

# CATS: Clustering After Transformation and Smoothing

Nicoleta Serban and Larry Wasserman<sup>1</sup>

Department of Statistics  
Carnegie Mellon University

August 11, 2004

CATS – Clustering After Transformation and Smoothing – is a technique for nonparametrically estimating and clustering a large number of curves. Our motivating example is a genetic microarray experiment but the method is very general. The method includes: transformation and smoothing multiple curves, multiple nonparametric testing for screening out flat curves, clustering curves with similar shape, and nonparametrically inferring the clustering estimation error rate.

Key words and phrases: Multiple testing, False discovery rate, clustering, clustering error rate, smoothing, genetic microarrays.

## 1 Introduction

CATS – Clustering After Transformation and Smoothing – is a technique for nonparametrically estimating and clustering a large number of curves (or profiles). We first screen out curves which are nearly flat, smooth the remaining curves, and then cluster the smoothed curves. A novel feature of our method is that we estimate the error due to the fact that we are clustering the estimated curves rather than the true curves.

CATS is quite general but, for clarity, we will discuss the method in the context of microarray experiments. This problem is challenging because of the large number of expression profiles. Our motivating example is a genetic microarray experiment conducted at the University of Pittsburgh. This experiment produced time series of gene expression levels for 5355 genes over 15 time points.

There is now a substantial literature on genetic microarrays on various topics such as clustering (Eisen et al 1998; Hastie et al 2000; Bar-Joseph,

---

<sup>1</sup>Research supported by NIH Grant R01-CA54852-07, NIH grant number MH57881, NSF Grant DMS-98-03433 and NSF Grant DMS-0104016. The authors are grateful to David Peters and Rob O’Doherty for allowing them to use the data from the fat cell experiment, Dan Handley, Clark Glymour, Peter Spirtes, Richard Scheines, Greg Cooper and the other members of the CMU-University of Pittsburgh Gene group for their invaluable input, and the referees and associate editor for helpful comments.

Gerber, Gifford and Jaakkola 2002; Wakefield, Zhou, Self 2002) and multiple testing (Dudoit et al 2000; Efron, Storey and Tibshirani 2001; Newton et al 2001). For related work on curve clustering in the context of microarray data see Bar-Joseph, Gerber, Gifford and Jaakkola(2002) and Wakefield, Zhou, Self (2002).

## 2 The Model

We consider data of the form,

$$Y_{ij} = f_i(t_{ij}) + \sigma_i \epsilon_{ij}, \quad i = 1, \dots, N, \quad j = 1, \dots, m. \quad (1)$$

where  $\mathbb{E}(\epsilon_{ij}) = 0$ . Thus,  $Y_{ij}$  is the  $j^{\text{th}}$  observation on the  $i^{\text{th}}$  curve. In the examples of interest,  $N$  is much larger than  $m$ . In the microarray setting,  $Y_{ij}$  is the log gene expression of gene  $i$  at time  $t_{ij}$ .

We assume that the curves  $f_i$  belong to a class of smooth functions  $\mathcal{F}$  as defined in the appendix. Let  $\phi_1, \phi_2, \dots$  be an orthonormal basis for  $\mathcal{F}$  and write

$$f_i(t) = \sum_{j=1}^{\infty} \theta_{ij} \phi_j(t) \quad (2)$$

where

$$\theta_{ij} = \int f_i(t) \phi_j(t) dt. \quad (3)$$

Generally, we cannot estimate more parameters than data points so rather than estimate  $f_i$  we actually estimate its projection  $\sum_{j=1}^m \theta_{ij} \phi_j(t)$  onto the first  $m$  basis functions which we continue to denote by  $f_i$ . We estimate  $f_i$  by

$$\widehat{f}_i^J(t) = \sum_{j=1}^J \widehat{\theta}_{ij} \phi_j(t) \quad (4)$$

where the estimates  $\widehat{\theta}_{ij}$  and the choice of smoothing parameter  $1 \leq J \leq m$  are described below. We call a curve  $f_i$  *null* or *inactive* if  $f_i$  is constant as a function of  $t$ . Otherwise,  $f_i$  is *non-null* or *active*. Let  $\mathcal{A}$  denote the set of active curves.

Let  $\theta_i = (\theta_{i1}, \dots, \theta_{im})$  be the vector of coefficients for curve  $f_i$ . We will view the  $(\theta_i, \sigma_i)$ 's as a random draw from some distribution  $\mathbb{P}$ . We assume that  $\mathbb{P}$  has compact support. In a slight abuse of notation, we also use  $\mathbb{P}$  to denote the marginal law of the  $\theta_i$ 's.

### 3 Clusters of Curves

Since our goal is to cluster the curves, we need a measure of the efficacy of a set of clusters. Let  $\mathcal{C} = \{f_1, \dots, f_N\}$  denote a finite set of curves. A clustering algorithm may be viewed as a map

$$T : \mathcal{C} \times \mathcal{C} \rightarrow \{0, 1\}$$

where

$$T(f, g) = \begin{cases} 1 & \text{if } f \text{ and } g \text{ are in the same cluster} \\ 0 & \text{otherwise.} \end{cases}$$

The cluster map  $T$  induces a partition  $\{\mathcal{C}_1, \dots, \mathcal{C}_k\}$  of  $\mathcal{C}$  where two curves  $f$  and  $g$  are in the same partition element if and only if  $T(f, g) = 1$ . The numbering of the partition elements is arbitrary. Generally, one uses an algorithm that can produce  $k$  clusters for any given  $k$ . Thus, let us write  $T_k$  for the cluster map that yields  $k$  clusters. For example,  $T_k$  might be the output of the  $k$ -means clustering algorithm.

We address two different questions for the efficacy of the clusters. The first is: how good are the estimated clusters? The second is: how close is the clustering using the estimated curves  $\widehat{\mathcal{C}} = \{\widehat{f}_1, \dots, \widehat{f}_N\}$  to the clustering using the true curves  $\mathcal{C} = \{f_1, \dots, f_N\}$ ? The first concerns cluster quality; the second concerns estimation error. We will define two parameters,  $\Omega$  and  $\eta$  associated with these questions.

#### 3.1 Cluster Quality

Many such criteria have been proposed to measure cluster quality. We shall use the following. Suppose that  $\mathcal{C}_1, \dots, \mathcal{C}_k$  are clusters, that is, they form a partition of  $\mathcal{C}$ . Define the cluster *quality*

$$\Omega = \min_{1 \leq j \leq k} \min_{f, g \in \mathcal{C}_j} \rho(f, g)$$

where

$$\rho(f, g) = \frac{\int (f(x) - \bar{f})(g(x) - \bar{g}) dx}{\sqrt{\int (f(x) - \bar{f})^2 dx \int (g(x) - \bar{g})^2 dx}}$$

and  $\bar{f} = \int f(x) dx$ .

Thus,  $\rho(f, g)$  is the Pearson correlation between the curves  $f$  and  $g$  and  $\Omega$  measures the worst pairwise correlation over all the clusters. Note that

$-1 \leq \Omega \leq 1$  and  $\Omega = 1$  if and only if all the curves in each cluster are proportional to each other. We write  $\Omega(k)$  if we want to emphasize the dependence on the number of clusters  $k$ .

In the Fourier domain we can rewrite  $\Omega$  as follows. Let  $f = \sum_j a_j \phi_j$  and  $g = \sum_j b_j \phi_j$ . From  $a = (a_1, a_2, \dots)$  define a new vector  $\tilde{a} = (\tilde{a}_2, \tilde{a}_3, \dots)$  obtained by discarding  $a_1$  and normalizing:

$$\tilde{a}_j = \frac{a_j}{\sqrt{\sum_{j=2}^m a_j^2}}, \quad j \geq 2. \quad (5)$$

Define  $\tilde{b} = (\tilde{b}_1, \tilde{b}_2, \dots)$  similarly. Then,

$$\rho(f, g) = 1 - \frac{\|\tilde{a} - \tilde{b}\|^2}{2}. \quad (6)$$

Hence, correlation clustering in function space is equivalent to Euclidean clustering in the Fourier domain, after the transformation  $a \mapsto \tilde{a}$ .

Generally,  $\Omega(k)$  will increase as  $k$  increases. We will examine  $\Omega$  as a function of  $k$ . If we want to choose one value of  $k$ , we take the smallest  $k$  such that  $\Omega \geq 1 - \epsilon$  for some user-specified  $\epsilon$ . This gives the smallest number of clusters that guarantees that all curves within a cluster are  $(1 - \epsilon)$ -similar.

