Modelling time course gene expression data with finite mixtures of linear additive models

Bettina Grün¹, Theresa Scharf² and Friedrich Leisch³

¹Department of Applied Statistics, Johannes Kepler University Linz, Altenbergerstrasse 69, 4040 Linz, ²Austrian Centre of Industrial Biotechnology, Muthgasse 11, 1190 Wien and ³Institute of Applied Statistics and Computing, University of Natural Resources and Life Sciences Vienna, Gregor Mendel Strasse 33, 1180 Wien, Austria

ABSTRACT
Summary: A model class of finite mixtures of linear additive models is presented. The component-specific parameters in the regression models are estimated using regularized likelihood methods. The advantages of the regularization are that (i) the pre-specified maximum degrees of freedom for the splines is less crucial than for unregularized estimation and that (ii) for each component individually a suitable degree of freedom is selected in an automatic way. The performance is evaluated in a simulation study with artificial data as well as on a yeast cell cycle dataset of gene expression levels over time.

Availability: The latest release version of the R package flexmix is available from CRAN [http://cran.r-project.org].

Contact: bettina.gruen@jku.at

Received on September 8, 2011; revised on November 21, 2011; accepted on November 22, 2011

1 INTRODUCTION
Time-course microarray experiments make it possible to look at the gene expression of thousands of genes at several time points simultaneously. Clustering of gene expression patterns is, in general, used to identify common temporal or spatial expression patterns. Cluster results contribute to the regulatory network of gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes. In the literature, numerous methods for clustering gene expression, i.e. suggest functional pathways and interaction between genes.
cycle dataset from Speelman et al. (1999) is presented in Section 4. The article concludes with a summary and an outlook.

2 MODEL SPECIFICATION AND ESTIMATION

The mixture density \( h \) of a finite mixture model with \( K \) components is given for gene \( i \) by

\[
h(y_i|\theta, \pi) = \sum_{k=1}^{K} \pi_k f(y_i|\mu_k, \sigma_k^2)
\]

where \( Y_i = (y_{ij})_{j=1,...,n_i} \) is the response for gene \( i \), and \( \pi_k \) are the mixing probabilities. \( \mu_k \) and \( \sigma_k^2 \) are the mean and variance parameters for the mixture density \( f \). In the following, we assume that \( \mu_k \) and \( \sigma_k^2 \) are independent parameters. The component density function \( f \) is assumed to belong to the same parametric family for all components and differ only with respect to the mean parameter \( \mu_k \).

The penalized complete data log-likelihood is given by

\[
\log\mathbb{L}_p(\theta|X,Z) = \sum_{i=1}^{n} \log \left( \sum_{k=1}^{K} \prod_{j=1}^{n_i} f(y_{ij}|\mu_k, \sigma_k^2) \right) - \sum_{k=1}^{K} \lambda_k b_k b_k^\top
\]

where \( \lambda_k \) is the smoothing parameter for component \( k \), \( b_k \) is the vector of estimated effects for component \( k \), and \( \sigma_k^2 \) is the variance parameter. The smoothing parameter \( \lambda_k \) is a hyperparameter, this step is referred to as H-step: the hyperparameter \( \lambda_k \) is adapted. Since the penalized log-likelihood is linear in the unobserved component membership assignments, the penalized log-likelihood corresponds to the joint log-likelihood of the observations and the random effects. In a mixture of random effects models, the marginal log-likelihood after integrating out the random effects would be considered.

The penalized complete data log-likelihood given the data and the component membership assignments \( c \) is equal to

\[
\log\mathbb{L}_p(\theta|X,Z,c) = \sum_{i=1}^{n} \log \left( \sum_{k=1}^{K} \prod_{j=1}^{n_i} f(y_{ij}|\mu_k, \sigma_k^2) \right) - \sum_{k=1}^{K} \lambda_k b_k b_k^\top
\]

where \( c \) is the component membership assignment for gene \( i \). Each gene is assigned exactly to one component. The penalized complete data log-likelihood is linear in the unobserved component membership assignments and the expected component membership assignments are given by the \( a posteriori \) probabilities \( \hat{c} \).

