BMC Bioinformatics

BioMed Central

Research article

Open Access

Including probe-level uncertainty in model-based gene expression clustering

Xuejun Liu[1], Kevin K Lin[2], Bogi Andersen[2] and Magnus Rattray*[3]

Address: [1]College of Information Science and Technology, Nanjing University of Aeronautics and Astronautics, 29 Yudao Street, Nanjing 210016, China, [2]Departments of Biological Chemistry and Medicine, Institute for Genomics and Bioinformatics, University of California, Irvine, CA 92697, USA and [3]School of Computer Science, University of Manchester, Kilburn Building, Oxford Road, Manchester M13 9PL, UK

Email: Xuejun Liu - xuejun.liu@nuaa.edu.cn; Kevin K Lin - kklin@uci.edu; Bogi Andersen - bogi@uci.edu; Magnus Rattray* - magnus.rattray@manchester.ac.uk

* Corresponding author

Published: 21 March 2007

BMC Bioinformatics 2007, **8**:98 doi:10.1186/1471-2105-8-98

Received: 14 September 2006
Accepted: 21 March 2007

This article is available from: http://www.biomedcentral.com/1471-2105/8/98

Abstract

Background: Clustering is an important analysis performed on microarray gene expression data since it groups genes which have similar expression patterns and enables the exploration of unknown gene functions. Microarray experiments are associated with many sources of experimental and biological variation and the resulting gene expression data are therefore very noisy. Many heuristic and model-based clustering approaches have been developed to cluster this noisy data. However, few of them include consideration of probe-level measurement error which provides rich information about technical variability.

Results: We augment a standard model-based clustering method to incorporate probe-level measurement error. Using probe-level measurements from a recently developed Affymetrix probe-level model, multi-mgMOS, we include the probe-level measurement error directly into the standard Gaussian mixture model. Our augmented model is shown to provide improved clustering performance on simulated datasets and a real mouse time-course dataset.

Conclusion: The performance of model-based clustering of gene expression data is improved by including probe-level measurement error and more biologically meaningful clustering results are obtained.

Background

Microarrays [1,2] are routinely used for the quantitative measurement of gene expression levels on a genome-wide scale. Microarray experiments are complicated multiple step procedures and variability can be introduced in every step, so that the resulting data are often very noisy, especially for weakly expressed genes. Appropriate statistical analysis of this noisy data is very important in order to obtain meaningful biological information [3,4]. The analysis of microarray data is usually performed in multiple stages, including probe-level analysis, normalisation and higher level analyses. The aim of the probe-level analysis is to obtain reliable gene expression measurements from the image data. Various higher level analyses, such as detecting differential gene expression or clustering, can then be carried out depending on the biological aims of the experiment.

Unsupervised clustering is the most frequently used approach for exploring gene function. By clustering, a huge number of genes can be organised into a much smaller number of categories according to their shared expression patterns. It is hoped that these shared patterns reflect similar function or common transcriptional regulation. Exploring and studying the obtained gene clusters is an important way to infer the function of uncharacterised genes from other known genes in the same cluster. There are many unsupervised algorithms which have been used to cluster gene expression data, including the most popular hierarchical clustering [5] and *k*-means [6], which are based on similarity measures, and self-organising maps [7]. Most of these conventional algorithms are largely heuristically motivated. They are easily implemented and their application is usually computationally efficient. However, these methods lack the capability to deal in a principled way with the experimental variability in the gene expression data. Furthermore, there is no formal way to determine the number of clusters with these algorithms. It is hard to say which one is generally better than the others [8]. Probabilistic models provide a principled alternative to these conventional methods. In particular, model-based approaches have been proposed as useful methods for clustering gene expression data in a probabilistic way [9-12]. By using a probabilistic model, the experimental noise can be included explicitly in the model and estimated from the data, making this approach more robust to noise. There are also useful and principled model selection methods that can be used to determine the optimal number of clusters. The advantages of model-based probabilistic approaches over heuristic methods are already well established [10].

Affymetrix arrays contain multiple probes for each target gene and this internal replication can be used to obtain an estimate of the technical measurement error associated with each gene expression measurement [13-17]. This source of error is especially significant for weakly expressed genes. The recently developed model, multi-mgMOS [18], provides accurate gene expression measurements along with the associated uncertainty in this measurement. It has been shown that the probe-level measurement error obtained from multi-mgMOS can be propagated through a downstream probabilistic analysis, thereby improving the performance of the analysis [16,17]. Existing model-based clustering methods do not consider this probe-level measurement error and they therefore discard this rich source of information about variability. Although standard model-based clustering methods are relatively robust to noise, very noisy measurements can still have a detrimental effect on these clustering methods, resulting in poor performance and many biologically irrelevant clusters. In this paper, we aim to include information about probe-level measurement error into the standard Gaussian mixture model in order to improve performance compared to standard model-based clustering. Our augmented Gaussian mixture clustering model is called PUMA-CLUST (Propagating Uncertainty in Microarray Analysis – CLUSTering) and has been implemented in the R-package *pumaclust* which is available from [19].

Results and discussion

We examine the performance of the extended Gaussian mixture model on two simulated datasets and a real-world mouse time-course dataset [12]. The simulated datasets are generated to reflect the noise commonly seen in real microarray experiments. The extended mixture model is compared with the standard Gaussian mixture models implemented in MCLUST [20], which includes all variants of standard Gaussian mixture models in terms of the representation of the covariance matrix. However, these models do not take the probe-level measurement error into consideration.

The performance of different clustering methods on datasets with known structures can be evaluated by using the adjusted Rand index [21,22]. The adjusted Rand index measures the similarity of two clusterings on a dataset and it is widely used by the clustering research community [10,23-25]. The adjusted Rand index lies between 0 and 1, and is calculated based on whether pairs are placed in the same or different clusters in two partitions with a higher value meaning better agreement between two clusterings. For the simulated datasets, since the true structure of the data is known, we use the adjusted Rand index to evaluate the different partitioning ability of the extended mixture model which incorporates the probe-level measurement error and the standard mixture model. For the real mouse time-course dataset, gene ontology (GO) enrichment analysis is used to compare the performance of the two clustering methods.

Clustering on simulated data sets

Simulated periodic data

Periodic patterns are often observed in real-world time-course microarray data [12,26]. However, the true structure of the real datasets is unavailable. We generate simulated periodic data and include noise with magnitude estimated from real microarray data. Similar to the methods used by [23] and [25], the simulated data is generated by the following four steps.

At the first step, the logged gene expression within each known group is generated. There are six groups and 600 genes in the dataset. Each group has 100 genes. The first four groups have a periodic sine pattern. The expression of gene i in group q, q = 1, 2, 3, 4, is generated by

$$x_{qij} = A_i \sin(2\pi j/10 - \pi q/2) + S, \quad (1)$$

where j = 1, 2,..., J and J is the number of conditions or time points. A_i is a random scaling factor which is sampled from U(0, 7), where U represents the uniform distribution. S is a shifting factor which is set as 7. This assignment of A_i and S is to make the gene expression level lie between 0 and 14 which is the normal range of the logged gene expression level from real Affymetrix datasets. The gene expression levels of group 5 and group 6 are generated by linear functions

$$x_{qij} = jA_{qi}/J \text{ and } x_{qij} = -jA_{qi}/J + S, \quad (2)$$

respectively, where A_{qi} is sampled from U (0,14) and S = 14 when q = 6 so as to ensure that the simulated expression level lies within the accepted logged expression range.

The simulated data from the first step follows perfectly the same sine wave within the same group except for a different magnitude. However, in practice there is biological and technical noise in the experiment distorting the true sine wave. At the second step, the real mouse dataset (described in the next section) is used to obtain an estimate of the combined noise from biological and technical sources which is related to the variance of observed gene expression level from replicated experiments. The mouse dataset has three or four replicates for each condition. Using the gene expression summaries from MAS 5.0 [27] which is the standard software provided by Affymetrix, an estimate of the combined technical and biological noise can be obtained from Cyber-T [28]. Cyber-T is a Bayesian hierarchical model which calculates the variance between replicates using point estimates of gene expression level from each replicate. Since the variance has a dependence on gene expression level, the combined noise, σ^2_{qij}, is sampled from a subset of variances calculated from Cyber-T whose corresponding expression levels are close to x_{qij}. Thus, the final simulated expression level, $\hat{x}_{qij}$, is

$$\hat{x}_{qij} = x_{qij} + \varepsilon_{qij}, \quad (3)$$

where ε_{qij} is drawn from $\mathcal{N}$ (0, σ^2_{qij}). When J = 10, the simulated expression level for group three is shown in Figure 1(a). It can be seen that there is more noise for the lower expressed genes than the highly expressed ones, which is a common feature of real datasets.

At the third step, in order to show the clustering improvement by including probe-level measurement error, we sample the corresponding probe-level variance of the sim-

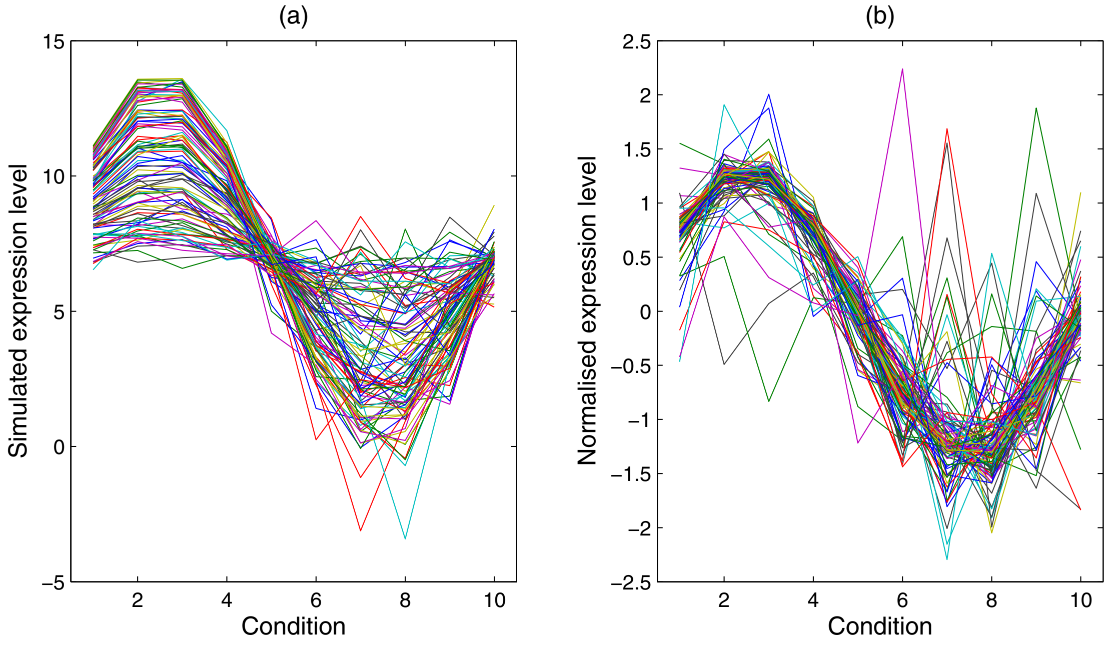

Figure 1
Simulated expression profiles. Simulated expression profiles for one group under 10 conditions. (a) are the raw data on a log scale and (b) are the normalised profiles with zero mean and standard deviation one.

ulated expression level from the real mouse dataset processed by multi-mgMOS. Similar to the second step, since the measurement error has a dependence on the gene expression level, the standard deviation for each simulated expression value, $\hat{\sigma}_{qij}$ is sampled from a subset of standard deviation calculated from multi-mgMOS whose corresponding expression levels are close to $\hat{x}_{qij}$. Figure 2(a) shows the scatter plot of the sampled standard deviation against the simulated expression level for one randomly selected condition. It can be seen that the variance of the measured gene expression for the weakly expressed genes is generally larger than that for the highly expressed genes as is commonly observed in real datasets. This is consistent with the plot in Figure 1(a). At the final step, we normalise the simulated expression level for each gene over all conditions by subtracting the mean expression level and dividing by the standard deviation such that the profile of each gene has zero mean and standard deviation one. The simulated standard deviation is also divided by the standard deviation of the expression level to determine the corresponding measurement error of the normalised data. The normalised profile is shown in Figure 1(b) when $J = 10$.

