Technical Report 1174-v1, Statistics Department, University of Wisconsin, Madison, May 17, 2013.

A multi-functional analyzer uses parameter constraints to improve the efficiency of model-based gene-set analysis

ZHISHI WANG, QIULING HE

Department of Statistics, University of Wisconsin, Madison, WI, USA

BRET LARGET

Departments of Statistics and of Botany, University of Wisconsin, Madison, WI, USA

MICHAEL A. NEWTON*

Departments of Statistics and of Biostatistics and Medical Informatics, University of Wisconsin, Madison, WI, USA

Summary

We develop a model-based methodology for integrating gene-set information with an experimentally-derived gene list. The methodology uses a previously reported sampling model, but takes advantage of natural constraints in the high-dimensional discrete parameter space in order to work from a more structured prior distribution than is currently available (MGSA). We show how the natural constraints are expressed in terms of linear inequality constraints within a set of binary latent variables. We show how the MGSA prior gives low probability to these constraints in complex systems, such as the Gene Ontology (GO), thus reducing the efficiency of statistical inference. We develop two computational advances to enable posterior inference within the constrained parameter space: one using integer linear programming for optimization, and one using a penalized Markov chain sampler. Preliminary numerical experiments demonstrate the

*To whom correspondence should be addressed (newton@stat.wisc.edu). QH is currently employed by Novartis Pharmaceuticals.

utility of the new methodology for a multivariate integration of genomic data with GO or related information systems.

Key words: gene-set enrichment; Bayesian analysis; integer linear programming

1. Introduction

The gene list is a recurring data structure in experimental genomics. We have in mind situations where experimental results amount to a collection of genes measured to have some property. Examples include: RNA expression studies, in which the property might be differential expression of the gene between two cell types; genome-wide RNA knock-down studies, in which the property is significant phenotypic alteration caused by RNA interference; chromatin studies recording genes in the vicinity of transcription factor binding sites or having of certain epigenetic marks. In all cases, the reported gene list is really the result of inference from more basic experimental data. These more basic data may be available to support subsequent analyses, but we are concerned with the important and relatively common case in which the gene list itself is the primary data set brought forward for analysis.

The statistical question of central importance in the present paper is how to interpret the experimental gene list in the context of pre-existing biological knowledge about the functional properties of all genes, as these exogenous data are recorded in database systems, notably Gene Ontology (GO), the Kyoto Encyclopedia (KEGG), the reactome, among others. For us, exogenous data form a collection of gene sets, with each set equaling those genes previously determined, by some evidence, to have a specific biological property. At writing, for example, the full GO collection contains 16,527 sets (GO terms) annotating 17,959 human genes (Bioconductor package `org.Hs.eg.db`, version 2.8.0). Needless to say, genes are typically annotated to multiple gene sets (median 7 sets per gene among the genes annotated to sets which contain between 3 and 30 genes, for example), covering all sorts of functional properties. The task of *gene-set analysis* is to efficiently interpret the functional content of an experimentally-derived gene list by somehow integrating these endogenous and exogenous data sources.

Our starting point is an exciting development in the methodology of gene-set analysis. Model-based gene-set analysis (MGSA) expresses gene-level indicators of presence on the experimental gene list as Bernoulli trials whose success probabilities are determined in a simple way by latent activity states of binary variables associated with the gene sets (Bauer *et al.*, 2010, 2011). Inference seeks to identify the *active* gene sets, as these represent functional drivers of the experimental data. Inference is computationally difficult because the activity state of a given gene set depends not only on experimental data for genes in that set, but also on the activity states of all other gene sets that annotate these same genes. And since none of these activity states is really known there is a serious problem. MGSA overcomes the problem through Bayesian inference implemented with an efficient Markov chain Monte Carlo (MCMC) sampler, and thus provides marginal posterior probabilities that each gene set is in the active state. The MGSA methodology is compelling. Because it treats all gene sets in the collection simultaneously, it provides a truly multivariate analysis of the exogenous data source, where most available approaches are univariate (one set at a time). Where set/set overlaps are a nuisance in most gene-set methodologies, MGSA utilizes them directly in modeling and inference. This accounts for pleitropy, that genes have multiple biological functions, reduces the risk of spurious associations, and leads to cleaner output whereby a typical list of gene sets inferred to be active is simpler and exhibits less redundancy than in standard univariate analyses (Bauer *et al.*, 2010; Newton *et al.* 2012).

Our analysis reveals a feature of MGSA that adversely affects its statistical properties. In ever-denser collections of gene sets, the MGSA prior distribution puts more and more mass on logically inconsistent joint activity states. As a result, data need to work ever harder, so to speak, to overcome this mis-guided prior probability. The effect is tangible; for a given amount of data, fewer truly activated gene sets are inferred to be active, compared to what is achievable with an alternative formulation. We propose a new methodology, the multi-functional analyzer (MFA), which aims to improve the statistical efficiency of MGSA. It uses two computational advances that enable posterior inference in the high-dimensional constrained space of joint activity states. One is an efficient MCMC sampling scheme constructed by penalizing the log-posterior in the unconstrained space; and one is a discrete optimizaton scheme that translates the inference problem into an integer linear programming (ILP) problem.

