Chapter 18

Clusterwise Regression using Dirichlet Mixtures

Changku Kang[1] and Subhashis Ghosal[2]

[1] *Economics Statistics Department, The Bank of Korea; 110, 3-Ga, Namdaemun-Ro, Jung Gu, Seoul, Korea. E-mail: koncap@gmail.com*

[2] *Department of Statistics, North Carolina State University; 2501 Founders Drive, Raleigh, North Carolina 27695, U.S.A. E-mail: sghosal@stat.ncsu.edu,*

The article describes a method of estimating nonparametric regression function through Bayesian clustering. The basic working assumption in the underlying method is that the population is a union of several hidden subpopulations in each of which a different linear regression is in force and the overall nonlinear regression function arises as a result of superposition of these linear regression functions. A Bayesian clustering technique based on Dirichlet mixture process is used to identify clusters which correspond to samples from these hidden subpopulations. The clusters are formed automatically within a Markov chain Monter-Carlo scheme arising from a Dirichlet mixture process prior for the density of the regressor variable. The number of components in the mixing distribution is thus treated as unknown allowing considerable flexibility in modeling. Within each cluster, we estimate model parameters by the standard least square method or some of its variations. Automatic model averaging takes care of the uncertainty in classifying a new observation to the obtained clusters. As opposed to most commonly used nonparametric regression estimates which break up the sample locally, our method splits the sample into a number of subgroups not depending on the dimension of the regressor variable. Thus our method avoids the curse of dimensionality problem. Through extensive simulations, we compare the performance of our proposed method with that of commonly used nonparametric regression techniques. We conclude that when the model assumption holds and the subpopulation are not highly overlapping, our method has smaller estimation error particularly if the dimension is relatively large.

18.1. Introduction

Consider the problem of estimating a regression function $m(x) = \mathrm{E}(Y|X = x)$ based on sampled data $(X_1, Y_1), \ldots, (X_n, Y_n)$, where X is d-dimensional. Typically

$m(x)$ is estimated under the homoscedastic signal plus noise model

$$Y_i = m(X_i) + \varepsilon_i, \quad \varepsilon_1, \ldots, \varepsilon_n \text{ i.i.d. with } E(\varepsilon) = 0, \quad \text{var}(\varepsilon) = \sigma^2, \tag{18.1}$$

assuming that $m(x)$ has a parametric form $\psi(x;\theta)$ or $m(x)$ is estimated nonparametrically based only on smoothness assumptions. Numerous methods such as those based on the nearest neighbors, kernel, local linearity, orthogonal series or spline smoothing for estimating the regression function exist in the literature; see any standard text such as Ref.6 These methods are extremely powerful in lower dimension. In higher dimension, because observations are relatively sparsely distributed over the space, one typically needs to assume additional structure in the regression function. Generalized additive models (GAM) [Hastie and Tibshirani,1990], classification and regression tress (CART) [Breiman et al.,1984] and multivariate adaptive regression splines (MARS) [Friedman,1991] are among the popular estimation methods in higher dimension.

Bayesian methods for the estimation of a regression function have been developed by putting priors directly on the regression function. A popular method of assigning a prior on functions is by means of a Gaussian process. Ref.21 considered an integrated Wiener process prior for $m(x)$. The resulting Bayes estimate, in the sense of a noninformative limit, is a smoothing spline. Priors based on series expansions with normally distributed coefficients, which are also Gaussian priors, have been investigated well for various choices of basis functions. An indirect method of inducing a prior on the regression function from that on the joint density $f(x,y)$ of (x,y) was considered by Ref., who used a Dirichlet mixture of normal prior for the joint density. The idea is to use the relation $m(x) = \int yf(x,y)dy / \int f(x,y)dy$. This method requires estimating density functions in a higher dimensional space and the growth of the regression function is restricted by that of a linear function, since multivariate normal regressions are linear. In addition, the method suffers heavily from curse of dimensionality in higher dimension.

Sometimes, however, X is a sampled from a population which can be best described as a mixture of some hidden subpopulations. In this case, it is plausible that the regression function has different parameter values when the samples come from different groups. This situation will typically arise if there is one unobservable categorical label variable which influences both X and Y, but conditional on that missing label variable, the relationship between Y and X is linear up to an additive error. However, as we do not observe the group labels, the overall regression function of Y on X is a weighted average of the individual regression functions, where the weights are given by the probabilities of the observation being classified into different groups given its value. In such a situation, an easy

calculation (see the next section) shows that the model 18.1.1 fails. The mixture distribution arises naturally in biology, economics and other sciences. For example, in a marketing survey, suppose that consumers rate the quality of a product. Different consumers may give different weights to various factors depending on their background and mentality. In other words, the population actually consists of different subpopulations where different regression functions are in effect, but subpopulation membership is abstract and is not observable. Had we observed the labels, regression analysis would have been straightforward. In the absence of the labels, we use the auxiliary measurements to impute the lost labels and use a simple regression analysis within each hypothetical group. The number of groups and group membership labels are nuisance parameters in the analysis, which should be integrated out with respect to their posterior distribution.