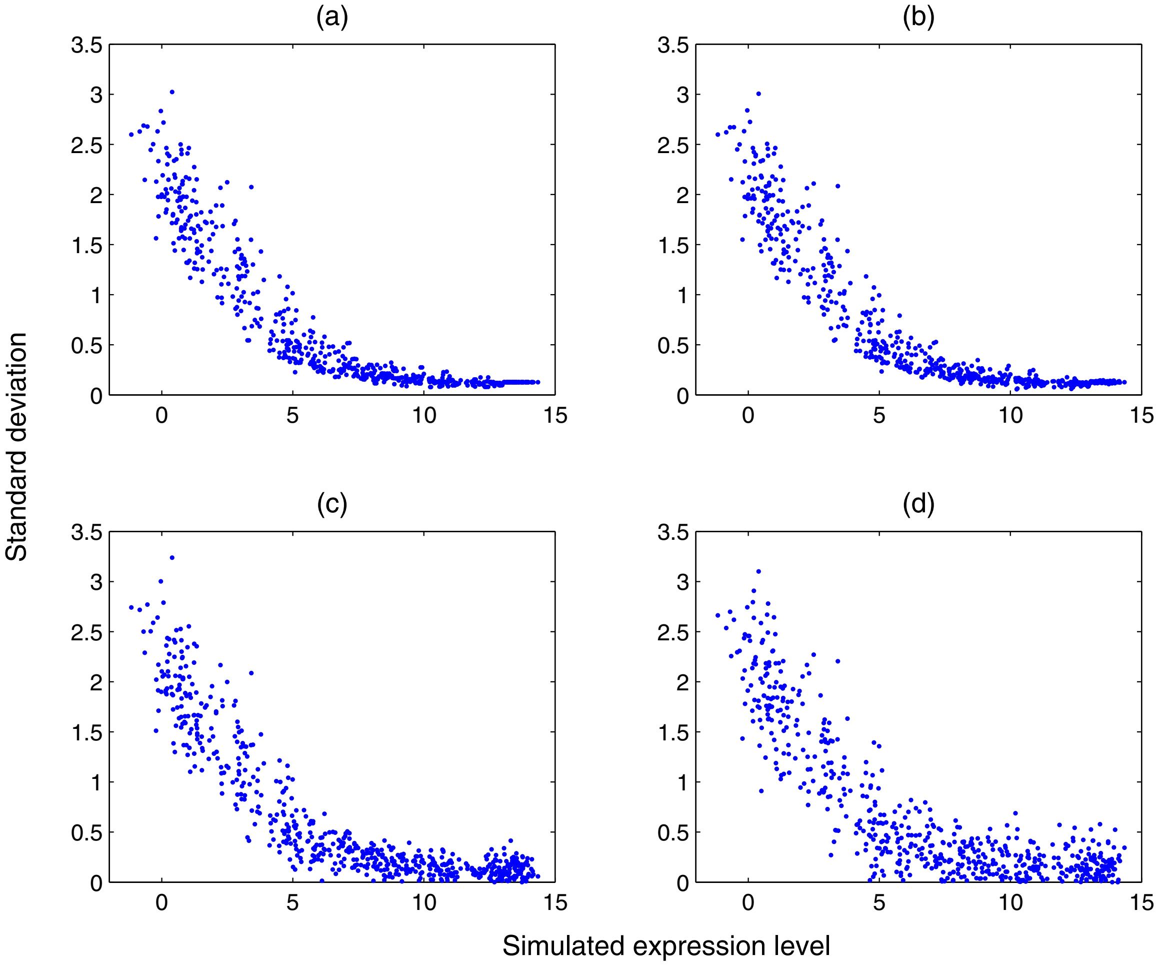

Figure 2

Standard deviation against the simulated gene expression level. Scatter plots of standard deviation against the simulated gene expression level. The standard deviation in (a) is sampled from the multi-mgMOS results obtained from the mouse dataset. The standard deviation is randomly changed by adding a noise drawn from (b) $\mathcal{N}$ (0, 0.01), (c) $\mathcal{N}$ (0, 0.1) and (d) $\mathcal{N}$ (0, 0.2).

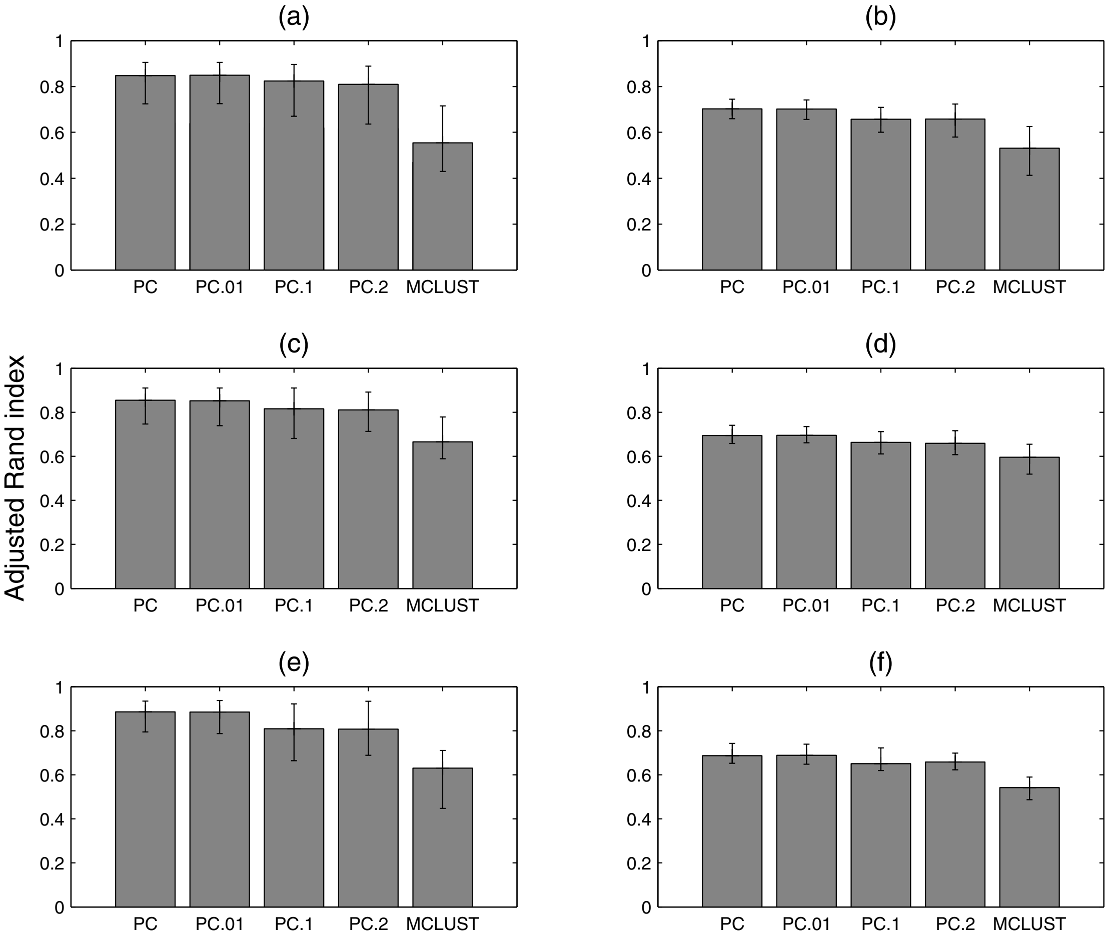

Figure 3

Average adjusted Rand index. The average adjusted Rand index of the clustering results from PUMA-CLUST and MCLUST on the simulated data. The first column is for the six group dataset and the second column is for the seven group dataset with one noise group added. The upper panel shows results on datasets with 10 conditions, the middle panel is for 20 conditions and the lower panel is for 30 conditions. PC represents PUMA-CLUST results on the original simulated data. PC.01, PC.1 and PC.2 represent the PUMA-CLUST results on the datasets with added noise drawn from $\mathcal{N}$ (0, 0.01), $\mathcal{N}$ (0, 0.1) and $\mathcal{N}$ (0, 0.2) respectively. The average adjusted Rand index is calculated over 10 simulated datasets for each plot and the range of the adjusted Rand index of each case is shown by error bars.

Since the true partition of the simulated dataset is known, the agreement of the clustering results from different methods with the true partition can be assessed by the adjusted Rand index. The true number of groups, six, is selected for both MCLUST and PUMA-CLUST. Three sets of datasets are generated to evaluate the different performance of PUMA-CLUST and MCLUST with number of conditions 10, 20 and 30. For each set, 10 random simulated datasets are generated. The average adjusted Rand index from PUMA-CLUST and MCLUST are shown in the first column of Figure 3. For the three sets of simulated datasets, PUMA-CLUST results in markedly better performance compared with MCLUST and the *p*-values of a paired t-test, shown in Table 1, indicate that the difference in performance is highly significant.

Including a noise group

In a real-world microarray dataset, there are usually a certain fraction of genes whose expression levels are indistin-

Table 1: *P*-values obtained from a paired t-test of adjusted Rand index from MCLUST and PUMA-CLUST. A paired t-test is performed for MCLUST and each of PUMA-CLUST results. The 10 simulated datasets in Figure 3 are used for each test. PC represents PUMA-CLUST results on the original simulated data. PC.01, PC.1 and PC.2 represent the PUMA-CLUST results on the datasets with added noise drawn from $\mathcal{N}$ (0, 0.01), $\mathcal{N}$ (0, 0.1) and $\mathcal{N}$ (0, 0.2) respectively.

