Open Access

Justina Žurauskienė[a], Paul D.W. Kirk[a] and Michael P.H. Stumpf*

A graph theoretical approach to data fusion

DOI 10.1515/sagmb-2016-0016

Abstract: The rapid development of high throughput experimental techniques has resulted in a growing diversity of genomic datasets being produced and requiring analysis. Therefore, it is increasingly being recognized that we can gain deeper understanding about underlying biology by combining the insights obtained from multiple, diverse datasets. Thus we propose a novel scalable computational approach to unsupervised data fusion. Our technique exploits network representations of the data to identify similarities among the datasets. We may work within the Bayesian formalism, using Bayesian nonparametric approaches to model each dataset; or (for fast, approximate, and massive scale data fusion) can naturally switch to more heuristic modeling techniques. An advantage of the proposed approach is that each dataset can initially be modeled independently (in parallel), before applying a fast post-processing step to perform data integration. This allows us to incorporate new experimental data in an online fashion, without having to rerun all of the analysis. We first demonstrate the applicability of our tool on artificial data, and then on examples from the literature, which include yeast cell cycle, breast cancer and sporadic inclusion body myositis datasets.

Keywords: clustering; data integration; functional genomics; graph-theoretic methods.

1 Introduction

Given the broad variety of high-throughput biological datasets now being generated, attention is increasingly focusing on methods for their integration (Kirk et al., 2012; Lock and Dunson, 2013; Wang et al., 2014). Different technologies allow us to probe different aspects of biological systems, and combining these complementary perspectives can yield greater insight (Altman, 2013; Savage et al., 2013; Schimek et al., 2015).

Computational techniques for data fusion are rapidly evolving, and moving towards potential clinical applications (Altman, 2013). For example, recently proposed methods try to identify cancer subtypes using fused similarity networks applied to a combination of DNA methylation, mRNA expression and miRNA expression datasets (Wang et al., 2014). Cancer subtype discovery has also been a focus of other recent data integration efforts (Yuan et al., 2011; Savage et al., 2013; Lock and Dunson, 2013). Further, data exploration via e.g. simultaneous clustering of gene expression networks (Narayanan et al., 2010), inference of transcriptional module networks (Lemmens et al., 2009), and bi-clustering with connected biologically relevant information (Reiss et al., 2006) are also becoming prominent in the recent data fusion literature.

Data integration methods can be categorized as either unsupervised (the subject of the present work), or supervised (Troyanskaya et al., 2003; Myers et al., 2005; Myers and Troyanskaya, 2007; Huttenhower et al., 2009). Existing unsupervised techniques for the fusion of multiple (i.e. more than two) datasets fall into one of two broad categories: those which are Bayesian (Kirk et al., 2012; Lock and Dunson, 2013) and those which are not (Shen et al., 2009; Wang et al., 2014). For many real-world or large-scale analyses the lack of suitable

[a]Justina Žurauskienė and Paul D.W. Kirk should be joint first authors.

***Corresponding author: Michael P.H. Stumpf,** Theoretical Systems Biology Group, Centre for Bioinformatics, Imperial College London, South Kensington Campus, SW7 2AZ, London, UK, e-mail: m.stumpf@imperial.ac.uk
Justina Žurauskienė: Theoretical Systems Biology Group, Centre for Bioinformatics, Imperial College London, South Kensington Campus, SW7 2AZ, London, UK
Paul D.W. Kirk: MRC Biostatistics Unit, Cambridge, CB2 0SR, Cambridge, UK

©2016, Michael P.H. Stumpf et al., published by De Gruyter.
This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 3.0 License.

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

training sets places severe limitations on supervised algorithms, and unsupervised learning methods have thus gained in popularity. Current Bayesian approaches for unsupervised data fusion rely on mixture model representations of each dataset, with dependencies between datasets modeled either using coefficients that describe the similarity between pairs of datasets (Kirk et al., 2012), or by assuming that each dataset has a structure that adheres – to a lesser or greater degree – to an overall consensus structure that is common to all datasets (Lock and Dunson, 2013). Bayesian approaches have the advantage of having firm probabilistic foundations, of forcing all prior beliefs and assumptions to be formally expressed from the outset, and of allowing the uncertainty in unknown quantities to be encapsulated in (samples from) posterior densities. For these reasons, Bayesian approaches are usually preferred; however, computational cost may prohibit their application to large (e.g. genome-scale) datasets.

In this work we introduce a new approach for the fusion of heterogeneous datasets; the methodology presented here is closely related to a graph-theoretic approach, which may be used to test for associations between disparate sources of data (Balasubramanian et al., 2004). Our approach has two basic steps. In the first, we obtain (independently) for each dataset either an ensemble of networks (in the Bayesian case), or a single network (in the non-Bayesian case), where networks are used to represent the structure and dependencies present in the dataset. In the second, we perform a post-processing step which compares the networks obtained for different datasets, in order to identify common edges, and to thereby assess and quantify the similarities in dataset structure.

Although, in principle, we could consider any type of structure that may be represented by a network, in the present work we are specifically interested in identifying and comparing the clustering structure possessed by each dataset, and it is at this level that we perform data fusion (Figure 1). In the Bayesian case, we employ for the purpose of cluster identification Dirichlet process mixture (DPM) models with either Gaussian

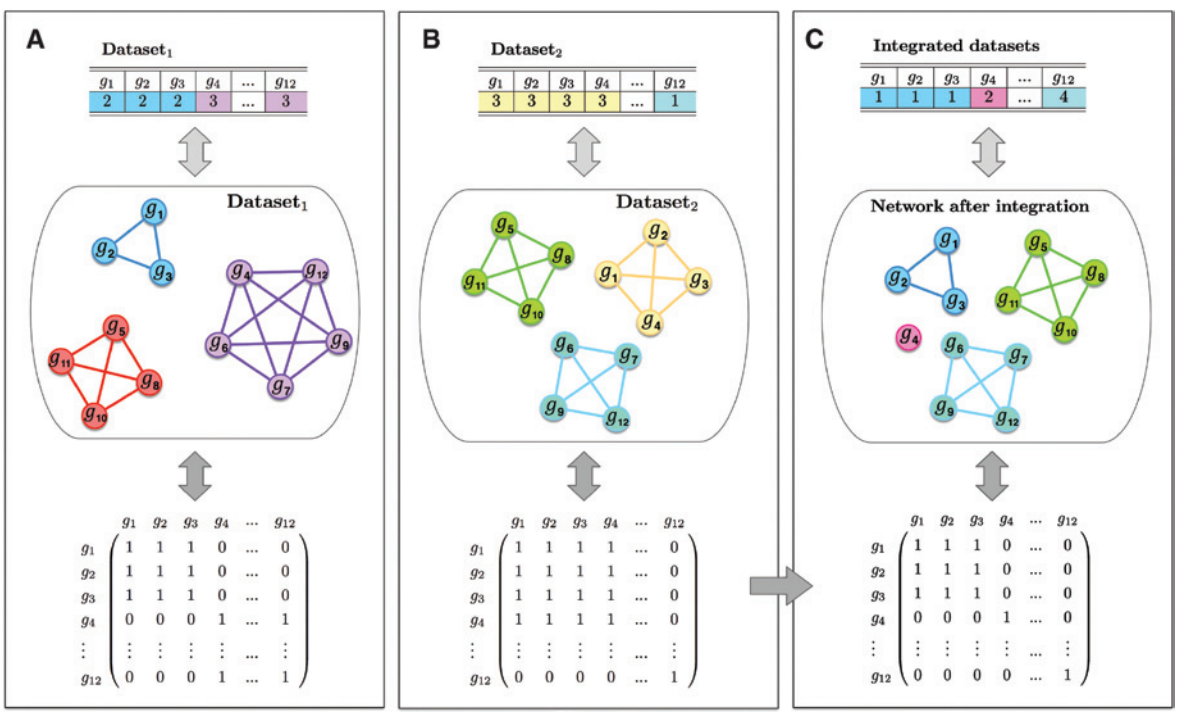

Figure 1: Method illustration. It is common to visualize the clustering outcome in a "table-like" fashion, by listing all genes next to their associated cluster labels. To visualize this, we construct corresponding graphs; here, each node in the network represents a gene and a line indicates that two genes cluster together. We use different color schemes to represent cluster labels. By adopting a graph-theoretical approach we can represent each network as an adjacency matrix, which in turn can be used for data integration. (A) Illustrates an artificial example of 12 genes that are assigned into three clusters (e.g. control case) from the first dataset. (B) Illustrates the same list of genes assigned into different three clusters (e.g. disease case) from a second dataset. (C) Illustrates the corresponding network (and cluster assignment) after performing data integration step.

