Die approbierte Originalversion dieser Dissertation ist in der Hauptbibliothek der Technischen Universität Wien aufgestellt und zugänglich.
http://www.ub.tuwien.ac.at

The approved original version of this thesis is available at the main library of the Vienna University of Technology.
http://www.ub.tuwien.ac.at/eng

TECHNISCHE
UNIVERSITÄT
WIEN

VIENNA
UNIVERSITY OF
TECHNOLOGY

DISSERTATION

Probabilistic modeling of high-dimensional and high-throughput biological data

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Doktors der technischen Wissenschaften unter der Anleitung von

Ao. Univ.-Prof. Dipl.-Ing. Dr.techn. Peter Filzmoser

eingereicht an der Technischen Universität Wien
bei der Fakultät für Mathematik und Geoinformation

von

Brian Godsey
Matrikelnummer: 0641175
1432 Marshall St
Baltimore 21230, Maryland, USA

Wien, im Oktober 2013

Abstract

Probabilistic modeling is a type of statistical analysis that focuses on the inherent randomness of natural systems, and which avoids taking any summary value—such as an expected value—prior to the final results, and even then the results are often listed alongside a measure of probability or certainty. Clearly, all statistical methods involve some measure of certainty, but probabilistic modeling emphasizes the uncertainty of all quantities, including intermediate values and data points.

High-dimensional data consist of relatively few measurements of a large number of quantities, and tools which measure thousands of quantities simultaneously, resulting in high-dimensional data, are called "high-throughput"; a popular example in bioinformatics is a microarray experiment, which may comprise fewer than twenty measurements of each of thousands of genes. This often creates under-determined systems, depending on the statistical model chosen, in which there could be many possible parameter solutions that are able to produce the same result. There are ways to address this under-determinedness, among which probabilistic modeling has clear advantages, both theoretically and practically.

Analysis of high-dimensional data can benefit from probabilistic modeling primarily because the consideration of inherent randomness allows a probabilistic model to consider not only the existence of one or many parameter solutions to the system, but also consider the probability or likelihood of a particular solution. For instance, if a data set contains replicates, the reproducibility (a type of inherent randomness) can be exploited to prefer the most reproducible good solution. The biological sciences contain many such situations, in which we have relatively few measurements of many quantities, and thus provide the opportunity to demonstrate the usefulness of probabilistic modeling, while solving some important biological problems.

This work presents probabilistic modeling as an approach to data-oriented scientific problems, including both knowledge-based model design as well as parameter inference via two popular inference algorithms: variational Bayesian learning and Markov Chain Monte Carlo (MCMC) sampling. The probabilistic framework is then applied to three problem classes in the biological sciences: gene-gene interaction, microRNA-gene targeting, and the prediction and comparison of athletic performances. In all three cases, the inference and/or prediction proved valuable in both understanding the underlying system as well as indicating likely candidates for further study—perhaps via more sensitive individual experiments.

Overall, this work presents the methods and illustrative applications that detail the concept and process of probabilistic modeling of high-dimensional data, allowing interested researchers to follow similar steps in their own work to create and derive insight from probabilistic models of biological systems.

Kurzfassung

Probabilistic Modeling ist einer Form der statistischen Analyse, die auf die inherenten Zufälligkeit von natürlichen Systemen fokussiert ist, und quantitiative Zusammenfassungen—zum Beispiel Mittelwertbildung—vermeidet, bis zum letzten Schritt der Analyse, und selbst dann bleibt eine Unsicherheit oder Wahrscheinlichkeit. Natürlich spielt Wahrscheinlichkeit eine Rolle in jeder statistischen Analyse, aber Probabilistic Modeling betont, dass wir immer eine Menge Unsicherheit haben, mit jedem Wert.

High-dimensional Data (hoch-dimensionale Daten) bestehen aus relativ wenige Datenpunkten für eine große Anzahl von Variablen. Messinstrumente, die tausende Werte gleichzeitig messen können und dadurch hoch-dimensionale Daten produzieren, wurden dann *high-throughput* genannt. In der Bioinformatik, ein Beispiel dafür wäre ein Microarray Experiment, bei dem man typischerweise weniger als zwanzig Messungen durchführt, aber jede Messung besteht aus tausenden Genen. Die nachfolgende Analyse ist also oft unterbestimmt, abhängig von den verwendeten Methoden, worin es viele mögliche Losungen für die Systeme gibt, die alle die gleiche Ergebnisse oder Daten liefern können. Probabilistic Modeling hat verschiedene theoretische und praktische Vorteile, die die folgende Probleme zu überwinden helfen.

Die Analyse von hoch-dimensionalen Daten ist eine gute Anwendung für Probabilistic Modeling aufgrund seiner Berücksichtigung von Zufälligkeit und lässt ein Modell alle mögliche Losungen beachten und dann nach Wahrscheinlichkeit bewerten, um genauer zu sagen, welche Lösungen besser passen.

Diese Arbeit präsentiert Probabilistic Modeling als Betrachtungsweise für datenorientierte wissenschaftliche Probleme; diese inkludiert nämlich die Entwicklung von Modellen, die auf Fachwissen basiert sind, und Parameterinferenz durch bekannte Methoden: *variational Bayesian Learning* und *Markov Chain Monte Carlo (MCMC) Sampling*. Diese Methoden werden dann in drei biologische Problemklassen angewendet: Gene-Gene Interaktionen, microRNA-Gene Targeting, und das Prognostizieren und Vergleichen von Leistungen in Leichtathletik. In allen drei Fällen, erweist sich die probabilistische Inferenz oder Prognose als bedeutend, in Hinblick auf das Verstehen der natürlichen Systeme, von denen die Daten stammen, und auf das Identifizieren von Kandidaten für weitere Studien.

Im Allgemeinen beschreibt diese Arbeit probabilistische Methoden und demonstriert Anwendungen, die das Konzept und Ablauf des Probabilistic Modeling zeigen. Hoch-dimensionale Daten und Systeme sind ein wissenschaftlicher Bereich, für den diese Betrachtungsweise besonders geeignet ist. Das Ziel ist solche Methoden für alle interessierten Wissenschaftler erreichbar zu machen.

Acknowledgments

First and foremost, I would like to thank my adviser, Peter Filzmoser, for taking me on at such a late stage in my Ph.D. research. It was very kind of him to work with me despite my stubbornness concerning my research plan and thesis, as well as my absence from Austria for the vast majority of the time I have worked with him. He has helped me tremendously, and I cannot thank him enough.

Next, my sincerest appreciation belongs to my family, who endured my commentary and complaints from afar, but often still seemed as genuinely concerned about my research progress as I was myself. Specifically, I thank my parents for their patience, my brother and sister-in-law for their understanding, and my two little nephews, particularly the younger, for being born before I could earn the title of "Dr.", and emphasizing how very long it has been since I became a Ph.D. candidate.

Lastly, and most importantly, I am extremely grateful to all of my friends, in particular all of my wonderful colleagues who became my friends and helped me both scientifically and personally during the years we worked together, and afterward. They are, in no particular order: Thomas, Paweł, Germán, Anaïs, Alex G. and Alex P., all of the Andis (S., R., and P.), Nancy, and Eva.

Contents

1 Introduction **1**
1.1 Background 1
1.2 High-dimensional biological data 2
1.2.1 Challenges of modeling high-dimensional data 3
1.2.2 Probabilistic modeling 4
1.3 Bayesian inference 5
1.3.1 Prior distributions 7
1.3.2 Hyper-priors 8
1.3.3 Conjugate prior distributions 8
1.4 Variable inference algorithms 8
1.4.1 Variational Bayes 9
1.4.2 Markov chain Monte Carlo sampling 12
1.5 Applications 14
1.5.1 Gene interaction models 15
1.5.2 Models of miR targeting 17
1.5.3 miR targeting with clustering 19
1.5.4 Athletic performance 19
1.6 Outline of the rest of this thesis 21

2 Clustered gene regulatory networks **22**
2.1 Introduction 22
2.2 Methods 25
2.2.1 A simple example 25
2.2.2 Model 28
2.2.3 Inference 29
2.2.4 Data 29
2.3 Results 30
2.4 Discussion 32

3 miR targeting using paired samples **34**
3.1 Introduction 35
3.2 Methods 38
3.2.1 Data 40
3.2.2 Partial ordering 41
3.2.3 Pre-processing 42

3.2.4 Target prediction algorithms 42
3.2.5 Model 43
3.2.6 Fitting the model using variational Bayesian methods 46
3.3 Results 47
3.3.1 Interaction validation by *miRWalk* 48
3.3.2 KEGG pathway enrichment 49
3.3.3 Top candidate interactions 49
3.3.4 Inferred miR regulators in multiple myeloma development . 51
3.4 Discussion 52

4 miR targeting with clusters 54
4.1 Introduction 55
4.2 Methods 56
4.2.1 Model specifications 57
4.2.2 Fitting the model using variational Bayes methods 59
4.2.3 Target prediction algorithms 60
4.2.4 Simulations 60
4.2.5 Multiple myeloma data 61
4.3 Results 61
4.3.1 Simulations 61
4.3.2 Multiple myeloma data 62
4.4 Discussion 65

5 Comparing athletic performances 67
5.1 Introduction 68
5.2 Methods 70
5.2.1 Data 71
5.2.2 The model 72
5.2.3 Development of an empirical prior 73
5.2.4 Fitting the model to the data 74
5.2.5 Some meaningful statistics 75
5.2.6 Correlation with future performances 77
5.3 Results 78
5.3.1 Convergence 78
5.3.2 Predictions made prior to 2000 79
5.3.3 Predictions made prior to 2008 81
5.3.4 Comparison with IAAF scoring tables 82
5.3.5 Predictions for 2012 and beyond 84
5.4 Discussion 86

Bibliography 89

Chapter 1

Introduction

1.1 Background

Many great discoveries have been made in recent years due to the ever-increasing volume and quality of data sets. As we collect and analyze more data, if we are careful, we will continue to make small improvements in our knowledge of the world around us. Sometimes, new data from recently developed, increasingly accurate tools allows us to make a breakthrough discovery, advancing the field significantly. On the other hand, we cannot always rely on improved quantity and quality of data to be the basis of new discoveries. A good argument can be made that re-visiting old data with new methods of analysis is as important as generating newer and better data sets. This may be obvious to many researchers in mathematical and statistical methods, but is an important point to consider for the purposes of project planning and funding.

Entire sub-fields of biology have prospered in recent years based solely on the idea that quantitative methods need improvement: bioinformatics, computational biology, biostatistics, etc, to name a few. These fields have developed in parallel with—and sometimes because of—advances in biochemistry and bioengineering that have led to new tests and tools that can measure biological activity more accurately than ever before. This is great for the related fields, but to rely heavily on technological advances would be a mistake.

However, I do not intend simply to promote data analysis methods to the exclusion of biotechnological research. But, I do wish to contribute to the idea that—even while better technology and better, bigger data sets are leading to huge advances in science—improving analysis methods, particularly by developing methods specific to the problem at hand, can lead to significant advances in scientific knowledge. The sport of auto racing gives a relevant analogy: faster cars lead to ever-faster lap times, but no matter how fast the cars become, the skill of the driver determines whether or not the lap times reflect the improved capabilities of the car. Some might even say that the skill of the driver becomes more important as the cars become faster, and certainly the driver will be able to operate at higher speeds after more experience and familiarity behind the wheel.

Such is the case with the analysis of biological data; as laboratory instruments become more refined and more sensitive, the analysis of the resulting data requires more rigor, and benefits from methods that are suited specifically for the highly specialized data source. Hence, every data source, every experiment, and every quantitative, biological question that needs to be answered can benefit from a data analysis method designed specifically for itself. Only through familiarity, rigor, and experience can such methods be developed to their fullest potential.

Therefore, this work describes a process by which quantitative methods developed from a thorough consideration of the biological system at hand, as well as an acute awareness of the questions that the project intends to answer, can lead to conclusions that are more usable than those generated by more generic statistical methods.

The following sections begin to describe, in more detail, problems that arise in everyday analysis of biological data sets, including properties and limitations of the data, obstacles and difficulties inherent in some analysis tasks, and potential misunderstandings of both the initial experimental question as well as the final results. This work then gives some ways to address these problems, as well as a series of specific cases in which tailor-made analysis methods—in particular, using probabilistic modeling and latent variable inference—have outperformed existing, less specialized methods.

1.2 High-dimensional biological data

I use the term “high-dimensional data” to mean the class of data sets for which the number of parameters or variables we typically estimate is far greater than the number of data samples. For instance, a typical microarray experiment measures the expression of thousands of genes under ten or fewer experimental conditions; when such a large number of things are measured simultaneously, this is often called “high-throughput”. For these data sets, fitting most common models of gene activity or interaction would require estimating—at a minimum—one parameter per gene, and thus thousands of parameters must be estimated from ten or fewer data points for each gene. Even a simple analysis of variance (ANOVA) model whose purpose is to detect up- or down-regulation of genes under certain conditions requires a statistical test—and thus a parameter estimation—for each gene under consideration.

In addition to the dimensionality being far greater than the number of samples, in the biological sciences, the number of samples taken is usually also significantly smaller than the number of samples it is possible to take. For instance, in most microarray experiments, it is theoretically possible to take hundreds of samples throughout a time course experiment, but due to monetary and time costs, most experiments include no more than ten or twenty. Contrast this with constant electronic monitoring of a chemical process that might occur in larger-scale commercial applications; in this case, sensor data can be collected and analyzed arbitrarily often.

Within this paper, I address mainly this class of data sets that contain far fewer samples than presumed model parameters and, in addition, contain significantly fewer samples than it is possible, in theory, to collect. Both of these distinctions are important for various reasons, which I outline in the next section.

1.2.1 Challenges of modeling high-dimensional data

When data contain far fewer samples than measurable dimensions, most modeling tasks include the estimation of parameters in an under-determined system. To be more specific, a system is "under-determined" if the number of facts that we know about a system is fewer than the number of things we are trying to estimate. This notion is most commonly applied to linear systems, whenever the number of equations is fewer than the number of unknown quantities. A specific case of this is a linear gene interaction model, whereby we attempt to infer the interaction coefficients (in the style of a continuous Markov model) of thousands of genes based on a series of ten to twenty time points; assuming that most genes vary their expression over time, there must be a very large number of possible linear combinations of gene expression values that can give the same result. This would be an under-determined system.

Because there can be many possible solutions to an under-determined system, and because under-determined systems exist in many fields, including biology, much work has been done on developing methods that aid the selection of the best possible, most biologically relevant of those solutions. There is a large number of such methods—including minimum norm and maximum entropy preference among possible solutions (a good survey of these appears in Madych [1991])—and I do not propose to review them all here; however, the vast majority of these methods attempt to accomplish one of the following goals:

1. to reduce the dimensionality of the data
2. to constrain, regularize, or reduce the dimensionality of the model parameter space
3. to rank the set of possible solutions based on evidence in their favor

Examples of goal (1) include principal component analysis (PCA, Jolliffe [2005]), independent component analysis (ICA, Lee [1998]), or clustering of various types. Goal (2) might be accomplished by a regularization technique such as Tikhonov regularization (ridge regression), LASSO (least absolute shrinkage and selection operator, Tibshirani [1996]) regression, or some heuristic involving de-activation of less helpful parameters. Lastly, the ranking of solutions as in goal (3) could be accomplished by a statistical test and accompanying p-value or, as I demonstrate throughout this paper, the ranking of solutions can be accomplished via a probabilistic model and the parameter likelihoods that such a model implies, given the data.

Estimated parameter likelihood distributions can be used in statistical tests to give direct estimations of the significance of the parameter values themselves. Thus, we can rank the possible solutions of the system by the significance of their corresponding parameters. Specific methods for accomplishing this are given in the following sections, but for now suffice it to say that in the presence of data replication (i.e. some number of specific experimental data points have been repeated) an estimate of uncertainty can be included in the modeling task. Furthermore, this measure of uncertainty can propagate through the parameter inference task so that the significance of parameter estimates is affected and the corresponding solution set can be ranked according to the resulting estimated certainty.

1.2.2 Probabilistic modeling

A probabilistic model is a formalization of the relationships of random variables to one another in the form of probability distributions which may include conditional relationships between variables. Formally, a probabilistic model may be indistinguishable from a statistical model; however, I use the term “probabilistic model” to emphasize two ideals:

1. Parameters are random variables, not unknown fixed values.
2. A parameter estimate consists of an estimated probability distribution, and not a single value.

Admittedly, there is some redundancy in these ideals, and in fact they can be summarized in one phrase: every quantity, observed or inferred, is a random variable. However, separating the ideals as above has some practical use in the formulation of specific probabilistic models.

To clarify the above ideals, let us consider a case where we have a set of sample data points, and we would like to know whether the samples come from a distribution with mean significantly different from some value, for example zero. Let us assume that the samples are normally distributed. A (frequentist) statistician might remark that this is a perfect example of an application of Student’s t-test. In fact, this is what Student’s t-test was designed to do. William Sealy Gosset, forced to write his statistical articles under a pseudonym by his employer, Guinness brewery, developed the t-test with the explicit purpose of addressing small samples and the corresponding uncertainty in mean values [Wikipedia 2013].

Specifically, for small-sample data sets, it is beneficial to admit some uncertainty in the value of the sample mean and variance; this is the main scientific advancement of the development of the t-test, and is at least partially in agreement with ideal (1) above. A one-sample t-test assumes that the samples are normally distributed, with unknown mean as well as an unknown variance that follows a chi-squared distribution. This contrasts with a Z-test, whose test statistic is assumed to follow a normal distribution, and where the sample variance is assumed to be the value estimated directly from the sample. Thus, whereas the null hypothesis of a Z-test is that the samples follow a normal distribution with known mean and known, estimated variance, the null hypothesis of the t-test is that the samples come from a normal distribution with known mean and unknown variance. Because the t-test admits uncertainty in the variance, the corresponding t-distribution used in the test has heavier tails than the normal distribution of the Z-test. This admission of uncertainty is in agreement with probabilistic ideal (1) written above.

So, while a Z-test essentially admits no variance in parameter values (though uncertainty about the correct value is still present), and a t-test admits uncertainty in the sample variance, it is possible to go even further in this direction. The purpose of a one-sample t-test or Z-test is to consider two alternatives, the null hypothesis and an alternative hypothesis, but since the canonical alternative hypothesis is simply to reject the null hypothesis, very little need be assumed about what model might be a more appropriate model, if not the null hypothesis model. A probabilistic model typically does not ask, figuratively, “Does this model fit?” but instead presents some well-defined alternatives and asks, “Which alternative fits best?” Therefore, probabilistic modeling is usually best achieved with some Bayesian ideals in mind; more detail is presented in the following section.

1.3 Bayesian inference

Bayesian inference can be summed up succinctly with a single equation, Bayes’ Theorem:

$$P(\Theta \mid D) = \frac{P(D \mid \Theta)\, P(\Theta)}{P(D)} \tag{1.1}$$

which relates the probability densities of two quantities Θ and D. This theorem, first proposed by Thomas Bayes in the 18th century, is now often put to use in “Bayesian updates”, whereby some prior belief about Θ—denoted $P(\Theta)$—can be updated given new information resulting from the outcome D, which was not present before the prior belief about Θ was constructed. The updating of beliefs is often touted as the principal benefit of using Bayesian methods, since the calculation of updates need not consider all prior data, but only the latest data, which can save considerable time. However, even when we do not wish to update any beliefs, Bayes' Theorem can be very useful in determining or comparing the fit of models to data.

If, in equation (1.1) above, we let Θ be a set of model parameters, and we let D be the data collected from the system we wish to describe using the model, then Bayes' Theorem defines the concept of a *posterior probability*: our best estimate of the distribution of the parameters Θ given the data D.

This approach to estimating model parameters may seem similar to more traditional, frequentist methods of parameter estimation, most notably maximum likelihood estimation, which consider the optimization of a function related to $P(D \mid \Theta)$ as in equation (1.1) above, but there are a couple of important differences. First of all, Bayes' Theorem describes a probability distribution of the parameter values, not just an estimate of the best-fit values. Secondly, by utilizing a prior distribution of the model parameters in $P(\Theta)$, we consider the model parameters as random variables and not as unobserved fixed values, which is a perspective that in practice often seems closer to the truth. To illustrate, consider a task of inferring a day's rainfall based on data from several measuring devices; the random nature of raindrops falling and small changes in climate between the locations of the devices creates variances that may be better described by a probability distribution than a point estimate.

One disadvantage of Bayesian methods is that in order to calculate $P(\Theta \mid D)$, the posterior probability distribution of the model parameters given the data, we first need $P(\Theta)$, $P(D)$, and $P(D \mid \Theta)$. The evidence $P(D)$ does not depend on the model parameters Θ, so in most cases of estimating Θ we can ignore $P(D)$ as a constant value that does not affect the shape of the posterior probability distribution even if it does change the scale. The more complex of the other two required densities, $P(D \mid \Theta)$, which describes the probability of the data given the model parameters, is usually the more cumbersome to deal with during estimation. The prior probability density (often written simply "prior") $P(\Theta)$ of the model parameters can usually be selected in such a way as to simplify the calculations or to be non-informative, if desired.

The choices of specific functions for both $P(D \mid \Theta)$ and $P(\Theta)$ are very important. The choice of $P(D \mid \Theta)$ may be considered to be the model itself—as it is in frequentist approaches—and thus depends heavily on knowledge or beliefs about the system that created the data. In many cases, common distributions such as the normal distribution, binomial distribution, or gamma distribution can be justified, but in other cases these may not be appropriate. Likewise, the prior distribution $P(\Theta)$ can also significantly affect results. This is often cited in criticism of Bayesian methods, but in fact in most applications it is possible to create a prior that is "non-informative" or otherwise equivalent to a frequentist method that does not have an explicit prior distribution. The prior distribution is one way, along with other aspects of the model design, to use the knowledge that we have about the system we are measuring.

1.3.1 Prior distributions

As stated above, with respect to Bayes' Theorem in equation (1.1), the probability density $P(\Theta)$ represents the prior beliefs or knowledge about the model parameters, or latent variables, Θ. The goal, then, of Bayesian inference, is typically to update our beliefs about the distribution of Θ using the data D, represented by $P(\Theta \mid D)$. Sometimes these prior beliefs come from existing experiments or analyses, sometimes they are merely a guess at what constitutes a reasonable value, and sometimes we have literally no idea what the values of Θ might actually be. Each of these cases requires a different approach to construction of priors.

In the case where we have previous analyses giving hard evidence about the distribution of Θ, we can construct what is usually called a "strong prior". A strong prior is intended to impose previous knowledge about Θ into the current analyses; this can be very useful whenever the new data set D is too small or too incomplete to derive statistically significant results, and thus the strong prior can be used to regularize the possibly anomalous new data. If we have no hard evidence about Θ, but we do have educated guesses about the value of Θ, a weak prior might be used. For instance, if we suspect that a certain value is usually relatively close to zero, but we can't say for sure how much it varies, then we might choose a weakly informative normal prior with mean zero and a high variance. In this case, the prior is designed to penalize values of Θ that are vastly different from expectations, but largely treats all reasonable values the same. A non-informative prior, on the other hand, does not penalize any possible value of Θ; all values are treated the same. These priors often come in the form of "improper" priors, meaning that they are not, in fact, probability distributions at all, but often an analytical form that is the result of taking the limit of a probability distribution as one or more of its parameters approaches infinity or zero. For instance, a commonly-used non-informative prior is a "flat" prior, which assigns the same probability to all possible values. Though a constant value of 1 is commonly used, this value cannot represent a true probability density (its unbounded integral is infinite) but could be thought of as, say, the limit of a normal distribution as the variance approaches infinity, adjusted by an arbitrarily large scaling factor.

1.3.2 Hyper-priors

Clearly, in some cases, a prior distribution may include some set of parameter values, chosen as described above. Usually, these parameters of the prior distribution are referred to as "hyper-priors" since they are one level further abstracted from the data. It is the goal of selecting priors and hyper-priors that specific values chosen do not meaningfully affect the final results. In this sense, the actual values of the hyper-priors should not matter as long as they are within reasonable ranges, as determined by the context of the experiment itself. Therefore, every probabilistic modeling task using priors and hyper-priors should conduct a sensitivity analysis, in which, at the very least, the values of the hyper-priors should be varied throughout a reasonable range in order to make sure that nothing significantly changes.

1.3.3 Conjugate prior distributions

Sometimes, the prior $P(\Theta)$ and the model probability density $P(D \mid \Theta)$ can be chosen in such a way as to guarantee that the posterior density $P(\Theta \mid D)$ is of the same family of probability distributions as $P(\Theta)$, then $P(\Theta)$ and $P(D \mid \Theta)$ are considered conjugate. This often simplifies inference tasks, and can make the involved integrals analytically tractable. Notably, for a Gaussian model density $P(D \mid \Theta)$, choosing a Gaussian prior over the mean and a gamma prior over the precision (inverse variance) results in posterior distributions of the same forms. This is very useful particularly in variational Bayesian inference, described later in this paper.

1.4 Variable inference algorithms

Once a probabilistic model has been designed, from the priors and hyper-priors down to the distributions assumed for the data, the canonical probabilistic modeling task is to infer the values of some of the latent variables or parameters involved. Again referring to Bayes' Theorem in equation (1.1), with latent parameter variables Θ and data D, we would like to draw conclusions about some of those parameters given the data, and the posterior density $P(\Theta \mid D)$ may achieve this.

Depending on the assumed distributions in the model, $P(\Theta \mid D)$ may be calculated analytically, for example through appropriate integration techniques. However, this is often not possible even for relatively simple models; for more complex probabilistic models, it is almost never possible to evaluate $P(\Theta \mid D)$ analytically—a necessary integral is intractable—and thus some approximation must be made. This is one reason that many people shy away from using Bayesian methods: the lack of an analytic solution, together with the approximation of a complex probability distribution (instead a single, best point estimate), can lead to greatly increased computation times.

Hence, many methods of approximate Bayesian inference have been developed, each having strengths and weaknesses. There are two popular classes of such algorithms—among others—to which I will refer as "iterative" and "sampling". Iterative algorithms, such as variational Bayes (VB), create successively better approximations to the posterior via some approximating function, whereas sampling algorithms test the density of $P(D \mid \Theta)P(\Theta)$ using many points in the parameter space of Θ to approximate $P(\Theta \mid D)$. The main advantages of iterative methods often include guaranteed convergence and well-defined posterior distributions, but sampling methods are often a better choice when convergence is not expected to be a problem (e.g. for known unimodal posteriors) and in models that include analytically inconvenient distributions that could complicate iterative algorithms. Further information about these algorithms is given in the following sections.

1.4.1 Variational Bayes

Variational Bayesian methods are a class of iterative algorithms that generate an analytical approximation to the desired posterior probability such that the approximation provides a lower bound to the actual marginal likelihood of the data. That is, the result of variational Bayesian inference is a probability distribution $Q(\Theta)$ over the latent parameter variables Θ that

1. is of a simpler analytical form than $P(\Theta \mid D)$,
2. approximates $P(\Theta \mid D)$ as closely as possible, and
3. gives a marginal likelihood that is a lower bound for the true marginal likelihood.

In practice, it is enough to assume that Q is a factorable distribution density among the variables in Θ. That is,

$$Q(\Theta) = Q_1(\theta_1)Q_2(\theta_2)Q_3(\theta_3)\dots Q_n(\theta_n) \tag{1.2}$$

where the θ_i are disjoint groups of variables, and the entropy of each $Q_i(\theta_i)$

$$\mathcal{H}(Q_i) = \int_{-\infty}^{\infty} Q_i(\theta_i) \log Q_i(\theta_i) d\theta_i \tag{1.3}$$

is analytically tractable.