No of conditions	6 groups				7 groups			
	PC	PC.01	PC.1	PC.2	PC	PC.01	PC.1	PC.2
10	1.10e-8	9.37e-8	5.90e-8	5.67e-7	5.67e-9	7.77e-9	3.87e-7	5.87e-6
20	2.39e-8	1.80e-8	2.30e-7	4.22e-7	4.03e-9	4.10e-9	1.13e-7	8.56e-8
30	3.54e-7	1.38e-6	2.99e-6	5.00e-6	9.96e-7	4.34e-7	1.14e-7	3.75e-6

guishable from random noise. These genes do not belong to any pattern group in the dataset [25].

To assess the performance of PUMA-CLUST on this kind of dataset, we add a group of random noise genes into the previously simulated datasets. The first generating step of the gene expression level for group seven is

$$x_{qij} = A_{qi}, \quad (4)$$

where A_{qi} is sampled from $U(0,14)$. The following steps of the simulation are the same as those for the former six groups. Three sets of simulated datasets with 10 randomly generated datasets for each set are also sampled and the average adjusted Rand index for three cases with 10, 20, and 30 conditions are shown in the second column of Figure 3. The number of groups for both MCLUST and PUMA-CLUST is assigned to seven. From the three plots it can be seen that the performance of the clustering from both PUMA-CLUST and MCLUST decreases with the inclusion of the noise group, but PUMA-CLUST still outperforms MCLUST over all three noise levels with the three different data dimensions. The *p*-values in Table 1 indicate that the improvement is statistically significant.

Testing the robustness to misspecified technical variance

The probe-level variance in the simulated datasets generated above is sampled from multi-mgMOS results from the real mouse dataset. When applying PUMA-CLUST it was assumed that the level of noise is known, but in practice it would be estimated using multi-mgMOS. We would like to test robustness to errors in estimating the measurement error variance. We therefore add some noise to the sampled standard deviation, $\hat{\sigma}_{qij}$, to simulate the error made in estimating this quantity. For the six-group and seven-group datasets, three kinds of random noise are added by sampling from $\mathcal{N}$ (0, 0.01), $\mathcal{N}$ (0, 0.1) and $\mathcal{N}$ (0, 0.2). The scatter plots of the error-added standard deviation against the simulated gene expression are shown in Figure 2(b)–(d). Figure 3 gives the average adjusted Rand index of the clustering results from PUMA-CLUST on the error-added standard deviation for various cases. In the case of PC.01, the added noise is quite small so that the clustering results of PC.01 are very close to the clustering results on the original simulated data. As the added noise variance increases, the performance of PUMA-CLUST decreases. The *p*-values in Table 1 mostly increase when larger noise is added to the variances but all *p*-values remain small and demonstrate a significant improvement for PUMA-CLUST over MCLUST. These results demonstrate that clustering is most accurate when the measurement error variance is known, but that the method is robust to errors in the estimate of the measurement error.

Clustering on a real mouse time-course dataset

The improved performance of the new model, PUMA-CLUST, over the standard Gaussian mixture model on simulated datasets was shown in the previous section. Here, we evaluate the performance of PUMA-CLUST on a real mouse dataset showing periodic behavior [12] by comparing with the results of the standard mixture model implemented in MCLUST.

This time-course dataset profiles the gene expression changes during the hair growth cycle, which is synchronised for the first two cycles following birth. After two cycles the hair growth cycle becomes progressively unsynchronised. Lin et al. use Affymetrix MG-U74Av2 microarray chips to profile mRNA expression in mouse back skin from eight representative time points in order to discover regulators in hair-follicle morphogenesis and cycling [12]. The microarray dataset utilised a total of 25 chips with each time point consisting of three or four replicates. The first five time points (day 1, 6, 14, 17 and 23) cover the first synchronised cycle and the last three time points (week 9, month 5 and year 1) belong to the asynchronous cycles. They identified 2,461 potential hair cycle-associated genes using a *F* test comparing synchronous and asynchronous time points. This dataset is available at [29].

We apply both PUMA-CLUST and MCLUST clustering over the first five time points which belong to the synchronised cycle and includes 15 chips. For MCLUST the raw mouse dataset is processed using the popular probe-level method GCRMA [30]. For PUMA-CLUST the raw data is processed by multi-mgMOS. We also applied MCLUST to MAS5.0 and multi-mgMOS gene expression measurements and the performance was found to be similar to the results presented here using GCRMA.

The clustering is performed on the 2,461 potential hair cycle-associated genes. The obtained expression level for each probe-set from both probe-level methods are normalised to have zero mean and standard deviation one. The Bayesian Information Criterion (BIC [31]) is used to determine the number of clusters. The calculated BIC for various numbers of clusters is shown in Figure 4. It can be seen that the optimal BIC for PUMA-CLUST is obtained at K = 22 and the optimal BIC for MCLUST is obtained at K = 30. In both cases, MCLUST converges to the model having the same full rank covariance matrix for each component (the 'EEE' model [32]). In order to make the different clustering methods comparable, the number of clusters for each method should be the same. Therefore, the 22-cluster and the 30-cluster cases are compared separately. The 22 clusters obtained from PUMA-CLUST and MCLUST are shown in Figure 5 and Figure 6 respectively, and the 30 clusters obtained are shown in Figure 7 and Figure 8, respectively. For visualisation, the average expression level at each time point over replicates is shown for both the gene profile and the cluster center.

To assess whether biologically relevant clusters are created using the two methods, we systematically performed GO annotation enrichment analysis for the individual clusters using DAVID 2006 (The Database for Annotation, Visualization and Integrated Discovery, [33]). The GO enrichment analysis allows the direct assessment of the biological significance for gene clusters found based on the enrichment of genes belonging to a specific GO functional category. The enrichment calculation performed in DAVID is a modified Fisher Exact test. The resulting p-value shows the biological significance for gene clusters. Based on our experience, GO Biological Process term level 5 gives more precise category definitions which are useful in further biological interpretations. Therefore, a meaningful GO enrichment analysis is to examine enriched categories of GO Biological Process at term level 5 and to select an enrichment cutoff at a conventional p-value of 0.05.

We found that for the 22-cluster results from the two methods PUMA-CLUST produced more clusters (21 of 22) with at least one enriched GO category in comparison to MCLUST (17 of 22), as shown in Figure 9(a). A visual inspection of these MCLUST clusters without an enriched GO category indicates that four out of five of these clusters (Cluster #1,6,8,15 in Figure 6) contain heterogeneous temporal expression profiles (i.e. not tightly clustered). Since the number of enriched GO categories found varies greatly among clusters (shown in Figure 10(a)), the average number (13.1) of enriched categories among the 22 PUMA-CLUST clusters is only slightly greater than the average among the MCLUST clusters (11.5). A more meaningful indicator of the distribution differences is the median number of enriched categories in PUMA-CLUST clusters (14) and MCLUST clusters (7). The same enrichment analysis method was repeated using the 30 clusters for both methods, and the results still clearly indicate that the PUMA-CLUST method results in more biologically meaningful clusters than the MCLUST method. Using 30 clusters, all clusters generated by PUMA-CLUST have at least one enriched GO category, in comparison to only 21 out of 30 clusters created by MCLUST as shown in Figure 9(b). The median number of enriched categories for PUMA-CLUST and MCLUST are 7 and 3, respectively, as shown in Figure 10(b). Based on these GO enrichment analyses, it is evident that PUMA-CLUST generated more biologically relevant clusters than MCLUST.

For further validation of the performance of PUMA-CLUST, we also applied MCLUST on multi-mgMOS measurements so that we can compare PUMA-CLUST with MCLUST using exactly the same probe-level summary method. MAS 5.0 is another popular probe-level method and therefore we also applied MCLUST to MAS 5.0 processed data for comparison. Enrichment analyses on the 22-cluster results for all four approaches (Figure 11 and Figure 12) show that MCLUST on multi-mgMOS processed data performed similarly to MCLUST on GCRMA processed data. Both have five clusters without any enriched category, but MCLUST with GCRMA had slightly higher median value for the number of enriched categories (7 vs. 5). Although MCLUST with MAS5.0 only had two clusters without any enriched category, its median value for the number of enriched categories is notably less than that of PUMA-CLUST with multi-mgMOS (5.5 vs. 14). Thus, PUMA-CLUST with multi-mgMOS still performs best in comparison to MCLUST using the three different expression summary methods. For 30-cluster results and for results with other numbers of clusters we found similar results. In particular, when the same probe-level method, multi-mgMOS, is used, PUMA-CLUST always outperforms MCLUST. The improved performance is due to the inclusion of the probe-level measurement