We note that inference about gene-set activity states may be interesting from the general perspective of high-dimensional statistics. Typically, dependence among data from different inference units (sets, in this case) is considered a nuisance and testing aims to identify non-null units (active sets) by some methodology that is robust to dependencies, since these dependencies are often difficult to estimate from available data. In the present context dependencies are complicated but explicit, and inference benefits by using them to advantage. Finally, we also note that the probability model underlying our methodology – the *role model* – has potential utility in other domains of application. It provides a simple way to relate data collected at one level (genes, in this case) to inference units that are unordered collections of the former (gene sets, in this case).

2. Role model

2.1 *Model*

Following the description in Newton *et al.*, (2012), we have a finite number of *parts* p and a finite number of *wholes* w, where each whole is an unordered set of parts. The incidence matrix $I = (I_{p,w})$ is assumed to be known, where $I_{p,w} = 1$ if and only if $p \in w$. The intended correspondence is that genes are parts and gene sets (i.e., functional categories) are wholes. The matrix I encodes a full collection of gene sets. We will have measured data on the parts and aim to make inference on properties of the wholes.

The experimentally-derived gene list may be viewed as a vector of Bernoulli trials $X = (X_p)$, with $X_p = 1$ if and only if part (gene) p is on the list. First proposed in Bauer *et al.* (2010), the role model describes the joint distribution of X in terms of latent binary (0/1) activity variables $Z = (Z_w)$ and by part-level activities induced by them:

$$A_p = 1 \quad \iff \quad Z_w = 1 \text{ for any } w \text{ with } p \in w,$$

or equivalently,

$$A_p = \max_{w:p\in w} Z_w. \tag{2.1}$$

This conveys the simple assumption that a part is activated if it is in any whole that is activated. For two rate parameters $\alpha, \gamma \in (0,1)$, with $\alpha < \gamma$, the model for X entails mutually independent components (conditionally on latent activities), with

$$X_p \sim \text{Bernoulli} \begin{cases} \alpha & \text{if } A_p = 0 \\ \gamma & \text{if } A_p = 1 \end{cases} \tag{2.2}$$

Simply, activated parts (*i.e.*, those with $A_p = 1$) are delivered to the list at a higher rate than inactivated parts. A key feature of the model is that a part (gene) is activated by virtue of any one of its functional *roles*; this implies that a gene may be activated and yet be part of a functional category that is inactivated, which is in contrast to most other gene-set inference methods (*e.g.*, Sartor *et al.*, 2009), and which provides for a fully multivariate analysis of the gene list. In Bauer *et al.* (2010) it is further assumed, for the sake of Bayesian analysis, that uncertainty in whole-level activities is represented with a single rate parameter $\pi \in (0, 1)$:

$$Z_w \sim_{i.i.d.} \text{Bernoulli}(\pi). \tag{2.3}$$

Taken together, the model (2.2) and the prior (2.3) determine a joint posterior for Z given X. The `R` package `MGSA` (model-based gene set analysis) reports MCMC-computed marginal posterior probabilities $P(Z_w = 1|X)$, also integrating uncertainty in the system parameters α, γ, and π, and thus provides a useful ranking of the wholes (Bauer *et al.* 2011).

In addition to the system incidence matrix I, a useful data structure for computations turns out to be the bipartite graph $\mathcal{G}$, having whole nodes and part nodes, and an edge between w and p if and only if $I_{p,w} = 1$ (i.e., iff $p \in w$).

2.2 *Activation Hypothesis*

As defined above, the role model allows that a whole can be inactive while all of its parts are active. This can happen because of overlap among the wholes. Specifically, if w is contained in the union of other wholes $\{w'\}$ then all $Z_{w'} = 1$ will force $A_p = 1$ for all $p \in w$, regardless of the value of Z_w. This rather odd situation calls into question the meaning of *active* and what we might realistically expect can be inferred from data. Indeed the issue is related to identifiability of the activity vector Z, since it shows that distinct Z vectors may produce the same part-level activity vector $A = (A_p)$. (In the case above, switching Z_w from 0 to 1 does not change A.) The mapping $Z \longrightarrow A$ given by (2.1) is not necessarily invertible, depending on the system as defined in I. Lack of identifiability would not necessarily create difficulty in a Bayesian analysis, however in the present case we are specifically interested in inferring the activity states of the gene sets and prioritizing these sets, and so it stands to reason that we ought to confer a real, if still only model-based, meaning on the activities.

When *activity* is defined more fully, there is a simple solution to the problem. The *activation hypothesis* asserts that a set of parts is active if and only if all parts in the set are active. It was shown previously (Newton *et al.*, 2012):

Theorem 2.1 Under the activation hypothesis (AH), the mapping $Z \longrightarrow A$ defined by

$$A_p = \max_{w:p\in w} Z_w$$

is invertible, with inverse $A \longrightarrow Z$

$$Z_w = \min_{p:p\in w} A_p.$$

The inverse mapping is simply that a whole is inactive if and only if any of its parts is inactive. So the odd case at the beginnng of the section cannot occur under AH; if all parts are active, then $Z_w = 1$ must hold. Further, with parameters α and γ fixed, the Z vector is identifiable under AH, since different Z vectors necessarily give different probability distributions to data X.

