Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

OXFORD

Subject Section

Consensus clustering for Bayesian mixture models

Stephen Coleman [1]*, Paul DW Kirk [1, 2†] and Chris Wallace [1,2†]

[1]MRC Biostatistics Unit, University of Cambridge, Cambridge, CB2 0SR, United Kingdom and
[2]Department of Medicine, University of Cambridge, Cambridge, CB2 0AW, United Kingdom.

*To whom correspondence should be addressed.
† These authors provided an equal contribution.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Cluster analysis is an integral part of precision medicine and systems biology, used to define groups of patients or biomolecules, as well as more general analyses and data exploration. However, problems such as choosing the number of clusters and issues with high dimensional data arise consistently. An ensemble approach, such as consensus clustering, can overcome some of the difficulties associated with high dimensional data, frequently exploring more relevant clustering solutions than individual models. Another tool for cluster analysis, Bayesian mixture modelling, has alternative advantages, including the ability to infer the number of clusters present and extensibility. However, inference of these models is often performed using Markov-chain Monte Carlo (MCMC) methods which can suffer from problems, such as poor exploration of the posterior distribution and long runtimes, when applied to high dimensional data. This makes applying Bayesian mixture models and their extensions to 'omics data challenging. We apply the clustering ensemble method, consensus clustering, to Bayesian mixture models to address these problems from clustering complex data.
Results: Consensus clustering of Bayesian mixture models successfully finds generating structure in our simulation study. This approach successfully captures multiple modes in the likelihood surface and offers significant reductions in runtime when a parallel environment is available compared to use of a single long chain for traditional Bayesian inference. We propose a heuristic to decide upon the ensemble size and then apply consensus clustering to Multiple Dataset Integration, an extension of Bayesian mixture models for integrative analyses, on three 'omics datasets for the cell-cycle of budding yeast. We find clusters of genes that are co-expressed and have common regulatory proteins which we validate using external knowledge. Consensus clustering can be any MCMC-based clustering method and offers improvements in runtime in a parallel environment and better exploration of clustering space.
Contact: stephen.coleman@mrc-bsu.cam.ac.uk
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

From defining a taxonomy of disease to creating molecular sets, grouping items can help us to understand and make decisions using complex biological data. For example, grouping patients based upon disease characteristics and personal omics data may allow the identification of more homogeneous subgroups, enabling stratified medicine approaches. Defining and studying molecular sets can improve our understanding of biological systems as these sets are more interpretable than their constituent members (Hejblum *et al.*, 2015), and study of their interactions and perturbations may have ramifications for diagnosis and drug targets (Bai *et al.*, 2013; Emmert-Streib *et al.*, 2014).

The act of identifying such groups is referred to as "cluster analysis". Traditional methods such as K-means clustering (Lloyd, 1982; Forgy, 1965) or hierarchical clustering condition upon a user inputted choice of K, the number of occupied clusters present. These methods are often heuristic in nature, relying on rules of thumb to decide upon a final model choice. For example, different choices of K are compared under some metric such as silhouette or the within-cluster sum of squared errors (**SSE**) as a function of K. For K-means clustering, its sensitivity to initialisation

means multiple runs are often used in practice, with that which minimises SSE used (Arthur and Vassilvitskii, 2006). This problem arises as the algorithm has no guarantees on finding the global minimum of SSE.

In many analyses or decision-making processes, quantifying confidence in the clustering can be of interest. Returning to the stratified medicine example of clustering patients, there might be individuals with almost equal probability of being allocated between several clusters which might influence decisions made. However, many clustering algorithms provide only a point clustering.

Ensemble methods offer a solution to the problems of sensitivity to initialisation and the lack of measure of uncertainty. These approaches have had great success in supervised learning, most famously in the form of Random Forest (Breiman, 2001) and boosting (Friedman, 2002). In clustering, consensus clustering (Monti *et al.*, 2003) is a popular method which has been implemented in R (Wilkerson *et al.*, 2010; Gu *et al.*, 2020) and been applied to problems such as cancer subtyping (Lehmann *et al.*, 2011; Verhaak *et al.*, 2010) and identifying subclones in single cell analysis (Kiselev *et al.*, 2017). Consensus clustering uses W runs of some base model or learner (such as K-means clustering) and compiles the W proposed partitions into a *consensus matrix*, the $(i, j)^{th}$ entries of which contain the proportion of model runs for which the i^{th} and j^{th} individuals co-cluster (for this and other definitions see section 1 of the Supplementary Material). This proportion represents some measure of confidence in the co-clustering of any pair of items. Furthermore, ensembles can offer reductions in computational runtime. This is as the individual learners can be weaker (and thus use either less of the available data or stop before full convergence) and because the learners in most ensemble methods are independent of each other and thus enable use of a parallel environment for each of the quicker model runs (Ghaemi *et al.*, 2009).

Monti *et al.* (2003) proposed some methods for choosing K using the consensus matrix, but this remains a problem in the methods mentioned so far. An alternative clustering framework, model-based clustering or *mixture models*, embeds the cluster analysis within a formal, statistical framework (Fraley and Raftery, 2002). This means that models can be compared formally, and problems such as the choice of K can be addressed as a model selection problem with all the associated tools. Mixture models are also attractive, as they have great flexibility in the type of data they can be applied to due to different choice of densities.

Furthermore, *Bayesian mixture models* can treat K as a random variable that is inferred from the data and thus the final clustering is not conditional upon a user chosen value, but K is jointly modelled along with the clustering. Such inference can be performed through use of a Dirichlet Process (Ferguson, 1973), a mixture of finite mixture models (Richardson and Green, 1997; Miller and Harrison, 2018) or an over-fitted mixture model (Rousseau and Mengersen, 2011; Van Havre *et al.*, 2015). These models and their extensions have a history of successful application to a diverse range of biological problems such as finding clusters of gene expression profiles (Medvedovic and Sivaganesan, 2002), cell types in flow cytometry (Chan *et al.*, 2008; Hejblum *et al.*, 2019) or scRNAseq experiments (Prabhakaran *et al.*, 2016), and estimating protein localisation (Crook *et al.*, 2018). Bayesian mixture models can be extended to jointly model the clustering across multiple datasets (Kirk *et al.*, 2012; Gabasova *et al.*, 2017). More details of Bayesian mixture models and an example of an extension to an integrative setting can be found in section 2 of the Supplementary Material.

Inference may be performed upon Bayesian mixture models using variational inference (**VI**, Blei *et al.*, 2006). While VI is powerful, "it is not yet well understood" (Blei *et al.*, 2017). Furthermore, VI can struggle with multi-modality, underestimates the variance in the posterior distribution (Turner and Sahani, 2011) and it has been shown to have a very computationally heavy initialisation cost to have good results (Wang and Dunson, 2011). Implementation is difficult, requiring either complex derivations (see the Appendix Supplementary Methods of Argelaguet *et al.*, 2018, for an example) or black-box, approximate solutions (Kucukelbir *et al.*, 2017).

