Original citation:

Permanent WRAP url:
http://wrap.warwick.ac.uk/50805

Copyright and reuse:
The Warwick Research Archive Portal (WRAP) makes the work of researchers of the University of Warwick available open access under the following conditions.

This article is made available under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported (CC BY-NC-ND 3.0) license and may be reused according to the conditions of the license. For more details see: http://creativecommons.org/licenses/by-nc-nd/3.0/

A note on versions:
The version presented in WRAP is the published version, or, version of record, and may be cited as it appears here.

For more information, please contact the WRAP Team at: wrap@warwick.ac.uk

http://go.warwick.ac.uk/lib-publications
Bayesian correlated clustering to integrate multiple datasets

Paul Kirk 1, Jim E. Griffin 2, Richard S. Savage 1, Zoubin Ghahramani 3 and David L. Wild 1 ∗

1 Systems Biology Centre, University of Warwick, Coventry, CV4 7AL, UK
2 School of Mathematics, Statistics and Actuarial Science, University of Kent, CT2 7NF, UK
3 Department of Engineering, University of Cambridge, Cambridge, CB2 1PZ, UK

Associate Editor: Dr. Olga Troyanskaya

ABSTRACT

Motivation: The integration of multiple datasets remains a key challenge in systems biology and genomic medicine. Modern high-throughput technologies generate a broad array of different data types, providing distinct – but often complementary – information. We present a Bayesian method for the unsupervised integrative modelling of multiple datasets, which we refer to as MDI (Multiple Dataset Integration). MDI can integrate information from a wide range of different datasets and data types simultaneously (including the ability to model time series data explicitly using Gaussian processes). Each dataset is modelled using a Dirichlet-multinomial allocation (DMA) mixture model, with dependencies between these models captured via parameters that describe the agreement among the datasets.

Results: Using a set of 6 artificially constructed time series datasets, we show that MDI is able to integrate a significant number of datasets simultaneously, and that it successfully captures the underlying structural similarity between the datasets. We also analyse a variety of real S. cerevisiae datasets. In the 2-dataset case, we show that MDI’s performance is comparable to the present state of the art. We then move beyond the capabilities of current approaches and integrate gene expression, ChIP-chip and protein-protein interaction data, to identify a set of protein complexes for which genes are co-regulated during the cell cycle. Comparisons to other unsupervised data integration techniques – as well as to non-integrative approaches – demonstrate that MDI is very competitive, while also providing information that would be difficult or impossible to extract using other methods.

Availability: A Matlab implementation of MDI is available 1 from https://sites.google.com/site/mdipackage/.
Contact: D.L.Wild@warwick.ac.uk

1 INTRODUCTION

The wide range of modern high-throughput genomics technologies has led to a rapid increase in both the quantity and variety of functional genomics data that can be collected. For example, large-scale microarray (Schena et al., 1995; Lockhart et al., 1996), chromatin immunoprecipitation (ChIP) chip (Solomon et al., 1988), and tandem affinity purification (TAP) (Rigaut et al., 1999; Puig et al., 2001) datasets are available for a broad selection of organisms, providing measurements of mRNA expression, protein-DNA binding, and protein-protein interactions. In the forthcoming era of personal genomic medicine, we may reasonably expect genome sequences and other forms of high-throughput data (such as gene expression, alternative splicing, DNA methylation, histone acetylation, and protein abundances) to be routinely measured for large numbers of people. The development of novel statistical and computational methodology for integrating diverse data sources is therefore essential, and it is with this that the present work is concerned.

As is common in statistics and machine learning, data integration techniques can be broadly categorised as either supervised (where a training/gold-standard set with known labels is used in order to learn statistical relationships) or unsupervised (where there is no training dataset, but we nevertheless seek to identify hidden structure in the observed data; e.g. by clustering). Our proposed method is unsupervised, but there are also a number of supervised learning algorithms that are designed to integrate multiple data sources; we now briefly mention these for the sake of completeness. These have proven highly successful in several contexts, often when predicting whether a link or interaction exists between two genes or proteins. Depending on the application, the link might represent (to provide just a few examples) protein-protein binding (Jansen et al., 2003; Rhodes et al., 2005), or a synthetic sick or lethal interaction (Wong et al., 2004), or might indicate that the two genes have been implicated in the same biological process (Myers and Troyanskaya, 2007). Approaches for predicting these links often proceed by collecting a gold-standard set of positive and negative interactions (see, for contrasting examples, Jansen et al., 2003; Lee et al., 2004; Myers et al., 2005), and then training statistical models (e.g. decision trees, naive Bayes classifiers) that predict the presence/absence of these interactions. These models may then be applied in order to predict the presence/absence of previously unknown interactions. Since training and prediction are performed on the basis of information collected from multiple different data sources, these approaches provide a form of data integration. Such supervised data integration techniques have proven highly effective for predicting interactions, some of which may then be verified experimentally (e.g. Rhodes et al., 2005; Huttenhower et al., 2009). Moreover, the work of Huttenhower et al. (2009) demonstrates that such approaches may be used to integrate whole-genome scale datasets. The Bayesian network approach of Troyanskaya et al. (2003) was a precursor to many of these supervised approaches, but differs from the others in that it uses knowledge from human experts in order to integrate predictions derived from diverse datasets.