E-step: \( \hat{c}_i = a_{i|\pi}(\theta) \) denotes the \( a posteriori \) probability for gene \( i \) to be from component \( k \) given the current parameter estimates \( \hat{\theta} \). It is determined by

\[
\hat{c}_i \propto \pi_k b_k^\top f(y_i|\mu_k, \sigma_k^2)
\]

where \( f(y_i|\mu_k, \sigma_k^2) = f(y_i|\mu_k, \sigma_k^2) \).

H-step: the hyperparameter \( \lambda_k \) is estimated in this step. For each component, the likelihoods of the observations weighted with the \( a posteriori \) probabilities are used. The smoothing parameter \( \lambda_k \) is estimated separately for each component by integrating out the penalized coefficients and maximizing the resulting likelihood with respect to \( \lambda_k \). In the Appendix A, it is shown that this is equivalent to maximizing the likelihood of a multivariate normal distribution with a suitable variance-covariance matrix.

M-step: the component sizes are determined separately from the component distribution-specific parameters. The determination of the component sizes are the same as for the normal EM algorithm. They are determined for each \( k \) by

\[
\hat{b}_k = \frac{1}{\sum_i \hat{c}_i}
\]

For the smoothing parameter \( \lambda_k \), the component distribution specific parameters \( b_k, h_k \) and \( \sigma_k^2 \) are...
increased or identical in each step conditional on
the log-likelihood is smaller than a pre-specified threshold. The relative
that a fixed point has been reached. The algorithm is stopped if either a
However, no change of the likelihood during the HEM algorithm indicates
an M-step. After modification of the EM algorithm, the likelihood is not
a posteriori
Hence it follows
Proposition 1. The penalized log-likelihood \( \log \Delta_p(\theta, Y, Z, \Lambda) \) is increased or identical in each step conditional on \( \Lambda \).
Proof. Assume that \( \theta^* \) is the new and \( \theta \) is the old parameter estimate. In the M-step, the following function is maximized with respect to \( \theta \).
\[
Q(\theta'|\theta, \Lambda) = \mathbb{E}[\log \Delta_p(\theta', Y, X, Z, \Lambda|\theta)] = \log \Delta_p(\theta', Y, X, Z, \Lambda) + H(\theta'|\theta, \Lambda)
\]
The Gibbs’ inequality implies that
\[
H(\theta'|\theta, \Lambda) \leq H(\theta|\theta, \Lambda) \quad \text{for all} \quad \theta, \theta'.
\]
Because of the maximization in the M-step it holds that
\[
Q(\theta'|\theta, \Lambda) \geq Q(\theta|\theta, \Lambda).
\]
Hence it follows
\[
\log \Delta_p(\theta'|Y, X, Z, \Lambda) \geq \log \Delta_p(\theta|Y, X, Z, \Lambda).
\]
The HEM algorithm is deterministic given a specific initialization, e.g.,
by providing a posteriori probabilities for the observations to start with an
M-step. After modification of the EM algorithm, the likelihood is not
maximized any more, but the penalized likelihood is maximized conditional
on the estimates of the smoothing parameters. The likelihood is not increased in each step, because stronger regularization might also induce a decrease. However, no change of the likelihood during the HEM algorithm indicates that a fixed point has been reached. The algorithm is stopped if either a maximum number of iterations has been performed or if the relative change of the log-likelihood is smaller than a pre-specified threshold. The relative change is determined using \( |L_{n-1} - L_n|/|L_n| + 0.1 \) where \( L_n \) denotes the log-likelihood at the \( n \)-th iteration. In the following, the maximum number of iterations is always set to 5000.
Especially for mixtures where the component distribution is a normal
distribution, problems might occur during the EM as well as the HEM
algorithm. In this case, the mixture likelihood is unbounded because infinite values emerge if the variance of one of the components is zero. To avoid estimation problems in the M-step due to very small components, component sizes where the number is smaller than a pre-specified proportion are
omitted during the HEM algorithm. The a posteriori probabilities are then re-calculated for the remaining components. In the following, this threshold is always set to 0.005.
For finite mixture models, multimodality of the likelihood is generally observed and the initialization strategy is crucial to determine a good estimate. Different initialization strategies were proposed for the EM
algorithm for finite mixture models. For an overview and a comparison of
different methods for mixtures of LMs and linear mixed models, see Schaal
et al. (2010). In the context of time-course gene expression data analysis,
Schaal et al. (2010) recommend the strategy of several short runs of the EM
algorithm with a liberal convergence criterion followed by a long run of the EM algorithm initialized in the best solution from the short runs, where convergence is determined with a strict criterion. This procedure was originally proposed for finite mixtures of multivariate Gaussian distributions in Biernacki et al. (2003).
Model selection needs to address the determination of (i) the number of components and (ii) the maximum number of degrees of freedom for
the components. In both cases, the BIC criterion is proposed. The BIC is
the general recommendation to determine the number of components for
model-based clustering (Fraley and Raftery, 2002) and mixtures of regression
models (Celeux et al. 2000; Biernacki et al. 2000). Other
model selection criteria such as the Akaike information criterion (AIC) are
also suggested. Information criteria have the advantage that they are easily
derived from the fitted model because essentially only the log-likelihood
needs to be evaluated as well as the effective degrees of freedom determined.
In addition, they select a suitable model according to a compromise between
model fit and model complexity.