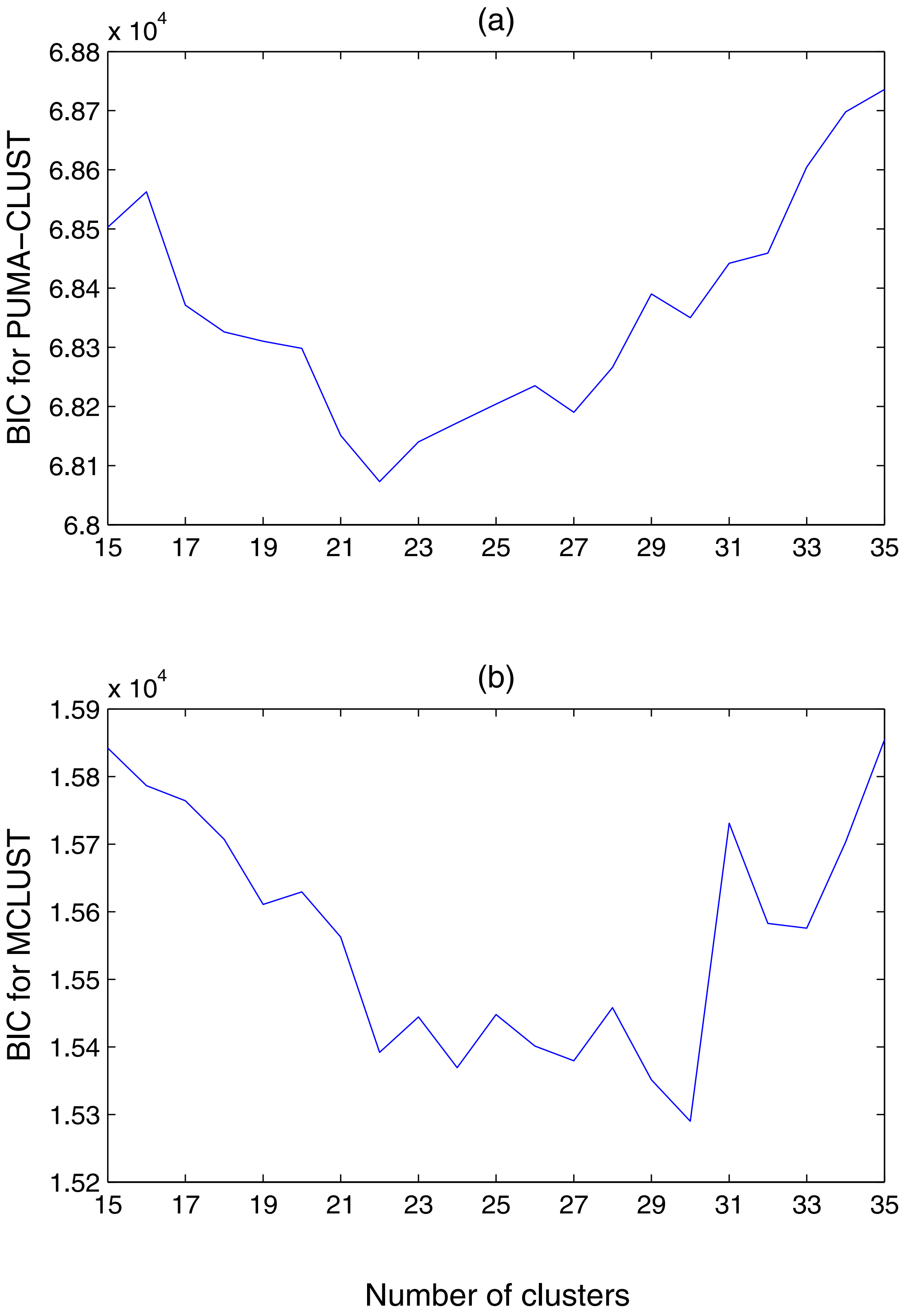

Figure 4

BIC for PUMA-CLUST and MCLUST. BIC for (a) PUMA-CLUST and (b) MCLUST against the number of mixture components on the 2,461 potential hair growth-associated genes from the mouse time-course dataset. PUMA-CLUST obtains the minimum BIC at K = 22 and MCLUST obtains the minimum at K = 30.

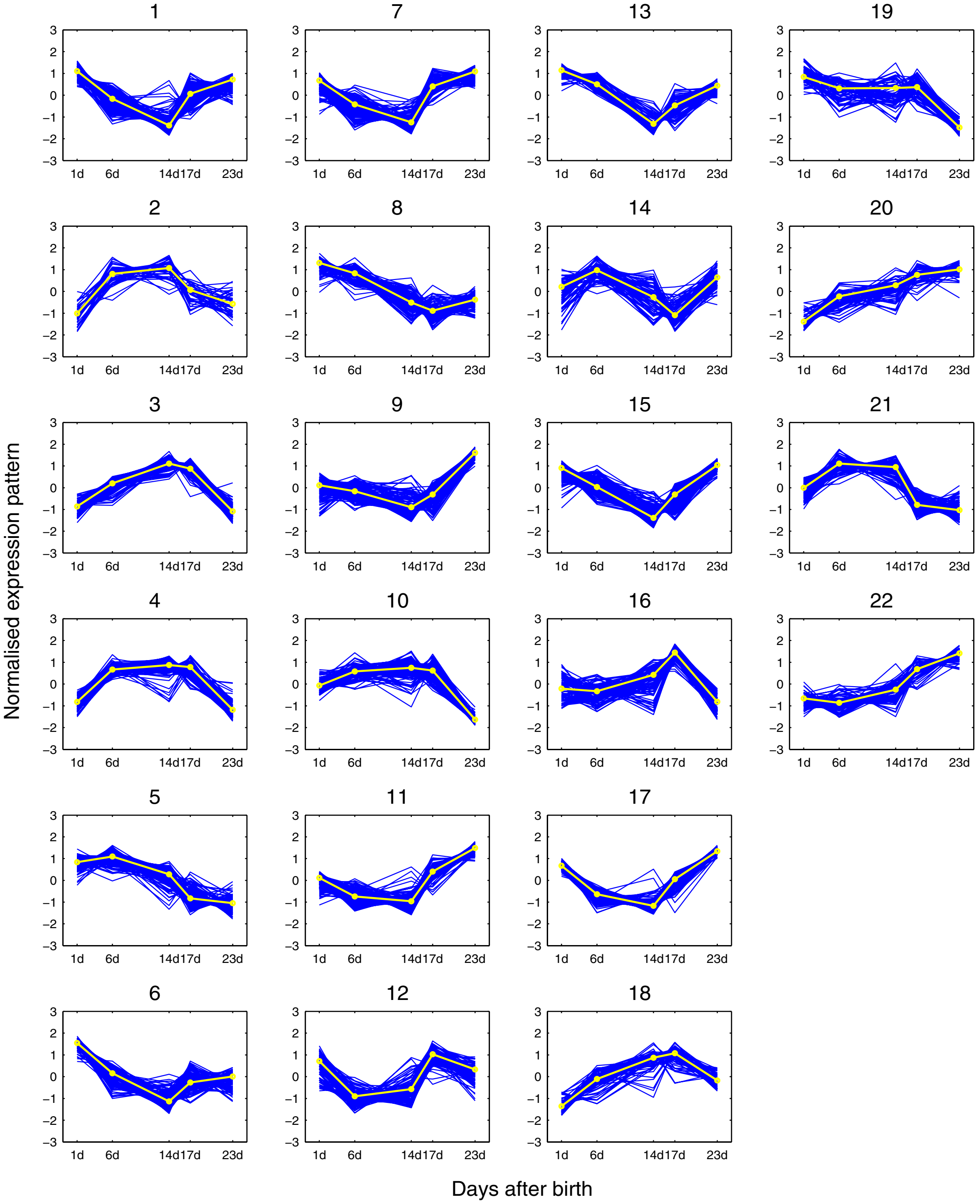

Figure 5

Expression pattern clusters from PUMA-CLUST when K = 22. The clusters are for the 2,461 potential hair cycle-associated genes of the mouse time-course dataset when K = 22. The expression pattern for each probe-set is shown as dark lines for five time points. The light line on each plot is the clustering center for each group. At each time point, the expression value is the average of the three replicated measurements.

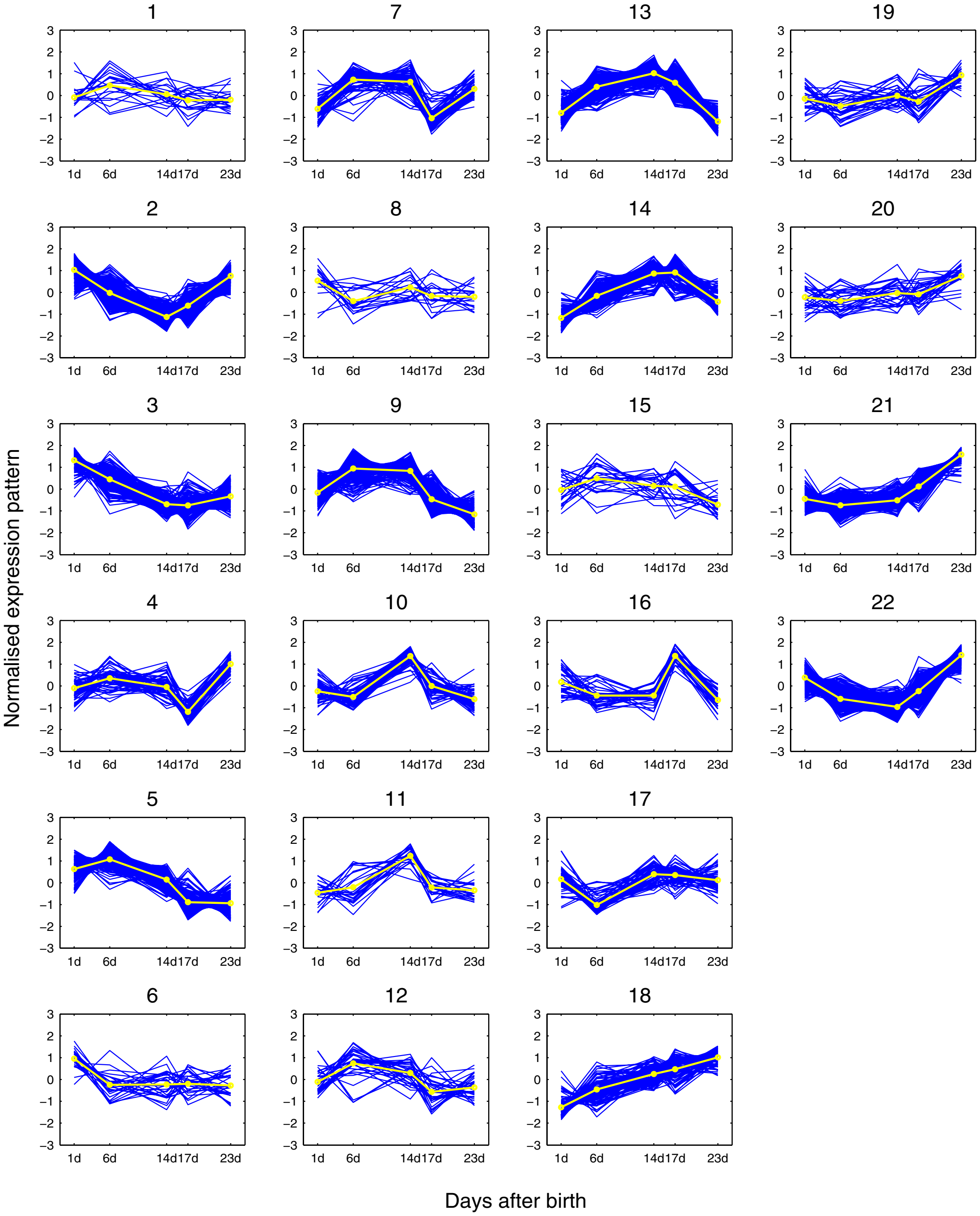

Figure 6

Expression pattern clusters from MCLUST when K = 22. The clusters are for the 2,461 potential hair cycle-associated genes of the mouse time-course dataset when K = 22. The expression pattern for each probe-set is shown as dark lines for five time points. The light line on each plot is the clustering center for each group. At each time point, the expression value is the average of the three replicated measurements.