### 3.2 Estimation Error

Regarding estimation error, we proceed as follows. Let  $\mathcal{C} = \{f_1, \dots, f_n\}$  denote the true curves and let  $\hat{\mathcal{C}} = \{\hat{f}_1, \dots, \hat{f}_n\}$  denote the estimated curves. Let  $T$  and  $\hat{T}$  denote the corresponding clustering maps. Various methods have been proposed to compare two clusterings. See, for example, Rand (1971), Fowlkes and Mallows (1983), and Meilă (2000).

We define the *clustering estimation error rate* for  $k$  clusters  $\eta(k)$  by

$$\eta(k) = \frac{1}{\binom{N}{2}} \sum_{r < s} I\left(T_k(f_r, f_s) \neq \hat{T}_k(\hat{f}_r, \hat{f}_s)\right). \quad (7)$$

Thus,  $\eta$  is the fraction of all pairs which are either incorrectly put in the same cluster or are incorrectly put in separate clusters. We write  $\eta(k)$  to indicate the dependence on the number of clusters  $k$ . The clustering estimation error rate can be expressed as one minus the Rand index (Rand, 1971).

### 3.3 k-means clustering

Since we make use of k-means clustering, we briefly review some facts about this method. Let  $\theta_1, \dots, \theta_N \sim \mathbb{P}$  where each  $\theta_i$  is a vector in  $\mathbb{R}^d$ . The  $k$ -means algorithm searches for the  $k$  vectors  $a = \{a_1, \dots, a_k\}$  that minimize

$$\frac{1}{n} \sum_{i=1}^n \min_{1 \leq j \leq k} \|\theta_i - a_j\|^2.$$

This is equivalent to minimizing

$$W(a, \mathbb{P}_N) = \int \min_{a \in \mathcal{A}} \|\theta - a\|^2 d\mathbb{P}_N(\theta)$$

over all possible choices of sets  $\mathcal{A}$  containing  $k$  or fewer points, where  $\mathbb{P}_N$  is the empirical measure putting mass  $1/N$  on each  $\theta_i$ . The centers  $a$  determine a *tessellation*  $\{\mathbb{A}_1, \dots, \mathbb{A}_k\}$  where  $\theta \in \mathbb{A}_j$  if  $\theta$  is closer to  $a_j$  than any other center  $a_i$ .

Pollard (1981) shows, under weak conditions, that the minimizer  $a = (a_1, \dots, a_k)$  converges almost surely to the population minimizer  $\bar{a}$  of  $W(a, \mathbb{P})$ . Also, Pollard (1982) shows that

$$\sqrt{N}(a - \bar{a}) \rightsquigarrow N(0, S)$$

for some  $kd \times kd$  non-singular matrix  $S$ .

## 4 CATS

Our strategy for analyzing data of this form involves a series of steps summarized below; see also Figure 1.

### 4.1 Transforming the Data

Without loss of generality, assume that all time points lie in  $[0, 1]$ . We transform the data into the Fourier domain as follows. Let

$$\phi_1(t) \equiv 1, \quad \text{and} \quad \phi_j(t) = \sqrt{2} \cos((j-1)\pi t), j \geq 2$$

### Summary of CATS

1. Transform:  $(Y_{i1}, \dots, Y_{im}) \longrightarrow (\hat{\theta}_{i1}, \dots, \hat{\theta}_{im})$ .
2. Smooth:  $\hat{f}_i(t) = \sum_{j=1}^J \hat{\theta}_{ij} \phi_j(t)$ .
3. Screen: test  $H_{0i} : f_i(t) = \text{constant}$  and remove unrejected cases.
4. Cluster: apply clustering algorithm to coefficient estimates.
5. Error rate: estimate error due to using estimated curves.

Figure 1: Summary of the steps in the CATS procedure.

denote the cosine basis. Define the  $N \times m$  matrix

$$\Phi = \begin{pmatrix} \phi_1(t_{11}) & \phi_1(t_{12}) & \cdots & \phi_m(t_{1m}) \\ \phi_1(t_{21}) & \phi_1(t_{22}) & \cdots & \phi_m(t_{2m}) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_1(t_{N1}) & \phi_1(t_m) & \cdots & \phi_m(t_{Nm}) \end{pmatrix}.$$

(If necessary, perform a Gram-Schmidt orthogonalization on the columns of  $\Phi$  to make the columns orthogonal.) Let  $\hat{\theta}_i = (\hat{\theta}_{i1}, \dots, \hat{\theta}_{im})$

$$\hat{\theta}_{ir} = \frac{1}{m} \sum_{j=1}^m Y_{ij} \phi_r(t_{ij}).$$

Under weak conditions, we have that  $\hat{\theta}_i \approx N(\theta_i, m^{-1}\Sigma_i)$  where  $\Sigma_i$  is diagonal with  $(j, j)$  element  $\sigma_j^2$ .

## 4.2 Smoothing

The function  $\hat{f}_i^J(t) = \sum_{j=1}^J \hat{\theta}_{ij} \phi_{jt}$  is an estimate of the  $i^{\text{th}}$  profile. The parameter  $J$  controls the amount of smoothing. The optimal amount of smoothing will vary from curve to curve. Rather than trying to find an optimal amount

of smoothing separately for each curve, instead we will find a single smoothing parameter that does reasonably well for all the curves. Let

$$R_i(J) = \mathbb{E} \left( \int (\hat{f}_i^J(t) - f_i(t))^2 dt \right)$$

denote the risk of the estimate of  $f_i$ . We will estimate the risk function for each curve. Then we choose  $J$  to minimize the total regret, the risk minus the minimum risk for each curve. We restrict  $J$  to be less than  $m$ . Here are the steps.

We estimate the variance  $\sigma_i^2$  using the high component variance estimator of Beran and Dümbgen (1998):

$$\hat{\sigma}_i^2 = \frac{m}{m-L} \sum_{j=L+1}^m \hat{\theta}_{ij}^2$$

where  $L$  is an integer which we take to be  $L = m/3$ . Now define

$$\hat{R}_i(J) = \frac{J\hat{\sigma}_i^2}{m} + \sum_{j=J+1}^m \left( \hat{\theta}_{ij}^2 - \frac{\hat{\sigma}_i^2}{m} \right)_+ \quad (8)$$

which is an estimate of the risk of  $\hat{f}_i^J$  (Beran and Dümbgen, 1998). Define the regret

$$\hat{r}_i(J) = \hat{R}_i(J) - \min_{1 \leq k \leq m} \hat{R}_i(k) \quad (9)$$

which measures how much risk is sacrificed for curve  $f_i$  if smoothing parameter  $J$  is used. Define the total regret

$$t(J) = \sum_{i=1}^n \hat{r}_i(J). \quad (10)$$

Finally, take

$$\hat{J} = \operatorname{argmin}_J t(J).$$

**Remark 1** *If replications are available at each time or if  $m$  is very large, then we can estimate a time varying variance function. Otherwise, there is little choice but to assume a constant variance across time. We also assume that the  $\epsilon_{ij}$  are independent. If information is available about correlation of the residuals, then, as in any regression, this information can be included by using weighted regression, time series methods etc.*

### 4.3 Screening Out Flat Curves

In many cases, there will be a number of curves that are flat, or close to flat. It is inefficient to include these in the clustering algorithm. Thus, we screen out flat curves by testing

$$H_{0i} : f_i(t) = c_i \quad \text{for some constant } c_i.$$

If  $H_{0i}$  is true then  $\sum_{j=1}^m \theta_i^2 = 0$ . This suggests the test statistic

$$T_i = \sum_{j=1}^m \hat{\theta}_{ij}^2.$$

We reject the null hypothesis for large value of  $T_i$ . We compute a permutation-based  $p$ -value as follows. For each  $i$ , permute the time points  $t_{ij}$  and compute  $\hat{\theta}_i^*$  from the permuted data. Let  $T_i^* = \sum_{j=1}^m (\hat{\theta}_{ij}^*)^2$ . Repeat  $B$  times to get statistics  $T_{i1}^*, \dots, T_{iB}^*$ . The estimated  $p$ -value is

$$\hat{P}_i = \frac{1}{B} \sum_{b=1}^B I(T_{ib}^* > T_i).$$

In our examples we used  $B = 10,000$ .

To correct for the multiplicity problem we use the Benjamini-Hochberg (1995) method. Let  $P_{(1)}, \dots, P_{(n)}$  denote the ordered  $p$ -values and define  $P_{(0)} = 0$ . We reject  $H_{0i}$  if  $P_i \leq T$  where  $T = P_{(j)}$  and

$$j = \max \left\{ i : P_{(i)} \leq \frac{i\alpha}{n} \right\}. \quad (11)$$

This method controls the expected fraction of false discoveries to be less than or equal to  $\alpha$ . See Benjamini and Hochberg (1995). Finally, we set  $\hat{\mathcal{A}} = \{i : \hat{P}_i \leq T\}$ . In our analysis, the significance level is  $\alpha = .05$ . The FDR procedure assumes independent test statistics but the gene expression levels tend to be correlated. However, as shown in Storey and Tibshirani (2002), the method works well even in the presence of dependence.

