BIOINFORMATICS

Vol. 00 no. 00 2012
Pages 1–10

Bayesian Hierarchical Clustering with a Gaussian Conjugate Prior

Korsuk Sirinukunwattana [1], Richard Savage [2], Nasir M. Rajpoot [3,*]

[1]Centre for Complexity Science, The University of Warwick Coventry CV4 7AL, UK
[2]Warwick Systems Biology Centre, The University of Warwick Coventry CV4 7AL, UK
[3]Department of Computer Science, The University of Warwick Coventry CV4 7AL, UK

ABSTRACT

A standard tool in studying gene expression profiling data is clustering analysis. Bayesian hierarchical clustering (BHC) (Heller and Ghahramani, 2005) employing a multinomial prior has been shown to produce biologically meaningful results (Savage *et al.*, 2009) and the algorithm automatically learns the optimal number of clusters in the data. We show in this paper that the standard BHC algorithm is vulnerable to noise. In this paper, we present an extension of the BHC algorithm for non-time-series microarray data. We incorporate BHC with a Gaussian model assumption together with a normal-gamma conjugate prior to capture intrinsic structure of the data. We demonstrate our algorithm on four cancer gene expression datasets. The results show that our algorithm consistently produces biologically plausible results than other commonly used algorithms

1 INTRODUCTION

Microarray technology has become one of the indispensable tools in cancer studies (Babu, 2004). It enables the measurement of genetic signature of cancer cells in terms of gene expression data. With this type of data, one can perform several kinds of analyses, one of which is to identify groups of genes with similar expression profile across different experimental conditions (Dhaeseleer *et al.*, 2000). Genes in the same group are likely to be co-regulated by the same transcriptor (Eisen *et al.*, 1998). Also, these genes can be set as a classifier between subtypes of cancer, aiding stratificate of cancer patients for personalised medication (Kim *et al.*, 2010). On the other hand, one can identify groups of genes exhibiting similar expression pattern across observations, which can lead to a discovery of a new subtype of cancer (Alizadeh *et al.*, 2000; Golub *et al.*, 1999; Tritchler *et al.*, 2009).

Identifying a group of genes with a similar profile is not straightforward. With an advance in microarray technology, expression levels of several thousands of genes can be monitored in a parallel fashion, resulting in high-dimensional data. It is often the case that only a subset of genes provides useful information. Therefore, filtering needs to be carried out to get rid of non-informative genes (Hackstadt and Hess, 2009; Bourgon *et al.*, 2010; Tritchler *et al.*, 2009). Even if the number of dimension has been reduced, finding a structure inside the data can still be difficult. Clustering analysis plays an important role here. The aim of clustering analysis is simply to group similar objects together in the same place.

Clustering analysis has become a vital tool in microarray analysis as it has demonstrated significant results in numerous studies. Eisen *et al.* (1998) was a pioneer to use clustering analysis in microarray study. In the work, hierarchical clustering is employed to find groups of genes with similar function of *Saccharomyces cerevisiae*. Hierarchical clustering is the most frequently used clustering algorithm in the gene expression data analysis literature (Golub *et al.*, 1999; Alizadeh *et al.*, 2000; Laiho *et al.*, 2006; Singh *et al.*, 2002). Gasch *et al.* (2002) used a heuristically modified version of fuzzy k-means clustering to identify overlapping clusters of yeast genes based on published gene expression data. In the study, they have found good correlation between between yeast genes and between the experimental conditions, which provides insights into the mechanism of the regulation of gene expression in yeast cells corresponding to the environmental changes. McLachlan *et al.* (2002) demonstrated the use of a mixture model-based approach to clustering microarray expression data, in particular, clustering relatively small number of tissue samples based on a very large number of genes. Wang *et al.* (2002) used self-organizing map to reanalyse the published data of diffuse large B-cell lymphomas. The results showed three patterns of expression described in the original paper, plus one novel pattern.

Quite a number of clustering algorithms have been proposed to aid the investigation of microarray data. Commonly used approach such as hierarchical algorithm (Sokal and Michener, 1958; McQuitty, 1960; Sokal *et al.*, 1963), k-means (MacQueen *et al.*, 1967), and self-organising map (SOM) (Kohonen, 1990) gained their popularity since they can be used at ease. There are few or no parameters to be adjusted. However, these algorithms provide no guide

*Correspondence address: N.M.Rajpoot@warwick.ac.uk

about the "correct" numbers of clusters in the data. For hierarchical algorithms, identifying the number of clusters or level at which to prune a tree depends mainly on visual identification. For k-means and SOM, the number of clusters need predefined. Often, finding the correct number of clusters is difficult, and ones might end up introducing some bias into their analysis for not choosing the right number of clusters. Another question that needs to be addressed is how to select a distance metric. In hierarchical clustering algorithms, the relation between objects is considered in terms of dissimilarity, and is measured by the distance between them.

Bayesian hierarchical clustering (BHC) algorithm (Heller and Ghahramani, 2005) was proposed to overcome the limitations of traditional clustering algorithms. The advantages of BHC are that it is formulated as a statistical inference framework. Describing data with a probabilistic model, it allows us to say how good or bad the clustering result is in terms of probability. It uses hypothesis testing to decide which merge is advantage as to avoid overfitting. More importantly, it recommends the number of clusters in the data.

In this paper, we extend a standard BHC algorithm in order to apply to non-time-series microarray data. We integrate a standard BHC algorithm with a Gaussian model assumption and a normal-gamma prior since there are several literature confirming that a Gaussian model is suitable for describing this type of data (Yeung *et al.*, 2001; De Souto *et al.*, 2008; Medvedovic and Sivaganesan, 2002; Dubey *et al.*, 2004). We then test our algorithm against other commonly used and recently proposed methods based on 4 case studies.

2 METHODS

2.1 Bayesian hierarchical clustering (BHC) algorithm

Bayesian hierarchical clustering algorithm (BHC), proposed by Heller and Ghahramani (2005), is a type of agglomerative hierarchical algorithm which is a bottom-up process. Let $\mathcal{D} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(n)}\}$ denote a dataset containing n data points. Initially, each object forms its own cluster. So, we will have n trivial clusters $\mathcal{D}_i = \{\mathbf{x}^{(i)}\}$. Then BHC merges a pair of clusters $\mathcal{D}_i$ and $\mathcal{D}_j$, which are the most similar according to a statistical criterion, into a new cluster $\mathcal{D}_k = \mathcal{D}_i \cup \mathcal{D}_j$. The algorithm will merge only one pair of clusters at each step. The merging process is repeated until all data points are placed in a single cluster. We can represent this merging process by a dendrogram (see Figure 1). For any i, let T_i be a subtree whose leaf is a cluster $\mathcal{D}_i$. If any subtrees T_i and T_j are linked by a horizontal line, it means they are merged into a new tree T_k with a leaf $\mathcal{D}_k = \mathcal{D}_i \cup \mathcal{D}_j$. Furthermore, the level of a horizontal line is connected to dissimilarity between clusters.

BHC is a probabilistic model-based algorithm. It assumes that a data point is distributed according to some probability distribution, and the characteristic parameter of that distribution is again governed by some prior belief. In other words, a data point $\mathbf{x}^{(i)}$ can be described by a hierarchical model:

$$\mathbf{x}^{(i)} \sim p(\mathbf{x}^{(i)}|\theta) \tag{1}$$

$$\theta \sim p(\theta|\beta) \tag{2}$$

in which θ denotes a parameter of a distribution governing $\mathbf{x}^{(i)}$, and a hyperparameter β characterises a prior distribution.