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

process (GP) or multinomial likelihood functions. In our approach, data fusion is performed by constructing the connectivity networks that represent each clustering, and then forming "consensus networks" that identify the clusters upon which multiple datasets agree.

The integration step is somewhat similar to the consensus clustering approach (Monti et al., 2003), which was developed for the purpose of assessing cluster stability and for identifying the consensus across multiple evaluations of the same clustering approach. In contrast with other existing techniques (Kirk et al., 2012; Lock and Dunson, 2013), in our approach each dataset is initially considered independently. Although this is likely to result in some loss of sensitivity, it also means that computations can be performed in a parallel fashion, which offers potentially significant advantages in cases where we are dealing with large datasets, or where it is necessary to rerun computations in order to consider additional datasets.

2 Methods

Here we develop a novel *graph theoretical approach* for integrating clustering outcomes across several datasets, which we refer to as "GTA" for brevity. GTA may be applied to the output of Bayesian or non-Bayesian clustering algorithms.

2.1 Data integration

There are many different methods for data integration, most of which set out to accomplish one (or both) of the following two key aims: (i) modeling the dependencies that exist within and between datasets; and (ii) combining the predictions derived from one dataset with those derived from another. Assuming for convenience of notation that we wish to integrate just a pair of datasets, we might consider that the "ideal" way to fulfill the first of these two aims would be to model the joint distribution, $p(q^{(1)}, q^{(2)}|D^{(1)}, D^{(2)})$, of the predictions, $q^{(1)}$ and $q^{(2)}$, derived from datasets, $D^{(1)}$ and $D^{(2)}$, respectively (for the sake of generality, we leave the definition of q deliberately vague, but these predictions could be, for example, assessments of disease risk). Such an approach poses many potential challenges, not least that the datasets may be of very different types and/or may be of (very) high dimension. In contrast, the second aim just requires us to define a *fusion function*, f, for combining the predictions. We shall assume that this function is deterministic; i.e. for given predictions, $q^{(1)}$ and $q^{(2)}$, we assume that f maps the predictions onto a single combined output, $f(q^{(1)}, q^{(2)})$. One challenge in this case is to assess the uncertainty/confidence in this output. If we were able to sample M pairs, $(q_1^{(1)}, q_1^{(2)}), (q_2^{(1)}, q_2^{(2)}), \ldots, (q_M^{(1)}, q_M^{(2)})$ from the joint distribution, $p(q^{(1)}, q^{(2)}|D^{(1)}, D^{(2)})$, then we could consider the set $\{f(q_m^{(1)}, q_m^{(2)})\}_{m=1}^M$ in order to assess the variability in the output.

In general, modeling the joint distribution, $p(q^{(1)}, q^{(2)}|D^{(1)}, D^{(2)})$, is much more computationally demanding than defining a fusion function. To make modeling $p(q^{(1)}, q^{(2)}|D^{(1)}, D^{(2)})$ more tractable, here we make an independence assumption and factorize the joint distribution as,

$$p(q^{(1)}, q^{(2)}|D^{(1)}, D^{(2)}) \approx p(q^{(1)}|D^{(1)})p(q^{(2)}|D^{(2)}). \tag{1}$$

While, in practice, the quality of this approximation will depend upon a number of factors (including the signal-to-noise ratio associated with the individual datasets), it has the advantage of circumventing many of the challenges associated with trying to model $p(q^{(1)}, q^{(2)}|D^{(1)}, D^{(2)})$. Moreover, this independence assumption means that we may perform computations for each dataset in parallel, allowing us to scale to potentially large number of datasets. Having made this independence assumption, we may obtain samples from the joint distribution by sampling independently from each of the factors, $p(q^{(1)}|D^{(1)})$ and $p(q^{(2)}|D^{(2)})$. For a given fusion function, f, we may then consider the set $\{f(q_m^{(1)}, q_m^{(2)})\}_{m=1}^M$, as described above. Extending to R datasets is straightforward: we simply obtain samples independently from each $p(q^{(r)}|D^{(r)})$ for $r=1, \ldots, R$, and then consider the set $\{f(q_m^{(1)}, q_m^{(2)}, \ldots, q_m^{(R)})\}_{m=1}^M$.

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

2.2 GTA: a graph theoretic approach to unsupervised data integration

We start with a collection of datasets, $D^{(1)}, \ldots, D^{(R)}$, each of which comprises measurements taken on a common set of N entities/items (e.g. genes or patients). We consider the situation in which the prediction, $q^{(r)}$, associated with dataset $D^{(r)}$ is the clustering structure possessed by these items. To proceed, we must therefore define: (i) a method for sampling from $p(q^{(r)}|D^{(r)})$; and (ii) a fusion function, f, for combining the predicted clustering structures derived from the various datasets. Here, our aim is to identify similarities between the clustering structures associated with each of the datasets.

2.2.1 Sampling from $p(q^{(r)}|D^{(r)})$

We consider two different approaches for sampling from $p(q^{(r)}|D^{(r)})$. Where possible, we use Dirichlet process mixture modeling, a nonparametric Bayesian approach, which is discussed in the Appendix A. However, in some cases the size of the datasets being considered prohibits the use of such approaches. In these instances, rather than using an ensemble of M samples from $p(q^{(r)}|D^{(r)})$, we instead take M=1 and treat the maximum likelihood estimate, $q_{ML}^{(r)}$, as a single, representative sample.

2.2.2 Defining the fusion function

In order to define a fusion function, f, we take inspiration from Balasubramanian et al. (2004) work and adopt a graph theoretic approach. We define $\mathbf{c}$ to be a clustering of the N items of interest (genes, patients, etc.), so that c_i is the cluster label associated with item i. The $N \times N$ adjacency matrix is given as,

$$(A)_{ij} = \begin{cases} 1, & \text{if } c_i = c_j; \\ 0, & \text{otherwise.} \end{cases}$$

Given a collection of R such clusterings, say $\mathbf{c}^{(1)}, \ldots, \mathbf{c}^{(R)}$, we may form a corresponding collection of adjacency matrices, $A^{(1)}, \ldots, A^{(R)}$, where $A^{(k)}$ is the adjacency matrix representation of clustering $\mathbf{c}^{(k)}$. The Hadamard (entry-wise) product of these matrices, $H = A^{(1)} \circ \cdots \circ A^{(R)}$, defines a new clustering, in which items i and j appear in the same cluster if and only if all of the clusterings $\mathbf{c}^{(1)}, \ldots, \mathbf{c}^{(R)}$ agree that they should appear in the same cluster. From a graph theoretic perspective, H corresponds to the graph formed by taking the intersection of the graphs defined by $A^{(1)}, \ldots, A^{(R)}$ (see Figure 1). Assuming that each of the clusterings, $\mathbf{c}^{(1)}, \ldots, \mathbf{c}^{(R)}$, corresponds to a different dataset, we define our fusion function f to act on these and then return the clustering corresponding to the Hadamard product, H. Equipped with this fusion function, and M samples from $p(q^{(r)}|D^{(r)})$ (for r=1, …, R), we may proceed to describe with the GTA algorithm (Algorithm 1).

2.2.3 Summarizing the fused output

The output of the GTA algorithm is a collection of M *fused clusterings*. While these provide a useful indication of the uncertainty in the fused output, it is often also helpful to condense these into a single, summary fused clustering, $\bar{\mathbf{c}}$. This can be done by constructing a posterior similarity matrix (Fritsch and Ickstadt, 2009) of $\hat{\mathbf{c}}_1, \ldots, \hat{\mathbf{c}}_p$ and maximizing the posterior expected adjusted Rand index (Hubert and Arabie, 1985).