Then, the goal of inference is to find such a probability density $Q(\Theta)$ that is as similar to $P(\Theta \mid D)$ as possible. We measure the similarity of these two distributions using the Kullback-Leibler (KL) divergence [Kullback and Leibler 1951; Winn 2003], given by

$$\mathrm{KL}(Q \parallel P) = \int_X Q(x) \log \frac{Q(x)}{P(x)} dx \tag{1.4}$$

where "log" is the natural logarithm. $\mathrm{KL}(Q \parallel P)$ is always non-negative, where zero indicates that $P(x)$ and $Q(x)$ are identical, and a larger value indicates higher dissimilarity.

Thus, the task at hand is to find the factorable density $Q(\Theta)$ as in equation (1.2) that is as similar as possible—measured by the KL-divergence—to our desired posterior density $P(\Theta \mid D)$. That is, we wish to find $Q(\Theta)$ that minimizes

$$\mathrm{KL}(Q \parallel P) = \int_\Theta Q(\Theta) \log \frac{Q(\Theta)}{P(\Theta \mid D)} d\Theta \tag{1.5}$$

Since we do not know the function $P(\Theta \mid D)$ (that's the goal of these calculations), we use Bayes' Theorem from equation 1.1 to make the substitution and simplification

$$\begin{aligned}
\mathrm{KL}(Q \parallel P) &= \int_\Theta Q(\Theta) \log \frac{Q(\Theta)P(D)}{P(D \mid \Theta)P(\Theta)} d\Theta && (1.6)\\
&= \int_\Theta Q(\Theta) \log \frac{Q(\Theta)}{P(D, \Theta)} d\Theta + \log P(D) && (1.7)
\end{aligned}$$

The final term of equation (1.7) does not depend at all on the choice of $Q(\Theta)$, so we can ignore it in the minimization task. Therefore, I will borrow notation from Winn [2003] and define $\mathcal{L}(Q)$ to be the negative of the remaining term of equation (1.7), as in

$$\begin{aligned}
\mathcal{L}(Q) &= -\int_\Theta Q(\Theta) \log \frac{Q(\Theta)}{P(D, \Theta)} d\Theta && (1.8)\\
&= \int_\Theta Q(\Theta) \log P(D, \Theta) - Q(\Theta) \log Q(\Theta) d\Theta && (1.9)
\end{aligned}$$

which due to the negation is a quantity we wish to maximize. Now, since $Q(\Theta)$ is factorable as in equation (1.2),

$$\mathcal{L}(Q) = \int\limits_{\Theta} \prod_i \Big(Q_i(\theta_i)\Big) \log P(D,\Theta) d\Theta - \sum_i \Bigg(\int\limits_{\theta_i} Q_i(\theta_i) \log Q_i(\theta_i) d\theta_i \Bigg) \tag{1.10}$$

and we can separate out a single factor j as in

$$\begin{aligned} \mathcal{L}(Q) = &\int\limits_{\theta_j} Q_j(\theta_j) \Bigg(\int \cdots \int\limits_{\theta_{i \neq j}} \prod_{i \neq j} \Big(Q_i(\theta_i)\Big) \log P(D,\Theta) d\theta_{i \neq j} \Bigg) d\theta_j \\ &- \int\limits_{\theta_j} Q(\theta_j) \log Q(\theta_j) d\theta_j - \sum_{i \neq j} \Bigg(\int\limits_{\theta_i} Q_i(\theta_i) \log Q_i(\theta_i) d\theta_i \Bigg) \end{aligned} \tag{1.11}$$

Now, let us define the density of a distribution Q_j^* over the parameters in factor j:

$$Q_j^*(\theta_j) = \frac{1}{Z} \exp \Bigg(\int \cdots \int\limits_{\theta_{i \neq j}} \prod_{i \neq j} \Big(Q_i(\theta_i)\Big) \log P(D,\Theta) d\theta_{i \neq j} \Bigg) \tag{1.12}$$

where Z is the constant that makes this a valid probability distribution. Note that the exponent here appears in equation (1.11) and is the density of the joint distribution $P(D,\Theta)$ marginalized over all variables in Θ except for θ_j. More details on the derivation of Q_j^* can be found in Beal [2003]; here, I provide only confirmation that it is useful.

Letting the notation $\mathcal{H}(Q_j)$ indicate the entropy of Q_j as in equation (1.3), and using the definition of Q_j^* in equation (1.12), we can re-write equation (1.11) as

$$\mathcal{L}(Q) = \int\limits_{\theta_j} Q_j(\theta_j) \log Q_j^*(\theta_j) d\theta_j - \log Z - \int\limits_{\theta_j} Q(\theta_j) \log Q(\theta_j) d\theta_j - \sum_{i \neq j} \mathcal{H}(Q_i(\theta_i)) \tag{1.13}$$

$$= \int\limits_{\theta_j} \Bigg[- Q_j(\theta_j) \log \frac{Q(\theta_j)}{Q_j^*(\theta_j)} \Bigg] d\theta_j - \log Z - \sum_{i \neq j} \mathcal{H}(Q_i(\theta_i)) \tag{1.14}$$

$$= - \mathrm{KL}(Q_j \parallel Q_j^*) - \log Z - \sum_{i \neq j} \mathcal{H}(Q_i(\theta_i) \tag{1.15}$$

As mentioned previously, KL-divergence is non-negative, and it is zero if and only if the two distributions are identical. Also, the final two terms here have nothing to do with Q_j. Therefore, $\mathcal{L}(Q)$ is maximized with respect to Q_j when $Q_j = Q_j^*$, and a local optimum $Q(\Theta)$ can be found by iteratively updating each of the $Q_j(\theta_j)$.

The result of each update step is a factorable probability density whose KL-divergence with $P(\Theta \mid D)$ is less than the previous iteration, and the factors of each density are analytically tractable for many applications, often including hypothesis testing and other subsequent statistical inference tasks.

Variational Bayesian algorithms have a lot in common with the popular expectation-maximization (EM) algorithm. Both iteratively update individual (or groups of) parameters or latent variables according to a set of specified, interrelated probability distributions. The primary difference is that the distribution updates in EM consider only the expectations of the other distributions (e.g. when updating Q_j, we use only the expectation of Q_i and not its full density) whereas variational Bayesian updates consider the other variables' full distributions. This leads the two algorithms to have very similar results in some applications and very different results in others, depending very much on the variance and certainty present in the data.

More detail and examples of variational Bayesian methods and applications can be found in Beal [2003]; Winn [2003].

1.4.2 Markov chain Monte Carlo sampling

One of the most popular sampling algorithms for latent variable inference is called Markov chain Monte Carlo (MCMC) sampling; it comes in many specific forms, but all of them include some kind of "random walk" (or similar idea) around the space of possible latent variable values of Θ. At each point along the random walk, the probability density of interest, $P(\Theta \mid D)$, or a function proportional to it, is evaluated, and the densities—or relative densities—can be used to approximate the posterior distribution.

The primary concern when using any sampling method, including MCMC, is that a successful approximation requires that the set of samples cover roughly all of the areas with non-negligible probability density. If, for instance, we had a bi-modal distribution and we took samples only from the area around one of the modes but not the other (perhaps we were unaware of that mode), it would result in a poor approximation. But, a good set of samples that cover the sample space well directly results in a good approximation for the distribution. Hence, when approximating a distribution using sampling, most of the hard work and computational time is used for finding a good set of samples.

One of the most popular MCMC sampling algorithms is the Metropolis-Hastings algorithm [Hastings 1970]. In this algorithm, starting from an arbitrary point within the variable space, a *proposal* step is generated randomly from a *proposal distribution*, and then that step is either accepted or not based on the *acceptance distribution.* Such steps are proposed, possibly accepted, and repeated with the goal of traversing roughly the entire area of significant probability mass.

For example, a common proposal distribution is a Gaussian distribution with mean at the current parameter value. Let us say that $Q(\Theta \mid \Theta_0)$ is the Gaussian proposal distribution centered on the current sample point Θ_0, or

$$Q(\Theta \mid \Theta_0) = \mathcal{N}(\Theta_0, \Sigma) \tag{1.16}$$

then the ideal choice for the covariance matrix Σ is one that results in proposal steps that are big enough to traverse the space in a reasonable amount of time but are small enough to stay within the region of probability mass. Here, some knowledge of the distribution to be estimated can be helpful.

The acceptance distribution $A(\Theta \mid \Theta_0)$, then, could take the form

$$A(\Theta \mid \Theta_0) = \min\left(1, \frac{P(\Theta_0)Q(\Theta \mid \Theta_0)}{P(\Theta)Q(\Theta_0 \mid \Theta)}\right) \tag{1.17}$$

where $P(\Theta)$ is the distribution we wish to approximate, which for the purposes of Bayesian inference is usually the posterior $P(\Theta \mid D)$. The acceptance distribution in equation (1.17) is known as the *Metropolis choice* and is quite common.

Because the proposal distribution and the acceptance distribution define a Markov process—i.e. the probability distribution of the next step, given the current state, does not depend on any previous states—the process converges to an equilibrium, called the *stationary distribution* of the process. Convergence to an equilibrium means that, given an initial state s_0, the expected probability distributions for the n^{th} state s_n and the $(n+k)^{th}$ state s_{n+k} are identical for sufficiently large n and k. In other words, over time, a Markov process tends to return to the same states with the same probabilities, and this set of states and probabilities is the stationary distribution. Since the Metropolis-Hastings algorithm designs the Markov process to have a stationary distribution identical to the posterior distribution we wish to estimate, the challenge lies in tuning the process to converge to this distribution.

As mentioned earlier, the covariance matrix of the proposal distribution (which determines the step size) might generate proposal steps that are either too big or too small. In the case that they are "too big", the proposed step takes the Markov process outside of the area of reasonable probability density, and will probably not be accepted, given an acceptance distribution such as equation (1.17). In this case, very few proposals are accepted. On the other hand, if the proposed steps are too small, then the density at the proposal is very similar to the density at the current state, and the acceptance rate approaches 1. In both cases, it would take a very large number of steps (and/or proposals) in order for the Markov process to reasonably cover the posterior distribution we wish to estimate. Therefore, in tuning the algorithm, it is optimal to choose a proposal distribution from which the acceptance rate is neither too close to 0 nor too close to 1. It has been suggested that 0.2 is a good acceptance rate, but it has been shown that this is sometimes far from optimal. [Gelman et al. 2004]

Given a well-tuned Markov process, Markov chains starting in different initial states should end up converging to the same stationary distribution, which we have defined to be the desired posterior. This is the basis for a convergence diagnostic known as the Gelman-Rubin diagnostic. [Gelman and Rubin 1992] Essentially, if a set of Markov chains with "over-dispersed" initial states result in—after some number of "burn-in" steps—sets of samples that are indistinguishable from one another, then the chains have converged to the posterior distribution, and the corresponding samples can be used as an empirical approximation to the true posterior.

Markov chain Monte Carlo sampling is particularly useful for practical data analysis because one needs only to define and code the model's probability distribution, and then let the chosen MCMC algorithm (such as in the *mcmc* package in R [Geyer. 2010; R Development Core Team 2009]) do the rest of the work, aside from tuning. No special calculations or parameter updates, like in variational Bayesian methods, are needed. However, for complex distributions, tuning the algorithm may prove to be a challenge, and convergence within a reasonable time is not guaranteed.

For more information about MCMC sampling and many other aspects of Bayesian data analysis, refer to Gelman et al. [2004].

1.5 Applications

A probabilistic model and inference derived from it can be successful only if the model sufficiently describes the natural process being measured. Thus, it is very important that knowledge of the system be taken into account when designing the model. Because this is often a difficult task, a main focus for the rest of this chapter as well as those following will be the selection and design of appropriate models.

Here, I describe the context for three general applications of probabilistic models—gene interaction, miR targeting, and athletic performance—and discuss the motivation for the choice of model with regard to the data and the goal of the analyses.

1.5.1 Gene interaction models

Gene interactions are known to play an important role in cell-level biological processes, and the principal way to measure gene activity is through the *gene expression*. Gene expression is typically measured using microarrays or, more recently, genetic sequencing; the output from both of these technologies is a set of relative abundances of genes within the sample, where the number of genes measured is typically in the thousands or tens of thousands. The number of expression measurements in a time-course experiment, though, is orders of magnitude smaller, usually approximately 10. Thus, if the goal is to detect gene interactions from such a data set, the problem is severely under-determined.

Though the mechanisms of gene expression and interaction are obviously continuous-time processes, studies have been done on the time-scales of these processes, and so it is at least somewhat justified that we can measure them at carefully chosen time points and subsequently model them as discrete-time processes. Work has been done on both continuous-time models and discrete-time models—see the review of Penfold and Wild [2011] for a comparison—with varied results. Given that we always have discrete-time data (only a finite number of measurements are possible) there are challenges to models both in continuous and discrete time. I have chosen to work with discrete-time models because they are in agreement with the data, but not the underlying process, and they are generally simpler in terms of complexity.

A simple discrete-time model of gene expression and interaction is a linear model such as:

$$x_{t+1} = Ax_t + \epsilon_1 \tag{1.18}$$

where the expression x_t of all genes at a given time point $t+1$ is assumed to be a linear combination—via the matrix A—of the expressions at the previous time point t, plus some random noise. If we treat the gene expression values x_t as latent variables for which we have noisy measurements y_t then our measurements can be modeled by

$$y_t = x_t + \epsilon_2 \tag{1.19}$$

If we define and fit such a model in a Bayesian fashion then this model could be called a *dynamic Bayesian network* (DBN). A DBN typically assumes that the noise terms and expression values are Gaussian random variables and the related precision (inverse of variance) terms are distributed according to gamma distributions. In fact, two relatively early works on the application of DBNs to gene expression and interaction did just that; see Kim et al. [2003] and Husmeier [2003] for more details on these methods.

One big advantage of using a DBN instead of the equivalent non-probabilistic model is that, because probabilistic models are explicitly concerned with variance and certainty, they can give preference to higher-certainty results in the case where two (or more) linear combinations can give the same result. In a severely underdetermined linear system such as we have here, this can be very useful.

Beyond the simplest cases of DBNs, there are many ways to extend the model to achieve better results. One such model, introduced in Lèbre [2009], is referred to as *G1DBN*; it exploits conditional (first-order) dependence within nodes of the network, as well as an assumption of relative sparseness, to efficiently infer network structure. Generally speaking, *G1DBN* makes use of two assumptions, conditional dependence and sparseness, about gene interactions that are generally considered to be valid in many cases, but not necessarily all.

Another DBN, from Beal et al. [2005], includes "hidden" states in the model, which are intended to represent unmeasured quantities that have a significant impact on gene expression. These hidden states could be compared with component analysis (e.g. PCA or ICA) in that a single hidden state can explain variance in any number of gene expression values, and thus potentially improve the model fit in cases where there seems to be an unknown external quantity driving a significant amount of gene expression changes. However, adding hidden states actually increases the complexity and the under-determinedness of the model, though the hidden states have been shown to improve interaction inference in many cases.

Because the application of DBNs to gene expression and interaction is reasonably well-studied, significant improvements in these methods are largely based on further incorporation of knowledge and assumptions about the underlying systems. Therefore, it is helpful to look at literature on all types of gene expression time-course experiments and analysis in order to learn from them and apply that knowledge to improve the inference of gene interactions.

A thorough review of gene expression time-series analysis can be found in Bar-Joseph et al. [2012] and an earlier version from the same principal author, Bar-Joseph [2004], indicates the rate at which the body of literature is expanding. Much of the work deals with profiles of gene expression over time (e.g. increasing, decreasing, peaking) and co-expression of genes, as well as interaction models such as the DBN. A considerable body of work on gene clustering exists, and this work seems relatively under-utilized in gene interaction models, and thus might provide valuable improvement in gene interaction inference.

Gene clustering, in this context, is the grouping of genes by expression pattern, such that genes in the same cluster can be said to have similar expression values under a variety of conditions or time points. It can then be said that genes in the same cluster might share a similar function, be co-regulated by some common regulator, or be related in some other meaningful way.

A significant proportion of gene clustering research looks at expression patterns across time; Ernst et al. [2005] collect short time-series data by assigning them to specific pre-defined profiles that have meaning within the particular experiment; Schliep et al. [2003] perform a similar cluster analysis of time-series, but instead of using pre-defined expression patterns, they use a hidden Markov model (HMM) to infer dynamics of a limited number of clusters between a small number of states; Sivriver et al. [2011] cluster genes to inferred expression profiles, focusing mainly on impulse models in experiments where one might expect peaks in the expression values.

Clustering on its own can give powerful insight into cellular processes, particularly because clustering is a well-established method of dimensionality reduction, and gene expression analysis contains some very high-dimensional problems. Thus, it stands to reason that clustering can be combined with other methods to infer meaning in a reduced-dimension space. In particular, gene interaction models might benefit from a smaller number of dimensions and reduced correlation between regulators. Hirose et al. [2008] and Shiraishi et al. [2010] have done work in this direction, combining state-space models and clustering heuristics for simultaneous, integrated inference. We extend this idea into a formally rigorous model—a DBN with integrating Bayesian clustering—in chapter 2, published in Godsey [2013b], finding that clustering gives significant advantages in gene interaction inference.

1.5.2 Models of miR targeting

MicroRNAs (miRs) are short RNA sequences which are known to affect expression of messenger RNA (mRNA), often by binding to complementary sequences and either inhibiting translation or directing cleavage of that mRNA. For more information about miRs, see `www.mirbase.org`. [Kozomara and Griffiths-Jones 2011; Griffiths-Jones et al. 2008, 2006]

Like with genes or mRNAs, miR expression can be measured using microarrays or sequencing, quantifying the relative abundance of the several hundred miRs which might be present in a sample. Since it is known that miRs can target and/or regulate mRNAs, much research has gone into detecting or inferring such interactions. Recently, some of the most successful attempts to identify likely target pairs include the integration of expression data—most often microarrays—with sequence-based target prediction algorithms that consider the binding affinities between a particular miR and a complementary or near-complementary section of an mRNA sequence. Each data source by itself is prone to error—expression data are noisy, correlation does not imply causation, and prediction algorithms are rife with "false" positives. But, the combination of information from two very different sources had led to vast improvements in the ability to identify likely candidate target pairs. A nice review of the topic can be found in Muniategui et al. [2013].

In the case where we have paired expression measurements for both mRNAs and miRs—i.e. we have both mRNA and miR expression data for the same set of samples—as well as sequence-based predictions, we can modify a standard DBN model so that the regulators (miRs) are separated from the regulatees (mRNAs) such that the vector of mRNA expression y_i in sample i can be expressed as a linear combination of miR expression values x_i via a matrix of coefficients A, which in the most basic form looks like:

$$y_i = Ax_i + \epsilon_i \tag{1.20}$$

Each entry in the matrix A can then be informed (via a prior distribution) by sequence-based predictions. A few algorithms have been developed starting with this basic idea.

GenMiR++ [Huang et al. 2007b] is an algorithm that uses variational Bayesian methods to infer negative (down-regulating) interactions in expression data, given an $n \times m$ binary matrix where a 1 in entry (i, j) indicates that miR i is predicted to target mRNA j. This algorithm was later updated, to become, eventually, *GenMiR3* [Huang et al. 2008], with the most prominent update being that the newer algorithm is able consider sequence-based information (not just presence or absence on a list of predictions) in determining the strength and likelihood of targeting interactions.

TaLasso [Muniategui et al. 2012] is another algorithm that combines the presence of an miR-mRNA pair in a targeting prediction database with expression data. It uses LASSO regression, restricted to non-positive interactions, and includes tuning parameters to adjust the sensitivity/sparseness of the solution. *TaLasso* has been shown to outperform *GenMiR++* in some cases. [Muniategui et al. 2012, 2013]

Another Bayesian model, proposed by Stingo et al. [2010], uses a Markov chain Monte Carlo (MCMC) algorithm to fit the model and estimate parameter values, using a model formulation that is similar to that of *GenMiR++* and *GenMiR3*. It restricts interactions to be non-positive using a combination of binomial and gamma distributions, like both *GenMiR* algorithms, and can include sequence-based algorithms and scores, as *GenMiR3* does. In addition, a “time-variant” version of the model is presented, in which targeting parameters are allowed to vary over time in a time-series data set.

In some analyses, a basic Pearson correlation is used to rank putative targets, possibly in combination with prediction algorithms. [Jayaswal et al. 2009; Muniategui et al. 2013] Spearman correlation and other varieties of regression have been proposed for the same task, but none have performed as well as *GenMiR* or *TaLasso*. [Muniategui et al. 2013]

In chapter 3, and published in Godsey et al. [2012], we describe a related

Bayesian model for inferring miR-mRNA targeting interactions based on target prediction algorithms and expression data, which we fit using variational Bayesian methods like the *GenMiR* algorithms, and which can utilize any sequence-based (or other external) information like *GenMiR3* and the Bayesian model from Stingo et al. [2010]. Unlike the models just described, which either assume unreplicated measurements or ignore/average replicates, our model considers replication and uses it to propagate uncertainty all the way into the inference results of interest.

1.5.3 miR targeting with clustering

If we take the main ideas from the previous two applications

1. Clustering can improve gene interaction inference, and
2. propagating uncertainty and incorporating exogenous information can improve miR targeting inference,

and we combine them into a single model; the result is a model of miR targeting that includes integrated clustering and which hopefully further improves inference.

miR clusters and clusters of miR-regulated mRNAs have previously been investigated [Tanzer and Stadler 2004; Wang et al. 2009], but the idea has not been incorporated into probabilistic models using paired expression data.

Due to my results in gene expression DBN models, we know that clustering of genes by their expression profiles improves interaction inference by reducing the interaction parameter space as well as the uncertainty arising from highly correlated potential regulators. [Godsey 2013b] To summarize, it is difficult to determine the true regulator if two (or more) potential regulators are highly correlated, and this high inferential uncertainty can cause both potential regulators (together with their common regulatee) to fall far down the list of top-ranked inferred interactions; thus, it is better to group highly correlated regulators together and allow all members of the group to maintain a top ranking than to allow competition to diminish the inferred contribution of all members.

This idea can be applied directly to probabilistic models of miR-mRNA interaction (i.e. miR targeting) such as that previously described in Godsey et al. [2012], or *GenMiR* or *TaLasso*. This is the goal of chapter 4, published in Godsey [2013a].

1.5.4 Athletic performance

Though clearly not within the field of bioinformatics, the task of comparison of athletic performances across different disciplines possesses some strikingly similar challenges to the probabilistic models in bioinformatics I have already outlined here. First of all, the data available—lists of the top few hundred performances of all time in each event—are only a very small subset of the data that are theoretically possible to attain. Secondly, though these data might not be seem high-dimensional at first glance, the number of athletes competing every year and the number of variables involved with training and competing indicate otherwise. This combination of a relatively complex process with the fact that available data comprises only a small fraction of what is it possible to measure allows us to utilize a few of the ideas from models in bioinformatics to improve the quality of inference. First, though, let us review prior work in the area, which is unusually sparse.

There are a few good models that are valid for running events only, particularly longer distances, namely those by McMillan [2011], Cameron [1998], Riegel [1977], and Daniels and Gilbert [1979]. These models rely on physiological measurements such as speed and running economy to compare performances at different race distances, either for men or for women, but not between them.

Purdy Points [Gardner and Purdy 1970] have long been used to compare marks from different events in both track and field, but these scores are based mainly on the world records of each event at a particular date in the past, which leads to two main disadvantages: (1) it is impossible to compare world records to each other if the model is based on them, and (2) basing the model on such a small data set leads to much uncertainty and variation in the scores as the records and model evolve over time.

Currently, the most popular method for comparing performances across all events in track and field as well as road running is to consult the IAAF scoring tables [Spiriev and Spiriev 2011]. These tables are updated every few years using methods that are not fully disclosed, with the last two updates occurring in 2008 and 2011. But, we know that point values P in these tables can be calculated using a formula of the form $P = a(M-b)^c$, where M is the measured performance (use $M = -T$ for running times T, where a lower performance is better) and a, b, and c are constants estimated by undisclosed methods. [International Association of Athletics Federations 2001]

No fully probabilistic model has been applied to such performance data, but athletic performances lend themselves well to such modeling. The random nature of the events, plus the expectations of the athletes that exist because of their previous performances, fit very well into the probabilistic framework. Therefore, chapter 5, and published in Godsey [2012], describes a probabilistic model where, for each event, I estimate a log-normal distribution, allowing the calculation of both the probability that a specific mark is exceeded as well as the expected number of such performances within a given time period. This paper was published just before the 2012 Olympic Games in London, with some measurable predictions that can be compared with subsequent athletic results from 2012 through the present day.

1.6 Outline of the rest of this thesis

Chapter 2 gives the basis for the dynamic Bayesian network and integrated clustering model mentioned above which improves the quality of inference of gene interactions when analyzing time course expression data. This chapter appeared in *PLoS ONE* in 2013. [Godsey 2013b]

Chapter 3 describes a probabilistic model of miR-mRNA interaction/targeting that incorporates exogenous, sequence-based target predictions. This chapter appeared in *PLoS ONE* in 2012. [Godsey et al. 2012]

Chapter 4 combines the ideas from chapters 2 and 3, and refines some details in order to create another probabilistic model of miR targeting of mRNAs that includes both exogenous prediction information and clustering. This chapter appeared in the *Journal of Integrative Bioinformatics* in 2013. [Godsey 2013a]

Chapter 5 shows how a probabilistic model of athletic performances can give significant insight into what it means to have a rare or exceptional performance, in addition to establishing a new benchmark of how to compare performances between two disparate disciplines. This chapter appeared in the *Journal of Quantitative Analysis in Sports* in 2012. [Godsey 2012]

Chapter 2

Improved inference of gene regulatory networks through integrated Bayesian clustering and dynamic modeling of time-course expression data

Published in *PLoS ONE* [Godsey 2013b]

by Brian Godsey[1]

1 Department of Statistics and Probability Theory, Vienna University of Technology, 1040 Vienna, Austria

Abstract: Inferring gene regulatory networks from expression data is difficult, but it is common and often useful. Most network problems are under-determined—there are more parameters than data points—and therefore data or parameter set reduction is often necessary. Correlation between variables in the model also contributes to confound network coefficient inference. In this paper, we present an algorithm that uses integrated, probabilistic clustering to ease the problems of under-determination and correlated variables within a fully Bayesian framework. Specifically, ours is a dynamic Bayesian network with integrated Gaussian mixture clustering, which we fit using variational Bayesian methods. We show, using public, simulated time-course data sets from the *DREAM4 Challenge*, that our algorithm outperforms non-clustering methods in many cases (7 out of 25) with fewer samples, rarely underperforming (1 out of 25), and often selects a non-clustering model if it better describes the data. Source code (*GNU Octave*) for BAyesian Clustering Over Networks (*BACON*) and sample data are available at: `http://code.google.com/p/bacon-for-genetic-networks`.

2.1 Introduction

Inferring gene regulatory networks from high-throughput gene expression data is a difficult task, in particular because of the high number of genes relative to the number of data points, and also because of the random noise that is present in measurement. Over the last several years, many new methods have been developed to address this problem; a nice review of these can be found in Penfold and Wild [2011]. This review directly compares several different types of approaches by summarizing the correctness of the genetic networks inferred from synthetic (*in silico*) data generated from a known network. Of particular interest are the results of each of the algorithms when applied to the *DREAM4 In Silico Network Challenge* data sets, which includes data types such as "knock-out", "knock-down", and time-series data among the sub-challenges. See Prill et al. [2011] for more details on the *DREAM* challenges.

Though Greenfield et al. [2010] have had success combining methods in order to infer genetic networks from different types of data simultaneously, here we focus on time-series data and the corresponding methods for network inference. In the review of Penfold and Wild [2011], two types of algorithms seem to outperform the others when applied to time-series data: dynamic Bayesian networks and causal structure identification (CSI) in non-linear dynamical systems (NDSs).

Dynamic Bayesian networks (DBNs) are typically some variation of the basic linear model

$$x_{t+1} = Ax_t + \epsilon_1 \tag{2.1}$$

$$y_t = x_t + \epsilon_2 \tag{2.2}$$

where in the context of gene regulatory networks, x_t is the vector of "true" gene expression levels at time t, y_t is a vector of observations of these expression levels, A is a matrix of interaction coefficients, and ϵ_1 and ϵ_2 are random (Gaussian) noise. More information on DBNs and their application to gene regulatory networks can be found in Kim et al. [2003] and Husmeier [2003].