Unlike traditional clustering algorithms which use distances, such as Euclidean metric, Manhattan, etc, as dissimilarity measure, BHC uses statistical hypothesis testing to decide which clusters should be fused together. Suppose we are considering to merge subtrees T_i and T_j into a subtree T_k as illustrated in Figure 1. We compare the following hypotheses. The null hypothesis, $\mathcal{H}_0$, states that all data points in $\mathcal{D}_k$ are identical and independently distributed according to the same probability distribution. The alternative hypothesis, $\mathcal{H}_1$, states that data points in $\mathcal{D}_i$ and data points in $\mathcal{D}_j$ are generated according to different distributions. Given this, we can express the marginal probability of data points in $\mathcal{D}_k$ under $\mathcal{H}_0$ as

$$P(\mathcal{D}_k|\mathcal{H}_0^k) = \int \left[\prod_{\mathbf{x}^{(i)} \in \mathcal{D}_k} p(\mathbf{x}^{(i)}|\theta) \right] p(\theta|\beta)\, d\theta. \tag{3}$$

The probability that $\mathcal{D}_k$ consists of two clusters according to $\mathcal{H}_1$ simply is

$$P(\mathcal{D}_k|\mathcal{H}_1^k) = P(\mathcal{D}_i|T_i)P(\mathcal{D}_j|T_j). \tag{4}$$

The above equation follows from the assumption that $\mathcal{D}_i$ and $\mathcal{D}_j$ are generated independently under different distributions. Using (3) and (4), we can recursively define the marginal

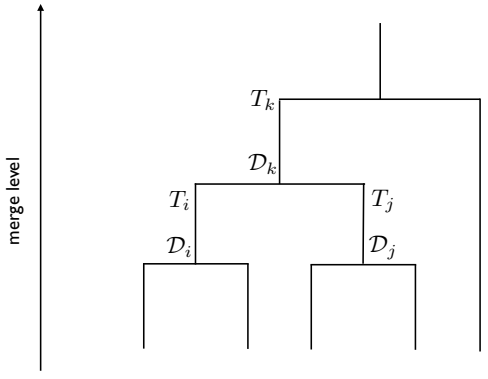


Fig. 1: An illustration of a part of a dendrogram. Subtrees T_i and T_j merge into a tree T_k, resulting in $\mathcal{D}_k = \mathcal{D}_i \cup \mathcal{D}_j$.

probability of data in T_k as

$$P(\mathcal{D}_k|T_k) = \pi_k P(\mathcal{D}_k|\mathcal{H}_0^k) + (1-\pi_k)P(\mathcal{D}_i|T_i)P(\mathcal{D}_j|T_j). \quad (5)$$

which is a weighted sum of probabilities of data under $\mathcal{H}_0$ and $\mathcal{H}_1$. The weight term π_k is defined as $\pi_k = P(\mathcal{H}_0^k)$, a prior that all data points in $\mathcal{D}_k$ are from the same cluster. Using Bayes' rule, we therefore obtain the merged hypothesis probability:

$$r_k = P(T_k|\mathcal{H}_0^k) = \frac{\pi_k P(\mathcal{D}_k|\mathcal{H}_0^k)}{P(\mathcal{D}_k|T_k)}. \quad (6)$$

The prior on merged hypothesis π_k is recursively given by

$$\pi_k = \frac{\alpha\Gamma(n_k)}{d_k} \quad (7)$$

$$d_k = \alpha\Gamma(n_k) + d_i d_j \quad (8)$$

where n_k denote the number of objects in $\mathcal{D}_k$, $\Gamma(\cdot)$ is a gamma function, and α is a concentration parameter. Note that the bigger the value of α, the higher the value of expected number of clusters. Here, π_k is intentionally defined in such a way that the probability of a new point joining an existing cluster is proportional to the number of data points already in that cluster. This property mimics a character of Dirichlet process mixture model. Indeed, BHC is a fast approximate method for Dirichlet process mixture model (Heller and Ghahramani, 2005) because it makes an inference based on a finite number of tree-consistent partitions rather than all possible partitions of a dataset.

One of the features that make BHC more desirable than other clustering algorithms is that it recommends at which level a dendrogram should be pruned. This automatically results in the final partition that we need. The algorithm prunes a tree T_k into T_i and T_j if the merged hypothesis probability r_k is less than 0.5. Intuitively, clusters are less likely to merge at this level.

Lastly, BHC algorithm can be implemented as follows (Heller and Ghahramani, 2005):

Inputs: a dataset $\mathcal{D} = \{\mathbf{x}^{(i)}, ..., \mathbf{x}^{(n)}\}$, a model $p(\mathbf{x}|\theta)$, a prior $p(\theta|\beta)$, and a concentration perameter α
Initialise: $\mathcal{D}_i = \{\mathbf{x}^{(i)}\}$, $d_i = \alpha, \pi_i = 1, n_i = 1$ for each leaf i, and set $c = n$
Construct a dendrogram:
while $c > 1$ **do**
  Calculate r_k for every pair of clusters.
  Find a pair $\mathcal{D}_i$ and $\mathcal{D}_j$ with the highest r_k
  Merge $\mathcal{D}_i$ and $\mathcal{D}_j$ into $\mathcal{D}_k$, i.e. $\mathcal{D}_k \leftarrow \mathcal{D}_i \cup \mathcal{D}_j$
  Delete $\mathcal{D}_i$ and $\mathcal{D}_j$. Set $c \leftarrow c - 1$
end while
Prune the dendrogram at level $r_k = 0.5$

2.2 A Gaussian model with a normal-gamma conjugate prior

Previously, a standard BHC algorithm as described in subsection 2.1 has been successfully applied to a gene expression data (Savage *et al.*, 2009), using a multinomial model assumption. On performing experimental sample clustering of *Arabidopsis thaliana* microarray dataset (de Torres-Zabala *et al.*, 2007), BHC with a multinomial model assumption shows higher dendrogram purity as well as more meaningful clusters in comparison with a complete-linkage hierarchical algorithm with uncentred correlation coefficient metric. For gene clustering on the same dataset, BHC with a multinomial model assumption still performs better. It produces more biologically meaningful clusters than those of the conventional hierarchical algorithm.

However, BHC with a multinomial model assumption has a downside. It requires us to first discretise continuous relative expression data into three bins, namely over-expressed, unchanged, and under-expressed. This imposes too strong an assumption on the data. Even if expression values are trivially fluctuating due to noise, they will be pushed into different bins anyway. In other words, it is highly sensitive to noise. Moreover, finding an optimal discretisation in a large dataset is very costly. This encourages us to find a way to improve it.

There are several factors pointing to a Gaussian model assumption as an alternative to the multinomial model assumption. Regarding a finite Gaussian mixture model (McLachlan and Basford, 1988), Yeung *et al.* (2001) used this method to cluster expression profiles. Given the true number of clusters, it can reliably assign an individual data to the correct cluster. This has been proven again recently by De Souto *et al.* (2008), where a finite Gaussian mixture model is the best approach among 7 different clustering algorithms to recover the underlying structure of cancer data, provided a correct number of clusters (see De Souto *et al.* 2008 for detail). Taking into account the number of clusters is often unknown, attention has been moved toward an infinite Gaussian mixture model (Ferguson, 1973; Neal, 2000; Rasmussen, 2000) which allows the data to automatically discover how many clusters it has. This model is also known as a Dirichlet process mixture model with a Gaussian model assumption. Medvedovic and Sivaganesan (2002) developed a clustering method for the microarray data based on an infinite multivariate Gaussian mixture model, where an optimal partition is found through MCMC and Gibbs sampler. Dubey *et al.* (2004) also used the same method to cluster protein sequences and discover protein families and subfamilies. Moreover, Gaussian process regression has been used in BHC for a microarray time-series data recently. This method consistently yields a high quality and biologically meaningful clustering results. As mentioned in Section 2.1, BHC is a fast inference method for a Dirichlet mixture

model, this encourages us a to try developing the BHC with a Gaussian model assumption.

