properties:

- To model autocorrelation across time,
- To model the marginal variance of $\mathbf{z}$, and
- To allow for the autocorrelation and variance to be fit in separate steps.

We will discuss models addressing autocorrelation and marginal variance independently, and then combine and derive the implied model for the joint distribution of the observed data, $\mathbf{Z}$. To take advantage of the Chebyshev integrator for calculating P_c values and computational convenience, we will confine ourselves to the realm of normal mixture models.

We will model autocorrelation across time with correlation matrices, *i.e.* positive definite matrices with unit diagonals. Due to both the large number of satellites we must fit and the potentially large size of these matrices, computational efficiency is a concern. For this reason, we restrict ourselves to approaches with algorithmic complexity better than $O(N^2)$. That is, we restrict ourselves to approaches where the asymptotic runtime of the algorithm increases at a rate lower than N^2 as a function of sample size: we wish for each doubling in sample size to less than quadruple the time it takes to evaluate the correlation matrix. Note that a dense matrix has $O(N^2)$ elements, and that common matrix operations such as multiplication and factorization as implemented in optimized BLAS libraries have time complexities of $O(N^3)$ (Al-Mouhamed et al., 2016). For this reason we cannot use dense matrices if we are to achieve our desired performance.

Instead, we will use structured matrices defined by a limited number of parameters and the observed time points. In particular, we will consider the following approaches:

- Order one autoregressive (AR(1)) matrix: the AR(1) approach models the correlation between observations at time t_2 and t_1 as $\rho^{|t_2-t_1|}, 0 \leq \rho \leq 1$. This model can be interpreted as one of errors randomly drifting over time, with a tendency to regress towards zero. Given two observations, the closer they are in time, the less time the errors have had to drift apart, and the closer they will be on average. If we have three observations, the distribution of the third is independent of the first when we condition on the second.

- Hiearchical off-diagonal low-rank (HODLR) matrices (Geoga et al., 2018): these allow modeling the correlation structure as a Gaussian process. The matrix is partitioned hierarchically into blocks. Diagonal blocks are dense, while off-diagonal

blocks are limited to a rank of k. Blocks further from the diagonal are larger. A kernel function indicating correlation between observations at times t_i and t_j is used to define the matrix.

- Banded correlation matrix: we can use a correlation but with a limited number B of non-zero offdiagonal bands; all other values are zero. This approach can only model correlations at up to B lags, but complexity and processing burden are reduced to O(N) by limiting the populated portion of the correlation matrix.

- Banded precision matrix: we can define the inverse of the correlation matrix to have some number B of non-zero off-diagonal bands; all other values are zero. The advantage of the banded precision matrix is that its inverse will generally be dense, allowing B bands to model correlations at any number of lags.

For each satellite, we fit the correlation matrix to each of $\mathbf{y}_i, i = 1, 2, 3$ where $Y_{n,i} = \frac{Z_{n,i}}{\text{st.dev.}(\mathbf{z}_i)}$, where we are using $\mathbf{a}_i$ to refer to the ith column of a matrix $\mathbf{A}$. We normalized by the standard deviation because our intention in this step is to fit the correlation between successive observations, while leaving the marginal variance to the following step as a mixture of normals; for fitting the marginal variances, we restore the original scale. Scaling is necessary because the fitted correlation matrix is dependent on the scale of the observations. Recall the P-variate multivariate normal log density function:

$$\ell_P(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = -\frac{P}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}\boldsymbol{\delta}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}$$
$$\boldsymbol{\delta} = \mathbf{x} - \boldsymbol{\mu}.$$

If $\boldsymbol{\Sigma}$ is restricted to being a correlation matrix, the correlation structure of $\hat{\boldsymbol{\Sigma}}$, the MLE estimate, is dependent on the scale of $\boldsymbol{\delta}$.

For example, let $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Xi})$, where $\boldsymbol{\Xi}$ is a correlation matrix. Additionally, let $\mathbf{L}$ be the lower triangular Cholesky factor of $\boldsymbol{\Xi}$, $\boldsymbol{\Sigma}$ also be a correlation matrix and $\boldsymbol{\Sigma} = \mathbf{L}\boldsymbol{\Lambda}\mathbf{L}^\top$, and we define $\boldsymbol{\delta} = \xi\boldsymbol{\xi}$, where ξ is an arbitrary scale. Additionally, let $\mathbf{Q}\boldsymbol{\Gamma}\mathbf{Q}^\top = \boldsymbol{\Lambda}$, where $\boldsymbol{\Gamma}$ is a diagonal matrix of eigenvalues, and $\mathbf{Q}$ is an orthonormal matrix of eigenvectors so that

$\mathbf{Q}\mathbf{Q}^\top = \mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$.

$$\begin{aligned}
\boldsymbol{\alpha} &= \mathbf{Q}\mathbf{L}^{-1}\boldsymbol{\xi} \\
\boldsymbol{\alpha} &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\
\ell_P(\boldsymbol{\delta}, \mathbf{0}, \boldsymbol{\Sigma}) &= -\frac{P}{2}\log(2\pi) - \frac{1}{2}\log\left|\mathbf{L}\boldsymbol{\Lambda}\mathbf{L}^\top\right| - \frac{1}{2}\boldsymbol{\delta}^\top\left(\mathbf{L}\boldsymbol{\Lambda}\mathbf{L}^\top\right)^{-1}\boldsymbol{\delta} \\
&= -\frac{P}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Xi}| - \frac{1}{2}\log|\boldsymbol{\Lambda}| - \frac{\xi^2}{2}(\mathbf{L}\boldsymbol{\alpha})^\top\boldsymbol{\Lambda}^{-1}\mathbf{L}\boldsymbol{\alpha} \\
&= -\frac{P}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Xi}| - \frac{1}{2}\sum_{p=1}^{P}\log(\boldsymbol{\Gamma}_{p,p}) - \frac{\xi^2}{2}\boldsymbol{\alpha}^\top\boldsymbol{\Gamma}^{-1}\boldsymbol{\alpha}
\end{aligned}$$

Noting that $E\left[\boldsymbol{\alpha}^\top\boldsymbol{\Gamma}^{-1}\boldsymbol{\alpha}\right] = \operatorname{tr}\left(\boldsymbol{\Gamma}^{-1}\right) = \sum_p^P \frac{1}{\boldsymbol{\Gamma}_{p,p}}$ (Bates, 2011),

$$E\boldsymbol{\rho}\ell_P(f_P(\boldsymbol{\delta}, \mathbf{0}, \boldsymbol{\Sigma})) = -\frac{P}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Xi}| - \frac{1}{2}\sum_{p=1}^{P}\log(\boldsymbol{\Gamma}_{p,p}) - \frac{\xi^2}{2}\sum_{p=1}^{P}\frac{1}{\boldsymbol{\Gamma}_{p,p}}.$$

If we let $\xi \to \infty$, this function will be maximized by letting $\sum_{p=1}^{P}\frac{1}{\boldsymbol{\Gamma}_{p,p}} \to 0$. That is, $\boldsymbol{\Gamma}_{p,p} \to \infty$ for $p = 1, \ldots, P$. If we let $\xi \to 0$, then the density is maximized as $\sum_{p=1}^{P}\log(\boldsymbol{\Gamma}_{p,p}) \to -\infty$. That is, when at least one of $\boldsymbol{\Gamma}_{p,p} \to 0$.

Therefore changing the scale of our inputs, ξ, alters $\boldsymbol{\Lambda}$, and therefore $\boldsymbol{\Sigma}$. Because we are fitting the marginal variance in a subsequent data processing step, we want the results of this step to be independent of variance; that we restrict $\boldsymbol{\Sigma}$ to be a correlation matrix, implying marginal variances of 1, means that scaling the data to a marginal variance of 1 is the natural choice.

The MLEs are found through numerical optimization techniques. Details differ for each method we consider.

1.4.3 Autoregressive Matrix

The autoregressive correlation matrix is defined by a single parameter, ρ. Recall that $\mathbf{t}$ is the set of N sorted observation times. Then, dropping the subscript is, we have under

the AR(1) model:

$$\mathbf{\Lambda}(\rho) = \begin{bmatrix} 1 & \rho^{t_2-t_1} & \rho^{t_3-t_1} & \rho^{t_4-t_1} & \dots & \rho^{t_N-t_1} \\ \rho^{t_2-t_1} & 1 & \rho^{t_3-t_2} & \rho^{t_4-t_2} & \dots & \rho^{t_N-t_2} \\ \rho^{t_3-t_1} & \rho^{t_3-t_2} & 1 & \rho^{t_4-t_3} & \dots & \rho^{t_N-t_3} \\ \rho^{t_4-t_1} & \rho^{t_4-t_2} & \rho^{t_4-t_3} & 1 & \dots & \rho^{t_N-t_4} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho^{t_N-t_1} & \rho^{t_N-t_2} & \rho^{t_N-t_3} & \rho^{t_N-t_4} & \dots & 1 \end{bmatrix}$$

which, if factored into $\mathbf{\Lambda} = \mathbf{L}\mathbf{L}^\top$, will produce $\mathbf{L}^{-1}$ of the form:

$$\mathbf{L}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ \frac{-\rho^{t_2-t_1}}{\sqrt{1-\rho^{2(t_2-t_1)}}} & \frac{1}{\sqrt{1-\rho^{2(t_2-t_1)}}} & 0 & 0 & \dots & 0 \\ 0 & \frac{-\rho^{t_3-t_2}}{\sqrt{1-\rho^{2(t_3-t_2)}}} & \frac{1}{\sqrt{1-\rho^{2(t_3-t_2)}}} & 0 & \dots & 0 \\ 0 & 0 & \frac{-\rho^{t_4-t_3}}{\sqrt{1-\rho^{2(t_4-t_3)}}} & \frac{1}{\sqrt{1-\rho^{2(t_4-t_3)}}} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \frac{1}{\sqrt{1-\rho^{2(t_N-t_{N-1})}}} \end{bmatrix}.$$

To solve for our estimator, we find the maximum of the log density of the approximating model $\mathbf{y}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Lambda}_i), i = 1, 2, 3$. Again dropping the sub-scripted is, because they may be inferred from context, we have

$$\begin{aligned} \mathcal{N}(\mathbf{y}_i|\mathbf{0}, \mathbf{\Lambda}) &= -\frac{N}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{\Lambda}| - \frac{1}{2}\mathbf{y}^\top\mathbf{\Lambda}\mathbf{y} \\ &= -\frac{N}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{L}\mathbf{L}^\top| - \frac{1}{2}\mathbf{i}^\top\left(\mathbf{L}\mathbf{L}^\top\right)^{-1}\mathbf{y} \\ &= -\frac{N}{2}\log(2\pi) - \log|\mathbf{L}| - \frac{1}{2}\mathbf{y}^\top\mathbf{L}^{-\top}\mathbf{L}^{-1}\mathbf{y} \\ &= -\frac{N}{2}\log(2\pi) + \log|\mathbf{L}^{-1}| - \frac{1}{2}\left(\mathbf{L}^{-1}\mathbf{y}\right)^\top\mathbf{L}^{-1}\mathbf{y}. \end{aligned}$$

Therefore, the only computations with respect to $\mathbf{\Lambda}$ we must calculate are the determinant $|\mathbf{L}^{-1}|$ and the quadratic form, $\left(\mathbf{L}^{-1}\mathbf{i}\right)^\top\mathbf{L}^{-1}\mathbf{i}$. Because the determinant of a triangular matrix

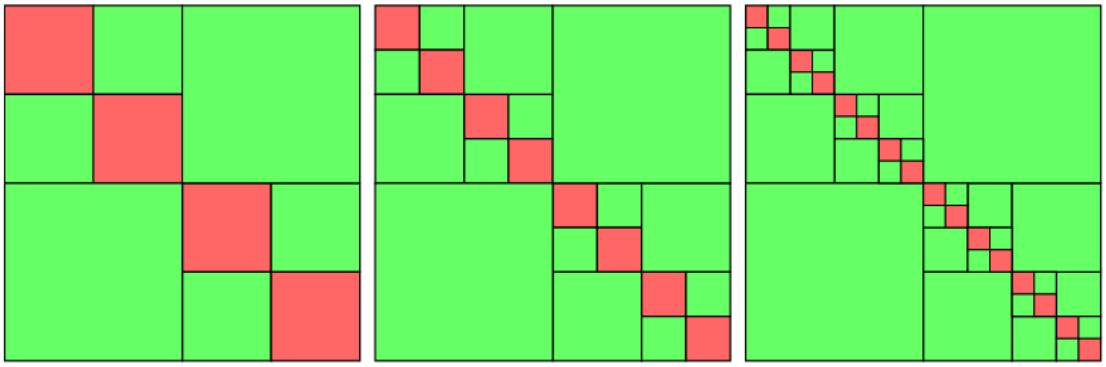

Figure 1.4. HODLR matrices with 2, 3, and 4 dyadic splits. Figure originally published in Kressner (2015).

is the product of the diagonals, these may be calculated as

$$|\mathbf{L}^{-1}| = \prod_{n=2}^{N} \frac{1}{\sqrt{1-\rho^{2(t_n - t_{n-1})}}}$$

$$\left(\mathbf{L}^{-1}\mathbf{y}_i\right)^\top \mathbf{L}^{-1}\mathbf{y}_i = Y_{1,i}^2 + \sum_{n=2}^{N} \frac{\left(Y_{n,i} - Y_{n-1,i}\rho^{t_n - t_{n-1}}\right)^2}{1-\rho^{2(t_n - t_{n-1})}}.$$

Therefore, evaluating the log density requires only $O(N)$ operations. We use the Newton-Raphson method to find the MLE, using analytical first derivatives and applying forward-mode automatic differentiation to find second derivatives (Revels et al., 2016).

1.4.4 HODLR Model

This approach allows us to define a Gaussian process by providing an efficient representation of matrices with low effective rank in off-diagonal blocks. See Fig. 1.4 for a visualization of the hierarchical partionioning of the matrix. We used 8 dyadic splits and allow a maximum off-diagonal rank of 24.

We use the software provided by Geoga et al. (2018) to both evaluate the HODLR matrix and approximate the off diagonal matrices using the Nyström method (Nyström, 1930). To find the MLE, we used Johnson (2019) through the interface provided by Geoga et al. (2018)'s library, which has runtime performance on the order of $O\left(N \log^2(N)\right)$.

We used the following five parameter kernel to define the degree of autocorrelation as a function of the time between two observations, t_m and t_n.

$$\boldsymbol{\lambda} = \begin{bmatrix} \alpha & \lambda & \sigma & \pi & \phi \end{bmatrix}^\top$$

$$K(\boldsymbol{\lambda}, t_m, t_n) = \pi \left(1 + (t_m - t_n)^2\right)^{-\alpha} + (1 - \pi) \left(\phi e^{-0.5\lambda(t_m - t_n)^2} + (1 - \phi)\, e^{-\sigma |t_m - t_n|}\right),$$

where

$$\pi, \phi \in (0, 1)$$

$$\alpha, \lambda, \sigma > 0.$$

This kernel is a composite kernel, with rational quadratic, squared exponential, and Ornstein-Uhleneck (Uhlenbeck and Ornstein, 1930) components. The parameters π and ϕ determine the relative contribution of each of these three kernels to correlation between times. The advantage of the composite kernel is that each of the constituent kernels are a special case: if $\pi = 1$, the kernel is rational, if $\pi = 0$ and $\phi = 1$ the kernel is a squared exponential, and if $\pi = 0, \phi = 0$ then it is an Ornstein-Uhleneck kernel. This gives the model greater flexibility in specifying the degree of correlation as a function of time.

1.4.5 Banded Correlation Matrix

Here we consider simply a correlation matrix with only a limited number, B, of non-zero off-diagonal bands below and above the diagonal, so that the time complexity of operations increases simply as a function of N, i.e. runtime will be $O(N)$. All correlations more than B lags apart are assumed to be zero.

We require that the correlation matrix be positive definite. To guarantee this, we define the correlation matrix implicitly through defining elements of its Cholesky decomposition. If a positive definite matrix has precisely B non-zero bands both below and above the diagonal, then its Cholesky decomposition will have B off-diagonal non-zero bands.

We follow a modified version of the approach used by Team (2018) for transforming unconstrained parameters into a correlation matrix. The principal modifications are

that we use kernel functions to produce our set of unconstrained parameters, and set each corresponding to a lag greater than B to 0.

We summarize our modified approach as follows: Letting $\mathbf{L}$ be the lower triangular Cholesky factor, the norm of each of its rows must be 1. This is because the elements of $\mathbf{L}\mathbf{L}^\top$ equal the dot products of the corresponding rows, and the diagonal elements of a correlation matrix must be 1. We can therefore take a stick breaking approach to allocating this unit norm among the B bands. That is, given a series of B values $x_b, b = 1, \ldots, B$, we translate these into values $y_b = \frac{2}{1+\exp(-x_b)} - 1$ so that $y_b \in (-1, 1)$. For $b = 1, \ldots, B$, $|y_b|$ equals the fraction of the remain "stick" we break off and allocate to the b^{th} diagonal.

$$z_b = y_b \sqrt{1 - \sum_{c=1}^{b-1} z_c^2}, \text{ for all } b = 1, \ldots, B$$

and the (B+1)st element, z_{B+1}, equals the remainder:

$$z_{B+1} = \sqrt{1 - \sum_{b=1}^{B} z_b^2}.$$

Then the dot product of of each row with itself equals

$$\begin{aligned} \sum_{b=1}^{B+1} z_b^2 &= \left(\sum_{b=1}^{B} z_b^2 \right) + 1 - \sum_{b=1}^{B} z_b^2 \\ &= 1, \end{aligned}$$

so that the diagonal elements of $\mathbf{L}\mathbf{L}^\top$ are all 1 and each diagonal element of $\mathbf{L}$, z_{B+1}, are positive. Therefore $\mathbf{L}\mathbf{L}^\top$ is a valid correlation matrix. The stick-breaking interpretation is that each y_b^2 equals the proportion of the remaining stick broken off and allocated to the bth band. The radical tracks how much has already been allocated to previous bands.

We use quadratic polynomials to define the initial values x_b, where x_b corresponds to a row of $\mathbf{L}$. That is,

$$x_b = a_b \left(t_{B+1} - t_b\right)^2 + b_b \left(t_{B+1} - t_b\right) + c_b.$$

Given B bands, this model has $3B$ parameters.

To find the MLE, we used the quasi-Newton Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm with a third order back-tracking line search (Mogensen and Riseth, 2018), using automatic differentation (Revels et al., 2016) for the derivatives.

1.4.6 Banded Precision Matrix

The banded correlation matrix assumes that all correlations beyond B lags equal zero, which we do not assume in general. If we instead define B non-zero bands on the precision matrix, $\mathbf{P}$, its inverse − the correlation matrix $\mathbf{\Lambda}-$ will be dense. This alligns closer with our expectation that correlations will decay slowly to zero as the number of lags increase.

Here, we must again ensure that the matrix be positive definite and that all the diagonal elements of the inverse equal 1. If we use the Cholesky decomposition to factor the correlation matrix $\mathbf{\Lambda}$ into lower triangular matrices $\mathbf{L}$ such that $\mathbf{L}\mathbf{L}^\top = \mathbf{\Lambda}$, $\mathbf{L}^{-1}$ will have B subdiagonal non-zero bands. We will have B functions defining these bands, $g_b\left(\boldsymbol{\lambda}, t_n, t_{n-b}\right), b = 1, \ldots, B = \lambda_b e^{-\lambda_{B+b}(t_n - t_{n-b})^2}$, $\lambda_b \in (-\infty, \infty)$, $\lambda_b > 0$, $b = 1+B, \ldots, 2B$. Now we must find the value of the diagonal elements that will result in the inverse being a correlation matrix.

$$\mathbf{L}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \ldots & 0 \\ g_1\left(\boldsymbol{\lambda}, t_2, t_1\right) & \left(\mathbf{L}^{-1}\right)_{2,2} & 0 & 0 & 0 & \ldots & 0 \\ g_2\left(\boldsymbol{\lambda}, t_3, t_1\right) & g_1\left(\boldsymbol{\lambda}, t_3, t_2\right) & \left(\mathbf{L}^{-1}\right)_{3,3} & 0 & 0 & \ldots & 0 \\ g_3\left(\boldsymbol{\lambda}, t_4, t_1\right) & g_2\left(\boldsymbol{\lambda}, t_4, t_2\right) & g_1\left(\boldsymbol{\lambda}, t_4, t_3\right) & \left(\mathbf{L}^{-1}\right)_{4,4} & 0 & \ldots & 0 \\ g_4\left(\boldsymbol{\lambda}, t_5, t_1\right) & g_3\left(\boldsymbol{\lambda}, t_5, t_2\right) & g_2\left(\boldsymbol{\lambda}, t_5, 5_3\right) & g_1\left(\boldsymbol{\lambda}, t_5, t_4\right) & \left(\mathbf{L}^{-1}\right)_{5,5} & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \ldots & \left(\mathbf{L}^{-1}\right)_{T,T} \end{bmatrix}$$

That is, we must choose values of $\left(\mathbf{L}^{-1}\right)_{t,t}$ such that the diagonal elements of $\mathbf{\Lambda}$ equal 1. In refering to contiguous submatrices, we introduce the following notation: $\mathbf{A}_{i:j,k:l}, i \leq j, k \leq l$ refers to the block of $\mathbf{A}$ with corners $A_{i,k}, A_{j,k}, A_{i,l}, A_{j,l}$. For convenience, the set $i, \ldots, 0$ is treated as empty for $i > 0$, and $i \ldots, j = 1 \ldots, j$ for $i \leq 1$. Recall that given

a multivariate normal distribution with covariance matrix $\mathbf{\Sigma}$, the marginal distribution of some set of elements $j, \ldots, k$ is a multivariate normal with covariance matrix $\mathbf{\Sigma}_{j:k,j:k}$.

