Quantitative Trait Associated Microarray Gene Expression Data Analysis

Yi Qu and Shizhong Xu
Department of Botany and Plant Sciences, University of California, Riverside

Selection on phenotypes may cause genetic change. To understand the relationship between phenotype and gene expression from an evolutionary viewpoint, it is important to study the concordance between gene expression and profiles of phenotypes. In this study, we use a novel method of clustering to identify genes whose expression profiles are related to a quantitative phenotype. Cluster analysis of gene expression data aims at classifying genes into several different groups based on the similarity of their expression profiles across multiple conditions. The hope is that genes that are classified into the same clusters may share underlying regulatory elements or may be a part of the same metabolic pathways. Current methods for examining the association between phenotype and gene expression are limited to linear association measured by the correlation between individual gene expression values and phenotype. Genes may be associated with the phenotype in a nonlinear fashion. In addition, groups of genes that share a particular pattern in their relationship to phenotype may be of evolutionary interest. In this study, we develop a method to group genes based on orthogonal polynomials under a multivariate Gaussian mixture model. The effect of each expressed gene on the phenotype is partitioned into a cluster mean and a random deviation from the mean. Genes can also be clustered based on a time series. Parameters are estimated using the expectation–maximization algorithm and implemented in SAS. The method is verified with simulated data and demonstrated with experimental data from 2 studies, one clusters with respect to severity of disease in Alzheimer’s patients and another clusters data for a rat fracture healing study over time. We find significant evidence of nonlinear associations in both studies and successfully describe these patterns with our method. We give detailed instructions and provide a working program that allows others to directly implement this method in their own analyses.

Introduction

Selection may act on phenotypes and thus on the underlying genetic components in an indirect manner. One class of genes that are important in evolution are the genes involved in developmental pathways because developmental processes shape the morphologies of species. In recent years, with the newly emerged microarray technology, great effort has been made to identify genes that are expressed in particular developmental stages according to specific patterns (Arbeitman et al. 2002; Baugh et al. 2003; Reinke et al. 2004). The study of the expression of these genes over time is an important component in understanding how the gene expression patterns covary, and regulation of these genes is then responsible for many downstream phenotypes. Quantitative trait locus mapping represents the traditional method for identifying genes associated with quantitative phenotypes. Here, in this study, we develop a novel statistical method that uses microarray gene expression data to describe the expression profiles for genes that exhibit similar profiles with regard to a quantitative phenotype. This method can also be applied to grouping expression profiles along a developmental time course because time is on a continuous scale.

Microarray technology was developed only recently (Schena et al. 1995). However, it already has a tremendous impact (i.e., Lander 1999; West et al. 2001). A single chip can measure the expression of thousands of genes simultaneously. Microarray data can and should be analyzed according to the purposes of the experiments and the data structures. Differential expression analysis aims at detecting genes differently expressed between groups, that is, control and treatment (i.e., Schena et al. 1995; Baldi and Long 2001; Tusher et al. 2001; Dudoit et al. 2002), whereas cluster analysis classifies genes into different groups (Eisen et al. 1998; Quackenbush 2001; Yeung et al. 2001; Qu and Xu 2004). In this paper, we focus on grouping or clustering profiles. Previous clustering is based on the similarity of expression profiles across discrete conditions, for example, tissues collected from different types of organs (Eisen et al. 1998), time points after a particular drug treatment (Saban et al. 2001; Luan and Li 2003), and doses of a drug injection (Peddada et al. 2003). The most commonly used methods include hierarchical clustering (Carr et al. 1997), K-means (Tavazoie et al. 1999), and self-organizing maps (Tamayo et al. 1999). These algorithms are “heuristic” algorithms that classify genes based on the “distance matrix” and do not require any underlying statistical models. As a possible alternative to the heuristic algorithms, model-based clustering methods have also been proposed and applied to microarray data (Fraley and Raftery 1998; Yeung et al. 2001; Fraley and Raftery 2002; Qu and Xu 2004). An advantage of using the model-based algorithms is that the uncertainty of cluster assignment can be evaluated. In addition, alternative models can be exploited and compared.

Gene expression profiles have also been used for classification of tissues for disease diagnosis (West et al. 2001; Bicciato et al. 2003; Liu et al. 2003; Meireles et al. 2004; Tadesse et al. 2005). Classification methods are conceptually the same as gene clustering in the sense that the data analyses are conducted with rows and columns being transposed. Special techniques have been developed for tissue classification, for example, dimension reduction (Liu et al. 2003) and variable selection (Tadesse et al. 2005).

Clustering based on discrete groups or classification prediction based on discrete categories is valuable. However, there are many evolutionary interesting phenotypes that are quantitative in nature. The association between quantitative phenotype and gene expression can be examined pairwise using a Pearson correlation coefficient between the expression of a single gene and a continuous phenotype (Quackenbush 2001; Kraft et al. 2003). Taking
this one step further, Blalock et al. (2004) ranked genes according to the correlation coefficients of gene expression with Mini Mental Status Examination (MMSE), a quantitative measurement of severity of Alzheimer’s disease (AD), and detected many genes associated with AD. Kraft et al. (2003) used the within-family correlation analysis to remove the effect of family stratification. Although the Pearson correlation coefficient is a straightforward way to assess a linear relationship between a single gene and a quantitative phenotype, it is limited to linear relationships with a single gene/phenotype combination. Potokina et al. (2004) investigated the association of gene expression with 6 malting quality phenotypes (quantitative traits) of 10 barley cultivars. They compared the distance matrix for each gene expression profile among the 10 cultivars with each of the remaining 3 ley cultivars. They compared the distance matrix for each cultivar with each of the distance matrix calculated from the phenotypes and applied a g-test statistic. This approach allows the examination of multiple phenotypes but still focuses on linear associations with a single gene. The linear association between groups of genes and a quantitative phenotype has been described in a regression setting (Jia and Xu 2005) and for groups of genes using factor analysis (Coffman et al. 2005). However, the relationship between the phenotype of interest and the gene expression profile may not be linear. For example, in the case of developmentally regulated genes, we may expect to see exponential or Poisson distribution.

In the AD microarray experiment of Blalock et al. (2004), we have seen a concrete example of nonlinear association between gene expression and MMSE. Figure 1 plots expression profiles (on the y-axis) against MMSE phenotype (on the x-axis) for 6 genes. Gene (ABCF1) shows no obvious association with the phenotype. Gene (37870) and (MYO1F) show positive and negative associations, respectively, with the phenotype. The remaining 3 genes (CALB1, GDF8, and ANKH) show nonlinear associations with the phenotype. Existing methods developed solely for detecting the linear association (Blalock et al. 2004; Jia and Xu 2005) will not identify an association between phenotype and gene expression in these cases (CALB1, GDF8, and ANKH). However, a method able to identify nonlinear association should identify the association between these genes and the phenotype MMSE.