Since our aim is to develop an alternative BHC method for a non-time-series microarray data, we assume explicitly that expressions across experiments are mutually independent. Also, we assume that expressions of different genes in the same experiment are identical and independently distributed according to a Gaussian distribution. Let $\mathbf{x}^{(i)} = (x_1^{(i)}, ..., x_d^{(i)})$ be an array containing expression values across d experiments of the i^{th} genes, and let its attribute $x_j^{(i)}$ denote an expression value of the i^{th} gene in the j^{th} experiment. Thus, we have that

$$x_j^{(i)} \sim \mathrm{N}(x_j^{(i)}|\mu, \sigma). \tag{9}$$

For a prior distribution on parameters μ and σ, we employ a normal-gamma distribution which is a conjugate of Gaussian distribution. Following the notation of DeGroot (2004), we write a normal-gamma prior as

$$\begin{aligned} &NG(\mu, \lambda|\mu_0, \kappa_0, \alpha_0, \beta_0) \\ &\quad = \mathrm{N}(\mu|\mu_0, (\kappa_0\lambda)^{-1})\mathrm{Gamma}(\lambda|\alpha_0, \beta_0) \\ &= \frac{1}{Z_{NG}(\mu_0, \kappa_0, \alpha_0, \beta_0)} \lambda^{\alpha_0 - \frac{1}{2}} \exp\left(-\frac{\lambda}{2}\left[\kappa_0(\mu - \mu_0)^2 + 2\beta_0\right]\right) \end{aligned} \tag{10}$$

where $\lambda = \sigma^{-2}$ is a precision parameter, $\mu_0, \kappa_0, \alpha_0$ and β_0 are hyperparameters, and

$$Z_{NG}(\mu_0, \kappa_0, \alpha_0, \beta_0) = \frac{\Gamma(\alpha_0)}{\beta_0^{\alpha_0}} \left(\frac{2\pi}{\kappa_0}\right)^{\frac{1}{2}}. \tag{11}$$

Let $\mathcal{D} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(n)}\}$ where n is the total number of data points, and $\mathcal{D}_j = \{\mathbf{x}_j^{(1)}, ..., \mathbf{x}_j^{(n)}\}$. A marginal distribution of data points in $\mathcal{D}_j$ is expressed as

$$P(\mathcal{D}_j|\mu_0, \kappa_0, \alpha_0, \beta_0) = \frac{\Gamma(\alpha_n)}{\Gamma(\alpha_0)} \frac{\beta_0^{\alpha_0}}{\beta_{n,j}^{\alpha_n}} \left(\frac{\kappa_0}{\kappa_n}\right)^{\frac{1}{2}} (2\pi)^{-\frac{n}{2}} \tag{12}$$

where

$$\kappa_n = \kappa_0 + n \tag{13}$$

$$\alpha_n = \alpha_0 + \frac{n}{2} \tag{14}$$

$$\beta_{n,j} = \beta_0 + \frac{1}{2}\left[\sum_{i=1}^{n}(x_j^{(i)} - \bar{x}_j)^2 + \frac{\kappa_0 n(\bar{x}_j - \mu)^2}{\kappa_n}\right]. \tag{15}$$

The total marginal distribution is then given by

$$P(\mathcal{D}|\mu_0, \kappa_0, \alpha_0, \beta_0) = \prod_{j=1}^{d}\left[p(\mathcal{D}_j|\mu_0, \kappa_0, \alpha_0, \beta_0)\right]. \tag{16}$$

The above equation is all we need in (3).

2.3 Other clustering methods

Apart from BHC with a Gaussian model assumption, we consider another 7 clustering algorithms: BHC with a multinomial model assumption, average-linkage, complete-linkage, k-means, divisive analysis (Diana), SOM, and affinity propagation (AP). These algorithms are frequently found in gene clustering analysis literature. Later on in this work, we shall refer to BHC with a multinomial model assumption as BHC(multinomial), and our BHC with a Gaussian model assumption as BHC(Gaussian).

BHC(multinomial) is constructed on the same basis as BHC(Gaussian), but it assumes that the data are multinomial distributed and therefore it is suitable for data of this type. To apply this method to microarray data, continuous expression levels for each gene need to be discretised into three levels (unchanged, under- or over-expressed). For more detail of how it is formulated, consult Heller and Ghahramani (2005) and Savage *et al.* (2009).

Average-linkage (Sokal and Michener, 1958) and complete-linkage methods (McQuitty, 1960; Sokal *et al.*, 1963; Macnaughton-Smith, 1965) are also a kind of agglomerative hierarchical algorithms. The differences between these algorithms and BHC are that firstly, they are not model-based algorithms. Therefore, distances are used as dissimilarity measures rather than for statistical hypothesis testing. Secondly, they do not tell how many clusters are there in the data. However, with the simplicity of algorithm, they are less computationally intensive. Another strong point is they are very easy to use. No parameters need to be adjusted. This makes these algorithms very popular in bioinformatic field (Costa *et al.*, 2004; Datta and Datta, 2003; Quackenbush *et al.*, 2001). Average-linkage and complete-linkage differ on how the dissimilarity between two clusters is calculated. Average-linkage uses the average distance of any pairs of data points between these clusters, whereas complete-linkage uses the largest distance among distances of any pair of data points between two clusters.

Diana (Kaufman *et al.*, 1990) is an example of divisive hierarchical algorithm. In contrast to agglomerative algorithms, it starts by gathering every data point into a single big cluster and then it splits a big cluster into two smaller clusters at each time step. The algorithm terminates when clusters are all singleton. Diana is also used in gene expression clustering (Datta and Datta, 2006; Sherlock, 2000; Jiang *et al.*, 2004) but vastly ignored in other literature.

k-means (MacQueen *et al.*, 1967) is a partitioning algorithm which is intensively used for gene expression data analysis. Given a pre-specified number of cluster, k-means find the optimal partition which minimises error sum of squares between data and their clusters' centroid. k-means has initial randomness. It starts with random partition. Then data points are assigned to a cluster of the nearest centroid. A centroid is recalculated every time there is a change occurring in a cluster. The algorithm stops when there are no changes in the assignment of data points to clusters taking place.

SOM (Kohonen, 1990) clusters a dataset by mapping a high-dimensional data into a lower dimensional space.

Higher-dimensional data points with close structure will be represented by the same object in the lower dimensional space. This mapping is unsupervisedly constructed through a neuron network.

AP (Frey and Dueck, 2007) is a partitioning algorithms which can also determine the number of clusters in the data by itself. Similar to k-means, it chooses a data point called examplar to represent a cluster. Picturing each data point as a node in a network, the algorithm recursively exchanges real-valued messages along edges of the network. At each time step, the affinity a data point has for choosing another point as its examplar is reflected by the magnitude of messages exchanged among them. This allows a set of examplars and their corresponding clusters to gradually emerge. AP has several applications in genes clustering analysis (Kiddle *et al.*, 2010; Frey and Dueck, 2007)

In this work, we have implemented BHC(Gaussian) and BHC(multinomial) algorithms by ourselves in MATLAB. AP is conducted using a MATLAB code written by Frey and Dueck (2007). All the rest of algorithms are carried out using freely available R packages. Average-linkage, complete linkage, and K-means algorithms are available in *stats* package. Diana is provided in *cluster* package, and SOM is provided in *kohonen* package.