However, Markov chain Monte Carlo (MCMC) methods are the most popularly method for Bayesian inference. In Bayesian clustering methods, MCMC methods are used to construct a chain of clusterings, and then an assessment of the convergence of this chain is made, to determine if its behaviour aligns with the expected asymptotic theory. This chain of samples will converge to the posterior distribution of the Bayesian model and explore its entire support given an infinite runtime. However, in practice problems arise. Individual chains often fail to explore the full support of the posterior distribution (an example of different chains becoming trapped in a single mode of the likelihood surface can be seen in the Supplementary Materials of Strauss *et al.*, 2020) and experience slow runtimes. Some MCMC methods make efforts to overcome the problem of exploration, often at the cost of increased computational cost per iteration. For a description of some of the problems and attempted solutions for MCMC methods, both generally and in clustering, see Robert *et al.* (2018); Yao *et al.* (2020); Chandra *et al.* (2020).

We propose that applying consensus clustering to Bayesian mixture models can overcome some of the issues endemic in high dimensional Bayesian clustering. Monti *et al.* (2003) suggest this application as part of their original paper, but no investigation has been attempted to our knowledge. This ensemble approach sidesteps the problems of convergence associated MCMC methods and offers computational gains through using shorter chains run in parallel. Furthermore, this approach could be directly used on any existing MCMC based implementation of Bayesian mixture models or their extensions and would avoid the re-implementation process that changing to newer MCMC methods or VI would entail.

We propose a heuristic for deciding upon the ensemble width (the number of learners used, W) and the ensemble depth (the number of iterations run within each chain, D), inspired by the use of scree plots in Principal Component Analysis (**PCA** Wold *et al.*, 1987).

We show via simulation that ensembles consisting of short chains can be sufficient to successfully recover generating structure. We also show that consensus clustering explores as many or more modes of the likelihood surface than either standard Bayesian inference or `Mclust`, a maximum likelihood method, all while offering improvements in runtime to traditional Bayesian inference.

We go on to perform an integrative analysis of cell cycle data from *Saccharomyces cerevisiae*. The cell cycle is the process by which a growing cell divides into two daughter cells. This involves virtually all cellular processes and diverse regulatory events (Granovskaia *et al.*, 2010). The cell cycle is crucial to biological growth, repair, reproduction, and development (Tyson *et al.*, 2013; Chen *et al.*, 2004; Alberts *et al.*, 2002). The regulatory proteins of the cell cycle are so highly conserved among eukaryotes that many of them function perfectly when transferred from a human cell to a yeast cell (Alberts *et al.*, 2002). This conservation means that a relatively simple eukaryote such as *Saccharomyces cerevisiae* can provide insight into a variety of cell cycle perturbations including those that occur in human cancer (Ingalls *et al.*, 2007; Chen *et al.*, 2004) and ageing (Jiménez *et al.*, 2015). Budding yeast is particularly attractive for genetic analysis as large numbers of cells may be synchronised in a particular stage of the cell cycle (Juanes, 2017). We apply consensus clustering to an extension of Bayesian mixture models, Multiple Dataset Integration (**MDI**), a multiple dataset clustering method. We determine the ensemble size using our proposed stopping rule. We use this ensemble to infer clusters of genes across datasets and validate these clusters using knowledge external to the analysis.

2 Material and methods

2.1 Consensus clustering for Bayesian mixture models

We apply consensus clustering to MCMC based Bayesian clustering models using the method described in algorithm 1. Our application of consensus clustering has two main parameters at the ensemble level, the chain depth, D, and ensemble width, W.

Data: $X = (x_1, \ldots, x_N)$
Input:
The number of chains to run, W
The number of iterations within each chain, D
A clustering method that uses MCMC methods to generate samples of clusterings of the data $Cluster(X, d)$
Output:
A predicted clustering, $\hat{Y}$
The consensus matrix $\mathbf{M}$
begin
  /* initialise an empty consensus matrix */
  $\mathbf{M} \leftarrow \mathbf{0}_{N \times N}$;
  for $w = 1$ **to** W **do**
    /* set the random seed controlling initialisation and MCMC moves */
    $set.seed(w)$;
    /* initialise a random partition on X drawn from the prior distribution */
    $Y_{(0,w)} \leftarrow Initialise(X)$;
    for $d = 1$ **to** D **do**
      /* generate a markov chain for the membership vector */
      $Y_{(d,w)} \leftarrow Cluster(X, d)$;
    end
    /* create a coclustering matrix from the D^{th} sample */
    $\mathbf{B}^{(w)} \leftarrow Y_{(D,w)}$;
    $\mathbf{M} \leftarrow \mathbf{M} + \mathbf{B}^{(w)}$;
  end
  $\mathbf{M} \leftarrow \frac{1}{W}\mathbf{M}$;
  $\hat{Y} \leftarrow$ partition X based upon $\mathbf{M}$;
end

Algorithm 1: Consensus clustering for Bayesian mixture models.

We use the `maxpear` function (Fritsch *et al.*, 2009) from the R package `mcclust` (Fritsch, 2012) to infer a point clustering from the consensus matrix (some details of which are given in section 3 of the Supplementary Material).

2.1.1 Determining the ensemble depth and width

As our ensemble sidesteps the problem of convergence within each chain, we need an alternative stopping rule for growing the ensemble in chain depth, D, and number of chains, W. We propose a heuristic based upon the consensus matrix to decide if a given value of D and W are sufficient. We suspect that increasing W and D might continuously improve the performance of the ensemble, but we observe in our simulations that these improvements will become smaller and smaller for greater values, approaching some asymptote for each of W and D. We notice that this behaviour is analogous to PCA in that where for consensus clustering some improvement might always be expected for increasing chain depth or ensemble width, more variance will always be captured by increasing the number of components used in PCA. However, increasing this number beyond some threshold has diminishing returns, diagnosed in PCA by a scree plot. Following from this, we recommend, for some set of ensemble parameters, $D' = \{d_1, \ldots, d_I\}$ and $W' = \{w_1, \ldots, w_J\}$, find the mean absolute difference of the consensus matrix for the d_i^{th} iteration from w_j chains to that for the $d_{(i-1)}^{th}$ iteration from w_j chains and plot these values as a function of chain depth, and the analogue for sequential consensus matrices for increasing ensemble width and constant depth.

If this heuristic is used, we believe that the consensus matrix and the resulting inference should be stable, providing a robust estimate of the clustering. In contrast, if there is still strong variation in the consensus matrix for varying chain length or number, then we believe that the inferred clustering is influenced significantly by the random initialisation. This means that the inferred partition that it is unlikely to be stable for similar datasets or reproducible for a random choice of seeds. This stability is often a desirous property in a clustering method (Von Luxburg and Ben-David, 2005; Meinshausen and Bühlmann, 2010).