In this study, we develop a novel algorithm to cluster genes based on nonlinear association between gene expression and a quantitative phenotype using orthogonal polynomials under the Gaussian mixture model. In our model, the whole data set is assumed to be a mixture of $K$ clusters. The expression of each gene is considered as the sum of a cluster mean (fixed effect), a normally distributed random effect specific for the gene (random effect), and a random measurement noise. We use a linear contrast to remove the mean expression of each gene, which is not of interest, from the model, so that we can cluster gene based solely on the pattern of association of their expression with the phenotypic values. Parameters in the model are estimated with the expectation–maximization (EM) algorithm (Dempster et al. 1977). The random effect for each gene is predicted with best linear unbiased predictor (BLUP)(Robinson 1991). Bayesian information criterion (BIC) is applied to choose the optimal number of clusters.

### Materials and Methods

#### Orthogonal Polynomials of Gene Expression Profile

Let $Z$ be the phenotypic value of a quantitative trait (a continuous variable) and $Y_i(Z)$ be the expression level of gene $i = 1, \ldots, M$ written as a function of the phenotype, where $M$ is the total number of genes studied. The following model may be used to describe the relationship between $Y_i(Z)$ and $Z$

$$Y_i(Z) = \alpha_i(Z) + \varepsilon_i,$$  

where $\alpha_i(Z)$ is an arbitrary function chosen to describe the relationship between the expressed gene and the phenotype and $\varepsilon_i$ is a random error reflecting the uncertainty of the fit of the model with an assumed $N(0, \sigma^2)$ distribution. When describing gene expression as a function of phenotype, we have borrowed the idea of Luan and Li (2003) in time-course microarray data analysis where gene expression is described as a function of time, that is, $Y_i(T) = \alpha_i(T) + \varepsilon_i$.

Note that function $\alpha_i(Z)$ has a subscript $i$, meaning that this function varies from gene to gene. In the model-based cluster analysis, Luan and Li (2003) further partitioned $\alpha_i(Z)$ into a fixed effect function and a random effect function. The fixed effect function is a mixture of several functional clusters, depending on which cluster the $i$th gene belongs to. For example, if the $i$th gene belongs to the $k$th cluster ($k = 1, \ldots, K$), the mixed-effects model is

$$Y_i(Z) = \beta_{ik}(Z) + \gamma_i(Z) + \varepsilon_i,$$

where $\beta_{ik}(Z)$ is a function shared by all genes in cluster $k$ and $\gamma_i(Z)$ is a gene-specific random effect function. As a result, model (2) is a functional mixed-effects model.

All the model parameters, except $\sigma^2$, are a function of $Z$. The functional relationships between the parameters and $Z$ may be described with an orthogonal polynomial. Define $\psi(Z) = [\psi_0(Z) \psi_1(Z) \cdots \psi_d(Z)]$ as a basis of the polynomial with order $d$. A basis also is called a submodel. A method to construct the basis of orthogonal polynomial is given in Supplemental Material A (Supplementary Material online). Let us define $\beta_k = [\beta_{k0} \beta_{k1} \cdots \beta_{kd}]^T$ as a vector of model effects representing the mean values of cluster $k$. The phenotype associated model effect $\beta_k(Z)$ can be described as a linear combination of $\beta_k$ weighted by the coefficients of the polynomials, that is, $\beta_k(Z) = \psi(Z)\beta_k$. Similarly, we can describe $\gamma_i(Z)$ by $\gamma_i(Z) = \psi(Z)\gamma_i$, where $\gamma_i = [\gamma_{i0} \gamma_{i1} \cdots \gamma_{id}]^T$. Because $\gamma_i$ is a vector of random effects, we assume that $\gamma_i$ is i.i.d. $N(0, \Sigma)$, where $\Sigma$ is a $(d+1) \times (d+1)$ positive definite covariance matrix. Note that $\beta_k(Z)$ and $\gamma_i(Z)$ are scalars, but $\beta_k$ and $\gamma_i$ are vectors each with $d + 1$ elements.

Assume that the $i$th gene belongs to the $k$th cluster, with the polynomial reparameterization, model (2) can be now rewritten as a linear function of the phenotype-independent parameters

$$Y_i(Z) = \psi(Z)\beta_k + \psi(Z)\gamma_i + \varepsilon_i.$$  

The expectation of $Y_i(Z)$ conditional on the fixed effects is

$$E(Y_i(Z)) = \psi(Z)\beta_k.$$
The covariance between \( Y_i(Z) \) and \( Y_i(S) \) conditional on the fixed effects is

\[
\text{Cov}(Y_i(Z), Y_i(S)) = \psi(Z)\Sigma\psi^\top(S) + \eta(Z, S)\sigma^2,
\]

where \( \eta(Z, S) = 1 \) for \( Z = S \) and \( \eta(Z, S) = 0 \) otherwise.

Model (3) is linear with the model parameters, which can be estimated using the mixed model framework. We first estimate \( \beta = [\beta_1, \cdots, \beta_K]^\top, \gamma = [\gamma_1, \cdots, \gamma_M]^\top, \Sigma, \) and \( \sigma^2 \) and then use the basis of the polynomial to convert \( \beta \) and \( \gamma \) into their phenotype-dependent functional parameters.

### Likelihood Function of Normalized Gene Expression

Assume that the expression profiles for \( M \) genes are measured in \( N \) microarrays, where each microarray represents a single subject. The subjects are sorted by the phenotypic values in either ascending or descending order and denoted by \( Z_1, \ldots, Z_N \). We use an \( N \times 1 \) vector to denote the expression levels of gene \( i \) collected from \( N \) subjects, \( Y_i = [Y_i(Z_1), Y_i(Z_2), \cdots, Y_i(Z_N)]^\top \), for \( i = 1, \ldots, M \), where \( M \) is the total number of genes. Define

\[
\Psi = \begin{bmatrix}
\psi(Z_1) & \psi(Z_2) & \cdots & \psi(Z_N)
\end{bmatrix}
\]

as a \((d + 1) \times N\) matrix, where \( \psi_j(Z_i) = 1 \) for all \( j = 1, \ldots, N \) (see Supplemental Material A for details on the method of

---

**FIG. 1.—Expression profiles of a few genes as a function of the AD phenotype (MMSE):**

- **ABCF1**: gene expression does not change with the phenotypic value.
- **37870**: gene expression increases as the phenotype value increases.
- **MYO1F**: gene expression decreases as the phenotypic value increases.
- **CALB1, GDF8, and ANKH**: gene expression is associated with the phenotypic value in a nonlinear fashion.
constructing matrix $\psi$, Supplementary Material online). Let $C$ be a cluster indicator variable for gene $i$, for example, $C = k$ if $i$ belongs to cluster $k$. In matrix notation, the linear model for vector $Y_i$ is

$$
y_i|C_i = k = \psi^T \beta_k + \psi^T \gamma_i + e_i,$$

where $Y_i|C_i = k$ is used to indicate that the model is presented conditional on gene $i$ belonging to cluster $k$ and $e_i = (e_{i1}, \ldots, e_{in})^T$ is now an $N \times 1$ vector for the residual effects with a multivariate normal distribution, that is, $e_i \sim \mathcal{N}(0, I_{n^2})$, where $I$ is an $N \times N$ identity matrix.