2.4 Dissimilarity measures

A dissimilarity measure is key for traditional clustering analysis. Without dissimilarity measure, we cannot tell how close two data points or clusters are. Consequently, we cannot group data points or clusters which are more similar together. We use two dissimilarity measure, namely Euclidean distance and Pearson's correlation. According to microarray clustering analysis guideline by D'haeseleer *et al.* (2005), these metrics often produce a good result.

2.5 Performance validation indices

First, biological homogeneity index (BHI) (Datta and Datta, 2006) is used to evaluate the performance of algorithms on gene clustering. In gene clustering, the aspect in which we are interested is meaning of clusters. BHI is an index that indicates how biologically meaningful a clustering result is. Its score is between 0 and 1. Higher score will be assigned to a partition whose clusters exhibit more biological homogeneity. The reference of biological homogeneity is an annotation set. In this case, we use gene ontology (GO) annotations, which indicates biological functional classes of genes in three domains, namely cellular component, molecular function, and biological process. In this work, BHI score of a partition is calculated using *clValid* package in R statistical program (Brock *et al.*, 2008).

Second, adjusted rand index (ARI) (Hubert and Arabie, 1985) is used to evaluate the performance of algorithms on sample clustering. When performing sample clustering, the question is how well an algorithm recovers the ground truth structure. In this case, since the underlying structure of the data is known, ARI is a suitable index. This index tells us how agreeable a pair of partitions is, excluding by chance agreement. This allows us to compare between a partition from clustering and a true partition. A score of ARI is bounded by 1. The higher the score, the more agreeable a pair of partitions is.

2.6 Datasets and associated annotation databases

To assess the performance of our methods, we used four case studies. Three out of four are published microarray datasets. The first dataset is of colon cancer (Laiho *et al.*, 2006), and the second one is of bone marrow cancer (Golub *et al.*, 1999). Both have been proposed as a benchmark dataset in comparative study of clustering methods (De Souto *et al.*, 2008). We used the filtered version of these datasets, available online[1]. Filtered colon cancer dataset contains 8 serrated and 29 conventional colorectal carcinomas. Each of which has 2,202 probes. In our analysis, it was filtered again with Wilcoxon rank sum test with significance level 0.05, resulting in only 431 probes left. The filtered bone marrow cancer dataset consists of 72 samples: 47 acute myeloid leukemia and 25 acute lymphoblastic leukemia. The data has 1,877 probes. In the same way as colon cancer dataset, we filtered it again using Wilcoxon rank sum test with significance level 0.01. There are 373 probes passing the filter.

The third dataset is of breast cancer used in studying mechanisms underlying breast cancer initiation and progression (Graham *et al.*, 2010). It has 42 samples: 9 normal breast epithelia from ER- breast cancer patients, 9 normal breast epithelia from ER+ breast cancer patients, 6 normal breast epithelia from prophylactic mastectomy patients, and 18 normal breast epithelia from reduction mammoplasty patients. The original data has 22,283 probes. We use Wilcoxon rank sum for testing between a group of samples from cancer patients and a group of samples from non-cancer patients with significance level 0.01 to reduce dimensionality of the data, yielding 753 probes left.

In the last case study, we illustrated our method on an unpublished lung cancer dataset, which consists of 16 samples: 7 normal lung biopsies, 5 large-cell neuroendocrine carcinomas (LCNEC), and 4 small-cell lung carcinomas (SCLC). The microarray experiment of this data has been conducted on AGilent SurePrint G3 Human Gene Expression 8x60K microarrays. Then the raw expression data was processed using a bioconductor package *limma* in R statistical program, and after that it was quantiled normalised. A-priori to our analysis, Kruskal-Wallis test for testing between three subgroups with significant level 0.003 has been carried to filter noisy genes out. Finally, there are 628 significant probes left.

[1] http://algorithmics.molgen.mpg.de/Static/Supplements/CompCancer/datasets.htm

An annotation database associated with a dataset provides the connection between genes and their GO terms. We used annotation databases available in bioconductor[2]. The following are annotation databases we used: *hgu133a.db* (colon cancer), *hu6800.db* (bone marrow), *hgu133a2.db* (breast cancer), and *hgug4112a.db* (lung cancer).

3 RESULTS AND DISCUSSION

3.1 Comparison of BHC(Gaussian) to other clustering algorithms

We compared BHC(Gaussian) to other clustering algorithms mentioned in Section 2.3. In the case of BHC(Gaussian), we performed clustering based on three types of data: the original expression data, pairwise correlation distance matrix, and pairwise Euclidean distance matrix. In addition, for k-means and SOM which have random initialisation, we ran the algorithms for 100 times and the results were averaged.

Datasets described in Section 2.3 are used here. For each dataset, we performed both gene clustering and sample clustering. Regarding gene clustering, BHI was used to gauge the performance of each algorithm in terms of how well a biologically meaningful partition was produced. Note that besides BHC(Gaussian), BHC(multinomial), and AP which can automatically determine the number of clusters by themselves, the rest of algorithms need as an input a pre-specified number of clusters. So, we first compared BHC(Gaussian) to BHC(multinomial) and AP using the partitions they had found. The results are given in Table 1. Then we compared the best result among three settings of BHC(Gaussian) with the rest of algorithms. However, in order to make a fair comparison, the numbers of clusters for the other algorithms were assigned to be that of the setting of BHC(Gaussian) that display highest value of BHI. The results are shown in Table 2.

We can see that the performance of BHC(Gaussian) in producing biologically meaningful results is among the top. According to Table 1, comparing between algorithms which can determine the number of clusters, BHI scores produced by BHC(Gaussian) with all different types of data are regularly taking the first three places on a rank. For BHC(Gaussian) with the original expression data, it is consistently ranked as one of the first two best ranking methods. On average, BHC(Gaussian) with the original expression data and BHC(Gaussian) with a pairwise Euclidean distance matrix are the first and the second, respectively, to give the highest BHI scores. Also in Table 2, when a comparison was made at the same level, BHC(Gaussian) still works very well. The BHI scores of the best setting of BHC(Gaussian) sit within the first four out of nine on the ranking. BHC(Gaussian) comes in the first place on average.

For sample clustering, since the underlying structure of the data is known, we are interested in how well algorithms can rediscover the true structure given the actual number of classes. We therefore assigned the actual number of classes to every algorithm including BHC(Gaussian) and BHC(multinomial). Then, ARI was used to assess their performances. The results are shown in Table 3. When we consider the results of all 12 different settings on average, BHC(Gaussian) with a pairwise Euclidean distance matrix and BHC(Gaussian) with the original expression data take the third and the forth places on the rank, respectively.

3.2 Comparison of BHC(Gaussian) and BHC(multinomial)

We compared BHC(Gaussian) and BHC(multinomial) via both performance and visualisation of the clustering results. The clustering results of both algorithms on all datasets are illustrated in Figures 2 - 5. In each figure, the red lines in row or sample dendrograms indicate where the dendrogram is pruned. If genes or samples are connected by blue lines or black lines, it indicates that they are in the same cluster.