In this paper, we propose a method to estimate an unknown regression function by viewing it as a mixture of an unknown number of some simple regression functions associated with each subpopulation. These corresponding subgroups in the sample are estimated by identifying clusters in the data cloud. The clusters are automatically produced by a Dirichlet mixture model, a popular Bayesian nonparametric method. Standard regression methods are used in fitting a regression function within each cluster. Each iteration of the Markov chain Monte-Carlo (MCMC) sampling based on the Dirichlet mixture process produces a partitioning of the data cloud and thus an estimate of the regression function. The final regression estimate is obtained by averaging out the estimate obtained in each iteration thus integrating out the nuisance label parameters with respect to their posterior distribution obtained from the Dirichlet mixture process prior. Thus our method may be viewed as an ensemble method like bagging [Breiman,1996] and random forest [Breiman,2001], but here the ensemble is produced by MCMC iterations rather than by bootstrap resamples.

The similarity and difference of our method with other nonparametric methods are interesting to note. Regression estimates based on free-knot splines, kernels or nearest neighbors partition the space and fit some standard regression function in each part and combine these individual estimates. We also combine individual estimates based on different components of the sample. However, unlike other standard methods, we partition the observed samples rather than the space where they lie. Under the assumed condition that the samples come from a fixed (although unknown) number of groups, our method requires partitioning the sample in a number of groups that essentially remains fixed independent of the dimension of the space. In contrast, many standard nonparametric methods in higher dimension require splitting the space in a relatively high number of subregions fueling the demand for a larger sample size, thus leading to the curse of dimensionality

problem. The simulations conducted in Section 18.3 confirm that our method performs better than standard nonparametric methods, particularly in higher dimension, provided that the mixture model representation is appropriate. In addition, we could get conservative pointwise credibility bands very easily which are usually difficult with other methods. Another operational advantage of our method is that there is no need to choose any smoothing parameter.

There have been attempts to utilize clusters to partition data and estimate regression. The term clusterwise regression was first used by Ref.18. Ref 4. proposed a conditional mixture maximum likelihood methodology to identify clusters and partition the sample before applying linear regression in each piece. The method estimates the parameters of the mixture distribution by maximum likelihood using the EM algorithm. The number of components k in the mixture is chosen by the Akaike information criterion. Their method is closest in spirit to ours. However, they do not take into account the uncertainty in the number of groups and each group membership. In the smoothly mixing regression method approach of Ref.10, one estimates the probability of an observation with value x belonging to different subpopulations by a multinomial probit model. The main reason for using this model is that it produces simple and tractable posterior distributions within the Gibbs sampling algorithm. However, there is no room for a data driven choice of k and the method suffers from the curse of dimensionality problem in higher dimension.

We shall compare our method with standard regression estimates. More specifically, by means of extensive simulation studies, we shall compare the performance of our method with the kernel and spline methods in one dimension. In higher dimension, we shall compare with the estimates obtained from GAM and MARS.

The paper is organized as follows. Section 18.2 provides detailed description of our method, including the MCMC algorithms and the construction of confidence bands. Simulation results are presented and discussed in Section 18.3.

18.2. Method Description

Suppose that we have k regression models

$$Y_i = m_j(X_i) + \varepsilon_i, \quad \varepsilon_i \text{ i.i.d. with } \mathrm{E}(\varepsilon_i) = 0,\ \mathrm{var}(\varepsilon_i) = \tau_j^2, \quad i = 1, \ldots, n, \quad (18.2)$$

corresponding to the jth subgroup in the populations, $j = 1, \ldots, k$. Let J be an unobserved random variable indicating the subgroup label. The prior distribution of J is given by $\pi_j = P(J = j)$ and thus the posterior distribution given $X = x$ is given by $\pi_j(x) = P(J = j|X = x)$. Then, after eliminating the unobserved J, the

regression of Y on X is given by

$$m(x) = \sum_{j=1}^{k} \mathrm{P}(J = j|X = x)\mathrm{E}(Y|X = x, J = j) = \sum_{j=1}^{k} \pi_j(x) m_j(x). \tag{18.3}$$

Also observe that the conditional variance is given by

$$\begin{aligned} \mathrm{var}(Y|X = x) &= \mathrm{E}_J(\mathrm{var}(Y|X = x, J)) + \mathrm{var}_J(\mathrm{E}(Y|X = x, J)) \\ &= \sum_{j=1}^{k} \tau_j^2 \pi_j(x) + \sum_{j=1}^{k} m_j^2(x)\pi_j(x) - \Big(\sum_{j=1}^{k} m_j(x)\pi_j(x) \Big)^2, \end{aligned}$$

which is not constant in x even if all τ_j's are equal. Hence the commonly assumed nonparametric regression model 18.2.1 fails to take care of the simple situation of missing labels.