The first contribution of the present work is to show that the activation hypothesis is equivalent to a set of linear inequality constraints on the activity variables. The finding is useful for posterior inference computations. We prove in Section 6 that:

Theorem 2.2 AH holds if and only if all of the following hold:

1. $Z_w \leqslant A_p$ for all p, w with $p \in w$

2. $A_p \leqslant \sum_{w:p\in w} Z_w$ for all p

3. $\sum_{p:p\in w} (Z_w - 2A_p + 2) \geqslant 1$ for all w

Evidently, the i.i.d. Bernoulli prior (2.3) does not respect AH in the sense that vectors Z which violate AH have positive prior probability. In simple systems such violation may be innocuous.

We provide evidence that in the complex systems such as Gene Ontology, this violation creates a substantial loss of statistical efficiency. We note first that alternative prior specifications are available that respect AH. A simple one is to condition prior (2.3) on the AH event, namely

$$P\left(Z=z\right)=\left(\frac{1}{c}\right)\pi^{\sum_w z_w}(1-\pi)^{\sum_w (1-z_w)} \qquad \text{if } z \text{ satisfies AH,} \tag{2.4}$$

otherwise $P(Z=z)=0$, where c is the probability, in prior (2.3), that Z satisfies AH, and z is a vector of binaries representing a possible realization of Z. In other words, with subscript '1' for the i.i.d. prior (2.3) and '2' for prior (2.4), we have: $P_2\left(Z=z\right)=P_1\left(Z=z|\text{AH}\right)$. Upon conditioning, the (Z_w) are not necessarily either mutually independent or identically distributed.

3. Statistical Properties

The role of the prior distribution in Bayesian analysis has surely been the subject of considerable debate. On one hand it helps by 'regularizing' inference, especially in high dimensions. On the other hand, data need to work against it to produce inferences that trade off empirical characteristics with prior assumptions. A fact of relevance to the present problem is that gene-list data must work against either prior (2.3 or 2.4) to deliver an inferred list of activated gene sets. For two Bayesian analysts, one using prior (2.3) and the other using prior (2.4), the true state is ascribed different prior mass. The ratio of these masses, ρ, represents the extra effort needed to be done by the data to overcome prior (2.3) compared to prior (2.4):

$$\rho=\frac{P_2\left(Z=z_{\text{true}}\right)}{P_1\left(Z=z_{\text{true}}\right)}=\frac{P_1\left(Z=z_{\text{true}}|\text{AH}\right)}{P_1\left(Z=z_{\text{true}}\right)}=\frac{1}{P_1\left(\text{AH}\right)}\geqslant 1. \tag{3.5}$$

Here we have used the particular structure of prior (2.4) and also the assumption that z_{true} satifies AH. If z_{true} did not satisfy AH, the target of inference it would be beyond the realm of any gene-level data set to estimate, owing to lack of identifiability. Indeed, it is difficult to see what meaning could be ascribed to z_{true} in that case. The observation to be gained from (3.5) is that the probability of AH under the i.i.d. prior affects the efficiency of inference. In systems where that probability is very small, there is reason to believe that improved inferences are possible. As to the precise effect of ignoring AH, that depends on the particular system I, the true activation state, and the system parameters α and γ. What our initial investigation finds is that a truly activated whole w may tend to have $P_1(Z_w = 1|X)$ smaller than $P_2(Z_w = 1|X)$, and if so the P_1 inference is too conservative.

Whether or not AH holds for a given state Z may be assessed by calculating the part-level activities A and then checking Theorem 2.2. Alternatively, we consider whole-level *violation* variables (V_w). These Bernoulli trials are defined:

$$V_w = \begin{cases} 1 & \text{if } Z_w = 0 \text{ and if for all } p \in w, \exists w' \text{ with } p \in w' \text{ and } Z_{w'} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

The probability, under P_1, that Z satisfies AH is equivalent to the probability of no violations, that is:

$$P_1(AH) = P_1\left(V_w = 0,\ \forall w\right), \tag{3.7}$$

and so AH probability might be approachable by considering the violation variables. Except in stylized examples we do not expect these variables to be mutually independent; indeed they may have a complicated dependence induced by overlaps of the wholes, and hence direct calculation of P_1(AH) is intractable. However, the expectations of V_w are readily computable for a given system, either by Monte Carlo or by a more sophisticated algorithm (Section 6). Considering the Chen-Stein result for Poisson approximations, we conjecture that $-\log P_1(AH)$ is approximately equal to $E_1\left(\sum_w V_w\right)$, though we have not been able to guarantee an error on this approximation (*c.f.* Arratia *et al.*, 1990).