In terms of biological homogeneity of clustering results, BHC(Gaussian) performs better than BHC(multinomial) in our experiments. Judging by visual examination of every figure, it appears that BHC(multinomial) does very well in placing genes with similar expression pattern into the same group. Surprisingly this is not the case according to their BHI scores. The reason that BHC(multinomial) does not do well is over-discretisation. As mentioned earlier, BHC(multinomial) tends to discriminate continuous expression values into different categories even if they do not differ significantly. BHC(Gaussian) which is built to deal directly with continuous values therefore performs better.

In our experiments, BHC(Gaussian) with the original expression data and BHC(Gaussian) with a pairwise correlation distance matrix appear better than BHC(multinomial) on average in terms of recognising the true structure of the data. In particular for colon cancer data, ARI scores of these 2 settings of BHC(Gaussian) are significantly much higher than that of BHC(multinomial). From the visual examination of BHC(Gaussian) and BHC(multinomial) clustering results illustrated in Figure 4, we are able to see that BHC(Gaussian) recognises the actual classes of samples better than BHC(multinomial). Even though a heat map produced by BHC(multinomial) is more orderly, it is again a result from over-discretisation.

[2] http://www.bioconductor.org/

Table 1. BHI scores of partitions found by BHC(Gaussian), BHC(multinomial) and AP on performing gene clustering

dataset name	BHC(Gaussian) original data		BHC(Gaussian) correlation		BHC(Gaussian) Euclidean		BHC(multinomial)		AP correlation		AP Euclidean	
	# clusters	BHI	# clusters	BHI	# clusters	BHI	# clusters	BHI	# clusters	BHI	# clusters	BHI
bone marrow cancer	4	0.303^{2}	7	0.289	20	$\mathbf{0.311}^{1}$	7	0.277	26	0.282	15	0.300^{3}
breast cancer	23	$\mathbf{0.295}^{1}$	17	0.278^{2}	98	0.275^{3}	5	0.267	19	0.265	8	0.268
colon cancer	6	$\mathbf{0.275}^{1}$	13	0.267^{2}	38	0.267^{2}	8	0.240	24	0.246	10	0.261^{3}
lung cancer	23	0.250^{2}	29	0.224	83	0.245^{3}	8	0.243	5	$\mathbf{0.259}^{1}$	12	0.244
mean		$\mathbf{0.281}^{1}$		0.265		0.274^{2}		0.257		0.263		0.268^{3}

superscript numbers behind the BHI values indicate the rank of values within a row

Table 2. BHI scores of partitions found by all cluster algorithms when the number of clusters are assigned to be that of the best setting from BHC(Gaussian) given in Table 1

dataset name	# clusters	BHC (Gaussian)	Average-linkage correlation	Average-linkage Euclidean	k-means	Diana correlation	Diana Euclidean	Complete-linkage correlation	Complete-linkage Euclidean	SOM
bone marrow cancer	20	0.311^{2}	0.256	0.283	0.296^{3}	0.265	$\mathbf{0.346}^{1}$	0.283	0.262	0.291
breast cancer	23	0.295^{3}	0.215	0.303^{2}	0.284	0.282	0.276	0.268	$\mathbf{0.305}^{1}$	0.276
colon cancer	6	$\mathbf{0.275}^{1}$	0.201	0.269^{3}	0.263	0.245	0.261	0.234	0.271^{2}	0.264
lung cancer	23	0.250^{4}	0.231	$\mathbf{0.276}^{1}$	0.242	0.260^{2}	0.245	0.222	0.258^{3}	0.248
mean		$\mathbf{0.283}^{1}$	0.226	$\mathbf{0.283}^{1}$	0.271	0.263	0.282^{2}	0.252	0.274^{3}	0.270

superscript numbers behind the BHI values indicate the rank of values within a row

4 CONCLUSIONS

In this paper, we presented an extension to the standard BHC algorithm (Heller and Ghahramani, 2005) for non-time-series microarray data, assuming that the data is independent and identically Gaussian distributed. We employed a normal-gamma distribution as a prior on a Gaussian distribution parameters. We have intensively compared performance of our algorithm to that of other well-known algorithms on four different datasets. The results showed that firstly BHC(Gaussian) with the original expression data is the best algorithm on average to produce biologically meaningful results among the algorithms which can determine the number of clusters. Secondly, when we compared BHC(Gaussian) with other algorithms that need predefined number of clusters, BHC(Gaussian) still performs better on average. Thirdly, in terms of how well the algorithm can recognise the actual label of the data, BHC(Gaussian) with a pairwise Euclidean distance matrix comes as the third best setting on average. Furthermore, being more robust to noise, our algorithm can also be consider as an improvement of to the BHC with multinomial model assumption for non-time-series microarray data proposed by Savage *et al.* (2009).

We cannot definitely claim that our method is significantly better than the others in general, as the number of datasets involving in our study is too small to allow a strong conclusion. Moreover, in the case that data is not Gaussian distributed, our algorithm might not perform well. Therefore, we shall include more case studies in the future work.

Our future work will look into how to effectively find optimal hyperparameters of a normal-gamma distribution. Right now, we fixed the hyperparameters κ_0 and μ_0 to be constant and perform numerical gradient search for the hyperparameters α_0 and β_0. Clustering is performed every time to evaluate the objective function, making it very expensive. There is also an issue of computational cost that need to be improved. BHC has computational complexity $O(n^2)$, which makes it unpromising for large-scale data. We therefore are going to develop the randomised version of our algorithm, according to randomised BHC algorithm proposed by Heller and Ghahramani (2005).

5 ACKNOWLEDGEMENTS

The authors would like to thank Furqan Bari, Ph.D. student at Warwick Medical School and his supervisor, Dr. David Snead, Head of Pathology Department at University

Table 3. ARI score evaluated between an algorithm discovered partition and the true partition

dataset name	BHC (Gaussian) actual data	BHC (Gaussian) correlation	BHC (Gaussian) Euclidean	BHC (multinomial)	Average-linkage correlation	Average-linkage Euclidean	k-means	Diana correlation	Diana Euclidean	Complete-linkage Correlation	Complete-linkage Euclidean	SOM
bone marrow cancer	0.178	0.203	0.188	0.235	-0.019	**0.382**^{1}	0.263	0.331^{2}	0.263	0.283^{3}	0.229	0.225
breast cancer	0.046	0.249	0.199	0.238	0.049	0.064	0.280^{2}	0.082	0.156	0.258	0.278^{3}	**0.295**^{1}
colon cancer	0.769^{3}	0.106	0.769^{3}	0.296	-0.088	**0.878**^{1}	0.531	-0.111	0.777^{2}	-0.102	-0.090	0.694
lung cancer	**1**	**1**	**1**	**1**	**1**	**1**	0.764	**1**	**1**	**1**	**1**	0.742
mean	0.498^{4}	0.390	0.539^{3}	0.442	0.236	**0.581**^{1}	0.459	0.325	0.549^{2}	0.360	0.354	0.489

superscript numbers behind the BHI values indicates the rank of values within a row

Hospitals Coventry and Warwickshire for preprocessing and providing us the lung cancer dataset. We are grateful to Katherine A. Heller for sharing her code for the standard BHC algorithm.

REFERENCES

Alizadeh, A., Eisen, M., Davis, R., Ma, C., Lossos, I., Rosenwald, A., Boldrick, J., Sabet, H., Tran, T., Yu, X., *et al.* (2000). Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, **403**(6769), 503–511.

Babu, M. (2004). Introduction to microarray data analysis. *Computational Genomics: Theory and Application*, pages 225–249.

Bourgon, R., Gentleman, R., and Huber, W. (2010). Independent filtering increases detection power for high-throughput experiments. *Proceedings of the National Academy of Sciences*, **107**(21), 9546.