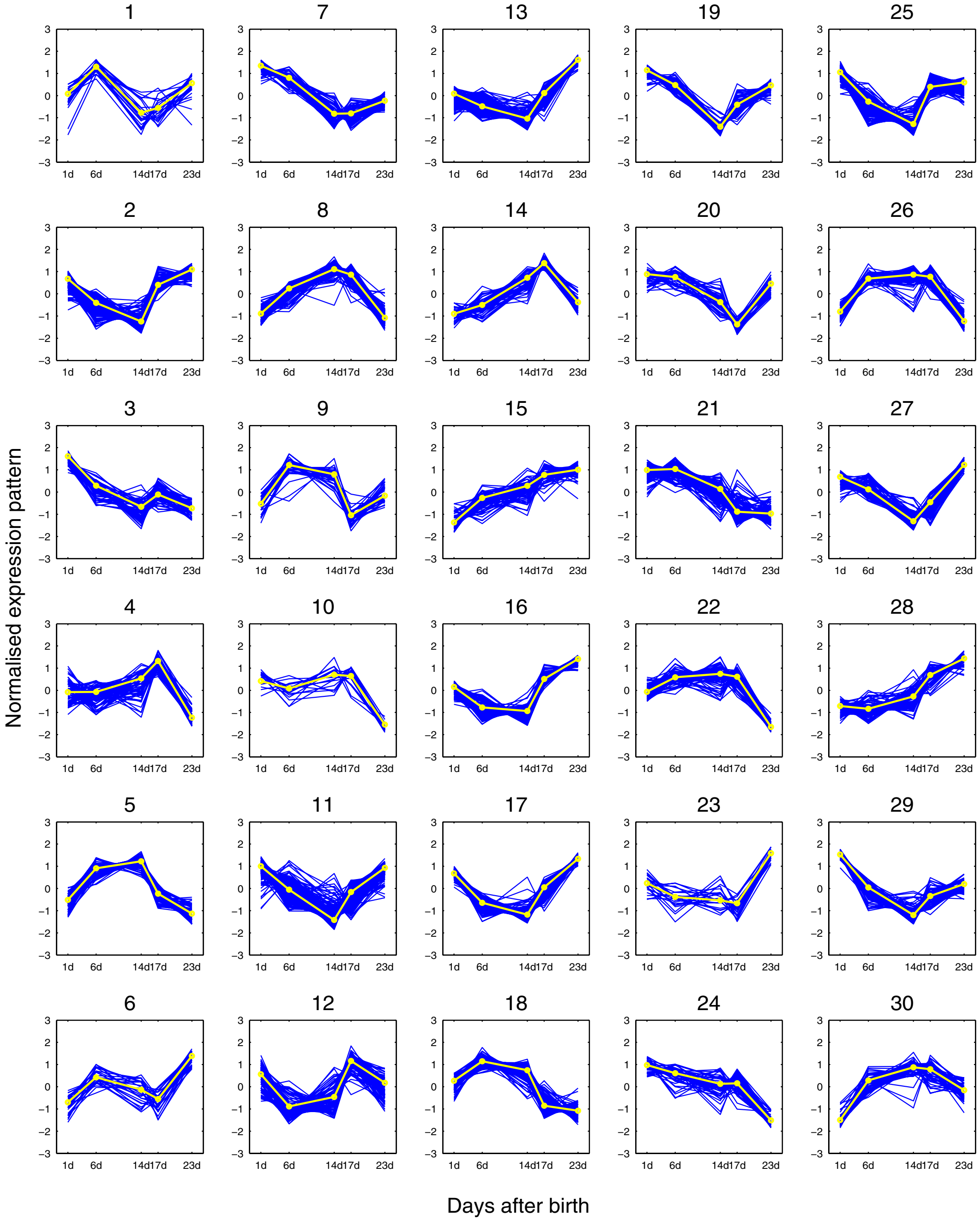

Figure 7

Expression pattern clusters from PUMA-CLUST when K = 30. The clusters are for the 2,461 potential hair-growth-associated genes of the mouse time-course dataset when K = 30. The expression pattern for each probe-set is shown as dark lines for five time points. The light line on each plot is the clustering center for each group. At each time point, the expression value is the average of the three replicated measurements.

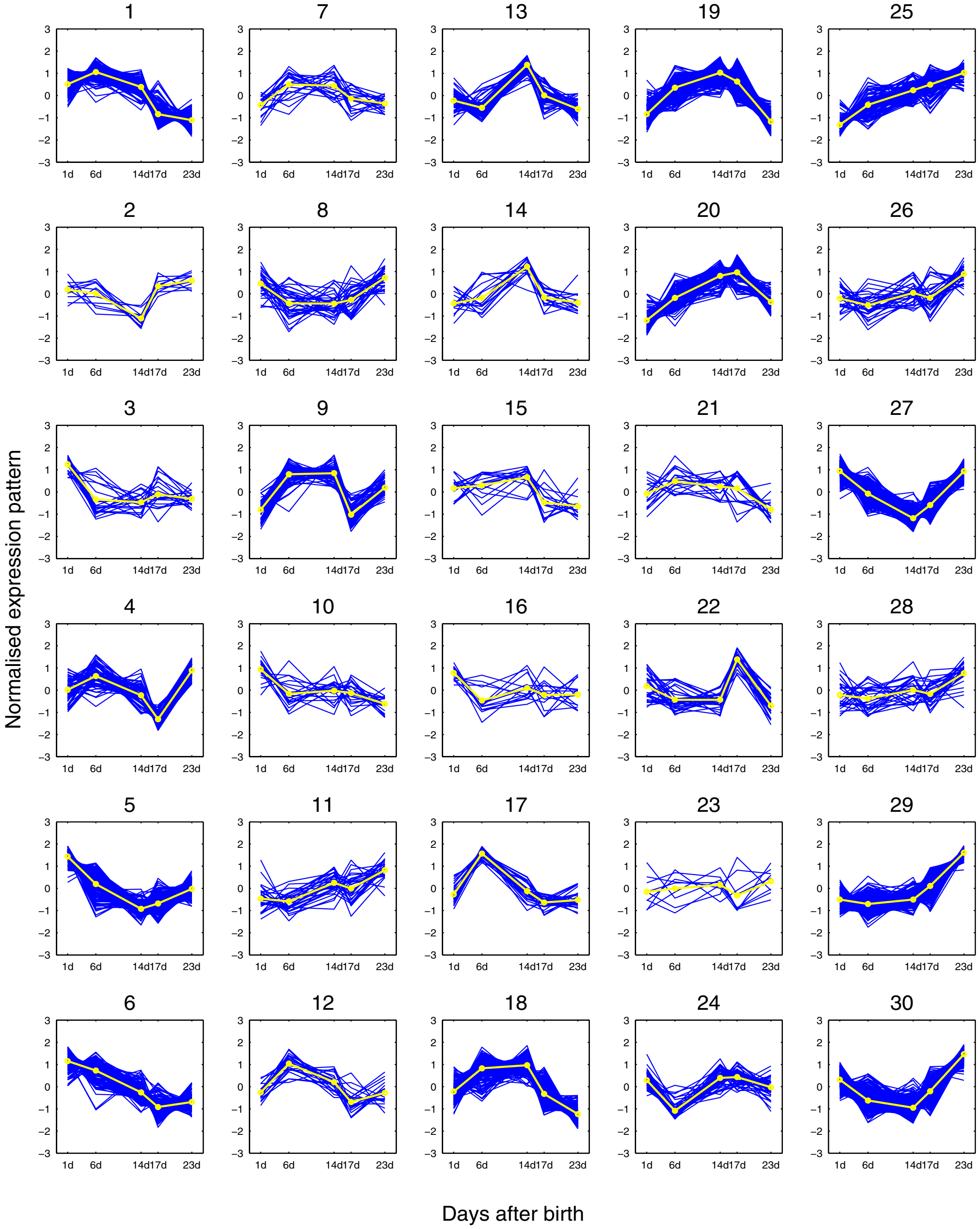

Figure 8

Expression pattern clusters from MCLUST when K = 30. The clusters are for the 2,461 potential hair-growth-associated genes of the mouse time-course dataset when K = 30. The expression pattern for each probe-set is shown as dark lines for five time points. The light line on each plot is the clustering center for each group. At each time point, the expression value is the average of the three replicated measurements.

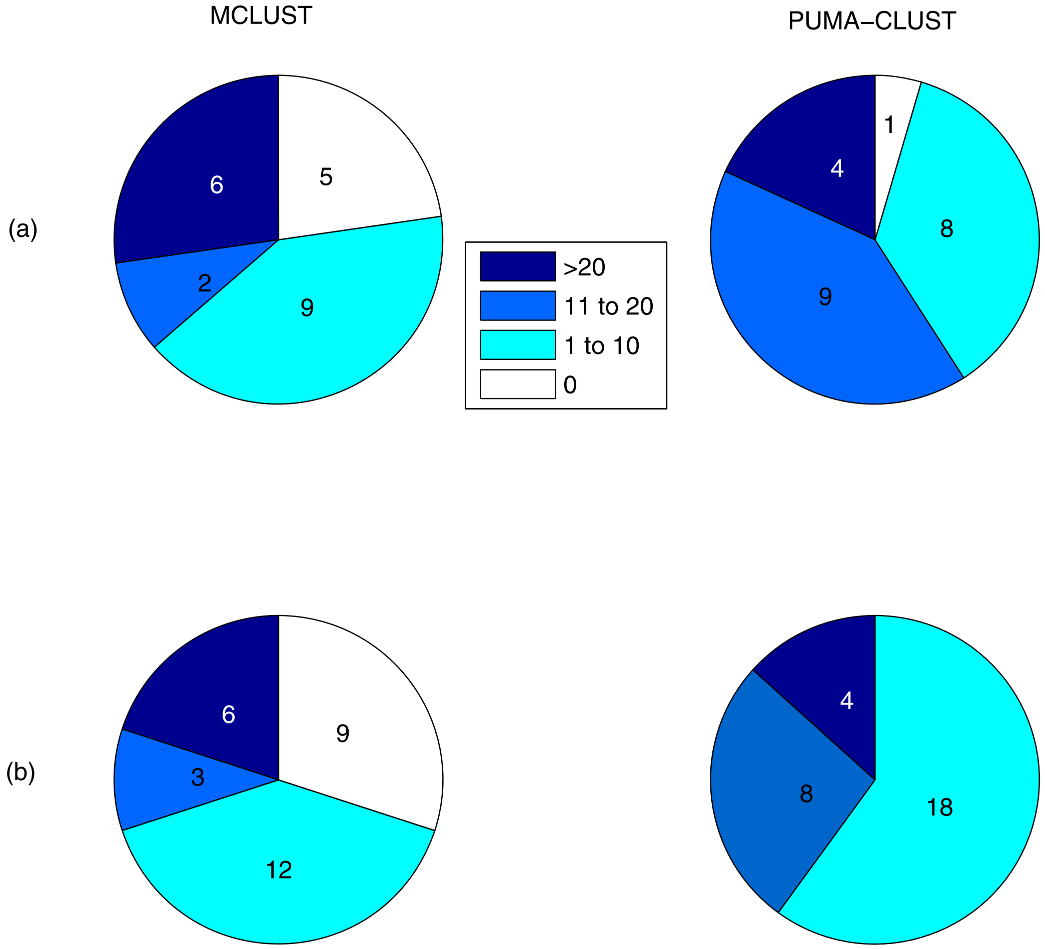

Figure 9

Comparison of the number of clusters found with the indicated ranges of enriched GO categories for MCLUST and PUMA-CLUST clusters. Comparison of the number of clusters found with the indicated ranges of enriched categories for MCLUST and PUMA-CLUST clusters using (a) 22 clusters and (b) 30 clusters. For both comparisons, the enriched categories were found using GO Biological Process term level 5, enrichment cutoff at p-value of 0.05, and mouse (*Mus Musculus*) as the population background.

error which down-weights the effect of the noisy low expressed genes.