Figure 1 charts the expected value $E_1\left(\sum_w V_w\right)$ over three recent versions of the Gene Ontology system, for $\pi = 1/100$. For concreteness it focuses on GO terms holding between 5 and 20 genes (for which an exact calculation of the expectation is feasible), though the key finding is not sensitive to that restriction, as evidenced by Monte Carlo computations (not shown). As one might expect by the increasing density and complexity of GO, the expected number of AH violations increases. This may very well reflect the fact that $P_1(AH)$ is decreasing over time, which indicates to us that ignoring AH is becoming an ever greater problem for gene-set analysis.

In terms of modeling assumptions there is no additional cost to accounting for AH in the Bayesian analysis; the cost is purely computational, since inference must now deal with the constraints imposed by AH on the space of latent activities. The next sections describe two computational advances that address the problem.

4. Decoding functional signals via constrained optimization

4.1 *MAP via ILP*

Decoding a discrete signal is frequently accomplished by algorithms that compute the parameter state having the highest posterior mass: the MAP estimate. Although limited as a posterior summary when noise levels are high, the MAP estimate contains useful multivariate information (Carvalho and Lawrence, 2009). Our representation of model-based gene set analysis reveals that under model (2.2) and prior (2.4), the log posterior is linear in the joint collection of whole and part activity variables. Up to an additive constant, this log posterior is:

$$\begin{aligned} l(Z, A) = &\sum_w \{Z_w \log(\pi) + (1 - Z_w)\log(1-\pi)\} + \\ &\sum_p \{A_p\left[x_p \log(\gamma) + (1 - x_p)\log(1-\gamma)\right] + (1 - A_p)\left[x_p \log(\alpha) + (1 - x_p)\log(1-\alpha)\right]\}, \end{aligned} \tag{4.8}$$

where x_p is the realized value of the gene-list indicator X_p, and α, γ, and π are system parameters, which are considered fixed in the present calculation. Considering Theorem 2.2, finding the MAP estimate $(\hat{Z}, \hat{A})$ amounts to maximizing a linear function in discrete variables subject to linear inequality constraints. As such, it fits naturally into the domain of *integer linear programming* (ILP), an active subfield of optimization. Our computations take advantage of ILP software available in the GNU Linear Programming Kit (`www.gnu.org/software/glpk`) through its interface with `R` (`cran.R-project.org/package=Rglpk`). We employed a series of basic code checks to assure our implementation worked in: (1) simple examples where the MAP estimate is computable by other means; and (2) limiting situations where X_p was Binomial having a high sample size, and thus where the MAP estimate must converge to the true activity state.

The reconstruction $\hat{Z}$ obtained through this optimization holds an estimate of the activated and inactivated gene sets. We refer to the overall method as the *multi-functional analyzer* (MFA), and specifically MFA-ILP to refer to the posterior mode computed by ILP. We note that by invertibility of the mapping $Z \longrightarrow A$ under AH, the log-posterior l could be expressed either as a function of Z only or as a function of A only; however in neither reduced case would l be linear in the input variables. Moreover, in neither reduced case could the constraints be expressed as linear inequality constraints. By expanding the domain we formulate the constrained optimization as an integer linear program.

4.2 *Numerical experiments*

In each experiment reported below we represented a system with a parts-by-wholes incidence matrix I; we fixed system parameters $\pi = 1/100$, and $\alpha = 1/10$ and $\gamma = 9/10$. We simulated 100 gene-lists X from model (2.2), each time using a simulated activity vector Z. For methods comparison, we applied: (1) the commonly used Fisher exact test for enrichment of each gene set in the data X, (2) MGSA, and (3) MFA-ILP. We allowed both model-based methods to know the system parameter settings. To evaluate performance we calculated specificity, sensitivity, and precision of the estimated activity vector $\hat{Z}$ for the true activity vector Z by averaging over the 100 replicates.

Experiment 1: Low overlap. Initially, I had size size 300 genes (parts) by 100 gene sets (wholes). We randomly picked 5 and 10 parts for each whole in columns 1-50 and 51-100, respectively. Then we removed parts not contained by any whole, leaving a 296×100 incidence matrix. We sampled Z from prior (2.3) and then projected it onto AH by constructing $A_p = \max_{w:p\in w} Z_w$ and then updating $Z_w = \min_{p:p\in w} A_p$. All methods exhibit similar operating characteristics in this case (Table 1).

Experiment 2: Higher overlap and parent-child structure. Initially, I had size 300 parts by 105 wholes. From column 1 to column 20, each column has 20 parts, of which 15 parts are in common with each other and 5 parts are randomly selected from the other parts; column 21 has 10 parts which are randomly picked from the 15 common parts shared by columns 1-20. Thus, columns 1-20 have a lot of overlaps and column 21 is a child of columns 1-20. Similarly we built columns 22-42, 43-63, 64-84 and 85-105. The common 15 parts in each column combination are all different. Then parts not contained by any whole were removed, which resulted in a 265×105 incidence matrix. We activated wholes by sampling one whole from columns 1-20, 22-41, 43-62, 64-83 and 85-104 as activated, respectively, and projected onto AH as above.