3 SIMULATION STUDY
The performance of finite mixtures of LAMs with regularized estimation
is first evaluated on artificial datasets. These datasets are designed to resemble time-course gene expression patterns. The datasets are generated in the same way as in Schaal et al. (2010). The number of components is 15 plus an additional noise component of genes, the number of time points is 16 and the component sizes vary between 10 and 100 yielding a total of 6500 genes. The difficulty of the problem is varied with respect to three parameters and for each of these parameters three different levels are used. The standard
deviation (SD) of genes around the component centers is varied with values 0.1, 0.3 and 0.5, the number of noise genes is varied with values 0, 1, and 2. No individual differences between the genes within the same component are specified. For each experimental setting, 50 different datasets were generated. The finite mixtures of LAMs are estimated in a regularized way with a penalized regression spline as smooth for time. Thin plate regression splines are used and the maximum number of degrees of freedom is 15 with an additional
intercept.
The models are estimated using (i) initialization in the true
classification and (ii) 10 short runs randomly initialized followed by a long run of the HEM algorithm initialized in the best solution of the short runs. The short runs are terminated if the change in the relative log-likelihoods is smaller than 0.01 and the long run if the change is smaller than \( 10^{-5} \). The long run is initialized in the best solution of the short runs with respect to the log-likelihood. The number of components are set to 16 for random initialization.
The performance of the mixtures of LAMs with regularized estimation is evaluated by determining the Rand index corrected for chance (Hubert and Arabie, 1985) when comparing the true
classification to the induced classification by the fitted model. The
cluster performance of the model fitted with initialization in the true
classification is shown in Figure 1. The resulting classification is nearly
perfect if the SD of the genes around the component centers as well as the SD of the noise genes is small regardless of the number
of noise genes. Increasing the SD of the noise genes around the component centers while keeping the SD of the noise genes small leads to still
good cluster results, even though the performance is clearly worse if the number of noise genes is only low. This indicates that forming a separate component for the noise genes is more difficult if there are less noise genes. If the SD of noise genes is at least medium (i.e. differs stronger from the SD of the other genes), the performance is best if the number of noise genes is low and the SD of genes is also only small.

The Rand indices adjusted for chance derived by comparing the true classification to the classification induced by the models fitted using the random initialization strategy are given in Figure 2. The cluster performance is clearly worse than for the models fitted with initialization in the true solution. The performance is better for lower number of noise genes regardless of the values of the other parameters. Small SD of noise genes also improves the cluster performance, whereas the SD of the other genes hardly seems to make any difference.