Conclusion

In this paper we demonstrate the usefulness of the measurement error in model-based clustering of gene expression data. A standard Gaussian mixture model with an unequal volume spherical covariance matrix is augmented to incorporate probe-level measurement error obtained from Affymetrix microarrays. Results from simulated datasets and a real mouse time-course dataset show that the inclusion of probe-level measurement error results in improved and more biologically meaningful clustering of gene expression data. The augmented clustering model has been implemented in an R package, *pumaclust*, for public use of the method.

The improved performance of the augmented model has been shown in this paper. It is possible that further improvement can also be made by considering the repli-

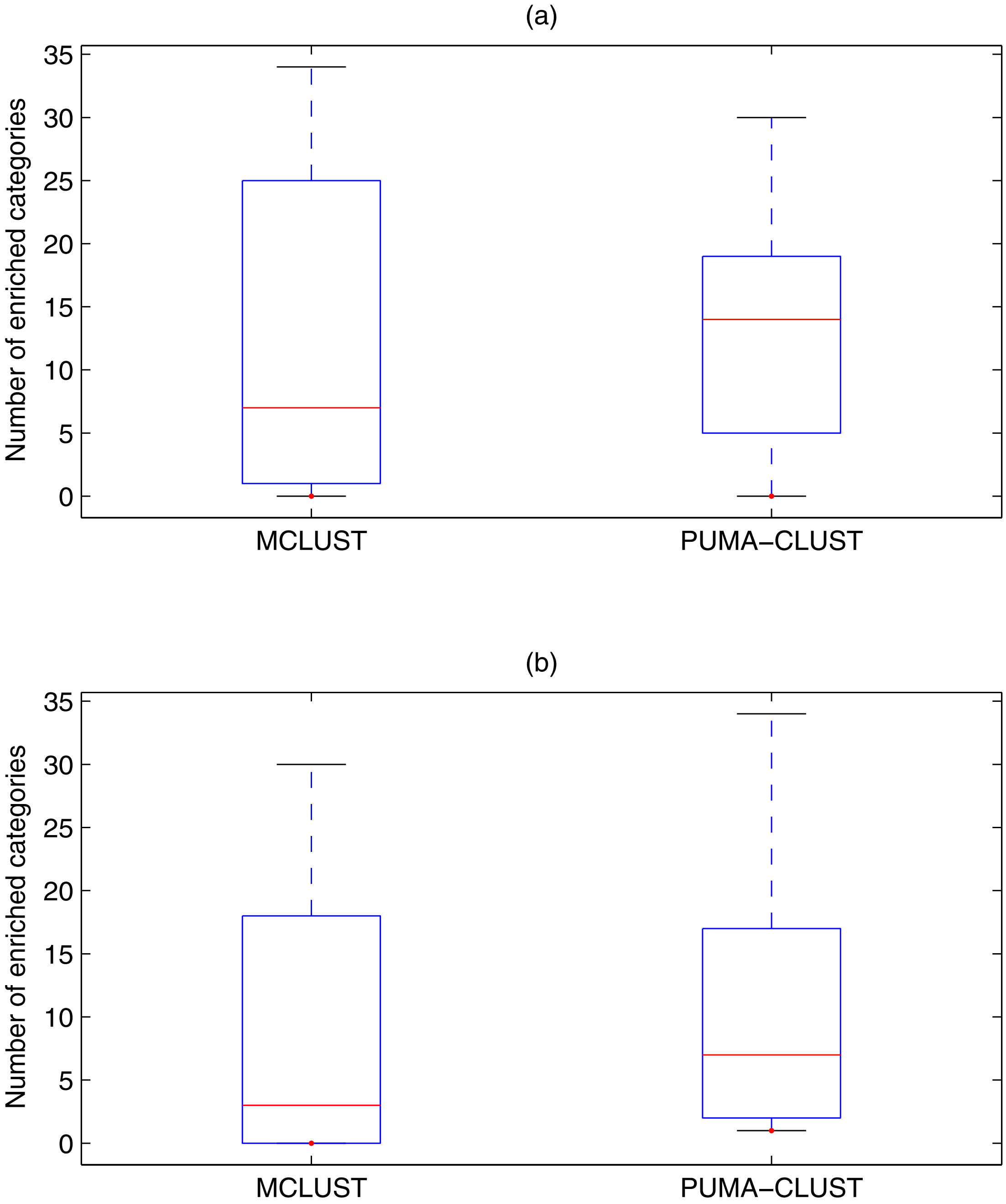

Figure 10

Boxplot of the number of enriched categories for MCLUST and PUMA-CLUST clusters. Boxplot of the number of enriched categories for MCLUST and PUMA-CLUST clusters using (a) 22 clusters and (b) 30 clusters. The boxes show the lower quartile, median, and upper quartile values. The dotted lines show the extent of the rest of the data. The number of enriched categories for MCLUST has larger variance than that for PUMA-CLUST.

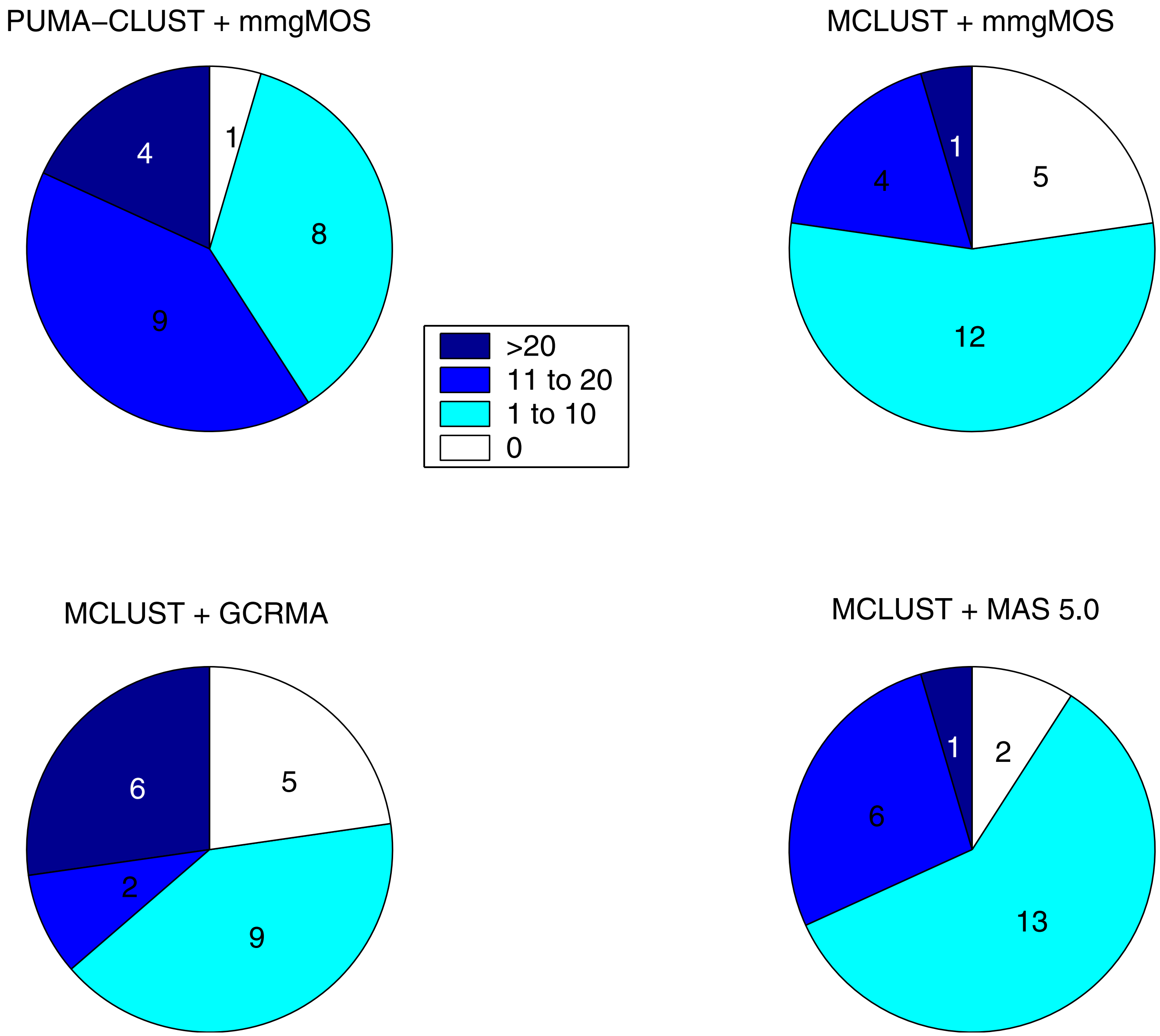

Figure 11

Comparison of the number of clusters found with the indicated ranges of enriched GO categories for MCLUST and PUMA-CLUST clusters using various probe-level methods. Comparison of the number of clusters found with the indicated ranges of enriched categories for MCLUST and PUMA-CLUST clusters using various probe-level methods when K = 22. For all comparisons, the enriched categories were found using GO Biological Process term level 5, enrichment cutoff at p-value of 0.05, and mouse (*Mus Musculus*) as the population background.

cate information where repeated measurements are available for time points. Clustering on repeated measurements has been considered by [12,23,25], but all of these approaches do not include the probe-level measurement error. Including both probe-level noise and replicate information in the clustering would be a useful extension of our work.