2.2 Simulation study

We use a finite mixture with independent features as the data generating model within the simulation study. Within this model there exist "irrelevant features" (Law *et al.*, 2003) that have global parameters rather than cluster specific parameters. The generating model is

$$p(X, c, \theta, \pi | K) = p(K)p(\pi|K)p(\theta|K) \prod_{i=1}^{N} p(c_i|\pi, K) \times \prod_{p=1}^{P} p(x_{ip}|c_i, \theta_{c_i p})^{\phi_p} p(x_{ip}|\theta_p)^{(1-\phi_p)}$$

for data $X = (x_1, \ldots, x_N)$, cluster label or allocation variable $c = (c_1, \ldots, c_N)$, cluster weight $\pi = (\pi_1, \ldots, \pi_K)$, K clusters and the relevance variable, $\phi \in \{0, 1\}$ with $\phi_p = 1$ indicating that the p^{th} feature is relevant to the clustering. We used a *Gaussian* density, so $\theta_{kp} = (\mu_{kp}, \sigma_{kp}^2)$. We defined three scenarios to simulate data within. Within each scenario 100 datasets are generated. The parameters defining the scenarios are described in Table 1 and a single example for each shown in Figure 1. We generate two dimensional datasets (the *2D* scenario), datasets representing the *small N, large P* paradigm and datasets with a large number of irrelevant features (the *irrelevant features* scenario). Additional details of the simulation process and additional scenarios are included in section 4.1 of the Supplementary Materials.

Table 1. Parameters defining the simulation scenarios as used in generating data and labels. $\Delta\mu$ is the distance between neighbouring cluster means within a single feature. The number of relevant features (P_s) is $\sum_p \phi_p$, and $P_n = P - P_s$.

Scenario	N	P_s	P_n	K	$\Delta\mu$	σ^2	π
2D	100	2	0	5	3.0	1	$(\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5})$
Small N, large P	50	500	0	5	1.0	1	$(\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5})$
Irrelevant features	200	20	100	5	1.0	1	$(\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5})$

In each of these scenarios we apply a variety of methods (listd below) and compare the inferred point clusterings to the generating labels using the Adjusted Rand Index (**ARI** Hubert and Arabie, 1985).

- `Mclust`, a maximum likelihood implementation of finite mixture models (for a range of modelled clusters, K),
- 10 chains of 1 million iterations, thinning to every thousandth sample for the overfitted Bayesian mixture model, and

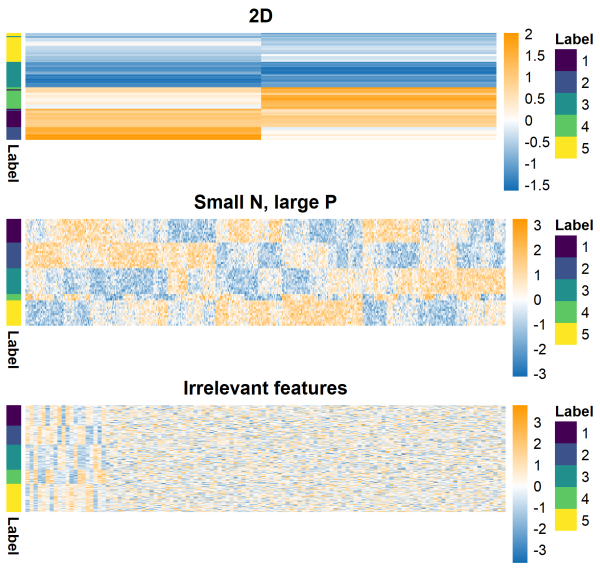

Fig. 1. Example of generated datasets. Each row is an item being clustered and each column a feature of generated data. The 2D dataset (which is ordered by hierarchical clustering here) should enable proper mixing of chains in the MCMC. The small N, large P case has clear structure (observable by eye). This is intended to highlight the problems of poor mixing due to high dimensions even when the generating labels are quite identifiable. In the irrelevant features case, the structure is clear in the relevant features (on the left-hand side of this heatmap). This setting is intended to test how sensitive each approach is to noise.

- a variety of consensus clustering ensembles defined by inputs of W chains and D iterations within each chain (see algorithm 1) with $W \in \{1, 10, 30, 50, 100\}$ and $D \in \{1, 10, 100, 1000, 10000\}$.

The ARI is a measure of similarity between two partitions,c_1, c_2, corrected for chance, with 0 indicating c_1 is no more similar to c_2 than a random partition would be expected to be and a value of 1 showing that c_1 and c_2 perfectly align. Details of how we used the methods in the simulation study can be found in sections 4.2, 4.3 and 4.4 of the Supplementary Material.

2.2.1 `Mclust`

`Mclust` (Scrucca *et al.*, 2016) is a function from the R package `mclust`. It estimates and compares Gaussian mixture models based upon the maximum likelihood estimator of the parameters for a range of choices of K, the number of clusters used, and different covariance structures. The method initialises upon a hierarchical clustering of the data cut to K clusters. The "best" model is determined using the Bayesian information criterion, (Schwarz *et al.*, 1978). See section 4.2 of the Supplementary Material for more details on how we applied `Mclust` to the simulated data.

2.2.2 Bayesian inference

To assess within-chain convergence of our Bayesian inference we use the Geweke Z-score statistic (Geweke *et al.*, 1991). Of the chains that appear to behave properly we then asses across-chain convergence using $\hat{R}$ (Gelman *et al.*, 1992) and the recent extension provided by Vats and Knudson (2018).

If a chain has reached its stationary distribution the Geweke Z-score statistic is expected to be normally distributed. Normality is tested for using a Shapiro-Wilks test (Shapiro and Wilk, 1965). If a chain fails this test (i.e., the associated p-value is less than 0.05), we assume that it has not achieved stationarity and it is excluded from the remainder of the analysis.

The Vats and Knudson extension of $\hat{R}$ is a summary statistic for the entire chain; this is the primary indicator of failure for convergence, but a visualisation of the original $\hat{R}$ diagnostic is also considered.

In theory we would expect the Bayesian chains to explore a common posterior distribution, but the practice sees chains become trapped in distinct modes in different scenarios. We believe that the mode within which the greatest number of chains become trapped would be that which is used to perform the inference in a real application (and longer chains did not solve the problem). As part of a pipeline where such analyses have to happen $12 \times 100 = 1,200$ times, we pool the Bayesian samples into a single Posterior Similarity Matrix (**PSM**, defined in section 1 of the Supplementary Material) to avoid model selection problems. If the chains all explore the same space pooling the samples has little effect on the inference. Based upon visual inspection of the PSMs indicates that the disagreement between modes in our simulations tends to be one of

- several clusters are merged or
- a generating cluster is represented by two or more components in the model,

each of the modes tends to have large overlap with all others. This means that the clustering inferred from the PSM created from the samples pooled across all stationary chains will represent the most popular mode and the metric of performance we use, the ARI will not be unduly inflated.