The adjusted Rand index when comparing the true classification to the induced classification by the fitted model is considerably lower for the models fitted with the random initialization strategy than for the models fitted with the true classification for all datasets. Two explanations are possible: (i) the HEM algorithm was not able to detect the global optimum and (ii) the optimal solution according to the likelihood criterion differs from the true underlying model. The log-likelihood values of the models fitted using initialization in the true classification and using the random initialization strategy are compared to investigate the reason. The comparison indicates that if the SD of the genes around the component centers is only small, the initialization in the true classification leads to considerably better results. For medium and large SD of the genes, the results are reversed: the random initialization strategy leads, in general, to higher likelihood values with a stronger difference for medium than for large SD of the genes around component centers.

During the HEM algorithm, components that are <0.005 are dropped. For initialization in the true solution, the number of components retained are 16 components in 68% of the cases and 15 components (one dropped during the EM algorithm) in 15% of the cases for the fitted models. However, for the random initialization strategy the median number of components converged to is 12. For the different parameter settings, only small differences in the number of components retained are observed.

The results of the simulation study indicate that finite mixtures of LAMs with regularized estimation give very good results in the optimal situation where the classification is known and this performance deteriorates slightly for more difficult classification problems. Even in the situation where the classification is not known, the random initialization strategy gives reasonable results.

4 APPLICATION TO YEAST CELL CYCLE DATA

In the following, the performance of mixtures of LMs with unregularized estimation and of mixtures of LAMs with regularized estimation is compared using the yeast cell cycle dataset from Spellman et al. (1998). Spellman et al. (1998) measured the genome-wide mRNA levels for 6178 yeast ORFs simultaneously over approximately two cell cycle periods. The yeast cells were sampled at 7 min intervals for 119 min leading to a total of 18 time points after synchronization. Among these genes, 800 were classified as cell cycle-regulated genes by Spellman et al. (1998).

In the following, we use the subset of genes that were classified as cell cycle-regulated genes and where the alpha factor arrest is available for all 18 time points. This leads to a final dataset of 613 genes. Almost the same dataset was used by Luan and Li (2003) to fit finite mixtures of mixed-effects models with B-splines. The time-course gene expression data split according to the grouping into five different cell cycle phases (M/G1, G1, S, G2/S and G2/M) proposed by Spellman et al. (1998) is given in the left panel in Figure 4. The levels over time for each gene are joint and are given by the black lines. The mean values for each time point are determined in each group and indicated by the thicker light gray lines.

The following models are fitted and compared: (i) finite mixtures of LMs are estimated unregularized with an intercept and cubic
estimation and LAMs with regularized estimation initialized in the Spellman classification from fitted finite mixtures of LMs with unregularized estimation indicating that the mixture of LAMs with regularized estimation provides a slightly better model fit. The effective degrees of freedom for each of the components for the mixtures of LAMs with regularized estimation varies distinctively from 2.0 to 17.6. The BIC criterion is equal to 25,378 for the best model of the mixtures of LMs with unregularized estimation and 25,356 for the mixture of LAMs with regularized estimation indicating that the mixture of LAMs with regularized estimation provides a slightly better model fit. The component sizes range from 0.03 to 0.16 for the best mixture of LMs with unregularized estimation and from 0.01 to 0.13 for the best mixture of LAMs with regularized estimation.

The implied partitions of the two models are shown in Figures 4 and 5. The expression patterns over time clearly differ between different groups provided by Spellman et al. (1998) classification into five groups for initialization of the (H)EM algorithm. Second, the number of components are varied to take all integer values from 4 to 20 and random initialization with 10 short runs followed by one long run for the best solution of the (H)EM algorithm is used. The best model is selected using the BIC. In addition, the BIC is also used to choose the suitable number of degrees of freedom for the B-splines for the mixtures of LMs and the maximum number of degrees of freedom for the thin plate regression splines for the mixtures of LAMs.