Here we propose a novel unsupervised approach for the integrative modelling of multiple datasets, which may be of different types. For brevity, we refer to our approach as MDI, simply as a shorthand for “Multiple Dataset Integration”. We model each dataset using a Dirichlet-multinomial allocation (DMA) mixture model (see Section 2.1), and exploit statistical dependencies between the datasets in order to share information (Section 2.2). MDI permits the identification of groups of genes that tend to cluster together in one, some, or all of the datasets. In this way, our method is able to use the information contained within diverse datasets.
datasets in order to identify groups of genes with increasingly specific characteristics (e.g. not only identifying groups of genes that are co-regulated, but additionally identifying groups of genes that are both co-regulated and whose protein products appear in the same complex).

Informally, our approach may be considered as a “correlated clustering” model, in which the allocation of genes to clusters in one dataset has an influence upon the allocation of genes to clusters in another. This contrasts with “simple” clustering approaches (such as k-means, hierarchical clustering, etc.) in which the datasets are clustered independently (or else concatenated and treated as a single dataset). It also clearly distinguishes our methodology from biclustering (e.g. Cheng and Church, 2000; Reiss et al., 2006). Biclustering is the clustering of both dimensions in a single dataset (for example, both genes and experiments in a gene expression dataset). MDI, in contrast, clusters a single dimension (for example genes) across multiple datasets. Biclustering is not applicable here as the datasets can be arbitrarily different, making any clustering across all features difficult. MDI avoids the problem of comparing different data types by instead learning the degree of similarity between the clustering structures (i.e. the gene-to-cluster allocations) in different datasets (see Section 2.2).

MDI makes use of mixture models, which have become widespread in the context of unsupervised integrative data modelling (e.g. Barash and Friedman, 2002; Liu et al., 2006, 2007), gaining increased popularity in recent years (Savage et al., 2010; Rogers et al., 2010). The principal advantages of using mixture models are: (i) they provide flexible probabilistic models of the data; (ii) they naturally capture the clustering structure that is commonly present in functional genomics datasets; and (iii) by adopting different parametric forms for the mixture components, they permit different data types to be modelled (see also Section 2.1). An early application to data integration is provided by Barash and Friedman (2002), who performed integrative modelling of gene expression and binding site data.

As part of our approach, we infer parameters that describe the levels of agreement between the datasets. Our method may thus be viewed as extending the work of Balasubramanian et al. (2004). In this regard, MDI is also related to the approach of Wei and Pan (2012), which models the correlation between data sources as part of a method that classifies genes as targets or non-targets of a given transcription factor (TF) using ChIP-chip, gene expression and DNA binding data, as well as information regarding the position of genes on a gene network. Perhaps most closely related to MDI (in terms of application) are the method of Savage et al. (2010) and iCluster (Shen et al., 2009). Savage et al. (2010) adopt a mixture modelling approach, employing a hierarchical Dirichlet process in order to perform integrative modelling of 2 datasets. As well as significant methodological differences, the principal practical distinction between this approach and MDI is that we are able to integrate more than 2 datasets, any or all of which may be of different types (see Section 2). Like MDI, the iCluster method of Shen et al. (2009) permits integrative clustering of multiple (>2) genomic datasets, but uses a joint latent variable model (see Shen et al., 2009, for details). In contrast to MDI, iCluster seeks to find a single common clustering structure for all datasets. Moreover, iCluster must resort to heuristic approaches in order to estimate the number of clusters, whereas MDI infers this automatically (see Section 2.1). We demonstrate that MDI provides results that are competitive with the 2-dataset approach of Savage et al. (2010) in Section 3.2, and provide a comparison of results obtained using MDI, iCluster and simple clustering approaches in the Supplementary Material.

The potential biological applications of our approach are quite diverse, since there are many experimental platforms that produce measurements of different types, which might be expected to possess similar (but not necessarily identical) clustering structures. For example, in the 2-dataset case, related methodologies have been used to discover transcriptional modules (Liu et al., 2007; Savage et al., 2010) and prognostic cancer subtypes (Yuan et al., 2011) through the integration of gene expression data with TF binding (ChIP-chip) data and copy number variation data, respectively. A related approach was also employed by Rogers et al. (2008) in order to investigate the correspondence between transcription and proteomic expression profiles. In the example presented in this paper, we focus on the biological question of identifying protein complexes whose genes undergo transcriptional co-regulation during the cell cycle.

The outline of this paper is as follows. In Section 2, we briefly provide some modelling background and present our approach. Inference in our model is performed via a Gibbs sampler, which is provided in the Supplementary Material. In Section 3, we describe three case study examples, in all of which we use publicly available S. cerevisiae (baker’s yeast) datasets. We present results in Section 4, and a discussion in Section 5.