2.2.4 Heuristic score of clustering similarity

In some instances, it is useful to measure the compatibility between data sources by comparing the independent clusterings (obtained before data integration) with the fused clustering (obtained using the GTA

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

Algorithm 1: The GTA algorithm.

Input: datasets $D^{(1)},\ldots,D^{(R)}$;
Output: a collection of fused clusterings, $\mathbf{c}_1,\ldots,\mathbf{c}_M$;
GTA algorithm: **for** $m=1,\ldots,M$ **do**
  for $r=1,\ldots,R$ **do**
    if $M=1$ **then**
      set $q_m^{(r)}=q_{ML}^{(r)}$;
    else
      sample $q_m^{(r)}$ from $p(q^{(r)}|D^{(r)})$;
    end
    let $A_m^{(r)}$ be the adjacency matrix corresponding to $q_m^{(r)}$;
  end
  set $H_m=A_m^{(1)}\circ\cdots\circ A_m^{(R)}$;
  let $\mathbf{c}_m$ be the clustering corresponding to H_m;
end

algorithm). Due to the nature of our technique it is expected to observe more clusters after the data integration process. This is particularly true when studying less related datasets and when integrating more than a few data sources. We therefore define the following measure of similarity between any two data sources D_r and D_l,

$$S(D_r, D_l)=\frac{(K_{D_r}+K_{D_l})/2}{K_{D_r\circ D_l}}, \tag{2}$$

where K_{D_r} and K_{D_l} correspond to the number of clusters before data integration, and $K_{D_r\circ D_l}$, the number of clusters after integration. Here, the score S can have any value between zero and one. The closer S is to 0, the more dissimilar datasets are. On the other hand, the closer S is to 1, the more substantial the similarity between the structure of the two datasets.

3 GTA applications

In this section we explore the capabilities of GTA to data integration. Here we are interested in comparing our result to the result delivered by data fusion methods, which explicitly model relationships and dependencies between datasets. We start by illustrating the applicability of *heuristic score* to identify similarities between several data sources. Then we test GTA performance on popular examples from literature, and on sporadic inclusion body myositis dataset.

Example R code and instructions are available from https://sites.google.com/site/gtadatafusion/.

3.1 Capturing similarities across artificial dataset

We consider a *S. cerevisiae* dataset (Cho et al., 1998) that contains *m*RNA transcription levels taken to study the cell cycle. From 416 genes that had previously been identified to have periodic changes in transcript levels we select 100, and assign them into seven clusters using DPM with GPR likelihood model (Figure 2A). The details regarding MCMC specification and diagnostics are summarized in Appendix B. To demonstrate the performance of GTA, we further consider the artificial dataset example (Kirk et al., 2012) that was constructed using the same 100 genes. Briefly, this example consists of six data sources, where the first source is the original dataset (see Figure 2A), and the other five were obtained sequentially, by randomly permuting a quarter of gene names in the previous dataset. Applying GTA on pairwise combinations of these datasets we identify the numbers of genes that cluster together before and after the fusion. These can be used for computing the score of agreement and for determining the similarities across all six datasets. The pairwise similarities are summarized in Figure 2B where columns and rows identify which combination of datasets were considered,

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

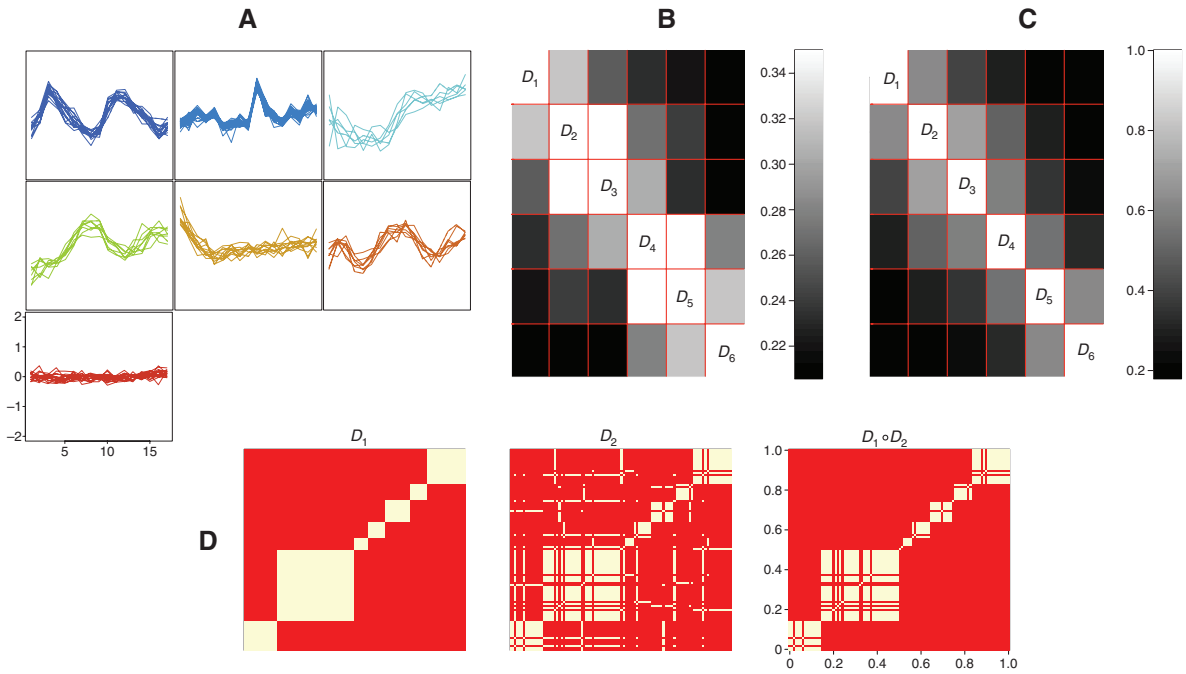

Figure 2: GTA applications to six artificial datasets. (A) Illustrates genes from Cho et al. (1998) sorted into seven clusters using DPM model with GP likelihood. (B) Illustrates similarity between six final clusterings using similarity measure $S(D_r, D_l)$ outlined in (2). (C) Illustrates similarity between the same clusterings using adjusted Rand index. (D) Illustrates the effects of Hadamard product – each heatmap corresponds to the adjacency matrix constructed from original dataset, D_1, and the first modified dataset, D_2.

and color illustrates the level of similarity. Alternatively, the similarity between these datasets can be identified from the final clusterings by computing the adjusted Rand index (ARI) (Hubert and Arabie, 1985). The ARI compares two given partitions of the same list of genes and is based on how often a gene (observation) is associated with the same cluster in both partitions (see Figure 2C).

3.2 Integrating cell cycle datasets

We next compare the results from GTA to the results by *Multiple Dataset Integration* (MDI) (Kirk et al., 2012). In this example we consider integrating a number of datasets from yeast cell cycle studies. The first dataset contains gene expression time courses (Granovskaia et al., 2010), where *m*RNA measurements are taken at 41 time points across 551 genes that exhibit oscillatory expression profiles. The second dataset is ChIP-chip data (Harbison et al., 2004) that contains binary information about proteins binding to DNA, and the third is a protein-protein interaction (PPI) dataset from BioGRID (Stark et al., 2006).

In order to apply GTA for data fusion, we initially cluster all datasets by running in parallel three independent DPM models (see Appendices A and B). To specify the DPM, we use Gaussian process likelihood to cluster gene expression time courses, and multinomial likelihood to cluster the transcription factor binding and PPI datasets. As before, further details regarding MCMC specifications and diagnostic plots are summarized in Appendix B. Although we use DPM here to automatically determine the number of cluster for each dataset; it is worth noting that initial clustering could also be done using other methods since GTA operates on pre-clustered dataset.

In this example we are aiming to compare the results from our method to the results from MDI, for this reason we consider two cases: (i) integration of gene expression and transcription factor binding data; (ii) integration of gene expression, transcription factor binding, and protein-protein interaction data.