The algorithms considered in Penfold and Wild [2011] include a model very similar to that of the basic DBN formulation above, but which exploits conditional [first-order] dependence within nodes of the network, as well as an assumption of relative sparseness, to efficiently infer network structure. This model, from Lèbre [2009] is referred to as *G1DBN* and is available as an *R* package from *CRAN* [R Development Core Team 2009]. The second DBN considered by Penfold and Wild [2011] is that of Beal et al. [2005], which adapts a state-space model with inputs to include hidden states, the quantity and values of which are inferred through variational Bayesian learning. This algorithm is referred to as *VBSSM*, as in the review. Causal structure identification (CSI) in non-linear dynamical systems (NDSs) avoids the restriction of linearity when determining network structure, and in the case of Klemm [2008], which is also considered in the review, latent interaction parameters of a discrete Gaussian process model are inferred using a

Bayesian framework. According to Penfold and Wild [2011], both the *G1DBN* and *VBSSM* algorithms performed well on the *DREAM4* data sets, as did the CSI algorithm of Klemm [2008]. Both DBNs and CSI outperformed ordinary differential equations (ODEs) and models using Granger causality.

Though these results are convincing, there is still room for improvement, and the discussion of optimal methods is still open; in fact, the body of research in the area of gene expression time-series analysis continues to grow quickly. A recent review, Bar-Joseph et al. [2012], outlines the state of the art in gene expression time-series analysis, including much information on clustering methods and software. We can see that, when compared to a similar, earlier review, Bar-Joseph [2004], a considerable amount of work has been done. However, we feel that there is still a branch of time-series data analysis that is under-utilized in gene regulatory network inference. Despite the vast amount of work that has been done on the clustering of gene expression data, much of which deals specifically with time-series, relatively little work has been done on inferring time-dependent interactions between gene clusters or between a gene cluster and an individual gene. Let us briefly discuss clustering methods for time-series data before continuing on to its potential use in inferring gene regulatory networks.

In order to successfully cluster time-series data, we need to utilize the stronger dependencies between data in consecutive time points relative to more distant time points. Quite often, researchers are interested in expression patterns across time; Ernst et al. [2005] cluster short time-series data around specific pre-determined profiles that may have meaning within the particular experiment. Schliep et al. [2003] perform a similar cluster analysis of time-series, but instead of using pre-determined expression patterns, they use a hidden Markov model (HMM) to infer dynamics of a limited number of clusters between a small number of states (*e.g.* nine discrete expression levels). Sivriver et al. [2011] take a slightly different approach by clustering genes to inferred profiles, focusing mainly on impulse models in experiments where one might expect peaks in the expression values.

In each of the above papers, it was shown that gene clustering can infer biological meaning, whether co-expression, co-regulation, involvement in particular biological processes, or some other effect. Such information may also be valuable in inferring genetic regulatory networks. Hirose et al. [2008] and Shiraishi et al. [2010] have done work in this direction, combining state-space models and clustering heuristics for simultaneous, integrated inference. However, both of these are demonstrated on data containing hundreds of genes which are clustered or grouped into a low (fewer than 20) number of clusters/modules and subsequently the large cluster size prevents any meaningful conclusions about regulatory interactions between specific genes.

In this paper, we describe a fully Bayesian model of gene cluster interaction, and we demonstrate that probabilistic gene clustering in conjunction with a dynamic Bayesian network can aid in the inference of gene regulatory networks, even in the *DREAM4* data sets, where no clusters were explicitly included. It achieves this by potentially reducing—in a fully Bayesian manner—the parameter space and helping solve the problem of solution identifiability in under-defined, noisy data models such as are common in gene expression analysis. The algorithm presented here is a variational Bayesian hybrid of a DBN and a Gaussian mixture clustering algorithm, both of which have been shown to infer meaningful solutions to their respective problems [Beal et al. 2005; Teschendorff et al. 2005], and which we show can work even better in tandem. We call this algorithm *BAyesian Clustering Over Networks* (*BACON*). *BACON* is built specifically to simultaneously consider multiple data sets based on the same network, such that for each data set, expression states are inferred independently, but that cluster membership and regulatory dynamics are assumed to be constant for all data from the given network, regardless of the particular data set. This gives more accurate results than a heuristic combination of interaction rankings based on the various time-series for each of the *DREAM4* networks.

2.2 Methods

In this paper we introduce an algorithm called *BACON*, which is a variational Bayesian algorithm that combines a Gaussian mixture clustering model with a DBN. However, before we give the specific formulation of our model, it may be helpful first to look at a simple case where integrated clustering can help infer gene regulatory networks, even if no "true" clusters are present.

2.2.1 A simple example

Assume, as an illustration, that we have a three genes, X, Y, and Z and that we have time-series expression data for each of them, such that the observed expression levels of these at time t are given by x_t, y_t, and z_t, respectively, for time points $t \in \{1, \ldots, T\}$. Let us, for simplicity's sake, assume that we are concerned only with potenial regulators of gene Z, and that X and Y are the only two candidates. Furthermore, we assume a simple linear model of dynamics, in which z_{t+1} is assumed to be a noisy observation of the dot/inner product of the vector of two interaction coefficients a_x and a_y with the vector of x_t and y_t, namely:

$$z_{t+1} = (a_x, a_y)(x_t, y_t)' + \epsilon \tag{2.3}$$

Note that this is a simple linear model on three variables, where all interaction coefficients except a_x and a_y are set to zero. It is a special case of the standard linear model, given in equation 2.1, where for this example we treat the observations as the true expression values, we attempt only to infer a_x and a_y, and for illustration purposes we ignore all other possible interaction coefficients. When attempting to infer a_x and a_y under a Bayesian framework, we make the following assumptions:

$$z_{t+1} \sim \mathcal{N}\big((a_x, a_y)(x_t, y_t)', \lambda\big) \tag{2.4}$$

$$(a_x, a_y) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda}) \tag{2.5}$$

where λ is a precision parameter (inverse of variance), $\boldsymbol{\mu}$ is the prior mean of the multivariate normal distribution, and $\boldsymbol{\Lambda}$ is a 2×2 precision matrix (inverse of the covariance matrix).

Given these assumptions, the data x_t, y_t, and z_t, and the precision parameter λ (fixed), the estimated posterior distribution for (a_x, a_y) under a variational Bayesian framework (see Beal [2003] and Winn [2003] for a detailed explanation) is multivariate normal, with mean $\hat{\boldsymbol{\mu}}$ and precision $\hat{\boldsymbol{\Lambda}}$, such that

$$\hat{\boldsymbol{\Lambda}} = \boldsymbol{\Lambda} + \sum_{t=1}^{T-1} \lambda \begin{bmatrix} x_t^2 & x_t y_t \\ x_t y_t & y_t^2 \end{bmatrix} \tag{2.6}$$

$$\hat{\boldsymbol{\mu}} = \Big(\boldsymbol{\mu} + \sum_{t=1}^{T-1} \lambda \begin{bmatrix} x_t & y_t \end{bmatrix}\Big) \hat{\boldsymbol{\Lambda}}^{-1} \tag{2.7}$$

Under some conditions, such inference works quite well, but if the expression profiles for X and Y are highly correlated (or negatively correlated), then the determinant of $\hat{\boldsymbol{\Lambda}}$ approaches zero, and the diagonal elements of $\hat{\boldsymbol{\Lambda}}^{-1}$ (the estimated variances of a_x and a_y) approach infinity. Such a problem can be overcome with a strong prior for a_x and a_y, but this is usually not desireable since typically $\boldsymbol{\mu}$ is set to zero (as in Beal et al. [2005]), and a high prior precision $\boldsymbol{\Lambda}$ merely pulls the estimate $\hat{\boldsymbol{\mu}}$ towards zero, and potentially decreases the statistical significance of the inferred interaction parameters. Thus, we are faced with a decision between strong priors or very high variances of posterior parameter estimates.

If the x_t and y_t are highly correlated, and if they are likewise correlated with z_{t+1}, then we might be able to say with near certainty that either X or Y regulates Z, but we could not say which one. This may be acceptable on a small scale, but would be difficult in a gene expression time-series experiment with hundreds of genes and thousands of putative interaction coefficients and covariances. It could be interesting to optimize the choice of a set of, for example, ten gene interactions, with respect to the probability of at least one of them being verifiable in an independent evaluation. But, this would be difficult for experiments with large numbers of genes, and so typically only the individual variances are considered when calculating the statistical significance of the estimated interaction parameter values. Estimating, potentially, thousands of interaction parameters is very difficult in dynamic gene expression time-series analysis because, for example in the basic linear model given in equation 2.1, there are generally many possible values for the transition matrix, each of which could produce the data. In other words, a given gene expression time-series could be reproduced by many different linear combinations of other [lagged] time-series. A particular case of this is when two potentially regulating genes have highly correlated expression profiles, which, as we have shown above, can cause some difficulty in inference.

Here, we propose that clustering genes and inferring the dynamics of the clusters can help avoid the case in which highly correlated gene profiles inhibit interaction inference. In our example, if genes X and Y have highly correlated expression profiles, then for weak priors the precision estimate in equation 2.6 is nearly singular, and thus by treating X and Y as two contributors to the same dynamic quantity, we avoid this particular singularity problem altogether. Then, a standard method (DBN or similar) could more easily infer that both X and Y (as one cluster) are likely regulators of Z. If, when creating a list of the most likely individual gene-gene interactions, we simply assign all the inferred interaction coefficients for a cluster to each of its members, we can obtain a ranking of interaction pairs that is comparable to the ranking obtained from a standard DBN.

It may seem, at first, that passing along inferred interaction coefficients to all cluster members would create many false positives. However, if clusters include—by definition—highly correlated expression profiles, then if a cluster appears to be a good potential regulator of a gene, all of the cluster's members must also have profiles that generally indicate potential regulation, and in the absence of clustering, it would be difficult to identify the best interaction parameters. This is true whether or not any or all of the concerned genes are actually verifiable regulators, and thus clustering together correlated expression profiles—regardless of the biological meaning of the clustered genes—could improve inference. For instance, in our example, the presence of gene Y (if highly correlated with X) adversely affects the identification of X as a regulator of Z, a problem that can be avoided if X and Y are treated as members of the same cluster. In a data set with hundreds of genes, the chance of having at least one pair of highly correlated expression profiles is rather large. Of course, we must be careful in our construction of clusters and their dynamics, but as we show, Bayesian inference provides the means to select a number of clusters, to assign cluster membership, and to estimate cluster interaction parameters in an optimal way. We describe this below.

2.2.2 Model

Given K clusters and G genes, we assume that the cluster expressions $\mathbf{F_t}$ for time points $t \in \{1, ..., T\}$ follow the standard linear dynamics

$$\mathbf{F_{t+1}} = \mathbf{SF_t} + \mathbf{S_c} + \epsilon \tag{2.8}$$

where $\mathbf{F}_t$ is a vector of length K, $\mathbf{S}$ is a $K \times K$ transition matrix, $\mathbf{S_c}$ is a column vector of length K, and ϵ is vector of Gaussian noise. The k^{th} element of $\mathbf{F_t}$, f_{kt}, is the expression of cluster k at time t. The vector $\mathbf{S_c}$ represents linear trends in cluster expression levels over the time points, and its inclusion in the model prevents such trends from being confused for interactions in similarly trending clusters.

The expression of gene g at time t is given by μ_{gt}, and the membership of gene g to cluster k is given by the k^{th} element of the indicator vector ξ_g. Each gene g belongs to exactly one cluster k, and so ξ_g contains a single 1 in in the k^{th} element and zeros elsewhere. The n^{th} observation/replicate of μ_{gt} is x_{gtn}. The corresponding prior distributions are:

$$x_{gtn} \sim \mathcal{N}(\mu_{gt}, \lambda) \tag{2.9}$$

$$\mu_{gt} \sim \mathcal{N}(\xi \mathbf{F_t}', \xi_g \gamma_t') \tag{2.10}$$

$$\mathbf{F_t} \sim \mathcal{N}(\mathbf{SF_{t-1}} + \mathbf{S_c}, \Sigma) \tag{2.11}$$

where the λ is a technical precision (inverse of variance) representing the measurement errors, assumed to be independent, γ_t is a vector of precisions of length K, and Σ is a $K \times K$ precision matrix which we require to be diagonal, as in Beal et al. [2005], so that in the posterior distribution estimates, the rows of $\mathbf{S}$ are independent. We also formulate our other prior distributions as in Beal et al. [2005]: for the elements of $\mathbf{F_1}$, $\mathbf{S}$, and $\mathbf{S_c}$, we use zero-mean normal distribution priors whose precisions we iteratively update to maximize the marginal likelihood estimate (discussed below). Likewise, for the hyper-parameters of the gamma distribution priors we assume for the elements of the precisions λ, γ_t, and Σ. For ξ_g, we use a uniform prior distribution over the K possible clusters.

For multiple time-course data sets from the same gene regulatory network, as we have in the *DREAM4 Challenge* data sets we use in this paper, we infer all of the parameters separately for each of the series, except for the dynamics parameters $\mathbf{S}$ and $\mathbf{S_c}$ and the membership indicator vectors ξ_g, which are shared and inferred simultaneously for all time-series from the given network.

2.2.3 Inference

To estimate the parameters of our model, we use a variational Bayesian algorithm analagous to those described in Beal [2003] and Winn [2003], which has been previously used to fit a DBN to gene expression time-series in Beal et al. [2005], as well as a Gaussian mixture model for gene clustering in Teschendorff et al. [2005].

In short, the algorithm used in this paper estimates the posterior parameter distribution $P(\boldsymbol{\theta} \mid \mathbf{D})$ given the data $\mathbf{D}$ using a factorable distribution $Q(\boldsymbol{\theta}) = Q(\theta_1)Q(\theta_2)\ldots Q(\theta_n)$ whose factors can be iteratively updated so that with each update, $Q(\boldsymbol{\theta})$ becomes a better approximation for $P(\boldsymbol{\theta} \mid \mathbf{D})$, as measured by the Kullback-Leibler divergence between the two. We have chosen conjugate prior distributions for each of the parameters we estimate, and therefore the posterior distribution estimate $Q(\theta_i)$ for each parameter is of the same form as its prior, and the parameters of these distributions are updated iteratively according to variational Bayesian inference, as in equations 2.6 and 2.7.

We fit the model using 10 starts with randomized initial parameter values, and with a range of cluster numbers less than or equal to the number of genes in the data set (in the case of the *DREAM4* data, $k \in \{5, 6, \ldots, 10\}$) and then accept the model that has the highest estimated marginal likelihood. Accepting the model with the maximum marginal likelihood is simpler than combining all models based on their likelihoods, when in fact it is rare for a second, different model to have a likelihood close enough (i.e. a log likelihood within 3 or 4) to the best model for it to make a significant impact on the interaction rankings.

We are concerned primarily with the transition matrix $\mathbf{S}$ and the membership indicators ξ_g; using posterior estimates for these, we can rank directed gene-gene interactions by their statistical strength. Specifically, for each directed cluster pair interaction $i \rightarrow j$ $(i \neq j)$, we calculate the posterior mean estimate for element (i, j) of $\mathbf{S}$ divided by its posterior standard deviation, assign this value to all possible directed pairs within the two clusters, and we rank by largest absolute value.

The *Octave* code implementing this algorithm—available at: `http://code.google.com/p/bacon-for-genetic-networks`—takes approximately 40 minutes on a single core of a 1.2 GHz processor for a single random start and a given number of clusters. Multiple starts and different numbers of clusters can be run in parallel; see the code for more details.

2.2.4 Data

We used the *DREAM4 In Silico Network Challenge* data sets to evaluate the performance of our model. See Prill et al. [2011, 2010]; Marbach et al. [2010, 2009] for more details on the *DREAM* challenges. We utilized only the 10-gene time-series data, which consists of five simulated networks. For each of the networks, there are five time-series experiments, each with 20 time points. No simulated technical replicates were included, but random noise was added. The list of actual, "gold standard", interactions was provided after the official challenges ended.

2.3 Results

For each of the five data sets, each corresponding to a single gene regulatory network, we inferred the network using all available time-series (five each) and used the inferred interactions and the known gold standard to calculate the area under the receiver operating characteristic (AUROC) curve and the area under precision-recall (AUPR) curve, as in Penfold and Wild [2011]. Table 2.1 gives the AUROC and AUPR for *BACON* both with and without clustering, as well as the corresponding scores from the *G1DBN* and *VBSSM* models, as reported in Penfold and Wild [2011]. *BACON* gives an AUROC score better than both *G1DBN* and *VBSSM* in two out of five data sets— likewise for the AUPR scores— and is comparable to the other two algorithms in the remaining data sets. Given that *BACON* without clustering compares favorably with other algorithms, and that *BACON* with clustering gives the exact same results as *BACON* without clustering (the inferred number of clusters in each case was 10, the number of genes), we conclude that both versions of *BACON* give satisfactory results for these data sets.

	Algorithm	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5
AUROC	*BACON*	**0.82**	**0.67**	0.72	0.81	0.88
	BACON (no clustering)	**0.82**	**0.67**	0.72	0.81	0.88
	G1DBN	0.73	0.64	0.68	**0.85**	**0.92**
	VBSSM	0.73	0.66	**0.77**	0.80	0.84
AUPR	*BACON*	**0.42**	0.36	**0.51**	0.49	0.57
	BACON (no clustering)	**0.42**	0.36	**0.51**	0.49	0.57
	G1DBN	0.37	0.34	0.45	**0.69**	**0.77**
	VBSSM	0.38	**0.41**	0.49	0.46	0.64

Table 2.1: Algorithm results comparison for the DREAM4 networks. The area under the receiver operating characteristic (AUROC) curve and area under precision-recall (AUPR) curve for each of the five data sets. Here, we included *BACON* without clustering in order to establish that the plain DBN algorithm is generally as good as the other two DBN algorithms. The scores for *G1DBN* and *VBSSM* were taken from Penfold and Wild [2011]. The best score for each data set is shown in bold.

However, the *DREAM4* time-series data are not typical; a single time-series with 20 time points is somewhat uncommon in practice (most experiments have 10 or fewer time points), and five independent time-series for the same gene network would be extremely rare. Thus, we subsequently consider each of the time-series individually, in order to see if an even more under-determined problem (only 20 data points for each of the 10 genes instead of 100) favors the model version with clsutering. We show in Table 2.2 the AUROC and AUPR of the 25 individual time-series (five from each of five data sets) for the *BACON* model both with and without clustering.

	Time-series	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5
AUROC	1	0.68 (**0.71**)	0.61 (0.61)	0.64 (0.64)	0.62 (0.62)	0.57 (0.57)
	2	**0.75** (0.66)	0.70 (0.70)	**0.68** (0.66)	0.77 (0.77)	0.64 (0.64)
	3	**0.67** (0.61)	0.62 (0.62)	**0.65** (0.60)	**0.60** (0.58)	0.65 (0.65)
	4	**0.61** (0.53)	0.66 (0.66)	0.59 (0.59)	**0.60** (0.60)	**0.74** (0.66)
	5	**0.66** (0.63)	0.64 (0.64)	0.59 (0.59)	0.76 (0.76)	0.78 (0.78)
AUPR	1	0.24 (**0.39**)	0.24 (0.24)	0.32 (0.32)	0.19 (0.19)	0.23 (0.23)
	2	**0.42** (0.27)	0.34 (0.34)	0.28 (**0.30**)	0.26 (0.26)	0.32 (0.32)
	3	**0.30** (0.19)	0.21 (0.21)	**0.24** (0.19)	0.15 (**0.16**)	0.24 (0.24)
	4	**0.24** (0.16)	0.38 (0.38)	0.21 (0.21)	**0.24** (0.18)	**0.27** (0.23)
	5	**0.22** (0.19)	0.20 (0.20)	0.17 (0.17)	0.34 (0.34)	0.33 (0.33)

Table 2.2: Results of BACON on individual DREAM4 time series. For each of five individual time-series in each of the five data sets, the area under the receiver operating characteristic (AUROC) curve and area under precision-recall (AUPR) curve. For each time series, we give two of each score, one for *BACON* with clustering and one for *BACON* without clustering (in parentheses). The higher of the two scores appears in bold. If the two scores are identical, neither is in bold.

In many cases, the with-clustering and without-clustering scores were identical—i.e. 10 clusters is optimal—but in several other cases, fewer clusters gave a higher marginal likelihood score, and the corresponding AUROC and AUPR were indeed better, more often than not. Specifically, for 15 of the 25 time-series, *BACON* with clustering performed identically to the version without, but in seven cases, the version with clustering gave higher scores for both AUROC and AUPR. In only one case, the without-clustering version outperformed the with-clustering version in both AUROC and AUPR. These tallies are summarized in Table 2.3. Clearly, for smaller data sets such as a single time series, there is some benefit to be had from clustering the genes, when compared to non-clustering DBNs.

		Higher AUPR		
		with clustering	equal	without
Higher AUROC	with clustering	7	0	2
	equal	0	15	0
	without	0	0	1

Table 2.3: Results comparison: with vs without clustering. Among the five individual time-series in each of the five data sets (25 total time series), here we give a tally of how many times *BACON* with clustering outperformed *BACON* without clustering, or vice versa, or if the AUROC and AUPR scores are equal.

2.4 Discussion

Inferring gene regulatory networks from expression data is not usually easy, but it is common and often useful. Because of the under-determined nature of the problem—there are more parameters than data points—some reduction of the parameter set is often necessary in order to reach any meaningful conclusion at all. Sometimes, we can accomplish this through heuristic methods and decisions about which data are more important prior to the main statistical analysis. Other times, this is not desirable. In this paper, we present a probabilistic model of time-series gene expression with an integrated, theoretically sound method of parameter space reduction. We have described its implemetation and use, including a simple analytically-tractable example in which clustering is advantageous to network inference even if no "true" cluster exists, and if we are not at all concerned with cluster membership.

Many of the expectations we had for the Bayesian model turned out to be true. In particular, we expected the model to favor clustering mainly in data sets with few samples; in fact, the model preferred (via the likelihood function) not to cluster when we included all data for each network (100 samples, 20 from each of five time-series), but elected to cluster for 10 of the 25 separate time series (20 samples each). Likewise, because of the under-determined nature of network inference, we also expected the clustering model to perform better than a model without clustering if there are fewer samples. This also proved true; of the 10 time-series for which the model's marginal likelihood was highest for less than 10 clusters, seven were indeed better than without clustering (when comparing both AUROC and AUPR scores), and only one proved worse.

We believe that probabilistic clustering could be very useful in gene network inference, though there are disadvantages. For one, the computational time is generally much higher when clustering. This is due to the need to do model fits for a range of possible cluster numbers. For the purposes of this paper, in addition to doing the 10 random starts for the non-clustering model version, we do 10 random starts for the cluster quantities we wish to consider. Of course, the algorithm is much faster for smaller cluster numbers, as the size of the parameter of primary interest, the interaction/transition matrix, varies with the square of the number of clusters. It would likely be beneficial, in the case of very large data sets, to use a sequential or iterative search over the number of clusters, rather than use the exhaustive search method as we have here, but we leave that for a future publication.

In summary, we have shown that there are benefits to be had by clustering genes as part of a network inference algorithm. The potential for significant correlation among genes is high in typical time-series data sets, particularly those with few samples. The algorithm we have presented here, which we call *BAyesian Clustering Over Networks* (*BACON*), can help avoid the negative consequences of inter-gene correlation for the purposes of network inference. In our tests, the algorithm outperformed its non-clustering version in 7 out of 25 time-series from the *DREAM4 Challenge*, underperforming only once, and most often electing to disregard clusters when the data did not support it. Therefore, we feel that there are significant benefits of using probabilistic clustering to aid in the inference of gene regulatory networks.

Source code (*GNU Octave*), more information about the software for BAyesian Clustering Over Networks, (*BACON*) and sample data can be found at: `https://github.com/briangodsey/bacon-for-gene-networks`.

Chapter 3

Inferring microRNA regulation of mRNA with partially ordered samples of paired expression data and exogenous prediction algorithms

Published in *PLoS ONE* [Godsey et al. 2012]

by Brian Godsey[1,2], Diane Heiser[3,4], and Curt Civin[4,5]

1 Department of Statistics and Probability Theory, Vienna University of Technology, Vienna, Austria

2 Institute for Genome Sciences, University of Maryland School of Medicine, Baltimore, Maryland, USA

3 Department of Oncology, Sidney Kimmel Comprehensive Cancer Center, Johns Hopkins University School of Medicine, Baltimore, Maryland, USA

4 Center for Stem Cell Biology and Regenerative Medicine, University of Maryland School of Medicine, Baltimore, Maryland, USA

5 Greenebaum Cancer Center, Departments of Physiology and Pediatrics, University of Maryland School of Medicine, Baltimore, Maryland, USA

Abstract: MicroRNAs (miRs) are known to play an important role in mRNA regulation, often by binding to complementary sequences in "target" mRNAs. Recently, several methods have been developed by which existing sequence-based target predictions can be combined with miR and mRNA expression data to infer true miR-mRNA targeting relationships. It has been shown that the combination of these two approaches gives more reliable results than either by itself. While a few such algorithms give excellent results, none fully addresses expression data sets with a natural ordering of the samples. If the samples in an experiment can be ordered or partially ordered by their expected similarity to one another, such as for time-series or studies of development processes, stages, or types, (e.g. cell type,

disease, growth, aging), there are unique opportunities to infer miR-mRNA interactions that may be specific to the underlying processes, and existing methods do not exploit this. We propose an algorithm which specifically addresses [partially] ordered expression data and takes advantage of sample similarities based on the ordering structure. This is done within a Bayesian framework which specifies posterior distributions and therefore statistical significance for each model parameter and latent variable. We apply our model to a previously published expression data set of paired miR and mRNA arrays in five partially ordered conditions, with biological replicates, related to multiple myeloma, and we show how considering potential orderings can improve the inference of miR-mRNA interactions, as measured by existing knowledge about the involved transcripts.

3.1 Introduction

MicroRNAs (miRs) are short RNA sequences which are known to affect expression of messenger RNA (mRNA), often by binding to complementary sequences and either inhibiting translation or directing cleavage of that mRNA. A large database of miR information and annotation can be found at `www.mirbase.org` [Kozomara and Griffiths-Jones 2011; Griffiths-Jones et al. 2008, 2006]. While much research has been performed on miR-mRNA interactions, it continues to be difficult to infer such interactions in large numbers. Typically, these interactions are validated one at a time, though high-throughput methods have recently been developed in an attempt to speed up the process of miR target discovery. We discuss these methods in the following paragraphs.

Recently, some of the most successful attempts to identify likely target pairs include the integration of expression data—most often microarrays—with sequence-based target prediction algorithms that consider the binding affinities between a particular miR and a complementary or near-complementary section of an mRNA sequence. Each data source by itself is prone to error—expression data are noisy, correlation does not imply causation, and prediction algorithms are rife with “false” positives. But, the combination of information from two very different sources had led to vast improvements in the ability to identify likely candidate target pairs. A nice review of the topic can be found in Muniategui et al. [2013].

Most algorithms that combine target predictions with expression data require such data for both miRs and mRNA, but even when miR expression data are unavailable, it is possible to infer miR activity and effective regulation under various experimental conditions using gene expression data and calculated binding strengths from target prediction algorithms. [Cheng and Li 2008]

When miR expression data is available, *GenMiR++* [Huang et al. 2007b], one of the first algorithms to combine sequence-based prediction and expression data for both miR and mRNA to infer interactions, uses variational Bayesian methods to infer negative (down-regulating) interactions in expression data, given an $n \times m$ binary matrix where a 1 in entry (i, j) indicates that miR i is predicted to target mRNA j. This algorithm was later updated, to become, eventually, *GenMiR3* [Huang et al. 2008], with the most prominent update being that the newer algorithm is able consider sequence-based information (not just presence or absence on a list of predictions) in determining the strength and likelihood of targeting interactions.