Brock, G., Pihur, V., Datta, S., and Datta, S. (2008). clValid: An R package for cluster validation. *Journal of Statistical Software*, **25**(4), 1–22.

Costa, I., Carvalho, F., and Souto, M. (2004). Comparative analysis of clustering methods for gene expression time course data. *Genetics and Molecular Biology*, **27**(4), 623–631.

Datta, S. and Datta, S. (2003). Comparisons and validation of statistical clustering techniques for microarray gene expression data. *Bioinformatics*, **19**(4), 459–466.

Datta, S. and Datta, S. (2006). Methods for evaluating clustering algorithms for gene expression data using a reference set of functional classes. *BMC bioinformatics*, **7**(1), 397.

De Souto, M., Costa, I., De Araujo, D., Ludermir, T., and Schliep, A. (2008). Clustering cancer gene expression data: a comparative study. *BMC bioinformatics*, **9**(1), 497.

de Torres-Zabala, M., Truman, W., Bennett, M., Lafforgue, G., Mansfield, J., Egea, P., Bögre, L., and Grant, M. (2007). Pseudomonas syringae pv. tomato hijacks the Arabidopsis abscisic acid signalling pathway to cause disease. *The EMBO journal*, **26**(5), 1434–1443.

DeGroot, M. (2004). *Optimal statistical decisions*, volume 82. John Wiley & Sons.

D'haeseleer, P. *et al.* (2005). How does gene expression clustering work? *Nature biotechnology*, **23**(12), 1499–1502.

Dubey, A., Hwang, S., Rangel, C., Rasmussen, C., Ghahramani, Z., and Wild, D. (2004). Clustering protein sequence and structure space with infinite Gaussian mixture models. In *Pacific Symposium on Biocomputing*, volume 9, pages 399–410. World Scientific Publishing: Singapore.

Dhaeseleer, P., Liang, S., and Somogyi, R. (2000). Genetic network inference: from co-expression clustering to reverse engineering. *Bioinformatics*, **16**(8), 707–726.

Eisen, M., Spellman, P., Brown, P., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, **95**(25), 14863.

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

Frey, B. and Dueck, D. (2007). Clustering by passing messages between data points. *Science*, **315**(5814), 972–976.

Gasch, A., Eisen, M., *et al.* (2002). Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. *Genome Biol*, **3**(11), 1–22.

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., *et al.* (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, **286**(5439), 531–537.

Graham, K., de Las Morenas, A., Tripathi, A., King, C., Kavanah, M., Mendez, J., Stone, M., Slama, J., Miller, M., Antoine, G., *et al.* (2010). Gene expression in histologically normal epithelium from breast cancer patients and from cancer-free prophylactic mastectomy patients shares a similar profile. *British journal of cancer*, **102**(8), 1284–1293.

Hackstadt, A. and Hess, A. (2009). Filtering for increased power for microarray data analysis. *Bmc Bioinformatics*, **10**(1), 11.

Heller, K. and Ghahramani, Z. (2005). Bayesian hierarchical clustering. In *Proceedings of the 22nd international conference on Machine learning*, pages 297–304. ACM.

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

Jiang, D., Tang, C., and Zhang, A. (2004). Cluster analysis for gene expression data: A survey. *Knowledge and Data Engineering, IEEE Transactions on*, **16**(11), 1370–1386.

Kaufman, L., Rousseeuw, P., *et al.* (1990). *Finding groups in data: an introduction to cluster analysis*, volume 39. Wiley Online Library.

Kiddle, S., Windram, O., McHattie, S., Mead, A., Beynon, J., Buchanan-Wollaston, V., Denby, K., and Mukherjee, S. (2010). Temporal clustering by affinity propagation reveals transcriptional modules in arabidopsis thaliana. *Bioinformatics*, **26**(3), 355–362.

Kim, W., Kim, E., Kim, S., Kim, Y., Ha, Y., Jeong, P., Kim, M., Yun, S., Lee, K., Moon, S., *et al.* (2010). Predictive value of progression-related gene classifier in primary non-muscle invasive bladder cancer. *Mol Cancer*, **9**(3).

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, **78**(9), 1464–1480.

Laiho, P., Kokko, A., Vanharanta, S., Salovaara, R., Sammalkorpi, H., Järvinen, H., Mecklin, J., Karttunen, T., Tuppurainen, K., Davalos, V., *et al.* (2006). Serrated carcinomas form a subclass of colorectal cancer with distinct molecular basis. *Oncogene*, **26**(2), 312–320.

Macnaughton-Smith, P. (1965). *Some statistical and other numerical techniques for classifying individuals*, volume 6. HMSO.

MacQueen, J. *et al.* (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA.

McLachlan, G. and Basford, K. (1988). Mixture models. inference and applications to clustering. *Statistics: Textbooks and Monographs, New York: Dekker, 1988*, **1**.

McLachlan, G., Bean, R., and Peel, D. (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, **18**(3), 413–422.

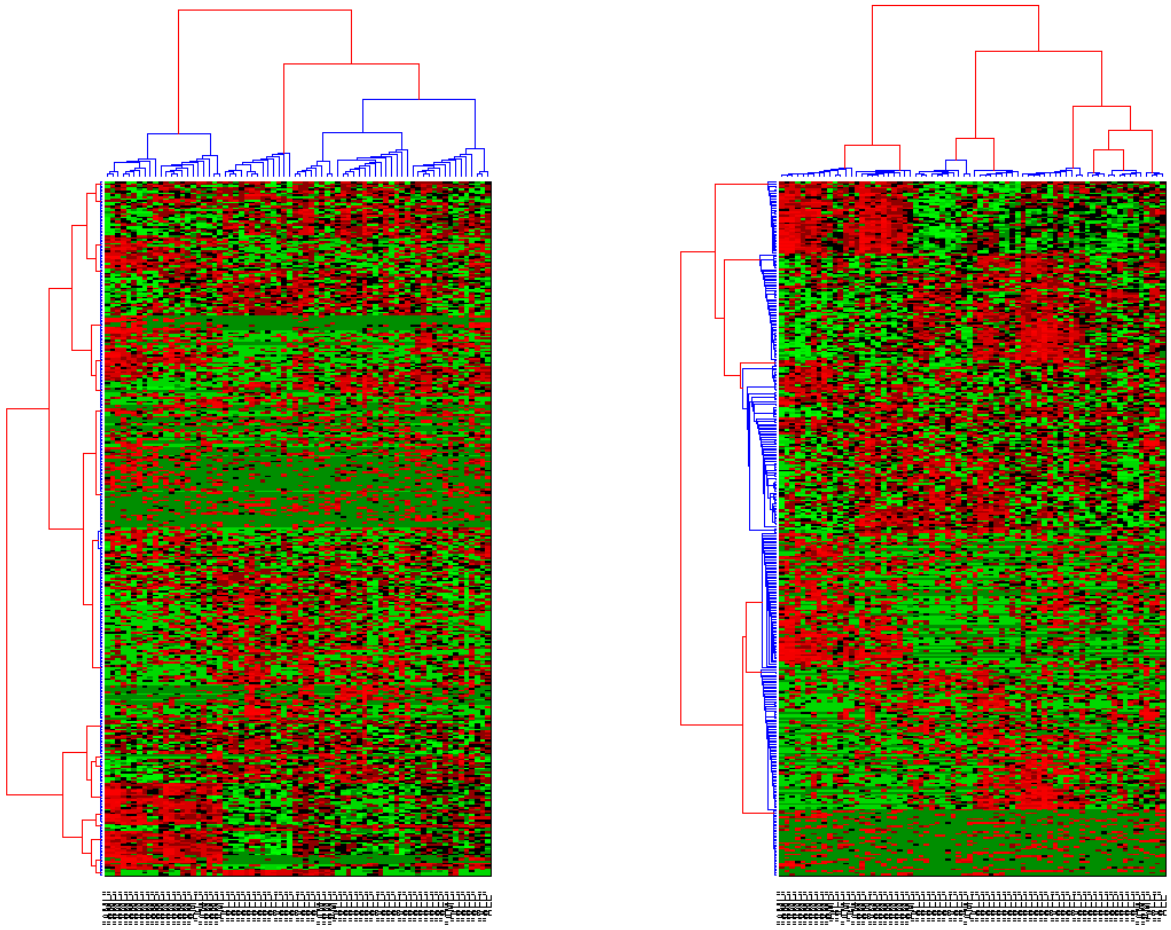
McQuitty, L. (1960). Hierarchical linkage analysis for the isolation of types. *Educational and Psychological Measurement*, **20**(1), 55–67.