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

MDI is a tool that jointly clusters all datasets by modeling dependencies between them. Using MDI approach the final clustering can be extracted by calculating a fusion probability – the probability that any two genes are fused in two datasets, and by removing those genes where this probability is less than 0.5. For this reason, using GTA we can also consider removing genes that lack the evidence of clustering together. This can be achieved by computing the matrix $\mathbf{P}=\frac{1}{M}\sum_{m=1}^{M} H_m$, where each matrix entry is probability, P_{ij}, for gene i and gene j to be in the same cluster in both gene expression and transcription factor binding datasets; and removing genes i, where $P_{ij}<0.5$ for every j. The above procedure is somewhat analogous to MDI in terms of looking only at those genes that are fused across both datasets. Then, applying the approach of Fritsch and Ickstadt (2009) to filtered posterior clusterings we obtain the final data partition that assigns genes into clusters based on information from both datasets. Figure 3A,C illustrates the performance of GTA, where genes are allocated into clusters based not only of their expression profiles but also on which transcription factors bind to DNA. In order to compare our results to the results obtained by MDI we use overlap score (OS) metrics (Nepusz et al., 2012; Zhang et al., 2015), $\mathrm{OS}(A, B)=|A\cap B|^2/|A||B|$, where $|\cdot|$ denotes the number of elements in the set; A is predicted network/clustering structure by GTA, and B is a benchmark structure, in this example it is obtained using MDI. A predicted clustering can be considered as a match to the benchmark clustering if their OS value is no less than a predefined threshold, which typically can be set to 0.2 (Bader and Hogue, 2003; Wu et al., 2009; Zhang et al., 2015). In this example we obtained OS=0.426, which indicates that GTA can provide the approximation to MDI result. In order to better understand the difference between both clusterings we looked at corresponding network structures (Figure 3A,B). In this case MDI maintains densely connected clusters, whereas the GTA output is dominated by small clusters. Although GTA is less sensitive in

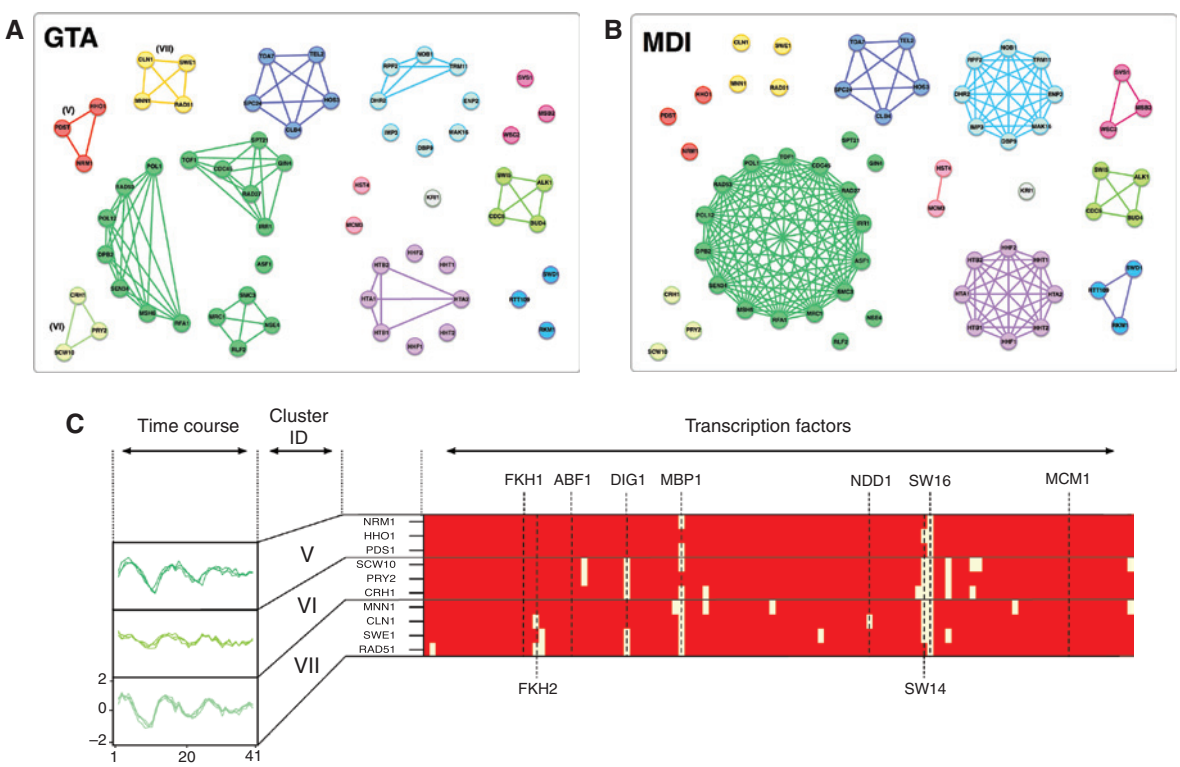

Figure 3: GTA applications to yeast cell cycle datasets. Integrating yeast cell cycle time course and transcription factor (TF) binding datasets. Here a line indicates that any two genes are assigned to the same cluster after the fusion ((A)/(B)). (A) Networks illustrates final clustering structure using fused samples from GTA. (B) Networks illustrates final clustering structure using MDI. (C) Further illustrate three additional clusters identified by GTA; on the left – gene expression time courses are projected on a heatmap of TF binding data on the right, here yellow color indicated that a TF is binding to a gene. Horizontal black lines mark cluster boundaries.

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

detecting larger clustering structures, it took less than a minute for algorithm to fuse M=1000 samples from posterior (running time=40.665 s, was obtained using R function *proc.time*()); it was reported (Kirk et al., 2012) that it took approximately 2 hours to execute MDI on the same example using bag-of-words model, and approximately 4 hours to run it using multinomial model. This suggest that GTA can be seen as a useful approximation to MDI, which offers considerable savings on computational times.

As a second step in our yeast cell cycle example we consider integrating three datasets: gene expression, transcription factor binding and protein-protein interaction (PPI) data. In order to also fuse the PPI dataset using MDI, it is necessary to rerun joint clustering from the beginning; however, using GTA we can benefit from previously pre-clustered data and perform data fusion in an "online-fashion" using only samples from the posterior. In this case GTA fuses three datasets in under a minute (40.889 s), whereas for MDI it took to generate approximately 90 samples per chain per hour Kirk et al. (2012). Applying thinning-out strategy (removing genes that lack the evidence of clustering together) we obtained a set of 14 genes that can be assigned into 6 clusters (for comparison MDI assigned 16 genes into 5 clusters). Supplementary Figures 3A,B show the final network structures that correspond to fused three datasets using GTA and MDI, respectively; in addition, Supplementary Figure 2 projects clusters from GE data on to TF and PPI. For convenience, Supplementary Material Tables 3 contains further details regarding gene function. In addition to this, we looked at the overlap score, OS=0.214, which is above predefined threshold and suggests that GTA results can be seen as a useful approximation.

3.3 Breast cancer data

In this example we explore the performance of our data integration technique on a breast cancer dataset, where we aim to integrate four different data sources taken from *The Cancer Genome Atlas* (Cancer Genome Atlas Network, 2012). We use a previously described dataset (Lock and Dunson, 2013) that consists of preselected 348 tumor samples characterized in four different ways: RNA gene expression (645 genes), DNA methylation (574 probes), *m*iRNA expression (423 *m*iRNAs), and reverse phase protein array (171 proteins). In order to cluster all data sources, we adopt a modified version of Bayesian consensus clustering (BCC) approach Lock and Dunson (2013). BCC is a data integration technique that seeks to simultaneously model data specific and shared features by inferring the overall clustering $\hat{C}$ (that describes all datasets), and by inferring data specific clusterings $\hat{L}_i$, i=1, …, 4. The source specific clustering is controlled by parameter $\alpha=[\alpha_1, \ldots, \alpha_4]$, which express the probability of how much each L_i contributes to the overall $\hat{C}$. Our goal is to pre-cluster each dataset without inferring the overall clustering $\hat{C}$ and then use GTA to fuse all four datasets. For this reason we fix the probability α=1 and cluster each dataset independently using publicly available R code (Lock and Dunson, 2013). Next, applying GTA with M=1000 on posterior samples we can identify the fused clustering $\bar{\mathbf{c}}$. To run GTA on pre-clustered data takes under a minute (running time was 40 s on a MacBook Pro), however to run a full joint clustering using BCC took about half an hour. In order to compare both clusterings we looked at the overlap scores (see Table 1), which indicate that GTA result can be considered as a fast approximation to the BCC fused clustering.