2 METHODS

In this section, we provide some background regarding Dirichlet-multinomial allocation mixture models (Section 2.1), and consider how these may be extended to allow us to perform integrative modelling of multiple datasets (Section 2.2). Inference in the resulting model (which we henceforth refer to as MDI) is performed using a Gibbs sampler (see Supplementary Material). We briefly describe in Section 2.4 how the resulting posterior samples may be effectively summarised.

2.1 Dirichlet-multinomial allocation mixture models

We model each dataset using a finite approximation to a Dirichlet process mixture model (Ishwaran and Zarepour, 2002), known as a Dirichlet-multinomial allocation mixture model (Green and Richardson, 2001). Such models have the following general form:

\[ p(x) = \frac{N}{\sum_{c=1}^{N} \pi_c} \prod_{i=1}^{N} \pi_c \frac{1}{f(x | \theta_c)}. \]  

In the above, \( p(x) \) denotes the probability density model for the data, which is here an \( N \) component mixture model. The \( \pi_c \)'s are mixture proportions, \( f \) is a parametric density (such as a Gaussian), and \( \theta_c \) denotes the vector of parameters associated with the \( c \)-th component. Importantly, different choices for the density \( f \) allow us to model different types of data (for example, a normal distribution might be appropriate for continuous data, while a multinomial might be appropriate for categorical data).

Given observed data \( x_1, \ldots, x_N \), we wish to perform Bayesian inference for the unknown parameters in this model. As is common in mixture modelling (e.g. Dempster et al. 1977; see also Friedman et al. 2004 for a graphical model perspective), we introduce latent component allocation variables \( c_j \in \{1, \ldots, N\} \), such that \( c_i \) is the component responsible for observation \( x_i \). We then specify the model as follows:

\[
\begin{align*}
\mathbf{x} | \mathbf{c}, \mathbf{\theta} & \sim F(\mathbf{c}, \mathbf{\theta}) \\
\mathbf{c} | \pi \sim \text{Multinomial}(\pi_1, \ldots, \pi_N), \\
\pi_1, \ldots, \pi_N & \sim \text{Dirichlet}(\alpha/N, \ldots, \alpha/N), \\
\theta_c & \sim g(0).
\end{align*}
\]
where \( F \) is the distribution corresponding to density \( f, \pi = (\pi_1, \ldots, \pi_N) \) is the collection of \( N \) mixture proportions, \( \alpha \) is a mass/concentration parameter (which may also be inferred), and \( G(0) \) is the prior for the component parameters. Bayesian inference for such models may be performed via Gibbs sampling (Neal, 2000). Note that a realisation of the collection of component allocation variables, \((c_1, \ldots, c_N)\), defines a clustering of the data (i.e. if \( c_i = c_j \), then \( x_i \) and \( x_j \) are clustered together). Since each \( c_j \) is a member of the set \( \{1, \ldots, N\} \), it follows that the value of \( N \) places an upper bound on the number of clusters in the data.

The Dirichlet process (DP) mixture model may be derived by considering the limit \( N \to \infty \) in Equation (1) (Neal, 1992; Rasmussen, 2000). In the present paper, it is convenient to persist with finite \( N \) (see Section 2.2). The important point is that \( N \) just places an upper bound on the number of clusters present in the data (since, as in the infinite DP case, not all of the components need to be “occupied”; i.e. not all components need to have observations associated with them), and hence \( N \) does not specify the precise number of clusters \( a \ priori \). Provided \( N \) is taken sufficiently large, the number of clusters present in the data will be (much) less than \( N \), and we will retain the ability to identify automatically the number of clusters supported by the data. Theoretical justifications for “large” mixture models such as this (in which the number of components in the mixture is larger than the true number of clusters in the data) are provided by Rousseau and Mengersen (2011). A choice of \( N = n \) would set the upper bound on the number of clusters to be equal to the number of genes. As a tradeoff with computational cost, we take \( N = \lfloor n/2 \rfloor \) throughout this paper.

2.2 Dependent component allocations

We are interested in the situation where we have a collection of \( n \) genes, for each of which we have measurements from \( K \) different data sources. One possible modelling approach would be to fit \( K \) independent DMA mixture models, represented graphically in Figure 1a for the case \( K = 3 \). However, this neglects to consider (and fails to exploit) structure within the data that may be common across some or all of the different sources. For example, a set of co-regulated genes might be expected to have similar expression profiles, as well as having a common collection of proteins that bind their promoters. We therefore propose a model in which we allow dependencies between datasets at the level of the component allocation variables, \( c_i \).