Table 2 exhibits properties of three methods when the system is relatively complicated. The univariate Fisher test tends to select the wholes with a high correlation (overlap) with the truly activated wholes, which results in high sensitivity but low specificity (or precision). The extra activation calls correspond to spurious associations that the multivariate, model-based approaches are able to recognize. The MGSA method often fails to discover truly activated wholes, which corresponds to a reduced sensitivity. The small *child* wholes tend to be missed by MGSA in this case. The proposed MFA-ILP method is right on target.

4.3 *ILP for large systems*

Large systems strain unaided ILP, but the special structure of the gene-set problem allows for several refinements.

Shrinking I: Observe that the objective function in (4.8) may be expressed:

$$l(Z, A) = c_1 \sum_w Z_w + c_2 \sum_{p \in P^-} A_p + c_3 \sum_{p \in P^+} A_p$$

where

$$c_1 = \log \pi - \log(1 - \pi)$$
$$c_2 = \log(1 - \gamma) - \log(1 - \alpha)$$
$$c_3 = \log \gamma - \log \alpha$$

and where P^- and P^+ denote the observed inactivated and activated parts, respectively. That is, $p \in P^-$ if $x_p = 0$ and $p \in P^+$ if $x_p = 1$. By assumption (2.2), $\alpha < \gamma$ and so $c_2 < 0$ and $c_3 > 0$. If we further insist that $\pi < 1/2$, then $c_1 < 0$ also. In some cases we can know which $\hat{Z}_w$ and $\hat{A}_p$ must equal 0 in the optimal solution, and if so we can remove these variables from the system prior to implementing ILP. For each whole w denote $P_w^+ = w \cap P^+$ and similarly $P_w^- = w \cap P^-$, and define $W^* = \left\{ w : c_1 + c_3 \sum_{p \in P_w^+} 1 < 0 \right\}$. Clearly those wholes containing no reported parts are in W^*, but there may be others. We prove in Section 6 that if W^* is not empty then we may be able to shrink the system prior to solving the constrained optimization problem via ILP.

Theorem 4.1 Let w_0 denote an element of W^*. If there exists $p_0 \in w_0$ such that $\{w : p_0 \in w\} \subset W^*$, then $\hat{Z}_{w_0} = \hat{A}_{p_0} = 0$.

Letting W_0 and P_0 denote wholes and parts for which the optimal solution is known (in advance of computation), we may remove these from the incidence matrix I, effectively shrinking it. The amount of shrinkage may be dramatic, but it depends on the observed data x, the system I, and system parameters α, γ, and π. When α is small and γ is large the effects may be minimal.

A sequential approach: In the unlikely event that the system matrix I is separable into blocks of wholes that do not overlap between blocks, then ILP may be applied separately to these distinct blocks in order to identify the MAP activities. We do not expect this separability in GO or related systems, but we can take advantage of size variation of the wholes and work sequentially from small ones to larger ones. As an example, let S_{10} denote the sets containing no more than $n.up = 10$ genes. In order to obtain the optimal solution for the full problem, we start from the sub-matrix $I.10$ obtained by extracting these sets from I. Suppose Z_{10}^* is the MAP solution based on the data for $I.10$, and use notation S_{10}^* to denote the active sets in S_{10} as inferred by Z_{10}^*. We aim to find the optimal solution Z_{11}^* for $I.11$ using what has already been computed in the smaller system. Denote the newly added sets in $I.11$ by S_{10}^{11} (*i.e.*, the sets containing exactly 11 genes). We just need to consider the sets with the possibility being active in the optimal solution on $I.11$. First of all, S_{10}^* and S_{10}^{11} should be included, in the case we have no any other prior knowledge about Z_{11}^*. Second, by the 3rd AH inequality (Theorem 2.2), any set in $S_{10} \backslash S_{10}^*$ which is a subset of some set in S_{10}^{11}, denoted by D, should also be included. But these sets already considered are not enough. Actually, for each set w_1 in S_{10}^{11}, we need to check whether there exist some set w_2 in $S_{10} \backslash (S_{10}^* \cup D)$ satisfying

$$c_1 + c_2 \sum\nolimits_{p \in P^- \cap P_{w_1}^{w_2}} A_p + c_3 \sum\nolimits_{p \in P^+ \cap P_{w_1}^{w_2}} A_p > 0, \tag{4.9}$$

where $P_{w_1}^{w_2}$ denote the set of genes contained by w_2 and not by w_1. We do this since each set in S_{10}^{11} may be active in the optimal solution Z_{11}^*, and we need to check whether some sets in S_{10} should be activated towards maximizing the objective function. We denote the sets in $S_{10} \backslash (S_{10}^* \cup D)$ satisfying the condition (4.9) by E. Finally, by the 3rd AH inequality (Theorem 2.2), any set in $S_{10} \backslash (S_{10}^* \cup D \cup E)$ which is a subset of some set in E, denoted by F, should also be included. Thus, we need to run the ILP on the incidence matrix only for $S_{10}^* \cup S_{10}^{11} \cup D \cup E \cup F$, instead of $I.11$. Hence, we obtain a sequential approach to solve the full ILP problem from a sequence of smaller problems. Examples show this is feasible in GO for sub-systems holding sets of up to 50 genes, without excessive computational burden.