Breast cancer is a heterogeneous disease, for this reason four biologically distinct molecular subtypes can be connected to these dataset (Cancer Genome Atlas Network, 2012; Lock and Dunson, 2013). They are known as *Her2*, *Basal*, *Luminal A*, *Luminal B*, and are associated with different clinical prognosis (Rakha et al., 2008; Dawood et al., 2011). In order to further assess our results we look at cancer subtypes that are associated with each cluster and compared these to the BCC clusters. Confusion matrices in Table 2 illustrates the summarized results. The first matrix corresponds to the outcome using BCC method (a single run

Table 1: Overlap scores between GTA clusters and corresponding BCC clusters.

$OS(Cl_1^{gta}, Cl_1^{bcc})=0.57$
$OS(Cl_2^{gta}, Cl_2^{bcc})=0.89$
$OS(Cl_3^{gta}, Cl_3^{bcc})=0.69$

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

Table 2: Comparison between BCC overall and GTA final clusterings.

TCGA tumor subtypes	BCC single run			GTA		
	Cl_1	Cl_2	Cl_3	Cl_1	Cl_2	Cl_3
Her2	5	14	20	28	6	5
Basal	3	**65**	4	6	**65**	1
Luminal A	**77**	3	**81**	**98**	4	**59**
Luminal B	3	0	**73**	17	0	**59**

In the table are given numbers of tumor samples per cluster; e.g. GTA cluster 3 contains six samples of *Her2*, 65 samples of *Basal* and four samples of *Luminal A*, for this reason, cluster 3 can be summarized as containing mostly *Basal* type tumors. Clusters are classified by particular cancer subtype using publicly available R code (Lock and Dunson, 2013). In bold we highlight tumor samples that could be associated with cluster identity.

of publicly available code), while the second matrix summarize results from GTA. This further demonstrates that clusters identified using our technique bear similarity to the BCC results.

3.4 Sporadic inclusion body myositis

In this section we apply our technique on clinical gene expression datasets that include: (i) *sporadic Inclusion Body Myositis* (sIBM), which is an inflammatory muscle disease that progresses very slowly, causes muscular weakness and eventually muscle atrophy (Grau and Selva-O'Callaghan, 2008; Machado et al., 2009); (ii) *polymyositis* (PM) which causes chronic inflammation of the muscles; and (iii) a dataset containing human protein-protein interactions (BPPI) (see Supplementary Material, Section B for further details). Both sIBM and PM are associated with aging but interestingly sIBM can be frequently misdiagnosed as PM, and the explicit diagnosis can only be confirmed via a muscle biopsy (Dalakas, 2006). Current understanding is that sIBM is driven by two coexisting processes (autoimmune and degenerative); however, the actions by which sIBM occurs are still only poorly understood (Dalakas, 2006; Needham and Mastaglia, 2007).

3.4.1 PM and sIBM data integration

Here we focus on experimental sIBM and PM datasets that have 5 and 3 data points (clinical cases). In order to apply our technique, we select 424 genes that where previously identified to have the largest variation in their expression across all data points (Thorne et al., 2013). In order to cluster these datasets, we employ "*mclust*" package in R, which fits a Gaussian mixture model and uses the Bayesian information criterion to estimate the number of components. Because we do not have access to the clustering samples from the posterior for each dataset, we use GTA with a special case $M=1$ to integrate single clusterings from both datasets. The integrative analysis enables partitioning of large PM and sIBM clusters into smaller ones; however, this results in a greater number of clusters after the fusion, see Figure 4. To validate our results we use the "ToppGene" (https://toppgene.cchmc.org) tool, which performs gene set enrichment analysis and allows the detection of functional enrichment for phenotype (disease) and GO terms such as biological function. Such analysis enabled us to identify those clusters (5 out of 12) that are enriched with diseases like rheumatoid arthritis and myositis, and biological processes related to response from immune system. This suggests that genes in disease enriched clusters might play an important and shared role in both PM and sIBM, and for this reason they are promising targets for further analysis.

4 Discussion

In this paper we have developed and illustrated an alternative way to integrate data generated from different sources. GTA mainly relies on the outcomes from Bayesian clustering approaches and is based on concepts

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

Figure 4: GTA applications to myositis datasets. Illustrate clusters obtained by integrating PM and sIBM gene expression datasets using GTA. Each network represents a fused cluster, which is associated with one of enriched ontologies, biological process (BP), and disease (D).

from graph theory. We have demonstrated that our technique, can produce similar data fusion results when compared to recently proposed data integration methods. Equally, while GTA can fuse the outcomes from Bayesian clustering algorithms, the special case M=1 can be applied in order to fuse a single clusterings across various datasets.

Here we have demonstrated the applicability of our technique to a variety of typical biological and biomedical problems: the identification of potentially underlying regulatory mechanisms in the yeast cell cycle; sub-typing tumors in breast cancer data; and exploring similarity patterns across inflammatory muscle

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

diseases. We have compared our technique to MDI and BCC, the current state-of-the-art techniques used to address similar data integration problems. The main benefits of our graph-theoretical approach include: (i) the applicability to Bayesian and non-Bayesian type clustering approaches. This means that our methodology can be applied in order to model multiple sources on a genome scale data without facing computational challenges; and (ii) ability to perform clustering in a parallel fashion. These features might be advantageous in situations where it is necessary to rerun computations in order to consider additional datasets. As part of our modeling approach, we have defined a measure of similarity between two data sources. This can be used to evaluate the effects of data integration routine and in order to assess the agreement between data sources.

We note that it is possible to apply our graph-theoretical approach to the outcomes from simple clustering techniques, for example hierarchical or K-means clustering. This could be done by first performing a bootstrapping approach on each gene within a dataset D_r, M times; for further details on bootstrapping see (Efron, 1981) and (Kerr and Churchill, 2001). This would produce a set of bootstrapped datasets $D_r^{b_1}, \ldots, D_r^{b_M}$. The bootstrapped datasets together with the initial dataset can be clustered using a standard hierarchical clustering algorithm. The clustering outcomes can to some extent be viewed as being the analogs to the samples from posterior. Then after bootstrapping all data sources we can use GTA to perform integrative modeling. This process can be seen as a frequentist modeling approach to data integration.

Acknowledgments: This work was supported by the Leverhulme Trust (to JŽ and MPHS), the Royal Society (to MPHS), HFSP (to PK and MPHS) and BBSRC (to MPHS).