We consider \( K \) mixture models (one for each dataset), each defined as in Equations (1) and (2). We add right subscripts to our previous notation in order to distinguish between the parameters of the \( K \) different models (so that \( \alpha_k \) is the mass parameter associated with model \( k \), etc.) and take \( N_k = N \) in all mixture models. Note that each model is permitted to have a different mass parameter, \( \alpha_k \). MDI links these models together at the level of the component allocation variables via the following conditional prior:

\[
p(c_1, c_2, \ldots, c_K | \phi) \propto \prod_{k=1}^{K} \pi_{c_1k} \prod_{k=1}^{K-1} \prod_{l=k+1}^{K} (1 + \phi_{kl} I(c_{lk} = c_{lk})),
\]

where \( I \) is the indicator function, \( \phi_{kl} \in \mathbb{R}_{>0} \) is a parameter that controls the strength of association between datasets \( k \) and \( l \), and \( \phi \) is the collection of all \( K(K-1)/2 \) of the \( \phi_{kl} \)'s. For clarity, note that \( c_{ik} \in \{1, \ldots, N\} \) is the component allocation variable associated with gene \( i \) in model \( k \), and that \( \pi_{c_1k} \) is the mixture proportion associated with component \( c_{ik} \) in model \( k \). Informally, the larger \( \phi_{kl} \), the more likely it is that \( c_{ik} \) and \( c_{lk} \) will be the same, and hence the greater the degree of similarity between the clustering structure of dataset \( k \) and dataset \( l \). In Figure 1b, we provide a graphical representation of our model in the case \( K = 3 \). If all \( \phi_{kl} = 0 \), then we recover the case of \( K \) independent DMA mixture models (Figure 1a). Note that \( (1 + \phi_{kl} I(c_{lk} = c_{lk})) \geq 1 \), hence if \( \phi_{kl} > 0 \) then we are up-weighting the prior probability that \( c_{lk} = c_{lk} \) (relative to the independent case).

Linking the mixture models at the level of the component allocation variables provides us with a means to capture dependencies between the datasets in a manner that avoids difficulties associated with the datasets being of different types and/or having different noise properties.

2.3 Modelling different data types

To specify our model fully, we must provide parametric densities, \( f \), appropriate for each data source. It is important to note that we may tailor our choice of \( f \) to reflect the data sources that we seek to model. In the present work, we use Gaussian process (GP) models (Rasmussen and Williams, 2006; Kirk and Stumpf, 2009; Cooke et al., 2011) for gene expression time course data, and use multinomial models for categorical data (e.g. discretised gene expression levels). For comparison with the results of Savage et al. (2010), we also consider in our second example (Sections 3.2 and 4.2) a bag-of-words model for ChIP-chip data. Full details of all of these models are given in the Supplementary Material, where we also provide a Gibbs sampler for performing inference. As in Nieto-Barajas et al. (2004), posterior simulation for our model is aided by the strategic introduction of an additional latent variable (see Supplementary Material for details).

2.4 Extracting fused clusters from posterior samples

We wish to identify groups of genes that tend to be grouped together in multiple datasets. Suppose we have a collection of \( K \) datasets, which we label as Dataset 1, . . . , Dataset \( K \). We are interested in identifying groups of genes that tend to cluster together amongst some subcollection of the datasets. Let \( \{k_1, k_2, \ldots, k_m\} \) be a subset of \( \{1, \ldots, K\} \). Our aim is to identify groups of genes that cluster together in all of Dataset \( k_1 \), . . . , Dataset \( k_m \). Adapting terminology from Savage et al. (2010), we define the probability of the \( i \)-th gene being fused across datasets \( k_1, \ldots, k_m \) to be the posterior probability that \( c_{ik_1} = c_{ik_2} = \ldots = c_{ik_m} \). For brevity, we denote this posterior probability by \( p(c_{ik_1} = c_{ik_2} = \ldots = c_{ik_m}) \).

We calculate this quantity as the proportion of posterior samples for which \( c_{ik_1} = c_{ik_2} = \ldots = c_{ik_m} \) are all equal. We may clearly calculate these posterior fusion probabilities for any combination of the datasets (pairs, triplets, etc.), simply by considering the appropriate subset of \( \{1, \ldots, K\} \). We say that
the \( i \)-th gene is fused across datasets \( k_1, k_2, \ldots, k_m \) if \( p(c_{ik_1} = c_{ik_2} = \ldots = c_{ik_m}) > 0.5 \), and we denote the set of all such fused genes by \( \mathcal{F}_{k_1, k_2, \ldots, k_m} \).

If gene \( i \) is a member of \( \mathcal{F}_{k_1, k_2, \ldots, k_m} \), this simply tells us that the component allocation variables \( c_{ik_1}, c_{ik_2}, \ldots, c_{ik_m} \) tend to be equal (i.e. gene \( i \) tends to be allocated to the same component across datasets \( k_1, k_2, \ldots, k_m \)). We also wish to identify the clustering structure that exists amongst these fused genes. From our Gibbs sampler, we have a collection of sampled component allocations for each member of \( \mathcal{F}_{k_1, k_2, \ldots, k_m} \). We identify a final clustering for the set of fused genes by searching amongst the sampled component allocations in order to find the one that maximises the posterior expected adjusted Rand index (see Fritsch and Ickstadt, 2009). The resulting fused clusters contain groups of genes that tend to cluster together across datasets \( k_1, k_2, \ldots, k_m \).