### 4.4 Confidence Set for $f_i$

We use the method in Beran and Dümmbgen (1998) for constructing a confidence ball  $\mathbb{B}_i$  for  $f_i$ . Fix  $\alpha > 0$  and let

$$\mathbb{B}_i = \left\{ (\theta_{i1}, \dots, \theta_{im}) : \sum_{j=1}^m (\theta_{ij} - \hat{\theta}_{ij})^2 \leq s_i^2 \right\} \quad (12)$$

where

$$s_i^2 = \frac{z_\alpha \widehat{\tau}_i}{\sqrt{m}} + \widehat{R}_i,$$

$z_\alpha$  is the  $\alpha$  quantile of the standard normal and  $\widehat{\tau}_i$  is given in the Appendix. The corresponding confidence ball for  $f_i$  is

$$\left\{ \sum_{j=1}^m \theta_{ij} \phi_j(x) : \theta \in \mathbb{B}_i \right\}.$$

For notational convenience, the confidence ball for  $f_i$  will also be denoted by  $\mathbb{B}_i$ . The next theorem follows directly from the results of Beran and Dümbgen.

**Theorem 2** *Let  $\mathcal{F}_\beta(c)$  denote a Sobolev space of order  $\beta$  and radius  $c$ . Then, for any  $\beta > 1/2$  and any  $c > 0$ ,*

$$\liminf_{m \rightarrow \infty} \inf_{f_1, \dots, f_N \in \mathcal{F}_\beta(c)} \mathbb{P} \left( f_i \in \mathbb{B}_i \text{ for all } i = 1, \dots, N \right) \geq 1 - \alpha.$$

Recall the mapping  $\theta \mapsto_M \widetilde{\theta}$  (see Section 3.1). The set

$$\widetilde{\mathbb{B}}_i = \{ \widetilde{\theta} : \theta_i \in \mathbb{B}_i \} \tag{13}$$

is then a confidence set for  $\widetilde{\theta}$ .

We can use the confidence balls  $\mathbb{B}_i$  to further screen out flat curves by removing curve  $i$  if  $(0, 0, \dots) \in \mathbb{B}_i$ .

## 4.5 Clustering

We want to cluster curves with similar shape. In the microarray setting for example, genes with similar expression profiles are co-expressed gene. Co-expressed genes are likely to be co-regulated and hence co-expression can suggest functional pathways and interactions between genes.

For  $r, s \in \widehat{\mathcal{A}}$  define

$$d(r, s) = \sum_{j=2}^J (\widetilde{\theta}_{rj} - \widetilde{\theta}_{sj})^2$$

where  $J$  is the smoothing parameter,  $\widehat{\theta}_r$  and  $\widehat{\theta}_s$  are the cosine transforms for the curves  $r$  and  $s$ , and  $\widehat{\theta} \mapsto \widetilde{\theta}$  is the transform described in Section 3.1. Now we simply apply the  $k$ -means clustering algorithm with the distance defined above. Any other clustering method could also be used. We shall see that smoothing and screening improves the clustering.

## 4.6 Estimating the Clustering Error Rate

In the following, we provide a confidence interval for the clustering error rate

$$\eta = \frac{1}{\binom{N}{2}} \sum_{r < s} I\left(T(f_r, f_s) \neq \widehat{T}(\widehat{f}_r, \widehat{f}_s)\right). \quad (14)$$

We give a one-sided confidence bound for the clustering error rate in the next theorem which follows easily from Theorem 1.

**Theorem 3** *Suppose that  $\mathbb{B}_i$  is a  $1 - (\alpha/N)$  confidence set for  $f_i$ . Let*

$$\overline{\eta} = \frac{1}{\binom{N}{2}} \sum_{r < s} \max_{f \in \mathbb{B}_r, g \in \mathbb{B}_s} I_{rs} \quad (15)$$

where

$$I_{rs} = I\left(T(f, g) \neq T(\widehat{f}_r, \widehat{f}_s)\right). \quad (16)$$

Then,

$$\mathbb{P}(\eta \in [0, \overline{\eta}]) \geq 1 - \alpha. \quad (17)$$

Computing (15) is very hard because we need to compute  $I_{rs}$  for all  $N(N - 1)$  pairs. Computing this for even a single pair is hard. We now find an approximation to  $\overline{\eta}$  that involves less computation.

Recall that  $k$ -means clustering produces a set of cluster centers  $a_1, \dots, a_k$ . This, in turn, produces the Voronoi tessellation  $\{\mathbb{A}_1, \dots, \mathbb{A}_k\}$  where  $f \in \mathbb{A}_j$  if  $f$  is closer to  $a_j$  than any other cluster center. In this case,  $T(f, g) = 1$  if and only if  $f$  and  $g$  belong to the same member of the tessellation. Similarly,  $\widehat{T}(\widehat{f}, \widehat{g}) = 1$  if and only if  $\widehat{f}$  and  $\widehat{g}$  belong to the same member of the tessellation of the estimated curves  $\{\widehat{\mathbb{A}}_1, \dots, \widehat{\mathbb{A}}_k\}$ . An upper bound for the clustering estimation error using only the tessellation of the estimated curves is given as follows. The proof is in Appendix.

**Theorem 4** *Assume the conditions of the main theorem in Pollard (1982). Let  $\{\mathbb{A}_1, \dots, \mathbb{A}_k\}$  be the tessellation based on the true curves and let  $\{\widehat{\mathbb{A}}_1, \dots, \widehat{\mathbb{A}}_k\}$  be the tessellation from the estimated curves. Let  $\widehat{\mathbb{A}}(i)$  denote the tessellation element containing  $\widehat{f}_i$ . Then*

$$\overline{\eta} \leq \widehat{\eta} + O_P\left(\frac{1}{\sqrt{m}}\right) \quad (18)$$

where

$$\hat{\eta} = \frac{|\mathcal{M}|}{N} \left( 1 + \frac{N - |\mathcal{M}|}{N - 1} \right) \quad (19)$$

and  $\mathcal{M} = \{i : \mathbb{B}_i \not\subset \hat{\mathbb{A}}(i)\}$ .

In the next section we describe an algorithm for computing  $\hat{\eta}$ .

## 4.7 Algorithm for Computing $\hat{\eta}$

To compute  $\hat{\eta}$  we need to compute  $|\mathcal{M}| = \sum_i \delta_i$  where

$$\delta_i \equiv I\left(\mathbb{B}_i \cap \hat{\mathbb{A}}_r \neq \emptyset \text{ for some } r \neq j(i)\right)$$

where  $\mathbb{B}_i$  is the confidence ball and  $j(i)$  is the index of the tessellation element containing  $\hat{f}_i$ .

The following algorithm can be used.

1. For  $r \neq j(i)$  do:
  - (a) Let  $\mathcal{H}$  be the hyperplane that bisects the line joining  $a_{j(i)}$ , the center of the tessellation element  $j(i)$ , and  $a_r$ , the center of an arbitrary tessellation element  $r \neq j(i)$ . Thus we construct the hyperplane,  $\mathcal{H}$ , which bisects the segment joining  $a_{j(i)}$  and  $a_r$ .
  - (b) Let  $\mathbb{B}_i$  be the confidence set of the profile  $f_i$ . Define  $\tilde{c}_i^{\min}$  the closest point in  $\tilde{\mathbb{B}}_i$  to the hyperplane of bisection.
  - (c) If  $d(a_{j(i)}, \tilde{c}_i^{\min}) > d(a_r, \tilde{c}_i^{\min})$  set  $\delta_{ir} = 1$ . Otherwise set  $\delta_{ir} = 0$ .
2. Set  $\delta_i = \max_{r \neq j(i)} \delta_{ir}$ .

We present an analytic solution to  $\tilde{c}_i^{\min}$  in the appendix. Having the coordinates of the closest point in  $\tilde{\mathbb{B}}_i$  to the bisection hyperplane it is not difficult to compute the distances  $d(C_{j(i)}, \tilde{c}_i^{\min})$ ,  $d(C_r, \tilde{c}_i^{\min})$  and thus  $\delta_i$ .