Appendix

A Further details regarding clustering using Dirichlet processes

A.1 Dirichlet process mixtures

The methodology for modeling heterogeneity in genomic datasets bears similarity to the structure of an infinite Gaussian mixture model. A Dirichlet process mixture (DPM) model can be derived as a limit of a finite mixture model when the number of mixture components grows to infinity. Precisely, let consider a dataset $D=\{x_1, \ldots, x_N\}$ that we intend to model by the following mixture model,

$$p(x_1, \ldots, x_N|\pi, \theta) \sim \sum_{k=1}^{K} \pi_k \mathscr{F}(D|\theta_k), \quad \pi_k > 0, \quad \sum_{k=1}^{K} \pi_k = 1, \tag{3}$$

where K is the number of components, π_k are the mixing proportions, and $\mathscr{F}(D|\theta_k)$ are component density functions parameterised with a set of parameters θ_k. Furthermore, we associate each data point, x_i, with a component indicator variable $c_i \in \{1, \ldots, K\}$. This allow us to track which mixture component generated a data point x_i. We can allocate a symmetric Dirichlet prior to the mixing proportions, $p(\pi_1, \ldots, \pi_K|\alpha) = \Gamma(\alpha)/\Gamma(\alpha/K)^K \prod_{k=1}^{K} \pi_k^{\alpha/K-1}$, where α is a concentration parameter; and a multinomial prior, $p(c_1, \ldots, c_K|\pi) = \prod_{k=1}^{K} \pi_k^{n_k}$, to the indicator variables with n_k indicating the number of times $c_i = k$ (the number of observations that have same indicator value). Then a conditional prior for a single indicator, all others being given, is obtained by integrating over the mixing proportions

$$p(c_i = k|c_{-i}, \alpha) = \frac{n_{-i,k} + \alpha/K}{N-1+\alpha},$$

where the subscript, "$-i$," is a short notation for all indicators excluding i; and $n_{-i,k}$ denotes the number of observations within the cluster k not including observation x_i. Now, by taking the limit as K goes to infinity the conditional prior has the following limits,

$$p(c_i = k|c_{-i}, \alpha) = \frac{n_{-i,k}}{N-1+\alpha}, \tag{4a}$$

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

$$p(c_i \neq c_{i'},\ i' \neq i | c_{-i},\ \alpha) = \frac{\alpha}{N-1+\alpha}. \tag{4b}$$

Combining conditional priors (4) with a likelihood function, $\mathscr{F}(x_i|\theta_k)$, will result in a conditional posteriors

$$p(c_i = k | c_{-i},\ \alpha) \propto \frac{n_{-i,k}}{N-1+\alpha} \mathscr{F}(x_i|\theta_k), \tag{5a}$$

$$p(c_i \neq c_{i'},\ i' \neq i | c_{-i},\ \alpha) \propto \frac{\alpha}{N-1+\alpha} \int \mathscr{F}(x_i|\theta_j) H(\theta_j) d\theta_j, \tag{5b}$$

that are necessary to perform the inference of all parameters associated with model (3), including the number of clusters K. This can be achieved via Markov chain Monte Carlo (MCMC) methods. In equations (5), H denotes a prior for parameters θ_k, which might be a conjugate prior and depends on the likelihood model.

The DPM model is a general framework, and here we will focus on specific likelihood models. Specifically, we will employ DPM of Gaussian process regression (GPR) models to cluster gene expression time series, and DPM of multinomial models to model categorical/discrete functional genomics data. These two likelihood options are incorporated in our *Matlab* package *DPMSysBio*.

A.2 Likelihood function for the time course data

For convenience, bellow we will use the following notation, $\mathbf{x_i} \equiv x_i$. Instead of specifying a parametric (e.g. a multivariate-Gaussian) likelihood function, in this work we capture the time course observations $\mathbf{x_i} = \{x_i(t_1), \ldots, x_i(t_p)\}$, where $x_i(t_j)$ denotes the measurement taken on gene i at time point t_j, with a regression model. In a regression approach, each gene x_i can be expressed as

$$x_i(t_j) = f_i(t_j) + \varepsilon_{ij},$$

where f_i is a regression function and $\varepsilon_{ij} \sim \mathscr{N}(0, \sigma^2)$ is added to express the potential uncertainty in measurements. In our case we are modeling observations (genes) that tend to cluster together. This means that each cluster can be described by the same "data generating" function $f_i \equiv f_k$ and noise $\sigma_i^2 \equiv \sigma_k^2$ model; here, $k=1, \ldots, K$. In order to identify function $f_k = [f_k(t_1), \ldots, f_k(t_p)]$ for each cluster, we adopt a Bayesian nonparametric approach by specifying a Gaussian process prior for the function f_k. Formally, a Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution Rasmussen and Williams (2006). In order to specify a GP prior we need to define two main characteristics – a mean, m, and covariance, cov, functions. Such GP prior allow us to describe the Gaussian distributions that are associated with unique gene clusters. In simple terms this means that the function f_k evaluated at a finite number of input points $t_1, \ldots, t_p$ will have a multivariate Gaussian distribution with zero mean and there exists a covariance function, cov, such that,

$$[f_k(t_1), \ldots, f_k(t_p)]^T \sim \mathscr{N}(0, \mathrm{cov}(t_i, t_j)); \quad t_i, t_j \text{–are any two inputs.}$$

Here, for simplicity, we adopted a zero mean function ($m(t)=0$, for all t) and squared exponential function,

$$\mathrm{cov}(t_i, t_j) = a_k^2 \exp\left(-\frac{(t_i - t_j)^2}{2l_k}\right), \tag{6}$$

where $a_k, l_k > 0$ are the hyper-parameters. Then, the genes (observations) within each cluster, k,

$$\mathbf{x}_1^{(k)}, \ldots, \mathbf{x}_{N_k}^{(k)} | f_k, \sigma_k^2 \sim \mathscr{N}(f_k, \sigma_k^2 I_p).$$

Here, N_k is the number of observations in cluster k. For convenience we can rewrite the above in an expanded form,

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

$$[x_1^{(k)}(t_1), \ldots, x_{N_k}^{(k)}(t_1), \ldots, x_1^{(k)}(t_p), \ldots, x_{N_k}^{(k)}(t_p)]|f_k, \sigma_k^2$$
$$\sim\mathcal{N}([f_k(t_1), \ldots, f_k(t_1), \ldots, f_k(t_p), \ldots, f_k(t_p)], \sigma_k^2 I_{N_k p}),$$

where $[f_s(t_1), \ldots, f_k(t_1), \ldots, f_k(t_p), \ldots, f_k(t_p)]^T$ contains N_k replicates of each $f_k(t_j)$.

Now, we can define a Gaussian process prior,

$$[f_k(t_1), \ldots, f_k(t_1), \ldots, f_k(t_p), \ldots, f_k(t_p)]^T|a_k, l_k \sim\mathcal{N}(\mathbf{0}, \mathrm{cov}^{(k)}).$$

Here $\mathrm{cov}^{(k)}$ is an $N_k p \times N_k p$ matrix that is composed of smaller block matrices,

$$\begin{pmatrix} [\mathrm{cov}(t_1, t_1)] & \cdots & [\mathrm{cov}(t_1, t_p)] \\ \vdots & \ddots & \vdots \\ [\mathrm{cov}(t_p, t_1)] & \cdots & [\mathrm{cov}(t_p, t_p)] \end{pmatrix},$$

where $[\mathrm{cov}(t_i, t_j)]$ denotes i, j-th a smaller matrix structure. Here, each block matrix is symmetric and positive definite. This enable us to specify the following likelihood function within each cluster k,

$$[x_1^{(k)}(t_1), \ldots, x_{N_k}^{(k)}(t_1), \ldots, x_1^{(k)}(t_p), \ldots, x_{N_k}^{(k)}(t_p)]|a_k, l_k, \sigma_k \sim\mathcal{N}(\mathbf{0}, \mathrm{cov}^{(k)}+\sigma_k^2 I_{N_k p}) \tag{7}$$

A.3 Likelihood function for the discretised data

For convenience in this section we will describe a multinomial model Kirk et al. (2012) to capture categorical data. Typically, the categorical dataset consist of a list of genes (objects) where measurements, $r \in \{1, \ldots, R\}$, for each gene are taken at Q distinctive attributes. For genes that tend to cluster together, x_{rq} denotes the number of times q-th attribute receives a value r. Thus, the multivariate probability mass function for categorical data,

$$p(\mathbf{x}_q|\theta_{1q}, \ldots, \theta_{Rq}) \propto \prod_{r=1}^{R} \theta_{rq}^{x_{rq}}, \quad \sum_{r=1}^{R} \theta_{rq} = 1,$$

where $\mathbf{x}_q = [x_{1q}, \ldots, x_{Rq}]$; and θ_{rq} denotes the cluster related probability for attribute q to receive a value r.