3 EXAMPLES

To demonstrate the usage and utility of MDI, we consider 3 examples using publicly available \emph{S. cerevisiae} datasets. We specify the priors adopted for unknown parameters and provide MCMC running specifications in the Supplementary Material. Each of our examples serves a different purpose. In the first (Section 3.1), we consider an easily interpretable synthetic dataset, which allows us to illustrate the types of results that can be obtained using MDI. In the second (Section 3.2), we seek to compare our method to the present state of the art in data integration (namely, the approach of Yeung et al., 2003). Although this approach is limited to integrating 2 datasets only, it provides a useful benchmark for MDI. Finally, in Section 3.3, we provide an example that allows us to explore the benefits offered by MDI that go beyond the existing state of the art. We consider the integration of 3 datasets, two of which comprise static measurements (ChIP-chip and PPI), and the other of which comprises gene expression time course data.

3.1 6-dataset synthetic example

To illustrate the properties of our model, we start with a 6-dataset synthetic example. Dataset 1 is constructed by taking a 100-gene subset of the gene expression time course data of Cho et al. (1998), and may be partitioned into 7 easily distinguishable clusters (Figure 2a). We therefore associate with each time course a cluster label, \( Z \in \{1, \ldots, 7\} \). For \( i = 1, \ldots, 5 \), we form Dataset \( i + 1 \) by randomly selecting 25 time courses from Dataset \( i \) and randomly permuting their associated gene names (but not their cluster labels). Thus, for a maximum of 25 genes, the cluster label associated with gene \( g \) in Dataset \( i \) may be different from the cluster label associated with the same gene in Dataset \( i + 1 \). Figures 2b and 2c further illustrate this dataset. A formal approach for comparing the allocation of genes to clusters is to calculate the adjusted Rand index (ARI) between each pair of clustering partitions (Rand, 1971; Hubert and Arabie, 1985). Figure 2d provides a heatmap depiction of the similarity matrix formed by calculating pairwise ARIs.

3.2 Integrating expression and ChIP data

To compare our method to an existing approach for unsupervised data integration, we apply MDI to an example previously considered by Savage et al. (2010) in the context of transcriptional module discovery. We take expression data from a 205-gene subset of the galactose utilisation data of Ideker et al. (2001), which we integrate with ChIP-chip data from Harbison et al. (2004). The expression data were discretised, as in Savage et al. (2010). The 205 genes appearing in this dataset were selected in Yeung et al. (2003) to reflect four functional Gene Ontology (GO) categories. Although this functional classification must be used with some degree of caution (see Yeung et al., 2003), it provides a reasonable means by which to validate the groupings defined by our method. We use the same version of the Harbison et al. dataset as considered by Savage et al. (2010) (significance threshold \( p = 0.001 \)), which provides binding information for 117 transcriptional regulators. For brevity, we henceforth refer to the data of Harbison et al. as “ChIP data”, although we emphasise that this dataset comprises measurements corresponding to a compendium of 117 TFs, rather than to a single

Fig. 2: (a) The data for the 6-dataset synthetic example, separated into 7 clusters. (b) A representation of how the cluster labels associated with each gene vary from dataset to dataset. Genes are ordered so that the clustering of Dataset 1 is the one that appears coherent. (c) A table showing the number of genes having the same cluster labels in datasets \( i \) and \( j \). (d) A heatmap depiction of the similarity matrix formed by calculating the adjusted Rand index between pairs of datasets.
particular TF. Discretising the data (both expression and ChIP-chip) might seem like an unnecessary simplification (since our model can accommodate continuous static measurements through an appropriate choice of component density function, f), but it helps to ensure that our comparison to the results of Savage et al. (2010) is fair. Moreover, discretisation of the ChIP data simplifies modelling and interpretation of the data (the ij-entry of our ChIP data matrix is 1 if we have high confidence that TFj is able to bind the promoter region of gene i, and 0 otherwise), although we acknowledge that this is likely to incur some small information loss.

3.3 Integrating expression, ChIP and PPI data

For an example with 3 diverse data types, we integrate the ChIP data of Harbison et al. with binary protein-protein interaction (PPI) data obtained from BioGRID (Stark et al., 2006) and a gene expression time course dataset of Granovskaia et al. (2010), with the initial intention of identifying protein complexes whose genes undergo transcriptional co-regulation during the cell cycle. We consider the Granovskaia et al. cell cycle dataset that comprises measurements taken at 41 time points, and which was obtained from cells synchronised using alpha factor arrest. We considered only genes identified in Granovskaia et al. (2010) as having periodic expression profiles. After removing those for which there was no ChIP or PPI data, we were left with 551 genes. Our binary PPI data matrix then has rows indexed by these 551 genes, and columns indexed by all of the proteins for which physical interactions identified via yeast 2-hybrid or affinity capture assays have been reported in BioGRID. The ij-entry of the PPI data matrix is 1 if there is a reported interaction between protein j and the protein product of gene i (and 0 otherwise). In an effort to reduce the number of uninformative features, we removed columns containing fewer than five 1’s, leaving 603 columns.