TaLasso [Muniategui et al. 2012] is another prominent algorithm that combines the presence of an miR-mRNA pair in a targeting prediction database with expression data. It uses LASSO regression, restricted to non-positive interactions, and includes tuning parameters to adjust the sensitivity/sparseness of the solution. *TaLasso* has been shown to outperform *GenMiR++* in some cases. Muniategui et al. [2012, 2013]

Another Bayesian model proposed by Stingo, *et al* [Stingo et al. 2010], uses a Markov chain Monte Carlo (MCMC) algorithm to fit the model and estimate parameter values, using a model formulation that is similar to that of *GenMiR++* and *GenMiR3*. It restricts interactions to be non-positive using a combination of binomial and gamma distributions, like both *GenMiR* algorithms, and can include sequence-based algorithms and scores, as *GenMiR3* does. In addition, a "time-variant" version of the model is presented, in which targeting parameters are allowed to vary over time in a time-series data set.

In some cases, a basic Pearson correlation is used to rank putative targets, possibly in combination with prediction algorithms. [Jayaswal et al. 2009; Muniategui et al. 2013] Spearman correlation and other varieties of regression have been proposed for the same task, but none have performed as well as *GenMiR* or *TaLasso* [Muniategui et al. 2013].

In this paper, we propose a Bayesian model for inferring miR-mRNA targeting interactions based on target prediction algorithms and expression data, which we fit using variational Bayesian methods like the *GenMiR* algorithms, and which can utilize any sequence-based (or other external) information like *GenMiR3* and the Bayesian model from Stingo, *et al.*

However, in contrast to one or more of the aforementioned algorithms, our model:

1. considers both positive and negative interactions between miR and mRNA.
2. uses a normal distribution to characterize interaction strength.
3. optimizes the weights/coefficients placed on sequence/prediction information via the same variational Bayesian algorithm that estimates the rest of the

model parameters.

4. accounts for data replicates, biological or technical, and propagates uncertainty throughout the model parameter estimates.
5. can consider a partial ordering of the samples.

With respect to these points, we enumerate how the three algorithms described—*GenMiR3*, *TaLasso*, and the Stingo model—differ from our model:

1. All three algorithms consider only negative interactions, but we chose to consider both positive and negative interactions since some positive indirect effects may, in some cases, better explain changes in expression values than negative effects only. [Vasudevan et al. 2007; Jopling et al. 2006] We still have the option, when searching for direct miR targets, to consider only the inferred negative interactions; we explore this option in the *Results* section.

2. We chose to use a normal distribution to characterize the interaction coefficients where *GenMiR3* and the Stingo model have used combined binomial and gamma distributions. The binomial-gamma combination more strongly enforces sparseness in interactions, but considers only negative interactions, as mentioned. *TaLasso* is non-Bayesian and provides no distribution for these coefficients.

3. Both our model and the Stingo model estimate the influence of external target prediction information in the same manner as other parameters (variational Bayes and MCMC, respectively) while *GenMiR3* uses the [non-Bayesian] conjugate-gradient method to optimize the weights placed on the target prediction information. *TaLasso* doesn't consider such information.

4. Based on their descriptions and implementations, none of the algorithms explicitly account for technical/replicate variance or otherwise allow for grouping of samples without taking their average value before starting the algorithm.

5. With the exception of the Stingo model, which in its "time-variant"version allows some interaction parameters to change over time, none of the models considers an ordering of the expression samples.

In the following sections, we specify our model and demonstrate its ability to reliably infer miR-mRNA interactions in an expression data set of samples taken from multiple myeloma patients in different stages of the disease. We use the *miRWalk* [Dweep et al. 2011] database of validated targets and compare our results with those obtained from *TaLasso*, as well as Pearson correlation, as a benchmark. Throughout, we illustrate how the natural partial ordering of the data can be used to improve interaction inference, particularly if we are concerned mainly with interactions specific to the progression of multiple myeloma development.

3.2 Methods

We have developed a Bayesian model of miR-mRNA interactions for matched expression data (*i.e.* we have both miR and mRNA expression data for each biological sample) that was designed specifically for partially ordered samples, where "partially ordered" refers to the case where every sample can be said to be "before" or "after" at least one other sample in the data set. The partial ordering could indeed be a time-course experiment (which is usually fully ordered, linearly in time) or it could comprise multiple branches of experimental development, such as a disease study wherein healthy and diseased samples—possibly originally all starting from the same healthy population—are collected over time, or in stages.

We have developed a model for partially ordered samples because prior work in using expression data to infer miR-mRNA targeting interactions has focused on methods that do not depend on the order of the samples, the most common of which is Pearson correlation. We find it both theoretically and practically desireable to consider an ordering of samples because in most cases we expect that samples whose sources are more similar—in this case by disease type or stage—should also have the most similar expression values. If we consider such a natural ordering, we should be more likely to infer significant targeting interactions that occur from one stage to the next but whose expression levels are not necessarily the most correlated throughout the entire data set.

Let us consider a simple example of how using an ordering of data can help infer interaction coefficients. Assume a fully-ordered data set of n stages $\boldsymbol{S} = \{s_1, \ldots, s_n\}$, and we have [correctly] inferred the mean log expression values x_s and y_s in stage s for a single miR x and mRNA y, which are perfectly negatively correlated and where each has been normalized to have mean zero and standard deviation of one. A simple model formulation for each y_s, without considering the ordering, could be

$$P(y_s \mid \beta, \lambda, x_1, \ldots, x_n) = \mathcal{N}(\beta x_s, \lambda) \tag{3.1}$$

for interaction coefficient β and precision (inverse variance) λ. If we use a non-informative but improper uniform prior distribution for β—$P(\beta) = \lim_{\theta \to 0} \mathcal{N}(0, \theta)$—then the variational Bayesian estimates for μ_β and κ_β are

$$\hat{\mu}_\beta = \sum_{s \in \{1,\ldots,n\}} \frac{y_s}{x_s} \tag{3.2}$$

$$\hat{\kappa}_\beta = \lambda \sum_{s \in \{1,\ldots,n\}} x_s^2 \tag{3.3}$$

and likewise if we consider the ordering of samples such that for $s \neq 1$,

$$P(y_s \mid \lambda, x_1, \ldots, x_n, \beta) = \mathcal{N}(y_{s-1} + \beta(x_s - x_{s-1}), \lambda) \tag{3.4}$$

the corresponding estimates are

$$\begin{aligned}
\hat{\mu}_\beta = &\sum_{s \in \{2,\ldots,n\}} \frac{y_s - y_{s-1}}{x_s - x_{s-1}} && (3.5)\\
\hat{\kappa}_\beta = &\ \lambda \sum_{s \in \{2,\ldots,n\}} (x_s - x_{s-1})^2 && (3.6)\\
= &\ \lambda \sum_{s \in \{1,\ldots,n\}} x_s^2 && (3.7)\\
&+ \lambda \sum_{s \in \{2,\ldots,n-1\}} x_s^2 && (3.8)\\
&- 2\lambda \sum_{s \in \{2,\ldots,n\}} x_s x_{s-1} && (3.9)
\end{aligned}$$

With the assumed perfect negative correlation and unit standard deviation, both estimates (3.2) and (3.5) for $\hat{\mu}_\beta$ give a [correct] value of -1. Also, equation (3.3) is the same as line (3.7); therefore lines (3.8) and (3.9) give the adjustments to the estimated precision of β in the ordered model version as compared to the standard version. If the sum of these is positive, the estimate for β given by the ordered model has a higher precision and thus is more statisically significant. This occurs, for example, in the simple case where $n = 4$, $\langle x_s \rangle = \langle x_1, x_2, x_3, x_4 \rangle = \langle -1, 1, 1, -1 \rangle$ and $\langle y_s \rangle = \langle 1, -1, -1, 1 \rangle$, where the ordered precision estimate is 8λ while the unordered one is 4λ. In contrast, if we take a different ordering of the same paired data, $\langle x_s \rangle = \langle 1, 1, -1, -1 \rangle$ and $\langle y_s \rangle = \langle -1, -1, 1, 1 \rangle$, both versions of the model give a precision estimate of 4λ. If one thinks sequentially about the data, it seems that in the former example, the miR and mRNA make two simultaneous but opposing expression changes, one between stages 1 and 2 and another between stages 3 and 4. In the latter example, only one such simultaneous change is made. In many cases of miR-mRNA interaction inference, it would be desireable to make a distinction between these two cases, particularly in experiments designed to measure stage-by-stage development of a process, where we might expect the expression levels of an miR to rise for some specific period of the process and then fall again.

We can also generalize a bit from these simple examples. Since the summation in (3.8) is an approximation of the variance of the $\{x_s\}$ (without the first and last stages), which we have normalized to 1, and the summation in (3.9) is the [lag 1] autocorrelation of the $\{x_s\}$, generally speaking, the ordered model gives higher precision for interaction coefficients when the autocorrelation of the normalized expression data is less than 0.5. This is not an exact rule since (3.8) does not include the first or last stages, but we can see that it does not take an unreasonable amount of expression variation between adjacent stages for the ordered model to give higher statistical significance for interaction coefficients, when such an ordering exists.

In addition the possibility of higher statistical significance in highly varying miR, considering the order of samples in an experiment allows us to detect positive or negative trends in expression value with respect to the process being investigated—a feature that may prove useful in identifying the main drivers of a developmental process such as disease, growth, aging, etc. The model also includes scores from existing prediction algorithms for miR-mRNA targeting to better determine the existence of a targeting interaction. In this paper, we have used data (including prediction scores) from the *TargetScan* [Lewis et al. 2005; Friedman et al. 2009; Grimson et al. 2007; Garcia et al. 2011] and *miRanda* [John et al. 2004; Enright et al. 2003] databases, but any exogenous, quantitative information about the putative target pair could be included.

Below, we define and fit our model to a previously published multiple myeloma data set using variational Bayesian methods. Then, we check our results against the *MiRWalk* [Dweep et al. 2011] database of experimentally-validated targeting interactions, as well as against rankings of predicted target pairs by most negative Pearson correlation coefficient.

3.2.1 Data

We demonstrate our model using the multiple myeloma data set from Lionetti et al. [2009], which can be downloaded from *GEO* [Barrett et al. 2009], accession number GSE17498. In this data set, there are both miR (Agilent-019118 Human miRNA Microarray 2.0) and mRNA (Affymetrix Human Genome U133A Array) expression values for samples from 36 patients, 34 of whom have been diagnosed with multiple myeloma (MM) and 2 of whom have been diagnosed with plasma cell leukemia (PCL). Unfortunately, from healthy donors there is only miR expression data and no mRNA data, so we cannot include healthy samples in our study. However, the diseased samples can be arranged into four Durie-Salmon stages: IA, IIA, IIIA, and IIIB, which gives an obvious ordering for our non-PCL samples. Because PCL is a condition related closely to, but not necessarily developing directly from (or progressing to) MM, we treat it as a separate branch of the partial ordering, as described in the subsequent section. However, within our partial-ordering framework it is necessary to specify some relation to other samples, and thus we assume that PCL follows the healthiest, most normal MM stage—in the absence of truly healthy samples—Durie-Salmon IA.

3.2.2 Partial ordering

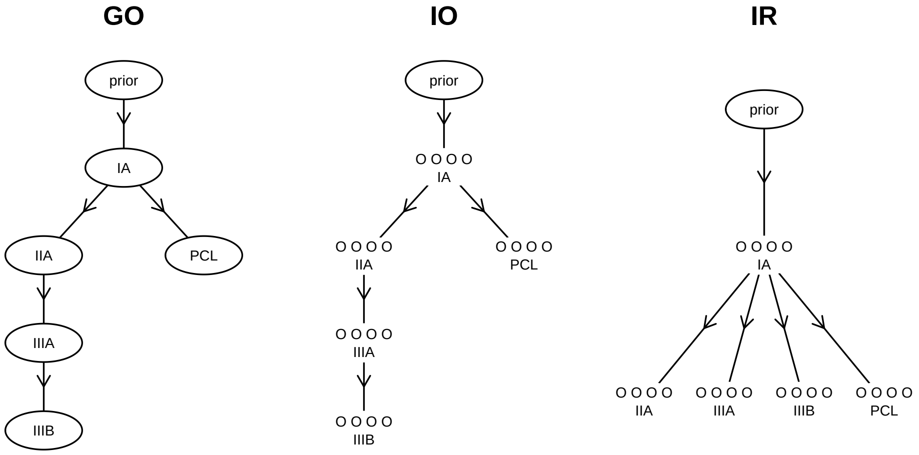

Figure 3.1: Partial orderings. The graphs above show the three different partial orderings of the data that we explore in this paper. Arrows give the direction of the ordering, where sample A can be said to *precede* sample B (i.e. $A < B$) if in the graph an arrow points from A to B. *G-O* refers to the grouped-ordered model version, in which samples of the same Durie-Salmon stage are grouped together as replicates. *I-O* is the individual-ordered model version, where the groups of smaller circles represent that different samples are not grouped as replicates, but the ordering of Durie-Salmon stages is the same as in the *G-O* ordering (i.e. $A < B$ if and only if the stage of A precedes the stage of B in the *G-O* ordering). And, *I-R* is the individual-reference model version, where the samples are again not grouped as replicates, but each of the Durie-Salmon stage IA samples precedes each sample of every other stage. In each partial ordering, there is a prior distribution over the samples which are not preceded by any other samples.

In our analyses, we compare three different partial orderings, which we show in Figure 3.1. The first is the natural ordering: the Durie-Salmon stages in order, plus PCL as a separate branch off of the initial stage, stage IA. This ordering treats patients with the same disease type and stage as replicates, and should give results that are specific to the disease itself since it effectively ignores the differences between individuals with the same disease type. We call this ordering the grouped-ordered (G-O) ordering.

The second partial ordering is the same as the first, but with the individuals separated. We call this the individual-ordered (I-O) ordering. In this version of the model, each individual comes after all of the individuals of the prior Durie-Salmon stage, and again the PCL samples are after all of the IA samples. This arrangement places less focus on the disease itself but allows the model to infer miR-mRNA interactions based on variations between individuals of the same type.

The third partial ordering considers the stage IA samples to be references, while all other samples come directly after them. Individuals are still considered separately. Hence, this is called the individual-reference (I-R) ordering. This reference-based design largely ignores the natural ordering of the data and focuses on differences between individuals. Comparing the results from this ordering with the results from the other orderings could indicate some of the advantages (and disadvantages) of considering the natural ordering of these samples.

3.2.3 Pre-processing

Prior to the main analysis, we performed quantile normalization across all arrays of the data set using the *limma* package for *R* [Smyth and Speed 2003; R Development Core Team 2009]. We then performed a probewise ANOVA to test for differential expression across the stages IA, IIA, IIIA, IIIB, and PCL (using individuals within a stage as replicates) and removed those probes/probesets (for both miR and mRNA) whose (unadjusted) p-value from the ANOVA F-test was greater than or equal to 0.05, as well as those miRs and mRNAs not involved in any predicted targeting interactions, leaving 28 miRs and 367 mRNAs as possible candidates for targeting interaction. Lastly, we re-scaled the data so that each probe/set—across all samples—had a mean of zero and a standard deviation of one.

3.2.4 Target prediction algorithms

For these analyses, we included miR-mRNA target prediction data from both *TargetScan* [Lewis et al. 2005; Friedman et al. 2009; Grimson et al. 2007; Garcia et al. 2011] and *MiRanda* [John et al. 2004; Enright et al. 2003]. For each of these, we downloaded from the corresponding web site a table of predicted miR-mRNA interactions and the targeting scores calculated by the respective algorithms. *TargetScan* includes a *context+* score and *MiRanda* includes a *mirSVR* score. [Betel et al. 2010] In our model, described below, we consider the predicted interactions and these prediction scores.

3.2.5 Model

Let us define a stage as a set of expression levels s that are, for each probe/set, to be considered replicates of each other. We assume a partial ordering O on the set of stages S such that each stage s_0 has a set of parent stages $\rho_{s0} = \{s \in S : s < s_0 \, in \, O\}$. The set of parent stages for s_0 can be defined as $\rho_{s0} = \{\text{“}prior\text{”}\}$ if s_0 has no parents and is therefore an initial stage in O (the existence of at least one initial stage is guaranteed by the acyclic property of partial orderings). Furthermore, let us define a “development” parameter $\delta_{\rho,s}$, which intuitively represents a kind of distance from a parent ρ to its child s, and which we include in the equations below. Likewise, let τ_i be the “trend” of probe/set i in either the positive or negative direction with respect to the ordering O, and let Λ_i be the precision (inverse of variance) parameter of the expression of probe/set i thoughout all stages.

miR parameters

Then, we assume the log expression value $\upsilon_{i,s}$ of each miR i in stage s, to be normally distributed with mean

$$\mu_{i,s} = \frac{\sum\limits_{\rho \in \rho_s} \frac{1}{\delta_{\rho,s}} \left(\upsilon_{i,\rho} + \delta_{\rho,s}\tau_i\right)}{\sum\limits_{\rho \in \rho_s} \frac{1}{\delta_{\rho,s}}} \tag{3.10}$$

and precision

$$\lambda_{i,s} = \frac{\sum\limits_{\rho \in \rho_s} \left(\frac{1}{\delta_{\rho,s}}\right)^2}{\sum\limits_{\rho \in \rho_s} \frac{1}{\delta_{\rho,s}}} \Lambda_i \tag{3.11}$$

Thus, the prior mean $\mu_{i,s}$ is the weighted sum of the parents' expression values with the developmental trend added (the trend τ_i multiplied by the development factor $\delta_{\rho,s}$). The weights in the weighted sum are the inverses of the developments $\delta_{\rho,s}$. Likewise, the prior precision is the weighted average of inverse developments multiplied by the probe's stage-wise precision parameter Λ_i. This formulation gives two parents equal weight in the prior distribution of a common child s if their development parameters to that child, $\delta_{\rho,s}$, are equal. Also, as $\delta_{\rho,s}$ increases for one parent to the child, the influence of its expression value on the child's prior distribution diminishes to zero.

Note that in the formula for the stage's prior precision $\lambda_{i,s}$, the probewise precision Λ_i is moderated by $\delta_{\rho,s}$, in that a parent stage ρ_s that is more similar to its child s—if it has a smaller value $\delta_{\rho,s}$—carries more weight and increases the precision in the prior distribution of the probes in stage s. This allows for varying developmental distances between stages, where larger $\delta_{\rho,s}$ imply that all probes experience lower precision (more random noise) between stages ρ and s, and vice-versa for smaller $\delta_{\rho,s}$.

mRNA parameters

The distributions we have assumed for mRNA expression are identical to those of the miR, except that the developmental trend component (the product $\delta_{\rho,s}\tau_i$) is exchanged for an interaction component with the miR expressions $\upsilon_{i,s}$. Let $\boldsymbol{\upsilon_s}$ be the vector of all miR expression values in the stage s, and let $\boldsymbol{\eta_j}$ be the vector of interaction coefficients between $\boldsymbol{\upsilon_s}$ and mRNA expression level $\omega_{j,s}$, then $\boldsymbol{\eta_j} \bullet (\boldsymbol{\upsilon_s} - \boldsymbol{\upsilon_\rho})$—where $\bullet$ is the standard vector dot product—expresses the total interaction effect of all miRs on the mRNA j from stage ρ to stage s.

Specifically, we assume the log expression value $\omega_{j,s}$ of each mRNA j in stage s, to be normally distributed with mean

$$\mu_{j,s} = \frac{\sum\limits_{\rho \in \rho_s} \frac{1}{\Delta_{\rho,s}} \left(\omega_{j,\rho} + \boldsymbol{\eta_j} \bullet (\boldsymbol{\upsilon_s} - \boldsymbol{\upsilon_\rho}) \right)}{\sum\limits_{\rho \in \rho_s} \frac{1}{\Delta_{\rho,s}}} \tag{3.12}$$

and precision

$$\lambda_{j,s} = \frac{\sum\limits_{\rho \in \rho_s} \left(\frac{1}{\Delta_{\rho,s}} \right)^2}{\sum\limits_{\rho \in \rho_s} \frac{1}{\Delta_{\rho,s}}} \Lambda_j \tag{3.13}$$

where the $\Delta_{\rho,s}$ are analagous to, but distinct from, the $\delta_{\rho,s}$ we used in the miR distributions. Likewise, the Λ_j are analagous to the Λ_i from the miR distributions, but are inferred separately for each mRNA j, as they are for each miR i.

Technical and replicate variance

We assume two technical precisions (inverse variances) in our model. One precision corresponds to an expression set (i.e. the precision/variance between microarrays from the same stage) and one corresponds to replicates within one expression set (i.e. multiple spots for the same probe/set or transcript within a microarray).

For the miR and mRNA expression levels, υ_i and ω_j, above, we assume that the expression levels $\varepsilon_{i,s}$ and $\varepsilon_{j,s}$ each probe i (miR) or probeset j (mRNA) in an expression set m from stage s is normally distributed as

$$\varepsilon_{i,m,s} \sim \mathcal{N}(\upsilon_{i,s}, \kappa_{miR}) \tag{3.14}$$

or

$$\varepsilon_{j,m,s} \sim \mathcal{N}(\omega_{i,s}, \kappa_{mRNA}) \tag{3.15}$$

where the second parameter $\kappa_{\square}$ in the normal distribution $\mathcal{N}()$ is a precision, not a variance or standard deviation.

Furthermore, within each expression set m, we assume that the expression data $x_{i,m,n,s}$ (miR) and $y_{j,m,n,s}$ (mRNA) for within-set replicate n and stage s are normally distributed as

$$x_{i,m,n,s} \sim \mathcal{N}(\varepsilon_{i,m,s}, \kappa'_{miR}) \tag{3.16}$$

or

$$y_{j,m,n,s} \sim \mathcal{N}(\varepsilon_{j,m,s}, \kappa'_{mRNA}) \tag{3.17}$$

for the two second-level technical precisions $\kappa'_{\square}$.

Interaction parameters

Each element $\eta_{i,j}$ of the vector of interaction coefficients $\boldsymbol{\eta_j}$ metioned above is also normally distributed as

$$\eta_{i,j} \sim \mathcal{N}(\boldsymbol{\beta} \bullet \boldsymbol{P}, \varphi) \tag{3.18}$$

where $\boldsymbol{P}$ is the vector of fixed parameters from target prediction algorithms, $\boldsymbol{\beta}$ is a vector of estimated coefficients, and φ is again a precision. Note that there is no restriction of the interaction coefficients $\eta_{i,j}$ to only negative values, as in some models, as we choose to allow for all regulatory effects, positive or negative, direct or indirect.

If one of the included algorithms predicts that miR i targets mRNA j, we include the vector $\langle 1, \alpha_{i,j} \rangle$ where $\alpha_{i,j}$ is the prediction score from the algorithm. We concatenate the vectors from multiple prediction algorithms such that, if a pair $\{i, j\}$ is predicted by more than one algorithm, the vector $\boldsymbol{P}$ is of the form $\langle 1, \alpha_{i,j}, 1, \alpha'_{i,j} \rangle$. In this way, the coefficients of $\boldsymbol{\beta}$ that correspond to a 1 in $\boldsymbol{P}$ determine the effect that inclusion on a particular list of predicted targets has on the expression data (indirectly through estimation of $\eta_{i,j}$), while the coefficients corresponding to algorithm scores may further refine the value of the algorithm in this model. (Also, an algorithm score of zero does not necessarily indicate zero chance of targeting, and thus if we include the scores, we must also include a constant.)

Prior distributions

We chose conjugate prior distributions for each model parameter that required a prior. Thus, we use vaguely informative normal distributions on the parameters $\upsilon_{i,0}$, $\omega_{j,0}$, and τ_i . Specifically, they all follow the distribution $\mathcal{N}(0, 10^{-10})$. Similarly, the prior distribution for $\boldsymbol{\beta}$ is the equivalent multivariate normal with zero mean and precision matrix $10^{-10}\mathbf{I}$, where $\mathbf{I}$ is the identity matrix of the appropriate size. We use vaguely informative gamma prior distributions on the parameters Λ_i, Λ_j, κ_{miR}, κ_{mRNA}, κ'_{miR}, κ'_{mRNA}, and φ.

The development parameters $\delta_{\rho,s}$ and $\Delta_{\rho,s}$ are special cases. Foremost, they have no obvious conjugate priors, and as we have defined this model, their optimal values are not unique since, for example, doubling the estimated values for $\delta_{\rho,s}$ and $\Delta_{\rho,s}$ along with the τ_i would give an identical likelihood, if priors are ignored. However, we are not concerned with the specific values of these parameters; we need only their values relative to each other. Thus, in order to obtain unique optimal values, we specify a gamma prior on the $\delta_{\rho,s}$ and $\Delta_{\rho,s}$ with shape and rate (i.e. inverse-scale) set equal to 1.

3.2.6 Fitting the model using variational Bayesian methods

To estimate the parameters of our model, we use variational Bayesian methods. These methods are closely related to expectation-maximization algorithms [Welling and Kurihara 2006] and have been used previously in discovering miR-mRNA target pairs [Huang et al. 2007b,a, 2008] as well as other analyses of gene expression data. [Beal et al. 2005; Teschendorff et al. 2005]

In short, variational Bayesian methods find a probability distribution $\prod_i Q(\theta_i)$ (factorizable over all parameters) that is increasingly similar (via iterative updates) to the desired posterior distribution $P(\boldsymbol{\theta} \mid \boldsymbol{X})$ for model parameters $\theta_i \in \boldsymbol{\theta}$ and data $\boldsymbol{X}$, through use of the Kullback-Leibler divergence as a measure of dissimilarity. As part of the calculations, one also obtains a lower bound $\mathcal{L}(Q)$ to the *evidence* $P(\boldsymbol{X})$, which can be helpful in judging the goodness of fit of alterative model formulations. For a thorough explanation of variational Bayesian methods, see Winn [2003] or Beal [2003].

The result of variational Bayesian calculations is, like with most Bayesian methods, a set of estimated posterior probability distributions over the model parameters. Unlike Markov chain Monte Carlo and related methods, we need not worry much about convergence of the estimated parameter distributions, since, if implemented properly, a variational Bayesian algorithm guarantees an improvement in every iterative update. Of course, calculations can be quite slow when compared to non-Bayesian methods.

All parameters were estimated using variational Bayesian (VB) methods, with the exception of the $\delta_{\rho,s}$ and $\Delta_{\rho,s}$, since they have no simple conjugate distributions. Specifically, we treat the $\delta_{\rho,s}$ and $\Delta_{\rho,s}$ as fixed values while iteratively updating the other parameters, and then we update these by maximizing the lower bound $\mathcal{L}(Q)$ over their possible value range. More specifically, after first initializing the variables to reasonable values, the algorithm sequentially updates the posterior distribution estimate for each variable (excepting $\delta_{\rho,s}$ and $\Delta_{\rho,s}$) and that posterior estimate (of the same form as the conjugate prior) is utilized in all subsequent steps. After all such posterior estimates have been updated, $\delta_{\rho,s}$ and $\Delta_{\rho,s}$ are optimized by finding the maximum likelihood estimates. Then, once again, the posterior distribution estimates are updated for all other parameters, and so on. This process is repeated until the parameter values change very little with each subsequent iteration and thus it becomes no longer beneficial to continue updates. We found that 200 such iterations gave sufficient results.

This algorithm was coded in the *R* [R Development Core Team 2009] statistical programming language.

3.3 Results

We applied our model to the multiple myeloma data set from Lionetti et al. [2009] in three different configurations based on our choice of partial orderings (shown in Figure 3.1). For these analyses, we consider only those interactions which were predicted by at least one prediction algorithm (*TargetScan* or *miRanda*), but we would like to note that this is not necessary in our framework. We have limited the set of candidate interactions in this way because the high number of possible parameters in the model (all possible interactions) can be significantly reduced by considering only predicted interactions. Furthermore, target predictions are known to have significant significant sensitivity [Sethupathy et al. 2006], when compared to validated targets, even if they have unknown specificity since a complete list of true targets does not exist. Thus, we attempt to rank only the 1754 interactions predicted between the 28 miRs and 367 mRNAs, ensuring that our top candidate interactions have been predicted as well as supported by expression data.