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

Neal, R. (2000). Markov chain sampling methods for Dirichlet process mixture models. *Journal of computational and graphical statistics*, pages 249–265.

Quackenbush, J. *et al.* (2001). Computational analysis of microarray data. *Nature Reviews Genetics*, **2**(6), 418–427.

Rasmussen, C. (2000). The infinite Gaussian mixture model. *Advances in neural information processing systems*, **12**(5.2), 2.



(a) A heatmap found by BHC(Gaussian) algorithm.

(b) A heatmap found by BHC(multinomial) algorithm.

Fig. 2: Hierarchical clustering results on the bone marrow cancer dataset. In Figure 2a, gene clustering was performed on a pairwise Euclidean distance matrix of genes in the original expression data, and sample clustering was performed on a pairwise correlation distance matrix of sample in the original expression data. In Figure 2b, for each gene, the original expression values were discretised into three groups (over-expressed, unchanged, and under-expressed) prior to clustering. In dendrograms, the merge that BHC algorithms do not prefer to make is displayed by red line. The blue lines show the preferable merges. Up-regulated genes are presented by red colors and down-regulated genes are presented by green colors.

Savage, R., Heller, K., Xu, Y., Ghahramani, Z., Truman, W., Grant, M., Denby, K., and Wild, D. (2009). R/BHC: fast Bayesian hierarchical clustering for microarray data. *BMC bioinformatics*, **10**(1), 242.

Sherlock, G. (2000). Analysis of large-scale gene expression data. *Current opinion in immunology*, **12**(2), 201–205.

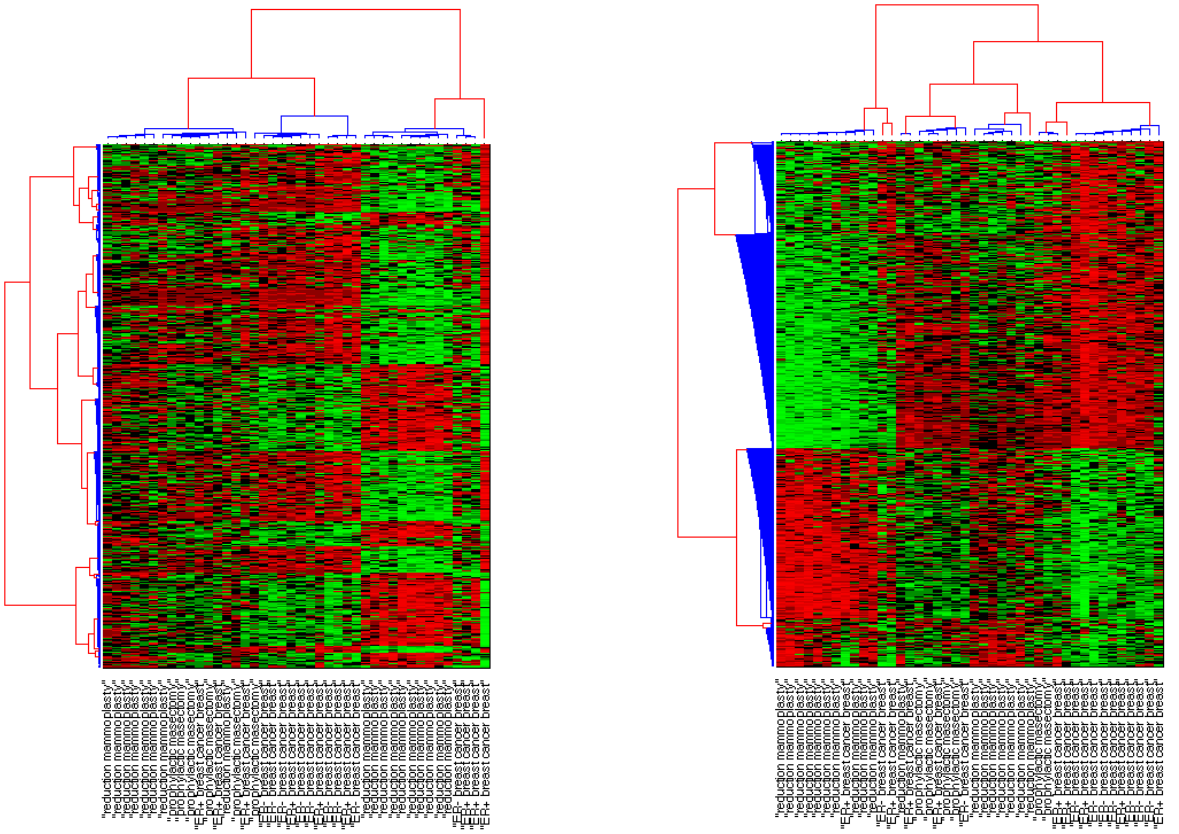
Singh, D., Febbo, P., Ross, K., Jackson, D., Manola, J., Ladd, C., Tamayo, P., Renshaw, A., D'Amico, A., Richie, J., *et al.* (2002). Gene expression correlates of clinical prostate cancer behavior. *Cancer cell*, **1**(2), 203–209.

Sokal, R. and Michener, C. (1958). A statistical method for evaluating systematic relationships. *Univ. Kans. Sci. Bull.*, **38**, 1409–1438.

Sokal, R., Sneath, P., *et al.* (1963). Principles of numerical taxonomy. *Principles of numerical taxonomy.*

Tritchler, D., Parkhomenko, E., and Beyene, J. (2009). Filtering genes for cluster and network analysis. *BMC bioinformatics*, **10**(1), 193.