5. Posterior sampling via penalized MCMC

To obtain a sample from the posterior distribution defined by prior (2.4) and model (2.2) in which the whole activity variables $Z = (Z_w)$ have positive probability only when Z satisfies AH, we designed a Markov chain to run according to a penalized posterior within the entire, unconstrained space. Up to a constant, this penalized log posterior is:

$$\tilde{l}(Z) = l(Z, A) - \lambda \sum_{w} V_w \tag{5.10}$$

where $l(Z, A)$ is defined in (4.8), V_w is the violation indicator (3.6), and $\lambda \geqslant 0$ is a tuning parameter. The desired sample is obtained by discarding any sampled states that do not satisfy AH. Note that there are no violations ($\sum_w V_w = 0$) for Z that satisfy AH, so that $\tilde{l}(Z) = l(Z, A)$ in this case and the conditional log posterior distribution under $\tilde{l}(Z)$ restricted to AH is identical to the target log posterior distribution. Increasing the tuning parameter λ increases the probability of AH in the larger state space, which is essential for efficient sampling when this probability is small. As the Markov chain samples from the larger space of all possible binary vectors Z, here the part activities A are determined by Z, the mapping from Z to A need not be invertible as in Theorem 2.1, and we can consider $\tilde{l}$ as a function only of Z.

It is helpful to visualize the Markov chain as operating by changing colors on the node-colored bipartite graph $\mathcal{G}$ having whole nodes and part nodes, with an edge between a whole node w and part node p if and only if $p \in w$, and where the coloring of the whole nodes $\{w\}$ and part nodes $\{p\}$ match the activities Z and A, respectively. It is useful in assessing the state of the Markov chain to associate with each node a count $n(\cdot)$ of its active connected neighbors in $\mathcal{G}$. $A_p = 1$ if and only if $n(p) > 0$ and $V_w = 1$ if and only if $Z_w = 0$ and $n(w) = \deg(w)$, the number of part nodes $p \in w$.

The Markov chain proceeds by selecting at random a whole node w and proposing a color swap (a change in the status of the activitity variable, $Z^*_w = 1 - Z_w$) for this node. This proposed change can, but need not, affect the activities of parts contained in this whole. When $Z^*_w = 1$, the active neighbor counts $n(p)$ increase by 1 for each $p \in w$. If A_p changes from 0 to 1, then each node w' that contains p (including w) gains an additional active neighbor and $n(w')$ increases by 1. This increase could cause a violation if p were the only remaining inactive neighbor of an inactive w', causing $V_{w'}$ to change form 0 to 1. If node w were in violation before this proposal, activating it would eliminate the violation. Similarly, when $Z^*_w = 0$, the active neighbor counts $n(p)$ decrease by 1 for each $p \in w$. If this decrease is from 1 to 0, then the activity A_p changes from 1 to 0 as well and all of the whole nodes w' connected to p would lose an active neighbor, $n(w')$ decreasing by 1. If whole node w' had been in violation, this change would eliminate the violation with $V_{w'}$ changing from 1 to 0.

Careful accounting of the changes to a few key counts allows for quick calculation of the change in $\tilde{l}(Z^*)$ and subsequent acceptance or rejection of the proposal by Metropolis-Hastings. The log posterior $\tilde{l}(Z^*)$ is a function of α, γ, π, the penalty λ and the counts of the numbers of active and inactive whole nodes ($\sum_w Z_w$ and $\sum_w (1 - Z_w)$, respectively), the number of whole nodes in violation ($\sum V_w$), the numbers of active part nodes with realized values 1 and 0 ($\sum_p A_p x_p$ and $\sum_p A_p(1 - x_p)$, respectively), and the numbers of inactive part nodes with realized values 1 and 0 ($\sum_p (1 - A_p) x_p$ and $\sum_p (1 - A_p)(1 - x_p)$, respectively).

Random selection of w for the proposed change may be uniform. Some increase in MCMC mixing efficiency is obtainable by using a nonuniform proposal distribution, for example, by selecting whole nodes with smaller set size and with larger proportions of parts with realized values equal to 1 with greater probability.

We note that penalizing the log posteriro within the unconstrained space leads to a rapidly mixing sampler, where our previous attempts to construct move types within the constrained space were less successful.

6. Proofs and other calculations

Proof of Theorem 2.2: Relative to all the sets and parts in the system I, we say AH holds if and only if $A_p = \max_{w: p \in w} Z_w$ for all p and $Z_w = \min_{p: p \in w} A_p$ for all w. Recall that all A_p and Z_w are binary, in $\{0, 1\}$. The first condition $A_p = \max_{w: p \in w} Z_w$ implies $A_p \geqslant Z_w$ for all

w with $p \in w$; that A_p achieves the max of the Z_w's has to account for the possibility that $A_p = 1$ when all $Z_w = 0$, but this is covered by having $A_p \leqslant \sum_{w:p\in w} Z_w$. Thus the condition $A_p = \max_{w:p\in w} Z_w$ is equivalent to the first two constraints in Theorem 2.2.