Methods

multi-mgMOS and probe-level measurement error

Affymetrix microarrays use multiple probe-pairs called a probe-set to measure the expression level for each gene. Each probe-pair consists of a perfect match (PM) probe and a mismatch (MM) probe. By design, the intensity of the PM probe measures the specific hybridisation of the target and the MM probe measures the non-specific

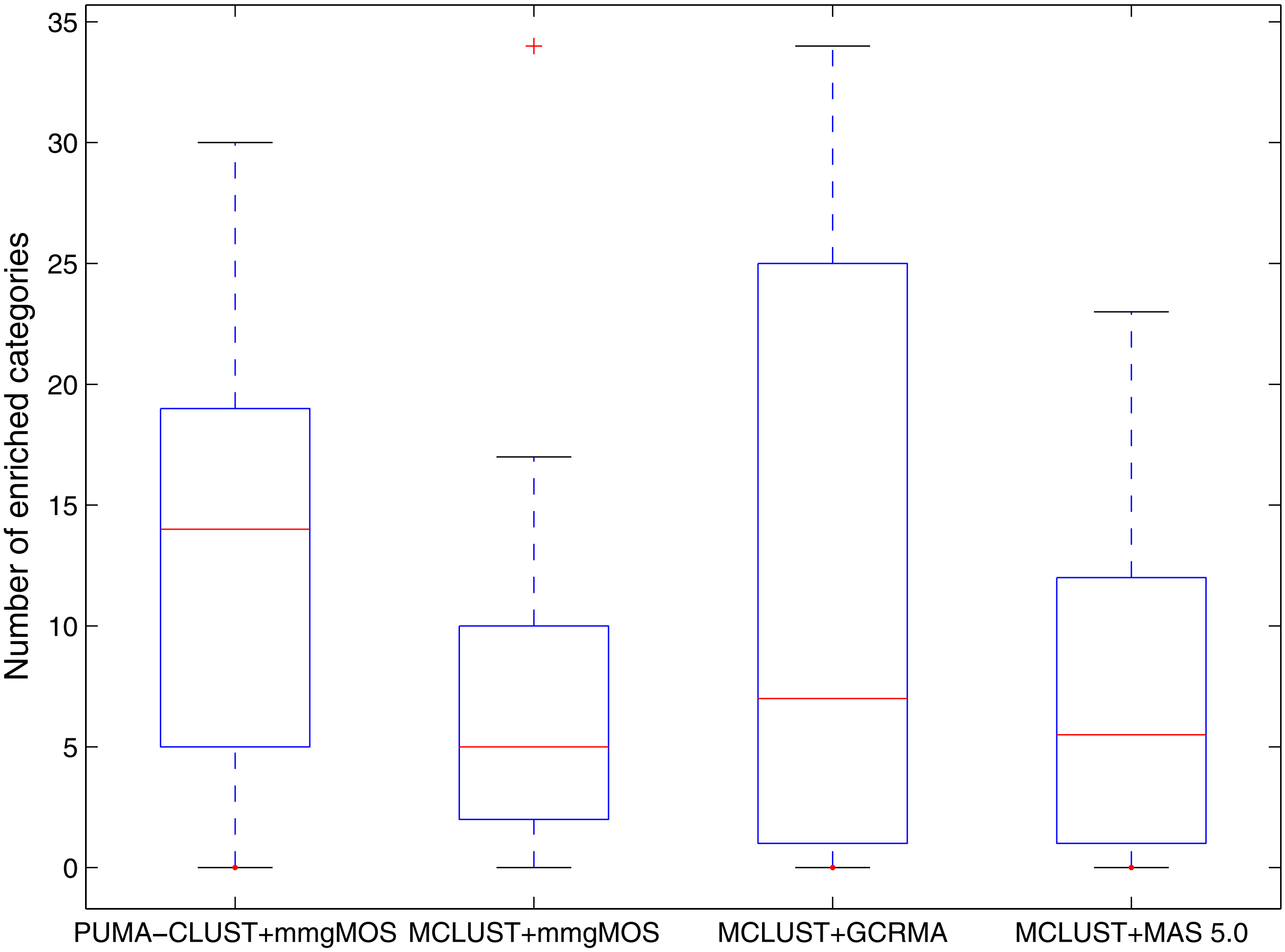

Figure 12

Boxplot of the number of enriched categories for MCLUST and PUMA-CLUST clusters using various probe-level methods. Boxplot of the number of enriched categories for MCLUST and PUMA-CLUST clusters using various probe-level methods when K = 22. The boxes show the lower quartile, median, and upper quartile values. The dotted lines show the extent of the rest of the data.

hybridisation associated to its corresponding PM probe. The microarray experimental data show that the intensities of both PM and MM probes vary in a probe-specific way and MM probes also detect some specific hybridisation. Based on these observations, multi-mgMOS [18] assumes the intensities of PM and MM probes for a probe-set both follow gamma distributions with parameters accounting for specific and non-specific hybridisation, and probe-specific effects. Let $y_{ijc}$and m_{ijc} represent the *j*th PM and MM intensities respectively for the *i*th probe-set under the *c*th condition. The model is defined by

$$y_{ijc} \sim \mathrm{Ga}(a_{ic} + \alpha_{ic}, b_{ij})$$

$$m_{ijc} \sim \mathrm{Ga}\ (a_{ic} + \phi\alpha_{ic}, b_{ij}) \quad (5)$$

$$b_{ij} \sim \mathrm{Ga}(c_i, d_i),$$

where Ga represents the gamma distribution. The parameter a_{ic} accounts for the background and non-specific hybridisation associated with the probe-set and α_{ic} accounts for the specific hybridisation measured by the probe-set. The parameter b_{ij} is a latent variable which models probe-specific effects. The Maximum a Posteriori (MAP) solution of this model can be found by efficient numerical optimisation. The posterior distribution of the logged gene expression level can then be estimated from the model and approximated by a Gaussian distribution with a mean, $\hat{x}_{ic}$, and a variance, v_{ic}. The mean of this distribution is taken as the estimated gene expression for gene i under the condition c and the variance can be considered the measurement error associated with this estimate. The Gaussian approximation to the posterior distribution is useful for propagating the probe-level measurement error in subsequent downstream analyses.

Mixture model

The mixture model is a useful tool for revealing the inherent structure of data. In a mixture model with K components, the data is generated by

$$p(\boldsymbol{x}_i) = \sum_{k=1}^{K} P(k) p(\boldsymbol{x}_i \mid k; \boldsymbol{\theta}_k), \qquad (6)$$

where $P(k)$ denotes the probability of selecting the kth component with parameters θ_k and $\theta = \{\theta_1, \theta_2, ..., \theta_K, P(k)\}$ is the complete parameter set of the mixture model. The parameters k ar latent variables determining which cluster the data belongs to.

Mixture models are usually solved by maximum likelihood using an Expectation-Maximisation (EM) algorithm [34]. With the initialised parameters at $t = 0$, the values of parameters can be determined iteratively through an E-step and M-step:

- E-step: Compute

$$P^t(k|\boldsymbol{x}_i) = P(k|x_i; \theta) \qquad (7)$$

for each data point $\boldsymbol{x}_i$ and each component k.

- M-step:

$$\boldsymbol{\theta}^{t+1} = \arg\max_{\boldsymbol{\theta}} \sum_i \sum_k P^t(k \mid \boldsymbol{x}_i) \log(p(\boldsymbol{x}_i \mid k; \boldsymbol{\theta}_k) P(k)) \qquad (8)$$

with constraint $\sum_k P(k) = 1$.

Standard Gaussian mixture model

For mixture component distributions from the exponential family, like the Gaussian, both steps are exactly tractable. In a Gaussian mixture model where $\theta_k = \{\mu_k, \Sigma_k\}$, each component k is modelled by a Gaussian distribution with mean μ_k and covariance matrix Σ_k,

$$\begin{aligned} p(x_i \mid k; \boldsymbol{\theta}_k) &= \mathcal{N}(x_i \mid \boldsymbol{\mu}_k, \Sigma_k) \\ &= \frac{1}{\sqrt{(2\pi)^p |\Sigma_k|}} \exp\left(-\frac{1}{2}(x_i - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (x_i - \boldsymbol{\mu}_k \right), \end{aligned} \qquad (9)$$

where $|\cdot|$ denotes determinant and p is the dimension of the data. As well as changing the number of components in the mixture, the covariance matrix Σ_k can be constrained to determine the flexibility of the model. The most constrained model is parameterised by $\Sigma_k = \sigma^2 I$ with only one free parameter in the covariance matrix for all components. The unconstrained model has full rank Σ_k with $p(p + 1)/2$ free parameters in the covariance matrix for each component where p is the data dimension. All representations of the covariance matrix are explored in [35]. Allowing the number of free parameters in the covariance matrix to vary leads to various models accommodating varying characteristics of data. All of these models have been implemented in MCLUST [20] and the BIC model selection criterion (described later) is used to select the most appropriate model.

Including measurement uncertainty in a Gaussian mixture model

From a probabilistic probe-level model, such as multi-mgMOS, for each data point one can obtain the measurement error, ν_i, which is a vector giving the variance of the measured expression level on each chip. Suppose $\boldsymbol{x}_i$ is the true expression level for data point i. The kth component of the Gaussian mixture model is modelled by $p(\boldsymbol{x}_i|k; \theta_k) = \mathcal{N}(\boldsymbol{x}_i|\mu_k, \Sigma_k)$. The measured expression level $\hat{\boldsymbol{x}}_i$ can be expressed as $\hat{\boldsymbol{x}}_i = \boldsymbol{x}_i + \varepsilon_i$. A zero-mean Gaussian measurement noise is assumed, $\varepsilon_i \sim \mathcal{N}(0, \text{diag}(\nu_i))$, where $\text{diag}(\nu_i)$ represents the diagonal matrix whose diagonal entries starting in the upper left corner are the elements of ν_i. Since $\hat{\boldsymbol{x}}_i$ is a linear sum of $\boldsymbol{x}_i$ and ε_i, the kth Gaussian component can be augmented as

$$p(\hat{\boldsymbol{x}}_i|k; \theta_k) = \mathcal{N}(\hat{\boldsymbol{x}}_i|\mu_k, \Sigma_k + \text{diag}(\nu_i)) \qquad (10)$$

We therefore augment the mixture model to account for the measurement error of each data point,

$$p(\boldsymbol{x}_i) = \sum_{k=1}^{K} P(k) \mathcal{N}(\boldsymbol{x}_i \mid k; \boldsymbol{\mu}_k, \Sigma_k + \text{diag}(\boldsymbol{\nu}_i)). \qquad (11)$$