To infer a final clustering from the PSM, we use the `maxpear` function, as we do in the consensus clustering.

Details of the convergence diagnoses and `maxpear` in a Bayesian setting are given in section 4.3 of the Supplementary Material.

2.3 Analysis of the cell cycle in budding yeast

2.3.1 Datasets

We aim to create clusters of genes that are co-expressed, have common regulatory proteins and share a biological function. To achieve this, we use three datasets that were generated using different 'omics technologies and target different aspects of the molecular biology underpinning the cell cycle process.

- Microarray profiles of RNA expression from Granovskaia *et al.* (2010). This dataset comprises measurements of cell-cycle-regulated expression at 5-minute intervals for 200 minutes (up to three cell division cycles) and is referred to as the **time course** dataset. The cells are synchronised at the START checkpoint in late G1-phase using alpha factor arrest (Granovskaia *et al.*, 2010). We include only the genes identified by Granovskaia *et al.* (2010) as having periodic expression profiles.
- Chromatin immunoprecipitation followed by microarray hybridization (**ChIP-chip**) data from Harbison *et al.* (2004). This dataset discretizes p-values from tests of association between 117 DNA-binding transcriptional regulators and a set of yeast genes. Based upon a significance threshold these p-values are represented as either a 0 (no interaction) or a 1 (an interaction).
- Protein-protein interaction (**PPI**) data from BioGrid (Stark *et al.*, 2006). This database consists of of physical and genetic interactions between gene and gene products. The interactions included are a collection of results observed in high throughput experiments and some computationally inferred interactions. The dataset we used contained 603 proteins as columns. An entry of 1 in the $(i, j)^{th}$ cell indicates that the i^{th} gene has a protein product that is believed to interact with the j^{th} protein.

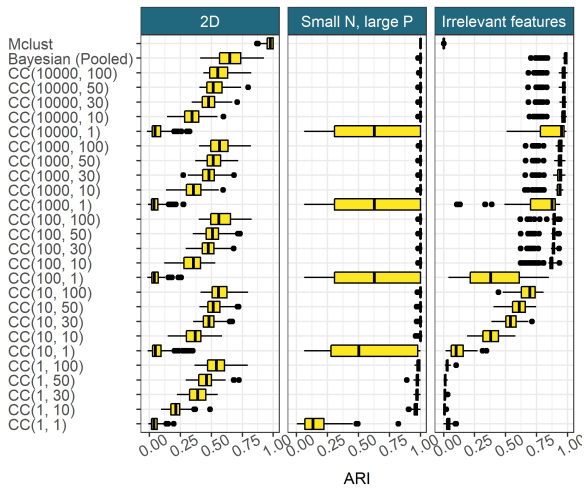

Fig. 2. Model performance in the 100 simulated datasets for each scenario, defined as the ARI between the generating labels and the inferred clustering. $CC(d, w)$ denotes consensus clustering using the clustering from the d^{th} iteration from w different chains.

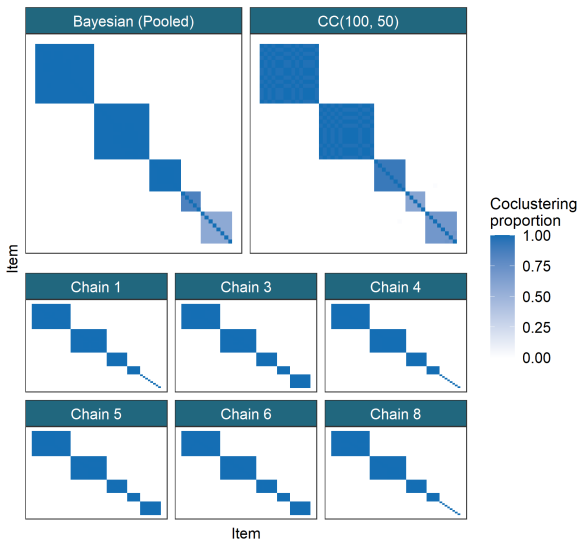

Fig. 3. Comparison of similarity matrices from a dataset for the Small N, large P scenario. In each matrix, the $(i, j)^{th}$ entry is the proportion of clusterings for which the i^{th} and j^{th} items co-clustered for the method in question. In the first row the PSM of the pooled Bayesian samples is compared to the CM for CC(100, 50), with a common ordering of rows and columns in both heatmaps. In the following rows, 6 of the long chains that passed the tests of convergence are shown.

The datasets were reduced to 551 items by considering only the genes with no missing data in the PPI and ChIP-chip data. The choices to reduce the datasets to these 551 genes are the same steps as in Kirk *et al.* (2012).

2.3.2 Multiple dataset integration

We applied consensus clustering to MDI for our integrative analysis. MDI jointly models the clustering in each dataset, inferring individual clusterings for each dataset that are informed by similarity in the other clusterings. MDI learns the similarity between the datasets being analysed and does not assume global structure. This means that the similarity between datasets is not strongly assumed in our model; individual clusters or genes that align across datasets are based solely upon the evidence present in the data not strong modelling assumptions. Thus, datasets that share less common information can be included without fearing that this will warp the final clusterings in some way. For more details on MDI see section 2.2 of the Supplementary Material and Kirk *et al.* (2012).

The datasets were modelled using a mixture of Gaussian processes in the time course dataset and Multinomial distributions in the ChIP-chip and PPI datasets.

3 Results

3.1 Simulated data

We use the ARI between the generating labels and the inferred clustering of each method to be our metric of predictive performance. A summary of the results for a selection of the simulation scenarios are shown in Figure 2. In the 2D and Small N, large P scenarios, `Mclust` performs very well, correctly identifying the true structure However, a large number of irrelevant features completely erodes the ability of `Mclust` to uncover subpopulation structure. Instead, the method is blinded by the irrelevant features and identifies a clustering of $K = 1$.

The pooled samples from multiple long chains performs very well. This suggests that the pooling effect means that any multi-modality present in the data is less of a problem. In this case the pooled samples, themselves a consensus of 10 models, acts as an upper bound on the more practical implementations of consensus clustering and any individual long chain.

Consensus clustering does uncover some of the generating structure in the data, even using a small number of short chains. With sufficiently large ensembles and chain depth, consensus clustering is close to the pooled Bayesian samples in predictive performance.

For the consensus clustering methods, we can also see from Figure 2, that for a constant chain depth increasing the number of chains used follows a pattern of diminishing returns. There are strong initial gains for a greater ensemble width, but the improvement decreases for each successive chain. A similar pattern emerges in increasing chain length for a constant number of chains.