Our approach to solving this problem in $O(N)$ time hinges on the observation that if we have a block $\mathbf{\Lambda}_{1:i,1:i}$, we can define it in terms of the precision matrix $\mathbf{P}$ as follows:

$$\begin{aligned}
\mathbf{\Lambda}_{1:i,1:i} &= \begin{bmatrix} \mathbf{\Lambda}_{1:i-B-1,1:i-B-1} & \mathbf{\Lambda}_{i-B:i,1:i-B-1}^\top \\ \mathbf{\Lambda}_{i-B:i,1:i-B-1} & \mathbf{\Lambda}_{i-B:i,i-B:i} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{P}_{1:i-B-1,1:i-B-1} & \mathbf{P}_{i-B:i,1:i-B-1}^\top \\ \mathbf{P}_{i-B:i,1:i-B-1} & \mathbf{P}_{i-B:i,i-B:i} \end{bmatrix}^{-1} \\
&= \begin{bmatrix} \mathbf{X} & \mathbf{Y}^\top \\ \mathbf{Y} & \left(\mathbf{P}_{i-B:i,i-B:i} - \mathbf{P}_{i-B:i,1:i-B-1}\mathbf{P}_{1:i-B-1,1:i-B-1}^{-1}\mathbf{P}_{i-B:i,1:i-B-1}^\top\right)^{-1} \end{bmatrix}.
\end{aligned}$$

The last row of $\mathbf{P}_{i-B:i,1:i-B-1}$ is a zero (row) vector of length $i-B-1$:

$$\begin{aligned}
\mathbf{P}_{i-B:i,1:i-B-1} &= \begin{bmatrix} 0 & \ldots & P_{i-B,i-2B} & P_{i-B,i-2B+1} & \ldots & P_{i-B,i-B-2} & P_{i-B,i-B-1} \\ 0 & \ldots & 0 & P_{i-B+1,i-2B+1} & \ldots & P_{i-B+1,i-B-2} & P_{i-B+1,i-B-1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \ldots & 0 & 0 & \ldots & 0 & P_{i-1,i-B-1} \\ 0 & \ldots & 0 & 0 & \ldots & 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{P}_{i-B:i-1,1:i-B-1} \\ \mathbf{0}_{i-B-1} \end{bmatrix}
\end{aligned}$$

For the matrix quadratic form, we have

$$\begin{aligned}
&\mathbf{P}_{i-B:i,1:i-B-1}\mathbf{P}_{1:i-B-1,1:i-B-1}^{-1}\mathbf{P}_{i-B:i,1:i-B-1}^\top \\
&= \begin{bmatrix} \mathbf{P}_{i-B:i-1,1:i-B-1} \\ \mathbf{0}_{i-B-1}^\top \end{bmatrix} \mathbf{P}_{1:i-B-1,1:i-B-1}^{-1} \begin{bmatrix} \mathbf{P}_{i-B:i-1,1:i-B-1} \\ \mathbf{0}_{i-B-1}^\top \end{bmatrix}^\top \\
&= \begin{bmatrix} \mathbf{P}_{i-B:i-1,1:i-B-1}\mathbf{P}_{1:i-B-1,1:i-B-1}^{-1}\mathbf{P}_{i-B:i-1,1:i-B-1}^\top & \mathbf{0}_{i-B-1} \\ \mathbf{0}_{i-B-1}^\top & 0 \end{bmatrix}.
\end{aligned}$$

Now we have that

$$
\begin{aligned}
\left(\mathbf{\Lambda}_{i-B:i,i-B:i}\right)^{-1} &= \mathbf{P}_{i-B:i,i-B:i} - \mathbf{P}_{i-B:i,1:i-B-1}\mathbf{P}^{-1}_{1:i-B-1,1:i-B-1}\mathbf{P}^{\top}_{i-B:i,1:i-B-1} \\
\left(\left(\mathbf{\Lambda}_{i-B:i,i-B:i}\right)^{-1}\right)_{1:B+1,B+1} &= \left(\mathbf{P}_{i-B:i,i-B:i} - \mathbf{P}_{i-B:i,1:i-B-1}\mathbf{P}^{-1}_{1:i-B-1,1:i-B-1}\mathbf{P}^{\top}_{i-B:i,1:i-B-1}\right)_{1:B+1,B+1} \\
&= \mathbf{P}_{i-B:i,i} \\
&= \mathbf{P}^{\top}_{i,i-B:i}.
\end{aligned}
$$

Therefore, if we can determine $\mathbf{\Lambda}_{i-B:i,i-B:i}$ for $i = 1, \ldots, T$, we can determine all the non-zero elements of the matrix $\mathbf{P}$. Recall that our functions g_b define the elements of the inverse of the Cholesky decomposition of $\mathbf{\Lambda}$; we can use these to define the elements of $\mathbf{P}_{i-B:i-1,i}$ as a function of the unknown $\left(\mathbf{L}^{-1}\right)_{i,i}$:

$$
\begin{aligned}
\mathbf{g}_i &= \begin{bmatrix} \left(\mathbf{L}^{-1}\right)_{i:i-B} \\ \left(\mathbf{L}^{-1}\right)_{i:i-B-1} \\ \vdots \\ \left(\mathbf{L}^{-1}\right)_{i:i-1} \end{bmatrix} \\
&= \begin{bmatrix} g_B\left(bL, t_i, t_{i-B}\right) \\ g_B\left(bL, t_i, t_{i-B-1}\right) \\ \vdots \\ g_B\left(bL, t_i, t_{i-1}\right) \end{bmatrix} \\
\mathbf{Q}_i &= \text{cholesky}\left(\mathbf{\Lambda}_{i-B:i,i-B:i}\right)
\end{aligned}
$$

$$
\begin{aligned}
(\mathbf{\Lambda}_{i-B:i,i-B:i})^{-1} &= \mathbf{Q}_i^{-\top}\mathbf{Q}_i^{-1} \\
&= \begin{bmatrix}
(\mathbf{Q}_i^{-1})_{1,1} & (\mathbf{Q}_i^{-1})_{2,1} & \dots & (\mathbf{Q}_i^{-1})_{B,1} & (\mathbf{Q}_i^{-1})_{B+1,1} \\
0 & (\mathbf{Q}_i^{-1})_{2,2} & \dots & (\mathbf{Q}_i^{-1})_{B,2} & (\mathbf{Q}_i^{-1})_{B+1,2} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \dots & (\mathbf{Q}_i^{-1})_{B,B} & (\mathbf{Q}_i^{-1})_{B+1,B} \\
0 & 0 & \dots & 0 & (\mathbf{Q}_i^{-1})_{B+1,B+1}
\end{bmatrix} \\
&\quad \times \begin{bmatrix}
(\mathbf{Q}_i^{-1})_{1,1} & 0 & \dots & 0 & 0 \\
(\mathbf{Q}_i^{-1})_{2,1} & (\mathbf{Q}_i^{-1})_{2,2} & \dots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
(\mathbf{Q}_i^{-1})_{B,1} & (\mathbf{Q}_i^{-1})_{B,2} & \dots & (\mathbf{Q}_i^{-1})_{B,B} & 0 \\
(\mathbf{Q}_i^{-1})_{B+1,1} & (\mathbf{Q}_i^{-1})_{B+1,2} & \dots & (\mathbf{Q}_i^{-1})_{B+1,B} & (\mathbf{Q}_i^{-1})_{B+1,B+1}
\end{bmatrix} \\
&= \begin{bmatrix}
\begin{bmatrix}
(\mathbf{Q}_i^{-1})_{1,1} & (\mathbf{Q}_i^{-1})_{2,1} & \dots & (\mathbf{Q}_i^{-1})_{B,1} \\
0 & (\mathbf{Q}_i^{-1})_{2,2} & \dots & (\mathbf{Q}_i^{-1})_{B,2} \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \dots & (\mathbf{Q}_i^{-1})_{B,B}
\end{bmatrix} & \mathbf{g}_i \\
\mathbf{0}_B^\top & \overset{-1}{i,i}
\end{bmatrix} \\
&\quad \times \begin{bmatrix}
\begin{bmatrix}
(\mathbf{Q}_i^{-1})_{1,1} & 0 & \dots & 0 \\
(\mathbf{Q}_i^{-1})_{2,1} & (\mathbf{Q}_i^{-1})_{2,2} & \dots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
(\mathbf{Q}_i^{-1})_{B,1} & (\mathbf{Q}_i^{-1})_{B,2} & \dots & (\mathbf{Q}_i^{-1})_{B,B}
\end{bmatrix} & \mathbf{0}_B \\
\mathbf{g}_i^\top & L_{i,i}^{-1}
\end{bmatrix} \\
&= \begin{bmatrix}
(\mathbf{Q}_i^{-1})^\top_{1:B,1:B}\,(\mathbf{Q}_i^{-1})_{1:B,1:B} + \mathbf{g}_i\mathbf{g}_i^\top & \mathbf{g}_i L_{i,i}^{-1} \\
\mathbf{g}_i^\top L_{i,i}^{-1} & L_{i,i}^{-2}
\end{bmatrix}
\end{aligned}
$$

Now, let us assume that we have already determined $\mathbf{\Lambda}_{i-B:i-1,i-B:i-1}$. Then we can calculate $L_{i,i}$ as follows, recalling again that the diagonal elements of $\mathbf{\Lambda}$ equal 1:

$$\begin{aligned}
1 &= \left(L_{i,i}^{-2} - L_{i,i}^{-2}\left(\mathbf{g}_i^\top\left(\left(\mathbf{Q}_i^{-1}\right)_{1:B,1:B}^\top\left(\mathbf{Q}_i^{-1}\right)_{1:B,1:B} + \mathbf{g}_i\mathbf{g}_i^\top\right)^{-1}\mathbf{g}_i\right)\right)^{-1} \\
&= L_{i,i}^2\left(1 - \left(\mathbf{g}_i^\top\left(\left(\mathbf{Q}_i^{-1}\right)_{1:B,1:B}^\top\left(\mathbf{Q}_i^{-1}\right)_{1:B,1:B} + \mathbf{g}_i\mathbf{g}_i^\top\right)^{-1}\mathbf{g}_i\right)\right)^{-1} \\
L_{i,i} &= \sqrt{1 - \left(\mathbf{g}_i^\top\left(\left(\mathbf{Q}_i^{-1}\right)_{1:B,1:B}^\top\left(\mathbf{Q}_i^{-1}\right)_{1:B,1:B} + \mathbf{g}_i\mathbf{g}_i^\top\right)^{-1}\mathbf{g}_i\right)}
\end{aligned}$$

Additionally, we can calculate $\mathbf{\Lambda}_{i-B:i-1,i}$ as follows:

$$\mathbf{\Lambda}_{i-B:i-1,i} = -\left(\left(\mathbf{Q}_i^{-1}\right)_{1:B,1:B}^\top\left(\mathbf{Q}_i^{-1}\right)_{1:B,1:B} + \mathbf{g}_i\mathbf{g}_i^\top\right)^{-1}\mathbf{g}_i.$$

This means we now have $\mathbf{\Lambda}_{i-B:i,i-B:i}$. Critically, we also have $\mathbf{\Lambda}_{(i+1)-B:(i+1)-1,(i+1)-B:(i+1)-1}$, and can proceed to calculate $L_{i+1,i+1}$. All that remains is to prove we can calculate $L_{i,i}$ for $i = 1, \ldots, T$ is realizing that the cases of $i = 1$ and $i = 2$ are trivial: when $i = 1$, $\mathbf{\Lambda}_{1-B:0,1-B:0}$ is empty, and when $i = 2$, $\mathbf{\Lambda}_{2-B:1,2-B:1} = 1$. Thus we can determine each of the diagonal $\mathbf{\Lambda}$ lambda blocks needed to calcualte each $L_{i,i}$ and produce a banded precision matrix whose inverse is a valid correlation matrix.

We fit four bands, determining the MLE following again using the BFGS algorithm with a third-order back-tracking linesearch. However, rather than using the implementation provided by Mogensen and Riseth (2018), the algorithm was reimplemented with a focus on attaining better runtime performance. Derivatives were still calculated using forward-mode automatic differentiation (Revels et al., 2016).

1.5 Results

We now assess the effectiveness of the four techniques. We do this by calculating the autocorrelations of $\mathbf{x}_i = \mathbf{L}_i^{-1}\mathbf{y}_i$, where $\mathbf{\Lambda} = \mathbf{L}\mathbf{L}^\top$. If $\mathbf{\Lambda}$ accurately models the correlation between elements of $\mathbf{y}_i$, the elements $\mathbf{x}_i$ will be uncorrelated. We examined the results for the test satellites previously mentioned.

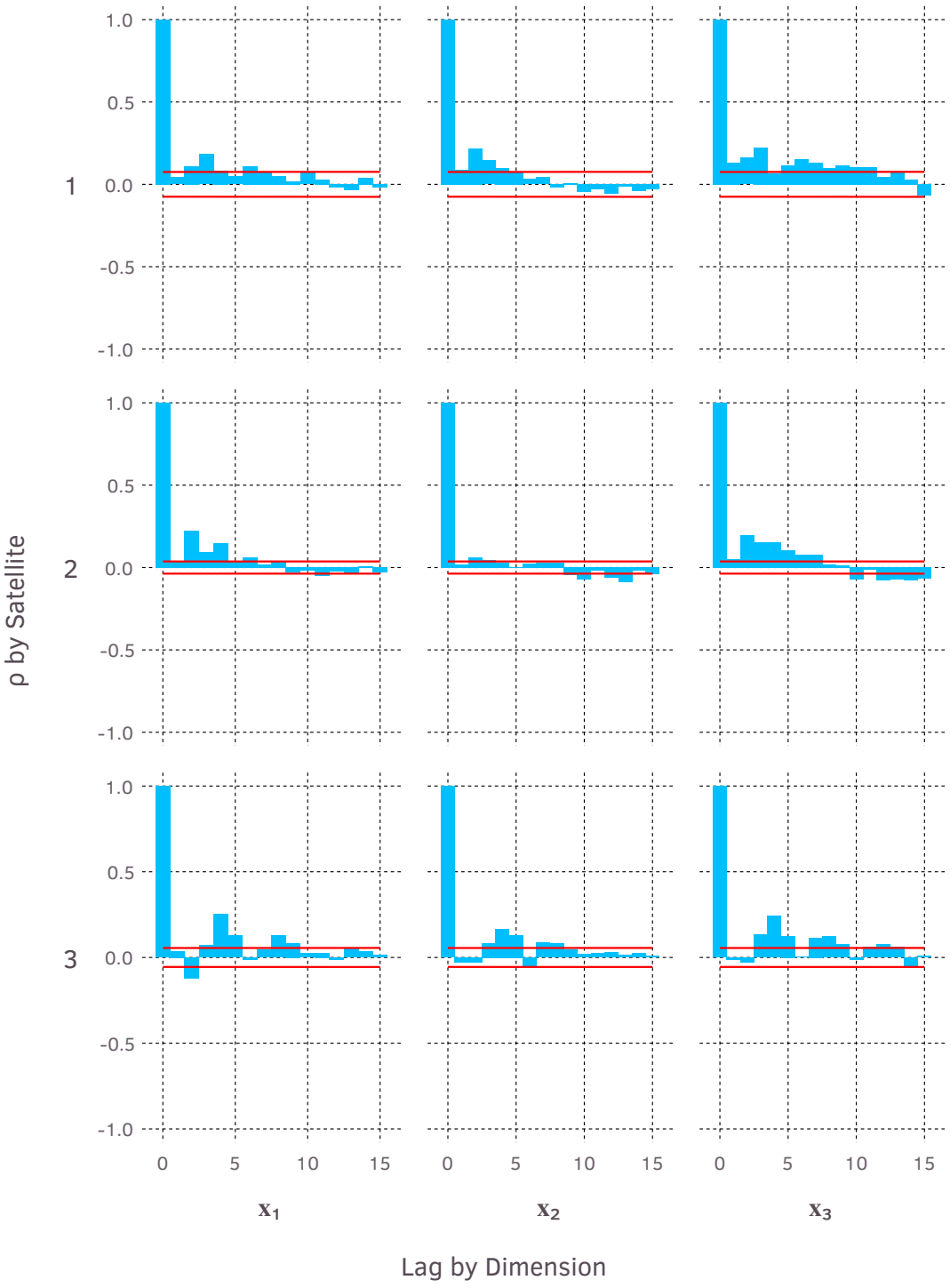

Figure 1.5. Autocorrelations of vectors $\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3$ when $\mathbf{\Lambda}$ was an AR(1) matrix.

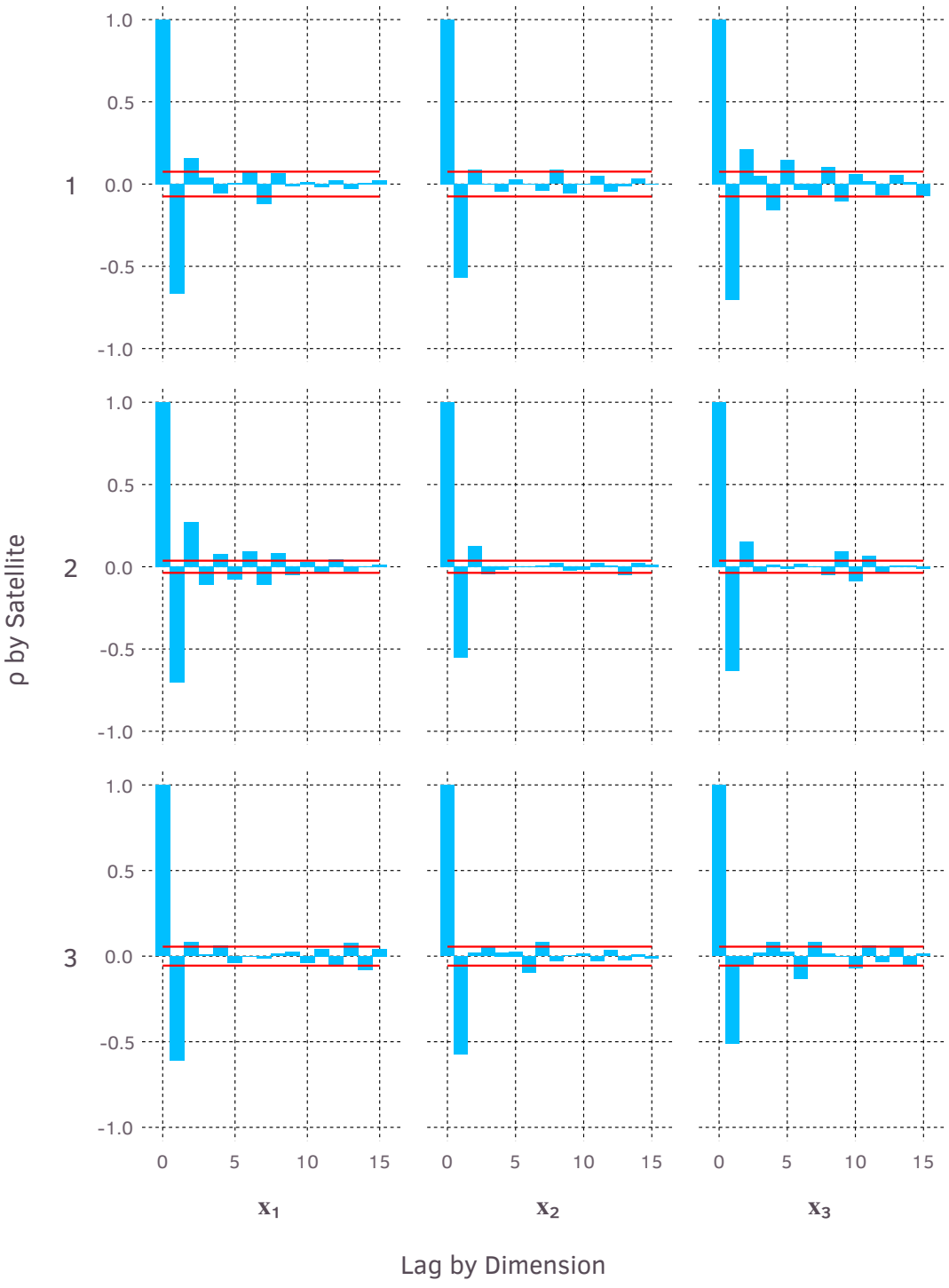

Figure 1.6. Autocorrelations of vectors $\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3$ when $\mathbf{\Lambda}$ was a HODLR matrix.

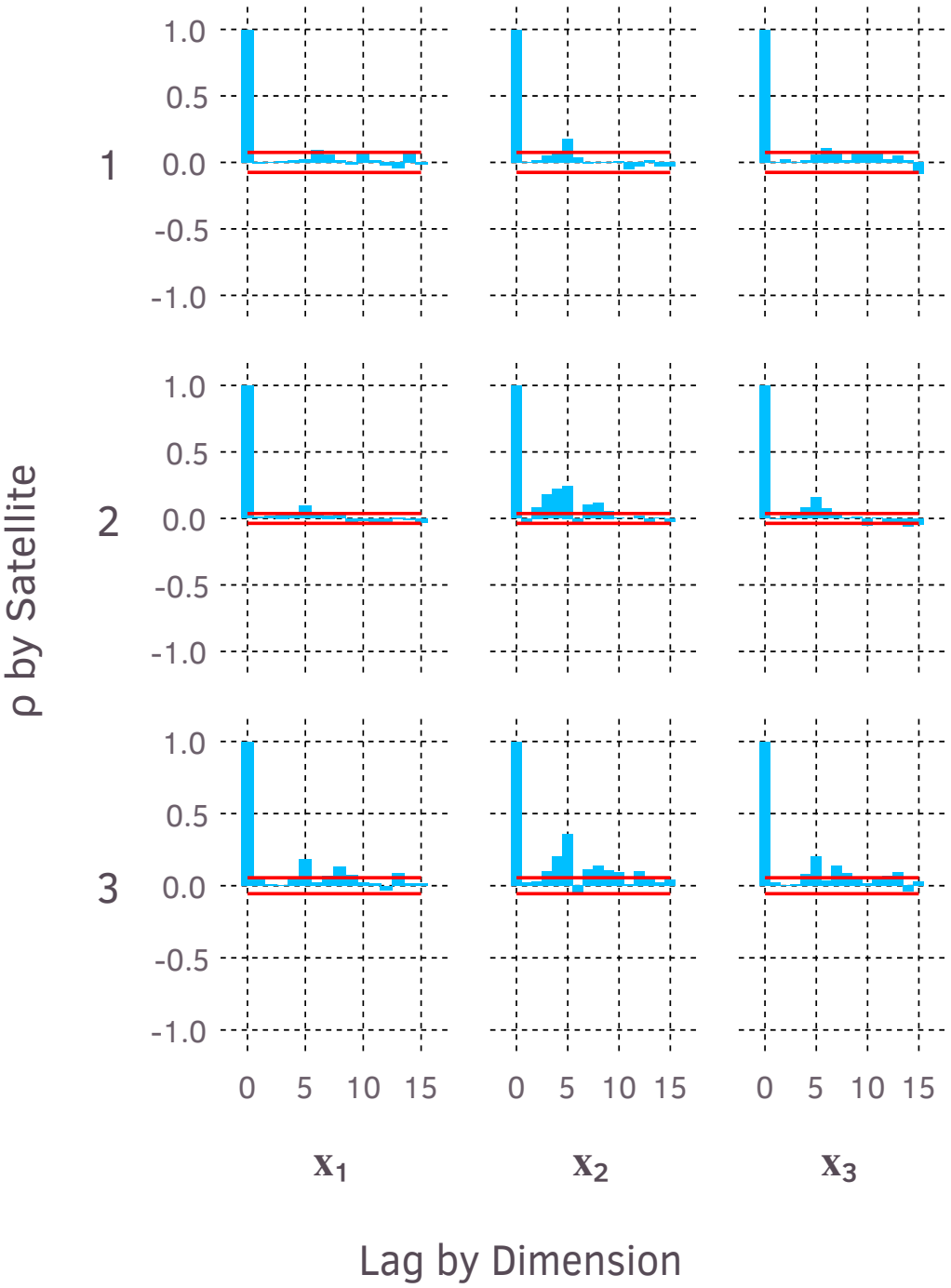

Figure 1.7. Autocorrelations of vectors $\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3$ when the the Cholesky factor of $\mathbf{\Lambda}$ had 4 nonzero subdiagonal bands.

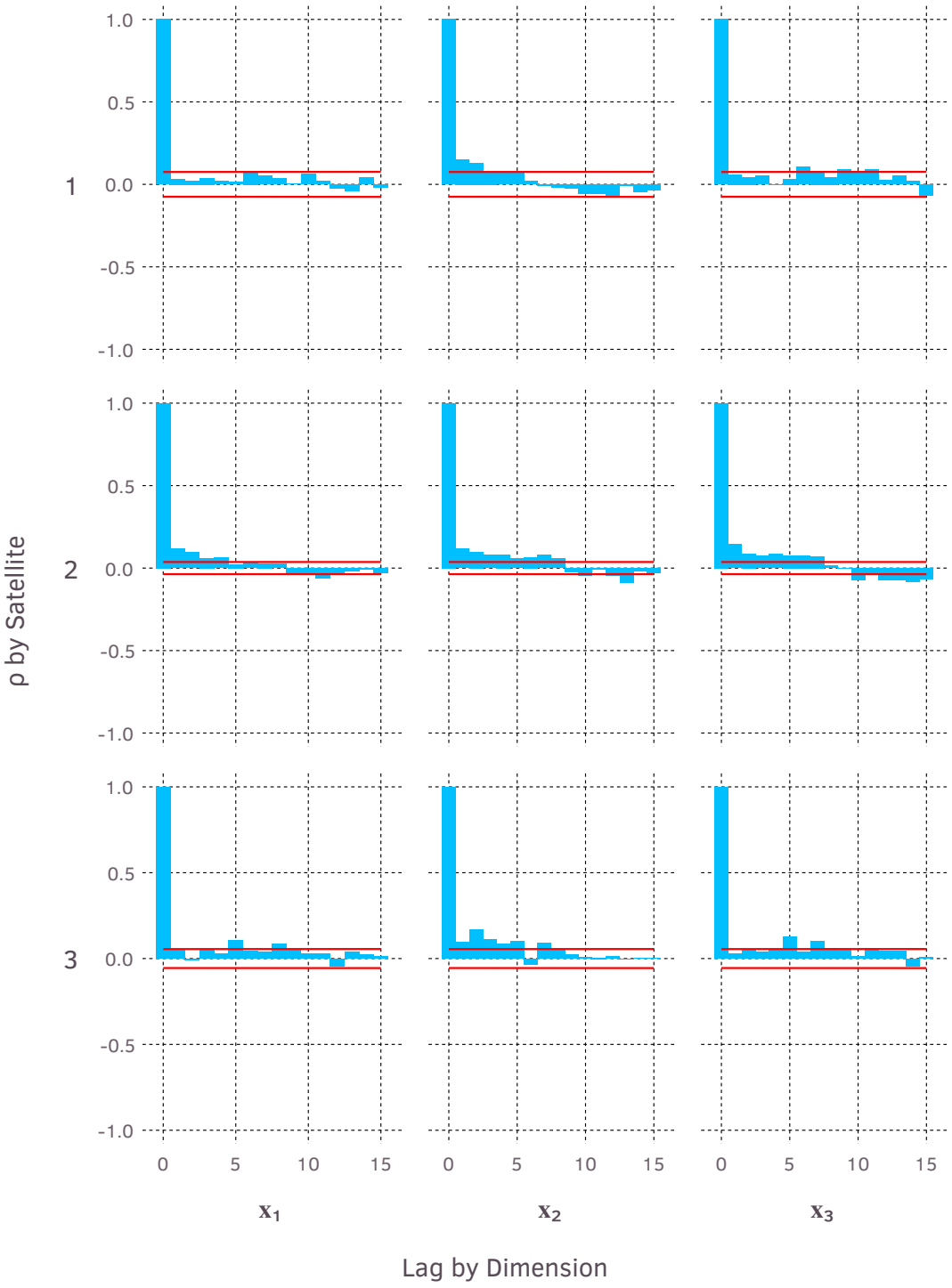

Figure 1.8. Autocorrelations of vectors $\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3$ when the inverse of the Cholesky factor of $\mathbf{\Lambda}$ had 4 nonzero subdiagonal bands.