There are 2 points that need to be examined with care. One is that the phenotypic value $Z$ needs to be rescaled before it enters the polynomial analysis. This is to ensure that the shape of the curve rather than the mean value is clustered. The rescaled variable, $z_i$, runs between $-1$ and $1$, that is, $-1 \leq z_i \leq 1$ (Hayes 1974). The other point is that we assume all preprocessing for the gene expression profiles has already occurred (e.g., transformation). Let $Y_{ij} = Y_i(z_j)$ be the log-transformed expression for gene $i$ measured from subject $j$; then the normalized expression is

$$
y_{ij} = y_i(z_j) = Y_{ij} - \bar{Y}_i = Y_{ij} - N^{-1} \sum_{j=1}^{N} Y_{ij}. \tag{8}\$$

Note that after normalization, the intercepts, $\beta_{00}$ and $\gamma_{00}$, have been removed from the model (see Supplemental Material B for the proof, Supplementary Material online). In addition, we do not have $N$ independent observations of the normalized expression for each gene. Therefore, the last element of vector $y_i$ should be excluded and $y_i$ becomes an $n \times 1$ vector for $n = N - 1$. Accordingly, $\beta_k$ and $\gamma_i$ each becomes an $d \times 1$ vector, $\Sigma$ is an $d \times d$ matrix, and $\psi$ becomes a $d \times n$ matrix. Finally, the vector of residual errors for the normalized data is redefined as $Le$, where $L$ (an $n \times N$ matrix) is obtained by deleting the last row of matrix $I - \frac{1}{N}1_N 1_N^T$ (see Supplemental Material B for the definition and derivation of this matrix, Supplementary Material online). Therefore, the covariance matrix of the normalized residuals is

$$\text{Var}(Le) = L\text{Var}(e_i)L^T = LL^T \sigma^2 = R \sigma^2, \tag{9}\$$

where $R = LL^T$ is a known $n \times n$ positive definite matrix.

Given the parameters $\theta = \{\beta, \Sigma, \tau^2, \pi\}$ and the data $D = \{y, z\}$, the likelihood function given in equation (14) with all the missing values integrated out is the observed likelihood function. Instead of maximizing the observed likelihood function, the EM algorithm deals with the following complete data likelihood function,

$$L(\theta, C, \gamma) = \prod_{j=1}^{M} p(y_i|C_i, \gamma_i)p(C_i)p(\gamma_i), \tag{15}\$$

where

$$p(y_i|C_i, \gamma_i) = (\tau^2)^{-n/2}|R|^{-1/2} \times \exp \left[ -\frac{1}{2\tau^2} \sum_{k=1}^{K} \eta(C_i, k) \times (y_i - \psi^T \beta_k - \psi^T \gamma_i)^T R^{-1} \times (y_i - \psi^T \beta_k - \psi^T \gamma_i) \right], \tag{16}\$$

and

$$p(C_i) = \prod_{k=1}^{K} \pi_k^{n(C_i, k)}. \tag{18}\$$

For convenience of presentation, we defined variable $\eta(C_i, k) = 1$ for $C_i = k$ and $\eta(C_i, k) = 0$ for $C_i \neq k$. Because the complete data likelihood function involves missing values, integration (or expectation) is taken with respect to the missing values, not for the complete data likelihood.
function (15) but for the log-transformed complete likelihood function given below,
\[ l(\theta, C, \gamma) = \ln[L(\theta, C, \gamma)] = \sum_{i=1}^{M} \{\ln[p(y_i|C_i, \gamma_i)] + \ln p(C_i) + \ln p(\gamma_i)\}. \] (19)

The expectation of equation (19) with respect to the missing values is
\[ l(\theta) = E[l(\theta, C, \gamma)] = T_1(\theta) + T_2(\theta) + T_3(\theta), \] (20)
where
\[ T_{11}(\theta) = -\frac{1}{2}nM \ln(\sigma^2) - \frac{1}{2}M \ln|\Sigma|, \]
\[ T_{12}(\theta) = -\frac{1}{2\sigma^2} \sum_{i=1}^{M} \sum_{c=1}^{K} E[\eta(C_i, k)](y_i - \psi^T \beta_k - \psi^T \gamma_i)^T \Sigma^{-1} \]
\[ \times (y_i - \psi^T \beta_k - \psi^T \gamma_i), \] (22b)
\[ T_2(\theta) = \frac{M}{2} \sum_{i=1}^{M} \left\{ \sum_{c=1}^{K} E[\eta(C_i, k)] \ln(\pi_c) \right\}, \] (22)
and
\[ T_3(\theta) = -\frac{1}{2}M \ln|\Sigma| - \frac{1}{2} \sum_{i=1}^{M} E[\psi^T \Sigma^{-1} \psi]. \] (23)

The EM algorithm requires taking the partial derivatives of \( l(\theta) \) with respect to the parameters, setting these partial derivatives equal to zero and finding explicit solutions for the parameters as a function of the missing values. This completes the maximization step. The expectation step requires taking the expectations of all terms involving the missing values. Because the complete data likelihood function is just a likelihood function for a typical mixed model, we can simply utilize existing mixed model EM algorithm to find the MLE of parameters (Dempster et al. 1977). Followings are the EM steps for the mixed model analysis.

Step 0: Let \( t = 0 \) and initialize all parameters with values in their legal domain, denoted by \( \theta^{(0)} \).