If we look beyond the ARI at the PSMs and consensus matrix for one of the ensembles in Figure 3, we see very little difference between the similarity matrix from the pooled samples and the consensus clustering. Similar clusters emerge, and we see comparable confidence in the pairwise clusterings. In the next two rows the PSMs constructed from 6 of the long chains that passed the stationarity test are displayed. Each PSM is binary, with all entries being 0 or 1. This means only a single clustering is sampled within each chain, implying very little uncertainty in the partition. However, three different modes emerge across the chains showing that the chains are failing to explore the full support of the posterior distribution of the clustering. This shows that consensus clustering is exploring more possible clusterings than any individual chain and explores a similar space to the union of the samples from the long chains.

Figure 3 shows an example of different long chains becoming trapped in different modes and failing to explore a common partition space. We see that if any one chain is used then some of the uncertainty that should be present, some part of the target distribution, is discarded as each chain only represents a single possible clustering. Interestingly, if we take the union of the long chains and construct a similarity matrix, which might be considered to have sampled from various parts of the posterior distribution, this is more similar to the consensus matrix for a number of short chains than it is to any single long chain (as seen in the similarity of the PSM for the Bayesian (Pooled) and CC(100, 50) in Figure 3), suggesting that the

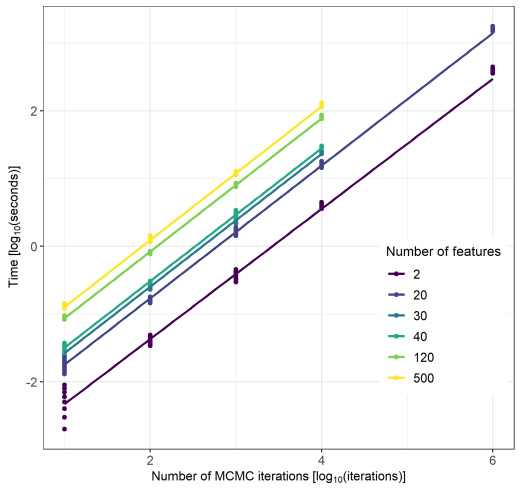

Fig. 4. The time taken for different numbers of iterations of MCMC moves in $\log_{10}(seconds)$. The relationship between chain length, D, and the time taken is linear (the slope is approximately 1 on the $\log_{10}$ scale), with a change of intercept for different dimensions. The runtime of each Markov chain was recorded using the terminal command `time`, measured in milliseconds.

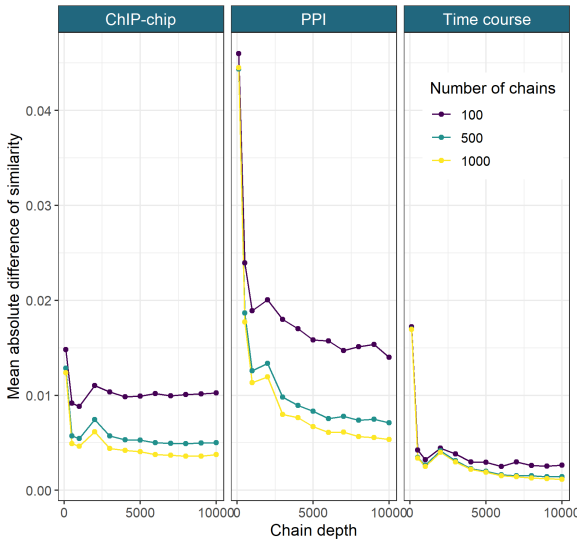

Fig. 5. The mean absolute difference between the sequential Consensus matrices. For a set of chain lengths, $D' = \{d_1, \ldots, d_I\}$ and number of chains, $W' = \{w_1, \ldots, w_J\}$, we take the mean of the absolute difference between the consensus matrix for (d_i, w_j) and (d_{i-1}, w_j) (here $D' = \{101, 501, 1001, 2001, \ldots, 10001\}$ and $W' = \{100, 500, 1000\}$).

consensus matrix is describing the uncertainty present in the final clustering more fully than any single long chain.

Figure 4 shows that chain length is directly proportional to the time taken for the chain to run. This means that using an ensemble of shorter chains, as in consensus clustering, can offer large reductions in the time cost of analysis when a parallel environment is available compared to standard Bayesian inference. Even on a laptop of 8 cores running an ensemble of 1,000 chains of length 1,000 will require approximately half as much time as running 10 chains of length 100,000 due to parallelisation, and the potential benefits are far greater when using a large computing cluster.

Additional details of the results for the simulated data can be found in section 4.4 of the Supplementary Material, with an additional 9 scenarios also included.

3.2 Multi-omics analysis of the cell cycle in budding yeast

We use the stopping rule proposed in 2.1.1 to determine our ensemble depth and width. In Figure 5, we see that the change in the consensus matrices from increasing the ensemble depth and width is diminishing in keeping with results in the simulations. We see no strong improvement after $D = 6,000$ and increasing the number of learners from 500 to 1,000 has small effect. We therefore use the largest ensemble available, an ensemble of depth $D = 10001$ and width $W = 1000$ with a base learner of MDI, with the belief that it is achieving the asymptotic performance described in the simulation results. Some additional evidence for this choice is given in section 5.1 of the Supplementary Material.

We focus upon the genes that tend to have the same cluster label in both the time course and ChIP-chip datasets as being of the most interest for an integrative analysis. 261 genes (nearly half of the genes present) in this pair of datasets have a common label in most chains, whereas only 56 genes have a common label across all three datasets. Thus, we focus upon this pairing of datasets as they appear to share much common signal. More formally, we analyse the clustering structure among genes for which $\hat{P}(c_{nl} = c_{nm}) > 0.5$, where c_{nl} denotes the cluster label of gene n in dataset l. We show the gene expression and regulatory proteins of these genes separated by their cluster in Figure 6. In this plot we include only the clusters with more than one member and more than half the members having some interacitons in the ChIP-chip data. We find that a small number of transcription factors dominate, with different combinations emerging across the 10 clusters (these and some relevant details are listed in Table 2 of the Supplementary Material). Many of these 10 correspond to transcription factors that are well known to regulate cell cycle expression (Simon *et al.*, 2001).