1.5.1 Autoregressive Matrix

The variables $\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3$ had substantially less autocorrelation than $\mathbf{y}_1, \mathbf{y}_2$, and $\mathbf{y}_3$ (Fig. 1.5). However, autocorrelations of $0.2 - 0.25$ were still common for lags two through five.

1.5.2 HODLR Model

The first autocorrelations are all strongly negative, ranging from about -0.5 to -0.75, while the second autocorrelation is often positive (Fig. 1.6). In the three example satellites, it was as high as 0.24 and 0.34.

Additionally, the implementation of this method was more than two hundred times slower than the AR(1) and banded inverse Cholesky methods.

1.5.3 Banded Correlation

We fit four bands, therefore autocorrelation in the fitted results spiked at lag 5 (Figure 1.7). This approach would require at least as many bands as the number of lags that feature non-zero autocorrelation. Because of this, a large number of bands − and therefore parameters − will be needed in practice. Additionally, the number of bands needed is also likely to increase as a function of sample size; given more frequent observations, there will be more observations close enough in time to be correlated. For this reason this model was not deemed to be an appropriate solution.

1.5.4 Banded Inverse Precision

We again fit four subdiagonal bands using the same approach as for the banded Cholesky. The implementation was not manually optimized, and gradients were provided by forward mode automatic differentiation rather than analytically. The linear algebra operations were unrolled to allow the compiler to optimize when the number of bands is small. Despite the lack of manual-optimization and the loop-carried dependencies preventing vectorization, this implementation was faster than the AR(1) and HODLR implementations tested.

This method also performed the best of the three tested at reducing autocorrelation in $\mathbf{y}_1, \mathbf{y}_2$, and $\mathbf{y}_3$ (Fig. 1.8). However, these autocorrelations still regularly exceeded the null sapling distributions 2.5% and 97.5% percentiles. We nonetheless deemed the performance adequate for proceeding to fit the vectors $\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3$ with a mixture.

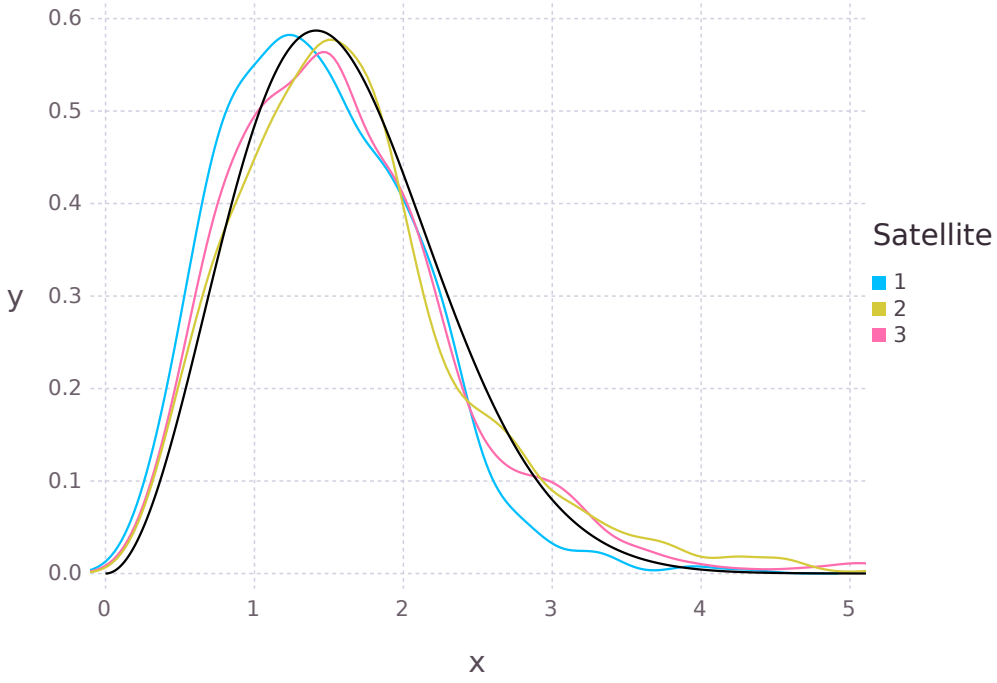

Figure 1.9. Distribution of the Mahalanobis distances of three selected satellites, and the theoretical $\mathcal{X}_3$ distribution.

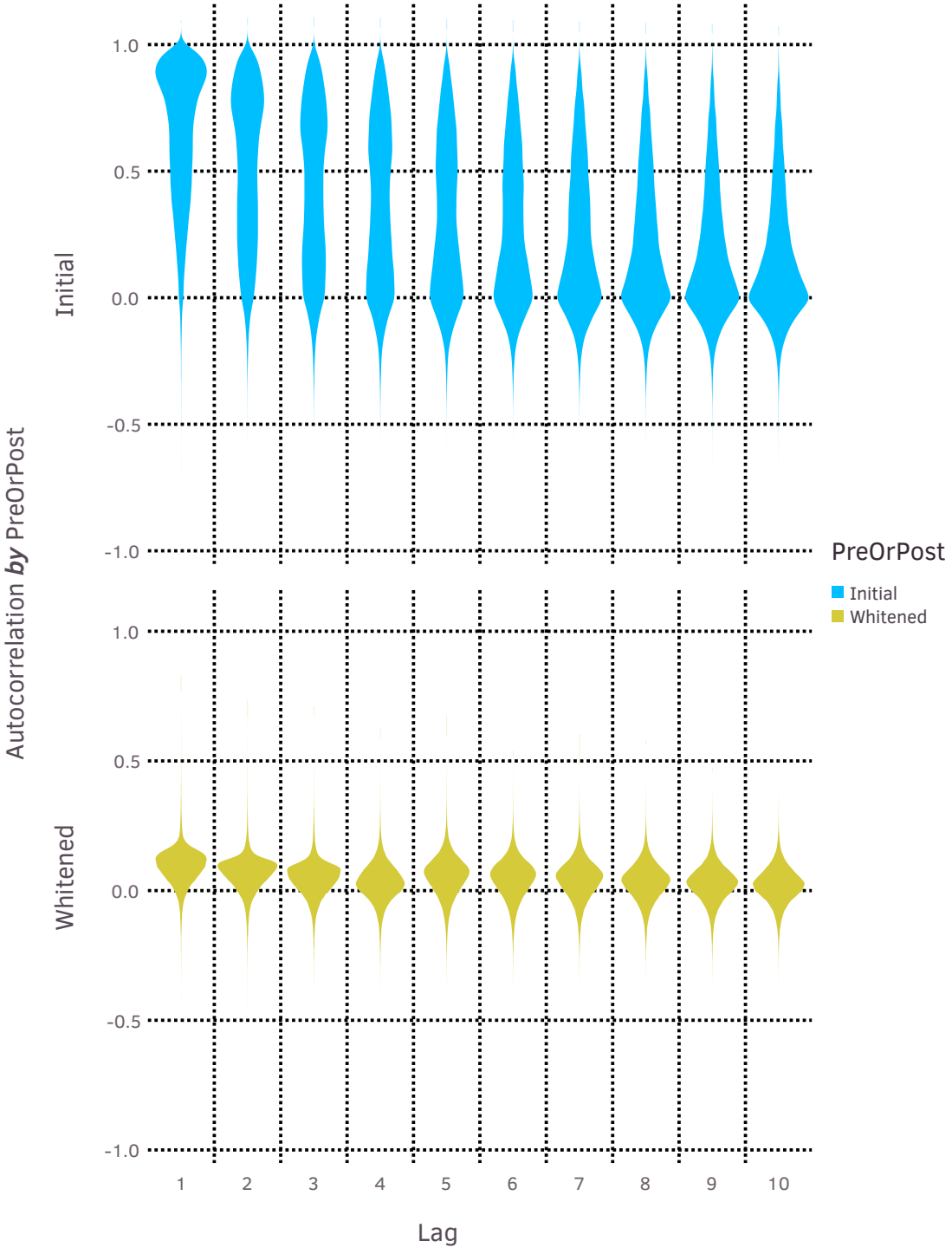

Figure 1.10. Violin plots showing the distribution of autocorrelations at lags 1-10 of the initial and whitened datasets for the second dimension.

1.5.5 Applying the Banded Inverse Cholesky to All the Datasets

Having favored the banded inverse Cholesky, we applied it to data sets on 17,183 satellites and five propagation points. We focus on the second dimension (analogue of $\mathbf{v}$), which manifested the highest autocorrelations. Fig. 1.10 shows the distributions of initial and whitened autocorrelations. While the bulk remained biased above 0, the mean is much closer to zero and the variability in autocorrelation between satellites was substantially reduced.

1.6 Covariance Realism: Mixture Distributions

Recall that if the multivariate normal assumption was correct and the propogated covariance matrices accurate, then $\mathbf{\Sigma}_g = \mathbf{I}_3$, so that

$$m_n^2 = \left(q_n^2 + r_n^2 + s_n^2\right) \sim \mathcal{X}_3^2$$
$$m_n = \sqrt{q_n^2 + r_n^2 + s_b^2} \sim \mathcal{X}_3,$$

where m denotes the Mahalanobis distance. Using the banded inverse-Cholesky model for autocorrelation, the observed distributions do not fit the theoretical $\mathcal{X}_3$ distribution, regularly featuring heavier tails and different means. The apparent behavior of each satellite is unique (Fig. 1.9). Comparing $\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3$, the results of the whitening process, of the three satellites with a $\mathcal{X}_3$ distribution with a one-sample Cramér-von Mises test yielded p values of$< 10^{-30}$ for all three satellites.

For this reason, we seek to improve the accuracy of the covariance matrices by fitting mixture distributions:

$$\mathcal{H}\left(\mathbf{\Sigma}, \boldsymbol{\theta}\right) = \sum_{g=1}^{G} \theta_g \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_g\right),$$

where $\sum_{g=1}^{G} \theta_g = 1$ and $\theta_g \geq 0$ for $g = 1, \ldots, G$. We will remain general in this section, discussing the specific possibilities we evaluated in the following subsections.

Let the error distributions for the primary and secondary be f_p and f_s, respectively. In the calculation of P_c, these are typically both assumed to be multivariate normal, as in section 2.2. We can still utilize this scheme for evaluating P_c when substituting f_s for a

mixture of multivariate normals, $f_s = \sum_{g=1}^{G} \theta_g f_{s,g}(\mathbf{s})$.

$$\begin{aligned} P_c &= \int\int_{|\mathbf{p}-\mathbf{s}|<HBR} f_p(\mathbf{p})\, f_s(\mathbf{s})\, d\mathbf{p} d\mathbf{s} \\ &= \int\int_{|\mathbf{p}-\mathbf{s}|<HBR} f_p(\mathbf{p}) \left[\sum_{g=1}^{G} \theta_g f_{s,g}(\mathbf{s})\right] d\mathbf{p} d\mathbf{s} \\ &= \sum_{g=1}^{G} \theta_g \int\int_{|\mathbf{p}-\mathbf{s}|<HBR} f_p(\mathbf{p})\, f_{s,g}(\mathbf{s})\, d\mathbf{p} d\mathbf{s}. \end{aligned}$$

We can calculate the probability of collision using our mixture distributions by performing a weighted sum of the P_c calculations using each mixture component.

To evaluate the accuracy of our mixtures distributions, we performed five-fold cross-validation on five different propagation points for each satellite. To assess the fits, we then generate a large number of samples m from the mixture distributions, weighing each sample by the probability of the corresponding component. We then smooth the samples with a weighted kernel density estimate, and use them to calculate one-sample Cramér-von Mises test statistics.

1.6.1 Percentile Matching

Hejduk and Johnson (2016) discussed the problem and provided a history of approaches. Unfortunately, earlier authors and practitioners did not have information on residuals available, and thus these solutions were ad hoc and not data driven. The authors proposed a new method, percentile matching, taking advantage of this data.

Formulating this method as a mixture, given a data set of N observations so that $\mathbf{q}, \mathbf{r}$, and $\mathbf{s}$ are vectors of length N, we have

$$\mathcal{H}(\boldsymbol{\Sigma}, \boldsymbol{\theta}) = \sum_{n=1}^{N} \frac{1}{N} \mathcal{N}\left(\mathbf{0}, \frac{F_{\mathcal{X}_3^2}\left(\frac{n}{N+1}\right)}{m_{(n)}^2} \mathbf{I}_3\right), \tag{1.1}$$

where $F_{\mathcal{X}_3^2}\left(\frac{n}{N+1}\right)$ is the cdf of the $\mathcal{X}_3^2$ distribution, and $m_{(n)}^2$ denotes the nth-largest squared Mahalanobis distance.

1.6.2 Finite Scalar Mixture

Laurens et al. (2017) proposed a small finite mixture of scale factors. Precise details on the method and implementation were sparse. Therefore, we cite them as the progenitor of the method discussed here and develop an implementation of their model that we believe to be in the same vein as their proposal.

Here, the model is simply:

$$\mathcal{H}\left(\boldsymbol{\Sigma}, \boldsymbol{\theta}\right) = \sum_{g=1}^{G} \theta_g \mathcal{N}\left(\mathbf{0}, \boldsymbol{\sigma}_g^2 \mathbf{I}_3\right).$$

Laurens et al. (2017) considered a variety of values of G, using undisclosed criteria not only to fit the model but to choose an appropriate scale value for each particular data set. We instead used $G = 4$ and the maximum likelihood estimate of $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\sigma}}$, employing the quasi-Newton BFGS algorithm with a second-order backtracking line search as usual for multivariate optimization problems. We chose a four component mixture because this was the largest number of components Laurens et al. (2017) reported to fit, and mixtures with more components are more flexible. Maximum likelihood is a standard non-Bayesian means of finding point estimates, and the BFGS algorithm is a reliable means of solving for local extrema numerically.

1.6.3 Interior Matrix Mixture.

We propose a mixture model using matrices rather than scalar scale factors. The provided covariance matrices may be biased not only in magnitude, but in orientation and shape. That is, we propose the mixture model with G groups:

$$\mathcal{H}\left(\boldsymbol{\Sigma}, \boldsymbol{\theta}\right) = \sum_{g=1}^{G} \theta_g \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_g\right), \tag{1.2}$$

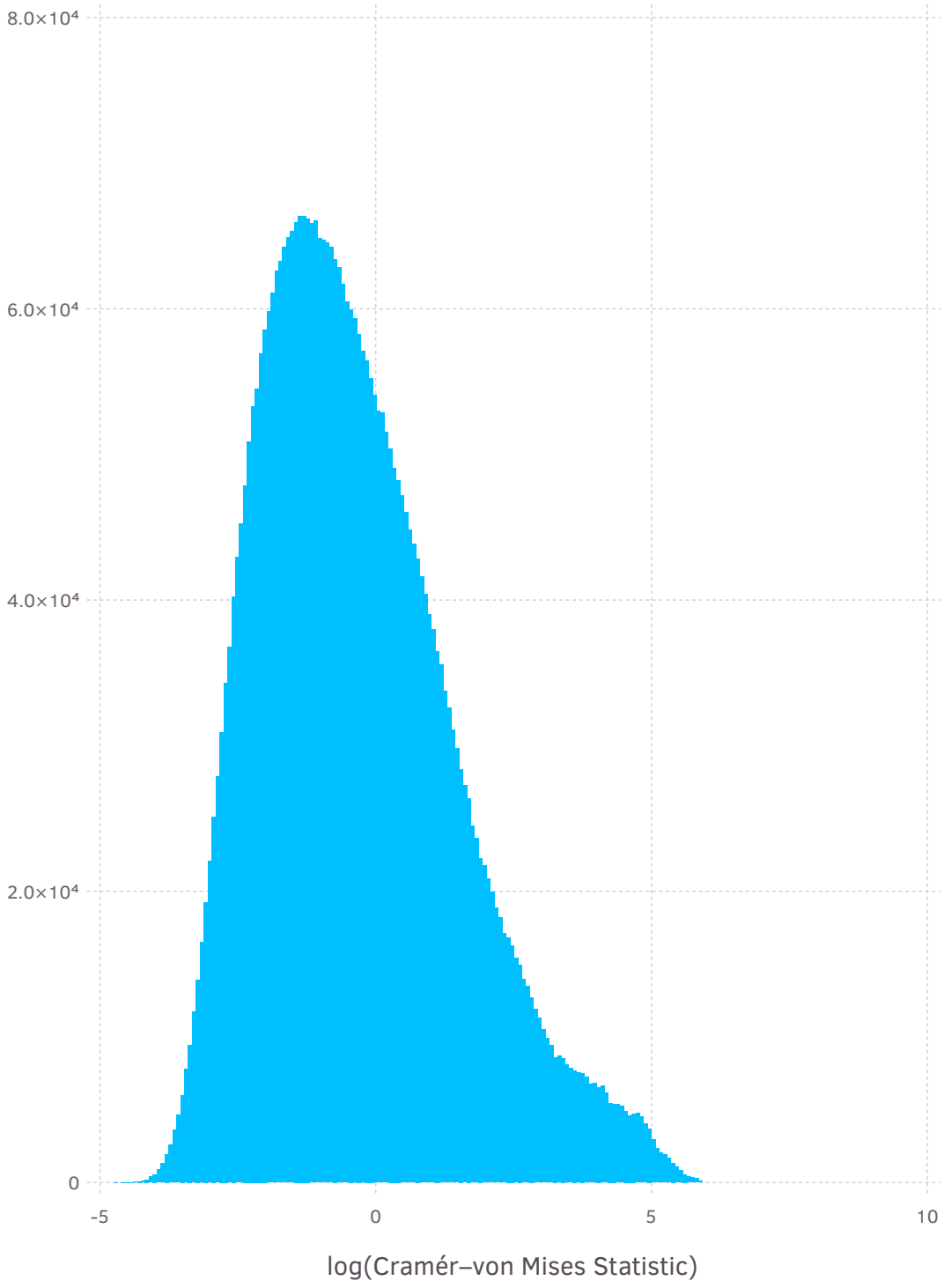

Figure 1.11. Histogram of the 3,951,260 log Cramér-von Mises test statistics.

Table 1.1. Mean effects and standard deviations on the logarithm of Cramér-von Mises test statistics; lower values indicate better fits.

Parameter	Mean	St. Dev	$\frac{\text{Mean}}{\text{St. Dev}}$
Prop Point 5	0	0	
Prop Point 7	0.01671	0.00159	1.052
Prop Point 8	0.00097	0.00159	0.613
Prop Point 9	0.00091	0.00159	0.575
Prop Point 10	0.00472	0.00159	2.972
$\mathcal{X}_3^2$	0	0	
Gibbs: 2	-1.617	0.00213	-758
Gibbs: 4	-1.818	0.00213	-853
Gibbs: 6	-1.863	0.00213	-874
Gibbs: 8	-1.870	0.00213	-877
Gibbs: 12	-1.848	0.00213	-867
Gibbs: 16	-1.800	0.00213	-844
Percentile Matching	-1.443	0.00213	-677
Finite Scalar Mixture	-1.349	0.00213	-633

We have three sets of unknowns in this model:

- $\boldsymbol{\theta}$, the vector of probabilities of group membership.
- $\mathbf{g}$, vector of group memberships.
- $\boldsymbol{\Sigma}_g$, the covariance matrix corresponding to each mixture component.

To enable efficient Gibbs sampling, we assign an Inverse-Wishart($\boldsymbol{\Psi}_0, \nu_0$) prior to each $\boldsymbol{\Sigma}_k$ and a Dirichlet($\boldsymbol{\alpha}$) to $\boldsymbol{\theta}$. In particular, we used $\boldsymbol{\Psi}_0 = b\mathbf{I}_3$, $\nu_0 = 5$, and $\alpha_g = \frac{1.5^{G-g}}{G}$. We chose the Inverse-Wishart priors so that the prior expected value of each covariance matrix is $\mathbf{I}_3$, while the prior variance is undefined, making the prior relatively non-informative. The choice for α_g was to place a decreasing relative prior likelihood on each subsequent group. That is, the prior probability of membership in group g is 1.5 times greater than the likelihood of membership in the following group, $g+1$. Let $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 \end{bmatrix}$,

Then we can implement a Gibbs sampler using the following full conditionals (Smith and Roberts, 1993):

$$\boldsymbol{\Sigma}_g|\mathbf{g},\mathbf{X} \sim \text{Inverse-Wishart}\left(\boldsymbol{\Gamma}_g, \nu_0 + \sum_{t=1}^{T} 1_g\left(g_t\right)\right)$$

$$\boldsymbol{\Gamma}_g = \boldsymbol{\Psi} + \sum_{t=1}^{T} 1_g\left(g_t\right) \begin{bmatrix} X_{t,1}X_{t,1} & X_{t,1}X_{t,2} & X_{t,1}X_{t,3} \\ X_{t,2}X_{t,1} & X_{t,2}X_{t,2} & X_{t,2}X_{t,3} \\ X_{t,3}X_{t,1} & X_{t,3}X_{t,2} & X_{t,3}X_{t,3} \end{bmatrix}$$

$$\boldsymbol{\theta}|\mathbf{g},\mathbf{X} \sim \text{Dirichlet}\left(\alpha_1 + \sum_{t=1}^{T} 1_1\left(g_t\right), \alpha_2 + \sum_{t=1}^{T} 1_2\left(g_t\right), \ldots, \alpha_G + \sum_{t=1}^{T} 1_G\left(g_t\right),\right)$$

$$g_t|\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \ldots, \boldsymbol{\Sigma}_G, \boldsymbol{\theta} \sim \text{Categorical}\left(\rho_1, \rho_2, \ldots, \rho_G\right)$$

$$\rho_g = \frac{\theta_g f\left(\left[X_{t,1}, X_{t,2}, X_{t,3}\right], \mathbf{0}, \boldsymbol{\Sigma}_g\right)}{\sum_{j=1}^{G} \theta_j f\left(\left[X_{t,1}, X_{t,2}, X_{t,3}\right], \mathbf{0}, \boldsymbol{\Sigma}_j\right)},$$

where f is a trivariate normal pdf, and $1_x\left(()\, y\right)$ is the indicator function, equal to 1 if $y \in x$ or $y = x$, and 0 otherwise. The implementation of this sampler was optimized for run-time performance, sampling approximately two thousand times faster than a `JAGS` implementation on x86_64 CPUs supporting the AVX512F instruction set. `Julia`, `C++`, and `C++`/`Fortran` implementations are available on GitHub under the open source MIT license.

1.6.4 Assessing Results

We assessed the model fit for 17,865 satellites, calculating a total of 3,961,260 Cramér-von Mises test statistics. The logarithm of the distribution was only slightly skewed (Fig. 1.11) and reasonably bell shaped, so that a normal model was deemed appropriate.

We fit a standard ANOVA-style model on the Cramér-von Mises statistics recorded during the cross-validation assessment:

$$\log\left(cvm_{s,p,m,i}\right) = \mu_s + \tau_p + \alpha_m + \epsilon_i,$$

where μ_s, τ_p, and α_m refer to the satellite, propogation point, and method mean effects. For identifiability, we set $\tau_5 = 0$ and $\alpha_{nominal} = 0$. That is, the 5^{th} propogation point

and nominal, $\mathcal{X}_3^2$ models were treated as baseline effects. Posterior means and standard deviations are shown in Table 1.1. Note that lower values correspond to higher p-values and better fits.

Propogation point 10 tended to be fit less well by the nominal covariance matrices than the others, $P\left(\tau_{10} > \tau_i, i = 5, 7, 8, 9\right) = 0.96$. The differences between methods were more drastic, with negligible overlap in the posterior distributions. For example, using Monte Carlo to sample from the posterior normal distribution, we calculated that the posterior probability that the Gibbs mixture with 8 components is the most effective method as greater than 0.999.