Setting a Dirichlet prior, $\mathcal{D}(\beta_{1q}, \ldots, \beta_{Rq})$, for $\theta_{1q}, \ldots, \theta_{Rq}$, we obtain

$$p(\mathbf{x}_q|\beta_{1q}, \ldots, \beta_{Rq}) = \frac{\Gamma(B_q)}{\Gamma(S_q+B_q)} \prod_{r=1}^{R} \frac{\Gamma(x_{rq}+\beta_{rq})}{\Gamma(\beta_{rq})},$$

where $B_q = \beta_{1q}, \ldots, \beta_{Rq}$ and $S_q = x_{1q}, \ldots, x_{Rq}$. From the independence between the attributes follows the marginal likelihood function,

$$f(\mathbf{x}_1, \ldots, \mathbf{x}_Q|\beta) = \prod_{q=1}^{Q} \frac{\Gamma(B_q)}{\Gamma(S_q+B_q)} \prod_{r=1}^{R} \frac{\Gamma(x_{rq}+\beta_{rq})}{\Gamma(\beta_{rq})},$$

where $\beta_{R\times Q}$ is a matrix.

B Inference of the hyper–parameters

In this section we explain how inference is performed for DPM models. Additionally, we provide the Markov chain Monte Carlo (MCMC) run details for *S. cerevisiae* and *yeast cell cycle* (examples in the paper).

In the *DPMSysBio* package, the Gaussian process likelihood is controlled by three hyper-parameters, $\theta_c = \{a_c, l_c, \sigma_c\}$, that are necessary in order to learn the means and covariances of each cluster. We set a Gaussian

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

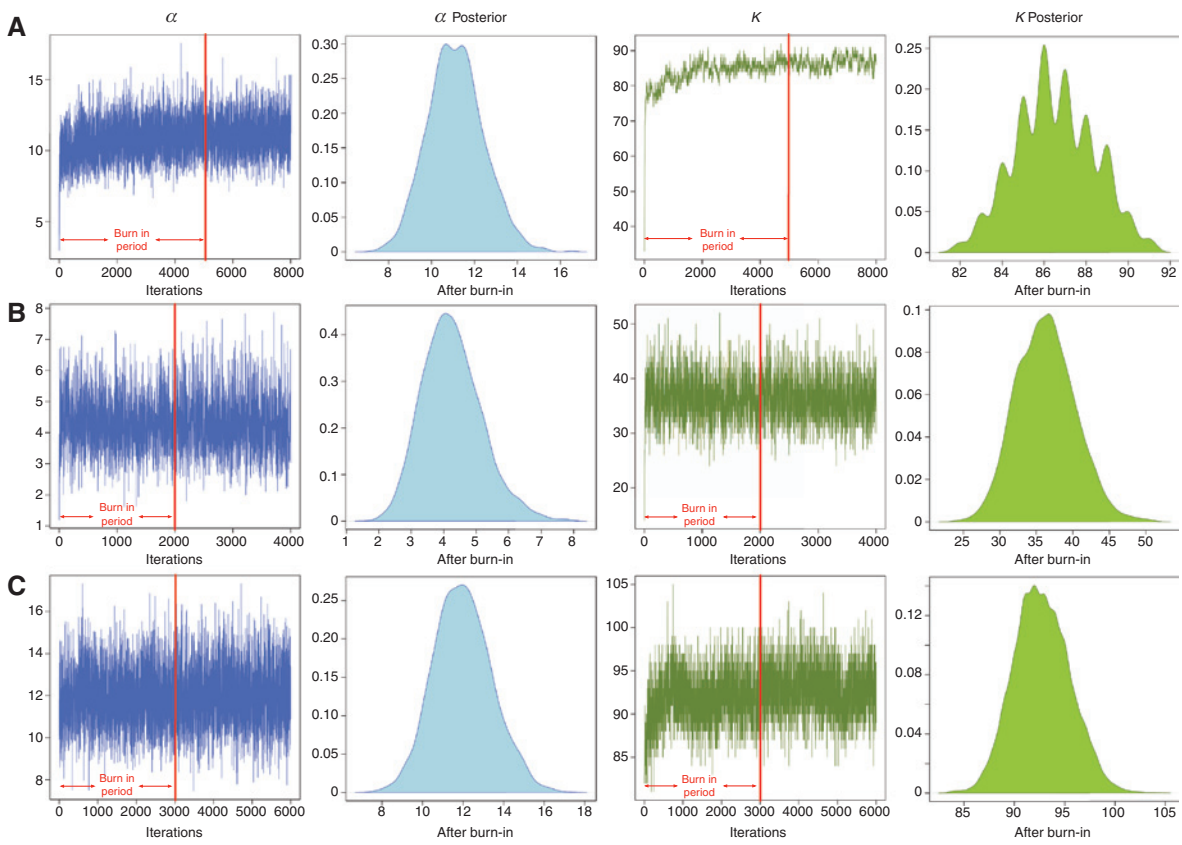

Figure 5: MCMC diagnostics plots. In the first column figures illustrate a "thinned-out" Markov chains for the concentration parameter, α; in the second column are given posterior distributions for α; in the third column – a "thinned-out" Markov chains for the number of clusters, K; in the fourth – posterior distributions for K. (A) Time course datasets Granovskaia et al. (2010). (B) Transcription factor binding dataset Harbison et al. (2004). (C) Protein-protein interaction dataset from BioGRID Stark et al. (2006).

priors for the logarithmic versions of hyper-parameters ($\log(\theta_c)$) and employ a Gibbs sampling algorithm as described in Neal (2000) (see Section 6 for details). This algorithm can be seen as the most general Gibbs sampling scheme that can deal with a non-conjugate priors. In order to learn the hyper-parameters of DPM model with multinomial likelihood, we use Dirichlet priors.

Finally, to infer the concentration parameter α we set the following *gamma* prior $\alpha \sim \mathcal{G}(2, 4)$, and adopt approach Escobar and West (1995) proposed by Escobar and West.

B.1 *S. cerevisiae* example

For the original time courses Cho et al. (1998) and five perturbed datasets, we ran *DPMSysBio* package with GPR likelihood for 50,000 iterations recording each 5th sample. This provided us with 10,000 "thinned-out" MCMC samples. Further, we discarded the first 5000 samples as a "burn-in" period and for further analysis we used 5000 samples per each dataset.

B.2 Yeast cell cycle datasets example

– For time course dataset (Granovskaia), we ran *DPMSysBio* package with GP likelihood for 40,000 iterations recording each 5th sample. This provided us with 8000 "thinned-out" MCMC samples.

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

Further, we discarded the first 5000 samples as a "burn-in" period and for further analysis we used 3000 samples.

- For transcription factor binding dataset (Harbison), we ran *DPMSysBio* package with multinomial likelihood for 20,000 iterations recording each 5th sample. This provided us with 4000 "thinned-out" MCMC samples. Further, we discarded the first 2000 samples as a "burn-in" period and for further analysis we used 2000 samples.
- For protein-protein interaction dataset (Biogrid), we ran *DPMSysBio* package with multinomial likelihood for 30,000 iterations recording each 5th sample. This provided us with 6000 "thinned-out" MCMC samples. Further, we discarded the first 3000 samples as a "burn-in" period and for further analysis we used 3000 samples.

Figure 5 illustrates the Markov chains and posterior distributions for the number of clusters, K, and the concentration parameter, α, for Granovskaia (TC), Harbison (TF) and Biogrid (PPI) datasets.

References

Altman, R. B. (2013): "Personal genomic measurements: the opportunity for information integration," Clin. Pharmacol. Ther., 93, 21–23.

Bader, G. D. and C. W. Hogue (2003): "An automated method for finding molecular complexes in large protein interaction networks," BMC Bioinformatics, 4, 2.

Balasubramanian, R., T. LaFramboise, D. Scholtens, and R. Gentlman (2004): "A graph-theoretic approach to testing associations between disparate sources of functional genomics data," Bioinformatics, 20, 3353–3362.

Cancer Genome Atlas Network (2012): "Comprehensive molecular portraits of human breast tumours," Nature, 490, 61–70.