First, the performance is evaluated using the Spellman et al. (1998) classification into five groups for initialization of the (H)EM algorithm. In this setting, the number of components is fixed and only the degrees of freedom are selected using the BIC. For the finite mixtures of LMs, the number of degrees of freedom including the intercept for the five components are as follows: 17.5, 17.5, 14.2, 16.3, 17.4. The BIC values are 26,315 for the LMs with unregularized estimation and 26,263 for the LAMs with regularized estimation. This indicates that with respect to the BIC, the finite mixture model of LAMs with regularized estimation is preferred. Given that the fitted mixture of LAMs has about the same amount of flexibility selected for each of the components the similarity in results is not surprising.

Figure 4 compares the three partitions into five groups determined using (i) Spellman et al. (1998), (ii) mixtures of LMs with unregularized estimation and (iii) mixtures of LAMs with regularized estimation. The cluster means are inserted for the Spellman et al. (1998) partition as well as the fitted mixture of LAMs with regularized estimation. The smooth curves are evaluated differently assigned. The estimated smooth curves evaluated only at the observed time points which are at intervals of 7 min (light gray line) and (ii) at time points at intervals of 1 min (dark gray line). The partitions determined using the finite mixture approach are about the same for LMs with unregularized estimation as well as LAMs with regularized estimation. In fact, only two genes (0.3%) were differently assigned. The estimated smooth curves evaluated only at the observed time points are equally similar. However, evaluation of the estimated curves on a finer grid shows the overfitting of the smooth curves for the mixture components. The smooth curves are evaluated at (i) the observed time points which are at intervals of 7 min (light gray line) and (ii) at time points at intervals of 1 min (dark gray line). The partitions determined using the finite mixture approach are about the same for LMs with unregularized estimation as well as LAMs with regularized estimation. In fact, only two genes (0.3%) were differently assigned. The estimated smooth curves evaluated only at the observed time points are equally similar. However, evaluation of the estimated curves on a finer grid shows the overfitting of the mixtures of LMs with unregularized estimation.

If the number of components as well as the number of degrees of freedom are selected using the BIC, the best model for the LMs with unregularized estimation has 13 components and for LAMs with regularized estimation 16 components. The selected number of degrees of freedom including the intercept for the LMs with unregularized estimation are 11 and the selected maximum number of degrees of freedom including the intercept for the LAMs with regularized estimation are 18. The effective degrees of freedom for each of the components for the mixtures of LAMs with regularized estimation varies distinctively from 2.0 to 17.6. The BIC criterion is equal to 25,378 for the best model of the mixtures of LMs with unregularized estimation and 25,356 for the mixture of LAMs with regularized estimation indicating that the mixture of LAMs with regularized estimation provides a slightly better model fit. The component sizes range from 0.03 to 0.16 for the best mixture of LMs with unregularized estimation and from 0.01 to 0.13 for the best mixture of LAMs with regularized estimation.
Finite mixtures of linear additive models

5 SUMMARY AND OUTLOOK

Mixtures of LAMs with regularized estimation provide a convenient alternative and extension to mixtures of LMs using B-splines. Estimation within an ML framework is possible using an EM-type algorithm where a suitable smoothing parameter is selected iteratively between the E- and M-step in an additional H-step. The results on artificial data and the yeast cell cycle dataset are promising. Especially for the yeast cell cycle dataset, the advantage of automatically selecting different degrees of freedom for the components leads to superior results and allows to easily fit components with different degrees of smoothness.

The proposed model class provides a computationally more efficient way of determining the flexibility needed for the smoothing splines in each of the components. Already under the assumption that the flexibility needed is the same in all components a considerable number of models needs to be estimated and compared to choose the suitable number of degrees of freedom for finite mixtures of LMs, whereas the maximum number of degrees of freedom allowed when fitting finite mixtures of LAMs using regularized estimation is less crucial. If the smoothness of the fitted curves is allowed to vary over components, the number of different finite mixture models of LMs, which need to be compared, would be prohibitively large.