1.7 Assessing Conjunctions With Nominal $P_c > 10^{-10}$

We assessed a set of 7,564 conjunction events to evaluate how the methods impacted the nominal P_c calculations. Given a mixture model, we can calculate P_c as

$$\begin{aligned} P_c &= \int_{-HBR}^{HBR} \int_{x_0-\sqrt{HBR^2-y^2}}^{x_0+\sqrt{HBR^2-y^2}} f(x,y)dxdy \\ &= \int_{-HBR}^{HBR} \int_{x_0-\sqrt{HBR^2-y^2}}^{x_0+\sqrt{HBR^2-y^2}} \sum_{g=1}^{G} \theta_g f_g(x,y)dxdy \\ &= \sum_{g=1}^{G} \theta_g \int_{-HBR}^{HBR} \int_{x_0-\sqrt{HBR^2-y^2}}^{x_0+\sqrt{HBR^2-y^2}} f_g(x,y)dxdy. \end{aligned}$$

That is, the resulting P_c is a weighted average of the P_c of each mixture component. If each distribution f_g is a multivariate normal, we can efficiently calculate each constituent P_c as in section 2.2.3.

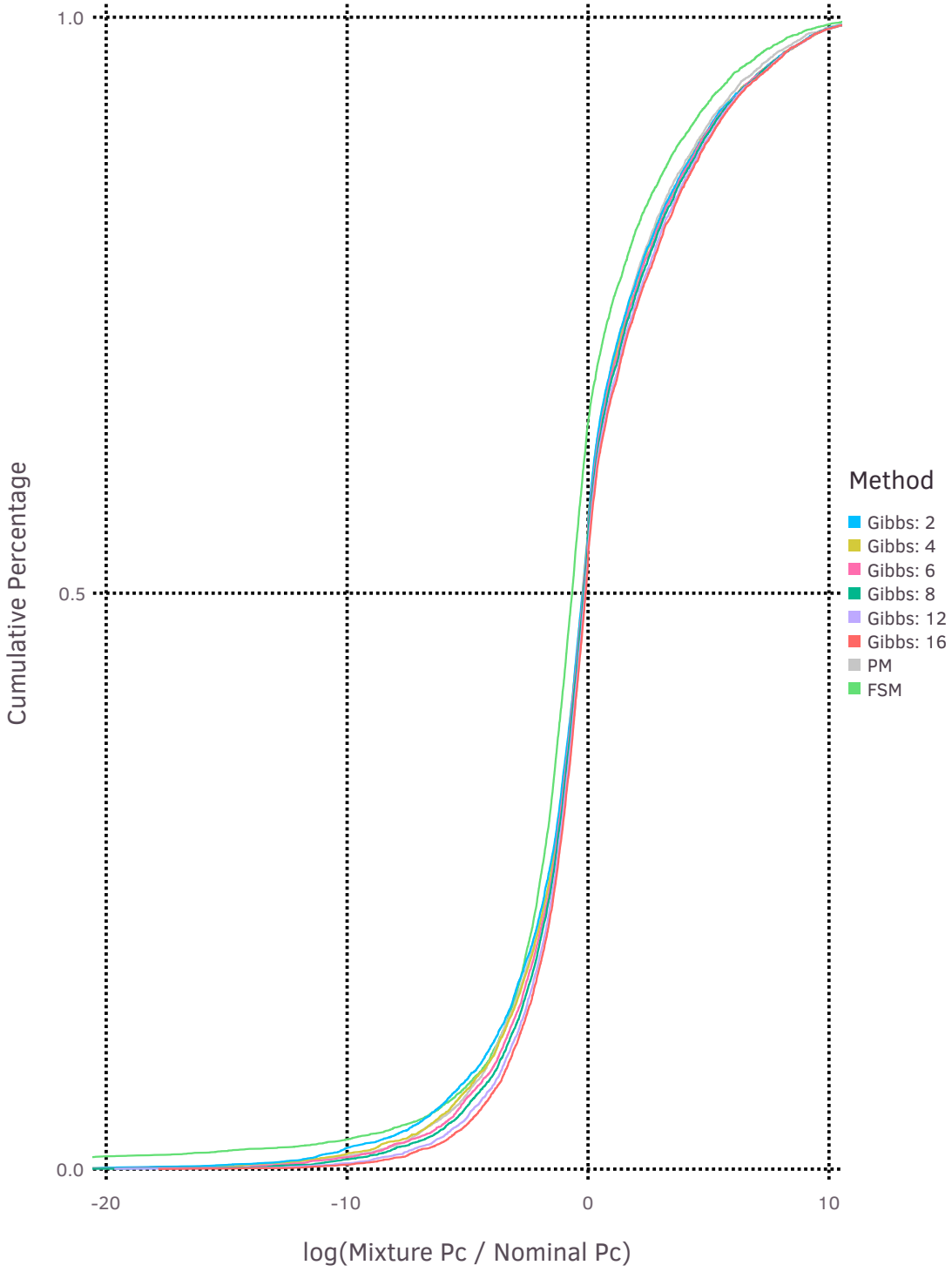

Figure 1.12. Emperical CDF of the log-ratio of mixture P_c values to nominal P_c values. 0 indicates equality.

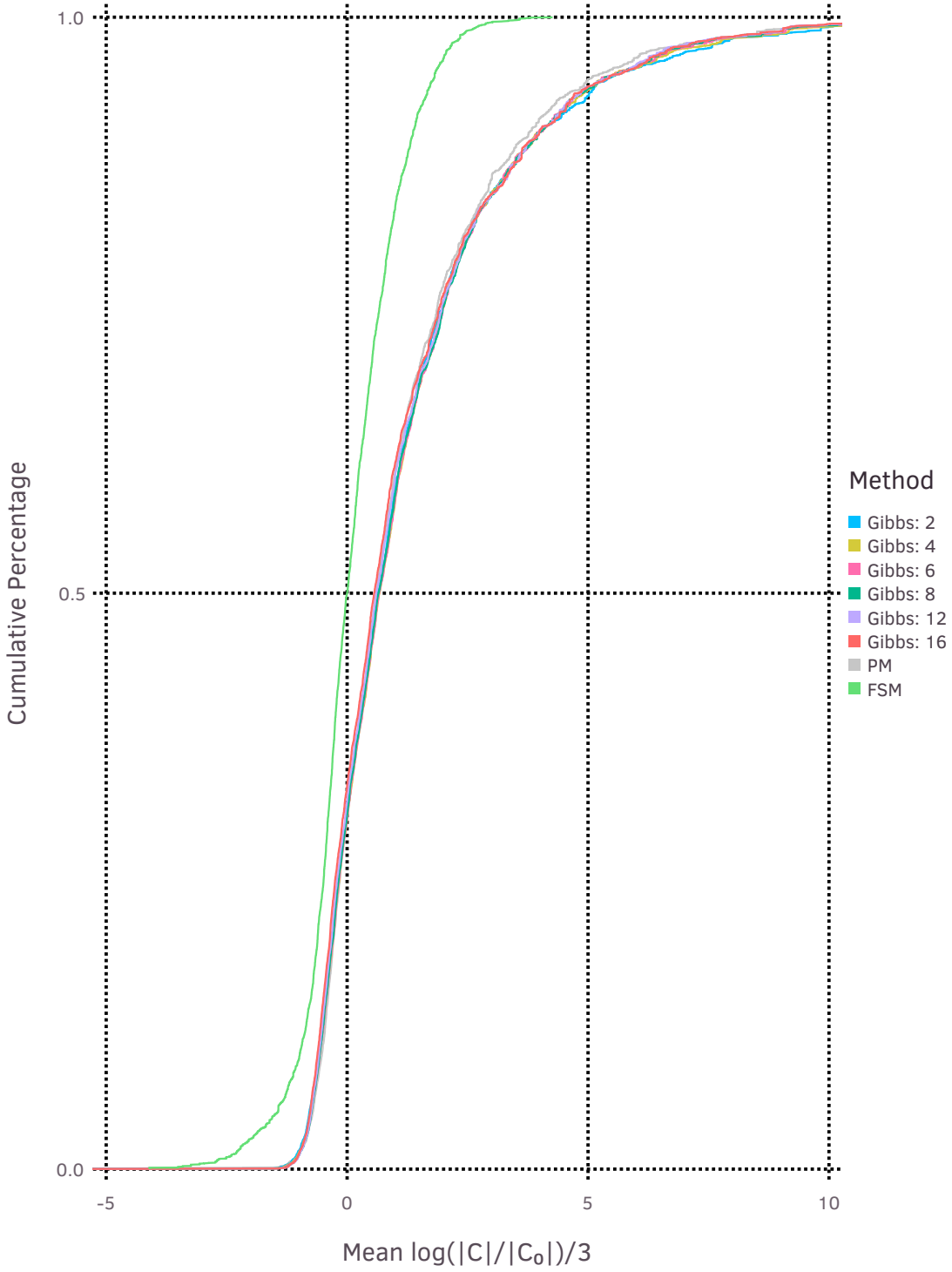

Figure 1.13. Emperical CDF of the expected logarithm of the cube root of the ratio of the determinants of the adjusted to the nominal covariance matrices.

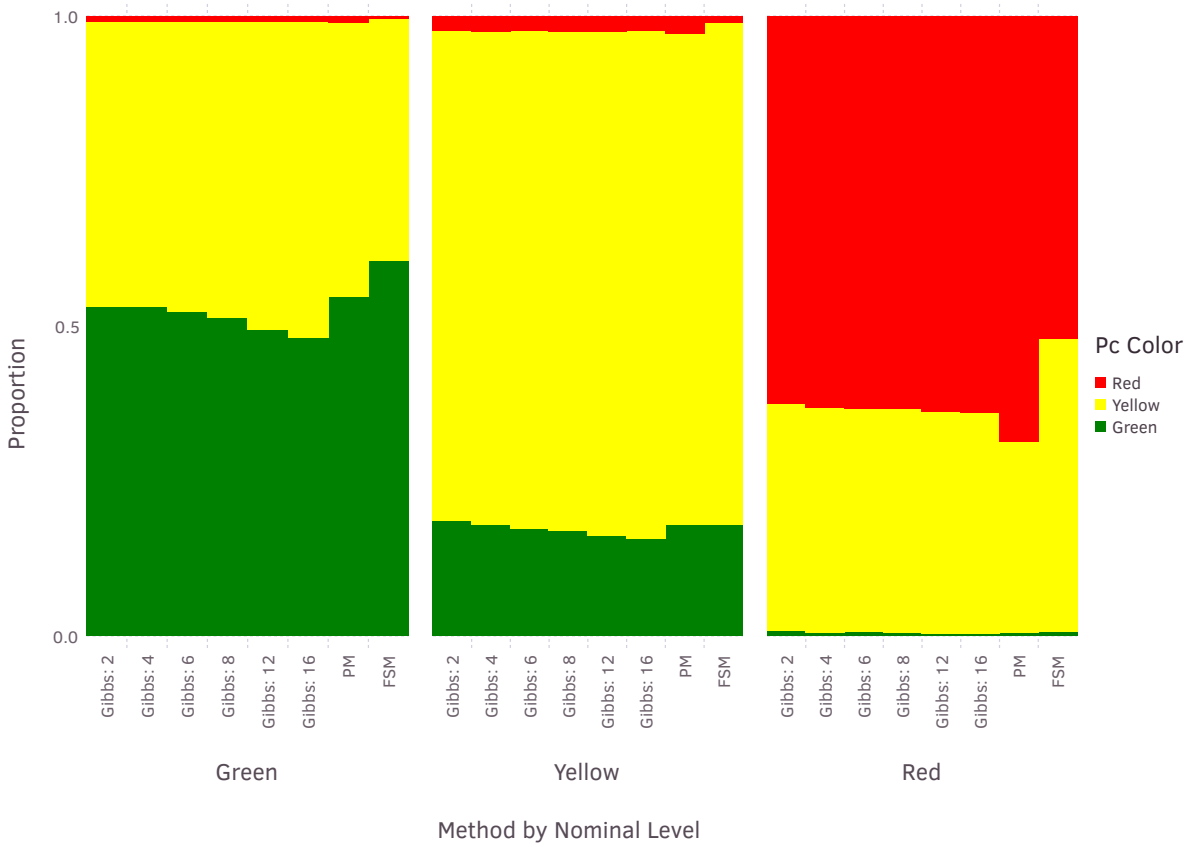

Figure 1.14. Plot showing reclassification rate of P_c threat levels of each method versus the nominal P_c value. Red threads correspond to $P_c \geq 10^{-4}$, yellow $10^{-4} > P_c \geq 10^{-7}$, and green are those such that $10^{-7} > P_c$.

Fig. 1.12 displays an emperical cdf of the log-ratio of P_c values of the mixture models versus the nominal value. While we restricted our analysis to events with nominal $P_c > 10^{-10}$, we did not censor our estimated P_c values, explaining some of the extremity on the left side of the plot. The Bayesian Inverse-Wishart models and percentile matching CDFs were broadly comparable, with the Bayesian Inverse-Wishart model with 16 mixture components broadly assigning the highest P_c values. All of these assigned roughly half of conjunctions a lower than nominal P_c, and the remaining half a higher P_c. The FSM model on the other hand assigned the lowest probabilities of collision, with roughly two thirds assigning lower than nominal P_c values. Figure 1.13 shows the distribution of the mean logarithm of the cube root of the ratio of mixture covariance matrix determinant to nominal

Table 1.2. Percentage agreement on color classification between methods.

	Gibbs: 2	Gibbs: 4	Gibbs: 6	Gibbs: 8	Gibbs: 12	Gibbs: 16	PM	FSM	Nominal
Gibbs: 2	1.0	0.9747	0.9642	0.9582	0.9484	0.9421	0.9126	0.7939	0.6909
Gibbs: 4	0.9747	1.0	0.9855	0.9771	0.9639	0.9541	0.9149	0.7992	0.695
Gibbs: 6	0.9642	0.9855	1.0	0.9906	0.9753	0.9644	0.9101	0.7977	0.6963
Gibbs: 8	0.9582	0.9771	0.9906	1.0	0.9841	0.9728	0.9086	0.7949	0.6953
Gibbs: 12	0.9484	0.9639	0.9753	0.9841	1.0	0.9873	0.9057	0.7955	0.6939
Gibbs: 16	0.9421	0.9541	0.9644	0.9728	0.9873	1.0	0.9004	0.7944	0.6934
PM	0.9126	0.9149	0.9101	0.9086	0.9057	0.9004	1.0	0.8061	0.7044
FSM	0.7939	0.7992	0.7977	0.7949	0.7955	0.7944	0.8061	1.0	0.7162
Nominal	0.6909	0.695	0.6963	0.6953	0.6939	0.6934	0.7044	0.7162	1.0

covariance matrix determinant. This proxy for relative covariance matrix magnitude reflects the generally lower P_c values assigned by the FSM model, by showing it favors generally smaller covariance matrices.

A discrete threat-level hierarchy is often used in practice to simplify decision making and summarize conjunction risk. In particular, conjunctions may be classified as "red" if they have $P_c \geq 10^{-4}$, "yellow" if they have $10^{-4} > P_c \geq 10^{-7}$, and green if $10^{-7} > P_c$. Fig. 1.14 assesses the extent to which conjunctions given these classifications have been reclassified. Percentile matching, as the most conservative method in Fig. 1.12, is the most likely to reclassify a nominally "green" event as "yellow", but least likely to reclassify a nominally "red" event. The two component Gibbs sampler reversed these behaviors, being least likelty to reclassify a "green" event, and most likely to reclassify a "red" event. These differences are deemed operationally significant.

Table 1.2 shows percentage agreement in classification between each of the methods and the nominal P_c values. The matrix-mixture models using a Gibbs sampler agreed in classification more strongly than the other approaches.

1.8 Discussion on Operational Implementation

Conjunction assessment operations are typically conducted by calculating the nominal probability of collision (P_c) and pursuing a mitigation action when that probability exceeds a given threshold; there are of course some additional subtleties, but direct use of the P_c in this way is the standard procedure. It has been known for some time that object covariances are not entirely realistic, but the severity of the problem and the overall effect on the assessed risk of conjunctions had not been established, a difficult undertaking anyway

given that there are no single "truth" covariances that can be constructed ex post facto to determine what the actual collision probability in fact was. The present study, by pursuing a sophisticated technique that characterizes and compensates for data autocorrelation and then models historical Mahalanobis distance distributions by as much as a 16-fold mixture model with a different derivative covariance for each mixture, comes about as close as possible to a definitive solution for the P_c, given the actual data for the event (represented by the nominal states and covariances) and adjustments due to historically-observed covariance errors for the particular secondary object. One could of course deploy this complex model for operational use, accepting the computational and software maintenance costs as necessary investments for higher-fidelity results. A second approach is to examine the present study's comparative results among all of the different models to determine whether any of the simpler models may render acceptable performance and therefore present themselves as candidates.

It is interesting that all of the models have the overall effect of downgrading serious events (red to yellow and yellow to green) and moderately upgrading dispensable events (green to yellow). As pointed out previously, however, different models do this with different intensities: the finite scalar mixture model recategorized the most aggressively while the percentile-matching and Gibbs models the most leniently.

One approach that can be taken is to begin an operational roll-out of the overall functionality using the percentile-matching approach, arguably the simplest of all the models to implement. Because this model recategorizes similarly to the Gibbs models, and does so more leniently for red events than the alternatives, it is conservative and an approach with which satellite conjunction assessment practitioners are likely to be the most comfortable, and it still substantially outperforms the use of the nominal P_c only. As comfort with this alternative calculation approach grows, operations personnel can decide whether moving to one of the more sophisticated models makes sense and seems desirable as a way to focus conjunction mitigation efforts on only those events that are the most likely actually to be dangerous.

CHAPTER TWO

Third Order Asymptotic Approximations for Sample Size Determination

2.1 Introduction

Diagnostic testing is commonly used in the areas of medicine, epidemiology, and quality control. It is often the case that an infallible gold standard is either too expensive or is not available, meaning most if not all subjects are classified with one or more fallible tests. Considerable work has been done on combining the results of multiple fallible tests in order to obtain more accurate estimates of population prevalences along with estimates of the sensitivities and specificities of the tests. As more complicated models are proposed for inference, simulation procedures that are used to study the properties of the estimation methods and for determining sample sizes have become more computationaly expensive.

Bayesian approaches for estimation of a proportion in the presence of misclassification with multiple diagnostic tests have been considered by Joseph et al. (1995) and Johnson et al. (2001) for the conditionally independent case and extended to the conditionally dependent case by Dendukuri and Joseph (2001), among others. Clarke and Jones (2015) consider an interesting extension where multiple tests are used, but for some subjects not all the tests are applied. Extension of these models to the logistic regression case has also been considered by Cheng et al. (2009).

Because of the expense of administering the diagnostic tests, sample size determination has been an important part of these studies. Bayesian sample size determination is often computational in nature. Wang et al. (2002) discuss a simulation based approach to sample size determination based on a two priors approach. One prior distribution, sometimes referred to as the "design" or "sampling" prior, is used to simulate data sets from the prior predictive distribution accounting for pre-experimental uncertainty. The second prior distribution, called the "analysis" or "fitting" prior is used in the computation of the posterior distribution and posterior quantities of interest. Thus when using these methods

to determine a sample size, a large number of data sets must be generated and then simulation based approaches are usually used to compute the posterior distribution, leading to considerable computing time even for problems of modest complexity.

Bayesian approaches to sample size determination for models with misclassification include Rahme et al. (2000), Stamey et al. (2005), Stamey and Gerlach (2007), Cheng et al. (2009), and Wang et al. (2017), among others. Most previous research in this area has used packages such as OpenBUGS, JAGS, or Stan to do the posterior computation via Markov Chain Monte Carlo (MCMC). The MCMC approximations to the posteriors often require significant computing time. Stamey et al. (2005) use a first order normal approximation to the postrerior distribution, which is shown to be accurate for large sample sizes for comparing two binomial proportions when counts are misclassified. Ventura and Reid (2014) provide an approach to determine third-order approximations to posterior distributions, allowing for highly accurate approximations without requiring as much computational time as MCMC methods do.

Here, we extend the work of Stamey et al. (2005), who developed a procedure for determining the sample size required for comparing two proportions with two conditionally independent tests using a first order normal approximation, in two ways. First, the criterion they considered was interval based and was primarily for the two response proportions. Here, we consider a non-inferiority test for determining if the sensitivity and/or the specificity of a cheaper test can be considered non-inferior to a more expensive test. Second, we use a much more accurate, third order approximation to the posterior that can be used for smaller sample sizes.

The rest of the paper is organized in the following way. In Section 3.2 we discuss the models for two diagnostic tests assuming the tests are conditionally independent. In Section 3.3 we review two Bayesian sample size determination criteria. In Section 3.4 we discuss the third order approximation method and give details for our implementation. In Section 3.5 we apply the method to two situations, an interval based method for estimating a single population prevalence and a hypothesis testing method to show non-inferiority for

Table 2.1. Observed data from diagnostic tests for each pathway, where each test is either positive (+) or negative (0).

	+	-	
+	n_1	n_2	
-	n_3	n_4	
			N

the sensitivities of two diagnostic tests. Finally, we provide concluding comments in Section 3.6.

2.2 Models

Suppose interest is in estimating the parameters associated with a prevalence study with multiple imperfect diagnostic tests. Let $D = 1$ denote the subject is truly diseased and let $D = 0$ denote not diseased. Let $T_j = 1$ if the jth test is positive while $T_j = 0$ if the test is negative. We denote the prevalence of the ith population as $\pi_i = \Pr(D = 1)$, the sensitivity of the jth test as $S_j = \Pr(T = 1|D = 1)$ and the specificity of the jth test as $C_j = \Pr(T = 0|D = 0)$. For the sample size determination procedure, interest may be in the population prevalences, the sensitivities, specificities or a function of the parameters such as the difference in the sensitivities. Significant prior information is needed if only a single population is considered. Adding more distinct populations with different population prevalences adds degrees of freedom useful for estimating parameters as long as the sensitivities and specificities remain constant across the populations (Johnson et al., 2001).

The testing procedure works as follows. For a total sample of size N_i from the ith population, subjects are assessed with both tests. The observed data for a single population is as displayed in Table 2.1. The quantities $n_1 - n_4$ form a 2×2 table as displayed in Table 2.1.

When using Gibbs sampling to approximate posterior distributions or the EM algorithm to get maximum likelihood estimators (MLE) or posterior modes, the observed data is often augmented with the unobserved true positives in each of the categories, which simplifies computations because most full conditionals are known distributions. See Joseph

et al. (1995) for a one population example and Johnson et al. (2001) for a multiple population example. For the approximations we implement here, the likelihood based only on the observed data is used. The model uses the following multinomial probabilities:

$$\pi_1 = \pi S_1 S_2 + (1-\pi)(1-C_1)(1-C_2)$$
$$\pi_2 = \pi S_1 (1-S_2) + (1-\pi)(1-C_1) C_2$$
$$\pi_3 = \pi (1-S_1) S_2 + (1-\pi) C_1 (1-C_2)$$
$$\pi_4 = \pi (1-S_1)(1-S_2) + (1-\pi) C_1 C_2.$$

For the one sample, two test case, π is the population prevalence, S_1 and S_2 are the sensitivities of the two tests, and C_1 and C_2 are the specificities of the two tests. The likelihood is then found by raising each of these probabilities to the corresponding n_i in Table 2.1.

We assume beta prior distributions for all prevalences, sensitivities and specificities. For the conditional independence model with just two diagnostic tests, moderately informative priors are required for two parameters to achieve the partial identifiability discussed in Gustafson et al. (2014).

As Hui and Walter (1980) point out, the one population model where both sensitivities and specificities are less than 1 is over-parameterized, regardless of whether the tests are conditionally independent or correlated. However, adding a second, distinct, population yields a model that is identifiable for the conditionally independent case. While the likelihood technically has two maxima, one of these, without the aid of prior information, is a mirror image of the other where all sensitivities and specificities equal one minus that of their mirrored counterparts. That is, the labels have been swapped. Constraining these parameters, done here with the trivial constraints that $S_j > 0.5$ and $C_j > 0.5$, yields an identifiable model.

One motivation for adding multiple populations is to be able to estimate the sensitivities and specificities without having to rely on strongly informative priors. Secondly, interest can be in the simultaneous estimation of multiple population prevalences, possibly

in a hierarchical model, such as Hanson et al. (2003). For these models, we assume the sensitivity and specificity are all constant across the populations.

The full likelihood for the R population case is a simple extension of the one population case where the likelihood is the product of R one-sample likelihoods with a different prevalence for each population.

2.3 Sample Size Determination

Bayesian sample size determination often uses two sets of prior distributions in the sample size algorithm. These have been referred to as 'sampling' and 'fitting' priors by Wang et al. (2002) and 'design' and 'analysis' priors by Brutti et al. (2008). The idea is as follows.

The design priors, denoted by $p_D(\theta)$ are used at the design stage of the experiment and represent information about the portion of the parameter space the experimenter is most interested in investigating. If interest is in a single parameter, for instance one of the population prevalences, this could include a point mass prior on a specific value of interest or a beta distribution that allows for a range of values, weighted by the elicited parameters of the beta distribution. The design prior is then used to compute a prior predictive density,

$$m_{pD} = \int_{\theta} f(y_n, \theta) p_D(\theta) d\theta$$

where the data vector, y_N, is indexed with N to indicate the particular sample size being considered. This prior predictive density is then used as a weighting function, giving data sets more consistent with the design prior more weight than those that are not. Sample size determination can then be performed using interval width, hypothesis testing, or other types of criteria.