4 RESULTS

4.1 6-dataset synthetic example

Figure 3a shows estimated posterior densities for the mass parameters, αk (obtained from the samples generated by our Gibbs sampler using kernel density estimation). Since each of our datasets is identical (up to permutation of gene names), these distributions should be close to identical, as is the case. For each pair of datasets, we used the posterior αk samples to estimate posterior means, φkℓ. We used these to form a similarity matrix whose kℓ-entry is φkℓ (with φkℓ defined to be φkℓ whenever k > ℓ, and with φkk left undefined). This is shown as a heatmap in Figure 3b. Although they do so in different ways, both the ARI and the dataset association parameters quantify the degree of similarity between the allocation of genes to clusters in pairs of datasets. The similarity of Figures 2d and 3b is therefore reassuring.

To test our ability to identify fused genes, we calculated pairwise fusion probabilities, p(ci,k = cℓ,l), for each gene i and each pair of datasets (k, ℓ). If the true cluster label of gene i is the same in datasets k and ℓ, then p(ci,k = cℓ,l) should be high (greater than 0.5) so that the gene may be correctly identified as fused. Across all pairs of datasets, the minimum pairwise fusion probability for such genes was 0.90 and the mean was 0.97. Conversely, for genes having different cluster labels in datasets k and ℓ, the maximum pairwise fusion probability was 0.05 and the mean was 0.01. Since our fusion threshold is 0.5, we are in this case able to identify the fusion status correctly for all genes.

4.2 Expression + ChIP example

We ran MDI using a multinomial likelihood model for both the discretised expression data and the binary ChIP-chip data. We estimated pairwise fusion probabilities and extracted fused clusters, as described in Section 2.4. We identified 52 fused genes, grouped into 3 clusters. We compared these clusters to the functional classes defined in Yeung et al. (2003). Within each cluster, all genes had the same functional classification, while genes in different clusters possessed different classifications.

In Savage et al. (2010), a bag-of-words model was used to model TF binding data. To permit a fair comparison of the two approaches, we therefore re-ran MDI using a bag-of-words likelihood model for the ChIP data. Following Savage et al. (2010), we then calculated the Biological Homogeneity Index (BHI; Datta and Datta, 2006) for the resulting fused clusters. To calculate the BHI scores, we used the R package clValid (Brock et al., 2008) together with the GO annotations in the org.Sc.sgd.db Bioconductor package (Carlson et al., 2010). The clValid package provides 4 different BHI scores, depending on which GO functional categories are used to define the set of annotations. All categories may be considered, or just one of: biological process (bp), cellular component (cc), and molecular function (mf). We report all 4 BHI scores in Table 1, for the fused clusters defined by: (i) the method of Savage et al. (2010); (ii) MDI using a bag-of-words likelihood; and (iii) MDI using a multinomial likelihood. The BHI scores for MDI (bag-of-words) and the method of Savage et al. (2010) are almost identical, although MDI (bag-of-words) identifies a greater number of fused genes.

Table 1. BHI scores for the fused clusters obtained using the method of Savage et al. (2010), together with those obtained using MDI.

<table>
<thead>
<tr>
<th>BHI values</th>
<th>Number of genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHI (all)</td>
<td>BHI (bp)</td>
</tr>
<tr>
<td>0.98</td>
<td>0.85</td>
</tr>
<tr>
<td>0.98</td>
<td>0.85</td>
</tr>
<tr>
<td>1.00</td>
<td>0.89</td>
</tr>
</tbody>
</table>

4.3 Expression + ChIP + PPI example

We applied MDI to the example of Section 3.3 (using GP models for the gene expression time courses, and multinomial models for the ChIP and PPI datasets), in order to identify groups of genes that are co-regulated during the yeast cell cycle, and whose protein products appear in the same complex. We identified genes fused across all 3 datasets, as well as genes fused across pairs of datasets. We then determined the fused clusters for each of these combinations (see Section 2.4). Additionally, we identified clusters...
for the “single dataset fusion” case (which amounts to identifying a single clustering partition for each of our 3 datasets considered separately). We assess the quality of our clusterings using GO Term Overlap (GOTO) scores (Mistry and Pavlidis, 2008). These assign a score to a pair of genes according to how many GO terms they have in common. This contrasts with BHI, which just assigns a score of 0 or 1 to gene pairs depending on whether or not they share a common GO term. The GOTO scores therefore provide a more finely grained assessment, which implicitly takes into account the hierarchical structure of the Gene Ontology. This is invaluable here, since (as a result of selecting only genes found to have periodic expression profiles during the cell cycle) any 2 randomly selected genes are likely to share some high-level GO terms (see the Supplementary Material for more details).

Table 2. GOTO scores for fused clusters obtained for all combinations of the expression, ChIP and PPI datasets.