As an example, we briefly analyse clusters 9 and 16 in greater depth. Cluster 9 has strong association with MBP1 and some interactions with SWI6, as can be seen in Figure 6. The Mbp1-Swi6p complex, MBF, is associated with DNA replication (Iyer *et al.*, 2001). The first time point, 0 minutes, in the time course data is at the START checkpoint, or the G1/S transition. The members of cluster 9 begin highly expressed at this point before quickly dropping in expression (in the first of the 3 cell cycles). This suggests that many transcripts are produced immediately in advance of S-phase, and thus are required for the first stages of DNA synthesis. We used the `org.Sc.sgd.db` (Carlson *et al.*, 2014) package to find gene descriptions (which are included in Table 3 of the Supplemenatry Material). These descriptions support this hypothesis, as we find that many of the genes in cluster 9 are associated with DNA replication, repair and/or recombination. Members of the replication checkpoint, TOF1, MRC1 and RAD53 (Bando *et al.*, 2009; Lao *et al.*, 2018) also emerge in the cluster as do some members of the cohesin complex. Cohesin is associated with sister chromatid cohesion, which is established during the DNA synthesis phase of the cell cycle (Tóth *et al.*, 1999), but also contributes to transcription regulation, DNA repair, chromosome condensation, homolog pairing (Mehta *et al.*, 2013), fitting the theme of cluster 9.

Cluster 16 consists of genes whose products form the histones H1, H2A, H2B, H3 and H4 and then has three other members, GAS3, NRM1 and PDS1. Histones are the chief protein components of chromatin (Fischle *et al.*, 2003) and are important contributors to gene regulation (Bannister and Kouzarides, 2011). They are known to peak in expression in S-phase (Granovskaia *et al.*, 2010), which matches the first peak of this cluster early in the time series. Of the other members, NRM1 is a transcriptional co-repressor of MBF-regulated gene expression acting at the transition from G1 to S-phase (de Bruin *et al.*, 2006; Aligianni *et al.*, 2009). Pds1p

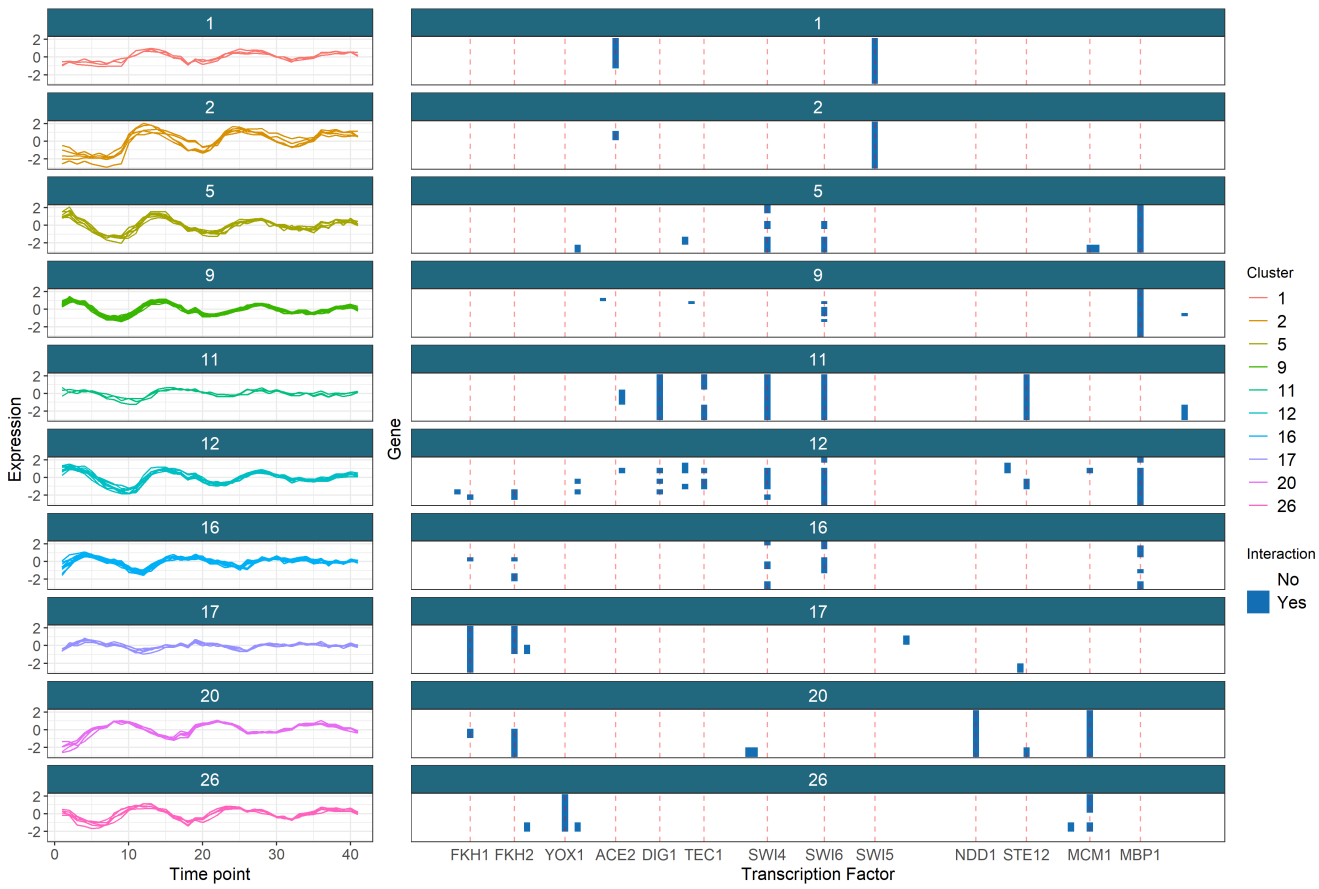

Fig. 6. The genes which tend to have a common label across the time course and ChIP-chip datasets separated out by their clusters. We exclude the clusters with no interactions in the ChIP-chip dataset and include a red line for the transcription factors that dominate the clustering structure in the ChIP-chip dataset. The clusters in the time series data are quite well banded and distinct (having different periods, amplitudes, or both) and in the ChIP-chip data a small number of Transcription factors dominate the clustering structure. The clusters have tight, unique signatures in the time course dataset and tend to be defined by a small number of well-studied transcription factors in the ChIP-chip dataset.

binds to and inhibits the Esp1 class of sister separating proteins, preventing sister chromatids separation before M phase (Ciosk *et al.*, 1998; Tóth *et al.*, 1999). The remaining member, GAS3, is poorly understood. It interacts with SMT3 which regulates chromatid cohesion, chromosome segregation and DNA replication (among other things). Chromatid cohesion ensures the faithful segregation of chromosomes in mitosis and in both meiotic divisions (Cooper *et al.*, 2009) and is instantiated in S-phase (Tóth *et al.*, 1999). These results, along with the very similar expression profile to the histone genes in the time course data, suggest that GAS3 may be involved more directly in DNA replication or chromatid cohesion than is currently believed.

We attempt to perform a similar analysis using traditional Bayesian inference of MDI, but after 36 hours of runtime there is no consistency or convergence across chains. Each chain provides different parameter distributions and clustering estimates, leaving no clear clustering solution. For details of this analysis, see section 5.2 of the Supplementary Material.