To address the second condition, that $Z_w = \min_{p:p\in w} A_p$ for all w, define a new variable

$$T_w = 1 + \sum_{p:p\in w} (A_p - 1)\,,$$

and notice that $T_w = 1$ if and only if $A_p = 1$ for all $p \in w$, otherwise $T_w \leqslant 0$. Observe that the second condition is equivalent to

$$\sum_{p:p\in w} (Z_w - A_p + 1) - T_w \geqslant 0 \tag{6.11}$$

since if all $A_p = 1$, for $p \in w$, then $T_w = 1$, and Z_w must equal 1 to satisfy (6.11); otherwise, if at least one of the A_p's equals 0, then $T_w \leqslant 0$, and noting that the summation in (6.11) is positive confirms the claim. Next, replacing T_w in (6.11) with its definition, we obtain the third stated inequality

$$\sum_{p:p\in w} (Z_w - 2A_p + 2) \geqslant 1.$$

An algorithm to evaluate the violation probability $P_1(V_w = 1)$: A violation of AH occurs for whole w if (and only if) $Z_w = 0$ and yet $A_p = 1$ for all $p \in w$. Immediately, we see $P_1(V_w = 1) = (1 - \pi)P\left(A_p = 1 \forall p \in w | Z_w = 0\right)$. The second factor is difficult to compute by full enumeration since all configurations of overlapping sets must be considered, and this is exponential in the number of overlapping sets. One simple fact is that if $A_p = 1$, then at least one element in $\{Z_{w'} : w' \neq w, p \in w'\}$ should be equal to 1. Based on this observation, one idea is to find disjoint sets of $Z_{w'}$'s which will result in $\prod_{p\in w} A_p = 1$ (this probability is easy to be computed using their independence), and then consider all the cases. We dervied an algorithm and coded a recursive function to implement it. In the function, we do something like a Depth First Search.

- Activate the whole containing the most parts (parts contained in this whole are thus turned on), and see which parts are left to be activated; if those remaining parts are contained by disjoint sets of wholes, compute the probability directly and go for the next whole containing the second most parts, and so on; if not, invoke the function recursively.

- Overlook wholes already activated and check if there are duplicate wholes which contain the same parts; if true (this happens a lot in GO), retain only one of them and record the number of duplicates;

- If there exists some part left to be activated which is not contained any left whole, stop the execution of the function.

One advantage of this algorithm is that we check if there are duplicate wholes and combine them at each recursion, which makes use of the local structure of the incidence matrix to the fullest extent and can save a lot on running time.

The pseudo code of the algorithm is

```
GetValue(p, wholes, parts)
  ## p is the Bernoulli parameter, wholes and parts are
  ## two lists storing the containing information
  if (there are duplicate wholes which contain the same parts)
      only retain one of them and record the number of duplicates;
  endif
  sort all the wholes according to the decreasing order of the number of parts
  they contain;
  for (i in 1:n )## length(wholes) = n
      remove wholes[0: i-1];
      turn on wholes[i] and check which parts are left to be activated;
      if (some part left to be activated is not included by any whole in wholes[i:n])
            break;
      endif
      if (sets of wholes containing these parts are disjoint)
            compute the probability, stored by prob[i];
      else (prob[i] = Getvalue(p, trimmed wholes, trimmed parts))
      ## trimmed wholes for ones containing parts left to be activated (parts in
    ## wholes[i] will be removed from the containing information) and
    ## trimmed parts for ones left to be activated
    endif
endfor
return sum(prob);
```

Proof of the Theorem 4.1: Compared to $Z_{w_0} = 0$, the possible maximal value added to the objective function by letting Z_{w_0} be 1 is $c_1 + c_3 \sum_{p \in P_{w_0}^+} 1$ (considering parts in $P_{w_0}^-$ may already be activated or w_0 has no parts in P^-, the best case), however, which is negative since $w_0 \in W^*$. So $Z_{w_0} = 0$ is preferred towards maximizing the objective function. Next we need to prove that letting $Z_{w_0} = 0$ and $A_{p_0} = 0$ will not violate the inequalities in Theorem 2.2.

Denote by W_0 and P_0 the sets of w_0 and p_0 satisfying the state in the Theorem, respectively. We claim that for each $p_0 \in P_0$, $\{w : I_{p_0,w} = 1\} \subset W_0$. If not, then there exists $w^* \in W^* \backslash W_0$ such that $I_{p_0,w^*} = 1$, so w^* will be in W_0. This is a contradiction. Thus $A_{p_0} = \max_{\{w:I_{p_0,w}=1\}} Z_w = 0$, so the first two AH inequalities are satisfied. It is readily verified that the third inequality is also satisfied.

Acknowledgments

We thank Christina Kendziorski for comments that helped to guide the development of this research. This research was supported in part by R21 HG 006568 and by a fellowship from the Morgridge Institute of Research.

References

Bauer, S. and Gagneur J. and Robinson, P. N. (2010). GOing Bayesian: model-based gene set analysis of genome-scale data. *Nucleic Acids Research*, 38 (11): 3523-3532.

Bauer, S. and Robinson, P. N. and Gagneur, J. (2011). Model-based gene set analysis for Bioconductor. *Bioinformatics*, 27 (13): 1882-1883.