Step 1 (E1): Compute the posterior probabilities of cluster assignment for each gene
\[ \pi_c = p(C_i = k|y_i) = E[\eta(C_i, k)] = \frac{\pi^{(0)}_c p(y_i|C_i = k)}{\sum_{c'=1}^{K} \pi^{(0)}_{c'} p(y_i|C_i = k')}. \] (24)

Step 2 (E2): Find the posterior distribution of the random effect. This posterior distribution turns out to be a mixture of \( K \) normal distributions. If gene \( i \) belongs to cluster \( k \), then the conditional expectation is
\[ E(\gamma_i|C_i = k) = \Sigma^{(i)} \psi V^{-1}(y_i - \psi^T \beta^{(i)}_k), \] (25)
and the covariance matrix
\[ \text{Var}(\gamma_i|C_i = k) = \Sigma^{(i)} - \Sigma^{(i)} \psi V^{-1} \Sigma^{(i)}. \] (26)

Step 3 (M1): Update the regression coefficients for each cluster
\[ \beta^{(i+1)}_k = (\pi^{(i)}_c M \psi V^{-1} \psi^T)^{-1} \psi V^{-1} \sum_{i=1}^{M} \pi_{u} y_i, \] (27)

Step 4 (M2): Update the covariance matrix of the random effect
\[ \Sigma^{(i+1)} = \frac{1}{M} \sum_{i=1}^{M} \sum_{k=1}^{K} \pi_{u} E[\gamma_i | C_i = k] \]
\[ = \frac{1}{M} \sum_{i=1}^{M} \sum_{k=1}^{K} \pi_{u} [\text{Var}(\gamma_i | C_i = k) + E(\gamma_i | C_i = k)E(\gamma_i | C_i = k)]. \] (28)

Step 5 (M3): Update the residual variance
\[ \sigma^{2(i+1)} = \frac{1}{nM} \sum_{i=1}^{M} \gamma_i^T \psi R^{-1} (y_i - \psi^T \sum_{k=1}^{K} \pi_{u} b^{(i)}_k) \]
\[ - \psi^T \sum_{k=1}^{K} \pi_{u} E(\gamma_i | C_i = k). \] (29)

Step 6 (M4): Update the mixing proportions
\[ \pi^{(i+1)}_c = \frac{1}{M} \sum_{i=1}^{M} E[\eta(C_i, k)] \]
\[ = \frac{1}{M} \sum_{i=1}^{M} p(C_i = k|y_i) = \frac{1}{M} \sum_{i=1}^{M} \pi_{u}. \] (30)

Step 7: Set \( t = t + 1 \) and repeat from step 1 to step 6 (2 E-steps and 4 M-steps) many times until \( ||\theta^{(t+1)} - \theta^{(t)}|| \leq 10^{-8} \) holds.

The Optimal Number of Clusters

The above EM algorithm applies to situations where the number of clusters \( K \) is fixed. In reality, \( K \) must be estimated from the data. The commonly used method for inferring \( K \) is to calculate the BIC and choose \( K \) that maximizes the BIC value (Schwarz 1978). BIC is considered as an approximation to the Bayesian factor and is defined as
\[ \text{BIC}_k = 2L(\hat{\theta}|K) - \dim(\theta|K) \ln(M), \] (31)
for \( K \) clusters, where \( \theta|K \) is the MLE of the parameter vector \( \theta \) under \( K \) clusters and \( \dim(\theta|K) \) is the dimension of \( \theta \), that is, the number of parameters in the model.

After the EM algorithm converges and the number of clusters is determined, we obtain all the estimated parameters. We also have the posterior probability that the \( i \)th gene belongs to the \( k \)th cluster, \( \pi_{ik} = p(C_i = k|y_i) \), for
\( i = 1, \ldots, M \) and \( k = 1, \ldots, K \). Based on these probabilities, we can assign the \( i \)th gene into the \( k \)th cluster if \( \pi_{ik} = \max (\pi_{i1}, \ldots, \pi_{iK}) \). The \( \pi_{ik} \) also provides a measure of the uncertainty about the resulting clustering. We can also try only cluster those genes with the maximum \( \pi_{ik} \) greater than a predetermined cutoff value and declare other genes as unresolved (Luan and Li 2003).

After gene clustering, we obtain estimates of the gene expression profiles. For the \( i \)th gene in the \( k \)th cluster, the BLUP of the random regression coefficients \( \gamma_i \) is given in equation (25) and is now denoted by \( \hat{\gamma}_i \). Let \( \hat{\beta}_k \) be the MLE of \( \beta_k \) after the EM algorithm converges. The corresponding estimate of the individual gene expression profile for the \( i \)th gene in the \( k \)th cluster is

\[
\hat{y}_i | C_i = k = \psi^T \hat{\beta}_k + \psi^T \hat{\gamma}_i. \tag{32}
\]

Note that \( \hat{y}_i | C_i = k \) is a vector of the BLUP estimates of the \( i \)th gene expressions on \( n \) subjects. The actual gene expression profile is

\[
\hat{y}_i(z) | C_i = k = \psi(z) \hat{\beta}_k + \psi(z) \hat{\gamma}_i, \tag{33}
\]

which can be examined visually by plotting \( \hat{y}_i(z) \) against \( z \).

**Results**

**Simulation Study**

In order to validate the method and the computer program that we developed to implement the method, we undertook 2 simulation studies.

**Data Set I**

In this data set, we simulated \( M = 1200 \) genes from \( K = 5 \) clusters. The first cluster contained 400 genes, whereas each of the other 4 clusters contained 200 genes. The expression levels of each gene were measured from \( N = 50 \) subjects. The phenotypic value for each subject \( Z_i \) was simulated from \( U(0.75, 1.25) \), a uniform distribution from 0.75 to 1.25. The phenotypic values were then rescaled to \( z_i \), so that \(-1 \leq z_i \leq 1\). The order of the polynomial regression was \( d = 4 \). To simulate the expression values of genes in the \( k \)th cluster for all the 50 subjects, we used the following model to generate the data,

\[
Y_i = \psi(z_i) \beta_k + \psi(z_i) \gamma_i + e_{i},
\]

\[
= \psi_0(z_i) \beta_0 + \sum_{j=1}^{d} \psi_j(z_i) \beta_j + \psi_0(z_i) \gamma_0 + \sum_{j=1}^{d} \psi_j(z_i) \gamma_j + e_{i}, \tag{34}
\]

where

\[
\begin{array}{c}
\beta = (\beta_1, \beta_2, \beta_3, \beta_4, \beta_5)^T \\
\beta_0 \beta_1 \beta_2 \beta_3 \beta_4 \beta_5 \beta_6 \beta_7 \beta_8 \beta_9 \beta_{10} \beta_{11} \beta_{12} \beta_{13} \beta_{14} \beta_{15} \beta_{16} \beta_{17} \beta_{18} \beta_{19} \beta_{20} \beta_{21} \beta_{22} \beta_{23} \beta_{24} \beta_{25} \beta_{26} \beta_{27} \beta_{28} \beta_{29} \beta_{30} \beta_{31} \beta_{32} \beta_{33} \beta_{34} \beta_{35} \beta_{36} \beta_{37} \beta_{38} \beta_{39} \beta_{40} \beta_{41} \beta_{42} \beta_{43} \beta_{44} \beta_{45} \beta_{46} \beta_{47} \beta_{48} \beta_{49} \beta_{50} \beta_{51} \beta_{52} \beta_{53} \beta_{54}
\end{array}
\]