The analysis priors, denoted $p_A(\theta)$, are prior distributions that will be used in the final analysis of the observed data in the actual experiment and are used in the sample size determination scheme in the analysis of the data sets determined from the prior predictive density. For instance, suppose interest is in testing the hypothesis that the sensitivity of test 1 is greater than the sensitivity of test 2, thus interest is in the posterior probability $\Pr_{p_A}(S_1 > S_2 | y_n)$. Note that this probability is computed based on the analysis prior.

For illustrative purposes, we consider two criteria applied to two different examples. Modifying our examples to other criteria or other scenarios is straight forward.

2.3.1 Single Prevalence Using Average Length Criterion

For a first example, we assume data for two distinct groups are available and interest is in estimating the prevalence of the first group, π_1, with a 95% interval of length l using the average length criterion (ALC) of Joseph et al. (1997). The average length criterion finds the sample size required that guarantees a pre-specified probability content and achieves a desired interval width on average. Mathematically, this is defined by

$$\mathbb{E}_{m_{pD}} = \int_Y l_A(y_n) m_D(y_n) dy_n$$

where $l_A(y_n)$ is the length of the $(1-\alpha)100\%$ interval computed using the analysis prior. The integral averages over the sample space, yielding an average width for the chosen value of N. We implement a numerical search to find the value of N where the desired width is obtained. After eliciting design and analysis priors, we proceed using the following algorithm.

(1) All model parameters are generated from the associated design priors. If point mass design priors are used, the same parameter values are used for all iterations in the simulations.

(2) For sample size N, generate counts n_1 through n_4 using the appropriate likelihood, accounting for the number of populations being sampled.

(3) Using the analysis priors, fit the Bayesian model and compute the (1 - α)100% interval for π_1.

(4) Repeat steps 1 - 3 B times in order to get a Monte Carlo estimate of the average width for the sample size N.

(5) Compute the average width of the B intervals.

(6) Repeat steps 1 - 5 in a numerical search procedure to determine the required sample size for the desired average interval width.

We initiate the search by providing an initial sample size N_1, and recording the expected interval width at $\left\lfloor \frac{N}{2} \right\rfloor, N$, and $2N$, where $\lfloor x \rfloor$ rounds x down to the nearest integer. Then, in order to propose additional sample sizes in our numerical search, we used ordinary least squared regression to fit $l = \alpha + \beta \log(N)$, recomputing estiamtes $\hat{\alpha}$ and $\hat{\beta}$ after each newly evaluated N on the most recent 75% of evaluated sample sizes. The next proposed size is then calcualted as $\left\lceil e^{\frac{D-\hat{\alpha}}{\hat{\beta}}} \right\rceil$, where D is the desired interval width, and $\lceil x \rceil$ rounds x up to the nearest integer. If we already evaluated the proposed N, we evaluated $\left\lfloor e^{\frac{D-\hat{\alpha}}{\hat{\beta}}} \right\rfloor$ instead. If this too had been evaluated, we terminate the search.

2.3.2 Determining Sample Size for Test of Non-Inferiority

Next, we suppose interest is in showing the sensitivity of a new test is non-inferior to the sensitivity of a standard test. That is, we want to show the sensitivity of the less expensive test is within some non-inferiority margin, δ, that is specified by the study designer. We extend the test and sample size criterion of Chen et al. (2011) to the misclassification model we consider here. Define $h(\theta) = \log\left(\frac{S_2 - 0.5}{1 - S_2}\right) - \log\left(\frac{S_1 - 0.5}{1 - S_1}\right)$ where S_1 is the sensitivity of the cheaper test and S_2 is the sensitivity of the more expensive test. Interest is in testing $H_0 : h(\theta) \geq \delta$ versus $H_1 : h(\theta) < \delta$.

Following Chen et al. (2011), the following quantity is crucial to the sample size determination scheme:

$$\beta_d^N = \mathbb{E}_d[IP(h(\theta) < \delta | y_n) \geq \gamma] \tag{2.1}$$

This quantity takes the expectation across the prior predictive distribution from the design prior of the probability of accepting H_1, that is, non-inferiority. Two design priors are required, one for under the null hypothesis and one for under the alternative. This is so both Type I and Type II errors can be controlled. The posterior probability, $P(h(\theta) < \delta | y_N)$, is computed based on the analysis prior. The algorithm is as follows:

(1) All model parameters are generated from the associated design priors. If point mass design priors are used, the same parameter values are used for all iterations in the simulations.

(2) For sample size N, generate counts n_1 through n_4 using the appropriate likelihood.

(3) Using the analysis priors, fit the Bayesian model and compute the $\Pr\left(h(\theta) < \delta | y_n\right)$. If this probability is greater than γ, record a 1, otherwise, a 0.

(4) Repeat steps 1 - 3 B times in order to get a Monte Carlo estimate of the expectation (3.1) for sample size n.

(5) Compute the expectation.

(6) Repeat steps 1 - 5 in a numerical search procedure to determine the required sample size both under the null and alternative hypotheses.

As in Chen et al. (2011), to determine the total required sample size we choose the maximum of

$$n_{\alpha_0} = \min\left[n : \beta^{(n)}_{\pi_{d0}} \leq \alpha_0\right]$$

and

$$n_{\alpha_1} = \min\left[n : \beta^{(n)}_{\pi_{d1}} \geq 1 - \alpha_1\right]$$

where $\beta^{(n)}_{\pi_{d0}}$ is the probability of a Type I error using the design prior for the null hypothesis and $\beta^{(n)}_{\pi_{d1}}$ is the power computed using the design prior for the alternative hypothesis.

Our numerical search begins by evaluating an initially proposed sample size, and determining if it is too large or too small by our criterion. We then iteratively halve or double the sample size, until we have found two sample sizes, N_l and $2N_l$, where the former is too small and the latter too large. From here, we switch to a bisection algorith, iteratively bisecting the interval until we have converged on a single sample size, the minimum meeting our criteria.

2.4 Third Order Asymptotic Approximation

The approximation method we use here is an adaptation of the third order asymptotic approximation for a marginal posterior distribution presented by Ventura and Reid (2014),

specifically, the posterior can be approximated with:

$$\pi_m(\psi|y) \approx \frac{1}{\sqrt{2\pi}} \exp\left(\ell_p\left(\psi\right) - \ell_p\left(\hat{\psi}\right)\right) \frac{\left|j\left(\hat{\psi}, \hat{\boldsymbol{\lambda}}\right)\right|^{\frac{1}{2}}}{\left|j_{\boldsymbol{\lambda}\boldsymbol{\lambda}}\left(\psi, \hat{\boldsymbol{\lambda}}\right)\right|^{\frac{1}{2}}}. \tag{2.2}$$

One difference from the notation of Ventura and Reid (2014) is that we let ℓ be the product of the likelihood and prior, so that we no longer have additional terms for the prior. The parameter of interest is ψ, $\boldsymbol{\lambda}$ are the parameters that are marginalized out, hats refer to the modes, $j\left(\psi, \boldsymbol{\lambda}\right)$ to the observed information matrix and $j_{\boldsymbol{\lambda}\boldsymbol{\lambda}}\left(\psi, \boldsymbol{\lambda}\right)$ is the $\boldsymbol{\lambda}\boldsymbol{\lambda}$ block of the information matrix. $\hat{\boldsymbol{\lambda}}$ values are always maximized conditioning on the concurrent value of ψ. Note that

$$\left|j\left(\hat{\psi}, \hat{\boldsymbol{\lambda}}\right)\right|^{\frac{1}{2}} = \left|j_p\left(\hat{\psi}\right)\right| \left|j_{\boldsymbol{\lambda}\boldsymbol{\lambda}}\left(\hat{\psi}, \hat{\boldsymbol{\lambda}}\right)\right|^{\frac{1}{2}},$$

where $j_p\left(\hat{\psi}\right)$ is the second derivative of the profile likelihood.

Additionally, the approximation to the cdf is

$$\begin{aligned}
\int_{-\infty}^{\psi_0} \pi_m\left(\psi|y\right) d\psi &\approx \boldsymbol{\Phi}\left(-r_p^*\left(\psi_0\right)\right) \\
r_p^*\left(\psi_0\right) &= r_p\left(\psi\right) + \frac{1}{r_p\left(\psi\right)} \log\left(\frac{q_B\left(\psi\right)}{r_p\left(\psi\right)}\right) \\
r_p\left(\psi\right) &= \operatorname{sign}\left(\hat{\psi} - \psi\right) \left[2\left(\ell_p(\hat{\psi}) - \ell_p(\psi)\right)\right]^{\frac{1}{2}} \\
q_B\left(\psi\right) &= \ell_p^{\top}\left(\psi\right) \frac{\left|j_{\boldsymbol{\lambda}\boldsymbol{\lambda}}\left(\psi, \hat{\boldsymbol{\lambda}}\right)\right|^{\frac{1}{2}}}{\left|j\left(\hat{\psi}, \hat{\boldsymbol{\lambda}}\right)\right|^{\frac{1}{2}}}.
\end{aligned}$$

Here, $\ell_p^{\top}\left(\psi\right)$ is the derivative of the profile log likelihood (profile score). Let $s\left(\psi, \hat{\boldsymbol{\lambda}}\right)$ refer to the full score, and subscripts to corresponding blocks. To avoid requiring an explicit form of the profile likelihood, we can calculate the profiled score as

$$\ell_p^{\top}\left(\psi\right) = s_{\psi}\psi, \hat{\boldsymbol{\lambda}} - s_{\boldsymbol{\lambda}}\left(\psi, \hat{\boldsymbol{\lambda}}\right) j_{\boldsymbol{\lambda}\boldsymbol{\lambda}}\left(\psi, \hat{\boldsymbol{\lambda}}\right)^{-1} j_{\boldsymbol{\lambda}\psi}\left(\psi, \hat{\boldsymbol{\lambda}}\right).$$

Our implementation of the asymptotic posterior requires only a function that takes unconstrained parameter values as input and provides the logaritm of the product of the likelihood and the prior. Automatic differentiation is used to obtain all required derivatives, and we avoid requiring explicit derivatives of the profile likelihoods as above by expressing

them as functions of the full likelihood. We then use BFGS for all maximizations. To find roots to produce credible intervals, we first used Newton-Raphson iterations, and once the root search traversed both sides of the interval, it transitions to Brent's method (Brent, 2013).

Two limitations of our approach is that it only produces marginal distributions, and requires unconstrained parameters. Combined, these limitations means that any quantity of interest must be a one-to-one function of an unconstrained parameter in the model. This is our reason for defining $h(\theta) = h(S_1, S_2) = \log\left(\frac{S_2-0.5}{1-S_2}\right) - \log\left(\frac{S_1-0.5}{1-S_1}\right)$. We define η_1 and η_2 as unconstrained parameters, defined on the entire real line. Then we let

$$\begin{aligned} h(\theta) &= \eta_1 \\ S_1 &= \frac{\exp(\eta_1 + \eta_2) + 0.5}{\exp \eta_1 + \eta_2 + 1} \\ S_2 &= \frac{\exp(\eta_2) + 0.5}{\exp \eta_2 + 1}. \end{aligned}$$

The log jacobian of these transformations is

$$2\log(0.5) + \log(2S_1 - 1) + \log(2 - 2S_1) + \log(2S_2 - 1) + \log(2 - 2S_2).$$

The other jacobians are calculated similiarly.

2.4.1 Comparison of Asymptotic Intervals with Markov Chain Monte Carlo

Table 2.2. Sample data sets for two populations with same layout as 2.1.

	+	-		+	-	
+	4	2		21	2	
-	14	80		13	64	
			100			100

We generated two data sets with $N = 100$ using prevalences of 0.05 and 0.25, respectively, and $S_1 = 0.9$, $S_2 = 0.95$, $C_1 = 0.85$, $C_2 = 0.97$ (2.2). We used uniform analysis priors for all constrained parameters. We fit the model using the No U-Turn Sampling (NUTS) Hamilitonian Monte Carlo (HMC) algoritm, drawing 100,000 samples from

Table 2.3. Posterior Quantiles for Hamiltonian Monte Carlo and our 3rd order asymptotic approximation.

Parameter	Method	ESS	2.5%	25&	75%	97.5%
π_1	Asymptotics	N/A	0.01738	0.04307	0.089	0.1613
π_1	NUTS	1,138,000	0.01718	0.04413	0.09125	0.1646
π_2	Asymptotics	N/A	0.17	0.232	0.3128	0.3992
π_2	NUTS	1,631,000	0.1718	0.2351	0.3159	0.4017
$h(S_1, S_2)$	Asymptotics	N/A	-2.35	0.2208	2.47	5.091
$h(S_1, S_2)$	NUTS	968,000	-2.222	0.2957	2.526	5.316
$h(C_1, C_2)$	Asymptotics	N/A	-4.768	-2.731	-1.581	-0.6337
$h(C_1, C_2)$	NUTS	667,000	-4.674	-2.712	-1.563	-0.592
S_2	Asymptotics	N/A	0.5414	0.6968	0.8896	0.9878
S_2	NUTS	1,294,000	0.5382	0.6885	0.8835	0.9865
C_2	Asymptotics	N/A	0.9413	0.9678	0.9883	0.9885
C_2	NUTS	1,685,000	0.9414	0.9681	0.9883	0.9884

18 chains. The minimum effective sample size (ESS) among the parameters was 667,000, so that we can use the samples as a low-error approximation of the true posterior.

The resulting 50% and 97.5% equal tailed credible intervals are displayed in 2.3 alongside the corresponding intervals produced by the asymptotic approximation.

2.5 Examples

2.5.1 Estimating a Single Prevalence

We use the summaries of the real data analyses described in Clarke and Jones (2015) to motivate our examples. They considered estimation of rates of *Pucciniapsidii* (*P.psidii*) using two different labs for testing where the labs may differ in terms of sensitivity and specificity. Many observations were tested in both labs, but some were only tested in one lab or the other. In their work, they found that Lab 2 had higher sensitivity and specificity than Lab 1.

Table 2.4. Design and analysis priors for sample size determination with the average width criteron.

Parameter	Design Prior	Analysis Prior
π_1	0.05	Beta(1,1)
π_2	0.15	Beta(1,1)
S_1	Beta(87,13)	Beta(1,1)
C_1	Beta(83,17)	Beta(1,1)
S_2	Beta(95,5)	Beta(1,1)
C_2	Beta(94,6)	Beta(1,1)

We first consider finding the sample size for estimating a prevalence with a specified width. The design and analysis priors are provided in Table 2.4. The design and analysis priors for the sensitivities and specificities have the same means, but are considerably more informative at the design stage. Fixed values are used for the prevalences at the design stage while diffuse priors are used at the analysis stage. We applied the algorithm to determine

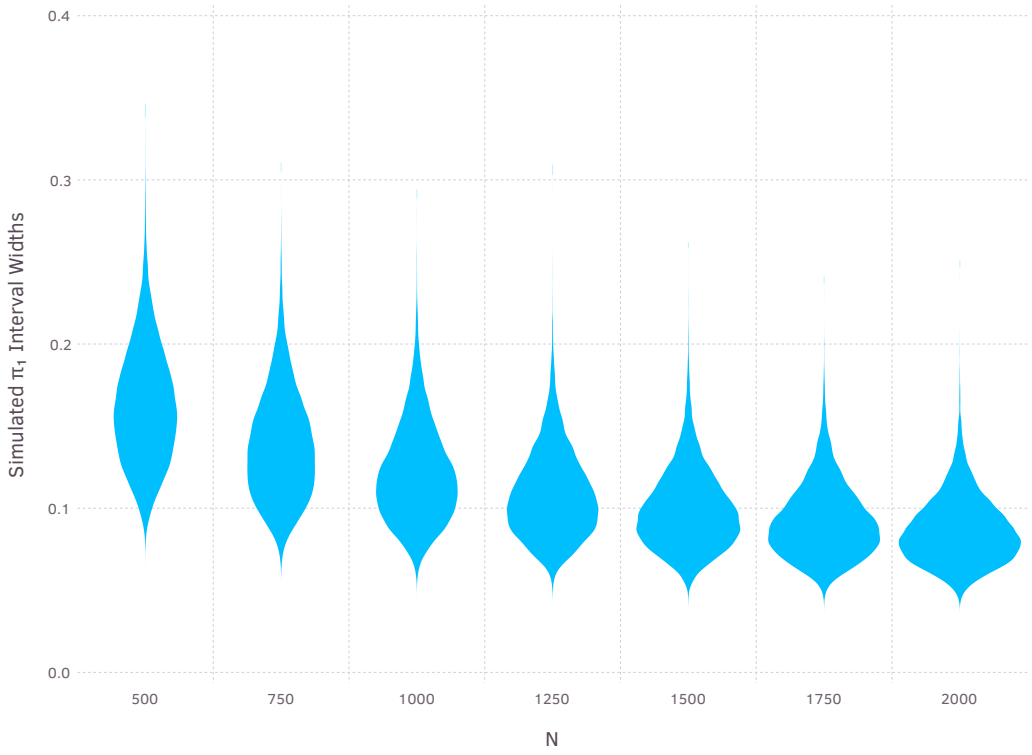

Figure 2.1. Distribution of interval widths as a function of sample size.

Table 2.5. Average Length Criteria search.

Iteration	N	Averager Width
0	500	0.1615
0	1000	0.1209
0	2000	0.08649
1	1529	0.09846
2	1509	0.09925
4	1504	0.09932
5	1500	0.09925
6	1496	0.09991
7	1483	0.09991
8	1482	0.09972
9	1481	0.1003

the sample size required for a 95% interval with an average width of 0.1. Simulating 20,160 data sets per sample size, and beginning with an initial sample size of 1000, the search proceeds as in Table 2.5 before terminating in about 4.25 seconds. Despite the large number of simulated data sets per sample size, the Monte Carlo error still results in imprecission in the search. Nonetheless, it quickly hones in on appropriate sample sizes. Figure 2.1 shows how the distribution of interval widths at various sample sizes.

2.5.2 Power for Non-Inferiority

We next illustrate the procedure in a test for non-inferiority. Here we seek to find the sample size required to determine the sensitivity of the less expensive test/lab is not inferior to the more expensive one. This procedure could be expanded to simultaneously test non-inferiority for the sensitivity and specificity. The design and analysis priors are

Table 2.6. Design and analysis priors for sample size determination for testing non-inferiority.

Parameter	Null Design Prior	Alternative Design Prior	Analysis Prior
π_1	Beta(5,95)	Beta(5,95)	Beta(1,1)
π_2	Beta(25,75)	Beta(25,75)	Beta(1,1)
S_1	0.9	0.95	Beta(1,1)
C_1	Beta(83,17)	Beta(83,17)	Beta(1,1)
S_2	Beta(95,5)	Beta(95,5)	Beta(1,1)
C_2	Beta(94,6)	Beta(94,6)	Beta(1,1)

provided in Table 2.6. As described in Section 3.3.2, the algorithm needs to be run twice. Once under the null and then under the alternative. Here, we set $\gamma = 0.99$ and $\delta = 0.81$. This choice of δ was so that if $S_2 = 0.95$, $S_1 \approx 0.9$. For H_0, we set $S_2 = 0.95$ and $S_1 = 0.9$ to determine the Type I error probability. For H_1, we set both S_2 and $S_1 = 0.95$ to determine the power.

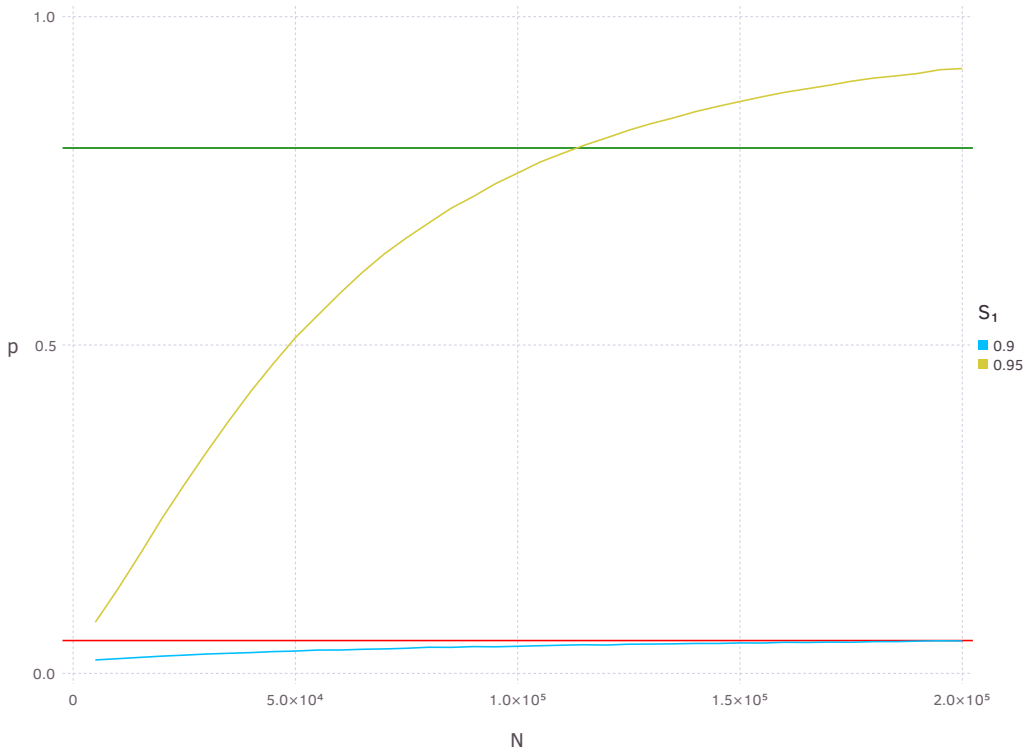

Figure 2.2. Distribution of interval widths as a function of sample size.

The algorithm found 80% power under H_1 at a sample size of about $113,190$, and that $\alpha \approx 0.0437 < 0.05$ at this sample size. Figure 2.2 shows how the power (yellow line) and type 1 error rate (blue line) change as a function of sample size. The green and red lines mark 80% and 5% rejection probabilities. While we have $\alpha < 0.05$ with a sample size sufficient for 80% power, α actually increases as a function of sample size over this range, exceeding 0.05 by $N = 200,000$.

CHAPTER THREE

Totality of Evidence

3.1 What is Totality of Evidence?

DiMasi et al. (2016) reported that in 2013, the out-of-pocket and capitalized costs for drug development were 1.861 and 2.87 billion US dollars respectively, with an annual cost increase of 8.5% greater than inflation since the year 2000. Over this period, clinical development times have increased for diabetes drugs from an average of 4.7 to 6.7 years, while the drug approval success rate declined by 10 percentage

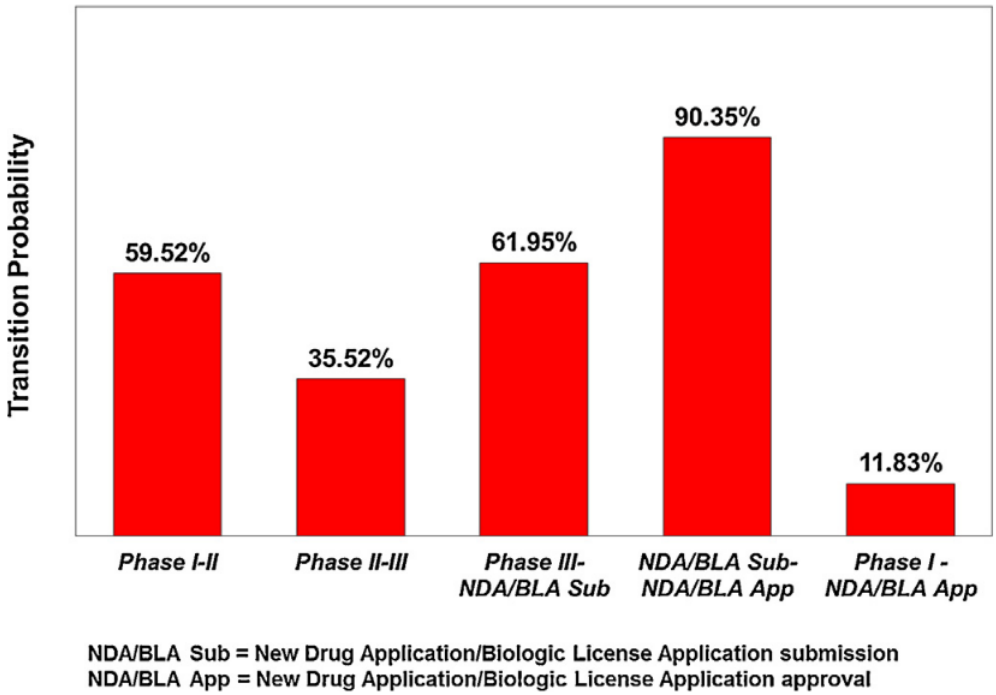

Figure 3.1. Estimated phase transition probability and overall clinical approval success rates for self-originated new molecular entity (NME) and new therapeutically significant biological entity (NBE) investigational compounds first tested in humans anywhere between 1995 to 2007. Figure and caption reproduced from DiMasi et al. (2016).

points, or nearly half, from 21.5% to 11.83% over this period. This 11.83% figure is the product of four individual transition probabilities for a new compound: the probabilities of transitioning from Phase I to Phase II, from Phase II to Phase III, from Phase III to submitting an application liscense, and finally from liscense submission to approval. See Figure 3.1. Accurate internal modeling of the ultimate success probability can enable a pharmaceutical company to reduce patient exposure to ineffective treatments and cut costs by dropping drugs unlikely to be approved earlier, thus allowing a shift in focus to more promising entities.