## 5 Example: Synthetic Data

We generate synthetic data according to the regression model:

$$Y_{ij} = f(t_{ij}) + \sigma\epsilon_{ij}$$

with  $j = 1, \dots, m$  with  $t_{ij} = j/m$ . The regression functions for  $f$  are:

$$\begin{aligned} F_1(t) &= \left(\frac{2-5t}{2}\right) \wedge \left(\left(\frac{5t-2}{3}\right)^2 + \sin\frac{5\pi t}{2}\right), \\ F_2(t) &= -F_1(t), \\ F_3(t) &= \cos(2\pi t), \\ F_4(t) &= -F_3(t). \end{aligned}$$

The synthetic data consist of 150 curves for each of the 4 curve shapes,  $F_1, F_2, F_3, F_4$ . We take the noise  $\epsilon_{ij}$  to be normally distributed. Among the 2000 curves in the synthetic data, 1400 are constant curves ( $Y_i = N_m(\nu_i, \Sigma_i)$ ). The 600 non-constant curves are defined over  $m = 15, 25, 50$  design points ( $Y_i = \mu_i(F(t_1), \dots, F(t_m)) + N_m(0, \Sigma_i)$ ) with  $F$  taking one of the four shapes  $F_1, F_2, F_3$  and  $F_4$ .

We consider low noise  $\sigma \sim \text{Unif}(0.4, 0.7)$  and high noise  $\sigma \sim \text{Unif}(1.0, 1.2)$ . In Table 1 we present the simulation results for low noise and for different numbers of time points. Column 2 provides the smoothing parameter estimated using our regret approach. Columns 3 to 5 provide the results of the screening step. For  $m \geq 15$  we discover a considerable number of non-constant curves. The clustering error rate for the true number of clusters  $k = 4$  is small for  $m = 50$  but not 0. The clustering estimation error rate is low for  $k = 4$  clusters but increases for  $k \geq 4$  (see Figure 8).

Table 2 shows the same results as Table 1 when  $\sigma$  is larger:  $\sigma_i \sim \text{Unif}(1.0, 1.2)$ . The number of recovered non-constant curves decreases for smaller  $m$ . For  $m = 15$ , we also simulate data with 3 replicates for each time point and take the mean over replicates. The number of recovered non-constant curves is considerable larger. Moreover, the clustering estimation error is large even for  $m = 50$ .

Evidently, the cluster estimation error is large except when  $m$  is large and the noise is small. The clustering results from CATS are shown in Figure 3 using four clusters. The figure shows the mean, the 5th and 95 percentile for the set of curves in the clusters. The clustering works well; on the other hand, accurate estimation of  $\eta$  requires large samples.

Time points	Smoothing parameter	Rejections	True positives	False positives	Clustering estimation error
15	3	529	510	19	0.559
25	3	618	593	25	0.44
50	3	608	596	12	0.11

Table 1: Results for 4 clusters (the true number of clusters) for different numbers of time points. The number of non-constant curves is 600 with different noise levels:  $\sigma \sim \text{Unif}(0.4, 0.7)$ .

Time points	Smoothing parameter	Rejections	True positives	False positives	Clustering estimation error
15	3	229	220	9	0.92
15 with 3 replicates	3	524	508	16	0.84
25	3	467	457	10	0.91
50	3	599	580	19	0.87

Table 2: The same as in Table 1 but  $\sigma \sim \text{Unif}(1.0, 1.2)$ . In addition, we add one row for  $m = 15$  time points and 3 replicates for each time point.

**Comparison With Other Methods.** Now we compare our method with the following alternatives:

1. No screening and no smoothing: hierarchical clustering applied to the raw data. ( $k$ -means is very expensive in this case so we use hierarchical clustering instead with one minus the Pearson correlation as the similarity measure.)
2. Smoothing but no screening:  $k$ -means applied to all the smoothed curves.
3. Screening but no smoothing: test raw data, eliminate non-rejected genes, then apply hierarchical clustering.

The results are shown in Figures 4, 5, and 6. The results suggest that CATS performs much better than the alternatives except for the case of smoothing but no screening. However, screening may help when estimating the number of clusters.

## 6 Example: Adipose Cell Experiment

In this section we apply CATS to data from an experiment on adipose cells. We examined data from spotted cDNA microarrays (Research Genetics, Carlsbad, California) experiment. The experiment was completed in February, 2002 at the University of Pittsburgh (Peters *et al.*). The spotted cDNA microarray experiment consists of a time-sequenced sampling of differential expression in mRNA from (3T3L1 cultured) adipose cells originally obtained from mice. These cells were treated with a drug, *troglistazone*, which is a member of a family of drugs known as thiazolidendiones (TZD's). In our experiment, the drug treatment of the cells lasted for different periods of time ranging from 0 hours to 24 hours.

The data consist of 15 measurements at different periods of time. For each measurement, target cDNA was obtained by mRNA extraction and reverse transcription (into complementary DNA). Then the cDNA targets were hybridized to microarrays. Each of the 15 hybridizations produced images, which were processed using the software package Pathways 3. The main quantity of interest reported by the image analysis methods is the intensity for each probe on each array.

After image processing, removing sources of experimental bias and variance, the gene expression data can be summarized by a matrix of intensities

smoothing parameter	J = 3	J = 4	J=5	J=15
# of rejections	291	238	264	0

Table 3: Number of rejection according to FDR for different values of smooth parameter - microarray data.

with 15 columns (corresponding to the number of arrays) and 3824 rows (corresponding to the number of probes). Each of the 3824 rows represents the expression profile over time of a DNA sequence.

The arrays need to be normalized to account for systematic differences between arrays. We use a global linear normalization that forces the log intensities to have median equal to zero at each array, making the median of the experiment array the same as that of the baseline array. This appears to perform well because we expect only a relative small proportion of the genes to vary significantly in expression between mRNA samples [23].

We find 291 active genes among the 3824 DNA sequences for the smoothing parameter  $\hat{J} = 3$ . The confidence balls of the non-constant gene expression profiles are in Figure 7 when clustering the smoothed expression profiles using  $k$ -means algorithm. The panel displays the second cosine transform component ( $\hat{\theta}_2$ ) vs. the third component ( $\hat{\theta}_3$ ) for the 291 significant expression profiles when the number of clusters is 2. We have clustered the observed/unsmoothed expression profiles using  $k$ -means. There is not a clear separation between the two clusters in the latter case.

For these observed data we compute the approximate upper bound of the clustering error rate  $\eta$  for  $J = 3, 4, 5$  as shown in figure 9. There is a large jump for the estimated clustering error bound from  $k = 2$  to  $k = 3$  suggesting that we cannot expect to estimate more than two clusters from these data.

We also evaluate the cluster quality,  $\Omega$ . The largest increase is from  $k = 1$  to  $k = 2$  clusters. In Figure 10, we provide the cluster quality for  $k = 1, \dots, 8$  clusters (upper plot) and the difference in cluster quality (bottom plot).

## 7 Discussion

The method introduced in this paper provides a means for efficiently clustering curves by smoothing the curves, screening out flat curves, and estimating the clustering estimation error. This method enables us to make inference on a large number of non-constant curves simultaneously.

We used a nonparametric test for filtering out the constant profiles. The

test was applied after transforming the profile data using cosine basis. The test proves to be powerful in identifying constant curves. This step is important, because a large number of noisy flat curves affects the clustering. We tried other tests but the nonparametric test presented in this paper proved to be the most powerful at a small number of design points.

Our method for clustering smoothed non-constant curves is an alternative to clustering by correlation. The correlation coefficient of two curves can be expressed as the Euclidean distance of the normalized cosine transforms in the Fourier space.

The last step is the inference on the cluster estimation. We estimated clustering error rate based on the fraction of all pairs put incorrectly in the same cluster or put incorrectly in different cluster. To the best of our knowledge, this approach to cluster estimation has not been considered previously. Note that the cluster error rate does not tell us how many clusters there are, but rather, how many we can actually estimate.

When  $m$  is large, the method appears to be quite effective both at producing meaningful clusters and at providing useful inferences about the clustering estimation errors. When  $m$  is small, the clustering still works well but the the inferences for the clustering estimation error is quite imprecise. In these cases, one might consider using prior information about the smoothness of the curves to put an upper bound on  $J$ . This will result in tighter confidence bounds on  $\eta$ .

In this paper we did not address non-constant variance or correlations within a curve. Non-constant variance has little effect on the estimating smooth curve but it does affect the confidence ball. When  $m$  is large, one can use nonparametric methods to estimate the variance as a function of time. If  $m$  is not large but there are replications at each time point, the replications can be used to estimate the variance at each time point. Without replication or large  $m$ , there is little one can do except assume a constant variance. The clustering algorithm might still yield useful clusters but the inferences for  $\eta$  may be overly optimistic.

Another assumption we made is independence of the residuals  $\epsilon_{ij}$  within each curve, over time. We did not assume independence between curves. Again, with large  $m$  or suitable background information, one can deal with correlation between times using standard time series methods. In our example, we expect dependence between curves but there is no reason to expect correlation of residuals over time as these are based on separate microarrays.