The random vector \( \gamma = [\gamma_0, \gamma_1, \gamma_2, \gamma_3, \gamma_4]^T \) was sampled from a \( N(0, \Sigma) \) distribution, where

\[
\Sigma = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 0.5 & 0 & 0 & 0 \\
0 & 0 & 0.4 & 0 & 0 \\
0 & 0 & 0 & 0.3 & 0 \\
0 & 0 & 0 & 0 & 0.3
\end{bmatrix}.
\]

The \( e_{ij} \) were sampled from a \( N(0, \sigma^2) \) distribution, where \( \sigma^2 = 0.16 \). The true proportions of the genes in the 5 clusters were \( \pi = [0.333, 0.167, 0.167, 0.167, 0.167] \). The simulated gene expression data were then normalized using \( y_i = L y_i \), where

\[
L = \begin{bmatrix}
1 & -1/50 & -1/50 & \cdots & -1/50 & -1/50 \\
-1/50 & 1 & -1/50 & \cdots & -1/50 & -1/50 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
-1/50 & -1/50 & -1/50 & \cdots & 1 & -1/50 & -1/50
\end{bmatrix}
\]

is a \( 49 \times 50 \) matrix. When we used the normalized data for cluster analysis, the intercepts \( \{\beta_0, \gamma_0\} \) were automatically deleted so that the parameters subject to estimation were

\[
\hat{\beta} = \begin{bmatrix}
\beta_{11} & \beta_{12} & \beta_{13} & \beta_{14} \\
\beta_{21} & \beta_{22} & \beta_{23} & \beta_{24} \\
\beta_{31} & \beta_{32} & \beta_{33} & \beta_{34} \\
\beta_{41} & \beta_{42} & \beta_{43} & \beta_{44} \\
\beta_{51} & \beta_{52} & \beta_{53} & \beta_{54}
\end{bmatrix}
\]

and

\[
\hat{\Sigma} = \begin{bmatrix}
0.5 & 0 & 0 & 0 \\
0 & 0.4 & 0 & 0 \\
0 & 0 & 0.3 & 0 \\
0 & 0 & 0 & 0.3
\end{bmatrix}.
\]

both of which had reduced dimensions.

We have implemented the algorithm in SAS, which is available as supplemental material online. The cluster analysis was performed using a variable number of clusters, \( K = 3, \ldots, 9 \). The maximum BIC value occurred when \( K = 5 \), the true number of clusters simulated. Under \( K = 5 \), we obtained the following estimated parameters:

\[
\begin{bmatrix}
0.037 & -0.2124 & 0.065 & -0.0879 \\
3.0284 & 0.1767 & 0.1112 & -0.228 \\
1.104 & -4.0589 & 0.1465 & 0.1952 \\
1.9410 & -2.0869 & 2.0113 & -3.9215 \\
-2.0048 & -1.0605 & 1.4529 & 0.1461
\end{bmatrix},
\]

\[
\begin{bmatrix}
0.4679 & -0.0155 & -0.0004 & -0.01 \\
-0.0155 & 0.4214 & 0.0031 & -0.0056 \\
-0.0004 & 0.0031 & 0.2731 & 0.0095 \\
-0.01 & -0.0056 & 0.0095 & 0.3056
\end{bmatrix},
\]

and \( \sigma^2 = 0.1604 \).
These estimated values were close to the simulated parameters. We next examined the rate of correct classification of genes for each cluster. We assigned the ith gene to the jth cluster if \( p_{ij} = \max(p_{i1}, \ldots, p_{ik}) \). The overall correct clustering rate for the whole data set was 98%, which was distributed as 390/400, 194/200, 199/200, 200/200, and 194/200, respectively, for the 5 clusters.

The true and estimated curves for the 5 clusters are depicted in figure 2. The true curve for each cluster was generated using

\[
y_i(z) = \psi(z) \beta_k,
\]

whereas estimated curve for each cluster was generated using

\[
\hat{y}_i(z) = \psi(z) \hat{\beta}_k,
\]

for \( k = 1, \ldots, 5 \), where \( \beta_k \) is the true value and \( \hat{\beta}_k \) is the estimate of the mean of cluster \( k \).

We used

\[
\hat{y}_i(z)|_{C_k = j} = \psi(z) \hat{\beta}_k + \psi(z) \hat{\gamma}_j
\]

to predict the profiles of individual gene expression for the 5 clusters. These profiles are depicted in figure 3.

**Data Set II**

In this simulation experiment, we generated \( M = 500 \) genes from \( K = 6 \) clusters. The first cluster contained 4000 genes, and each of the other 5 clusters contained 200 genes. Variables \( Z \) and \( Y \) were simulated in the same way as described for data set I. But this time, we let

\[
\beta = \begin{bmatrix}
\beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6 & \beta_7 & \beta_8 & \beta_9 & \beta_{10} \\
0.24 & -0.08 & -0.12 & -0.05 & 0.34 & -0.06 & 0.1 & 0.1 & 0.1 & 0.1
\end{bmatrix}
\]

\[
\Sigma = I_{4 \times 4} \times 0.005, \text{ where } I \text{ is the identity matrix and } \sigma^2 = 0.01.
\]

Using this data set, we compared our method with K-means. We implemented the K-means with SAS procedure “FASTCLUS” (SAS Institute 1999). The performance of our method was comparable to or slightly inferior to the K-means method for the small clusters (Clusters 2–6). However, it did much better in recognizing genes in the larger cluster (Cluster 1) and had a much better overall correct classification rate (see table 1). With our method, the estimated parameters were

\[
\hat{\beta} = \begin{bmatrix}
-0.002 & -0.001 & -0.001 & 0.001 \\
-0.235 & -0.091 & -0.126 & -0.066 \\
0.34 & -0.059 & 0.092 & 0.089 \\
0.112 & 0.188 & -0.193 & -0.094 \\
-0.204 & -0.198 & 0.312 & 0.198 \\
0.149 & -0.301 & 0.102 & -0.2
\end{bmatrix}
\]

\[
\hat{\Sigma} = \begin{bmatrix}
0.00497 & 0.00001 & 0.00002 & 0.00004 \\
0.00001 & 0.00492 & 0.00007 & 0.00006 \\
0.00002 & 0.00007 & 0.00497 & -0.00002 \\
0.00004 & 0.00006 & -0.00002 & 0.00505
\end{bmatrix}, \text{ and }
\]

\[
\hat{\sigma}^2 = 0.0101, \text{ and }
\]

\[
\hat{\gamma} = [0.8006, 0.0401, 0.0392, 0.0398, 0.0401, 0.0402]^T, \text{ which were quite close to the true parameter values.}
\]

We conducted additional simulations with a wide range of parameters and sample sizes. Results of all simulation experiments showed the following expected trends (data not shown): 1) errors of cluster assignments decreased as the differences among the cluster means increased and 2) estimation errors of the parameters decreased as the number of genes within the cluster increased.