Below, we evaluate the results and compare them with the interactions rankings one obtains from *TaLasso* as well as by ranking by Pearson correlation coefficient, as a simple benchmark. For this, we consider both the strongest absolute value correlations as well as the strongest negative correlation, as much evidence indicates that miR-mRNA interactions are predominantly negative, and thus ranking by most negative correlation generally improves results. [Muniategui et al. 2013] In each of the rankings based on our model, all of the top 900+ inferred interactions were negative; thus, restricting only to negative interactions has no effect on these results.

First, we checked the *miRWalk* database for target pairs that have been experimentally validated and we looked at their ranking according to each of the methods. Second, we looked for enrichment of *KEGG* pathway [Kanehisa et al. 2008] annotation among mRNAs involved in the top 100 targeting interactions on our lists. To do this, we used the "singular annotations" from the *GeneCoDis* web tool. [Carmona-Saez et al. 2007; Nogales-Cadenas et al. 2009] Then, we examined more closely specific target pairs near the very top of our rankings. Lastly, we looked at the miRs with the strongest trends through the stages of multiple myeloma according to the *G-O* ordering, which we hypothesize indicates a high likelihood of playing a role in the development of the disease.

3.3.1 Interaction validation by *miRWalk*

Among our data set, only five putative miR-mRNA targeting interactions have been validated, according to the validation database *miRWalk* [Dweep et al. 2011], though 13 more target pairs have been validated despite not being predicted by either *TargetScan* or *miRanda*. This may indicate that heavily favoring predicted pairs over non-predicted pairs detracts from the results more than expected. However, since one of our main goals here was to combine prediction data and expression data, we do not address the issue here.

All five of the validated, predicted target pairs involve the well-studied miR-17. These five interactions appear at positions 63, 229, 234, 273, and 612 on our interaction ranking based on the *G-O* ordering from Figure 3.1. This is considerably better than in the ranking by absolute value Pearson correlation (341, 402, 819, 877, and 893) and also by most negative Pearson correlation (162, 195, 468, 568, and 604). The *TaLasso* results gave rankings for only four of these five validated target pairs, as the *TargetScan*-predicted pair {hsa-miR-17, PKD1} seems to be missing; perhaps the results list was truncated or the pair was missing from the built-in list of predictions. However, the remaining four interactions are ranked 311, 351, 846, and 952, which is comparable to Pearson correlation.

If we divide our ranking positions, in increasing order, by the rankings by correlation (*e.g.* we divide 63 by 341, 229 by 402, and so on), we find that our rankings are, on average, 0.41 times those by absolute value correlation and 0.71 those by negative correlation. Repeating this using the four rankings from *TaLasso* and the top four from our model gives 0.35. This can be interpreted as an estimate of the relative number of target pairs that would need to be experimentally tested based on each ranking in order to arrive at the same number of positive validations, and from now on we will refer to this statistic as the "average relative rank statistic". If we consider our ranking using the partial ordering *I-O* from Figure 3.1, we obtain average relative rank statistics of 0.39 and 0.56 when compared to rankings by absolute value correlation and negative correlation, respectively. Partial ordering *I-R* gives average relative rank statistics nearly identical to these.

Though there are too few existing validations for us to draw strong conclusions, the fact that the rankings of these by our model are much closer to the top of the list than those by the correlation (negative or absolute value) indicates that there is at least some advantage to our partially ordered model.

3.3.2 KEGG pathway enrichment

We show in Table 3.1 all of the KEGG pathways that are enriched (FDR corrected $p < 0.05$) in the top-100 list for at least one of the four models (including ranking by negative correlation among predicted target pairs). There are considerably more pathways with enrichment among the lists generated by our model than by negative Pearson correlation. Specifically, six types of cancer appear in the lists for our models, while there are none for the correlation list. This is promising, as this data set comes from a cancer study, specifically of multiple myeloma. In total, among the 41 genes involved in the top 100 interactions inferred by the *G-O* ordering, there were 13 enriched pathways, and only two among the 56 genes in the top 100 interactions according to ranking by negative correlation.

Amongst the three partial orderings we have considered here, there is marginally more pathway enrichment and fewer genes involved in the top 100 targeting interactions for the *G-O* ordering, though both the *I-O* and *I-R* orderings both give much more enrichment than the rankings by *TaLasso* and negative correlation.

3.3.3 Top candidate interactions

Ultimately, our goal with this analysis is to enable the identification of the most promising candidates for further biological investigation. In Figure 3.2 we show the top ten interactions inferred by the model using the *G-O* ordering. The three strongest inferred interactions involve the NR3C1 glucocorticoid receptor, which first appears in the 109th interaction on a ranking of interactions by negative correlation. Myeloma patients with low expression of this receptor respond poorly to standard treatment with dexamethasone and have a poor overall prognosis, making this molecule an intrinsically interesting candidate for further investigation. [Heuck et al. 2012] Two of the miRs inferred as targeting this gene, miR-18a and miR-18b (part of the 5th and 6th ranked interactions by negative correlation), share a seed sequence, and are associated with the miR-17∼92 cluster—a downstream target of the c-myc oncogene. [O'Donnell et al. 2005] This cluster is well-known to play a role in cancer development as well as normal lymphoid development, and has recently been associated with tumorgenicity in multiple myeloma. [Chen et al. 2011] The next strongest inferred interaction involves the

	G-O	I-O	I-R	TaLasso	Neg.Cor
Number of unique genes in the top 100 interactions	41	58	53	85	56
05200 :Pathways in cancer	3				
05215 :Prostate cancer	2	3	3		
05219 :Bladder cancer	2	2	2		
05222 :Small cell lung cancer	2		2		
05216 :Thyroid cancer	2				
05214 :Glioma		2	2		
05218 :Melanoma		2	2		
05016 :Huntington's disease	3	4	3		
05014 :Amyotrophic lateral sclerosis (ALS)	2	2	2		
05010 :Alzheimer's disease		3			
04976 :Bile secretion		2			
04730 :Long-term depression			2		
04115 :p53 signaling pathway	2	3	4		
04210 :Apoptosis	2	2	2		
04010 :MAPK signaling pathway	3		3		
04722 :Neurotrophin signaling pathway	2				
04110 :Cell cycle	2				
04120 :Ubiquitin mediated proteolysis	2			4	
04622 :RIG-I-like receptor signaling pathway		2	2		
04144 :Endocytosis				4	
04914 :Progesterone-mediated oocyte maturation				3	
04114 :Oocyte meiosis				3	
04142 :Lysosome				3	
03060 :Protein export					3
04141 :Protein processing in endoplasmic reticulum					4

Table 3.1: Enriched KEGG pathways among genes in the top 100 interactions. The top row gives the number of unique genes present in the top 100 miR-mRNA interactions according to each model; the remaining rows give, per column, the number of these genes annotated by *KEGG* pathway terms with significant enrichment (FDR corrected $p < 0.05$) for for at least one of the models proposed. A blank entry indicates that the particular pathway was not significantly enriched in the model. The column *G-O* refers to the grouped-ordered model version, *I-O* is the individual-ordered model version, and *I-R* is the individual-reference model version, while *Neg.Cor* is the ranking by most negative Pearson correlation (between miR and mRNA expression profiles) among the predicted target pairs. The horizontal lines separate general categories of *KEGG* pathways, namely cancer-related pathways, other disease-related pathways, and then remaining pathways found to be enriched by at least one of the models.

gene UBE2D3, (targeted by miR-891b) which is a ubiquitin-conjugating enzyme known to be involved in p53 ubiquitination. [Tokumoto et al. 2011] The next ranked interaction on our list involves the p53 tumor-suppressor (TP53)—an extremely important gene in most, if not all, cancer types—inferred to be targeted by miR-let-7e. Unlike in many cancers, at diagnosis in multiple myeloma, p53 is rarely seen to be mutated or deleted. As it is not changed at the genomic level, it is therefore quite plausible that p53 may be manipulated at the level of translation by miR in this disease, making this pair an intriguing candidate interaction as well.

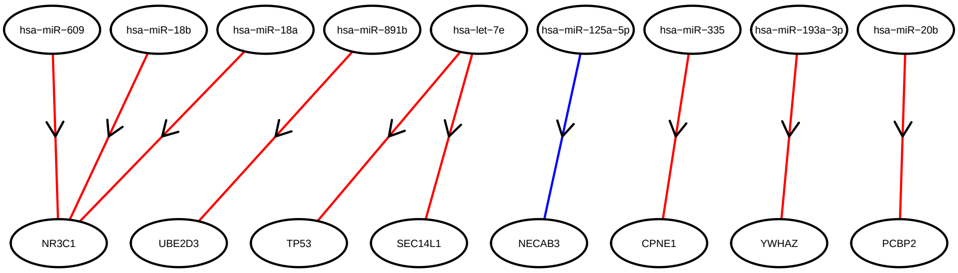

Figure 3.2: The top 10 interactions according to the *G-O* ordering. In the above diagram, we show the miRs (top row) and genes (bottom row) involved in the 10 most significant targeting interactions based on the *G-O* ordering from Figure 3.1. In each case, the inferred interaction is negative, meaning that the miR inhibits the expression of the corresponding gene. A red line from an miR to an mRNA indicates that the interaction was predicted by *TargetScan* and a blue line indicates that the interaction was predicted by *miRanda*.

3.3.4 Inferred miR regulators in multiple myeloma development

The inclusion of trend parameter τ_i for each miR in our model allows us to identify miRs whose expression levels increase or decrease significantly over the progression of stages with respect to the partial ordering. Table 3.2 shows the miRs with a corresponding trend parameter τ_i estimate whose posterior mean is more than three standard deviations (based on the posterior precision estimate) from zero, in either direction, positive or negative.

Top candidates from the table include miR-18a and miR-18b, which, as discussed above, are well-known to play a role in cancer development. Both of these showed increased expression in advanced stages of multiple myeloma. Another candidate is miR-194, which has been shown to be p53-dependent and a positive regulator of this well-known tumor-supressor, creating a positive feedback loop. Furthermore, down-regulation of miR-194 has been demonstrated to play a key role in multiple myeloma development through its modulation of p53 signaling. [Pichiorri et al. 2010] Our model inferred a significant positive trend for miR-194, which might be contrary to this prior expectation of down-regulation, but in any case adds to the evidence that miR-194 is involved—perhaps in a complex way—in the development of multiple myeloma.

trending miR	direction	z-score
miR-18b	+	3.88
miR-367	-	3.80
miR-18a	+	3.61
miR-194	+	3.57
miR-133b	+	3.54
miR-92a	+	3.38
miR-554	-	3.24
miR-551a	+	3.23

Table 3.2: miRs with a significant estimate for the trend parameter. Shown are the most significant trend parameter estimates (τ_i). A "+" in the table denotes that the expression of the miR increased throughout the progression of the partial ordering by disease stage (*G-O*), and tended to be higher during later stages. Likewise, a "-" denotes that the expression of that miR tended to be higher in early stages and lower in later stages.

3.4 Discussion

Combining miR-mRNA target prediction algorithms with expression data has proven to be one of the best strategies for high-throughput target pair inference. However, the exact way in which do this has been the subject of some discussion. Though many methods have addressed specific issues in target inference, and others have attempted a more general approach, none has fully addressed ordered and partially ordered data sets. We tried three different partial orderings in our model, as shown in Figure 3.1. They performed similarly to each other, but not quite the same, supporting the conclusion that the ordering does make a difference, and thus should be carefully considered before analysis. Grouping and ordering samples by disease stage seems to have enriched, if only slightly, the top target interactions according to our *KEGG* anaylsis.

As illustrated in the *Methods* section, the order of samples (if one exists) can affect the strength of inference of correlated miR and mRNA expression patterns, and in fact this additional statistical power can be seen in a very simple example. The model that we present in this paper addresses partially ordered data sets by assuming that closely related samples (with respect to the ordering) should be more similar than less closely related samples. This assumption allows the model to outperform *TaLasso* and Pearson correlation (*i.e.* ranking of target pairs by most negative correlation) by a noticeable margin, aided by the Bayesian framework that inherently places more weight on measurements and variables that have high certainty or precision. Our model's rankings of the few previously experimentally validated target pairs were significantly better in our model, and KEGG pathways were significantly more enriched, particularly for cancer-related pathways, which we would expect from this data set.

Both the mRNA targets and targeting miRs from our top-ranked interactions have been previously implicated in multiple myeloma development, suggesting that our analysis has successfully identified biologically-relevant pairs from this data set. Furthermore, some of the miRs that we have identified as having a significant trend through the ordering of the stages have been verified by literature as key players in both cancer and, more specifically, multiple myeloma. The remaining un-verified top interactions and trending miRs may be good candidates for further investigation.

Interestingly, though we didn't limit our interactions to be non-positive, virtually all of the top 1000 interactions were negative. This is likely an effect of utilizing the the prediction algorithms in the prior distributions for the interaction parameters, since our model estimates coefficients for the inclusion in (and targeting score of) each included prediction algorithm. It is well known that miRs typically down-regulate target mRNAs, and though there have been some reports of up-regulation, we would expect the estimated coefficients for predicted targets would lead to a negative prior distribution (see equation 3.18) on the majority of interactions, if not all of them.

One potential weakness of our model—which is shared by virtually all recent models of miR-mRNA targeting—is that we attempt to explain all changes in mRNA expression using miR targeting interaction coefficients. This assumption that miR targeting should account for all gene expression changes is patently untrue. There are other direct processes—involving transcription factors, for instance—as well as indirect processes that can affect mRNA expression. Though it would be quite cumbersome in both data and calculation, an expanded model taking into account other potential influences could prove very useful in inferring true interactions between the various nucleic acids, proteins, etc.

Lastly, though much literature has been published on the topic, we have a lot to learn about high-throughput inference of miR-mRNA target pairs. Experimental validations are so sparse that it is impossible to prove conclusively which prediction or inference techniques routinely give the best results, and in which cases each is most appropriate. Perhaps in the near future we will see a vast increase in the number of targets being validated, possibly through cooperation or organization between research groups to create more complete databases (both of positive and negative results) with which we can compare inference approaches to further refine our methods and in turn more efficiently focus our experimental efforts into the most promising areas.

Chapter 4

Discovery of miR-mRNA interactions via simultaneous Bayesian inference of gene networks and clusters using sequence-based predictions and expression data

Published in the *Journal of Integrative Bioinformatics* [Godsey 2013a]

by Brian Godsey[1]

1 Department of Statistics and Probability Theory, Vienna University of Technology, 1040 Vienna, Austria

Abstract: MicroRNAs (miRs) are known to interfere with mRNA expression, and much work has been put into predicting and inferring miR-mRNA interactions. Both sequence-based interaction predictions as well as interaction inference based on expression data have been proven somewhat successful; furthermore, models that combine the two methods have had even more success. In this paper, I further refine and enrich the methods of miR-mRNA interaction discovery by integrating a Bayesian clustering algorithm into a model of prediction-enhanced miR-mRNA target inference, creating an algorithm called *PEACOAT*, which is written in the *R* language. I show that *PEACOAT* improves the inference of miR-mRNA target interactions using both simulated data and a data set of microarrays from samples of multiple myeloma patients. In simulated networks of 25 miRs and mRNAs, our methods using clustering can improve inference in roughly two-thirds of cases, and in the multiple myeloma data set, KEGG pathway enrichment was found to be more significant with clustering than without. Our findings are consistent with previous work in clustering of non-miR genetic networks and indicate that there could be a significant advantage to clustering of miR and mRNA expression data as a part of interaction inference.

4.1 Introduction

It is known that microRNAs (miRs) can interfere with mRNA expression, potentially regulating a large number of critical biological processes; for comprehensive information and specific examples of ways that miRs affect biological processes, visit `mirbase.org`. [Kozomara and Griffiths-Jones 2011; Griffiths-Jones et al. 2008, 2006] Though many miR-mRNA interactions have been investigated and validated, the vast majority of such interactions is yet undiscovered. [Gäken et al. 2012] Therefore, considering the potential importance and lack of knowledge about miR interference, much effort has been put into both predicting interactions based on the short (≈22nt) sequence length of miRs as well as inferring interactions from expression data; an overview can be found in Muniategui et al. [2013]. Though the former approach addresses issues particular to short sequences, the latter approach is closely related to other interaction models, particularly genetic interaction models, for which there exists even more research and literature than for miR-mRNA interaction models. A recent review enumerates many state-of-the-art methods for genetic interaction modeling: Penfold and Wild [2011]. The primary difference between miR-mRNA interaction models and the canonical genetic interaction models is that, in genetic models, the set of potential regulators and the set of potential regulatees are one and the same, whereas in miR-mRNA models it is assumed that some miRs regulate some mRNAs, and no other interactions exist. Despite this notable difference, when designing models of miR-mRNA interactions, there is a lot to be learned from genetic interaction models of varying types.

Two prominent miR-mRNA interaction models, *TaLasso* [Muniategui et al. 2012] and *GenMir* [Huang et al. 2007b], utilize Lasso regression and Bayesian networks, respectively, both of which are also used in genetic interaction models [Beal et al. 2005; Lèbre 2009; Gustafsson et al. 2005; Fujita et al. 2007]. One popular method of inferring genetic interactions, the Dynamic Bayesian Network (DBN), applies only to time-series expression data, but has proven to be quite useful in inferring interactions. [Beal et al. 2005; Lèbre 2009] It has been shown in Godsey et al. [2012] that aspects of a DBN can be applied to miR-mRNA interaction models if paired expression data (i.e. miR and mRNA expression data from the same biological samples or groups) are available, given a partial ordering of samples or groups. The requirement of a partial ordering is much more flexible than in a traditional DBN, which requires a total ordering, and in addition often requires the size of each step (i.e. time elapsed) between stages to be equal. The model from Godsey et al. [2012] requires neither total ordering nor equal step size, a flexibility which is enabled by the mutual exclusivity of the regulator set from the regulatee set.

Another notable notion from genetic interaction models that has yet to be fully utilized in miR-mRNA interaction models is the idea of a regulatory cluster, and likewise a regulated cluster. Though miR clusters and clusters of miR-regulated mRNAs have indeed been investigated [Tanzer and Stadler 2004; Wang et al. 2009], the idea has not been incorporated into probabilistic models using paired expression data.

I have recently shown that, in genetic DBN models, clustering of genes by their expression profiles improves interaction inference by reducing the interaction parameter space as well as the uncertainty arising from highly correlated potential regulators. Godsey [2013b] To summarize, it is difficult to determine the true regulator if two (or more) potential regulators are highly correlated, and this high inferential uncertainty can cause both potential regulators (together with their common regulatee) to fall far down the list of top-ranked inferred interactions; thus, it is better to group highly correlated regulators together and allow all members of the group to maintain a top ranking than to allow competition to diminish the inferred contribution of all members. In this paper, I expand upon this idea and adapt it for miR-mRNA interaction models. More specifically, clustering components are included in a new, updated version of the sequential miR-mRNA interaction model presented in Godsey et al. [2012]. The resulting model and algorithm are called *PEACOAT*: a Prediction-Enhanced Algorithm for Clustered, Ordered Assessment of Targeting in miR-mRNA interactions. *PEACOAT* not only offers the ability to infer miR-mRNA interactions as well as cluster miRs and/or mRNAs, but can also incorporate arbitrarily many miR-mRNA target predictions and prediction scores, and it allows the user to place an arbitrarily high or low amount of weight on such prediction information.

PEACOAT is tested on simulated networks of different sizes, and both with and without the use of prediction information. It is shown that clustering is frequently advantageous in the inference of true miR-mRNA interactions, and that the inclusion of helpful prediction information is likewise advantageous. The paper analyzes when and how to use clustering and predictions, and then applies what is learned to a data set of miR and mRNA expression from multiple myeloma patients, comparing the results to those of the sequential model from Godsey et al. [2012] and *TaLasso*.

4.2 Methods

This paper describes a new Bayesian model of miR-mRNA interaction that adds substantial capabilities to a simpler model presented in Godsey et al. [2012]. That model was specifically designed to infer miR-mRNA interactions in partially ordered expression data, while making use of target prediction databases/algorithms. Here, an improved version of this model is presented and, in addition, it is combined it with a Bayesian clustering model, allowing us to reduce the dimension of the interaction space without reducing the data set in a manner that has been shown in genetic interaction models to infer interactions more reliably when correlation between interacting elements might be high Godsey [2013b].

Therefore, while this model requires paired miR-mRNA expression data and a partial ordering of samples or groups of samples, the partial ordering can be made to resemble a reference design, or indeed any other design of choice, as long as every sample is designated to precede or follow at least one other sample in the partial ordering. However, this model's strengths lie in inferring interactions in inherently partially ordered data (e.g. development stages, time series, etc.) where correlation between regulatory actors (i.e. miRs) is often high.

4.2.1 Model specifications

In short, the model is formulated as in Godsey et al. [2012], with two notable modifications as well as the incorporation of clustering. The first exception is the specified partial ordering of sample stages only to the mRNA and not to the miR expression values. Instead of prior distributions based on a stage's specified parent stage(s), it is assumed that the miR expression values in each stage are normally distributed from the same prior distribution. This gives a lower probabilistic penalty to large changes in expression between consecutive stages and better enables proper fitting of miR parameters across all stages. The second notable change from the model in Godsey et al. [2012] is that the two model precisions within the interaction parameters are tied together so that we may *a priori* specify the level of influence the target predictions have on the inferred interactions, when compared to the influence of the expression data. The model in Godsey et al. [2012] typically, through optimization of parameters and priors, placed high emphasis on the target predictions, but this is not always desireable and the new model includes a fixed parameter allowing the adjustment of this influence; more details can be found below.

miR and mRNA expression parameters

For a set of partially-ordered stages such that each stage s_0 has a set of parent stages $\rho_{s0} = \{s \in S : s < s_0\}$ according to the partial ordering, and given that there are N_{miR} total miRs, N_m total mRNAs, K_{miR} miR clusters, and K_m mRNA clusters, each miR cluster expression value $F^{miR}_{k,s}$, for cluster k and stage s, follows the distribution

$$F^{miR}_{k,s} \sim \mathcal{N}(0, \lambda_F) \tag{4.1}$$

where λ_F is a scalar precision. By not considering the partial ordering among the stages for miR expression, both the development and trend parameters included in Godsey et al. [2012] cannot be included, but we are left with a slightly simpler, more focused model.

The cluster expression values $F^m_{k,s}$ for mRNA cluster k and stage s are modeled in the same manner as individual mRNA expression was in Godsey et al. [2012]. That is, given a scalar development parameter $\Delta_{\rho,s}$ (intuitively a distance) from a parent stage ρ to its child s, the $F^m_{k,s}$ are assumed to be normally distributed with mean

$$\mu_{k,s} = \frac{\sum\limits_{\rho \in \rho_s} \frac{1}{\Delta_{\rho,s}} \left(F^m_{k,\rho} + \boldsymbol{\eta_k} \bullet (\boldsymbol{F^{miR}_s} - \boldsymbol{F^{miR}_\rho}) \right)}{\sum\limits_{\rho \in \rho_s} \frac{1}{\Delta_{\rho,s}}} \tag{4.2}$$

and precision

$$\lambda_{k,s} = \frac{\sum\limits_{\rho \in \rho_s} \left(\frac{1}{\Delta_{\rho,s}} \right)^2}{\sum\limits_{\rho \in \rho_s} \frac{1}{\Delta_{\rho,s}}} \Lambda_k \tag{4.3}$$

where the vector $\boldsymbol{\eta_k}$ contains interaction coefficients between mRNA cluster k and all miR clusters, the vector $\boldsymbol{F^{miR}_s}$ contains the values $F^{miR}_{s,l}$ for all miR clusters l, and Λ_k is a precision parameter of the expression of cluster k.

For each miR or mRNA i, it is assumed that i belongs to exactly one cluster k (either a miR cluster or an mRNA cluster), and that its expression $\omega^{\Xi}_{i,s}$—where Ξ is either "miR" or "m" (short for mRNA)—is normally distributed about the cluster expression $F^{\Xi}_{k,s}$, as in

$$\omega^{\Xi}_{i,s} \sim \mathcal{N}(F^{\Xi}_{k,s}, \gamma_{k,s}) \tag{4.4}$$

It is also assumed that the mean expression $\varepsilon^{\Xi}_{i,s,m}$ of miR or mRNA i from a given expression set m (e.g. the mean of repeated spots on microarray m) in stage s is normally distributed as

$$\varepsilon^{\Xi}_{i,s,m} \sim \mathcal{N}(\omega^{\Xi}_{i,s}, \kappa_{\Xi}) \tag{4.5}$$

Furthermore, within each expression set m, it is assumed that the expression data $x^{\Xi}_{i,s,m,n}$ for within-set replicate n (e.g. the n^{th} replicate spot on a microarray) and stage s are normally distributed as

$$x^{\Xi}_{i,s,m,n} \sim \mathcal{N}(\varepsilon^{\Xi}_{i,s,m}, \kappa'_{\Xi}) \tag{4.6}$$

where again Ξ is either “miR” or “m” (“mRNA”).

Interaction parameters

For miR cluster l and mRNA cluster k, each element $\eta_{k,l}$ of the vector of interaction coefficients $\boldsymbol{\eta_k}$ mentioned above is normally distributed according to

$$\eta_{k,l} \sim \mathcal{N}\left(\sum_{i \in l, j \in k} \Theta_{i,j},\, \varphi \right) \tag{4.7}$$

for miRs i and mRNAs j, so that the cluster interaction parameter $\eta_{k,l}$ is assumed to be the sum of all individual miR-mRNA interaction parameters $\Theta_{i,j}$ such that miR i is in miR cluster l and mRNA j is in mRNA cluster k. These individual miR-mRNA interaction parameters $\Theta_{i,j}$ are assumed to be distributed as in Godsey et al. [2012], but with the addition of the prediction weight value $\Upsilon \in (0, \infty)$, which is multiplied by the same precision parameter φ as above, as in

$$\Theta_{i,j} \sim \mathcal{N}(\boldsymbol{\beta} \bullet \boldsymbol{P}_{i,j}, \Upsilon\varphi) \tag{4.8}$$

allowing arbitrary weight to be placed on the targeting predictions relative to the $\eta_{k,l}$. The vector $\boldsymbol{P}_{i,j}$ contains fixed parameters concerning miR i and mRNA j from target prediction algorithms, and $\boldsymbol{\beta}$ is a vector of estimated coefficients.

Prior distributions

Conjugate prior distributions are chosen, where possible. Thus, normal prior distributions on the parameters $F^{miR}_{i,s}$ and $F^{m}_{i,0}$ were chosen, as well as gamma prior distributions on Λ_j, $\gamma_{k,s}$, κ_{miR}, κ_{mRNA}, κ'_{miR}, κ'_{mRNA}, and φ. The prior distribution for $\boldsymbol{\beta}$ is the equivalent multivariate normal with zero mean and precision matrix $10^{-10}\mathbf{I}$, where $\mathbf{I}$ is the identity matrix of the appropriate size. The model fitting begins with vaguely informative priors and then iteratively updates the prior distributions to maximize the marginal likelihood, as in Godsey et al. [2012] and Beal et al. [2005]. The development parameters $\Delta_{\rho,s}$ are again treated as fixed, though as in Godsey et al. [2012] the parameters are typically updated (unless stated otherwise) to maximize the marginal likelihood.

4.2.2 Fitting the model using variational Bayes methods

Variational Bayesian methods are used to estimate the parameters of our model, as in Godsey et al. [2012]. Our methods of model fitting (though not necessarily the model itself) are very similar to those of Huang et al. [2007b,a, 2008] in discovering miR-mRNA target pairs as well as other analyses of gene expression data in Beal et al. [2005] and Teschendorff et al. [2005]. For a thorough explanation of variational Bayesian methods, see Winn [2003] or Beal [2003]. This algorithm was coded in the *R* [R Development Core Team 2009] statistical programming language.