To this end, judicious modeling will take all available evidence accumulated in the trials into acount. References to 'totality of evidence' in the literature frequently refer to either meta-analysis (Krumholz, 2012; Lewis et al., 2005; Thorning et al., 2016) or a simple aggregation of evidence, as in the step-wise approval process for biosimilars (Holzmann et al., 2016; Strand et al., 2017). However, in this work, references to 'totality of evidence' refers narrowly to the case of incorporating different domains of data gathered in a clinical trial into a single model to enhance understanding of a compound's efficacy, safety, and impact on quality of life. This has the potential to facilitate internal decision making.

Our methods aggregate measurements accross several health outcome domains. Our objectives are similiar to those of Quality of Life (QoL) assessments, such as the Dementia Quality of Life assessment (Brod et al., 1999), Diabetes Quality of Life assessment (Yildirim et al., 2007), or the Diabetes 39 instrument (Boyer and Earp, 1997). These aggregate responses over many different domains. Guyatt et al. (1986) discuss a methodology for developing such assessments. In addition to these, models are likely to incoporate biomedical measurements taken from patients during the trial.

In order to make decisions on whether to proceed with a drug's development, we can specify a utility function. Decision theory is a rich and deep field, beyond the scope of this work. Multi-attribute utility theory is especially relevant here. For an overview of this field see, for example, Mühlbacher and Kaczynski (2016). Rather than explicitly define a utility function and seek to maximize its posterior expectation, we will base a hypothetical

go/no-go decision on relatively simple functions of effect parameters, such as the difference between sums of effect parameters between treatment and standard of care groups.

3.2 Motivating Example: Diabetes

Table 3.1. Domains and end points for diabetes example.

Domain	Endpoint	Abbreviation
Blood Measurements	Glycohemoglobin	A1c
	Glomerular filtration rate	GFR
Vitals	Weight	WT
	Systolic blood pressure	BP
Quality of Life assessment	Energy and mobility	EM
	Diabetes control	DC
	Anxiety and worry	AW

Consider a trial assessing a new drug's impact on diabetes. In this trial, outcomes are obsserved across three domains: blood measurements, vitals, and a quality of life assessment. These domains and their constituent endpoints are summarized in Table 3.1. We will assume we have results from a randomized clinical trial for two treatments, and that measurements are taken periodically over time.

A difficulty in modeling multiple endpoints is that they may have different domains. For example, A1c, GFR, weight, and systolic blood pressure are continuous and have positive support. QoL assessments, which are based on questionnaires, may be ordinal and discrete. Additionally, while high GFR values are healthy, lower A1c, blood pressure, and weight are preferred. In order to compare across endpoints and domains, we assume they are transformed to have unbounded domains, with higher values preferred.

Another difficulty is that similar domains will likely have dependent endpoints. For example, a patient's unusually poor affect on a given visit may induce correlation across domains. It may be tempting to model endpoints such as those in Table 3.1 as if they are mututally independent. This allows straight-forward estimation of summary measures such as mean change from baseline in A1c. However, if the correlations between parameters

are large enough to cause computational problems when parameters are modeled jointly, failure to model them will likely result in biased posterior estimates. Computational issues can sometimes be addressed with reparameterizations. For example, correlation between parameters can often be reduced by switching between centered and non-centered parameterizations, although which parameterization leads to the most efficient sampling may differ between Gibbs sampling and Hamiltonian Monte Carlo (Monnahan et al., 2017).

Since ignoring dependencies between endpoints and across domains can miss important aspects of their relationship, we move to a joint model.

3.3 Modeling Associations

In addition to correlation among endpoints in similar domains, there is a multilevel aspect of such dependency. A drug that helps dementia patients by treating the underlying disease may be expected to improve both memory and mood. We can model this relationship through a hierarchical prior structure. These hierarchical models also have a natural interpretation: the drug treats the disease (modeled at a higher level in the herarchical structure), and we are measuring the symptoms (effects modeled at a lower level). A hierarchical link models the extent to which the drug treats a disease, rather than affects symptoms.

The ability to borrow information between parameters presupposes similarity in scale between these parameters. For example, we may model the drug effect δ with an integrated two-component prediction (ITP) model following Fu and Manner (2010):

$$\delta = \beta \left(\frac{1 - e^{-\kappa t}}{1 - e^{-\kappa d}} \right), \tag{3.1}$$

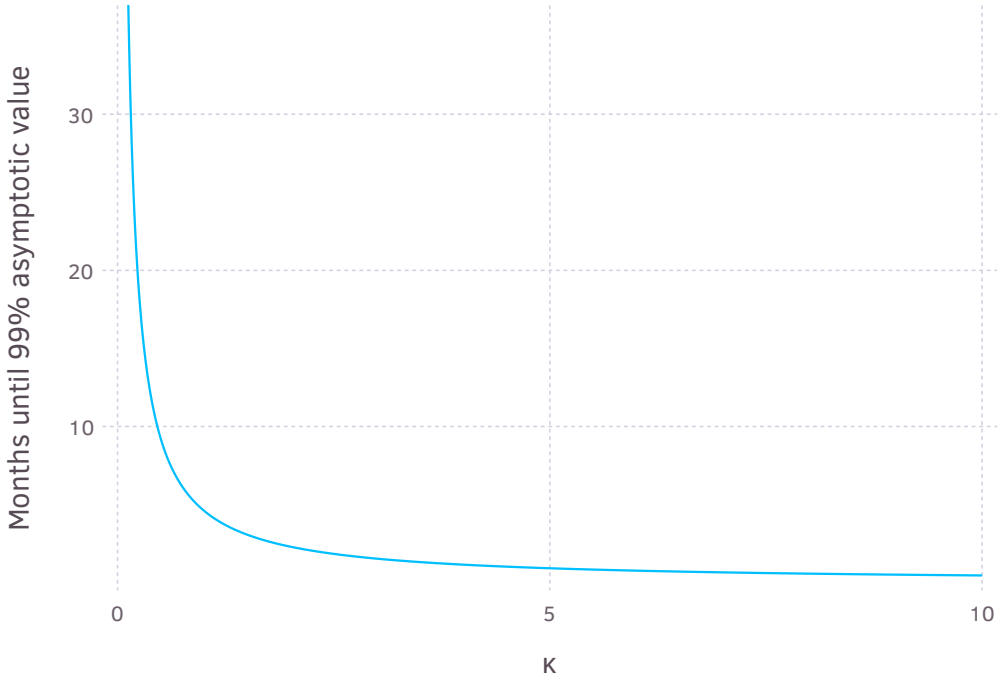

Figure 3.2. Months until 99% of the asymptotic value is reached as a function of κ.

where β controls the expected effect magnitude at the end of the trial, κ controls the shape of drug response over time by controlling the rate at which the drug influence levels off (Figure 3.2), and d is the duration of the trial. To borrow across endpoints and domains, we would have to apply this model as the expected value for each endpoint.

3.4 Ideal Model

Let $\mathbf{Y}_n = [y_{n,t,k}]$ be a $T \times K$ matrix of observations for the n^{th} patient at time t and endpoint k, $n = 1, \ldots, N, t = 1, \ldots, T$, and $k = 1, \ldots, K$. We assume that

$$\mathbf{Y}_n \sim \mathcal{N}\left(f\left(\boldsymbol{\theta}, \mathbf{X}_n\right), C\left(\boldsymbol{\theta}, \mathbf{X}_n\right)\right), \tag{3.2}$$

where $\boldsymbol{\theta}$ is a vector of parameters, f is the expected value function, and C is the covariance function. Here, $\mathbf{X}_n$ is a $T \times P$ matrix of covariates, such that the t^{th} row consists of the P covariates measured at time t.

As our covariance function, we let

$$C\left(\boldsymbol{\theta}, \mathbf{X}_n\right) = \boldsymbol{\Sigma} \otimes \mathrm{AR}\left(\rho, \mathbf{X}_n\right),$$

where $\boldsymbol{\Sigma}$ is a general covariance matrix, '$\otimes$' refers to the Kronecker product, and

$$\mathrm{AR}\left(\rho, \mathbf{X}_n\right) = \begin{bmatrix} 1 & \rho^{x_{n,2,1}-x_{n,1,1}} & \rho^{x_{n,3,1}-x_{n,1,1}} & \rho^{x_{n,4,1}-x_{n,1,1}} & \dots & \rho^{x_{n,T,1}-x_{n,1,1}} \\ \rho^{x_{n,2,1}-x_{n,1,1}} & 1 & \rho^{x_{n,3,1}-x_{n,2,1}} & \rho^{x_{n,4,1}-x_{n,2,1}} & \dots & \rho^{x_{n,T,1}-x_{n,2,1}} \\ \rho^{x_{n,3,1}-x_{n,1,1}} & \rho^{x_{n,3,1}-x_{n,2,1}} & 1 & \rho^{x_{n,4,1}-x_{n,3,1}} & \dots & \rho^{x_{n,T,1}-x_{n,3,1}} \\ \rho^{x_{n,4,1}-x_{n,1,1}} & \rho^{x_{n,4,1}-x_{n,2,1}} & \rho^{x_{n,4,1}-x_{n,3,1}} & 1 & \dots & \rho^{x_{n,T,1}-x_{n,4,1}} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho^{x_{n,T,1}-x_{n,1,1}} & \rho^{x_{n,T,1}-x_{n,2,1}} & \rho^{x_{n,T,1}-x_{n,3,1}} & \rho^{x_{n,T,1}-x_{n,4,1}} & \dots & 1 \end{bmatrix}. \tag{3.3}$$

This allows us to express our model concisely as

$$\mathbf{Y}_n = \mathbf{j}_T \boldsymbol{\beta}^\top \circ \left(\frac{1 - e^{-\mathbf{X}_n \boldsymbol{\kappa}^\top}}{1 - e^{-\left(x_{n,T,1}\mathbf{j}_T\right)\boldsymbol{\kappa}^\top}} \right) + \mathbf{j}_T \boldsymbol{\mu}^\top + \boldsymbol{\epsilon}_n,$$

where $\mathbf{j}_T$ is a column vector of ones with length T, '$\circ$' refers to the Hadamard product, the exponentiation and division operations are element-wise, and

$$\boldsymbol{\epsilon}_n \sim \mathcal{MN}\left(\mathbf{0}, \mathrm{AR}\left(\rho, \mathbf{X}_n\right), \boldsymbol{\Sigma}\right).$$

Here $\boldsymbol{\epsilon}_n$ is a $T \times K$ matrix, $\mathcal{MN}$ denotes the matrix-normal distribution, and AR is the autoregressive matrix defined in (1).

Matrix-variate regression generalizes multiple and multivariate regression, allowing one to fit multiple endpoints across multiple time points (Viroli, 2012). The matrix-normal distribution can be understood as indicating one covariance matrix specifying the correlation structure across rows, and another specifying the relationship across columns. More formally, let the $T \times 1$ vector $\boldsymbol{\epsilon}_{n,:,k}$ denote the k^{th} column of $\boldsymbol{\epsilon}_n$. Similarly, let the $1 \times K$ vector $\boldsymbol{\epsilon}_{n,t,:}$ be the t^{th} row of $\boldsymbol{\epsilon}_n$. Additionally, let $\sigma^2_{k,k}$ be the kth diagonal element of $\boldsymbol{\Sigma}$. Then the matrix-normal distribution indicates that marginally,

$$\boldsymbol{\epsilon}_{n,:,k} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2_{k,k}\mathrm{AR}\left(\rho, \mathbf{X}_n\right)\right), \tag{3.4}$$

and

$$\boldsymbol{\epsilon}_{n,t,:} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}\right)$$

Thus, $\boldsymbol{\Sigma}$ specifies the covariance across end points, while $\mathrm{AR}\left(\rho, \mathbf{X}_n\right)$ indicates the correlation across time.

3.5 Structured Missingness

In ideal applications of this model we would have, for each patient, a $T \times K$ matrix of observations. That is, at each of T time points, all K endpoints would be measured. We will refer to this as the ideal data structure. In practice, this will typically not be available. For example, A1c is a measure of the average blood glucose level over the past three months, and is unlikely to be measured more often than this. Measurements such as blood pressure, on the other-hand, may be taken much more frequently. For this reason, it will often be the case that the matrix of observations, $\mathbf{Y}_n$, will have a substantial amount of structured missingness: endpoints that were deliberately unobserved at given time points.

When fitting a model with Markov chain Monte Carlo (MCMC), multiple imputation is a common approach to handling missing data (Scheffer, 2002). Each missing data point is treated as an unknown parameter and sampled as the algorithm runs. However, this can cause sampling problems when a substantial fraction of the ideal data is missing. The result is an unindentfied model: the total number of parameters can swamp the available data. This is likely to cause problems for MCMC methods, such as Gibbs samplers or Hamiltonian Monte Carlo (HMC) No-U-Turn Samplers (NUTS).

Gibbs samplers iteratively sample from the full conditionals of all parameters in the distribution. If the variance of these conditional distributions is smaller than the variance of the marginals, each new draw from the conditionals must be close to the last draws, resulting in high autocorrelation in the chain. If our model parameters are identifiable, then the conditional distribution of these parameters will depend strongly on the data, and the data's distribution will likewise depend on the parameters. For this reason, substantial amounts of missinessness are likely to cause high autocorrelation and can threaten to devolve the sampler into random-walk behavior over practical numbers of iterations.

Similarly, while the NUTS HMC algorithm can make use of a mass matrix to rotate the parameter space, this can only address linear dependencies between parameters, and becomes increasingly difficult to estimate reliably as the number of parameters increases. Unaddressed correlations force NUTS to use a smaller step sizes and higher tree depths to effectively navigate the posterior, and may force the sampler to slow to a crawl.

Furthermore, autocorrelation between draws is likely to be high: the conditional distributions of identifiable parameters depend heavily on the data; if much of this data is being resampled conditional on the parameters, this can result in a slow walk through the parameter space. Therefore, we marginalize out all of the structurally missing data. As we shall see, this is easier to do with a multivariate normal than a matrix normal data model.

To illustrate this method, suppose we have a multivariate normal random vector $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. This can be written in partioned form as

$$\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{1,1} & \boldsymbol{\Sigma}_{1,2} \\ \boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_{2,2} \end{bmatrix} \right).$$

Then the marginal distributation of $\mathbf{y}_1$ is $\mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{1,1})$. By multiplying $\mathbf{y}$ with arbitrary permutation matrices, we can find the marginal distribution of any subset of $\mathbf{y}$ in this fashion.

Returning to our example, with multivariate normal $\mathbf{Y}_n$ as in (3.2), we must then multiply $\mathbf{Y}_n$ by a permutation matrix $\mathbf{P}$ that sorts the elements so all structurally missing observations follow the available data. Now we have

$$\mathbf{P}\text{vec}(\mathbf{Y}_n) \sim \mathcal{N}\left(\mathbf{P}\text{vec}(\boldsymbol{\mu}), \mathbf{P}(\boldsymbol{\Sigma}_2 \otimes \boldsymbol{\Sigma}_1)\mathbf{P}^\top\right).$$

If we have a total of A available data elements, then their marginal distribution is the multivariate normal with mean of the first A elements of $\mathbf{P}\text{vec}(\boldsymbol{\mu})$, and covariance equal to the upper left $A \times A$ block of $\mathbf{P}(\boldsymbol{\Sigma}_2 \otimes \boldsymbol{\Sigma}_1)\mathbf{P}^\top$.

Using a matrix normal forces the covariance matrix of the multivariate normal to follow the highly structured pattern in $\boldsymbol{\Sigma} \otimes \text{AR}(\rho, \mathbf{X}_n)$, as in (3.4). In particular, each column $\boldsymbol{\epsilon}_{n,:,k}$ must follow the same $\mathcal{N}\left(0, \sigma^2_{k,k}\text{AR}(\rho, \mathbf{X}_n)\right)$ distribution. That is, each endpoint has the same correlation structure across time. Using a multivariate normal when we have structured missingness suggests a new model, where we instead allow each endpoint a different degree of autocorrelation across time. Let the notation $\mathbf{B}_{:,i}$ and $\mathbf{B}_{i,:}$ refer to the ith column and ith row of the matrix $\mathbf{B}$. Addtionally, let ρ_k be the autocorrelation corresponding to the kth endpoint, $\mathbf{A}$ be the block diagonal matrix with diagonal blocks

ARρ

$$\mathbf{A} = \begin{bmatrix} \text{AR}\left(\rho_1, \mathbf{X}_n\right)^{-1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \text{AR}\left(\rho_2, \mathbf{X}_n\right)^{-1} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \text{AR}\left(\rho_K, \mathbf{X}_n\right)^{-1} \end{bmatrix},$$

$\boldsymbol{\Upsilon}$ be the inverse of the upper triangular Cholesky decomposition of $\boldsymbol{\Sigma}$, and

$$\begin{aligned} \mathbf{U} &= \begin{bmatrix} \mathbf{I}_T\Upsilon_{1,1} & \mathbf{I}_T\Upsilon_{1,2} & \dots & \mathbf{I}_T\Upsilon_{1,K} \\ \mathbf{I}_T\Upsilon_{2,1} & \mathbf{I}_T\Upsilon_{2,2} & \dots & \mathbf{I}_T\Upsilon_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{I}_T\Upsilon_{K,1} & \mathbf{I}_T\Upsilon_{K,2} & \dots & \mathbf{I}_T\Upsilon_{K,K} \end{bmatrix} \\ &= \boldsymbol{\Upsilon} \otimes \mathbf{I}_T. \end{aligned}$$

Then, for the quadratic form of our multivariate normal, we have

$$\begin{aligned} q\left(\mathbf{Y}, \boldsymbol{\rho}, \boldsymbol{\Upsilon}\right) &\equiv -\frac{1}{2}\left[\sum_{k=1}^{K}\left(\left(\mathbf{Y}_n\boldsymbol{\Upsilon}\right)_{:,k}\right)^{\top}\text{AR}\left(\rho_k, \mathbf{X}_n\right)^{-1}\left(\left(\mathbf{Y}_n\boldsymbol{\Upsilon}\right)_{:,k}\right)\right] \\ &= -\frac{1}{2}\left(\sum_{k=1}^{K}\begin{bmatrix} \sum_{i=1}^{k} y_{n,1,i}\Upsilon_{i,k} \\ \sum_{i=1}^{k} y_{n,2,i}\Upsilon_{i,k} \\ \vdots \\ \sum_{i=1}^{k} y_{n,T,i}\Upsilon_{i,k} \end{bmatrix}^{\top}\text{AR}\left(\rho_k, \mathbf{X}_n\right)^{-1}\begin{bmatrix} \sum_{i=1}^{k} y_{n,1,i}\Upsilon_{i,k} \\ \sum_{i=1}^{k} y_{n,2,i}\Upsilon_{i,k} \\ \vdots \\ \sum_{i=1}^{k} y_{n,T,i}\Upsilon_{i,k} \end{bmatrix}\right) \\ &= -\frac{1}{2}\left[\sum_{k=1}^{K}\left(\sum_{i=1}^{k}\Upsilon_{i,k}\mathbf{Y}_{n,:,i}\right)^{\top}\text{AR}\left(\rho_k, \mathbf{X}_n\right)^{-1}\left(\sum_{i=1}^{k}\Upsilon_{i,k}\mathbf{Y}_{n,:,i}\right)\right] \end{aligned}$$

$$
\begin{aligned}
&= -\frac{1}{2}\left[\sum_{k=1}^{K}\left(\begin{bmatrix}\mathbf{I}_T\Upsilon_{1,k}\\ \mathbf{I}_T\Upsilon_{2,k}\\ \vdots\\ \mathbf{I}_T\Upsilon_{K,k}\end{bmatrix}^\top \operatorname{vec}(\mathbf{Y}_n)\right)^\top \operatorname{AR}(\rho_k,\mathbf{X}_n)^{-1}\left(\begin{bmatrix}\mathbf{I}_T\Upsilon_{1,k}\\ \mathbf{I}_T\Upsilon_{2,k}\\ \vdots\\ \mathbf{I}_T\Upsilon_{K,k}\end{bmatrix}^\top \operatorname{vec}(\mathbf{Y}_n)\right)\right]\\
&= -\frac{1}{2}\left[\operatorname{vec}(\mathbf{Y}_n)^\top\left(\sum_{k=1}^{K}\begin{bmatrix}\mathbf{I}_T\Upsilon_{1,k}\\ \mathbf{I}_T\Upsilon_{2,k}\\ \vdots\\ \mathbf{I}_T\Upsilon_{K,k}\end{bmatrix}\operatorname{AR}(\rho_k,\mathbf{X}_n)^{-1}\begin{bmatrix}\mathbf{I}_T\Upsilon_{1,k}\\ \mathbf{I}_T\Upsilon_{2,k}\\ \vdots\\ \mathbf{I}_T\Upsilon_{K,k}\end{bmatrix}^\top\right)\operatorname{vec}(\mathbf{Y}_n)\right]\\
&= -\frac{1}{2}\left[\operatorname{vec}(\mathbf{Y}_n)^\top\left(\mathbf{U}\mathbf{A}\mathbf{U}^\top\right)\operatorname{vec}(\mathbf{Y}_n)\right].
\end{aligned}
$$

The covariance matrix is $\left(\mathbf{U}\mathbf{A}\mathbf{U}^\top\right)^{-1} = \mathbf{U}^{-\top}\mathbf{A}^{-1}\mathbf{U}^{-1}$, and the marginal covariance is the subset corresponding to the available, non-missing data. Here, $\mathbf{U}^{-1}$ and $\mathbf{A}^{-1}$ are

$$
\mathbf{A}^{-1} = \begin{bmatrix}\operatorname{AR}(\rho_1,\mathbf{X}_n) & \mathbf{0} & \dots & \mathbf{0}\\ \mathbf{0} & \operatorname{AR}(\rho_2,\mathbf{X}_n) & \dots & \mathbf{0}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{0} & \mathbf{0} & \dots & \operatorname{AR}(\rho_K,\mathbf{X}_n)\end{bmatrix},
$$

and

$$
\mathbf{U}^{-1} = \boldsymbol{\Upsilon}^{-1}\otimes\mathbf{I}_T,
$$

respectively.

3.5.1 Diabetes Example

Returning to the diabetes example, suppose we have a single covariate, time, so that $P = 1$. Let f be the ITP model (3.1) plus a baseline effect, and let $\boldsymbol{\mu}\in\mathbb{R}^K$. Thus, we have

$$
\begin{aligned}
\mathbb{E}\left[y_{n,t,k,g}\right] &= f\left(\boldsymbol{\theta},\mathbf{X}_n\right)\\
&= f\left(\beta_{k,g},\mu_k,\kappa,x_{n,t,1},x_{n,T,1}\right)\\
&= \beta_{k,g}\left(\frac{1-e^{-\kappa_k x_{n,t,1}}}{1-e^{-\kappa_k x_{n,T,1}}}\right)+\mu_k,
\end{aligned}
$$

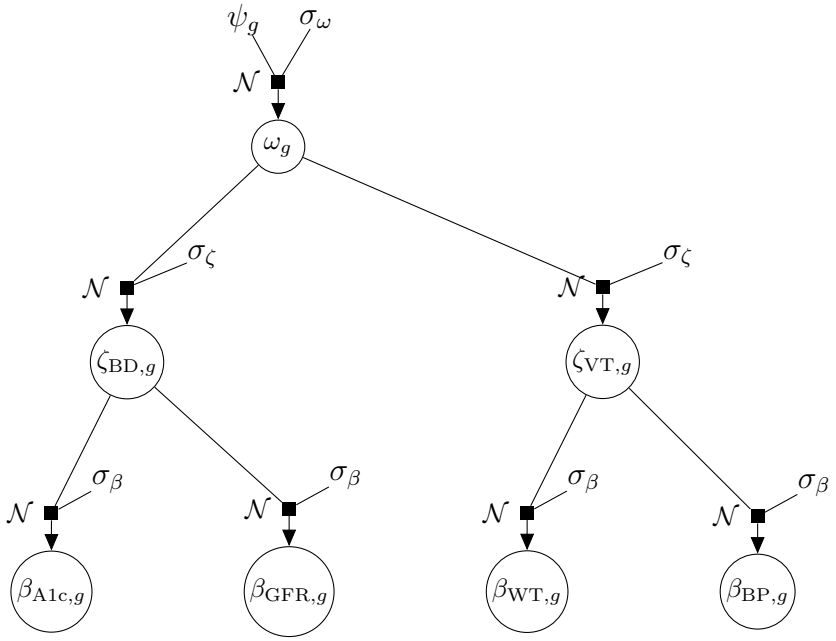

Figure 3.3. Structure of the parameter hierarchy on the β parameters for the first two domains: blood and vitals.

where the subscript $g = 1$ indicates standard of care, and $g = 2$ indicates treatment. The parameter $\beta_{k,g}$ determines the magnitude of the effect observed at the last time point, $x_{n,T,1}$, and κ_k determines the rate of convergence to the ultimate asymptotic value.