<table>
<thead>
<tr>
<th>GOTO (bp)</th>
<th>GOTO (mf)</th>
<th>GOTO (cc)</th>
<th>Number of genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChIP</td>
<td>6.36</td>
<td>0.97</td>
<td>8.53</td>
</tr>
<tr>
<td>PPI</td>
<td>11.04</td>
<td>1.51</td>
<td>11.11</td>
</tr>
<tr>
<td>Expression</td>
<td>7.66</td>
<td>1.15</td>
<td>9.48</td>
</tr>
<tr>
<td>ChIP+PPI</td>
<td>27.04</td>
<td>3.47</td>
<td>18.99</td>
</tr>
<tr>
<td>ChIP+Expression</td>
<td>24.46</td>
<td>2.93</td>
<td>16.87</td>
</tr>
<tr>
<td>PPI+Expression</td>
<td>26.04</td>
<td>3.69</td>
<td>22.35</td>
</tr>
<tr>
<td>ChIP+PPI+Expression</td>
<td>34.81</td>
<td>2.46</td>
<td>26.70</td>
</tr>
</tbody>
</table>

The GOTO scores generally increase as we require agreement across more datasets, while the number of fused genes decreases. Note that this decrease is simply a consequence of requiring agreement among a larger collection of datasets. For example, since the set $S_1 = \{\text{genes that are co-regulated and have protein products that appear in the same complex}\}$ is a subset of $S_2 = \{\text{genes that are co-regulated}\}$, it is inevitable that the number of genes of the former type will be less than or equal to the number of genes of the latter type. In other words, requiring agreement across multiple datasets enables us to identify clusters of genes that have increasingly specific shared characteristics. This is reflected in the increasing GOTO scores, which indicate that genes in the same cluster tend to share a greater number of lower-level (more specific) GO terms.

In Figure 4, we compare the clusters formed by the genes fused across all 3 datasets with those formed by the genes fused across just the PPI and ChIP datasets. Figures 4a and 4b illustrate fusion probabilities for the 31 genes identified as fused across the PPI and ChIP datasets. Each bar in Figure 4a corresponds to a particular gene (as labelled), and represents the posterior probability of that gene being fused across the ChIP and PPI datasets. The corresponding bar in Figure 4b represents the probability of the gene being fused across all 3 datasets. Figure 4c shows the expression profiles for genes identified as fused across the PPI and ChIP datasets, with genes fused across all three datasets shown in colour. Supplementary Figure 2 further illustrates the fused clusters, while Table 3 shows the fused cluster labels and provides descriptions for the genes fused across all 3 datasets.

Table 3. Clusters formed by the genes fused across all 3 datasets. Descriptions were derived from the Saccharomyces Genome Database (Cherry et al., 1998). The IDs in this table correspond to the cluster IDs in Figure 4, with singletons omitted.

<table>
<thead>
<tr>
<th>ID</th>
<th>Gene</th>
<th>Brief description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>NOB1</td>
<td>Involved in synthesis of 40S ribosomal subunits</td>
</tr>
<tr>
<td>2</td>
<td>ENP2</td>
<td>Required for biogenesis of the small ribosomal subunit</td>
</tr>
<tr>
<td>2</td>
<td>RPF2</td>
<td>Involved in assembly of 60S ribosomal subunit</td>
</tr>
<tr>
<td>2</td>
<td>IMP3</td>
<td>Component of the SSU processome</td>
</tr>
<tr>
<td>2</td>
<td>DBP9</td>
<td>Involved in biogenesis of 60S ribosomal subunit</td>
</tr>
<tr>
<td>3</td>
<td>HHF2</td>
<td>Histone H4, core histone protein</td>
</tr>
<tr>
<td>3</td>
<td>HTB2</td>
<td>Histone H2B, core histone protein</td>
</tr>
<tr>
<td>3</td>
<td>HTA1</td>
<td>Histone H2A, core histone protein</td>
</tr>
<tr>
<td>3</td>
<td>HHT1</td>
<td>Histone H3, core histone protein</td>
</tr>
<tr>
<td>3</td>
<td>HTB1</td>
<td>Histone H2B, core histone protein</td>
</tr>
<tr>
<td>3</td>
<td>HHT2</td>
<td>Histone H3, core histone protein</td>
</tr>
<tr>
<td>3</td>
<td>HHF1</td>
<td>Histone H4, core histone protein</td>
</tr>
<tr>
<td>5</td>
<td>SMC3</td>
<td>Subunit of the cohesin complex</td>
</tr>
<tr>
<td>5</td>
<td>IRR1</td>
<td>Subunit of the cohesin complex</td>
</tr>
</tbody>
</table>

We can see from Figures 4a and 4b that the integration of the expression data in addition to the ChIP and PPI data results in Cluster 1 (green) and Cluster 6 (black) being effectively removed. Although many of the genes in Cluster 1 are annotated as cell wall proteins (see Supplementary Material), and although the two genes in Cluster 6 are both cyclins, the genes within these clusters have different expression patterns to one another (Figure 4c, panels 1 and 6). Genes are also lost from Clusters 4 and 5 (shown pink and purple). However, further analysis suggests that this is due to data