**FIG. 2.—** True and estimated mean gene expression profiles for each of the 5 simulated clusters of genes. Left panel, true profiles for the 5 clusters; Right panel, estimated profiles for the 5 clusters. In each plot, the 5 clusters (1–5) are color coded with black, blue, red, purple, and green, respectively.
Application to AD

Blalock et al. (2004) measured hippocampal gene expressions from 9 normal persons (controls) and 22 affected persons (cases), \( N = 9 + 22 = 31 \). Each subject had a disease severity measurement, MMSE, whose value varied from 2 (most severe) to 30 (normal). The MMSE is a continuous measure that has been reported as a reliable index of AD-related cognitive status at a given point in time (Miller et al. 2001). All the subjects were assigned to 1 of the 4 groups (control, incipient, less severe, and most severe), reflecting the different levels of AD severity based primarily on the MMSE score (Blalock et al. 2004). The original scores and the orthogonal polynomials of the MMSE for the 31 subjects are shown in table 2.

The expression levels of 22,286 genes were measured from 31 subjects (microarray chips). Here we choose a subset of 9,754 genes for analysis based on 2 criteria used by Blalock et al. (2004): 1) a gene was included only if it is a gene with known function (other than a hypothetical gene or an expressed sequence tag) and 2) a gene was included only if its probe set is detected on at least 4 chips of this study. We clustered the 9,754 genes based on the dependency of each gene’s expression levels on the MMSE scores of the 31 subjects, as described in the method section. The gene expression data were log transformed before the analysis.

We analyzed the data using a SAS program (available from supplemental material online) where \( d = 4 \) was chosen as the order of the polynomial regression. We tested the number of clusters from \( K = 2 \) to 20 using the BIC score and found that the BIC scores reached its maximum at \( K = 14 \). Therefore, the following paragraphs only report the results when \( K = 14 \).
The estimates of the cluster mean $\hat{\mu}_k$ and mixing proportion $\hat{\pi}_k$ for each cluster are presented in Table 3. The estimated covariance matrix was

$$\Sigma = \begin{bmatrix}
0.000232 & 0.0000487 & 0.0000213 & -0.0000484 \\
0.0000487 & 0.0000246 & 0.0000106 & -0.000013 \\
0.0000213 & 0.0000106 & 0.0000409 & -0.000005 \\
-0.0000484 & -0.000013 & -0.000005 & 0.000117
\end{bmatrix}.$$  

The estimated residual variance was $\sigma^2 = 0.17$. We assigned each gene to the cluster with the highest posterior probability ($\hat{\pi}_{ik}$). Table 3 also shows the actual numbers of genes assigned to each cluster.

In the simulated data, the expression profile of each cluster was predicted using equation (36), and the genespecific expression profile was predicted using equation (37) for each of the 14 clusters. The profiles of all genes for the 14 clusters are depicted in Figure 4.

The first cluster contained 93.5% of the genes whose expression do not change significantly with the phenotypic value. Clusters 8, 10, 11, 12, and 13 accounted for 0.65%, 0.15%, 0.63%, 0.03%, and 3.01% of all genes, respectively.

### Table 2

**AD Phenotypes (MMSE scores) and the Coefficients of Orthogonal Polynomials**

<table>
<thead>
<tr>
<th>Subject ID</th>
<th>MMSE Score ($z'_j$)</th>
<th>Rescaled MMSE ($z_j$)</th>
<th>Coefficients of Orthogonal Polynomials ($\psi$)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Order 1 ($\psi_1$)</td>
</tr>
<tr>
<td>1</td>
<td>26</td>
<td>0.714286</td>
<td>0.981567</td>
</tr>
<tr>
<td>2</td>
<td>26</td>
<td>0.714286</td>
<td>0.981567</td>
</tr>
<tr>
<td>3</td>
<td>29</td>
<td>0.928571</td>
<td>1.410138</td>
</tr>
<tr>
<td>4</td>
<td>29</td>
<td>0.928571</td>
<td>1.410138</td>
</tr>
<tr>
<td>5</td>
<td>28</td>
<td>0.857143</td>
<td>1.267281</td>
</tr>
<tr>
<td>6</td>
<td>28</td>
<td>0.857143</td>
<td>1.267281</td>
</tr>
<tr>
<td>7</td>
<td>27</td>
<td>0.785714</td>
<td>1.124424</td>
</tr>
<tr>
<td>8</td>
<td>30</td>
<td>1</td>
<td>1.552995</td>
</tr>
<tr>
<td>9</td>
<td>26</td>
<td>0.714286</td>
<td>0.981567</td>
</tr>
<tr>
<td>10</td>
<td>29</td>
<td>0.928571</td>
<td>1.410138</td>
</tr>
<tr>
<td>11</td>
<td>24</td>
<td>0.571429</td>
<td>0.695853</td>
</tr>
<tr>
<td>12</td>
<td>25</td>
<td>0.642857</td>
<td>0.83871</td>
</tr>
<tr>
<td>13</td>
<td>20</td>
<td>0.285714</td>
<td>0.124424</td>
</tr>
<tr>
<td>14</td>
<td>21</td>
<td>0.357143</td>
<td>0.267281</td>
</tr>
<tr>
<td>15</td>
<td>25</td>
<td>0.642857</td>
<td>0.83871</td>
</tr>
<tr>
<td>16</td>
<td>26</td>
<td>0.714286</td>
<td>0.981567</td>
</tr>
<tr>
<td>17</td>
<td>18</td>
<td>0.142857</td>
<td>-0.16129</td>
</tr>
<tr>
<td>18</td>
<td>17</td>
<td>0.071429</td>
<td>-0.30415</td>
</tr>
<tr>
<td>19</td>
<td>14</td>
<td>-0.14286</td>
<td>-0.73272</td>
</tr>
<tr>
<td>20</td>
<td>18</td>
<td>0.142857</td>
<td>-0.16129</td>
</tr>
<tr>
<td>21</td>
<td>15</td>
<td>-0.07143</td>
<td>-0.58986</td>
</tr>
<tr>
<td>22</td>
<td>17</td>
<td>0.071429</td>
<td>-0.30415</td>
</tr>
<tr>
<td>23</td>
<td>18</td>
<td>0.142857</td>
<td>-0.16129</td>
</tr>
<tr>
<td>24</td>
<td>15</td>
<td>-0.07143</td>
<td>-0.58986</td>
</tr>
<tr>
<td>25</td>
<td>7</td>
<td>-0.64286</td>
<td>-1.73272</td>
</tr>
<tr>
<td>26</td>
<td>4</td>
<td>-0.85714</td>
<td>-2.16129</td>
</tr>
<tr>
<td>27</td>
<td>5</td>
<td>-0.78571</td>
<td>-2.01843</td>
</tr>
<tr>
<td>28</td>
<td>11</td>
<td>-0.35714</td>
<td>-1.16129</td>
</tr>
<tr>
<td>29</td>
<td>11</td>
<td>-0.35714</td>
<td>-1.16129</td>
</tr>
<tr>
<td>30</td>
<td>2</td>
<td>-1</td>
<td>-2.447</td>
</tr>
<tr>
<td>31</td>
<td>2</td>
<td>-1</td>
<td>-2.447</td>
</tr>
</tbody>
</table>
These genes were mainly up regulated with the MMSE score. Clusters 6, 7, and 14 accounted for 0.16%, 0.25%, and 0.49% of all genes, respectively. These genes were mainly down regulated with the MMSE score. Clusters 2, 3, 4, 5, and 9, which accounted for 0.33%, 0.06%, 0.43%, 0.11%, and 0.16%, respectively, represented the genes whose expression changes with the phenotypic values in a nonlinear fashion.