4.2.3 Target prediction algorithms

This paper analyzes miR-mRNA target prediction data from both *TargetScan* [Lewis et al. 2005; Friedman et al. 2009; Grimson et al. 2007; Garcia et al. 2011] and *MiRanda* [John et al. 2004; Enright et al. 2003]. For each of these, a table of predicted miR-mRNA interactions and the targeting scores calculated by the respective algorithms was downloaded. *TargetScan* includes a *context+* score and *MiRanda* includes a *mirSVR* score. [Betel et al. 2010]

If one of the included algorithms predicts that miR i targets mRNA j, the vector $\langle 1, \alpha_{i,j} \rangle$ is used, where $\alpha_{i,j}$ is the prediction score from the algorithm. The vectors from multiple prediction algorithms are concatenated such that, if a pair $\{i, j\}$ is predicted by more than one algorithm, the prediction parameter vector $\boldsymbol{P}_{i,j}$ is of the form $\langle 1, \alpha_{i,j}, 1, \alpha'_{i,j} \rangle$.

4.2.4 Simulations

Data were simulated in order to evaluate the performance of the model under different conditions. Since there is no gold standard data set including all true-positive and true-negative miR-mRNA interactions, the simulated data sets are used as a substitute in which it is known whether a true interaction exists in each case.

The simulated networks consist of miR and mRNA expression data for each of eight ordered stages/samples. For each of the eight stages, there are three replicates for each mRNA type. To do this it is assumed that, for any given miR-mRNA pair, there is a 10% chance that the pair is predicted by our fictional targeting database. Then, it is assumed that the predicted pairs have a higher probability of being true target pairs: each predicted pair has a 50% chance of being a true target pair while non-predicted pairs have a 10% chance. The choice of each pair as true, according to these probabilities, is independent, and each true target pair is assigned a random negative interaction coefficient by taking the negation of a random draw from a gamma distribution with shape=rate=1. These are treated as the true interactions.

The miR expression data are then simulated through a random walk process starting at $t = 0$ and ending at $t = 8$, with each subsequent step, starting at position 0, chosen by the standard normal distribution. The data from points $t = 1$ through $t = 8$ are then used as simulated miR data. From these simulated miR expression values, the true interaction matrix is used to generate mRNA expression data for the same eight time points. Finally, the three "technical" replicates of each of the eight time points are created and independent Gaussian noise, with variance 0.1, is added to each data point; this is the final data set.

4.2.5 Multiple myeloma data

The model is demonstated using the multiple myeloma data set from Lionetti et al. [2009], which can be downloaded from *GEO* [Barrett et al. 2009], accession number GSE17498. In this data set, there are both miR and mRNA expression values for samples from 36 patients, 34 of whom have been diagnosed with multiple myeloma (MM) and 2 of whom have been diagnosed with plasma cell leukemia (PCL). Unfortunately, from healthy donors there is only miR expression data and no mRNA data, so healthy samples cannot be included in our study. However, the diseased samples can be arranged into four Durie-Salmon stages: IA, IIA, IIIA, and IIIB, which gives an obvious ordering for our non-PCL samples.

As in Godsey et al. [2012], prior to the main analysis, quantile normalization was performed across all arrays of the data set using the *limma* package for *R*. [Smyth and Speed 2003; R Development Core Team 2009] I then performed a probewise ANOVA to test for differential expression across the stages IA, IIA, IIIA, IIIB, and PCL (using individuals within a stage as replicates) and removed those probes/probesets (for both miR and mRNA) whose (unadjusted) p-value from the ANOVA F-test was greater than or equal to 0.05, as well as those miRs and mRNAs not involved in any predicted targeting interactions, leaving 28 miRs and 367 mRNAs as possible candidates for targeting interaction. Lastly, the data were re-scaled so that each probe/set—across all samples—had a mean of zero and a standard deviation of one.

4.3 Results

Below, details are given for the performace of *PEACOAT* on simulated data sets as well as a microarray data set of multiple myeloma samples.

4.3.1 Simulations

Five simulated networks are evaluated, with 10 miRs and 10 mRNAs, as well as 5 networks with 25 miRs and 25 mRNAs. When inferring interactions for the simulations, all development parameters $\Delta_{\rho,s}$ are left at a fixed at a value of 1, and they are not updated. This saves a considerable amount of time without materially affecting the integrity of the simulation results. The prediction weight parameters are set to $\Upsilon = 1$. For each simulated network and model configuration mentioned, a range of are tried, and for each of these 5 models were fit, each starting from different random parameter values.

In order to evaluate the performance of the model in inferring predictions, the AUROC statistic (area under the receiver operator characteristic) was used. Given a ranked list of interactions by statistical significance and a cut-off point on that list, The receiver operating characteristic (ROC) is the true positive rate (proportion of true interactions appearing above the cut-off) divided by the false positive rate (proportion of false interactions appearing above the cut-off). The AUROC is the area under the curve generated by calculating the ROC for all possible cut-off points. In general, it is a good measure of whether or not the items at or near the top of the list are true or not, and thus provides a general idea of the verifiability in practice of miR-mRNA interactions with the highest statistical significance.

Table 4.1 show the AUROC scores for *PEACOAT* over a range of cluster numbers on the 5 simulated networks with 10 miRs and 10 genes. For each network and number of clusters the AUROC shown is that of the inferred model with the highest likelihood score among the five model fits with random parameter initializations. For all networks, the highest AUROC is obtained by having fewer than 10 clusters, leading us to believe that clustering aids in interaction inference; however, the best number of clusters differs between networks, and, as shown in the table, setting the number of clusters to 7 is the only choice that out-performs 10 clusters (*i.e.* no clustering) in most cases.

The simulated networks with 25 miRs and genes, on the other hand, are best inferred using 11-15 clusters. As shown in Table 4.2, using a number of clusters in the range of 11-15 out-performed the no-clustering model 60-80% of the time, depending on the exact number of clusters. Furthermore, a randomly selected model within this range (instead of choosing the most likely of the 5 randomly initialized models for each cluster number), has a 68% chance of out-performing the no-clustering model. That implies that, when given a choice between fitting a model without clustering and fitting one with clustering, it is advantageous, more often than not, to choose to cluster, given the manner by which our simulated data were generated.

4.3.2 Multiple myeloma data

Given that the multiple myeloma data set, after pre-processing, contains 28 miRs, *PEACOAT* was fit to the data using a number of clusters from the optimal range from the simulated networks of size 25. Since 28 is slighly larger than 25, 15 clusters was chosen as an appropriate number. The greatly increased computational time on this larger data set prevents us from trying more model fits and cluster

Number of clusters:	3	4	5	6	7	8	9	10
Network 1:	0.677	0.637	0.664	0.667	0.711	**0.732**	0.671	0.712
Network 2:	0.487	0.545	0.482	**0.608**	0.558	0.515	0.507	0.532
Network 3:	0.416	0.450	0.486	0.498	**0.581**	0.531	0.550	0.537
Network 4:	0.634	0.518	0.651	0.647	**0.655**	0.592	0.580	0.652
Network 5:	0.588	0.682	**0.825**	**0.825**	0.723	0.819	0.779	0.732
% better than no clustering:								
each best model (of 5):	0%	20%	20%	40%	60%	40%	40%	0%
all models (5 models × 5 data sets):	12%	36%	20%	44%	56%	32%	40%	0%

Table 4.1: Size 10 network AUROC scores. For each of 5 simulated data sets, the table shows the AUROC from the highest-likelihood inferred network (of 5) from each of a range of cluster numbers. The best score for each network is shown in bold. The bottom two rows give (1) the percentage of data sets for which the best model for the particular number of clusters out-performed the best no-clustering (*i.e.* 10 cluster) model, and (2) the percentage of the all models fit for the particular number of clusters that out-performed the best no-clustering model.

Number of clusters:	11	13	15	17	19	21	23	25
Network 1:	**0.709**	0.674	0.628	0.565	0.590	0.596	0.598	0.609
Network 2:	0.530	0.614	0.593	0.524	0.603	0.603	**0.639**	0.601
Network 3:	0.591	0.606	**0.639**	0.557	0.624	0.546	0.570	0.588
Network 4:	0.538	0.506	0.521	0.559	0.611	0.593	0.612	**0.617**
Network 5:	0.632	0.634	**0.641**	0.632	0.603	0.568	0.574	0.520
% better than no clustering:								
each best model (of 5):	60%	80%	60%	20%	60%	40%	40%	0%
all models (5 models × 5 data sets):	68%	68%	68%	40%	44%	28%	28%	0%

Table 4.2: Size 25 network AUROC scores. For each of 5 simulated data sets, the table shows the AUROC from the highest-likelihood inferred network (of 5) from each of a range of cluster numbers. The best score for each network is shown in bold. The bottom two rows give (1) the percentage of data sets for which the best model for the particular number of clusters out-performed the best no-clustering (*i.e.* 25 cluster) model, and (2) the percentage of the all models fit for the particular number of clusters that out-performed the best no-clustering model.

numbers. The results using 15 clusters were compared to the results from the unclustered individual-ordered (*I-O*; where samples are ordered by developmental stage and samples from the same stage are not treated as replicates) model from Godsey et al. [2012] in order to show that *PEACOAT* with clustering can improve the accuracy of predicted interactions.

Table 4.3 shows the KEGG pathway terms [Kanehisa et al. 2008] that are enriched within the set of unique genes appearing in the top 100 ranked interactions. Enrichment was calculated using the *GeneCoDis* tool. [Carmona-Saez et al. 2007; Nogales-Cadenas et al. 2009]

	I-O model	TaLasso	*PEACOAT*
Number of unique genes in the top 100 interactions	54	84	25
05215 :Prostate cancer	3 (0.000511)		2 (0.002343)
05214 :Glioma	2 (0.005557)		2 (0.001211)
05218 :Melanoma	2 (0.006818)		2 (0.001498)
05219 :Bladder cancer	2 (0.002509)		
05016 :Huntington's disease	4 (0.000296)		
04115 :p53 signaling pathway	3 (0.000239)		2 (0.001408)
05010 :Alzheimer's disease	3 (0.003057)		
05014 :Amyotrophic lateral sclerosis (ALS)	2 (0.003820)		
04622 :RIG-I-like receptor signaling pathway	2 (0.007008)		
04976 :Bile secretion	2 (0.007008)		
04210 :Apoptosis	2 (0.010362)		
04120 :Ubiquitin mediated proteolysis		4 (0.0005292)	
04960 :Aldosterone-regulated sodium reabsorption			2 (0.000539)
04722 :Neurotrophin signaling pathway			2 (0.004585)
05160 :Hepatitis C			2 (0.005331)

Table 4.3: Enriched KEGG pathways among genes in the top 100 interactions. The top row gives the number of unique genes present in the top 100 miR-mRNA interactions according to each model; the remaining rows give, per column, the number of these genes annotated by *KEGG* pathway terms with significant enrichment (FDR corrected $p < 0.05$) for each of the models. The [uncorrected] hypergeometric p-value is given in parentheses. A blank entry indicates that the particular pathway was not significantly enriched in the model. Cancer-related *KEGG* pathways are shown at the top of the table, with other enriched pathways below.

Though fewer KEGG pathways were found to be significantly enriched by *PEACOAT* (7 pathways) than by the *I-O* model (11 pathways), the main concern is cancer-related pathways, of which the two models indicated 3 and 4 pathways, respectively. Of note is that *PEACOAT* found 3 pathways to be significant among only 25 genes. More importantly, the enrichment p-values for these three pathways are more significant than all but one of the 4 pathways found significant by the *I-O* model, indicating that a researcher is more likely to find a gene related to one of these pathways by independently verifying some genes from the list generated by *PEACOAT* than by the non-clustering model. This is true admittedly only by a narrow margin, but nonetheless provides further evidence of the usefulness of clustering within a miR-mRNA targeting model.

4.4 Discussion

The results given in this paper have shown that there are significant advantages to using clustering within a miR-mRNA interaction model. Our algorithm, which expands upon a previous algorithm incorporating both expression data and sequence-based predictions, performs best when clustering is enabled, as shown with both simulated and biological data. Since the algorithm fully integrates a version of a dynamic Bayesian network with a clustering model, the clusters and miR-mRNA interactions can be inferred simultaneously via an interative variational Bayesian algorithm.

First, the results showed that, given our simulated data with 10 miRs and 10 mRNAs, the optimal number of clusters was typically 7, but varied from 5 to 8. For the 10-miR networks, choosing 7 clusters gave better results than no-clustering in the majority of cases. One might expect that choosing 8 or 9 clusters would also give better results, since 7 clusters was better than 10 (i.e. no clustering), but this was not the case. Most likely, this is the result of two factors: (1) there is some randomness in the results, particularly when there are only five networks that were tested, and (2) choosing 10 clusters by design disables the clustering algorithm and removes an entire layer of inference that could potentially cause a lot of uncertainty in results. That is, when clustering is disabled, the cluster membership variables are fixed to exact values, and this absolute certainty can make the interaction inference algorithm converge much more quickly. This is something that could be tested rigorously in the future.

For the simulated networks of 25 miRs and 25 mRNAs, clustering proved much more valuable. The optimal number of clusters, depending on specific network, ranged from 11 all the way to 25, but choosing a number of clusters between 11 and 15 gave better results than a no-clustering model 68% of the time. This is strong evidence that clustering improves inference, particularly when few samples are present (in this case, 8 time points).

Lastly, when applied to a multiple myeloma data set, *PEACOAT* inferred 3 cancer-related KEGG pathways that were significantly enriched among the top 100 interactions. The level of significance of these enriched pathways was better than the top three cancer-related pathways from the non-clustering model upon which *PEACOAT* was based. If the simulated data proved that *PEACOAT* could infer true interactions more strongly with clustering than without, then these results demonstrate that the algorithm is a step forward in miR-mRNA target inference from real expression data.

It is commonly acknowledged that some genes work together as groups or modules within certain processes to accomplish a task, and that these genes are often co-expressed or at least highly correlated under certain circumstances. There is no reason to believe that miRs behave otherwise. Admitting that miRs sometimes form functional or de-facto clusters gives extra statistical power to network inference tasks. In a linear model such as *PEACOAT* as well as the most commonly used network inference tools, causality cannot be determined if two potential regulators are highly correlated, and only clustering can avoid the adverse effects that such uncertainty causes.

The results have shown using simulated data that algorithms with clustering can outperform algorithms with no clustering with regards to the task of miR-mRNA network inference, particularly when more miRs are included as potential regulators. On biological data, *PEACOAT* found more highly enriched gene function than non-clustering algorithms. Collectively, these results suggest, first of all, that the clustering of miRs is potentially a critical part of faithfully inferring miR-mRNA network interactions, and second of all that *PEACOAT* is a valuable and significant step in the right direction.

PEACOAT represents the state of the art in miR-mRNA interaction inference, incorporating both expression data and sequence-based target prediction, plus an integrated clustering algorithm that infers cluster members concurrently with the interactions themselves.

Source code for *PEACOAT*, in the *R* language, is freely available from the author at `https://github.com/briangodsey/peacoat`.

Chapter 5

Comparing and forecasting performances in different events of athletics using a probabilistic model

Published in the *Journal of Quantitative Analysis in Sports* [Godsey 2012]

by Brian Godsey[1]

1 School of Medicine, University of Maryland, Baltimore, MD, USA

Abstract: Though athletics statistics are abundant, it is a difficult task to quantitatively compare performances from different events of track, field, and road running in a meaningful way. There are several commonly-used methods, but each has its limitations. Some methods, for example, are valid only for running events, or are unable to compare men's performances to women's, while others are based largely on world records and are thus unsuitable for comparing world records to one other. The most versatile and widely-used statistic is a set of scoring tables compiled by the IAAF, which are updated and published every few years. Unfortunately, these methods are not fully disclosed. In this paper, we propose a straight-forward, objective, model-based algorithm for assigning scores to athletic performances for the express purpose of comparing marks between different events. Specifically, the main score we propose is based on the expected number of athletes who perform better than a given mark within a calendar year. Computing this naturally interpretable statistic requires only a list of the top performances in each event and is not overly dependent on a small number of marks, such as the world records. We found that this statistic could predict the quality of future performances better than the IAAF scoring tables, and is thus better suited for comparing performances from different events. In addition, the probabilistic model used to generate the performance scores allows for multiple interpretations which can be adapted for various purposes, such as calculating the expected top mark in a given event or calculating the probability of a world record being broken within a certain time period. In this paper, we give the details of the model and

the scores, a comparison with the IAAF scoring tables, and a demonstration of how we can calculate expectations of what might happen in the coming Olympic year. Our conclusion is that a probabilistic model such as the one presented here is a more informative and more versatile choice than the standard methods for comparing athletic performances.

5.1 Introduction

Quantitatively comparing performances from different athletic events and specifying how much more impressive one performance is than another are not simple tasks. There are a few good models that are valid for running events, particularly longer distances, namely those by McMillan [2011], Cameron [1998], Riegel [1977], and Daniels and Gilbert [1979]. These models rely on physiological measurements such as speed and running economy to compare performances at different race distances, either for men or for women, but not between them.

Purdy Points [Gardner and Purdy 1970] have long been used to compare marks from different events in both track and field, but these scores are based mainly on the world records of each event at a particular date in the past, which leads to two main disadvantages: (1) it is impossible to compare world records to each other if the model is based on them, and (2) basing the model on such a small data set leads to much uncertainty and variation in the scores as the records and model evolve over time. In other words, if a particular world record is “weak” in some sense, Purdy points will likely unfairly assign a higher score to performances in that event when compared to others.

Currently, the most popular method for comparing performances across all events in track and field as well as road running is to consult the IAAF scoring tables. [Spiriev and Spiriev 2011] These tables are updated every few years using methods that are not fully disclosed, with the last two updates occurring in 2008 and 2011. The IAAF is the main official governing body for international athletics, and they also publish the official scoring tables for “combined events competitions” such as the heptathlon and decathlon. These “combined events” consist of seven women’s and ten men’s events, respectively, and which are contested at most major international athletics competitions, and the winner is declared to be the competitor with the highest point total from all of the events. These combined events scoring tables were intended to assign a similar amount of points to a performances that are “similar in quality and difficulty”. [International Association of Athletics Federations 2001] All point values P in these tables can be calculated using a formula of the form $P = a(M - b)^c$, where M is the measured performance (use $M = -T$ for running times T, where a lower performance is better) and a, b, and c are constants estimated by undisclosed methods. [International Association of Athletics Federations 2001] The combined events tables are not the same as the general IAAF scoring tables, but it may be deduced that both sets of tables are produced using similar methods. Which data are used and how exactly the constants are estimated is not clear.

In this publication, we introduce a method of scoring athletic performances based on the idea that a good performance is a rare or improbable performance. Two very common reasons why one might think that an athletic performance is good are:

1. A performance is good if few athletes improve upon it, or
2. A performance is good if it is close to or improves upon the [previous] best performance.

The first reason is important because it puts emphasis on what has actually happened. In other words, if an athlete is in the top ten in the world in her event, she is likely better than an athlete who is ranked 50th or 100th. On the other hand, the second reason is important because it focuses more on what is possible. Sometimes in sport, a revolution occurs, whether in training, technique, equipment, or facilities, and performances improve dramatically. Certain events in history cause people to re-think what they thought was good—Bob Beamon's 1968 Olympic long jump in Mexico City, Paula Radcliffe's 2003 London Marathon, and more recently Usain Bolt's 2009 World Championship 100m run in Berlin come to mind. In some of these cases, but not in others, what we once thought was unthinkable becomes commonplace. In 1996, many people thought that Michael Johnson's 200m world record would last an eternity—it was revolutionary—but now it is only fourth on the all-time list. The men's marathon record has dropped tremendously in recent years, carried in part by Haile Gebreselassie and Paul Tergat, who accomplished the same feat for the 10,000m run in the 1990s. The point is only that a superb, dominating performance might be one of the greatest feats ever witnessed, but it also might be an inevitability. Usain Bolt's 9.58s mark in the Berlin 100m dash in 2009 is certainly impressive, but we saw three men running 9.72s or faster in the 100m dash in 2008, all under the world record from 2007; so how impressive was 9.58s really? Is it a statistical outlier, or is it the expected result of a general increase in performance level which by chance had not yet produced the outstanding performance that was bound to happen? These are some questions this paper was intended to answer.

The methods introduced here utilize a large amount of historical data to estimate directly the improbability of athletic performances. Using a data set consisting of the top n performances of all time—where n is generally well over 100 and can be different for each event—we estimate a log-normal distribution for each event, allowing us to calculate directly both the probability that a specific mark is exceeded as well as the expected number of such performances within a given time period. We use this model to predict the number and quality of top performances in the subsequent years, for data up until the year 2000 and also 2008, and we show that our scoring tables based on data prior to 2008 correlate more highly with actual data than do the 2008 IAAF scoring tables. Lastly, we look ahead to the coming year and the 2012 Olympic Games in London, and we determine which world records are most in danger of being broken and which are most likely to last a while longer.

5.2 Methods

In general, we estimate a log-normal distribution for each athletic event k using a list of the best n_k marks from that event. Equivalently, we assume that the natural logarithms of performances from each event are normally distributed. We use this second formulation throughout this paper.

A list of best marks represents only one tail of the distribution, and so for simplicity we convert marks so that we perform all calculations on the lower tail. For running events, a lower time is better, and thus we take only the natural logarithm of the times, in seconds, before fitting a normal distribution to the data. For throwing and jumping events, a higher mark is better, so we assume that the inverse (negative) of the natural logarithm is normally distributed. This does not cause any adverse consequences as long as we again take the inverse before converting back to an actual mark, typically in centimeters (cm).

Figure 5.1 illustrates how a normal distribution can be fit to a list of top [log-]performances, represented by a histogram. Since we are working exclusively with the tail of the distribution, the parameters must be estimated from the shape of the tail.

In our first set of analyses, we fit the model to the data as it would have been at the beginning of 2000, and we test its predictive ability for the subsequent years. Below, we elaborate on exactly how we calculate these predictions and their comparison with actual outcomes.

In the second set of analyses, we fit the model to the data as it would have been at the beginning of 2008, and we test its predictive ability for the following four years. Then we generate a set of scoring tables analogous to the IAAF scoring tables and we compare some predictions that could be made from the tables to those of the IAAF scoring tables. Granted, the IAAF may not have intended for such specific predictions to be made, but we try to be as fair as possible based on what it might mean for one athletic performance to be “better” than another. We think that, generally, performances that are given equal scores should, in any

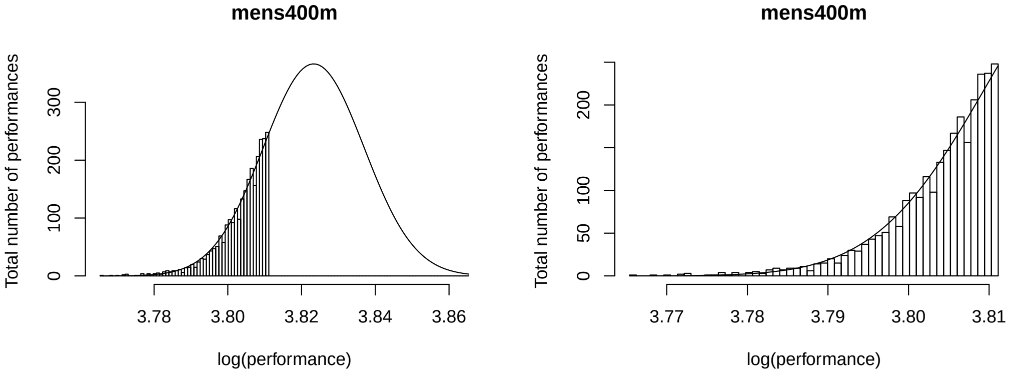

Figure 5.1: Illustration of model fit. Both panels in this figure show a histogram of log-performances for the men's 400m dash (all data until the present day) as well as the fitted normal distribution curve that is re-scaled to match the histogram. The left panel gives a wider view, while the right panel shows in more detail the area of the graph which contains performances appearing on the list of top marks.

given year, (1) have approximately the same number of marks exceed them, (2) should have the same chance of being broken, and (3) should have a comparable relative margin (in percent) between itself and the best mark of the year.

We then give the results of a third set of analyses that uses data through October 1st, 2011, including predictions about the numbers of top performances that will occur in the coming years as well as what we expect the top mark to be in each event and the probabilities of new world records being set.

5.2.1 Data

An ideal data set would consist of a complete list of every performance by an elite athlete in the modern era of athletics. Such a list, as far as we can tell, does not exist. We do have, however, lists of the best performances ever. The lists compiled by `www.alltime-athletics.com` [Larsson 2011] include all of the top performances of all time—list lengths ranging from a few hundred to several thousand, depending on the event. We have data for all track and field events contested in the modern Olympic Games for men and women, except the heptathlon and decathlon, plus the marathon, half marathon, one mile run and 3000m run. We assume that these lists are complete, in the sense that each list is indeed the best n_k performances for event k, with no missing marks.

For the three time periods we consider—to which we will refer by year, 2000, 2008, and 2012—we do two sets of analyses, one using all data prior to that year, and the other using data from only the prior 5 years.

The performance lists for performances prior to the present day (1 October 2011) have lengths between 215 and 9672, with a median of 1596.5. For the five years prior to the present day, list lengths range from 10 to 4630, with a median of 275. The women's one mile run is the shortest list, and the second shortest list has 51 entries.

The performance lists for performances prior to 2008 have lengths between 205 and 5547, with a median of 1273. Using five years of data prior to 2008 gives a range of list lengths from 18 to 4235, with a median of 298.5. The list of length 18 belongs to the women's one mile run, and the next shortest is the women's shot put, with 38 entries. These are special cases where either the event is rarely contested (one mile run) or has a dearth of recent top performances (shot put). All other lists include at least 68 performances.

The performance lists for performances prior to 2000 have lengths between 63 and 3761, with a median of 790. For the five years prior to 2000, list lengths range from 52 to 1288, with a median of 252.5.

5.2.2 The model

A normal (or log-normal) distribution takes two parameters: mean μ and variance σ^2. Given these parameters, we can calculate the probability p_a that a particular performance in event k exceeds a specified mark a using the formula:

$$p_a = \int_{-\infty}^{a} N(x \mid \mu_k, \sigma_k^2) dx \tag{5.1}$$

where a is a specified performance (natural logarithm of a mark, inverted for events in which greater marks are better) and $N(x \mid \mu_k, \sigma_k^2)$ is the normal distribution probability density function (pdf). Equation 5.1 is equivalent to the cumulative distribution function (cdf) of the normal distribution with mean μ_k and variance σ_k^2, which we call $F(a \mid \mu_k, \sigma_k^2)$. If we accurately estimate μ_k and σ_k^2, then p_a is easy to compute.

We can use $F(a \mid \mu_k, \sigma_k^2)$ to formulate the pdf of a normal distribution truncated at c_k as:

$$p_k(x \mid \mu_k, \sigma_k^2) = \begin{cases} \frac{N(x|\mu_k,\sigma_k^2)}{F(c_k|\mu_k,\sigma_k^2)} & \text{for } x \leq c_k \\ 0 & \text{elsewhere} \end{cases} \tag{5.2}$$

Bayes' Theorem then gives the un-normalized posterior density for the model parameters:

$$\ell(\mu_k, \sigma_k^2 \mid X_k) = \prod_{x \in X_k} \frac{N(x \mid \mu_k, \sigma_k^2) p(\mu_k) p(\sigma_k^2)}{F(c_k \mid \mu_k, \sigma_k^2)} \tag{5.3}$$

where X_k is the set of performances on the list for event k, and $p(\mu_k)$ and $p(\sigma_k^2)$ are the prior probability distributions of μ_k and σ_k^2, respectively.