4 Discussion

Our proposed method has demonstrated good performance on simulation studies, uncovering generating structure and approximating Bayesian inference when the Markov chain is exploring the full support of the posterior. However, we have shown that if a finite Markov chain fails to describe the full posterior and is itself only approximating Bayesian inference, our method has better ability to represent several modes in the data than individual chains and thus offers a more consistent and reproducible analysis. Furthermore, consensus clustering is significantly faster in a parallel environment than inference using individual chains, while retaining the ability to robustly infer K, the number of occupied components present.

Based upon the simulations we proposed a method of assessing ensemble stability and deciding upon ensemble size. We then performed an integrative analysis of yeast cell cycle data using MDI, an extension of Bayesian mixture models that jointly models multiple datasets. We sidestep convergence issues and decrease the computational cost of such a large model by applying consensus clustering to it, showing the flexibility of consensus clustering and its applicability to any clustering method that uses MCMC methods to sample partitions. We used our proposed stopping rule to decide upon our ensemble size and uncovered many genes with shared signal across several datasets. We then explored the biological meaning of some of the uncovered clusters, using data external to the analysis, finding sensible results as well as signal for possibly novel biology.

The results of our simulations and the multi-omics analysis show that consensus clustering can be used in a broad context, being applicable to any MCMC based clustering method, not solely to mixture models. It offers computational gains and greater applicability to these methods as a result

and, attractively, it can be applied to existing implementations, unlike improvements to the underlying MCMC methods or alternative methods for Bayesian inference such as VI which would require re-writing software. However, consensus clustering does lose the theoretical framework of true Bayesian inference. We attempt to mitigate this with our assessment of stability in the ensemble, but this diagnosis is heuristic and subjective, and while there is empirical evidence for its success, it lacks the formal results for the tests of model convergence for Bayesian inference.

We expect that researchers interested in applying some of the Bayesian integrative clustering models such as MDI and Clusternomics (Gabasova *et al.*, 2017) will be enabled to do so, as consensus clustering overcomes some of the unwieldiness of these large, joint models. More generally, we expect that our method will be useful to researchers performing cluster analysis of high-dimensional data where the runtime of MCMC methods becomes too onerous and multi-modality is more likely to be present.

Funding

SC is supported by an UKRI Studentship via the MRC https://mrc.ukri.org/ (MC UU 00002/4). PK is supported by the MRC (MC UU 00002/13). CW is supported by the Wellcome Trust https://wellcome.ac.uk/ (WT107881) and the MRC (MC UU 00002/4). This research was funded in whole, or in part, by the Wellcome Trust [Grant number WT107881]. For the purpose of open access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

References

Alberts, B. *et al.* (2002). The cell cycle and programmed cell death. *Molecular biology of the cell*, **4**, 983–1027.

Aligianni, S. *et al.* (2009). The fission yeast homeodomain protein Yox1p binds to MBF and confines MBF-dependent cell-cycle transcription to G1-S via negative feedback. *PLoS Genet*, **5**(8), e1000626.

Argelaguet, R. *et al.* (2018). Multi-omics factor analysis—a framework for unsupervised integration of multi-omics data sets. *Molecular systems biology*, **14**(6), e8124.

Arthur, D. and Vassilvitskii, S. (2006). k-means++: The advantages of careful seeding. Technical report, Stanford.

Bai, J. P. *et al.* (2013). Strategic applications of gene expression: from drug discovery/development to bedside. *The AAPS journal*, **15**(2), 427–437.

Bando, M. *et al.* (2009). Csm3, Tof1, and Mrc1 form a heterotrimeric mediator complex that associates with DNA replication forks. *Journal of Biological Chemistry*, **284**(49), 34355–34365.

Bannister, A. J. and Kouzarides, T. (2011). Regulation of chromatin by histone modifications. *Cell research*, **21**(3), 381–395.

Blei, D. M. *et al.* (2006). Variational inference for Dirichlet process mixtures. *Bayesian analysis*, **1**(1), 121–143.

Blei, D. M. *et al.* (2017). Variational inference: A review for statisticians. *Journal of the American statistical Association*, **112**(518), 859–877.

Breiman, L. (2001). Random forests. *Machine learning*, **45**(1), 5–32.

Carlson, M. *et al.* (2014). Org. sc. sgd. db: Genome wide annotation for yeast. *R package version*, **2**(1).

Chan, C. *et al.* (2008). Statistical mixture modeling for cell subtype identification in flow cytometry. *Cytometry Part A: The Journal of the International Society for Analytical Cytology*, **73**(8), 693–701.

Chandra, N. K. *et al.* (2020). Bayesian clustering of high-dimensional data. *arXiv preprint arXiv:2006.02700*.

Chen, K. C. *et al.* (2004). Integrative analysis of cell cycle control in budding yeast. *Molecular biology of the cell*, **15**(8), 3841–3862.

Ciosk, R. *et al.* (1998). An esp1/pds1 complex regulates loss of sister chromatid cohesion at the metaphase to anaphase transition in yeast. *Cell*, **93**(6), 1067–1076.

Cooper, K. F. *et al.* (2009). Pds1p is required for meiotic recombination and prophase I progression in Saccharomyces cerevisiae. *Genetics*, **181**(1), 65–79.

Crook, O. M. *et al.* (2018). A Bayesian mixture modelling approach for spatial proteomics. *PLoS computational biology*, **14**(11), e1006516.

de Bruin, R. A. *et al.* (2006). Constraining g1-specific transcription to late g1 phase: the mbf-associated corepressor nrm1 acts via negative feedback. *Molecular cell*, **23**(4), 483–496.

Emmert-Streib, F. *et al.* (2014). Gene regulatory networks and their applications: understanding biological and medical problems in terms of networks. *Frontiers in cell and developmental biology*, **2**, 38.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The annals of statistics*, pages 209–230.

Fischle, W. *et al.* (2003). Histone and chromatin cross-talk. *Current opinion in cell biology*, **15**(2), 172–183.

Forgy, E. W. (1965). Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *biometrics*, **21**, 768–769.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, **97**(458), 611–631.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational statistics & data analysis*, **38**(4), 367–378.

Fritsch, A. (2012). *mcclust: process an MCMC sample of clusterings*. R package version 1.0.

Fritsch, A. *et al.* (2009). Improved criteria for clustering based on the posterior similarity matrix. *Bayesian analysis*, **4**(2), 367–391.

Gabasova, E. *et al.* (2017). Clusternomics: Integrative context-dependent clustering for heterogeneous datasets. *PLoS computational biology*, **13**(10), e1005781.

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

Geweke, J. *et al.* (1991). *Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments*, volume 196. Federal Reserve Bank of Minneapolis, Research Department Minneapolis, MN.