Ideally, the covariance matrix should be of full rank to obtain the largest flexibility of the model. However, this will increase the complexity of the model. Since in (11) the additive measurement error $\text{diag}(\nu_i)$ accounts for inherent variability in the data, especially for extremely noisy gene expression data, the unequal volume spherical model (VI) described in [10] with the covariance $\Sigma_k = \sigma_k^2 I$ is adopted. This model allows the spherical components to have different variances which accounts for the variability within different gene function groups. Therefore, in this model the gene-specific variance ν_i is known and obtained from a probabilistic probe-level analysis model and the function-specific variance σ_k^2 is to be estimated from the mixture model via the EM algorithm. The parameters are denoted $\theta_k = \{\mu_k, \sigma_k^2\}$ for Gaussian component k and $\theta = \{\theta_1, \theta_2,..., \theta_k\}$ for all components, where K is the number of components. Using the K-means algorithm, one can obtain the initial parameters θ^0 for all components. Equal probability of the component prior is also assumed for the initial value of $P(k)$, $P^0(k)$. At the E-step, for each data point x_i the posterior probability of belonging to component k is calculated as,

$$\begin{aligned} P^t(k \mid \boldsymbol{x}_i) &= P^t(k \mid \boldsymbol{x}_i; \boldsymbol{\theta}^{t-1}) \\ &= \frac{P(\boldsymbol{x}_i \mid \boldsymbol{\theta}_k^{t-1}) P^{t-1}(k)}{\sum_k P(\boldsymbol{x}_i \mid \boldsymbol{\theta}_k^{t-1}) P^{t-1}(k)}. \end{aligned} \qquad (12)$$

At the M-step, the component prior and the parameters of components are optimised,

$$P^t(k) = \frac{1}{N} \sum_{n=1}^{N} P^t(k \mid \boldsymbol{x}_i) \qquad (13)$$

$$\boldsymbol{\theta}^t = \arg\max_{\boldsymbol{\theta}} \sum_i \sum_k P^t(k \mid \boldsymbol{x}_i) \log(p(\boldsymbol{x}_i \mid \boldsymbol{\theta}_k) P^t(k)). \qquad (14)$$

Equation (14) cannot be solved analytically due to the incorporation of v_i in the variance terms. However, with fast optimisation methods available such as SNOPT [36] and donlp2 [37], it is easy to calculate the optimal parameters numerically at the M-step. In our R implementation, *pumaclust*, we use donlp2.

Model selection

In the previous section the covariance matrix of the Gaussian mixture model is specified and the parameters are worked out via an EM algorithm for a given K. In practice the most appropriate number of clusters should also be determined. In mixture models, the Bayesian Information Criterion (BIC [31]) is usually used to decide the appropriate number of clusters. For model m with the number of clusters K, the calculation of BIC is

$$\mathrm{BIC}_m = -2\log p(D \mid \hat{\boldsymbol{\theta}}_m) + d_m \log(n), \qquad (15)$$

where D is the dataset, d_m is the number of free parameters to be estimated in model m, n is the number of genes and $\hat{\boldsymbol{\theta}}_m$ are the estimated maximum likelihood parameters obtained by the EM algorithm. For the unequal volume spherical model (VI), the number of free parameters is $d_m = K(p + 2) - 1$. MCLUST also uses BIC to select the most appropriate class of covariance model.

Adjusted rand index

The adjusted Rand index gives a measure of agreement between clustering results. Given a set of n data points $D = \{x_1,..., x_n\}$, suppose $C^1 = \{c_1^1,..., c_M^1\}$ and $C^2 = \{c_1^2,..., c_N^2\}$ represent two different partitions of the data points in D. Assume that n_{ij} is the number of data points belonging to cluster c_i^1 and c_j^2, and $n_{i\cdot}$ and $n_{\cdot j}$ are the number of data points in cluster c_i^1 and c_j^2 respectively. The adjusted Rand index can be calculated by

$$\frac{\sum_{i,j}\binom{n_{ij}}{2} - \left[\sum_i \binom{n_{i\cdot}}{2} \sum_j \binom{n_{\cdot j}}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{n_{i\cdot}}{2} + \sum_j \binom{n_{\cdot j}}{2}\right] - \left[\sum_i \binom{n_{i\cdot}}{2} \sum_j \binom{n_{\cdot j}}{2}\right] / \binom{n}{2}}. \qquad (16)$$

Authors' contributions

XL developed and implemented the new model, applied the new and standard models to the simulated and real data, and drafted the manuscript. KL and AB provided the mouse dataset, helped with the evaluation of the clustering results on this dataset and revised the manuscript. MR supervised the study and helped with the manuscript preparation. All authors read and approved the final manuscript.

Acknowledgements

XL was funded by the Overseas Scholarship Scheme from the University of Manchester and a studentship from the School of Computer Science. BA was supported by NIH awards AR44882 and AR52863. MR was supported by a BBSRC award 'Improved processing of microarray data with probabilistic models'.

References

1. Schena M, Shalon D, Davis RW, Brown PO: **Quantitative monitoring of gene expression patterns with a complementary DNA microarray.** *Science* 1995, **270(5235):**467-470.
2. Lockhart DJ, Dong H, Byrne MC, Follettie MT, Gallo MV, Chee MS, Mittmann M, Wang C, Kobayashi M, Horton H, Brown EL: **Expression monitoring by hybridization to high-density oligonucleotide arrays.** *Nat Biotechnol* 1996, **14(13):**1675-1680.
3. Slonim DK: **From pattern to pathways: gene expression data analysis comes of age.** *Nature Genetics* 2002, **32(Suppl):**502-508.
4. Quackenbush J: **Computational Analysis of Microarray Data.** *Nature Reviews Genetics* 2001, **2:**418-427.
5. Eisen MB, Spellman PT, Brown PO, Botstein D: **Cluster analysis and display of genome-wide expression patterns.** *Proc Natl Acad Sci USA* 1998, **95:**14863-14868.
6. Tavazoie S, Hughes JD, Campbell MJ, Cho RJ, Church GM: **Systematic determination of genetic network architecture.** *Nat Genet* 1999, **22:**281-285.
7. Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E, Lander ES, Golub TR: **Interpreting patterns of gene expression withself-organizing maps: methods and application to hematopoietic differentiation.** *Proc Natl Acad Sci USA* 1999, **22:**2907-2912.

8. D'haeseleer P: **How does gene expression clustering work?** *Nature Biotechnology* 2005, **23:**1499-1501.
9. Fraley C, Raftery AE: **Model-based clustering, discriminant analysis and density estimation.** *J Am Stat Assoc* 2002, **97:**911-931.
10. Yeung KY, Fraley C, Murua A, Raftery AE, Ruzzo WL: **Model-based clustering and data transformations for gene expression data.** *Bioinformatics* 2001, **17(10):**977-987.
11. Siegmund KD, Laird PW, Laird-Offringa IA: **A comparison of cluster analysis methods using DNA methylation data.** *Bioinformatics* 2004, **20:**1896-1904.
12. Lin KK, Chudova D, Hatfield GW, Smyth P, Andersen B: **Identification of hair cycle-associated genes from time-course gene expression profile data by using replicate variance.** *Proceedings of the National Academy of Science USA* 2004, **101:**15955-15960.
13. Hein AMK, Richardson S, Causton HC, Ambler GK, Green PJ: **BGX: afully bayesian integrated approach to the analysis of Affymetrix GeneChip data.** *Biostatistics* 2005, **4:**249-264.
14. Li C, Wong WH: **Model-based analysis of oligonucleotide arrays: model validation, design issues and standard error application.** *Genome Biology* 2001, **2(8):**research0032.
15. Rattray M, Liu X, Sanguinetti G, Milo M, Lawrence N: **Propagating Uncertainty in Microarray Data Analysis.** *Briefings in Bioinformatics* 2006, **7:**37-47.
16. Sanguinetti G, Milo M, Rattray M, Lawrence ND: **Accounting for probe-level noise in principal component analysis of microarray data.** *Bioinformatics* 2005, **21:**3748-3754.
17. Liu X, Milo M, Lawrence ND, Rattray M: **Probe-level measurement error improves accuracy in detecting differential gene expression.** *Bioinformatics* 2006, **22:**2107-2113.
18. Liu X, Milo M, Lawrence ND, Rattray M: **A tractable probabilistic model for Affymetrix probe-level analysis across multiple chips.** *Bioinformatics* 2005, **21(18):**3637-3644.
19. **PUMA – Propagating Uncertainty in Microarray Analysis** [http://www.bioinf.manchester.ac.uk/resources/puma/]
20. Fraley C, Raftery AE: **Mclust: software for model-based cluster analysis.** *J Classification* 2002, **16:**297-306.
21. Milligan GW, Cooper MC: **A study of the comparability of external criteria for hierarchical cluster analysis.** *Multivariate Behavioral Research* 1986, **21:**441-458.
22. Hubert L, Arable P: **Comparing partitions.** *Journal of classification* 1985, **2:**193-218.
23. Yeung KY, Medvedovic M, Bumgarner RE: **Clustering gene-expression data with repeated measurements.** *Genome Biology* 2003, **4:**R34.
24. Bolshakova N, Azuaje F: **Cluster validation techniques for genome expression data.** *Signal Process* 2003, **83:**825-833.
25. Medvedovic M, Yeung KY, Bumgarner RE: **Bayesian mixture model based clustering of replicated microarray data.** *Bioinformatics* 2004, **20:**1222-1232.
26. Tu BP, Kudlicki A, Rowicka M, McKnight SL: **Logic of the yeast metabolic cycle: temporal compartmentalization of cellular processes.** *Science* 2005, **310:**1152-1158.
27. Affymetrix: *Statistical algorithms reference guide* Affymetrix Inc, Santa Clara CA; 2002.
28. Baldi P, Long AD: **A Baysian framework for the analysis of microarray expression data: regularized t-test and statistical infrence of gene changes.** *Bioinformatics* 2001, **17:**509-519.
29. **Gene Expression Omnibus, accession number GDS912** [http://www.ncbi.nlm.nih.gov/projects/geo/]
30. Wu Z, Irizarry RA, Gentleman R, Martinez-Murillo F, Spencer F: **A model-based background adjustment for oligonucleotide expression arrays.** *Journal of the American Statistical Association* 2004, **99(468):**909-917.
31. Schwartz G: **Estimating the dimension of a model.** *Ann Stat* 1978, **6:**461-464.
32. Fraley C, Raftery AE: **MCLUST: Software for Model-Based Clustering, Discriminant Analysis and Density Estimation.** In *Tech Rep 415R* Department of Statistics, University of Washington; 2002.
33. Dennis G Jr, Sherman BT, Hosack DA, Yang J, Gao W, Lane HC, Lempicki RA: **DAVID: database for annotation, visualization, and integrated discovery.** *Genome Biology* 2003, **4(5):**P3.
34. Dempster AP, Laird NM, Rubin DB: **Maximum likelihood from incomplete data via the EM algorithm.** *Journal of the Royal Statistical Society B* 1977, **39:**1-38.
35. Banfield JD, Raftery AE: **Model-based Gaussian and non-Gaussian clustering.** *Biometrics* 1993, **49:**803-821.
36. Gill PE, Murray W, Saunders MA: **SNOPT: an SQP algorithm for large-scale constrained optimization.** *SIAM Journal on Optimization* 2002, **12:**979-1006.
37. Spellucci PA: **A SQP method for general nonlinear programs using only equality constrained subproblems.** *Mathematical Programming* 1998, **82:**413-448.