The proposed model class can also be used if the number of time points is only small. However, the computational advantages are less important, because a complete enumeration of all possible combinations of flexibility allowed in the components is more likely to be computationally feasible. Furthermore, the area of application is not restricted to time-course gene expression data. The proposed method could, for example, also be used to model stock prices over time.

In the future, the extension of regularized estimation of the model of finite mixtures of linear additive models to the regularized estimation of mixtures of linear additive mixed models could be considered. We assume that the performance, in general, as well as in comparison to linear mixed models should essentially be the same. Estimation, however, is more complex and the implementation is complicated by the fact that available standard tools for fitting linear mixed models do not allow for weighted ML estimation. This is necessary because the random effects are on the individual level and therefore they are integrated out before the individual log-likelihoods are weighted with the a posteriori probabilities in the M-step.

Funding: Austrian Science Fund (FWF): Hertha-Firnberg grant (T351-N18); Elise-Richter grant (V170-N18).

Conflict of Interest: none declared.

REFERENCES

Androulakis, I. et al. (2007) Analysis of time-series gene expression data: Methods, challenges, and opportunities. Ann. Rev. Biomed. Eng., 9, 205–228.
B. Grün et al.

Biernacki, C. et al. (2003) Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixtures. Comput. Stat. Data Anal., 41, 561–575.

Celeux, G. et al. (2005) Mixture of linear mixed models for clustering gene expression profiles from repeated microarray experiments. Stat. Model., 5, 245–267.

Dempster, A. P. et al. (1977) Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. B, 39, 1–38.

Fralley, C. and Raftery, A. E. (2002) Model-based clustering, discriminant analysis and density estimation. J. Am. Stat. Assoc., 97, 611–631.

Grün, B. and Leisch, F. (2008) FlexMix version 2: finite mixtures with concomitant variable selection. J. Stat. Softw., 12, 1–46.

Grün, B. and Leisch, F. (2009) gcExplorer: interactive exploration of gene clusters. R package version 1.3.1.

Hastie, T. and Tibshirani, R. (1990) Generalized Additive Models, vol. 43 of Monographs on Statistics and Applied Probability. 1st edn. Chapman and Hall, London.

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

Koch, T. (2006) Mixed Model Based Inference in Structured Additive Regression. PhD Thesis, Institut für Statistik, Ludwig-Maximilians-Universität München.

Leisch, F. (2004) FlexMix: a general framework for finite mixture models and latent class regression in R. J. Stat. Softw., 11, 1–18.

Luan, Y. and Li, H. (2003) Clustering of time-course gene expression data using a mixed-effects model with B-splines. Bioinformatics, 19, 474–482.

Maugeri, C. et al. (2009) Variable selection for clustering with Gaussian mixture models. Biometrics, 65, 701–709.

Ng, S. K. et al. (2006) A mixture model with random-effects components for clustering correlated gene-expression profiles. Bioinformatics, 22, 1745–1752.

R Development Core Team (2011) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.

Scharl, T. and Leisch, F. (2009) gcExplorer: interactive exploration of gene clusters. Bioinformatics, 25, 1089–1090.

Scharl, T. et al. (2010) Mixtures of regression models for time-course gene expression data: Evaluation of initialization and random effects. Bioinformatics, 26, 370–377.

Spelman, P. T. et al. (1998) Comprehensive identification of cell-cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. Mol. Biol. Cell., 9, 3273–3297.

Wood, S. N. (2006) Generalized Additive Models: An Introduction with R. Chapman and Hall/CRC Press, Boca Raton.

Wood, S. N. (2011) Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. J. R. Stat. Soc. B, 73, 3–36.