Fig. 4: (a) Pairwise fusion probabilities for the 31 genes identified as fused across the ChIP and PPI datasets in the “Expression + ChIP + PPI” example. Colours correspond to fused clusters and the dashed line indicates the fusion threshold. (b) Three-way fusion probabilities for the same 31 genes. Genes that do not exceed the fusion threshold have white bars. (c) The expression profiles for genes identified as fused according to the ChIP and PPI datasets. The coloured lines indicate genes that are also fused across the expression dataset as well.
normalisation effects (see Supplementary Material). Cluster 2 (blue) is robust to the additional inclusion of expression data, indicating that there is no significant disagreement amongst the three datasets regarding the existence of this cluster. Cluster 3 (red) is also relatively robust, with only one fewer gene when we consider the fusion of all 3 datasets, compared to the fusion of just the ChIP and PPI datasets (see Figures 4a and 4b). We note that the genes in Clusters 2 and 3 all have key roles, either encoding core histone proteins or being involved in ribosome biogenesis (see Table 3).

Interestingly, the gene lost from Cluster 3 (the histone cluster) is HTZ1, which encodes the variant histone H2A.Z (Jackson et al., 1996; Santisteban et al., 2000). The function of H2A.Z is different to that of the major H2As (e.g. Jackson and Gorovsky, 2000). We can see from Figure 4c (panel 3) that the expression of this gene (shown grey) is subtly different to the expression of others in the cluster.

4.4 Comparison to other methods

In Section G of the Supplementary Material, we provide a comparison of MDI to other clustering methods, both in terms of performance and the types of results that can be obtained. The key properties of MDI that distinguish it from other clustering methods are: (i) the clustering of genes in dataset } influences (and is influenced by) the clustering in dataset }, to an extent determined by the inferred } parameter; (ii) each dataset is permitted to have a different clustering structure (so each dataset may, for example, have a different number of clusters); (iii) the number of clusters is determined automatically as part of the inference procedure; and (iv) there is a correspondence between the cluster labels in different datasets, which enables us to identify clusters of genes that exist across some or all of the datasets. Simple clustering methods such as }-means and hierarchical clustering) can be used to cluster each of the datasets independently, but do not model the dependence/similarity between clustering structures in different datasets and do not enable clusters that exist across multiple datasets to be identified automatically. More sophisticated methods such as } (Shen et al., 2009) often share some of MDI’s properties, but do not allow for the identification of subsets of genes that cluster together across multiple datasets. The results of Section G of the Supplementary Material demonstrate that the ability to share information across datasets typically provides improvements in clustering quality, while MDI’s ability to pick out clusters that exist across multiple datasets permits the identification of groups of genes with very specific shared characteristics. Increasing the number of datasets across which we seek agreement in cluster assignment has the effect of increasing the specificity of these shared characteristics (which typically reduces the size of the gene subset – see Section 4.3 for further explanation).

4.5 Scaling and run-times

For typical examples (where the number of datasets, } is relatively small), the scaling of MDI will be } (see Supplementary Section D.5 for further details and specific run-times). MDI is particularly appropriate for applications in which a gene pre-selection step is performed (e.g. on the basis of differential expression). We anticipate applications to collections of ~5 datasets, each comprising ~1000 genes. Parallelising MDI using an approach such as the one described by Suchard et al. (2010) should be possible, and we are currently investigating this.

5 DISCUSSION

We have presented MDI, a novel Bayesian method for the unsupervised integrative modelling of multiple datasets. We have established that MDI provides competitive results with an existing method for integrating 2 datasets (Section 4.2), and is also able to integrate collections of more than 2 datasets (Sections 4.1 and 4.3). Our application to a 3-dataset example (Section 4.3) demonstrated that requiring agreement across multiple datasets of different types can enable us to identify clusters of genes with increasingly specific shared characteristics. Moreover, we have found that sharing information across multiple datasets can improve cluster quality.

MDI adopts a modelling approach distinctly different from those adopted by existing integrative modelling methods. For example, the model of Savage et al. (2010) performs integrative modelling of 2 datasets only, achieved by introducing a “fused context” (in which the 2 datasets are modelled together via a product of likelihoods) in addition to 2 “unfused contexts” in which the 2 datasets are modelled separately. This is analogous to introducing – and modelling – an additional dataset. In contrast, MDI introduces just a single parameter, }} for each pair of datasets (see Section 2.2), and it is this that provides MDI with the flexibility to perform integrative modelling of multiple datasets. The scalability of MDI may be further improved through parallelisation of the type described by Suchard et al. (2010). This is an important direction for future work.

ACKNOWLEDGEMENTS

We thank John Pinney for help with GO Term Overlap, and Maxime Huvet for useful discussions. Funding: PK, JEG, ZG and DLW acknowledge support from the EPSRC (grant EP/I036575/1). RSS was supported by an MRC Biostatistics Fellowship.

REFERENCES