Cho, R. J., M. J. Campbell, E. A. Winzeler, L. Steinmetz, A. Conway, L. Wodicka, T. G. Wolfsberg, A. E. Gabrielian, D. Landsman, D. J. Lockhart and R. W. Davis (1998): "A genome-wide transcriptional analysis of the mitotic cell cycle," Mol. Cell, 2, 65–73.

Dalakas, M. C. (2006): "Sporadic inclusion body myositis – diagnosis, pathogenesis and therapeutic strategies," Nat. Clin. Pract. Neurol., 2, 437–447.

Dawood, S., R. Hu, M. D. Homes, L. C. Collins, S. J. Schnitt, J. Connolly, G. A. Colditz and R. M. Tamimi (2011): "Defining breast cancer prognosis based on molecular phenotypes: results from a large cohort study," Breast Cancer Res. Treat., 126, 185–192.

Efron, B. (1981): "Nonparametric estimates of standard error: the jackknife, the bootstrap and other methods," Biometrika, 68, 589–599.

Escobar, M. and M. West (1995): "Bayesian density estimation and inference using mixtures," J. Am. Statist. Assoc., 90, 577–588.

Fritsch, A. and K. Ickstadt (2009): "Improved criteria for clustering based on the posterior similarity matrix," Bayesian Anal., 4, 367–391.

Granovskaia, M. V., L. J. Jensen, M. E. Ritchie, J. Toedling, Y. Ning, P. Bork, W. Huber and L. M. Steinmetz (2010): "High-resolution transcription atlas of the mitotic cell cycle in budding yeast," Genome Biol. 11, R24.

Grau, J. M., and A. Selva-O'Callaghan (2008): "Sporadic inclusion body myositis," In: Diagnostic criteria in autoimmune diseases. New York, NY: Humana Press, 165–168.

Harbison, C. T., D. B. Gordon, T. I. Lee, N. J. Rinaldi, K. D. Macisaac, T. W. Danford, N. M. Hannett, J.-B. Tagne, D. B. Reynolds, J. Yoo, E. G. Jennings, J. Zeitlinger, D. K. Pokholok, M. Kellis, P. A. Rolfe, K. T. Takusagawa, E. S. Lander, D. K. Gifford, E. Fraenkel and R. A. Young (2004): "Transcriptional regulatory code of a eukaryotic genome," Nature, 431, 99–104.

Hubert, L. and P. Arabie (1985): "Comparing partitions," J. Classif., 2, 193–218.

Huttenhower, C., E. M. Haley, M. A. Hibbs, V. Dumeaux, D. R. Barrett, H. A. Coller, and O. G. Troyanskaya (2009): "Exploring the human genome with functional maps," Genome Res., 19, 1093–1106.

Kerr, M. K. and G. A. Churchill (2001): "Bootstrapping cluster analysis: assessing the reliability of conclusions from microarray experiments," Proc. Natl. Acad. Sci. USA, 98, 8961–8965.

Kirk, P., J. E. Griffin, R. S. Savage, Z. Ghahramani and D. L. Wild (2012): "Bayesian correlated clustering to integrate multiple datasets," Bioinformatics, 28, 3290–3297.

Lemmens, K., T. De Bie, T. Dhollander, S. C. De Keersmaecker, I. M. Thijs, G. Schoofs, A. De Weerdt, B. De Moor, J. Vanderleyden, J. Collado-Vides, K. Engelen and K Marchal (2009): "Distiller: a data integration framework to reveal condition dependency of complex regulons in escherichia coli," Genome Biol., 10, R27.

Lock, E. F. and D. B. Dunson (2013): "Bayesian consensus clustering," Bioinformatics, 29:2610–2616.

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM

Machado, P., A. Miller, J. Holton and M. Hanna (2009): "Sporadic inclusion body myositis: an unsolved mystery," Acta Reumatol. Port., 34, 161–182.
Monti, S., P. Tamayo, J. Mesirov, and T. Golub (2003): "Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data," Machine Learn., 52, 91–118.
Myers, C. L., D. Robson, A. Wible, M. A. Hibbs, C. Chiriac, C. L. Theesfeld, K. Dolinski and O. G. Troyanskaya (2005): "Discovery of biological networks from diverse functional genomic data," Genome Biol., 6, R114.
Myers, C. L. and O. G. Troyanskaya (2007): "Context-sensitive data integration and prediction of biological networks," Bioinformatics, 23, 2322–2330.
Narayanan, M., A. Vetta, E. E. Schadt and J. Zhu (2010): "Simultaneous clustering of multiple gene expression and physical interaction datasets," PLoS Comput Biol., 6, e1000742.
Neal, R. M. (2000): "Markov chain sampling methods for dirichlet process mixture models," J. Comput. Graph. Stat., 9, 249–256.
Needham, M. and F. L. Mastaglia (2007): "Inclusion body myositis: current pathogenetic concepts and diagnostic and therapeutic approaches," Lancet Neurol., 6, 620–631.
Nepusz, T., H. Yu and A. Paccanaro (2012): "Detecting overlapping protein complexes in protein-protein interaction networks," Nat. Methods, 9, 471–472.
Rakha, E. A., J. S. Reis-Filho and I. O. Ellis (2008): "Basal-like breast cancer: a critical review," J. Clin. Oncol., 26, 2568–2581.
Rasmussen, C. and C. Williams (2006): Gaussian processes for machine learning, 55 Hayward Street, Cambridge, MA 02142: The MIT Press, first edition.
Reiss, D. J., N. S. Baliga and R. Bonneau (2006): "Integrated biclustering of heterogeneous genome-wide datasets for the inference of global regulatory networks," BMC Bioinformatics, 7, 280.
Savage, R. S., Z. Ghahramani, J. E. Griffin, P. Kirk and D. L. Wild (2013): "Identifying cancer subtypes in glioblastoma by combining genomic, transcriptomic and epigenomic data," arXiv preprint arXiv:1304.3577.
Schimek, M. G., E. Budinská, K. G. Kugler, V. Švendová, J. Ding and S. Lin (2015): "TopKLists: a comprehensive R package for statistical inference, stochastic aggregation, and visualization of multiple omics ranked lists," Stat. Appl. Genet. Mol. Biol., 14, 311–316.
Shen, R., A. B. Olshen and M. Ladanyi (2009): "Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis," Bioinformatics, 25, 2906–2912.
Stark, C., B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz and M. Tyers (2006): "BioGRID: a general repository for interaction datasets," Nucleic Acids Res., 34(suppl 1), D535–D539.
Thorne, T., P. Fratta, M. G. Hanna, A. Cortese, V. Plagnol, E. M. Fisher and M. P. H. Stumpf (2013): "Graphical modelling of molecular networks underlying sporadic inclusion body myositis," Mol. BioSyst., 9, 1736–1742.
Troyanskaya, O. G., K. Dolinski, A. B. Owen, R. B. Altman and D. Botstein (2003): "A bayesian framework for combining heterogeneous data sources for gene function prediction (in saccharomyces cerevisiae)," Proc. Natl. Acad. Sci. USA, 100, 8348–8353.
Wang, B., A. M. Mezlini, F. Demir, M. Fiume, Z. Tu, M. Brudno, B. Haibe-Kains and A. Goldenberg (2014): "Similarity network fusion for aggregating data types on a genomic scale," Nat. Methods, 11, 333–337.
Wu, M., X. Li, C.-K. Kwoh and S.-K. Ng (2009): "A core-attachment based method to detect protein complexes in ppi networks," BMC Bioinformatics, 10, 169.
Yuan, Y., R. S. Savage and F. Markowetz (2011): "Patient-specific data fusion defines prognostic cancer subtypes," PLoS Comput. Biol., 7, e1002227.
Zhang, X.-x., Q.-h. Xiao, B. Li, S. Hu, H.-j. Xiong and B.-h. Zhao (2015): "Overlap maximum matching ratio (ommr): a new measure to evaluate overlaps of essential modules," Frontiers of Information Technology & Electronic Engineering, 16, 293–300.

Supplemental Material: The online version of this article (DOI: 10.1515/sagmb-2016-0016) offers supplementary material, available to authorized users.

Brought to you by | Imperial College London
Authenticated
Download Date | 6/20/16 2:22 PM