# Appendix

## Sobolev Ellipsoid

We assume that each  $f_i$  belongs to a Sobolev ellipsoid  $\mathcal{F}$  defined as follows. Let  $\phi_1, \phi_2, \dots$  be an orthonormal basis for  $L_2$ . The Sobolev ellipsoid  $\mathcal{F}_\beta(c)$  of order  $\beta$  and radius  $c$  is

$$\left\{ f(x) = \sum_{j=1}^{\infty} \beta_j \phi_j(x) : \sum_{j=1}^{\infty} \beta_j^2 j^{2\beta} \leq c^2 \right\}.$$

### Theorem 3 - Outline of Proof

Let  $\theta = (\theta_1, \dots, \theta_N)$  denote the true coefficient vectors and let  $\hat{\theta} = (\hat{\theta}_1, \dots, \hat{\theta}_N)$  denote the estimated coefficient vectors. Let  $a = (a_1, \dots, a_k)$  denote the cluster centers based on  $\theta$  and let  $\hat{a} = (\hat{a}_1, \dots, \hat{a}_k)$  denote the cluster centers based on  $\hat{\theta}$ . Let  $\mu$  denote the population minimizer of  $\int \min_{a \in \mathcal{A}} \|\theta - a\|^2 d\mathbb{P}(\theta)$ . Now,

$$a = \operatorname{argmin}_u \int m_a(\theta) d\mathbb{P}_N(\theta)$$

where

$$m_a(\theta) = \min_{1 \leq j \leq k} \|\theta - a_j\|^2$$

and  $\mathbb{P}_N$  is the empirical distribution of the  $\theta_i$ s. Note that  $a$  has the form of an M-estimator. Using Pollard (1992) or Theorem 5.23 of van der Vaart (1998), we have that

$$\int m_a(\theta) d\mathbb{P}(\theta) = \int m_\mu(\theta) d\mathbb{P}(\theta) + \frac{1}{2}(a - \mu)^T V_\mu(a - \mu) + o(\|a - \mu\|^2)$$

where  $V_\mu$  is the second derivative of the map  $a \mapsto R(a) \equiv \int m_a(\theta) d\mathbb{P}(\theta)$ .

Moreover,

$$a = \mu + \frac{S}{N} \sum_{i=1}^N Y_i + o_P\left(\frac{1}{\sqrt{N}}\right)$$

where  $S = -V^{-1}$  and  $Y_i = \dot{m}_\mu(\theta_i)$ . Recall that  $\hat{\theta}_i \approx N(\theta_i, m^{-1}\Sigma_i)$ . Treating this approximation as exact, we can view  $\hat{\theta}_i$  as a draw from  $\mathbb{P}_m = \mathbb{P} \oplus Q_m$  where  $\mathbb{P}_m(\hat{\theta} \in B) = \int Q_m(\hat{\theta} \in B | \theta) d\mathbb{P}(\theta)$  and  $Q_m$  denotes the  $N(\theta_i, m^{-1}\Sigma_i)$ .

With  $R_m(a) = \int m_a(\hat{\theta}) d\mathbb{P}_m(\hat{\theta})$ , we see that  $\mu_m \equiv \operatorname{argmin} R_m(a) = \mu + O(m^{-1/2})$ . Also,

$$\hat{a} = \mu_m + \frac{S_m}{N} \sum_{i=1}^N \hat{Y}_i + o_P\left(\frac{1}{\sqrt{N}}\right)$$

where  $S_m = S + O(m^{-1/2})$  and  $\hat{Y}_i = Y_i + O_P(m^{-1/2})$ . It follows that

$$\|a - \hat{a}\| = O_P\left(\frac{1}{\sqrt{m}}\right).$$

Recall that  $\mathbb{P}$  has support on a compact set. Restricted to this compact set,  $d_H(\mathbb{A}_j \Delta \hat{\mathbb{A}}_j)$  is a continuous function of  $\mu - \hat{a}$ , where  $\Delta$  denotes symmetric set difference and  $d_H$  is the Hausdorff distance. It then follows that

$$d_H(\mathbb{A}_j \Delta \hat{\mathbb{A}}_j) = O_P(1/\sqrt{m}). \quad (20)$$

Let  $\mathbb{A}(f)$  denote the member of the tessellation that contains  $f$  and similarly for  $\hat{\mathbb{A}}(\hat{f})$ . Now,

$$T(f, g) = \sum_{j=1}^k I(f \in \mathbb{A}_j) I(g \in \mathbb{A}_j) \quad \text{and} \quad \hat{T}(\hat{f}, \hat{g}) = \sum_{j=1}^k I(\hat{f} \in \hat{\mathbb{A}}_j) I(\hat{g} \in \hat{\mathbb{A}}_j).$$

Let us define

$$T(\hat{f}, \hat{g}) \equiv \sum_{j=1}^k I(\hat{f} \in \mathbb{A}_j) I(\hat{g} \in \mathbb{A}_j).$$

It then follows from (20) that

$$\hat{T}(\hat{f}, \hat{g}) = T(\hat{f}, \hat{g}) + O_P\left(\frac{1}{\sqrt{m}}\right).$$

Thus,

$$I\left(T(f_i, f_j) \neq \hat{T}(\hat{f}_i, \hat{g}_i)\right) = S(i, j) + O_P\left(\frac{1}{\sqrt{m}}\right)$$

where  $S(i, j) = I(T(f_i, f_j) \neq T(\hat{f}_i, \hat{g}_i))$ . Let  $\mathcal{M} = \{i : \mathbb{B}_i \not\subset \hat{\mathbb{A}}(i)\}$ . Note that

$$S(i, j) \leq \max(I(i \in \mathcal{M}), I(j \in \mathcal{M})).$$

Thus,

$$\begin{aligned}
\eta &= \frac{1}{\binom{N}{2}} \sum_{r < s} I(T(f_r, f_s) \neq \widehat{T}(\widehat{f}_r, \widehat{f}_s)) \\
&= \frac{1}{\binom{N}{2}} \sum_{r < s} S(i, j) + O_P\left(\frac{1}{\sqrt{m}}\right) \\
&= \frac{1}{N(N-1)} \sum_{r=1}^N \sum_{s \neq r} S(i, j) + O_P\left(\frac{1}{\sqrt{m}}\right) \\
&\leq \frac{1}{N(N-1)} \sum_{r \in \mathcal{M}} (N-1) + \frac{1}{N(N-1)} \sum_{r \in \mathcal{M}^c} \sum_{s \neq r} I(s \in \mathcal{M}) + O_P\left(\frac{1}{\sqrt{m}}\right) \\
&= \frac{|\mathcal{M}|(N-1)}{N(N-1)} + \frac{1}{N(N-1)} (N - |\mathcal{M}|)|\mathcal{M}| + O_P\left(\frac{1}{\sqrt{m}}\right) \\
&= \frac{|\mathcal{M}|}{N} \left(1 + \frac{N - |\mathcal{M}|}{N-1}\right) + O_P\left(\frac{1}{\sqrt{m}}\right). \quad \blacksquare
\end{aligned}$$

## $\tau^2$ estimation

Let  $\widehat{\theta}_{ij}$  be the estimate of  $\theta_{ij}$ . Drop the subscript  $i$  for convenience. Now,

$$\widehat{\theta}_j \approx \theta_j + \frac{\sigma}{\sqrt{m}} \epsilon_j$$

with  $j = 1, \dots, m$  and  $\epsilon_j \sim N(0, 1)$ . Recall that we estimate  $\sigma^2$  with the high component variance estimator:

$$\widehat{\sigma}^2 = \frac{1}{m-L} \sum_{i=L+1}^m \widehat{\theta}_i^2.$$

Define

$$\widehat{d} = \sqrt{m}(L(\widehat{\theta}, \theta) - \widehat{R}(\theta)) \tag{21}$$

where  $L(a, b) = \sum_j (a_j - b_j)^2$  is the mean squared loss and

$$\widehat{R} = \frac{J\widehat{\sigma}^2}{m} + \sum_{j=J+1}^m \left( \widehat{\theta}_j^2 - \frac{\widehat{\sigma}^2}{m} \right)_+$$

is Stein's unbiased risk estimator (SURE). According to Beran & Dümbgen (1998) the asymptotic distribution for  $d$  is  $N(0, \tau^2)$  where  $\tau^2$  is the limiting variance of  $\widehat{d}$ .