APPENDIX A

A1. COMPUTATIONAL DETAILS

All computations are performed in the statistical computing environment R version 2.13.1 (R Development Core Team, 2011) with the packages flexmix 2.3-4, mclust 5.1.7, survival 2.36-8, multcomp 1.5-3, grid 2.13.1, lattice 0.19-33, and tools 2.13.1. The EM algorithm for ML estimation of finite mixture models is implemented in the R package flexmix (Grün and Leisch, 2008). For mixtures of linear additive models with regularized estimation, FL2DStgvg() is used as model driver for the H- and M-step. FL2DStgvg() uses functionality for regularized fitting of generalized additive models from the R package mvgg (Wood, 2011). The datasets for the simulation study using artificial data were conveniently generated using functions gc2sim() and gc2data() from the R package gcExplorer (Scharl and Leisch, 2009). The yeast cell cycle dataset is available in the Bioconductor package yeastCC.

B1. LIKELIHOOD MAXIMIZED IN THE H-STEP

In the H-step, the random effects are integrated out. This results in the following likelihood where only $\lambda_k$ is assumed to be a parameter, $\beta_k$ and $\sigma_k^2$ depend on $\lambda_k$ and $Y$, $X$ and $Z$ and $i = (c_1, c_2, \ldots, c_n)$ with $c_i = (c_{ik})_k$ are given.

$$\delta_k(\lambda_k|Y, X, Z, i) = \prod_{i=1}^{n} \prod_{c_k=1}^{C_k} \left( f_{\beta_k}(y_{ic_k}|x_{ic_k}) \right)^{\pi_{ic_k}}$$

Using the following identity

$$f(y_{ic_k}|x_{ic_k}) = (2\pi)^{\frac{C_k}{2}} \left| \sigma_k^{-2} \right|^{\frac{1}{2}} e^{-\frac{1}{2} (y_{ic_k} - \hat{Y}_{ic_k})^T \sigma_k^{-2} (y_{ic_k} - \hat{Y}_{ic_k})}$$

and as well as the fact that the marginal distribution for a linear random effects model is a multivariate normal distribution we have

$$\delta_k(\lambda_k|Y, X, Z, i) = \left( 2\pi \sigma_k^{-2} \right)^{\frac{n}{2}} \left( \prod_{i=1}^{n} \left| \sigma_k^{-2} \right|^{\frac{1}{2}} e^{-\frac{1}{2} (y_{ic_k} - \hat{Y}_{ic_k})^T \sigma_k^{-2} (y_{ic_k} - \hat{Y}_{ic_k})} \right).$$

where

$$u = \frac{1}{n} \sum_{i=1}^{n} z_{ic_k} (1 - \hat{z}_{ic_k}).$$

$\lambda_k$ denotes the identity matrix of suitable dimension, $\hat{Y}_{ic_k}$ and $\hat{X}_{ic_k}$ and $\hat{Z}_{ic_k}$ are analogously defined. The diag$(\hat{z}_{ic_k})$ denotes diagonal matrix with $\hat{z}_{ic_k}$ in the diagonal. Transforming this back to the original variables gives

$$\delta_k(\lambda_k|Y, X, Z, i) = \left( 2\pi \sigma_k^{-2} \right)^{\frac{n}{2}} \left( \prod_{i=1}^{n} \left| \sigma_k^{-2} \right|^{\frac{1}{2}} e^{-\frac{1}{2} (y_{ic_k} - \hat{Y}_{ic_k})^T \sigma_k^{-2} (y_{ic_k} - \hat{Y}_{ic_k})} \right).$$

C1. ALGORITHMIC IMPLEMENTATION

In the following, the building blocks of the implementation are outlined which make use of functionality from packages flexmix and mvgg.

(1) Data pre-processing: the vector of responses, the model matrix for the fixed effects as well as the smoothers are determined using functionality from package mgcv.

(2) E-step: given the current parameter estimates, the a posteriori probabilities are determined by predicting the mean values and evaluating the likelihood.

(3) H- and M-step: for each component separately $\lambda$ and the corresponding parameters are jointly determined with the fit function from package mvgg for the weighted data.