Wang, J., Delabie, J., Aasheim, H., Smeland, E., and Myklebost, O. (2002). Clustering of the SOM easily reveals distinct gene expression patterns: results of a reanalysis



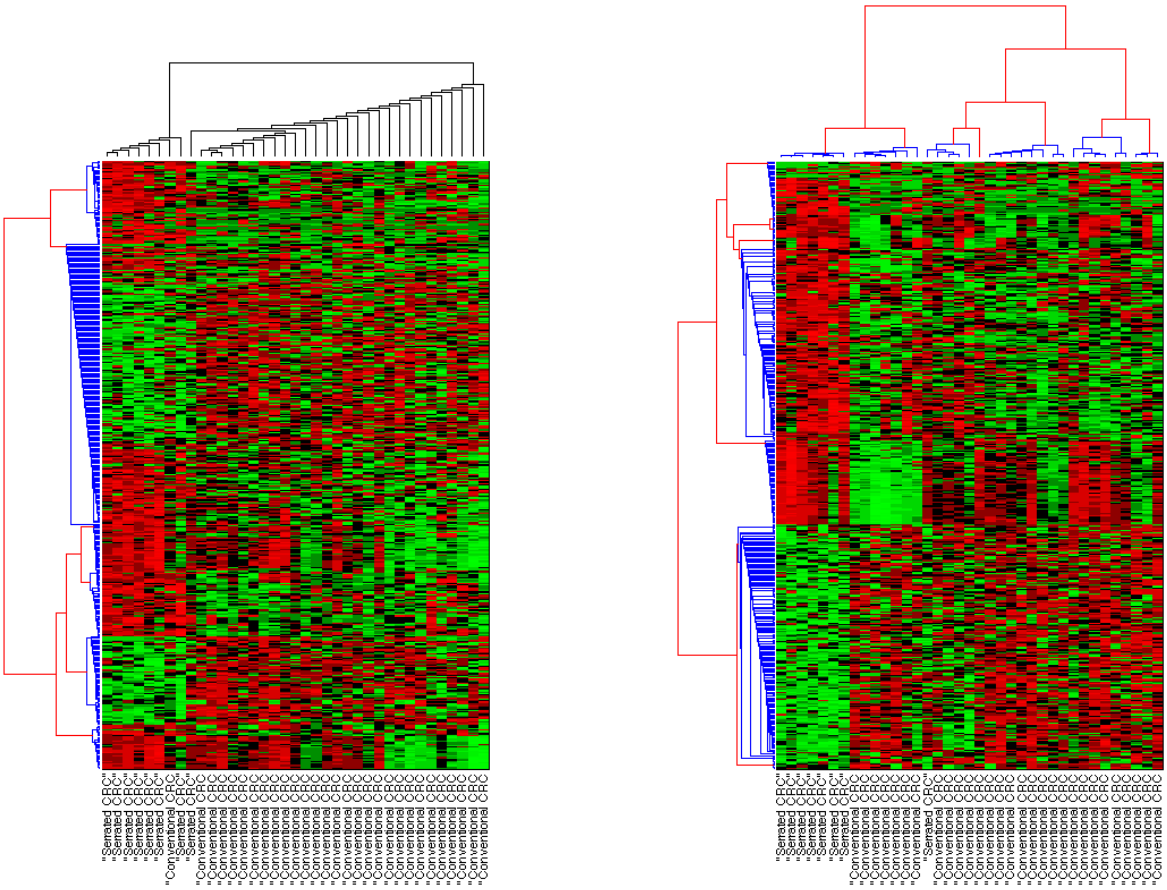
(a) A heatmap found by BHC(Gaussian) algorithm.

(b) A heatmap found by BHC(multinomial) algorithm.

Fig. 3: Hierarchical clustering results on the breast cancer dataset. In Figure 3a, gene clustering was performed on the original expression data, and sample clustering was performed on a pairwise correlation distance matrix of samples in the original expression data. In Figure 3b, for each gene, the expression values were discretised into three groups (over-expressed, unchanged, and under-expressed) prior to clustering. In dendrograms, the merge that BHC algorithms do not prefer to make is displayed by red line. The blue lines show the preferable merges. Up-regulated genes are presented by red colors and down-regulated genes are presented by green colors.

of lymphoma study. *BMC bioinformatics*, **3**(1), 36.

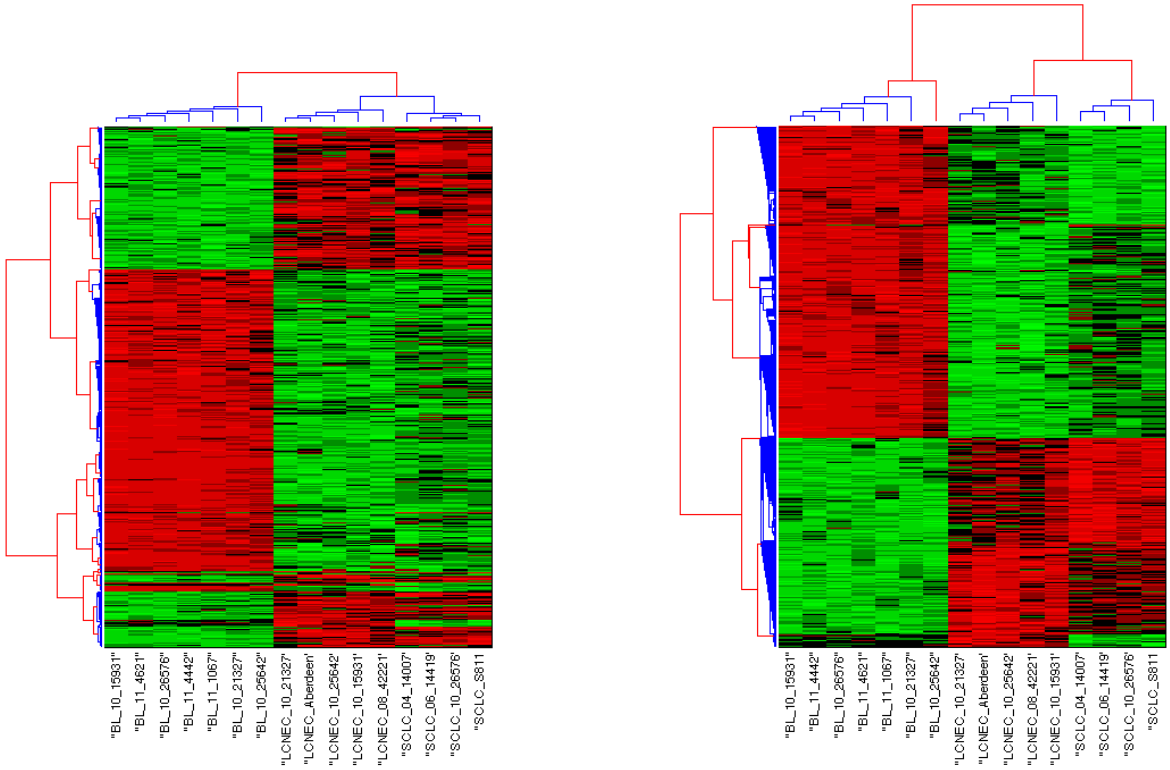
Yeung, K., Fraley, C., Murua, A., Raftery, A., and Ruzzo, W. (2001). Model-based clustering and data transformations for gene expression data. *Bioinformatics*, **17**(10), 977–987.



(a) A heatmap found by BHC(Gaussian) algorithm.

(b) A heatmap found by BHC(multinomial) algorithm.

Fig. 4: Hierarchical clustering results on colon cancer dataset. In Figure 5a, both gene and sample clusterings were performed on the original expression data. In Figure 5b, for each gene, the expression values were discretised into three groups (over-expressed, unchanged, and under-expressed) prior to clustering. In dendrograms, the merge that BHC algorithms do not prefer to make is displayed by red line. The blue and black lines both show the preferable merges. Up-regulated genes are presented by red colors and down-regulated genes are presented by green colors.



(a) A heatmap found by BHC(Gaussian) algorithm.

(b) A heatmap found by BHC(multinomial) algorithm.

Fig. 5: Hierarchical clustering results on lung cancer dataset. In Figure 5a, both gene and sample clusterings were performed on the original expression data. In Figure 5b, the expression values were discretised into three groups (over-expressed, unchanged, and under-expressed) prior to clustering. In dendrograms, the merge that BHC algorithms do not prefer to make is displayed by red line. The blue lines show the preferable merges. Up-regulated genes are presented by red colors and down-regulated genes are presented by green colors.