For $\boldsymbol{\beta}_g$, the vector with length K such that the k^{th} element is $\beta_{k,g}$, we use a three-tiered hierarchical structure (Figure 3.3). That is, at the bottom, we have K parameters $\beta_{k,g}$, one for each endpoint. Those $\beta_{k,g}$ corresponding to endpoints within the same domain share a common prior distribution. There is one such distribution per domain, and these are again linked on the graph by a common domain prior. A domain's prior parameters are also treated as unknown, and given hyperpriors. This allows information to propagate through the graph. We assign the $\beta_{k,g}$'s normal priors, as in Figure 3.3.

For κ_k, we assign the time to reach 99% of the asymptotic value, τ_k, a generalized gamma distribution. It is easier for domain-experts to reason about time to asymptote, therefore this transformation is chosen to facilitate prior ellicitation. Expert opinion about the time until clinical effect levels off can be used to obtain a model as well as upper and lower bounds. Given these three elicited values, we can solve for the three parameteters of the

generalized gamma to attain the same mode and bounds, yielding an induced distribution on κ_k.

The parameters κ_k attain 99% of its asymptotic value when

$$0.99 = 1 - \exp\left(-\kappa_k t\right),$$

$$0.01 = \exp\left(-\kappa_k t\right),$$

and

$$t = -\frac{\log\left(0.01\right)}{\kappa_k}.$$

Hence $\tau_k = -\frac{\log(0.01)}{\kappa_k}$. Letting f be the generalized gamma pdf,

$$f_\tau(\tau_k|\alpha,\beta,\delta) = \frac{\delta\beta^\alpha}{\Gamma\left(\alpha/\delta\right)}\tau_k^{\alpha-1}e^{-(\beta\tau_k)^\delta},$$

we perform a change of variables to find the prior on κ_k. We have

$$\tau_k = -\frac{\log\left(0.01\right)}{\kappa_k}$$

or

$$\tau_k = \frac{\log\left(100\right)}{\kappa_k}$$

so that

$$|\partial\gamma_k| = \frac{\log\left(0.01\right)}{\kappa_k^2}\partial\kappa_k$$

and

$$\begin{aligned} f_\kappa\left(\kappa_k|\alpha,\beta,\delta\right) &= f_\tau\left(\frac{\log\left(100\right)}{\kappa_k}\,\middle|\,\alpha,\beta,\delta\right)\frac{\log\left(0.01\right)}{\kappa_k^2} \\ &= \frac{\log\left(0.01\right)\delta\beta^\alpha}{\Gamma\left(\alpha/\delta\right)\kappa_k^2}\left(\frac{\log\left(100\right)}{\kappa_k}\right)^{\alpha-1}e^{-\left(\frac{\beta\log(100)}{\kappa_k}\right)^\delta}. \end{aligned}$$

The baseline effects, μ_k, are given normal priors. The autocorrelation parameters, ρ_k, of the auto-regressive matrix are given a beta prior, over the support $0 \leq \rho_k \leq 1$ for each k. We restrict ourselves to positive autocorrelations under the assumption that negative

autocorrelations are implausible, and to allow for non-integer time measurements. If all observed times are integer values, the support could be scaled to $-1 \leq \rho_k \leq 1$, and a scaled beta used.

Finally, we let $\boldsymbol{\Sigma} = \boldsymbol{\sigma}\boldsymbol{\Gamma}\boldsymbol{\sigma}$, where $\boldsymbol{\Gamma}$ is a correlation matrix, and $\boldsymbol{\sigma} = [\sigma_{k,k}]$ are diagonal matrices. This partitioning eases interpretation, as it is easier to reason about the marginal standard deviations, the diagonmal elements of $\boldsymbol{\sigma}$, and the correlations than variances and covariances. We assign the $\sigma_{k,k}$'s a common gamma prior, and the correlation matrix an LKJ correlation matrix prior, following Lewandowski et al. (2009). Additionally, we can perform a change of variables to the Cholesky factor of $\boldsymbol{\Gamma}$, using the LKJ correlation Cholesky prior, to sample the Cholesky factors directly, so that $\boldsymbol{\Upsilon}^{-1} = \boldsymbol{\sigma}\mathbf{C}$, where $\mathbf{C}$ is the lower triangular Cholesky factor of $\boldsymbol{\Gamma}$.

We used the rate-parameterization for the gamma distribution, that is $f(x|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma\alpha} x^{\alpha-1} e^{-\beta x}$, and indexed the univariate normal distribution with the standard deviation. The priors we used are

$$
\begin{aligned}
\omega_g &\sim \text{Normal}\left(0, 10\right) \text{ for } g = 1, 2, \\
\zeta_{d,g} &\sim \text{Normal}\left(\omega_g, \sigma_\zeta\right) \text{ for } d = 1, \ldots, D, \\
\beta_{k,g} &\sim \text{Normal}\left(\zeta_{d_k,g}, \sigma_\beta\right) \text{ for } k = 1, \ldots, K, \\
\sigma_\zeta &\sim \text{Half-Normal}\left(0, 10\right), \\
\sigma_\beta &\sim \text{Half-Normal}\left(0, 10\right), \\
\rho_k &\sim \text{Beta}\left(3, 1\right) \text{ for } k = 1, \ldots, K, \\
\frac{\boldsymbol{\kappa}}{\log\left(0.01\right)} &\sim \text{Generalized-Gamma}\left(8.5, \exp\left(-1.5\right), 3\right), \\
\mu_k &\sim \text{Normal}\left(0, 10\right) \text{ for } k = 1, \ldots, K, \\
\sigma_{k,k} &\sim \text{Gamma}\left(1.5, 0.25\right) \text{ for } k = 1, \ldots, K,
\end{aligned}
$$

and

$$
\boldsymbol{\Gamma} \sim \text{LKJ}(2).
$$

We selected nalf-normal priors for the hierarchical standard deviations so that the density plateaus close to zero, allowing for the possibility of pooling domains, or endpoints within a domain. In contrast, for the standard deviation on errors, $\sigma_{k,k}$, we selected a gamma prior with $\alpha > 1$ so that the density approaches zero as $\sigma_{k,k} \to 0$.

3.5.2 Model Assumptions

We assume exchangeability of patients. We additionally assume that the raw data may be preprocessed so that the multivariate normal distribution can reasonably approximate the distribution of the residuals. For example, log transforms may be appropriate for nonnegative measurements such as weight where fluctuations are often regarded as percent changes. Furthermore, in our hierarchical model, expected effects should be similar in magnitude and of the same sign to enable borrowing. In raw values, a successful treatment would lower A1c and raise GFR − or at least stop its decline. If these aren't handled in preprocessing, the associations must be more explicitly handled in the model, for example, expressing each parameter as an optionally negative scale multiple of a baseline. Furthermore, to take advantage of hierarchical borrowing, they must all be on the same scale of variation.

3.6 Posterior Sampling

We fit the model using Hamiltonian Monte Carlo (Betancourt, 2017). Hamiltonian Monte Carlo (HMC) requires a continuous log-density function, and its gradient, rather than the full conditionals. Automatic differention is generally used to elliminate the need to derive or implement gradients. This simplifies fitting complex models without conjugate priors, where finding full conditionals is difficult, as only the density is needed. We fit the model using the Julia library `ProbabilityModels.jl`, a library in active development by the author, which uses source transformations to derive the gradient and optimize the run-time performance of evaluating the log density and gradient. These are then utilized by `DynamicHMC.jl` which implements the NUTS with HMC, following descriptions in Betancourt (2017).

3.7 Simulating Data for the Diabetes Example

In order to validate the model and the behavior of estimators of interest, we simulate datasets with a known truth and fit them. In particular, we will assume that a go/no-go decision requires at least a 95% probability that $\sum_{k=1}^{K} \beta_{k,2} > \sum_{k=1}^{K} \beta_{k,1}$. Equivalently, if we define $\sum_{k=1}^{K} \beta_{k,2} - \beta_{k,1} = \delta_\beta$, our decision hinges on whether we find $\Pr(\delta_\beta > 0) > 0.95$. As the true values of $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ we set

$$\boldsymbol{\beta}_1 = \begin{bmatrix} -0.625 & -0.575 & -0.525 & -0.475 & -0.425 & -0.4 & -0.375 \end{bmatrix}^\top$$

and

$$\boldsymbol{\beta}_2 = \begin{bmatrix} 0.625 & 0.575 & 0.525 & 0.475 & 0.425 & 0.4 & 0.375 \end{bmatrix}^\top$$

so that $\delta_\beta = 6.8$. We randomly generated the true values for the remaining parameters, sampling them from the following distributions, where the normal is parameterized by strandard deviation, and the gamma by rate. Furthermore, we take $\rho_1 \sim \text{Beta}\,(4, 4)$ and $\rho_k = \rho_1, k = 2, \ldots, K$. Continuing, we set

$$\begin{aligned} \sigma &\sim \text{Gamma}(6, 6), \\ \boldsymbol{\Sigma} &\sim \text{Wishart}_{10}(\sigma^2 \mathbf{I}, 10), \\ \frac{\boldsymbol{\kappa}}{\log(0.01)} &\sim \text{Generalized-Gamma}\left(8.5, \exp\left(-1.5\right), 3\right), \\ \boldsymbol{\theta}_d &\sim \text{Normal}\,(0, 5) \;\text{ for } d = 1, \ldots, D, \end{aligned}$$

and

$$\mu_k \sim \text{Normal}\,(\boldsymbol{\theta}_{d_k}, 5) \text{ for } k = 1, \ldots, K,$$

where the subscript d_k indicates the domain corresponding to endpoint k. We use $T = 36$ months. The endpoints GFR, WT, BP, and EM (see Table 3.1) are assumed to be sampled each month, while A1c, DC, and AW are recorded every fifth month, including at the first and last time point.

Table 3.2. Posterior summaries of $\boldsymbol{\kappa}$ and the variance parameters.

Parameter	Truth	Mean	St.Dev.	2.5%	97.5%	ESS	$\hat{r}-1$
κ_{A1c}	0.8400	0.7912	0.1874	0.5381	1.254	968,856	1.607×10^{-6}
κ_{GFR}	0.8996	0.7869	0.1854	0.5356	1.244	1,108,000	6.918×10^{-6}
κ_{WT}	0.7017	0.8099	0.1998	0.5436	1.305	863,381	1.148×10^{-5}
κ_{BP}	0.5701	0.7981	0.1894	0.5417	1.265	772,239	3.342×10^{-5}
κ_{EM}	0.8279	0.7941	0.1894	0.5395	1.261	232,290	5.761×10^{-5}
κ_{DC}	0.6463	0.7958	0.1891	0.5404	1.262	1,101,881	8.455×10^{-6}
κ_{AW}	0.9897	0.7883	0.1873	0.5355	1.249	823,309	1.401×10^{-5}
ρ_{A1c}	0.2838	0.288	0.01541	0.2577	0.3181	1,108,000	6.751×10^{-7}
ρ_{GFR}	0.2838	0.306	0.01161	0.2833	0.3288	1,108,000	6.897×10^{-6}
ρ_{WT}	0.2838	0.2994	0.01211	0.2756	0.3231	1,108,000	7.573×10^{-7}
ρ_{BP}	0.2838	0.2948	0.01275	0.2698	0.3197	1,108,000	1.282×10^{-5}
ρ_{EM}	0.2838	0.2699	0.0184	0.2339	0.306	1,108,000	-6.851×10^{-6}
ρ_{DC}	0.2838	0.3649	0.1055	0.1363	0.5363	881,113	7.07×10^{-6}
ρ_{AW}	0.2838	0.3676	0.1055	0.138	0.5387	922,202	8.2×10^{-6}
σ_{A1c}	2.455	2.444	0.03433	2.377	2.512	1,108,000	5.259×10^{-6}
σ_{GFR}	3.435	3.473	0.03142	3.413	3.536	1,108,000	6.768×10^{-6}
σ_{WT}	3.368	3.356	0.03001	3.298	3.415	1,108,000	-4.323×10^{-6}
σ_{BP}	2.428	2.437	0.02151	2.395	2.479	1,108,000	3.714×10^{-6}
σ_{EM}	3.058	3.058	0.02682	3.006	3.111	1,108,000	4.118×10^{-6}
σ_{DC}	2.275	2.269	0.03881	2.194	2.347	1,108,000	7.285×10^{-7}
σ_{AW}	3.861	3.839	0.06103	3.722	3.961	1,108,000	8.562×10^{-6}
$\Gamma_{GFR,A1c}$	0.2795	0.2837	0.01584	0.2524	0.3145	976,626	7.324×10^{-6}
$\Gamma_{WT,A1c}$	0.4689	0.4572	0.01404	0.4294	0.4844	1,108,000	-4.745×10^{-6}
$\Gamma_{BP,A1c}$	−0.3393	−0.3156	0.01554	−0.3458	−0.2848	774,479	1.616×10^{-5}
$\Gamma_{EM,A1c}$	0.6155	0.6256	0.01157	0.6024	0.6478	828,848	5.097×10^{-6}
$\Gamma_{DC,A1c}$	−0.02754	−0.04352	0.02273	−0.08803	0.00121	1,108,000	4.745×10^{-7}
$\Gamma_{AW,A1c}$	0.2946	0.2856	0.01969	0.2466	0.3239	996,623	1.709×10^{-6}
$\Gamma_{WT,GFR}$	0.1684	0.1794	0.01145	0.1569	0.2018	1,108,000	-6.448×10^{-6}
$\Gamma_{BP,GFR}$	0.31	0.3192	0.01085	0.2977	0.3403	1,108,000	-8.858×10^{-7}
$\Gamma_{EM,GFR}$	0.2866	0.3013	0.01084	0.2799	0.3224	1,108,000	6.075×10^{-6}
$\Gamma_{DC,GFR}$	0.09241	0.1038	0.02186	0.06087	0.1465	1,108,000	-2.408×10^{-6}
$\Gamma_{AW,GFR}$	−0.3809	−0.3648	0.01745	−0.3986	−0.3302	1,108,000	1.443×10^{-6}
$\Gamma_{BP,WT}$	0.01912	0.02982	0.01192	0.006443	0.05315	1,108,000	-1.05×10^{-5}
$\Gamma_{EM,WT}$	0.3393	0.3504	0.01058	0.3296	0.371	1,108,000	-4.922×10^{-6}
$\Gamma_{DC,WT}$	−0.3316	−0.01978	0.02177	−0.06241	0.02297	1,108,000	-3.381×10^{-7}
$\Gamma_{AW,WT}$	0.05112	0.06127	0.01916	0.02362	0.09885	1,108,000	3.579×10^{-6}
$\Gamma_{EM,BP}$	0.1819	0.1955	0.01173	0.1724	0.2184	1,108,000	7.89×10^{-6}
$\Gamma_{DC,BP}$	0.4903	0.4984	0.01767	0.4631	0.5323	1,108,000	-3.719×10^{-7}
$\Gamma_{AW,BP}$	−0.1635	−0.1619	0.01909	−0.1991	−0.1243	1,108,000	-7.963×10^{-7}
$\Gamma_{DC,EM}$	0.1751	0.1751	0.02139	0.1328	0.2167	1,108,000	-4.048×10^{-7}
$\Gamma_{AW,EM}$	0.4594	0.4516	0.01623	0.4193	0.4829	1,108,000	3.382×10^{-6}
$\Gamma_{AW,DC}$	0.4832	0.4853	0.01679	0.452	0.5178	1,108,000	2.445×10^{-6}

Table 3.3. Posterior summaries of $\boldsymbol{\mu}$ and the hierarchical parameters.

Parameter	Truth	Mean	St.Dev.	2.5%	97.5%	ESS	$\hat{r} - 1$
μ_{A1c}	-4.655	-4.635	0.1017	-4.831	-4.431	325,522	3.683×10^{-5}
μ_{GFR}	-15.13	-15.16	0.1133	-15.38	-14.93	334,437	2.724×10^{-5}
μ_{WT}	-0.2571	-0.2814	0.1131	-0.5104	-0.06512	347,798	2.853×10^{-5}
μ_{BP}	0.08172	0.05801	0.1004	-0.1409	0.2537	367,781	2.182×10^{-5}
μ_{EM}	-3.552	-3.591	0.1198	-3.819	-3.345	258,885	4.867×10^{-5}
μ_{DC}	11.04	11.1	0.1142	10.88	11.33	285,526	5.39×10^{-5}
μ_{AW}	-3.553	-3.51	0.1309	-3.76	-3.244	299,935	4.345×10^{-5}
ω_1	-0.05	-0.08448	0.1182	-0.3212	0.1417	125,597	1.548×10^{-4}
ω_2	0.05	0.06936	0.118	-0.1688	0.2935	185,557	8.634×10^{-5}
$\zeta_{BD,1}$	-0.06	-0.07751	0.1071	-0.2921	0.1302	340,041	2.842×10^{-5}
$\zeta_{BD,2}$	0.06	0.05247	0.1107	-0.1729	0.2624	238,899	4.971×10^{-5}
$\zeta_{VT,1}$	-0.05	-0.07849	0.1057	-0.2851	0.1309	336,639	2.641×10^{-5}
$\zeta_{VT,2}$	0.05	0.1036	0.1074	-0.1041	0.3196	342,217	2.386×10^{-5}
$\zeta_{QoL,1}$	-0.04	-0.0979	0.1175	-0.3401	0.1254	275,599	4.589×10^{-5}
$\zeta_{QoL,2}$	0.04	0.05215	0.1195	-0.1983	0.2752	252,260	5.729×10^{-5}
$\beta_{A1c,1}$	-0.0625	-0.06537	0.1062	-0.2754	0.1433	351,198	3.634×10^{-5}
$\beta_{A1c,2}$	0.0625	0.04059	0.1097	-0.1844	0.248	321,135	3.004×10^{-5}
$\beta_{GFR,1}$	-0.0575	-0.08428	0.1121	-0.3125	0.1301	350,092	2.412×10^{-5}
$\beta_{GFR,2}$	0.0575	0.05709	0.1139	-0.1728	0.2772	312,271	2.891×10^{-5}
$\beta_{WT,1}$	-0.0525	-0.06605	0.1127	-0.28	0.1645	353,322	3.393×10^{-5}
$\beta_{WT,2}$	0.0525	0.1129	0.1126	-0.1013	0.343	357,755	2.736×10^{-5}
$\beta_{BP,1}$	-0.0475	-0.09184	0.1042	-0.2985	0.1113	367,771	2.559×10^{-5}
$\beta_{BP,2}$	0.0475	0.1146	0.1046	-0.08622	0.3252	368,881	2.24×10^{-5}
$\beta_{EM,1}$	-0.0425	-0.1041	0.1227	-0.3564	0.1298	278,860	4.352×10^{-5}
$\beta_{EM,2}$	0.0425	0.05202	0.1241	-0.2059	0.2858	268,833	4.732×10^{-5}
$\beta_{DC,1}$	-0.04	-0.1085	0.1198	-0.3549	0.1183	287,734	5.071×10^{-5}
$\beta_{DC,2}$	0.04	0.05705	0.1204	-0.1913	0.2848	277,710	5.756×10^{-5}
$\beta_{AW,1}$	-0.0375	-0.08531	0.1243	-0.3376	0.1561	306,691	4.025×10^{-5}
$\beta_{AW,2}$	0.0375	0.03425	0.1294	-0.2449	0.2687	244,493	5.323×10^{-5}
δ_β	0.68	1.074	0.3962	0.2899	1.845	1,108,000	-4.244×10^{-6}

3.7.1 Analyzing a Simulated Data Set

We fit a dataset with 100 patients each in the standard of care and treatment groups using the NUTS algorithm, defining the model using `ProbabilityModels.jl` and running NUTS with `DynamicHMC.jl`. We sampled 1,750 warmup iterations and 60,000 posterior samples from 18 chains, resulting in 1,080,000 total posterior samples. While more samples than necessary for accurate estimates, sampling finished in under an hour using an i9 7980XE CPU, and large numbers of samples and chains improve our ability to detect differences between chains that indicate convergence issues.

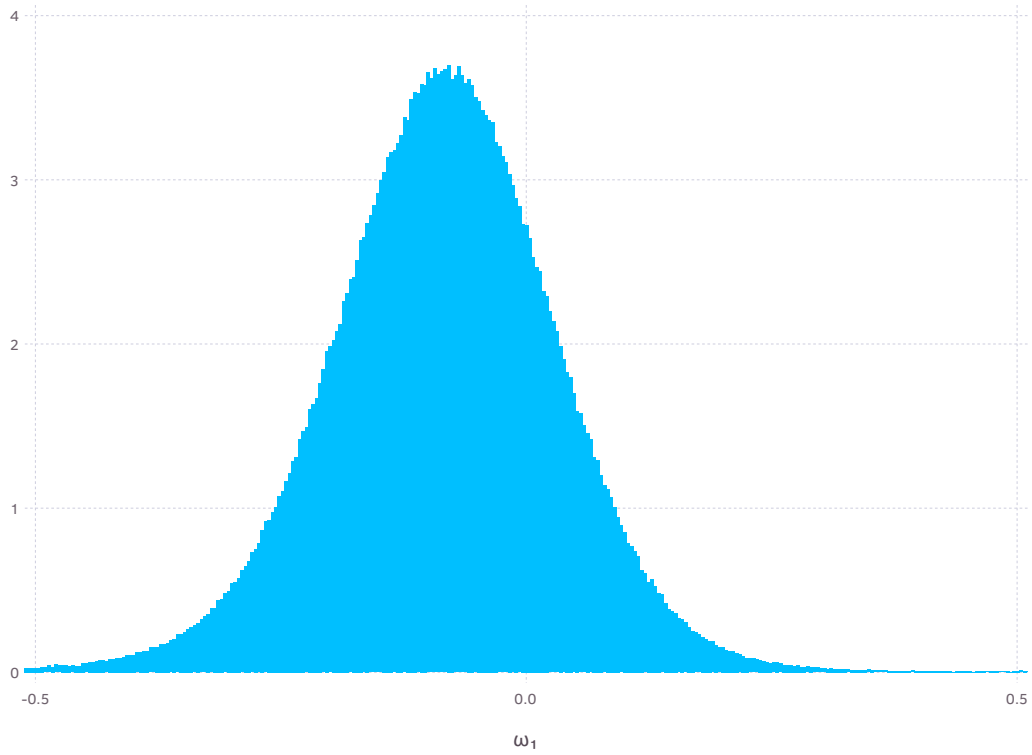

Figure 3.4. Histogram of posterior samples of ω_1, using 500 bins to highlight sampling variability.

Posterior summary statistics for $\boldsymbol{\kappa}$, $\boldsymbol{\rho}$, $\boldsymbol{\sigma}$, and $\boldsymbol{\Gamma}$ are shown in Table 3.2, while $\boldsymbol{\mu}$ and the parameters belonging to the $\boldsymbol{\beta}_g$ hierarchy are shown in Table 3.3. The 95% equal tailed credible intervals for each of the parameters encompass the true values, although many intervals, particularly those of the $\boldsymbol{\beta}_g$ hierarchy, are broad relative to the absolute value of the parameters. All intervals within the $\boldsymbol{\beta}_g$ hierarchy include 0, however the 95% equal tailed interval of δ_β does not. Additionally, the hierarchical parameters feature much lower effective sample sizes (ESS), while the ESS of most other parameters matched the number of posterior samples.