5.2.3 Development of an empirical prior

In general, we would like to use non-informative prior distributions for our model parameters μ_k and σ_k^2, but when first fitting our model to the data, it quickly became clear that there was much uncertainty about the total population size N_k for each event k. So, we used an empirical Bayes approach to estimate reasonable prior expectations for the N_k in order to reduce this uncertainty.

That is, the posterior densities suggested that when using non- or weakly-informative priors for each event, many $\{\mu_k, \sigma_k^2\}$ pairs were nearly equally likely, and they gave a wide range of values for N_k, as calculated according to the following relation:

$$F(w_k \mid \mu_k, \sigma_k^2) = \frac{n_k}{N_k} \tag{5.4}$$

where, n_k is the [constant] length of the list of best performances for event k, and w_k is the worst mark on that list. Equation 5.4 is inherently true, as it says only that the cumulative density through the region for which we have data—i.e. the tail—is equal to the size of the data set, n_k, divided by the size of the largest possible data set, N_k.

In order to reduce this uncertainty over the N_k and ensure that the estimated population sizes for different events were similar, we re-parametrized the model, using equation 5.4, to use N_k as a parameter instead of σ_k^2. Then, we assume a log-normal prior distribution for the N_k, with parameters μ_N and σ_N^2, as well as a uniform prior distribution over all real numbers for the μ_k, which is non-informative and improper.

We would, ideally, optimize the parameters μ_N and σ_N^2 of the prior for N_k, as suggested by Mackay [1999], iteratively as we fit the model, but since the model is fit independently for each event and because calculation takes a considerable amount of time, we are not able to use many iterations. We chose to approximate two such iterations, where in the first iteration we fit all models using a very weakly-informative prior for N_k (i.e. $\mu_N = 10{,}000$ and $\sigma_N^2 = e^{20}$), and then, in the second iteration, we re-fit the models with updated parameters μ_N and σ_N^2, which were optimized based on point estimates for the N_k. Specifically, we calculate from the first-iteration posterior distributions, for each k, the expected value of N_k,

$E[N_k]$, and then using these estimates to update μ_N and σ_N^2 according to the following:

$$\mu_N = \text{median}(\{\log(E[N_k]) : \text{for all } k\}) \tag{5.5}$$

$$\sigma_N^2 = \min_{K \subset \{\text{all } k\}} \text{var}(\{\log(E[N_k]) : k \in K\}) \tag{5.6}$$

where the subset K comprises 75% of the set of all events. Thus, both prior distribution parameters μ_N and σ_N^2 are robust to some outlying N_k, which we encountered in a few cases, particularly in events for which we have little data, as well as with data from the sprints, high jump, and pole vault, because those data are more discrete than others, as many competitors share the same mark. We chose the value 75% somewhat arbitrarily, but it ensures that most of the data are used while allowing for inaccurate values due, for example, to small or highly discrete data sets. Updating the prior distribution for the N_k only once in this manner gives a compromise between non-informative and fully optimized priors, while improving convergence and sharing some information between models for different events.

While we do not expect the population sizes from different events to be identical—there are many reasons why there could be more participants or performances in one event than another—we do not expect them to be vastly different, either. For example, there are more marathon times posted each year than in any other event, though admittedly most are not elite times. Also, the one mile run and the 1500m run are very similar in distance, yet each year there are far more 1500m races than mile races. Sprinters tend to run more races each year than long distance runners, as well. On the other hand, we expect the population sizes to be relatively similar, perhaps within an order of magnitude of each other, simply because—among other reasons—awards, medals, and championships are generally identical in nature and quantity for most events, and identical incentive leads us to believe that population sizes would be approximately equal. We have tried to address this in choosing our prior distributions.

5.2.4 Fitting the model to the data

To fit the model (5.3) for each event, we use Markov chain Monte Carlo (MCMC) methods as implemented in the *mcmc* package [Geyer. 2010] of the *R* programming language [R Development Core Team 2009], which is a version of the Metropolis-Hastings algorithm. [Hastings 1970] We use a "burn in" period of 1,000 steps, after which we test the sample acceptance rate, requiring it to be between 0.2 and 0.4 (we found that this range generally gives good convergence), and if unacceptable we re-do the burn-in with an adjusted sample step size. This process is automated.

Following burn-in, we use a subsequent 1,000 batches of 50 steps each with 10 random parameter initializations to determine the joint distribution of μ_k and σ_k^2—and/or N_k—for each k.

Convergence of the MCMC sampling was assessed visually using various plots as well as using the multivariate diagnostic of Gelman and Rubin [1992] as implemented in the *coda* [Plummer et al. 2006] package in *R*. [R Development Core Team 2009]

5.2.5 Some meaningful statistics

The value of p_a as calculated in equation 5.1 can be interpreted as the probability that in a given performance a specified member of the total elite athlete population for the given event performs better than the mark a. This is a natural measure of performance quality, but it is not easy to test its accuracy using real data. Therefore, in this section we give some other statistics based on the model that may be better at describing the performances we witness during an athletic season. They are based on the ideas stated in the introduction to this paper, that we can measure the rarity—and quality—of a performance by the number of marks that improve upon it or by comparing it with a reference performance. Unless stated otherwise, the statistics below are estimated using 1000 samples of the parameter values.

Expected number of performances improving upon a specified mark

If we fit the model to t_m years of data, then for any point estimates of μ_k, σ_k, and N_k (and hence the cdf $F(a \mid \mu_k, \sigma_k^2)$) for each event k, the expected number of performances during one calendar year that are better than a is:

$$A_k(a \mid \mu_k, N_k) = \frac{N_k}{t_m} F(a \mid \mu_k, N_k) \tag{5.7}$$

using the re-parametrized version of the cdf function F (with μ_k and N_k as given parameters instead of μ_k and σ_k^2). We can use our previously-obtained samples from the posterior distributions of the parameters to efficiently find the posterior expected value $\hat{n}_k(a)$ of $A_k(a \mid \mu_k, N_k)$:

$$\hat{n}_k(a) = \iint A_k(a \mid \mu_k, N_k) p(\mu_k, N_k \mid X_k)\, d\mu_k\, dN_k \tag{5.8}$$

This expected number of marks can be compared with data from future athletics seasons (i.e. data not included when fitting the models).

Probability of a record being broken

If we fit the model to t_m years of data, then for any point estimates of μ_k, σ_k^2, and N_k for each event k, the probability that the best performance over t_f calendar years is better than a performance a is:

$$B_k(a \mid \mu_k, N_k) = 1 - [1 - F(a \mid \mu_k, N_k)]^{\frac{t_f N_k}{t_m}} \tag{5.9}$$

We can compute the posterior expectation of $B_k(a \mid \mu_k, N_k)$ as we did in equation 5.8:

$$\hat{p}_k(a) = \iint B_k(a \mid \mu_k, N_k) p(\mu_k, N_k \mid X_k)\, d\mu_k\, dN_k \tag{5.10}$$

This estimated probability $\hat{p}_k(a)$ of a mark a being broken by anyone during the given year can be useful for comparing the very best performances—as we do in the *Results* section—but is less suitable for comparing lesser marks. This is because the probability of a lesser mark being broken in the course of a year is very high, and quickly approaches 1 as the quality of the mark a decreases.

Expected best performance

Equation 5.10 gives the estimated probability that a particular mark will be broken in a given calendar year. In other words, it is the estimated cdf of the best performance for the year. Therefore, the probability density of the best performance y_1 during that year is the derivative of $\hat{p}_k(a)$ from equation 5.10, and the expected best performance is:

$$\hat{y}_1 = \int_{-\infty}^{\infty} y_1 \left(\frac{d}{dy_1} \hat{p}_k(y_1) \right) dy_1 \tag{5.11}$$

The quantity $\hat{y}_1$ is the expectation of an order statistic on normally distributed data, for which there is no closed-form expression. Furthermore, we have calculated the values of the function $\hat{p}_k(a)$ using numerical integration over the posterior parameter distributions, so the calculation of $\hat{y}_1$ is not straight-forward. However, the high-density region of the derivative of $\hat{p}_k(y_1)$—i.e. the pdf of the year's best performance—is unimodal and in a predictable location, namely close to other years' best performances. Thus, to calculate $\hat{y}_1$, we first estimate the derivative of $\hat{p}_k(y_1)$ by estimating $\hat{p}_k(y_1)$ for a large number of values of y_1 (using samples $\{\mu_k, N_k\}$ from the parameter posterior distributions) and calculating the estimated differentials $\Delta\hat{p}_k(y_1)$ between adjacent values of y_1. Then, we use the estimates $\Delta\hat{p}_k(y_1)/\Delta y_1$ in place of the derivative to perform the integral in equation 5.11 numerically. Because the density function for y_1—the derivative of $\hat{p}_k(y_1)$—is unimodal and has high density only in a predictable location, this numerical integration is quick, easy, and accurate.

Proposed formula for performance scoring

We propose a formula for scoring that is analogous to the IAAF scoring tables. For this, we choose to define the quality of an elite performance mainly using $\hat{n}_k(a)$ above, i.e. the expected number of performances exceeding a given reference mark. That is, two elite-level marks may be considered equal if we expect them to be exceeded by the same number of individual performances during a calendar year. The statistic $\hat{n}_k(a)$ is itself valid only for the highest levels of competition—those represented on the lists of top performances that we have—but we would like our scoring formula to be valid for most events also at sub-elite levels. To do this, we took a particular value for $\hat{n}_k(a)$—we chose 0.125 because it was close to most of the current world records—and we defined the corresponding mark a_0 to be equal to 1300 points, which is approximately equivalent to most world records on the IAAF scoring tables. We then define the score S_a of any mark a to be

$$S_a = \begin{cases} 1300 \log_2(a_0) + 1 - \log_2(a) & \text{for times} \\ 1300 \log_2(a_0) - 1 + \log_2(a) & \text{for distances} \end{cases} \tag{5.12}$$

A problem that we encountered here is that a good mark in the one mile run is far more rare than than a comparable mark in the 1500m run, since the mile is run less often. Because the training and ability to run the two events are practically identical, we can assume that the athletes are interchangeable, and so, to remedy the discrepancy between the population sizes N_k for the two events, we set the population size N_k for the mile equal to that of the population size for the 1500m, for both men and women. This is a somewhat arbitrary choice, but the mile is not contested at the major championships and is thus rather dissimilar to the other events; rather than throwing it out entirely, we found that borrowing the N_k from the 1500m run produced satisfactory results.

5.2.6 Correlation with future performances

For each of the above-mentioned statistics, we would like to compare our predictions with those of other scoring methods. However, the other scoring methods give only a relative score, and no predictions. Thus, to compare our methods to the others, we must use a relative measure. Given a list of performances, one for each athletic event, we assign scores to each mark and then calculate the Pearson correlation coefficient between the scores and some future outcome, either the number of better performances for each event or the improvement in performance over some reference mark. For the purposes of comparing with the IAAF scoring tables, we define "improvement in performance" of a new mark a_{new} over an old mark a_{old} to be $-\log(a_{new}/a_{old})$. This gives a measure of the relative improvement, which could be negative if the new mark is worse than the old mark. As above, we use the inverse of this score for events in which a higher mark is better. The expected relative improvement is another estimate of the quality of a given performance. Below, we use as reference performances a_{old} the 10th, 25th, 50th, and 100th best all-time performances prior to the analysis year (2000, 2008, or 2012).

For example, for the year 2000 analysis, we calculate the expected best performance x_1 over the next two years (2000-2001) and we let this be a_{new} while the 10th, 25th, 50th, and 100th best performances prior to 2000 are each used as a_{old}. This gives four different versions of the expected improvement score for each athletic event for each analysis year, for which we can then calculate a Pearson correlation with actual performances in those subsequent years. If an a_{old} for a particular event is weaker than that of other events, we expect to see a larger improvement in subsequent years, and likewise a smaller improvement for stronger reference performances a_{old}.

Below, we list many such correlations for our scoring methods, and we compare them with correlations for the IAAF scoring tables.

5.3 Results

In this section, we give three sets of results: one for data preceding 2000, which we compare with later performances; one for data preceding 2008, which we compare with later performances as well as to the 2008 IAAF scoring tables; and one for data up to the present day (1 October 2011), which we use to make predictions for the coming years.

5.3.1 Convergence

For the three time periods, 2000, 2008, and 2012, and for each of these using all prior data and then only five years of data (thus, six cases in total), the MCMC sampling converged usually without using the empirical prior on the total population size. The slowest convergence in general occurred when using five years of data prior to 2008. Only 37 out of 48 events had Gelman-Rubin diagnostic statistics less than 1.1. When using the empirical prior, the Gelman-Rubin diagnostic was less than 1.1 for every event in every case, and in each case was less than 1.05 for at least 43 of the 48 events.

Population sizes varied between the events, and the use of the empirical prior on N_k improved convergence and moderated unreasonable population sizes. For example, for all data preceding 2008, the median population size was 19,028, and the robust standard deviation (using 75% of the events) of $\log(N_k)$ was 2.93. The smallest (unrestricted) estimated population size was 510.4 for the women's mile run, and the largest was 2.71×10^{16} for men's pole vault. Large population sizes such as that of the men's pole vault are clearly too large, and thus using the empirical prior makes intuitive sense as well as improves convergence. The estimated population size for men's pole vault when using the empirical prior was still 38.0 million (3.8×10^7), and that of the women's mile run was 1309.9, so some flexibility in the choice of population sizes was preserved.

A set of selected posterior expectations of parameter values are shown in table 5.1. Fans of track and field will notice that the marks e^{μ_k} are rather mediocre for elite athletes, and those events with larger estimated population sizes have less impressive values for e^{μ_k}, which makes sense intuitively. Assuming that the very best athletes are always participating in their respective events, a larger population size indicates that there are more less-talented athletes participating and making the average performance weaker.

Event	$e^{\mu_k-2\sigma_k}$	e^{μ_k}	$e^{\mu_k+2\sigma_k}$	$E[N_k]$
mens100m	10.55	11.28	12.05	1371048
mens200m	21.76	23.66	25.72	1543952
mens1500m	3:32.22	3:38.78	3:45.55	2469
mensMarathon	2:05:55.76	2:11:13.44	2:16:44.49	1284
mensHJ	2.00	2.11	2.22	695184
mensLJ	6.30	6.96	7.68	326672
womens100m	11.33	12.01	12.73	83707
womens200m	23.32	24.75	26.27	146548
womens1500m	3:59.52	4:05.33	4:11.29	625
womensMarathon	2:22:38.28	2:29:23.24	2:36:27.37	823
womensHJ	1.69	1.82	1.96	11229
womensLJ	5.53	6.00	6.51	174252

Table 5.1: Examples of fitted distributions. Shown here are a few summaries of selected fitted distributions. In the rightmost four columns, we give the log-normal equivalent of a normal distribution's (1) mean minus two standard deviations (i.e. $e^{\mu_k-2\sigma_k}$), (2) the mean, and (3) mean plus two standard deviations, as well as (4) the posterior expectation of the total population size. Running times are given in hours:minutes:seconds, where applicable, distances and heights are given in meters, and population sizes are the number of performances in the five-year period 2007-2011.

5.3.2 Predictions made prior to 2000

We used data from before 2000 to predict both the number of performances exceeding and the expected improvement over four different reference marks in each event, namely the 10th, 25th, 50th, and 100th best ever marks in each event at the end of 1999. The Pearson correlations of our predictions with the actual outcomes in the subsequent 12 years can be seen in tables 5.2 and 5.3.

years	**using all prior data**				**using 5 years of prior data**			
	10th	25th	50th	100th	10th	25th	50th	100th
2000-2001	-0.185	-0.139	-0.118	0.090	0.226	0.414	0.498	0.612
2000-2003	-0.198	-0.095	-0.100	0.062	0.163	0.380	0.463	0.581
2000-2005	-0.175	-0.082	-0.094	0.050	0.139	0.352	0.423	0.572
2000-2007	-0.164	-0.082	-0.096	0.049	0.124	0.331	0.397	0.554
2000-2009	-0.161	-0.085	-0.097	0.051	0.117	0.323	0.388	0.548
2000-2011	-0.158	-0.085	-0.097	0.049	0.116	0.319	0.382	0.552

Table 5.2: Correlations, 2000 number of better performances. Given in the table are the Pearson correlation coefficients between the predicted and actual number of performances exceeding a reference mark, based on the year 2000. The reference marks (the columns) are the 10th, 25th, 50th, and 100th best prior mark in each event.

years	**using all prior data**				**using 5 years of prior data**			
	10th	25th	50th	100th	10th	25th	50th	100th
2000-2001	-0.274	0.343	0.583	0.562	0.765	0.832	0.877	0.864
2000-2003	0.049	0.484	0.641	0.609	0.696	0.783	0.837	0.839
2000-2005	0.176	0.512	0.644	0.607	0.674	0.769	0.822	0.807
2000-2007	0.248	0.538	0.659	0.624	0.699	0.785	0.834	0.824
2000-2009	0.261	0.519	0.637	0.592	0.698	0.777	0.824	0.808
2000-2011	0.302	0.545	0.656	0.617	0.731	0.804	0.845	0.835

Table 5.3: Correlations, 2000 performance improvement. Given in the table are the Pearson correlation coefficients between the predicted and actual performance improvement over the reference mark, based on the year 2000. The reference marks (the columns) are the 10th, 25th, 50th, and 100th best prior mark in each event.

We can see in table 5.2 that the predicted number of better performances correlates much more highly with the actual outcomes when we used only the previous five years of data. In fact, the predictions using all data had very poor correlation (Pearson) with the actual outcomes, but the same is not true of the predicted performance improvement. The predicted improvements were significantly correlated with the actual improvements both when we used all data and when we used only the previous five years of data, though the latter still gives better results. We suspect that that the total number of athletes participating in the various events has changed more dramatically over time than has the quality of the very best performers, making our predictions of best performances—and the associated improvement score over the reference marks—more accurate than our predictions of numbers of athletes exceeding the same reference mark.

Table 5.4 gives the Pearson correlation of the predicted probabilities of a world record being set with the actual outcome (1 for a world record, 0 for none) over a given time period. Again, there is significant correlation between the predictions and the outcomes, and the predictions based on five years of data were generally better than those based on all data. Also, the correlations generally increased when more years were considered; this is likely due to the rarity of records, whereby the calculated probability of a world record occurring in the next 12 years will be more accurate than the probability for only one or two years. Based on only five years of data, we achieved Pearson correlation coefficients of approximately 0.7 for time periods of length 6-12 years.

years	**all data**	**5 years**
2000-2001	0.144	0.324
2000-2003	0.289	0.443
2000-2005	0.467	0.712
2000-2007	0.442	0.706
2000-2009	0.383	0.675
2000-2011	0.344	0.678

Table 5.4: Correlations, 2000 world records. Given are the Pearson correlation coefficients between the predicted probability of a world record being set and the actual occurrence (vector of zeros and ones), based on the year 2000.

5.3.3 Predictions made prior to 2008

In general, the predictions we made based on data prior to 2008 were much better than those from 2000. This could be due to a number of factors, such as the much larger data set, the increased modernization of training and competition, or the likely decrease in the use of performance-enhancing drugs. However, the predictions made using only five years of prior data were again considerably better than those using all prior data. In fact, our predictions of both the number of performances exceeding and the relative improvement over the 100th best performances of all time have Pearson correlations greater than 0.83 with the actual outcomes in the 2008 athletics season as well as for all seasons through 2011. Tables 5.5 and 5.6 show the details of the correlation coefficients.

Table 5.7 shows the Pearson correlation between predicted probabilities of world record being set and the actual outcomes. For the period 2008-2011, the predicted probabilities had a correlation coefficient of 0.48 with the actual outcomes, which is slightly higher than the corresponding correlation coefficient from the four-year period beginning in 2000, as shown in table 5.4. Thus, our predictions from the beginning of the year 2008 are better in nearly every case than those from the year 2000.

year(s)	using all prior data				using 5 years of prior data			
	10th	25th	50th	100th	10th	25th	50th	100th
2008	0.322	0.330	0.294	0.193	0.561	0.765	0.774	0.841
2008-2009	0.329	0.298	0.298	0.197	0.672	0.752	0.784	0.834
2008-2010	0.333	0.299	0.331	0.206	0.602	0.728	0.783	0.831
2008-2011	0.321	0.308	0.345	0.210	0.605	0.751	0.806	0.847

Table 5.5: Correlations, 2008 number of better performances. Given in the table are the Pearson correlation coefficients between the predicted and actual number of performances exceeding a reference mark, based on the year 2008. The reference marks (the columns) are the 10th, 25th, 50th, and 100th best prior mark in each event.

year(s)	using all prior data				using 5 years of prior data			
	10th	25th	50th	100th	10th	25th	50th	100th
2008	0.188	0.129	0.249	0.286	0.825	0.821	0.830	0.835
2008-2009	0.110	0.064	0.191	0.260	0.847	0.842	0.849	0.851
2008-2010	0.038	0.042	0.181	0.260	0.846	0.841	0.849	0.853
2008-2011	0.030	0.068	0.214	0.298	0.837	0.836	0.847	0.853

Table 5.6: Correlations, 2008 performance improvement. Given in the table are the Pearson correlation coefficients between the predicted and actual performance improvement over the reference mark, based on the year 2008. The reference marks (the columns) are the 10th, 25th, 50th, and 100th best prior mark in each event.

year(s)	all data	5 years
2008	0.225	0.338
2008-2009	0.363	0.497
2008-2010	0.278	0.491
2008-2011	0.257	0.484

Table 5.7: Correlations, 2008 world records Given are the Pearson correlation coefficients between the predicted probability of a world record being set and the actual occurrence (vector of zeros and ones), based on the year 2008.

5.3.4 Comparison with IAAF scoring tables

The scoring tables we have constructed based on the model described in this paper are designed to be analogous to the IAAF scoring tables [Spiriev 2008; Spiriev and Spiriev 2011], ranging from a score of zero for a relatively poor performance to approximately 1300 points for the current world records. A subset of scores from our tables can be found in table 5.8; a full table can be found in the supplementary materials. (Note: for the five years preceding 2012, there were only 10 marks in the data set for the women's one-mile run; though parameter convergence was achieved, the scores assigned were clearly not in line with the women's 1500m performances. We include the women's one-mile run in the scoring tables for completeness, but we discourage their use in performance comparison.) Thus, the two sets of tables have both been made mainly to compare elite-level performances, though they both are applicable to the performances of even recreational athletes. In previous sections, we tested the predictive ability of the model and the various statistics we calculate from it; in this section, we do the same tests on the predictive ability of the scoring tables constructed in this paper—using data prior to 2008—and we compare the results to those of the 2008 IAAF scoring tables, which to the best of our knowledge were constructed based on the same available data.

points	**800**	**900**	**1000**	**1100**	**1200**	**1300**	**1400**
mens100m	12.50	11.85	11.24	10.66	10.10	9.58	9.08
mens200m	25.12	23.82	22.58	21.41	20.30	19.24	18.24
mens400m	56.85	53.89	51.10	48.44	45.93	43.54	41.28
mens800m	2:11.64	2:04.81	1:58.33	1:52.18	1:46.36	1:40.84	1:35.60
mens1500m	4:30.89	4:16.82	4:03.49	3:50.84	3:38.86	3:27.49	3:16.72
mens3000m	9:35.72	9:05.83	8:37.48	8:10.62	7:45.14	7:20.99	6:58.09
mens5000m	16:26.21	15:35	14:46.45	14:00.43	13:16.79	12:35.42	11:56.20
mens10000m	33:54.31	32:08.69	30:28.54	28:53.60	27:23.59	25:58.25	24:37.34
mensHalfMara.	1:15:34.51	1:11:39.07	1:07:55.85	1:04:24.22	1:01:03.58	57:53.36	54:53.01
mensMarathon	2:40:02.90	2:31:44.29	2:23:51.58	2:16:23.40	2:09:18.50	2:02:35.66	1:56:13.74
womens100m	13.80	13.09	12.41	11.76	11.15	10.57	10.02
womens200m	28.29	26.82	25.43	24.11	22.86	21.67	20.54
womens400m	1:03.38	1:00.09	56.97	54.01	51.21	48.55	46.03
womens800m	2:29.47	2:21.71	2:14.35	2:07.37	2:00.76	1:54.49	1:48.55
womens1500m	5:08.60	4:52.57	4:37.38	4:22.98	4:09.32	3:56.38	3:44.11
womens3000m	10:56.41	10:22.33	9:50.01	9:19.38	8:50.34	8:22.80	7:56.69
womens5000m	18:13.36	17:16.59	16:22.77	15:31.74	14:43.36	13:57.49	13:14.01
womens10000m	38:35.22	36:35.01	34:41.04	32:52.98	31:10.54	29:33.42	28:01.34
womensHalfMara.	1:25:54.31	1:21:26.69	1:17:12.96	1:13:12.40	1:09:24.34	1:05:48.11	1:02:23.12
womensMarathon	3:01:13.87	2:51:49.27	2:42:53.98	2:34:26.49	2:26:25.36	2:18:49.20	2:11:36.73

Table 5.8: Subset of scoring tables. A sample of scores from the scoring tables based on our model, using five years of data prior to 2012. Here, we show only running events, but scores for other events can be found in the full table.

Table 5.9 gives the Pearson correlations of the reference performance scores (as assigned by the sets of scoring tables to the same reference performances we used in previous analyses) with the number of marks exceeding the reference performances in subsequent years. Similarly, table 5.10 gives the correlations of the same scores with the relative improvements over the reference performances. Note that these correlations should be negative because a higher score indicates a better performance, which should then see fewer better performances and less improvement in the subsequent years.

The scoring tables constructed in this paper using five years of data (but not those using all data) are more predictive of future performances than the IAAF tables. For example, using the 10th best all-time performance (as of 2008) as a reference, the scores assigned by the 2008 IAAF tables have a Pearson correlation coefficient of -0.22 with the numbers of better performances from 2008 to 2011, compared to -0.43 for our tables. Likewise, the relative improvements over this

year(s)	**IAAF scoring tables**				**our tables, all data**				**our tables, 5 years**			
	10th	25th	50th	100th	10th	25th	50th	100th	10th	25th	50th	100th
2008	-0.23	-0.34	-0.37	-0.51	-0.14	-0.27	-0.21	-0.29	-0.42	-0.54	-0.52	-0.64
2008-2009	-0.24	-0.31	-0.37	-0.47	-0.18	-0.24	-0.25	-0.28	-0.46	-0.51	-0.54	-0.62
2008-2010	-0.20	-0.29	-0.35	-0.45	-0.12	-0.20	-0.24	-0.26	-0.39	-0.49	-0.54	-0.60
2008-2011	-0.22	-0.30	-0.36	-0.47	-0.14	-0.22	-0.26	-0.27	-0.43	-0.52	-0.57	-0.62

Table 5.9: Correlations, scoring tables with number of better performances. Shown are the Pearson correlation coefficients between the points assigned by scoring tables and the actual number of better performances, based on the year 2008. The reference marks (the columns) are the 10th, 25th, 50th, and 100th best prior mark in each event. More negative correlations are better.

year(s)	**IAAF scoring tables**				**our tables, all data**				**our tables, 5 years**			
	10th	25th	50th	100th	10th	25th	50th	100th	10th	25th	50th	100th
2008	-0.65	-0.66	-0.68	-0.69	0.02	-0.06	-0.18	-0.28	-0.77	-0.77	-0.78	-0.80
2008-2009	-0.68	-0.69	-0.71	-0.72	0.06	-0.02	-0.14	-0.24	-0.78	-0.78	-0.79	-0.80
2008-2010	-0.68	-0.68	-0.71	-0.71	0.07	-0.01	-0.14	-0.24	-0.79	-0.79	-0.80	-0.81
2008-2011	-0.69	-0.69	-0.71	-0.72	0.05	-0.04	-0.18	-0.28	-0.80	-0.80	-0.81	-0.83

Table 5.10: Correlations, scoring tables with performance improvement. Shown are the Pearson correlation coefficients between the points assigned by scoring tables and the actual performance improvements over the reference mark, based on the year 2008. The reference marks (the columns) are the 10th, 25th, 50th, and 100th best prior mark in each event. More negative correlations are better.

same reference performance during the same time period had a correlation coefficient of -0.69 with the IAAF scores and -0.80 with our scores. Our scores were more predictive in all cases that we tested. See tables 5.9 and 5.10 for more details.