Substituting  $L$  and  $\widehat{R}$  in (21) we obtain

$$\widehat{d} = \sqrt{m} \left( \sum_{j=1}^m (\widehat{\beta}_j Z_j - \theta_j)^2 - \left( \sum_{j=1}^m \widehat{\sigma}^2 \beta_j^2 + \sum_{j=1}^m (Z_j^2 - \widehat{\sigma}^2) (1 - \beta_j^2) \right) \right).$$

We find that

$$\mathbb{V}(\widehat{d}) = \sum_{j=1}^m (2f_j - 1)^2 \left( 1 + \frac{1 - 2c_j}{m - J} \right) (4\theta_j^2 \sigma^2 + 2\sigma^4) + 4\sigma^2 \sum_{j=1}^m f_j \left( (1 - f_j) + \frac{2(2f_j - 1)c_j}{m - J} \right) \theta_j^2$$

where  $c_j$  is 1 for  $j \geq J + 1$  and 0 otherwise. Replace  $\sigma^2 \leftarrow \widehat{\sigma}^2$  (where  $\widehat{\sigma}^2$  is the high component variance of  $Z$ ) and  $\theta_j^2 \leftarrow (Z_j^2 - \widehat{\sigma}^2)_+$  to obtain the estimate  $\widehat{\tau}^2$  of  $\mathbb{V}(\widehat{d})$ . We have

$$\widehat{\tau}^2 = \sum_{j=1}^m \left( 1 + \frac{1 - 2c_j}{m - J} \right) (4(Z_j^2 - \widehat{\sigma}^2)_+ \widehat{\sigma}^2 + 2\widehat{\sigma}^4) + 4\widehat{\sigma}^2 \sum_{j=1}^k \frac{2c_j}{m - J} (Z_j^2 - \widehat{\sigma}^2)_+.$$

The estimated variance of  $\widehat{d}$  is different from the one presented in Beran (2000) because it takes into account the dependence between  $Z$  and  $\widehat{\sigma}$ .

## Analytic solution for $\bar{\eta}$

Let  $w$  and  $w_0$  the weights in the equation of the hyperplane which bisects the segment joining  $\mathbb{C}_r$  and  $\mathbb{C}_{j(i)}$ :

$$\mathcal{H} : h(x) = w^t x + w_0 = 0.$$

The bisection hyperplane is defined as follows. We know that the median point of the joining segment  $\mathbb{C}_r \mathbb{C}_{j(i)}$  is in the hyperplane  $\mathcal{H}$  and for any point  $H$  in the hyperplane, the line joining  $H$  and the median,  $M$ , is perpendicular on the line uniquely determined by  $\mathbb{C}_r$  and  $\mathbb{C}_{j(i)}$ .

Denote the coordinates of an arbitrary point in  $\mathcal{H}$ :  $h = (h_1, \dots, h_k)$ , the coordinates of  $\mathbb{C}_r$ , the arbitrary cluster center,  $p = (p_1, \dots, p_k)$ , and the coordinates of  $\mathbb{C}_{j(i)}$ :  $s = (s_1, \dots, s_k)$ .

We write that the joining segment  $HM$  is perpendicular on the line determined by  $\mathbb{C}_{j(i)}$  and  $M$ :

$$\sum_{i=1}^k \left( h_i - \frac{p_i + s_i}{2} \right) \left( s_i - \frac{p_i + s_i}{2} \right) = 0.$$

It follows that

$$w_i = p_i - s_i \text{ for } i = 1, \dots, k \text{ and } w_0 = \sum_{i=1}^k \frac{s_i^2 - p_i^2}{2}.$$

The confidence set is defined by:

$$\tilde{\mathbb{B}}_i = \{\tilde{\theta} : \tilde{\theta} = f(\theta), \theta \in \mathbb{B}_i\}.$$

We check  $\tilde{\mathbb{B}}_i \cap \mathcal{H} \neq \emptyset$  by computing the minimum distance from the confidence set  $\tilde{\mathbb{B}}_i$  to the bisection hyperplane.

$$\min_{\theta_i \in \mathbb{B}_i} d(f(\theta_i), \mathcal{H}) = \min_{\theta_i \in \mathbb{B}_i} \left[ \frac{\langle \theta_i, w \rangle}{\|\theta_i\| \|w\|} + \frac{w_0}{\|w\|} \right].$$

Because the minimum will be on the envelope, we solve

$$\min \frac{\langle \theta_i, w \rangle}{\|\theta_i\| \|w\|}, \text{ with } \|\theta_i - \hat{\theta}_i\| - r_i = 0. \quad (22)$$

The equivalent geometry problem to the optimization problem (22) is the following. We want to find the maximum angle to the origin between a fixed point  $W$  in the space and points on the envelope of the hypersphere  $\mathbb{B}_i$ . The problem reduces to finding the maximum angle,  $\widehat{TOW}$  where  $T$  falls on the envelope of a hypersphere. The problem (22) is easier to solve with the following constrains:

$$\min_{t_i} \frac{\langle t_i, w \rangle}{\|t_i\| \|w\|} \text{ with } \|t_i - \hat{\theta}_i\| = r_i, \langle t_i, (t_i - \hat{\theta}_i) \rangle = 0. \quad (23)$$

with the last equality due to the tangent in  $T$  from the origin. We rewrite again the minimization problem as:

$$\min_{t_i} \langle t_i, w \rangle \text{ with } \|t_i\|^2 = \|\hat{\theta}_i\|^2 - r_i^2, \langle t_i, \hat{\theta}_i \rangle = \|\hat{\theta}_i\|^2 - r_i^2. \quad (24)$$

We solve this optimization problem using Lagrange's theorem:

$$\begin{aligned} \nabla f(t) &= \mu \nabla g(t) + \lambda \nabla h(t) \\ f(t) &= \langle t, w \rangle = \sum_{j=1}^J t_j w_j \\ g(t) &= \|t\|^2 - (\|\hat{\theta}_i\|^2 - r_i^2) = \sum_{j=1}^J t_j^2 - C \\ h(t) &= \langle t, \hat{\theta}_i \rangle - (\|\hat{\theta}_i\|^2 - r_i^2) = \sum_{j=1}^J t_j \hat{\theta}_{ij} - C \end{aligned} \quad (25)$$

One solution to the problem gives the coordinates of  $c_i^{\min}$  with the minimum distance  $d(f(c_i^{\min}), \mathcal{H})$ .

The algorithm is different for the case when the origin is in the ball  $\mathbb{B}_i$ . For this case,  $C_i = \|\widehat{\theta}_i\|^2 - r_i^2 \leq 0$  and the coordinates of the maximum angle satisfies:

$$\begin{aligned} \langle t, w \rangle &= -\|t\|\|w\| \iff t = (-a)w \text{ with } a > 0 \\ \|t - \widehat{\theta}_i\| &= r_i^2. \end{aligned} \tag{26}$$

## References

- [1] Benjamini, Y., Hochberg, Y. (1995). "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing", *Journal of Royal Statistical Society, B*, 57, 1.
- [2] Bar-Joseph, Z., Gerber, G., Gifford, D.K., Jaakkola, T.S. (2002). "A new approach to analyzing gene expression time series data", *Proceedings of the 6th Annual International Conference on RECOMB*, pp 39-48.
- [3] Ben-Dorr, A, Shamir, R. and Yakhimi, Z. (1999). "Clustering gene expression patterns", *J. of Computational Biology*.
- [4] Beran, R. (2000), "REACT Scatterplot Smoothers: Superefficiency through basis economy", *Journal of the American Statistical Association*, 95, #449, pp 155-171.
- [5] Beran, R., Dúmbgen, L. (1998), "Modulation of estimators and confidence sets", *Annals of Statistics*, 26, 5, pp 1826-1856.
- [6] Duda, R.O., Hart, P.E. (1973). "Pattern classification and scene analysis", John Wiley & Sons, NY.
- [7] Dudoit, S., Yang, Y.H., Callow, M.J., Speed, T.P. (Aug 2000). "Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments", technical report.
- [8] Efron, B., Storey, J. D., Tibshirani, R.(July 2001). "Microarrays, Empirical Bayes Methods, and False Discovery Rates", *Journal of the American Statistical Association*, 96.
- [9] Eisen, M., Spellman, P., Brown, P., Botstein, D. (1998). "Cluster analysis and display of genome-wide expression patterns", *Proc. Nat. Acad. Sci* 95, 14863-14868.