To compare the predicted with the observed gene expressions, we randomly selected one gene from each cluster and presented the comparisons in figure 5. It shows that the predicted profiles do reflect the general trends indicated by the observed data.

Blalock et al. (2004) studied this data set using Pearson correlation analysis. Among the 9754 genes, 2852 of them (28.6%) were detected as differentially expressed (nominal $P < 0.05$) based on the correlation coefficients. In the AD data set, some of the genes classified into clusters 2–14 are not detected as differentially expressed by either correlation analysis or the linear association method.

Application to the Rat Fracture Healing Data

To demonstrate that our method can be applied to time-course gene expression data, we examined an experiment that studied rat fracture healing. The original microarray experiment was published by Li et al. (2005). In all, 21 rats were used in the study. A standard closed fracture was created in the right femur of each rat. Tissues were collected at different postfracture time points (0d, 1d, 4d, 7d, 14d, 21d, and 28d), each with 3 replicates. The expression levels of 12 434 genes were measured using Affymetrix Genechips. We picked a subset of 12 124 genes that expressed on at least 1 of the 21 arrays. The raw data were log transformed, and we took the average of the 3 replicates for the measurement of gene expression for each time point. The mean expression of each gene across the time course was removed by the linear contrast strategy described earlier.

Luan and Li (2003) used a mixed-effect model with B-splines to analyze gene expression in yeast cell cycle and identified 5 major clusters. We used the method of Luan and Li (2003) to fit cubic B-splines with 2 internal knots to classify gene expression profiles in the rat fracture healing experiment. The algorithm for generating the B-spline basis was initially described by De Boor (1978). Unlike the yeast cell cycle data set, in which most genes were differentially expressed, there were only a small proportion of genes in the rat fracture healing experiment whose expression were significantly altered in the time course.

We varied the number of clusters from $K = 5$ to 70 incremented by 5 and found that $K = 35$ gave the maximum BIC score. Among the 35 clusters, one large cluster contained 10 235 genes (84.4%) whose expression did not change during the time course, whereas the expression of genes in the other clusters evolved with time in different patterns. See Supplemental Material C (Supplementary Material online) for the estimated cluster mean and mixing proportion of each of the 35 clusters.

In order to compare the results of B-splines and polynomials, we reclassified the data with orthogonal polynomials of order $d = 3$. Again, we varied the number of clusters from $K = 5$ to 70 incremented by 5 and found that $K = 30$ gave the maximum BIC score. Similar to the B-splines approach, the majority of the genes (10 017 out of 12 124 genes) were assigned to a large cluster with no apparent change in expression (neutral cluster), whereas a relatively small number of genes (2107 out of 12 124 genes) were assigned to other clusters (see Supplemental Material C for more information, Supplementary Material online). Among the genes placed in clusters other than the neutral one using either the B-splines or the polynomial approach, 1663 of them (79%) were identified as differing from a flat profile by both methods. Therefore, the degree of consistency between the 2 methods is quite high. When the number of time points of measurement is small (7 time points in this experiment) or the curves to be estimated are simple, the method of orthogonal polynomials is perhaps more preferable for biologists because it is simple and easy to understand. If the curves are arbitrarily complex, B-splines should be more preferable.

**Discussion**

In this paper, “linear association” refers to the first order (or straight line) association between 2 variables.

**Table 3**

The Estimated Cluster Mean ($\hat{\beta}_k$), the Mixing Proportion ($\pi_k$), and the Number of Genes Assigned to the $k$th Cluster ($\hat{w}_k$) for the Data set of AD

<table>
<thead>
<tr>
<th>Cluster ($k$)</th>
<th>$\hat{\beta}_1$</th>
<th>$\hat{\beta}_2$</th>
<th>$\hat{\beta}_3$</th>
<th>$\hat{\beta}_4$</th>
<th>$\hat{\pi}_k$</th>
<th>$\hat{w}_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.0124</td>
<td>-0.0246</td>
<td>-0.0077</td>
<td>0.00286</td>
<td>0.914</td>
<td>9123</td>
</tr>
<tr>
<td>2</td>
<td>0.035</td>
<td>0.0177</td>
<td>0.189</td>
<td>-0.112</td>
<td>$3.77 \times 10^{-3}$</td>
<td>32</td>
</tr>
<tr>
<td>3</td>
<td>-0.0236</td>
<td>-0.113</td>
<td>0.172</td>
<td>0.206</td>
<td>6.51 $\times 10^{-4}$</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>0.0912</td>
<td>0.0706</td>
<td>-0.166</td>
<td>0.0626</td>
<td>4.98 $\times 10^{-3}$</td>
<td>42</td>
</tr>
<tr>
<td>5</td>
<td>-0.106</td>
<td>-0.228</td>
<td>0.143</td>
<td>-0.00372</td>
<td>1.05 $\times 10^{-3}$</td>
<td>11</td>
</tr>
<tr>
<td>6</td>
<td>-0.101</td>
<td>-0.183</td>
<td>-0.0995</td>
<td>-0.099</td>
<td>1.96 $\times 10^{-3}$</td>
<td>16</td>
</tr>
<tr>
<td>7</td>
<td>-0.286</td>
<td>-0.00798</td>
<td>-0.135</td>
<td>-0.0744</td>
<td>2.63 $\times 10^{-3}$</td>
<td>24</td>
</tr>
<tr>
<td>8</td>
<td>0.153</td>
<td>-0.153</td>
<td>0.0819</td>
<td>-0.0634</td>
<td>8.05 $\times 10^{-3}$</td>
<td>63</td>
</tr>
<tr>
<td>9</td>
<td>-0.0839</td>
<td>0.159</td>
<td>0.176</td>
<td>0.101</td>
<td>1.75 $\times 10^{-3}$</td>
<td>16</td>
</tr>
<tr>
<td>10</td>
<td>0.28</td>
<td>-0.0858</td>
<td>0.0908</td>
<td>-0.2</td>
<td>1.72 $\times 10^{-3}$</td>
<td>15</td>
</tr>
<tr>
<td>11</td>
<td>0.246</td>
<td>-0.0792</td>
<td>-0.0473</td>
<td>0.0858</td>
<td>7.38 $\times 10^{-3}$</td>
<td>61</td>
</tr>
<tr>
<td>12</td>
<td>0.435</td>
<td>0.227</td>
<td>-0.0404</td>
<td>0.00895</td>
<td>3.19 $\times 10^{-4}$</td>
<td>3</td>
</tr>
<tr>
<td>13</td>
<td>0.177</td>
<td>-0.00706</td>
<td>0.0134</td>
<td>-0.0512</td>
<td>0.0436</td>
<td>194</td>
</tr>
<tr>
<td>14</td>
<td>-0.197</td>
<td>-0.0922</td>
<td>-0.0632</td>
<td>0.0204</td>
<td>7.99 $\times 10^{-3}$</td>
<td>48</td>
</tr>
</tbody>
</table>
Nonlinear kinetics of gene regulation has been investigated by many authors (e.g., Kim et al. 2000; Spieth et al. 2005). The gene regulatory network is known to be a complex system. Multiple effects, such as the autoregulation (positive or negative self-feedbacks) of a gene, the cooperativity, and competition between genes, may result in nonlinear associations between the expression levels of multiple genes or between the gene expression levels and the phenotypic traits. Motivated by these observations, we have developed a new method, which is capable of describing