5.3.5 Predictions for 2012 and beyond

Heading into 2012, an Olympic year, it is interesting to examine the predictions we might make. Most interesting, we feel, is the probability that a new world record is set. Thus, we have compiled in table 5.11 all of the current world records and we have sorted them by probability of being broken in 2012.

The probabilities range from less than $1/100,000$ for women's discus to almost certain (0.95) for women's steeplechase. Most of the world records (26 out of 48) have less than a 10% chance of being broken, a quarter (12) have less than a 1% chance, and only two—women's steeplechase and men's 110m hurdles—are likely to get broken. In both of these events, the world record was set recently, in 2008 in both cases, and there are many other recent marks that come close to the record. In particular, there are nine women's steeplechase performances from the past five years that are within ten seconds of the world record, including the record itself. There are seven marks (including the record) in the men's 110m hurdles from the past five five years that are within 0.05s of the world record. This suggests that in both of these events, with so many recent marks that are close to the record, it

Event	WR Mark	Athlete	Date	Prob of WR in 2012
womensDisc	76.8	Gabriele Reinsch	09.07.1988	$7.44x10^{-06}$
womens1500m	3:50.46	Qu Yunxia	11.09.1993	$9.24x10^{-05}$
mensHJ	2.45	Javier Sotomayor	27.07.1993	$7.09x10^{-04}$
womensLJ	7.52	Galina Chistyakova	11.06.1988	$8.56x10^{-04}$
womens3000m	8:06.11	Wang Junxia	13.09.1993	$1.62x10^{-03}$
mensHammer	86.74	Yuriy Syedikh	30.08.1986	$1.86x10^{-03}$
womensMarathon	2:15:25	Paula Radcliffe	13.04.2003	$2.52x10^{-03}$
mensJav	98.48	Jan Zelezny	25.05.1996	$5.11x10^{-03}$
womens400m	47.60	Marita Koch	06.10.1985	$5.14x10^{-03}$
mens1mile	3:43.13	Hicham El Guerrouj	07.07.1999	$5.28x10^{-03}$
womensShot	22.63	Natalya Lisovskaya	07.06.1987	$8.20x10^{-03}$
mensPV	6.14	Sergey Bubka	31.07.1994	$9.53x10^{-03}$
womens200m	21.34	Florence Griffith-Joyner	29.09.1988	$1.14x10^{-02}$
mensLJ	8.95	Mike Powell	30.08.1991	$1.49x10^{-02}$
mens400mH	46.78	Kevin Young	06.08.1992	$1.62x10^{-02}$
mens1500m	3:26.00	Hicham El Guerrouj	14.07.1998	$2.08x10^{-02}$
womens800m	1:53.28	Jarmila Kratochvilova	26.07.1983	$2.11x10^{-02}$
mens400m	43.18	Michael Johnson	26.08.1999	$2.28x10^{-02}$
mens4x400m	2:54.29	United States	22.08.1993	$2.35x10^{-02}$
mensShot	23.12	Randy Barnes	20.05.1990	$2.86x10^{-02}$
mensDisc	74.08	Jurgen Schult	06.06.1986	$3.13x10^{-02}$
womens100mH	12.21	Yordanka Donkova	20.08.1988	$3.17x10^{-02}$
mensTJ	18.29	Jonathan Edwards	07.08.1995	$3.84x10^{-02}$
womens100m	10.49	Florence Griffith-Joyner	16.07.1988	$3.92x10^{-02}$
womensPV	5.06	Yelena Isinbayeva	28.08.2009	$6.62x10^{-02}$
mens200m	19.19	Usain Bolt	20.08.2009	$8.60x10^{-02}$
mens3000m	7:20.67	Daniel Komen	01.09.1996	$1.08x10^{-01}$
womens10000m	29:31.78	Wang Junxia	08.09.1993	$1.14x10^{-01}$
mens100m	9.58	Usain Bolt	16.08.2009	$1.23x10^{-01}$
womensHalfMara.	65:50	Mary Keitany	18.02.2011	$1.32x10^{-01}$
mens4x100m	37.04	Jamaica	04.09.2011	$1.42x10^{-01}$
womens4x100m	41.37	German Democratic Republic	06.10.1985	$1.43x10^{-01}$
womens1mile	4:12.56	Svetlana Masterkova	14.08.1996	$1.51x10^{-01}$
womens4x400m	3:15.17	Soviet Union	01.10.1988	$1.54x10^{-01}$
mens800m	1:41.01	David Rudisha	29.08.2010	$1.61x10^{-01}$
mens5000m	12:37.35	Kenenisa Bekele	31.05.2004	$1.84x10^{-01}$
mens3000mSC	7:53.63	Saif Saeed Shaheen	03.09.2004	$2.35x10^{-01}$
womensTJ	15.50	Inessa Kravets	10.08.1995	$2.40x10^{-01}$
womensHJ	2.09	Stefka Kostadinova	30.08.1987	$2.91x10^{-01}$
womens400mH	52.34	Yuliya Pechonkina	08.08.2003	$3.32x10^{-01}$
mens10000m	26:17.53	Kenenisa Bekele	26.08.2005	$3.82x10^{-01}$
womensJav	72.28	Barbora Spotakova	13.09.2008	$3.84x10^{-01}$
mensMarathon	2:03:38	Patrick Makau	25.09.2011	$3.91x10^{-01}$
mensHalfMara.	58:23	Zersenay Tadese	21.03.2010	$3.96x10^{-01}$
womensHammer	79.42	Betty Heidler	21.05.2011	$4.72x10^{-01}$
womens5000m	14:11.15	Tirunesh Dibaba	06.06.2008	$4.76x10^{-01}$
mens110mH	12.87	Dayron Robles	12.06.2008	$6.62x10^{-01}$
womens3000mSC	8:58.81	Gulnara Galkina	17.08.2008	$9.52x10^{-01}$

Table 5.11: World record probabilities, 2012. Shown is a list of the current world records for all athletic events considered in this paper, sorted by the probability of being broken in 2012.

is more likely than not that a record will be set in 2012.

On the other end of the spectrum, those records least likely to get broken are some of the older records, with only 6 of the 25 toughest (according to table 5.11) records occurring in the past 15 years, whereas 17 of the 25 weakest records have occurred in the past 15 years. In the women's discus, where the record is least likely to get broken, no one has produced a mark in the top 100 in nearly 20 years. The women's 1500m run, which has the second toughest record, has seen no time within five seconds of the record in over ten years.

Notably, two events, the one mile run and the 3000m run (non-Olympic events), are contested less frequently than the rest, and therefore the probabilities of their records being broken are lower than if they were contested more often. For instance, the men's one mile world record is obviously—to any track and field fan—easier for a well-trained athlete to break than the 1500m world record, but the probability of the mile record actually being broken is lower since there are far fewer attempts.

5.4 Discussion

This paper has been an attempt to rigorously quantify what it means for an athletic performance to be "good", and, alternatively, what it means for a performance to be better than another performance, particularly if the two performances are in different events. We use primarily two alternative reasons why an observer of track and field might believe that a performance is good, restated from the introduction:

1. A performance is good if few athletes improve upon it, or
2. A performance is good if it is close to or improves upon the [previous] best performance.

In the introduction, we suggested that the 9.58s 100m dash that Usain Bolt ran in the 2009 World Champtionships might be one of the greatest athletics feats ever. But, we can see in table 5.11 that there are many records that are less likely to be broken next year than Usain Bolt's 9.58s. In fact, his own 200m world record (19.19) is one of them. On the other hand, of the world records that were set since the year 2000 (18 of them), these are the third and fourth least likely to be broken, so perhaps they are so impressive because they are among the best records of recent memory.

In addition to calculating probabilities of world records, we also calculated expected number of performances improving upon a given mark, expected best performances, and a set of scoring tables intended to be analagous to the IAAF scoring tables. Our results, particularly tables 5.9 and 5.10, show that our model can predict the levels of future performances with considerable success, and better than the most common method of performance scoring, the IAAF scoring tables. Given a set of performances or records, we can predict which ones will be broken, how many times, and by how much, and these predictions have a Pearson correlation coefficient of over 0.8 in many cases with actual future outcomes. Our scoring tables, which are derived from the expected number of annual performances exceeding a given mark, outperformed the IAAF scoring tables for two different prediction types, each with four sets of reference marks and four time periods, giving 32 cases wherein our predictions correlated more highly in every case.

The keys to the success, we believe, are the large amount of data used in model fitting and the probabilistic approach. Past scoring methods typically have used a fixed number of top performances—in some cases very few—such as the top ten or one hundred within a particular time period; we wanted to avoid this restriction and use all available data to compute actual probabilities. In general, more data is better, though admittedly there were some outlying circumstances in the past when, for example, performance enhancing drugs have been used without detection, or marks were set under other questionable circumstances. One glaring example of this is the fact that no woman has produced a top-100 mark in the shot put or the discus in the past ten years. Likely because of these questionable performances, we have found that the most accurate way to predict the performances of the next year is by fitting the model only to recent data. Another example of a negative shift in performance is the recent switch to all-women road races, particularly in the marathon. Paula Radcliffe's marathon world record is one of the best marks in athletics, but it was set with men running alongside the women. It has been ruled (by the IAAF) that mixed-sex races are no longer eligible for women's records, but it seems that previous marks will be allowed to stand. Though not previously considered cheating, male pacers can help women significantly, and in their absence we have indeed seen a drop in the quality of women's marathon times, as most major marathons have in the past few years switched to separate men's and women's races. These shifts in performance level are a problem we might address in future research. It is reasonable to assume that performance levels improve over time due to improved training and technique, and any large-scale decline is the result of a reduction in the prevalence of performance enhancing drugs or other forms of performance aid or cheating. There are a number of ways we might detect and remove—or otherwise take into account—these questionable performances, possibly using robust statistics or parameter optimization techniques. In addition, other probability distributions might also be considered if they seem to fit the data.

In a more general sense, it would likely help the predictive ability of the model if time were included as a contributing variable. Modeling general performance changes over time would give us further abilities to discuss and describe the history of athletics, such as in detecting or predicting eras of great improvement or change and also in modeling the maturity of a event, in the sense that, for example, the women's steeplechase isn't quite mature yet since it has been an Olympic event since only 2008 and its records still fall quite often.

Lastly, the type of analysis demonstrated in this paper need not be limited to athletics. Any standardized competition with a large number of performances that are either normally or log-normally distributed can be modeled in this way. Swimming and rowing come to mind, though those are more dependent on technology than athletics and thus may be more difficult to model. All in all, a probabilistic approach to studying sports performances seems to be a practical and valuable tool in examining the history and predicting the future of sport.

Bibliography

Bar-Joseph, Z. (2004). Analyzing time series gene expression data. *Bioinformatics*, 20(16).

Bar-Joseph, Z., Gitter, A., and Simon, I. (2012). Studying and modelling dynamic biological processes using time-series gene expression data. *Nature Reviews Genetics*, 13(8):552–564.

Barrett, T., Troup, D. B., Wilhite, S. E., Ledoux, P., Rudnev, D., Evangelista, C., Kim, I. F., Soboleva, A., Tomashevsky, M., Marshall, K. A., Phillippy, K. H., Sherman, P. M., Muertter, R. N., and Edgar, R. (2009). NCBI GEO: archive for high-throughput functional genomic data. *Nucleic Acids Research*, 37(suppl 1):D885–D890.

Beal, M. J. (2003). *Variational algorithms for approximate Bayesian inference.* PhD thesis, Gatsby Computational Neuroscience Unit, University College London.

Beal, M. J., Falciani, F., Ghahramani, Z., Rangel, C., and Wild, D. L. (2005). A Bayesian approach to reconstructing genetic regulatory networks with hidden factors. *Bioinformatics*, 21(3):349–356.

Betel, D., Koppal, A., Agius, P., Sander, C., and Leslie, C. (2010). Comprehensive modeling of microRNA targets predicts functional non-conserved and non-canonical sites. *Genome Biology*, 11(8):R90+.

Cameron, D. F. (1998). Time equivalence model. web page. Accessed on 2011.10.01.

Carmona-Saez, P., Chagoyen, M., Tirado, F., Carazo, J. M., and Pascual-Montano, A. (2007). GENECODIS: a web-based tool for finding significant concurrent annotations in gene lists. *Genome biology*, 8(1):R3+.

Chen, L., Li, C., Zhang, R., Gao, X., Qu, X., Zhao, M., Qiao, C., Xu, J., and Li, J. (2011). miR-17-92 cluster microRNAs confers tumorigenicity in multiple myeloma. *Cancer letters*, 309(1):62–70.

Cheng, C. and Li, L. M. (2008). Inferring MicroRNA Activities by Combining Gene Expression with MicroRNA Target Prediction. *PLoS ONE*, 3(4):e1989+.

Daniels, J. and Gilbert, J. (1979). *Oxygen Power: Performance Tables for Distance Runners.* J Daniels and J Gilbert.

Dweep, H., Sticht, C., Pandey, P., and Gretz, N. (2011). miRWalk–database: prediction of possible miRNA binding sites by "walking" the genes of three genomes. *Journal of biomedical informatics*, 44(5):839–847.

Enright, A., John, B., Gaul, U., Tuschl, T., Sander, C., and Marks, D. (2003). MicroRNA targets in Drosophila. *Genome Biology*, 5(1):R1+.

Ernst, J., Nau, G. J., and Bar-Joseph, Z. (2005). Clustering short time series gene expression data. *Bioinformatics*, 21(suppl 1):i159–i168.

Friedman, R. C., Farh, K. K., Burge, C. B., and Bartel, D. P. (2009). Most mammalian mRNAs are conserved targets of microRNAs. *Genome Research*, 19(1):92–105.

Fujita, A., Sato, J., Malpartida, H. G., Yamaguchi, R., Miyano, S., Sogayar, M., and Ferreira, C. (2007). Modeling gene expression regulatory networks with the sparse vector autoregressive model. *BMC Systems Biology*, 1(1):39+.

Gäken, J., Mohamedali, A. M., Jiang, J., Malik, F., Stangl, D., Smith, A. E., Chronis, C., Kulasekararaj, A. G., Thomas, N. S., Farzaneh, F., Tavassoli, M., and Mufti, G. J. (2012). A functional assay for microRNA target identification and validation. *Nucleic acids research.*

Garcia, D. M., Baek, D., Shin, C., Bell, G. W., Grimson, A., and Bartel, D. P. (2011). Weak seed-pairing stability and high target-site abundance decrease the proficiency of lsy-6 and other microRNAs. *Nature structural & molecular biology*, 18(10):1139–1146.

Gardner, J. B. and Purdy, J. G. (1970). Computer generated track scoring tables. *Medicine and science in sports*, 2(3):152–161.

Gelman, A., Carlin, J., Stern, H., and Rubin, D. (2004). *Bayesian Data Analysis*. Chapman & Hall, second edition edition.

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

Geyer., C. J. (2010). *mcmc: Markov Chain Monte Carlo*. R package version 0.8.

Godsey, B. (2012). Comparing and Forecasting Performances in Different Events of Athletics Using a Probabilistic Model. *Journal of Quantitative Analysis in Sports*, 8(2).

Godsey, B. (2013a). Discovery of miR-mRNA interactions via simultaneous Bayesian inference of gene networks and clusters using sequence-based predictions and expression data. *Journal of Integrative Bioinformatics*, 10(1):227.

Godsey, B. (2013b). Improved Inference of Gene Regulatory Networks through Integrated Bayesian Clustering and Dynamic Modeling of Time-Course Expression Data. *PLoS ONE*, 8(7):e68358+.

Godsey, B., Heiser, D., and Civin, C. (2012). Inferring MicroRNA Regulation of mRNA with Partially Ordered Samples of Paired Expression Data and Exogenous Prediction Algorithms. *PLoS ONE*, 7(12):e51480+.

Greenfield, A., Madar, A., Ostrer, H., and Bonneau, R. (2010). DREAM4: Combining Genetic and Dynamic Information to Identify Biological Networks and Dynamical Models. *PLoS ONE*, 5(10):e13397+.

Griffiths-Jones, S., Grocock, R. J., van Dongen, S., Bateman, A., and Enright, A. J. (2006). miRBase: microRNA sequences, targets and gene nomenclature. *Nucleic Acids Research*, 34(suppl 1):D140–D144.

Griffiths-Jones, S., Saini, H. K., van Dongen, S., and Enright, A. J. (2008). miRBase: tools for microRNA genomics. *Nucleic Acids Research*, 36(suppl 1):D154–D158.

Grimson, A., Farh, K. K., Johnston, W. K., Garrett-Engele, P., Lim, L. P., and Bartel, D. P. (2007). MicroRNA Targeting Specificity in Mammals: Determinants beyond Seed Pairing. *Mol Cell*, 27(1):91–105.

Gustafsson, M., Hornquist, M., and Lombardi, A. (2005). Constructing and Analyzing a Large-Scale Gene-to-Gene Regulatory Network-Lasso-Constrained Inference and Biological Validation. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 2(3):254–261.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109.

Heuck, C., Szymonifka, J., Hansen, E., Shaughnessy, J. D., Usmani, S., Van Rhee, F., Anaissie, E., Nair, B., Waheed, S., Alsayed, Y., Petty, N., Bailey, C., Epstein, J., Hoering, A., Crowley, J. J., and Barlogie, B. (2012). Thalidomide in Total Therapy 2 Overcomes Inferior Prognosis of Myeloma with Low Expression of the Glucocorticoid Receptor Gene NR3C1. *Clinical Cancer Research*.

Hirose, O., Yoshida, R., Imoto, S., Yamaguchi, R., Higuchi, T., Charnock-Jones, D. S., Print, C., and Miyano, S. (2008). Statistical inference of transcriptional module-based gene networks from time course gene expression profiles by using state space models. *Bioinformatics*, 24(7):932–942.

Huang, J. C., Babak, T., Corson, T. W., Chua, G., Khan, S., Gallie, B. L., Hughes, T. R., Blencowe, B. J., Frey, B. J., and Morris, Q. D. (2007a). Using expression profiling data to identify human microRNA targets. *Nature Methods*, 4(12):1045–1049.

Huang, J. C., Frey, B. J., and Morris, Q. D. (2008). Comparing sequence and expression for predicting microRNA targets using GenMiR3. *Pacific Symposium on Biocomputing*, pages 52–63.

Huang, J. C., Morris, Q. D., and Frey, B. J. (2007b). Bayesian Inference of MicroRNA Targets from Sequence and Expression Data. *Journal of Computational Biology*, 14(5):550–563.

Husmeier, D. (2003). Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics*, 19(17):2271–2282.

International Association of Athletics Federations (2001). IAAF scoring tables for combined events. pdf document.

Jayaswal, V., Lutherborrow, M., Ma, D. D. F., and Hwa Yang, Y. (2009). Identification of microRNAs with regulatory potential using a matched microRNA-mRNA time-course data. *Nucleic Acids Research*, 37(8):e60.

John, B., Enright, A. J., Aravin, A., Tuschl, T., Sander, C., and Marks, D. S. (2004). Human MicroRNA Targets. *PLoS Biol*, 2(11):e363+.

Jolliffe, I. (2005). *Principal component analysis*. Wiley Online Library.

Jopling, Norman, and Sarnow, P. (2006). Positive and Negative Modulation of Viral and Cellular mRNAs by Liver-specific MicroRNA miR-122. *Cold Spring Harbor Symposia on Quantitative Biology*, 71:369–376.

Kanehisa, M., Araki, M., Goto, S., Hattori, M., Hirakawa, M., Itoh, M., Katayama, T., Kawashima, S., Okuda, S., Tokimatsu, T., and Yamanishi, Y. (2008). KEGG for linking genomes to life and the environment. *Nucleic acids research*, 36(Database issue):D480–484.

Kim, S. Y., Imoto, S., and Miyano, S. (2003). Inferring gene networks from time series microarray data using dynamic Bayesian networks. *Brief Bioinform*, 4(3):228–235.

Klemm, S. L. (2008). Causal Structure Identification in Nonlinear Dynamical Systems. Master's thesis, Department of Engineering, University of Cambridge, Cambridge, UK.

Kozomara, A. and Griffiths-Jones, S. (2011). miRBase: integrating microRNA annotation and deep-sequencing data. *Nucleic Acids Research*, 39(suppl 1):D152–D157.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Larsson, P. (2011). Alltime-athletics.com. Web site. Data as of 2011.10.01.

Lèbre, S. (2009). Inferring Dynamic Genetic Networks with Low Order Independencies. *Statistical Applications in Genetics and Molecular Biology*, 8(1):1–38.

Lee, T.-W. (1998). *Independent component analysis*. Springer.

Lewis, B. P., Burge, C. B., and Bartel, D. P. (2005). Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microRNA targets. *Cell*, 120(1):15–20.

Lionetti, M., Biasiolo, M., Agnelli, L., Todoerti, K., Mosca, L., Fabris, S., Sales, G., Deliliers, G. L. L., Bicciato, S., Lombardi, L., Bortoluzzi, S., and Neri, A. (2009). Identification of microRNA expression patterns and definition of a microRNA/mRNA regulatory network in distinct molecular groups of multiple myeloma. *Blood*, 114(25):e20–26.

Mackay, D. J. (1999). Comparison of Approximate Methods for Handling Hyperparameters. *Neural Comp.*, 11(5):1035–1068.

Madych, W. (1991). Solutions of underdetermined systems of linear equations. *Lecture Notes-Monograph Series*, pages 227–238.

Marbach, D., Prill, R. J., Schaffter, T., Mattiussi, C., Floreano, D., and Stolovitzky, G. (2010). Revealing strengths and weaknesses of methods for gene network inference. *PNAS*, 107(14):6286–6291.

Marbach, D., Schaffter, T., Mattiussi, C., and Floreano, D. (2009). Generating Realistic In Silico Gene Networks for Performance Assessment of Reverse Engineering Methods. *Journal of Computational Biology*, 16(2):229–239.

McMillan, G. (2011). The McMillan running calculator. Web page.

Muniategui, A., Nogales-Cadenas, R., Vázquez, M., Aranguren, Agirre, X., Luttun, A., Prosper, F., Pascual-Montano, A., and Rubio, A. (2012). Quantification of miRNA-mRNA Interactions. *PLoS ONE*, 7(2):e30766+.

Muniategui, A., Pey, J., Planes, F., and Rubio, A. (2013). Joint analysis of miRNA and mRNA expression data. *Briefings in Bioinformatics*, 14(3):263–278.

Nogales-Cadenas, R., Carmona-Saez, P., Vazquez, M., Vicente, C., Yang, X., Tirado, F., Carazo, J. M. M., and Pascual-Montano, A. (2009). GeneCodis: interpreting gene lists through enrichment analysis and integration of diverse biological information. *Nucleic acids research*, 37(Web Server issue):W317–322.

O'Donnell, K. A., Wentzel, E. A., Zeller, K. I., Dang, C. V., and Mendell, J. T. (2005). c-Myc-regulated microRNAs modulate E2F1 expression. *Nature*, 435(7043):839–843.

Penfold, C. A. and Wild, D. L. (2011). How to infer gene networks from expression profiles, revisited. *Interface Focus*, 1(6):857–870.

Pichiorri, F., Suh, S.-S., Rocci, A., De Luca, L., Taccioli, C., Santhanam, R., Zhou, W., Benson, D. M., Hofmainster, C., Alder, H., Garofalo, M., Di Leva, G., Volinia, S., Lin, H.-J., Perrotti, D., Kuehl, M., Aqeilan, R. I., Palumbo, A., and Croce, C. M. (2010). Downregulation of p53-inducible microRNAs 192, 194, and 215 Impairs the p53/MDM2 Autoregulatory Loop in Multiple Myeloma Development. *Cancer Cell*, 18(4):367–381.

Plummer, M., Best, N., Cowles, K., and Vines, K. (2006). Coda: Convergence diagnosis and output analysis for mcmc. *R News*, 6(1):7–11.

Prill, R. J., Marbach, D., Saez-Rodriguez, J., Sorger, P. K., Alexopoulos, L. G., Xue, X., Clarke, N. D., Altan-Bonnet, G., and Stolovitzky, G. (2010). Towards a Rigorous Assessment of Systems Biology Models: The DREAM3 Challenges. *PLoS ONE*, 5(2):e9202+.

Prill, R. J., Saez-Rodriguez, J., Alexopoulos, L. G., Sorger, P. K., and Stolovitzky, G. (2011). Crowdsourcing Network Inference: The DREAM Predictive Signaling Network Challenge. *Sci. Signal.*, 4(189):mr7+.

R Development Core Team (2009). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria.

Riegel, P. (1977). Time predicting. *Runner's World.*

Schliep, A., Schonhuth, A., and Steinhoff, C. (2003). Using hidden Markov models to analyze gene expression time course data. *Bioinformatics*, 19(suppl 1):i255–i263.

Sethupathy, P., Megraw, M., and Hatzigeorgiou, A. G. (2006). A guide through present computational approaches for the identification of mammalian microRNA targets. *Nature Methods*, 3(11):881–886.

Shiraishi, Y., Kimura, S., and Okada, M. (2010). Inferring cluster-based networks from differently stimulated multiple time-course gene expression data. *Bioinformatics*, 26(8):1073–1081.

Sivriver, J., Habib, N., and Friedman, N. (2011). An integrative clustering and modeling algorithm for dynamical gene expression data. *Bioinformatics*, 27(13):i392–i400.

Smyth, G. and Speed, T. (2003). Normalization of cDNA microarray data. *Methods*, 31:265–273.

Spiriev, B. (2008). IAAF scoring tables of athletics: 2008 revised edition. pdf document.

Spiriev, B. and Spiriev, A. (2011). IAAF scoring tables of athletics: 2011 revised edition. pdf document.

Stingo, F. C., Chen, Y. A., Vannucci, M., Barrier, M., and Mirkes, P. E. (2010). Spike and slab variable selection: Frequentist and Bayesian strategies. *The Annals of Applied Statistics*, 4(4):2024–2048.

Tanzer, A. and Stadler, P. F. (2004). Molecular Evolution of a MicroRNA Cluster. *Journal of Molecular Biology*, 339(2):327–335.

Teschendorff, A. E., Wang, Y., Barbosa-Morais, N. L., Brenton, J. D., and Caldas, C. (2005). A variational Bayesian mixture modelling framework for cluster analysis of gene-expression data. *Bioinformatics*, 21(13):3025–3033.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.

Tokumoto, M., Fujiwara, Y., Shimada, A., Hasegawa, T., Seko, Y., Nagase, H., and Satoh, M. (2011). Cadmium toxicity is caused by accumulation of p53 through the down-regulation of Ube2d family genes in vitro and in vivo. *The Journal of toxicological sciences*, 36(2):191–200.

Vasudevan, S., Tong, Y., and Steitz, J. A. (2007). Switching from repression to activation: microRNAs can up-regulate translation. *Science (New York, N.Y.)*, 318(5858):1931–1934.

Wang, L., Oberg, A. L., Asmann, Y. W., Sicotte, H., McDonnell, S. K., Riska, S. M., Liu, W., Steer, C. J., Subramanian, S., Cunningham, J. M., Cerhan, J. R., and Thibodeau, S. N. (2009). Genome-Wide Transcriptional Profiling Reveals MicroRNA-Correlated Genes and Biological Processes in Human Lymphoblastoid Cell Lines. *PLoS ONE*, 4(6):e5878+.

Welling, M. and Kurihara, K. (2006). Bayesian K-means as a " maximization-expectation" algorithm. In *Sixth SIAM International Conference on Data Mining*, volume 22, pages 474–478.

Wikipedia (2013). William sealy gossett — Wikipedia, the free encyclopedia. [Online; accessed 13-August-2013].

Winn, J. M. (2003). *Variational Message Passing and its Applications*. PhD thesis, St Johns College, Cambridge, Cambridge, England.