- [10] Fowlkes, E. B., Mallows, C. L. (1983), "A method for comparing two hierarchical clusterings", *Journal of the American Statistical Association*, 78 (383), pp. 553-569.
- [11] Fridlyand, J., Dudoit, S. (Sept 2001), "Applications of resampling methods to estimate the number of clusters and to improve the accuracy of a clustering method", technical report # 600.
- [12] Handley, D., Serban, N., Peters, D., O'Doherty, R., Field, M., Wasserman, L., Spirtes, P., Scheines, R., Glymour, C. (2004), "Evidence of systematic expressed sequence tag IMAGE clone cross-hybridization on cDNA microarrays", *Genomics*, 83(6):1169-75.
- [13] Hastie, T., Tibshirani, R., Eisen, M., Alizadeh, A., Levy, R., Staudt, L., Chan, W., Botstein, D., Brown, P. (2000), "'Gene shaving' as a method for identifying distinct sets of genes with similar expression patterns", *Genome Biology*, I(2):research0003.1-0003.21.
- [14] Li, Ker-Chau (1989), "Honest confidence regions for nonparametric regression", *Annals of Statistics*, 3, pp 1001-1008.
- [15] Meilă, Marina (2002), "Comparing clusterings", University of Washington, Statistics Technical Report 418.
- [16] Newton, M.A., Kendzioriski, C.M., Richmond, C.S., Blattner, F.R., Tsui, K.W. (2001), "On differential Variability of Expression Ratios: Improving Statistical Inference about Gene Expression Changes from Microarray Data", *Journal of Computational Biology*, 8, # 1, pp. 37-52.
- [17] Pollard, D. (1981), "A central limit theorem for k-means clustering", *Annals of Probability*, 1, pp. 919-926.
- [18] Pollard, D. (1982), "Strong consistency of k-means clustering", *Annals of Statistics*, 1, pp. 135-140.
- [19] Rand, W. M. (1971), "Objective criteria for the evaluation of clustering methods". *Journal of the American Statistical Association*, 66, pp. 846-850.
- [20] Storey, J. D. and Tibshirani, R. (2002), "Estimating FDR under Dependence with Applications to DNA microarrays", technical report.

- [21] Tibshirani, R., Hastie, T., Narasimhan, B., Eisen, M., Sherlock, G., Brown, P., Botstein, D. (2001). "Exploratory screening of genes and clusters from microarray experiments", technical report.
- [22] Tibshirani, R., Walther, G., Hastie, T. (Dec 2000), "Estimating the number of clusters in a dataset via the Gap statistic". Technical report, published in *JRSSB*, 2000.
- [23] Yang, Y. H., Dudoit, S., Luu, P., and Speed, T. P., "Normalization for cDNA Microarray Data", technical report.
- [24] Yeung, K.Y., Murua, A., Raftery, A., Ruzzo, W.L. (2001). "Model-Based Clustering and Data Transformations for Gene Expression Data", technical report.
- [25] van der Vaart, A.W. (1998). "Asymptotic statistics", Cambridge University Press.
- [26] Wakefield, J., Zhou, C., Self, S. (2002), *Modelling Gene Expression Data over Time: Curve Clustering with Informative Prior Distributions*, Bayesian Statistics 7, Proceedings of the Seventh Valencia International Meeting, 2003.

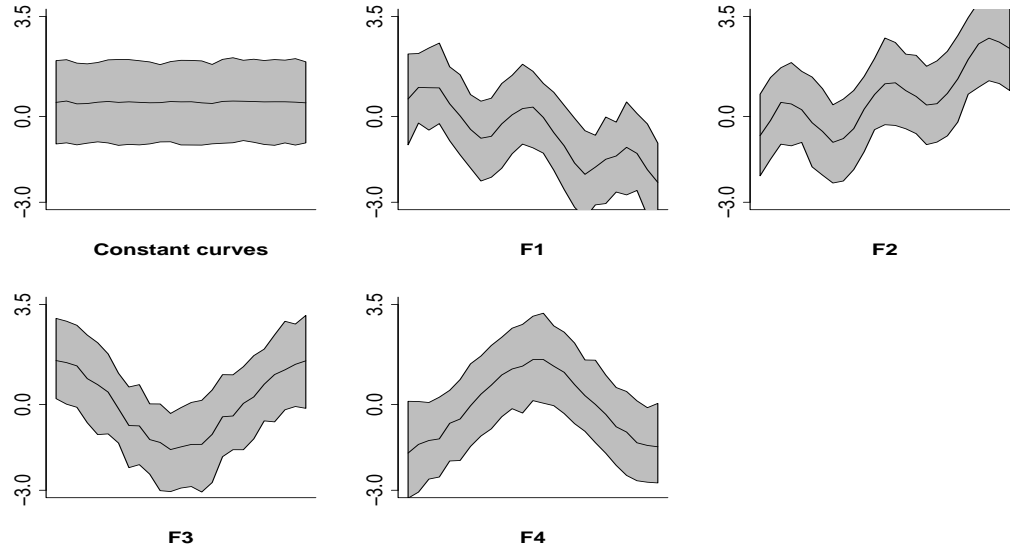


Figure 2: Mean, 5th and 95th percentile for the four true clusters in the synthetic data with  $\sigma \sim \text{Unif}(1.0, 1.2)$  and  $m = 25$ .

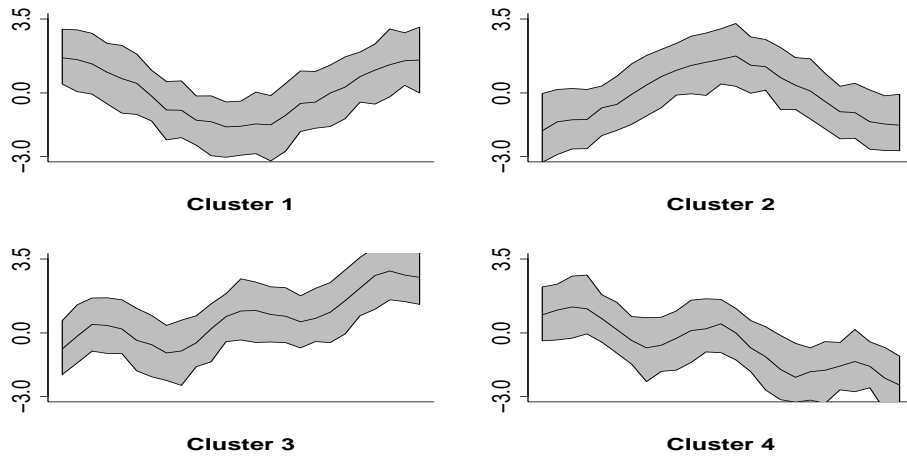


Figure 3:  $k$ -means clustering on simulated data ( $\sigma \sim \text{Unif}(1.0, 1.2)$ ) where the number of cluster is  $K = 4$ . For this case, we performed **smoothing and screening**.

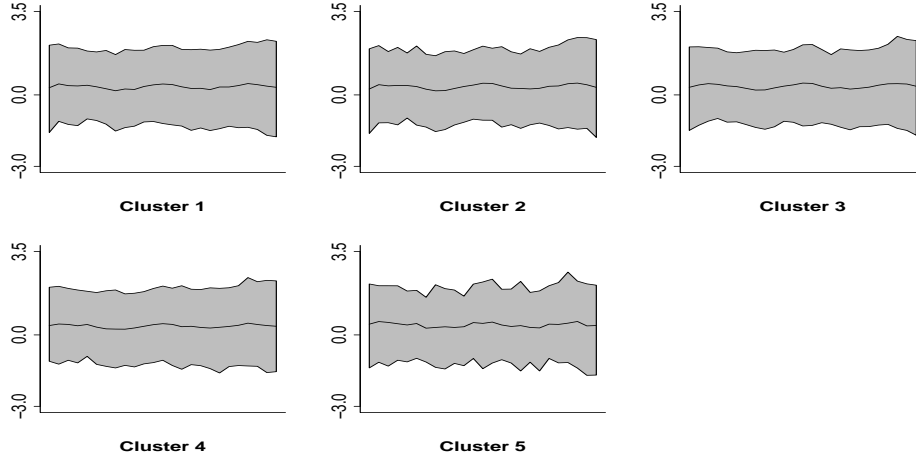


Figure 4: Hierarchical clustering on simulated data ( $\sigma \sim \text{Unif}(1.0, 1.2)$ ) where the number of cluster is  $K = 5$  (4 curve patterns and the constant curve pattern). For this case, we performed **neither screening nor smoothing**.