![Gene expression profiles for all genes in the 14 clusters estimated from the AD microarray data.](Fig. 4)
nonlinear associations between gene expression and phenotype that are not detected by methods using linear associations. Our novel approach to clustering microarray data is based on a nonlinear association of gene expression with a continuous phenotype. This new method is a useful tool for detecting genes associated with continuous phenotypes and to explore the patterns of gene-trait association. We verified our algorithm and implementation with simulated data and then applied the methods to 2 microarray data sets. For the analysis of AD, we found that the majority of the genes had expression profiles that did not show association with MMSE score. Some genes had expression profiles that showed interesting nonlinear association with MMSE score.

We also demonstrate that our method can be applied to time-course experiments. Luan and Li (2003) used a mixed-effects model to analyze the time-course gene expression data for yeast cell cycle (Spellman et al. 1998). They used B-splines basis of the time points as the design matrix. We used orthogonal polynomials with the time points as the "phenotype" and then compared our result with that of Luan and Li (2003) for the time-course expression data of fracture healing in rats (Li et al. 2005). We classified the data with both B-splines basis and orthogonal polynomials. In both cases, we found that the majority of the genes were expressed at relatively constant levels, whereas small subsets of the genes showed patterns that changed over the time course. Among the genes classified as different over time with the B-splines, 88% (1663/1889) were also classified as different over time with the polynomials, and 79% (1663/2107) of the genes classified as different over time with polynomials were also classified as different over time with B-splines. This result indicates a high degree of consistency between the 2 methods.

A B-spline can be described as a piecewise polynomial spline. Instead of fitting one polynomial curve across the data point, it fits different curves in different regions.
B-splines require specification of series of internal knots (breaking points), for example, time points or phenotypic values. The polynomial curves are, roughly speaking, equivalent to B-splines with no internal knots. There is no best way to choose the number of locations of the knots, and the choice may have an impact on the results (De Boor 1978). The flexibility of B-splines makes them particularly suitable to fit complex or periodical curves, for example, the curve of a sine function. To do so effectively, however, a relatively large number of measurements must be made on each
subject. Current gene expression data sets do not have the sampling density to effectively estimate complex curves.

Generally speaking, observations with \( n \) treatments can be used to fit polynomial curves up to \((n-1)\)th order, and a higher order always results in better fit. For real data sets, because of the underlying complexity of the data structures, the optimal number of clusters could be large, which results in a great number of parameters to fit. The optimal number of clusters tends to increase with the order of the polynomials used in the analysis. So, increasing the order even by one may result in a much more complex model without much additional information. We selected \( d = 4 \) for the AD data set and \( d = 3 \) for the fracture healing data. We examined other values of \( d \) around 4 for the AD data and other \( d \) values around 3 for the fracture healing data and found that the results were virtually the same, indicating that these were reasonable choices for the order of the polynomial in these data sets.

Theoretically, the optimal order \((d)\) can be found using the BIC score because it determines the dimension of the model. In practice, however, it is tedious because the number of clusters also determines the dimension of the model. Consider the combination of the order of polynomials and the number of clusters, the potential number of models for evaluation will quickly reach the limit of our capability. Therefore, we suggest that choosing \( d = 3 \) or \( d = 4 \) will be sufficient to describe the complexity of microarray data currently available. If the association is linear or quadratic, choosing \( d > 2 \) will overfit the model. This does not present a serious problem because the higher order regression coefficients will most likely have estimates close to zero, and the fitted curve will be effectively linear or quadratic. Recall that the shape of a curve is determined by both the order and the regression coefficients. Two drastically different curves may be described by polynomials of the same order but with quite different regression coefficients.
In analysis of the AD microarray data, we found that the EM algorithm was sensitive to the initial values of the parameters. This is a well-known phenomenon that the EM algorithm is a “hill-climbing” approach. The algorithm can only guarantee the reach of a local maximum. In attempt to find the global maximum, we explored many different initial values and chose the solution with the highest converged likelihood value.

In summary, we proposed a new method to identify genes associated in a nonlinear fashion with a quantitative trait using microarray gene expression data. In order to make inferences about the evolutionary processes, the non-linear relationship between gene expression and phenotype may be of interest in exploring functional and molecular evolution.

Supplementary Material

Supplementary Materials A, B, and C are available at Molecular Biology and Evolution online (http://www.mbe.oxfordjournals.org/).

Acknowledgments

We are grateful to Drs Blalock and Landfield at the University of Kentucky for providing the microarray data and the AD phenotypes. We also thank Dr Xinmin Li at the University of Chicago for providing us with the complete microarray data of the rat fracture healing experiment. We are particularly grateful to the Associate Editor Laurent McIntyre and 3 anonymous reviewers for their constructive comments and suggestions that significantly improved the presentation of the manuscript. The research was supported by the National Institutes of Health grant R01-GM55321 and National Science Foundation grant DBI-0345205 to S.X.

Funding to pay the Open Access publication charges for this article was provided by the National Institutes of Health grant R01-GM55321.

Literature Cited


Lauren McIntyre, Associate Editor
Accepted May 16, 2006