Ghaemi, R. *et al.* (2009). A survey: clustering ensembles techniques. *World Academy of Science, Engineering and Technology*, **50**, 636–645.

Granovskaia, M. V. *et al.* (2010). High-resolution transcription atlas of the mitotic cell cycle in budding yeast. *Genome biology*, **11**(3), 1–11.

Gu, Z. *et al.* (2020). cola: an R/Bioconductor package for consensus partitioning through a general framework. *Nucleic Acids Research*. gkaa1146.

Harbison, C. T. *et al.* (2004). Transcriptional regulatory code of a eukaryotic genome. *Nature*, **431**(7004), 99–104.

Hejblum, B. P. *et al.* (2015). Time-course gene set analysis for longitudinal gene expression data. *PLoS computational biology*, **11**(6), e1004310.

Hejblum, B. P. *et al.* (2019). Sequential Dirichlet process mixtures of multivariate skew t-distributions for model-based clustering of flow cytometry data. *The Annals of Applied Statistics*, **13**(1), 638–660.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of classification*, **2**(1), 193–218.

Ingalls, B. *et al.* (2007). Systems level modeling of the cell cycle using budding yeast. *Cancer informatics*, **3**, 117693510700300020.

Iyer, V. R. *et al.* (2001). Genomic binding sites of the yeast cell-cycle transcription factors SBF and MBF. *Nature*, **409**(6819), 533–538.

Jiménez, J. *et al.* (2015). Live fast, die soon: cell cycle progression and lifespan in yeast cells. *Microbial Cell*, **2**(3), 62.

Juanes, M. A. (2017). Methods of synchronization of yeast cells for the analysis of cell cycle progression. In *The Mitotic Exit Network*, pages 19–34. Springer.

Kirk, P. *et al.* (2012). Bayesian correlated clustering to integrate multiple datasets. *Bioinformatics*, **28**(24), 3290–3297.

Kiselev, V. Y. *et al.* (2017). SC3: consensus clustering of single-cell RNA-seq data. *Nature methods*, **14**(5), 483–486.

Kucukelbir, A. *et al.* (2017). Automatic differentiation variational inference. *The Journal of Machine Learning Research*, **18**(1), 430–474.

Lao, J. P. *et al.* (2018). The yeast DNA damage checkpoint kinase Rad53 targets the exoribonuclease, Xrn1. *G3: Genes, Genomes, Genetics*, **8**(12), 3931–3944.

Law, M. H. *et al.* (2003). Feature selection in mixture-based clustering. In *Advances in neural information processing systems*, pages 641–648.

Lehmann, B. D. *et al.* (2011). Identification of human triple-negative breast cancer subtypes and preclinical models for selection of targeted therapies. *The Journal of clinical investigation*, **121**(7), 2750–2767.

Lloyd, S. (1982). Least squares quantization in pcm. *IEEE transactions on information theory*, **28**(2), 129–137.

Medvedovic, M. and Sivaganesan, S. (2002). Bayesian infinite mixture model based clustering of gene expression profiles. *Bioinformatics*, **18**(9), 1194–1206.

Mehta, G. D. *et al.* (2013). Cohesin: functions beyond sister chromatid cohesion. *FEBS letters*, **587**(15), 2299–2312.

Meinshausen, N. and Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **72**(4), 417–473.

Miller, J. W. and Harrison, M. T. (2018). Mixture models with a prior on the number of components. *Journal of the American Statistical Association*, **113**(521), 340–356.

Monti, S. *et al.* (2003). Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Machine learning*, **52**(1-2), 91–118.

Prabhakaran, S. *et al.* (2016). Dirichlet process mixture model for correcting technical variation in single-cell gene expression data. In *International Conference on Machine Learning*, pages 1070–1079.

Richardson, S. and Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society: series B*, **59**(4), 731–792.

Robert, C. P. *et al.* (2018). Accelerating mcmc algorithms. *Wiley Interdisciplinary Reviews: Computational Statistics*, **10**(5), e1435.

Rousseau, J. and Mengersen, K. (2011). Asymptotic behaviour of the posterior distribution in overfitted mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **73**(5), 689–710.

Schwarz, G. *et al.* (1978). Estimating the dimension of a model. *The annals of statistics*, **6**(2), 461–464.

Scrucca, L. *et al.* (2016). mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, **8**(1), 289–317.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, **52**(3/4), 591–611.

Simon, I. *et al.* (2001). Serial regulation of transcriptional regulators in the yeast cell cycle. *Cell*, **106**(6), 697–708.

Stark, C. *et al.* (2006). Biogrid: a general repository for interaction datasets. *Nucleic acids research*, **34**(suppl_1), D535–D539.

Strauss, M. E. *et al.* (2020). Gpseudoclust: deconvolution of shared pseudo-profiles at single-cell resolution. *Bioinformatics*, **36**(5), 1484–1491.

Tóth, A. *et al.* (1999). Yeast cohesin complex requires a conserved protein, Eco1p (Ctf7), to establish cohesion between sister chromatids during DNA replication. *Genes & development*, **13**(3), 320–333.

Turner, R. E. and Sahani, M. (2011). Two problems with variational expectation maximisation for time-series models. In D. Barber, A. T. Cemgil, and S. Chiappa, editors, *Bayesian time series models*, chapter 5. Cambridge University Press, 1 edition.

Tyson, J. J. *et al.* (2013). Cell cycle, budding yeast. In W. Dubitzky, O. Wolkenhauer, K.-H. Cho, and H. Yokota, editors, *Encyclopedia of Systems Biology*, pages 337–341. Springer New York, New York, NY.

Van Havre, Z. *et al.* (2015). Overfitting Bayesian mixture models with an unknown number of components. *PloS one*, **10**(7), e0131739.

Vats, D. and Knudson, C. (2018). Revisiting the gelman-rubin diagnostic. *arXiv preprint arXiv:1812.09384*.

Verhaak, R. G. *et al.* (2010). Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in pdgfra, idh1, egfr, and nf1. *Cancer cell*, **17**(1), 98–110.

Von Luxburg, U. and Ben-David, S. (2005). Towards a statistical theory of clustering. In *Pascal workshop on statistics and optimization of clustering*, pages 20–26. Citeseer.

Wang, L. and Dunson, D. B. (2011). Fast Bayesian inference in Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, **20**(1), 196–216.

Wilkerson *et al.* (2010). ConsensusClusterPlus: a class discovery tool with confidence assessments and item tracking. *Bioinformatics*, **26**(12), 1572–1573.

Wold, S. *et al.* (1987). Principal component analysis. *Chemometrics and intelligent laboratory systems*, **2**(1-3), 37–52.

Yao, Y. *et al.* (2020). Stacking for non-mixing Bayesian computations: the curse and blessing of multimodal posteriors. *arXiv preprint arXiv:2006.12335*.