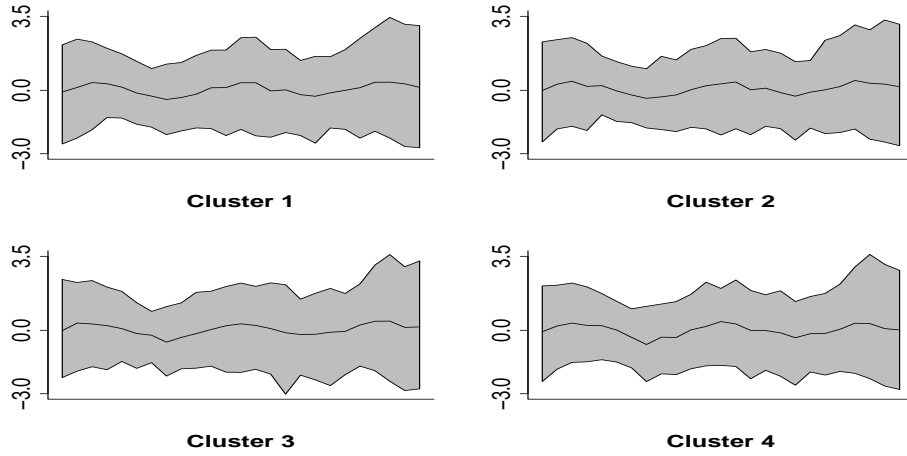


Figure 5: Hierarchical clustering on simulated data ( $\sigma \sim \text{Unif}(1.0, 1.2)$ ) where the number of cluster is  $K = 4$  (4 curve patterns and the constant curve pattern). For this case, we performed **screening but not smoothing**.

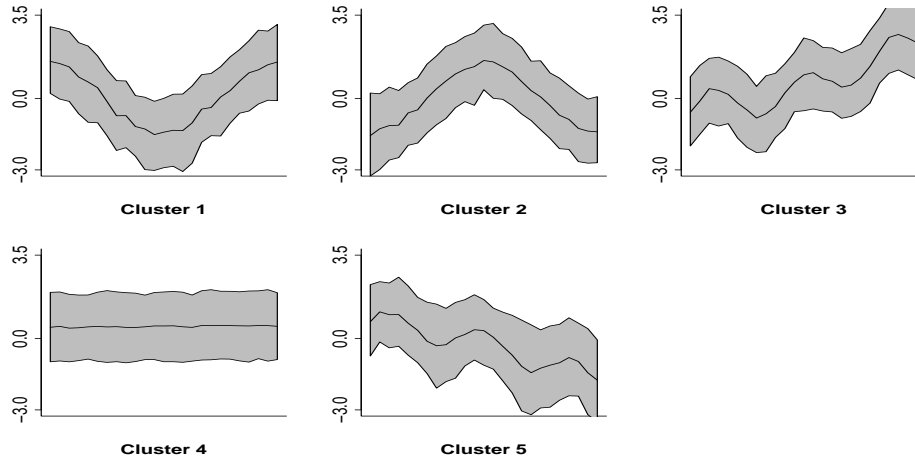


Figure 6:  $k$ -means clustering on simulated data ( $\sigma \sim \text{Unif}(1.0, 1.2)$ ) where the number of cluster is  $K = 5$ . For this case, we performed **smoothing but not screening**.

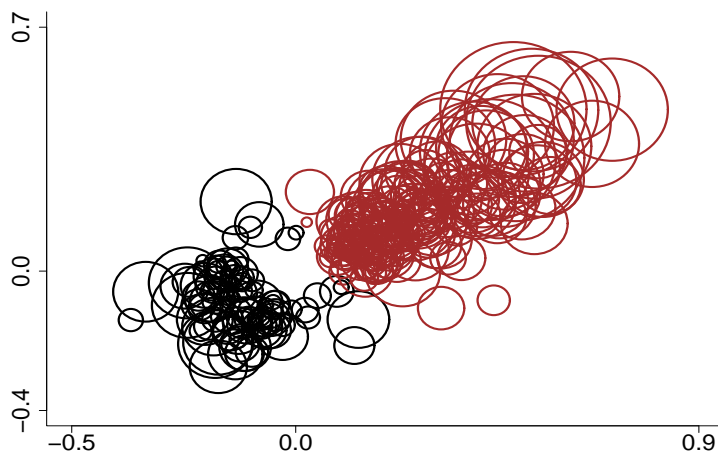


Figure 7: Two-dimensional confidence balls for significant genes. Each circle represents the 2D confidence ball of one gene. The clustering was performed with **smoothed** data using  $k$ -means algorithm and  $k = 2$ .

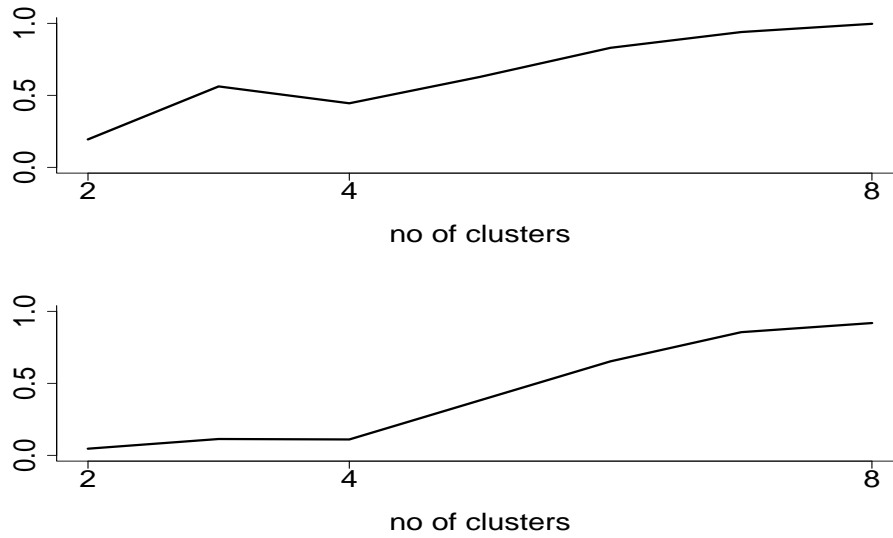


Figure 8: The clustering error rate  $\eta$  for  $m = 25$  (upper plot) and for  $m = 50$  (bottom plot).

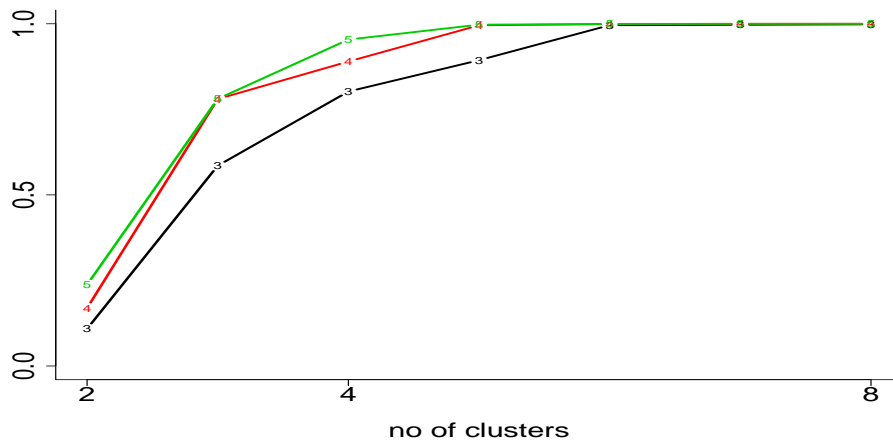


Figure 9: Upper bound for  $\eta$  - microarray data. Each curve represents the asymptotic upper bound for  $\eta$  for a given smoothing parameter  $J$  over the number of clusters. For example, curve 3 in the figure consists of the estimated upper bound for  $J = 3$ .



Figure 10: Cluster quality for gene expression data. In the upper plot, we provide the values of  $\Omega$  vs. the number of clusters and the difference in  $\Omega$  (bottom plot).

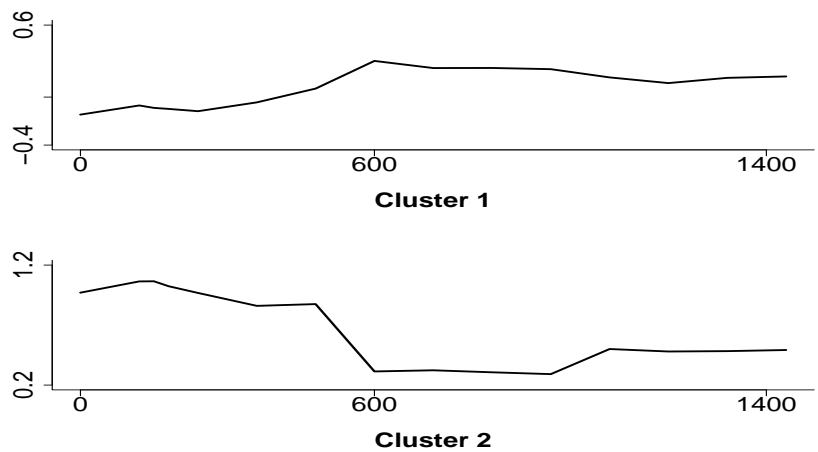


Figure 11: Average curves by cluster. In each plot, the average over all genes in one cluster is plotted over time.