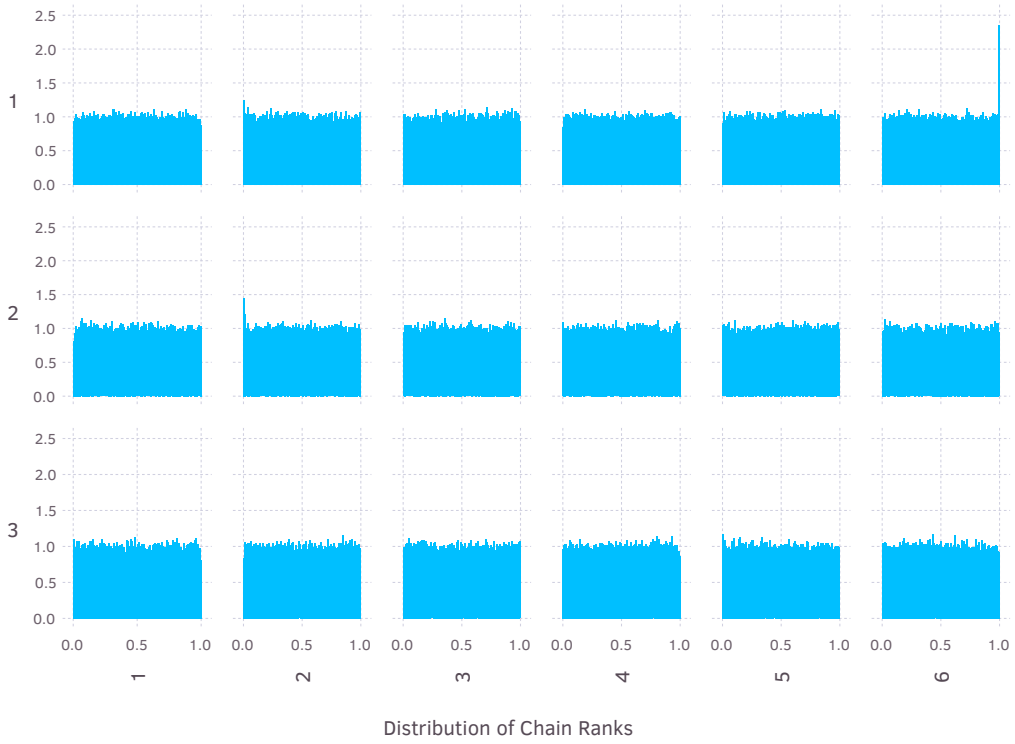

Figure 3.5. Chain rank plots of posterior samples for ω_1.

Of all parameters, ω_1 had the lowest effective sample size and highest Gelman-Rubin statistic $\hat{r}$ (Gelman et al., 1992). Trace plots can be difficult to assess when the number of samples and chains are large. It is likely that subtle biases in chains will go unnoticed. For these reasons, Simpson (2019) recommend rank plots. While the histogram looks smooth (Figure 3.4), the rank plots show a couple of abnormalities (Figure 3.5). We calculated the rank of each sample, divided the ranks by the number of samples, and then plotted histograms of the ranks corresponding to each chain. If each chain converged to the same posterior, all histograms should show a Uniform(0,1) distribution. However, the chain in position (1,6) shows a large spike at the highest rank, while the chain is position (2,2) shows a smaller spike at the bottom of the distribution. The longest streaks of successive rejections in these chains were 36 and 23. The remaining chains had a longest streak of 9.75 on average, with chain (2,1) featuring a streak of 24. This difference is not by itself able to account for these spikes. Further investigation will be necessary to elucidate the correct

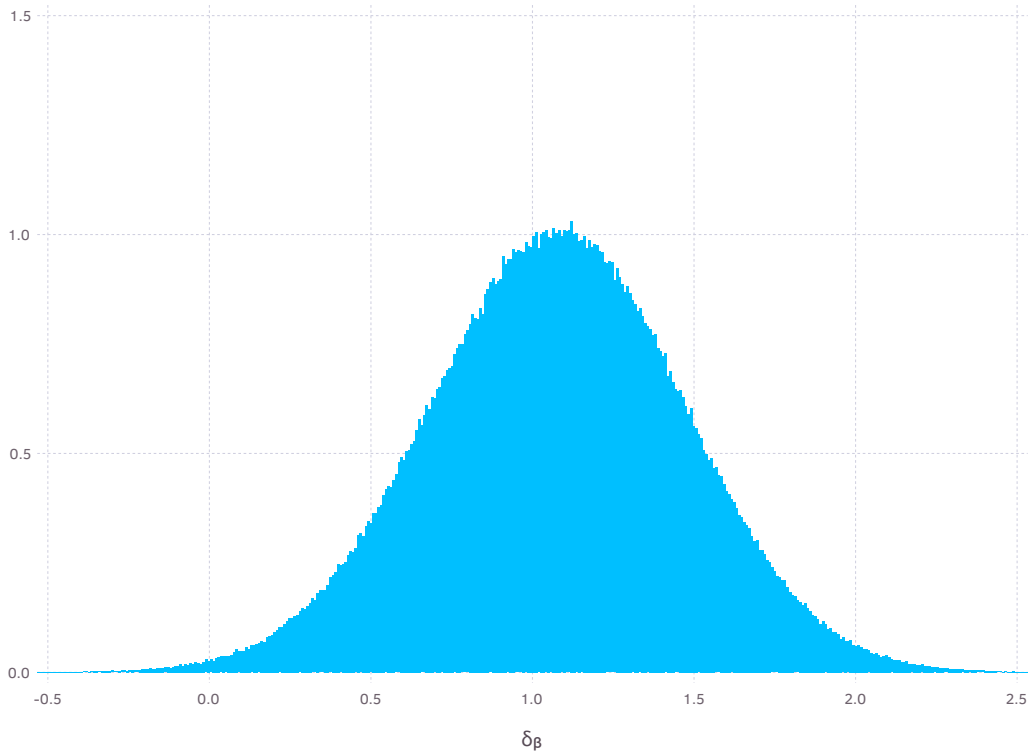

Figure 3.6. Histogram of posterior samples of δ_β, using 500 bins to highlight sampling variability.

thickness of the upper and lower tails. While $\hat{r}$ was close to 1, traditional $\hat{r}$ is insensitive to differences in the second moment (Simpson, 2019). Increasing the target acceptance probability from 0.95 to 0.99 may help.

Focusing on the distribution of δ_β, the posterior of δ_β is smooth and bell shaped (Figure 3.6). In Table 3.3 we see that there is a 97.5% probability that δ_β exceeeds 0.2899. In fact, $\Pr(\delta_\beta > 0.418) = 0.95$ and $\Pr(\delta_\beta > 0) = 0.996$. Therefore, although the effect parameter intervals overlapped heavily, the evidence provides strong support for our decision requring merely $\Pr(\delta_\beta > 0) > 0.95$.

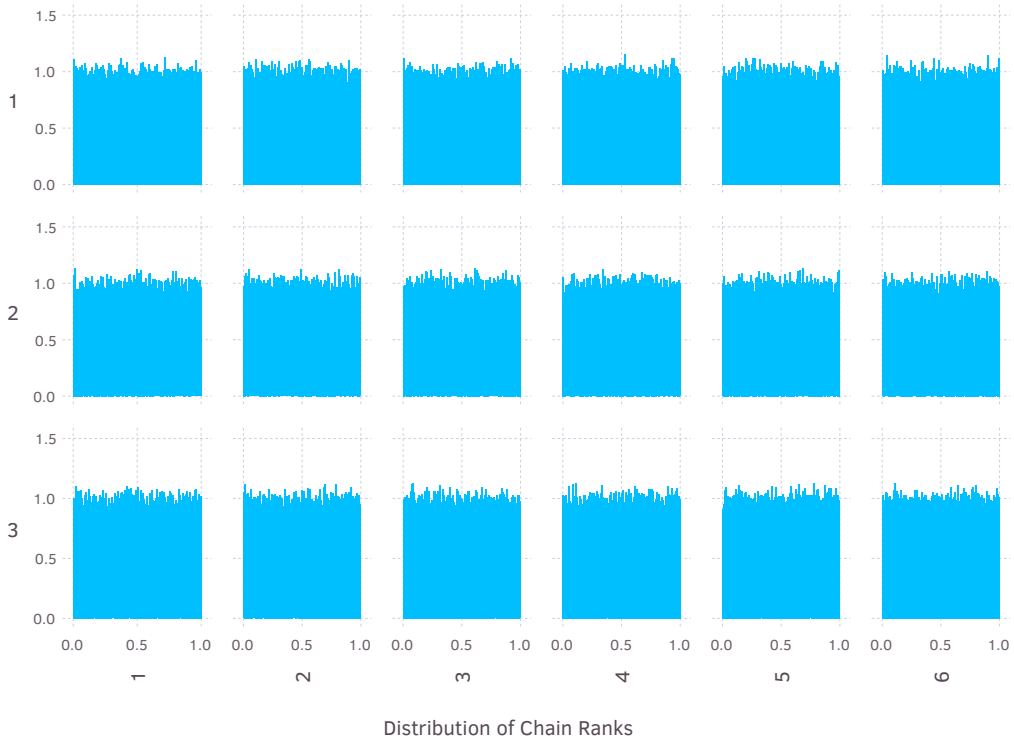

Figure 3.7. Chain rank plots of posterior samples for δ_β.

Additionally, δ_β's chain-rank plots do not appear to show any of the convergence issues oberved in ω_1 (Figure 3.7), and the effective sample size matched the number of posterior samples.

Table 3.4. Summary statistics of the simulation. $Q_{\delta_\beta}(N)$ refers to the N^{th} percentile of the posterior of δ_β when fit on virtual data sets. Columns indicate sample size, percentage of simulated intervals greater than 0, percentage of simulated 90% equal tailed credible intervals containing the true value, and mean interval width.

Sample Size	$\Pr\left(Q_{\delta_\beta}(5) > 0\right)$	$\Pr\left(Q_{\delta_\beta}(5) < 0.7 < Q_{\delta_\beta}(95)\right)$	mean $\left(Q_{\delta_\beta}(95) - Q_{\delta_\beta}(5)\right)$
50	0.335	0.933	2.10
100	0.53	0.929	1.46
200	0.745	0.939	1.02
400	0.915	0.923	0.726

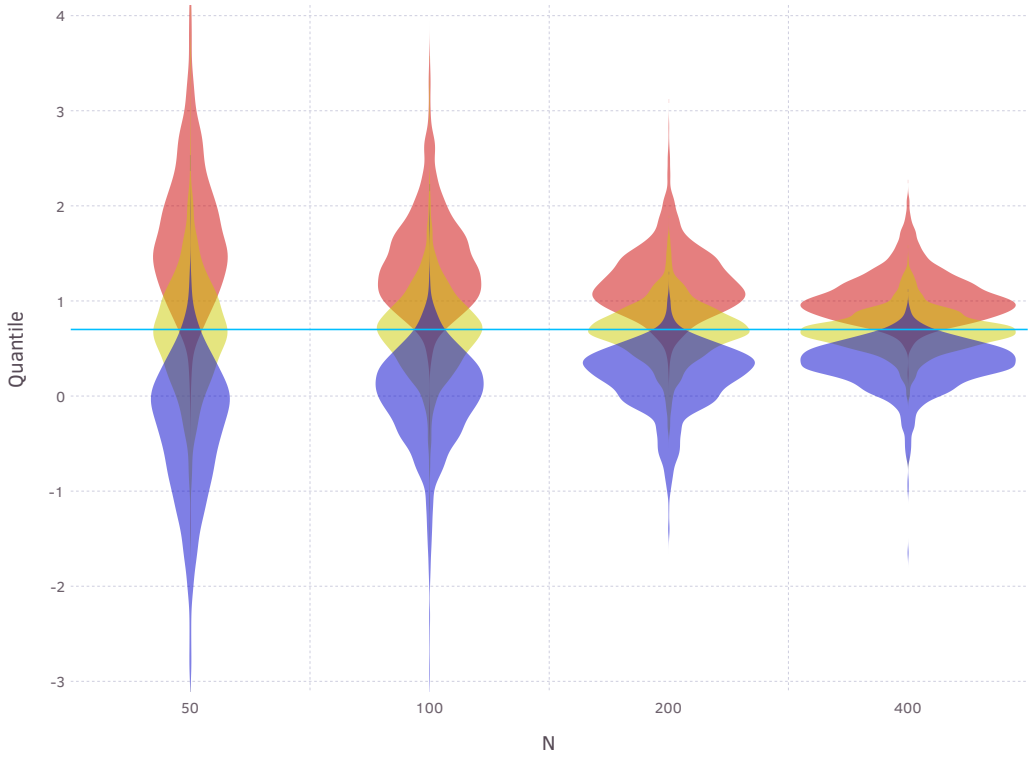

Figure 3.8. Violin plots displaying the distributions of the 5^{th}, 50^{th}, and 95^{th} quantiles of the posterior of δ_β from fits on simulated datasets.

3.8 Analyzing Simulated Data Sets

We generated 1,000 datasets for each of $N = 50, 100, 200, 400$ following the same procedure as in Section 3.6. For each sample size, half of the patients were assigned to standard of care, and half to treatment groups. For each dataset, we fit two chains with 900 warmup iterations followed by 2000 samples used for our analysis. Effective sample size and $\hat{r}$ were checked for δ_β. If the effective sample size was smaller than 200, or $\hat{r} > 1.05$, the dataset was reanalyzed.

Figure 3.8 contains violin plots of the distribution of posterior medians and equal tailed 90% credible intervals in the posteriors for δ_β, confirming that they are centered on the true value, and that the bounds tighten as sample size increases. As over 90% of these intervals contained the true values, the posterior intervals appear slighlty conservative (Table 3.4). However, as expected, the interval widths are roughly $\frac{C}{\sqrt{N}}$ where C is a constant,

so that quadrupling the sample size halves the interval width. As the posteriors concentrate more closely around the true value, the proportion of virtual data sets that allow us to meet our criteria $\Pr(\delta_\beta > 0) > 0.95$ increased from 0.357 with $N = 50$ to 0.907 with $N = 400$.

BIBLIOGRAPHY

Al-Mouhamed, M., Fatayer, A., Mohammad, N., et al. (2016). Optimizing the matrix multiplication using strassen and winograd algorithms with limited recursions on many-core. *International Journal of Parallel Programming*, 44(4):801–830.

Alfano, S. (2005). A numerical implementation of spherical object collision probability. *Journal of Astronautical Sciences*, 53(1):103–109.

Bates, D. (2011). Quadratic forms of random variables. *STAT*, 849.

Betancourt, M. (2017). A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*.

Boyer, J. G. and Earp, J. A. L. (1997). The development of an instrument for assessing the quality of life of people with diabetes: Diabetes-39. *Medical care*, pages 440–453.

Brent, R. P. (2013). *Algorithms for minimization without derivatives*. Courier Corporation.

Brod, M., Stewart, A. L., Sands, L., and Walton, P. (1999). Conceptualization and measurement of quality of life in dementia: the dementia quality of life instrument (dqol). *The Gerontologist*, 39(1):25–36.

Brutti, P., De Santis, F., and Gubbiotti, S. (2008). Robust bayesian sample size determination in clinical trials. *Statistics in Medicine*, 27(13):2290–2306.

Chan, F. K. (2008). *Spacecraft collision probability*. Aerospace Press El Segundo, CA.

Chen, L. M., Ibrahim, J. G., and Chu, H. (2011). Sample size and power determination in joint modeling of longitudinal and survival data. *Statistics in medicine*, 30(18):2295–2309.

Cheng, D., Stamey, J. D., and Branscum, A. J. (2009). Bayesian approach to average power calculations for binary regression models with misclassified outcomes. *Statistics in medicine*, 28(5):848–863.

Clarke, S. J. and Jones, S. A. (2015). Bayesian estimation for diagnostic testing of biosecurity risk material in the absence of a gold standard when test data are incomplete. *Journal of agricultural, biological, and environmental statistics*, 20(3):389–408.

Dendukuri, N. and Joseph, L. (2001). Bayesian approaches to modeling the conditional dependence between multiple diagnostic tests. *Biometrics*, 57(1):158–167.

DiMasi, J. A., Grabowski, H. G., and Hansen, R. W. (2016). Innovation in the pharmaceutical industry: new estimates of r&d costs. *Journal of health economics*, 47:20–33.

Fieker, C., Hart, W., Hofmann, T., and Johansson, F. (2017). Nemo/hecke: Computer algebra and number theory packages for the julia programming language. In *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation*, ISSAC '17, pages 157–164, New York, NY, USA. ACM.

Foster, J. and Estes, H. S. (1992). A parametric analysis of orbital debris collision probability and maneuver rate for space vehicles. *NASA JSC*, 25898.

Fu, H. and Manner, D. (2010). Bayesian adaptive dose-finding studies with delayed responses. *Journal of biopharmaceutical statistics*, 20(5):1055–1070.

Gelman, A., Rubin, D. B., et al. (1992). Inference from iterative simulation using multiple sequences. *Statistical science*, 7(4):457–472.

Geoga, C. J., Anitescu, M., and Stein, M. L. (2018). Scalable gaussian process computations using hierarchical matrices. *arXiv preprint arXiv:1808.03215*.

Gustafson, P. et al. (2014). Bayesian inference in partially identified models: Is the shape of the posterior distribution useful? *Electronic Journal of Statistics*, 8(1):476–496.

Guyatt, G. H., Bombardier, C., and Tugwell, P. X. (1986). Measuring disease-specific quality of life in clinical trials. *CMAJ: Canadian Medical Association Journal*, 134(8):889.

Hall, D. T., Casali, S. J., Johnson, L. C., Skrehart, B. B., and Baars, L. G. (2018). High fidelity collision probabilities estimated using brute force monte carlo simulations.

Hall, D. T., Hejduk, M. D., and Johnson, L. C. (2017). Time dependence of collision probabilities during satellite conjunctions.

Hanson, T., Johnson, W. O., and Gardner, I. A. (2003). Hierarchical models for estimating herd prevalence and test accuracy in the absence of a gold standard. *Journal of agricultural, biological, and environmental statistics*, 8(2):223.

Hejduk, M. (2008). Space catalogue accuracy modeling simplifications. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 6773.

Hejduk, M., Casali, S., Cappellucci, D., Ericson, N., and Snow, D. (2013). A catalogue-wide implementation of general perturbations orbit determination extrapolated from higher order orbital theory solutions. In *Spaceflight Mechanics Conference, Kauai, HI, USA*.

Hejduk, M. D. (2019). Satellite conjunction assessment risk analysis for "dilution region" events: Issues and operational approaches.

Hejduk, M. D. and Johnson, L. C. (2016). Approaches to evaluating probability of collision uncertainty.

Holzmann, J., Balser, S., and Windisch, J. (2016). Totality of the evidence at work: the first us biosimilar.

Hotelling, H. (1953). New light on the correlation coefficient and its transforms. *Journal of the Royal Statistical Society. Series B (Methodological)*, 15(2):193–232.

Hui, S. L. and Walter, S. D. (1980). Estimating the error rates of diagnostic tests. *Biometrics*, pages 167–171.

Johnson, S. G. (2019). The nlopt nonlinear-optimization package.

Johnson, W. O., Gastwirth, J. L., and Pearson, L. M. (2001). Screening without a "gold standard": the hui-walter paradigm revisited. *American Journal of Epidemiology*, 153(9):921–924.

Joseph, L., Du Berger, R., and Bélisle, P. (1997). Bayesian and mixed bayesian/likelihood criteria for sample size determination. *Statistics in medicine*, 16(7):769–781.

Joseph, L., Gyorkos, T. W., and Coupal, L. (1995). Bayesian estimation of disease prevalence and the parameters of diagnostic tests in the absence of a gold standard. *American journal of epidemiology*, 141(3):263–272.

Kressner, D. (2015). Matrices with hierarchical low-rank structures, part ii. Technical report.

Krumholz, H. M. (2012). Open science and data sharing in clinical research: basing informed decisions on the totality of the evidence.

Laurens, S., Seimandi, P., Couetdic, J., and Dolado, J. C. (2017). Covariance matrix uncertainty analysis and correction. In *68th International Astronautical Congress*.

Lewandowski, D., Kurowicka, D., and Joe, H. (2009). Generating random correlation matrices based on vines and extended onion method. *Journal of Multivariate Analysis*, 100(9):1989–2001.

Lewis, S. J., Ebrahim, S., and Smith, G. D. (2005). Meta-analysis of mthfr 677c t polymorphism and coronary heart disease: does totality of evidence support causal role for homocysteine and preventive potential of folate? *Bmj*, 331(7524):1053.

Mason, J. C. and Handscomb, D. C. (2002). *Chebyshev polynomials*. Chapman and Hall/CRC.

Mogensen, P. K. and Riseth, A. N. (2018). Optim: A mathematical optimization package for Julia. *Journal of Open Source Software*, 3(24):615.

Monnahan, C. C., Thorson, J. T., and Branch, T. A. (2017). Faster estimation of bayesian models in ecology using hamiltonian monte carlo. *Methods in Ecology and Evolution*, 8(3):339–348.

Mühlbacher, A. C. and Kaczynski, A. (2016). Making good decisions in healthcare with multi-criteria decision analysis: the use, current research and future development of mcda. *Applied health economics and health policy*, 14(1):29–40.

Newman, L. K., Frigm, R. C., Duncan, M. G., and Hejduk, M. D. (2014). Evolution and implementation of the nasa robotic conjunction assessment risk analysis concept of operations.

Nyström, E. J. (1930). Über die praktische auflösung von integralgleichungen mit anwendungen auf randwertaufgaben. *Acta Mathematica*, 54(1):185–204.

Patera, R. P. (2001). General method for calculating satellite collision probability. *Journal of Guidance, Control, and Dynamics*, 24(4):716–722.

Rahme, E., Joseph, L., and Gyorkos, T. W. (2000). Bayesian sample size determination for estimating binomial parameters from data subject to misclassification. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 49(1):119–128.

Revels, J., Lubin, M., and Papamarkou, T. (2016). Forward-mode automatic differentiation in julia. *arXiv:1607.07892 [cs.MS]*.

Sabol, C., Sukut, T., Hill, K., Alfriend, K. T., Wright, B., Li, Y., and Schumacher, P. (2010). Linearized orbit covariance generation and propagation analysis via simple monte carlo simulations. In *Paper AAS 10-134 presented at the AAS/AIAA Space Flight Mechanics Conference, February*, pages 14–17.

Scheffer, J. (2002). Dealing with missing data. *Research Letters in the Information and Mathematical Sciences*, 3(1):153–160.

Simpson, D. (2019). Maybe it's time to let the old ways die; or we broke r-hat so now we have to fix it.

Smith, A. F. and Roberts, G. O. (1993). Bayesian computation via the gibbs sampler and related markov chain monte carlo methods. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55(1):3–23.

Stamey, J. and Gerlach, R. (2007). Bayesian sample size determination for case-control studies with misclassification. *Computational statistics & data analysis*, 51(6):2982–2992.

Stamey, J. D., Seaman, J. W., and Young, D. M. (2005). Bayesian sample-size determination for inference on two binomial populations with no gold standard classifier. *Statistics in medicine*, 24(19):2963–2976.

Strand, V., Girolomoni, G., Schiestl, M., Ernst Mayer, R., Friccius-Quecke, H., and McCamish, M. (2017). The totality-of-the-evidence approach to the development and assessment of gp2015, a proposed etanercept biosimilar. *Current medical research and opinion*, 33(6):993–1003.

Team, S. D. (2018). *Stan Modeling Language Users Guide and Reference Manual*, version 2.18.0 edition.

Thorning, T. K., Raben, A., Tholstrup, T., Soedamah-Muthu, S. S., Givens, I., and Astrup, A. (2016). Milk and dairy products: good or bad for human health? an assessment of the totality of scientific evidence. *Food & nutrition research*, 60(1):32527.

Uhlenbeck, G. E. and Ornstein, L. S. (1930). On the theory of the brownian motion. *Physical review*, 36(5):823.

Ventura, L. and Reid, N. (2014). Approximate bayesian computation with modified log-likelihood ratios. *Metron*, 72(2):231–245.

Viroli, C. (2012). On matrix-variate regression analysis. *Journal of Multivariate Analysis*, 111:296–309.

Wang, F., Gelfand, A. E., et al. (2002). A simulation-based approach to bayesian sample size determination for performance under a given model and for separating models. *Statistical Science*, 17(2):193–208.

Wang, Z., Dendukuri, N., Pai, M., and Joseph, L. (2017). Taking costs and diagnostic test accuracy into account when designing prevalence studies: An application to childhood tuberculosis prevalence. *Medical Decision Making*, 37(8):922–929.

Yildirim, A., Akinci, F., Gozu, H., Sargin, H., Orbay, E., and Sargin, M. (2007). Translation, cultural adaptation, cross-validation of the turkish diabetes quality-of-life (dqol) measure. *Quality of Life Research*, 16(5):873–879.

Zaidi, W. and Hejduk, M. D. (2016). Earth observing system covariance realism. In *AIAA/AAS Astrodynamics Specialist Conference*, page 5628.