Arratia, R. and Goldstein L. and Gordon L. (1990). Poisson approximations and the Chen-Stein method. *Statistical Science*, 5, 403-424.

Carvalho, L. E. and Lawrence, C. E. (2008). Centroid estimators for inference in high-dimensional discrete spaces. *PNAS: USA*, 105, 3209-3214.

Newton, M. A. and He, Q. and Kendziorski, C. (2012). A model-based analysis to infer the functional content of a gene list. *Statististical Applications in Genetics and Molecular Biology*, 11(2), Art 9.

R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL `http://www.R-project.org/`.

Robert, C.P. and Cassella, G. (2002). *Monte Carlo Statistical Methods*, New York: Springer.

Sartor, M.A. and Leikauf, G. D. and Medvedovic, M. (2009). LRpath: a logistic regression approach for identifying enriched biological groups in gene expression data. *Bionformatics*, 25, 211-217.

Tables

Table 1. Simulation comparison of three gene-set methods, a case of low overlap among gene sets: Tabulated are mean values from 100 simulated data sets.

Method	# of true active sets	# of predicted active sets	sensi	speci	preci
MFA-ILP	7.3	7.4	0.963	0.997	0.958
Fisher (cut-off=0.05)	7.3	5.9	0.790	0.998	0.966
Fisher (cut-off=0.1)	7.3	6.8	0.873	0.996	0.948
MGSA (cut-off=0.5)	7.3	7.2	0.954	0.998	0.968

Table 2. Simulation comparison of three gene-set methods, a case of higher overlap among gene sets: Tabulated are mean values from 100 simulated data sets.

Method	# of true active sets	# of predicted active sets	sensi	speci	preci
MFA-ILP	10.1	10.15	0.975	0.997	0.993
Fisher (cut-off=0.05)	10.1	104.2	0.996	0.008	0.096
Fisher (cut-off=0.1)	10.1	104.8	0.996	0.002	0.096
MGSA (cut-off=0.5)	10.1	5.45	0.490	0.995	0.920

Figures

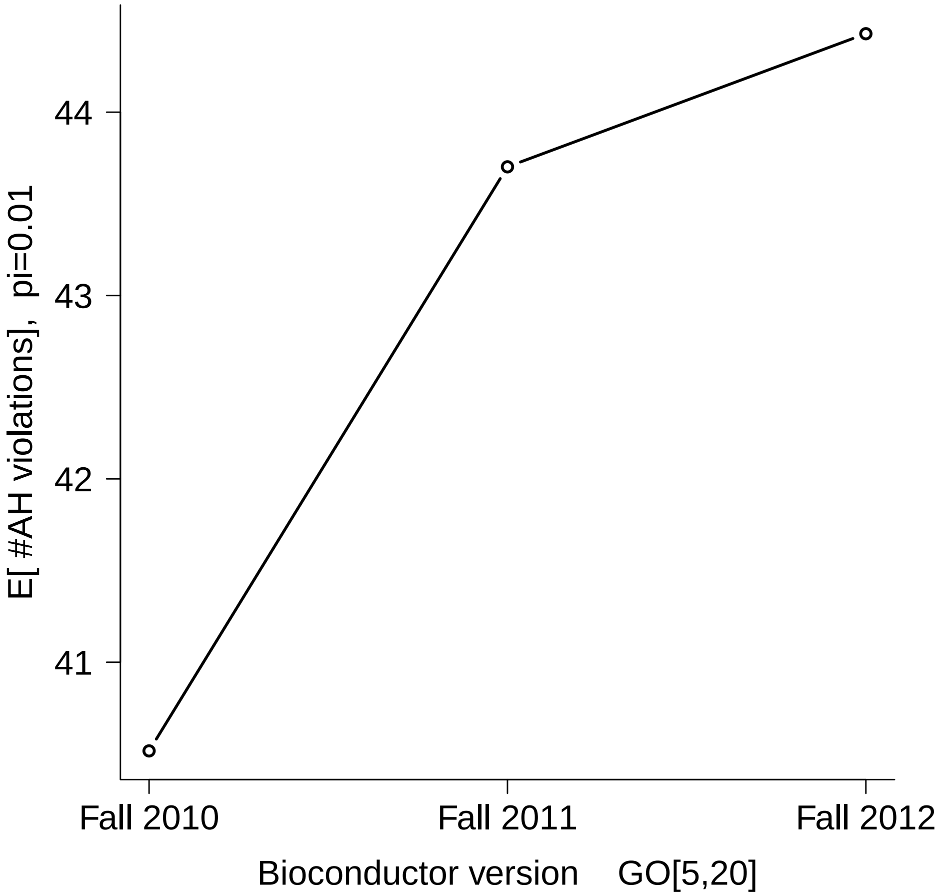

Fig. 1. Expected number of sets that violate AH for three recent versions of Gene Ontology (GO), considering sets holding between 5 and 20 genes. Calculations are done at $\pi = 1/100$. Respectively, these systems contain 3591, 4096, and 4449 gene sets.