The regression function $m(x)$ given by 18.2.2 may be estimated by its posterior expectation given the observed data. This can be evaluated in two stages — in the first stage we can obtain the posterior expectation given the hidden subpopulation labels and the observed data, while in the second stage we average the resulting quantity with respect to the conditional distribution of the labels given the data. The second stage averaging may be taken conditional on the X values only. For a discussion on how the Y observations may also be used to detect the hidden labels, see Ref. . Thus we need to model the joint distribution of the X and the missing subpopulation labels, which is done below. The first stage analysis reduces to k independent parametric regression problems. Therefore the posterior expectation of $m_j(x)$, $j = 1, \ldots, k$, may essentially be replaced by the corresponding least square estimates. Alternatively, one may view m_js as hyperparameters and the resulting substitution corresponds to an empirical Bayes approach. While a fully Bayesian analysis at this stage is quite possible, the minor difference between the posterior expectations and the least square estimates, particularly when the sample size is not very small, makes the additional computing requirement unappealing. Henceforth we shall consider this hybrid approach of least square estimation in the first stage coupled with a posterior analysis of missing subpopulation labels. The joint distribution of X and the missing label J will be specified somewhat indirectly, through another set of latent variables. The technique is based on one of most popular methods in Bayesian nonparametrics, namely that of Dirichlet mixture (DM). Let $X_1, \ldots, X_n$ be i.i.d. from a density

$$f(x; G, \Sigma) = \int \phi(x; \theta, \Sigma) dG(\theta), \tag{18.4}$$

where $\phi(\cdot; \theta, \Sigma)$ is the density function of the d-dimensional normal density with mean θ and dispersion matrix Σ. Let G follow $\mathrm{DP}(M, G_0)$, the Dirichlet process

with precision parameter M and center measure G_0 in the sense of Ref. . The model can be equivalently written in terms of latent variables $\theta_1, \ldots, \theta_n$ as

$$X_i|\theta_i \overset{\text{ind}}{\sim} N(\cdot;\theta_i,\Sigma), \quad \theta_i|G \overset{\text{iid}}{\sim} G, \quad G \sim \text{DP}(M, G_0). \tag{18.5}$$

Because Dirichlet samples are discrete distributions, the Dirichlet mixture prior has the ability to automatically produce clusters. More precisely, we may identify X_i and X_j as the member of a common cluster if $\theta_i = \theta_j$. Note that the last event occurs occasionally because the posterior distribution of $(\theta_1, \ldots \theta_n)$ given $(X_1, \ldots, X_n)$ is given by the Polya urn scheme described by

$$\begin{gathered}
\theta_i|(\theta_{-i}, \Sigma, X_1, \ldots, X_n) \propto q_{i0} dG_i(\theta_i) + \textstyle\sum_{j\neq i} q_{ij} \delta_{\theta_j}(\theta_i), \\
dG_i(\theta) \propto \phi(X_i;\theta,\Sigma) dG_0(\theta), \\
q_{i0} \propto M \textstyle\int \phi(X_i;\theta,\Sigma) dG_0(\theta), \quad q_{ij} \propto \phi(X_i;\theta_j,\Sigma), \quad q_{i0} + \textstyle\sum_{j\neq i} q_{ij} = 1;
\end{gathered}$$

here and throughout, the subscript "$-i$" stands for all but i. This way a probability distribution on partitions of $\{1, 2, \ldots, n\}$ according to subpopulation membership is obtained, which in effect describes a joint distribution of X observations and the corresponding labels. In particular, the distribution of the number of hidden subpopulations is also described by the process.

Taking advantage of the many ties in θ_i's, a more efficient algorithm can be constructed based on a reparameterization $(k, \varphi_1, \ldots, \varphi_k, s_1, \ldots, s_n)$, where k is the number of distinct θ_i values, $\varphi_1, \ldots, \varphi_k$ are the set of distinct θ_i's and $s_i = j$ if and only if $\theta_i = \varphi_j$. In fact, to identify the clusters, we need not even know the values of $\varphi_1, \ldots, \varphi_k$; it suffices to know the configuration vector $\mathbf{s} = (s_1, \ldots, s_n)$. This will allow substantial reduction of computational complexity at the MCMC step. The clusters are given by $I_j = \{i : s_i = j\}$, $j = 1, \ldots, k$, and $H = \{I_1, \ldots, I_k\}$ is a partition of $I = \{1, \ldots, n\}$. Integrating out $\varphi_1, \ldots, \varphi_k$ in the above step, we obtain the following algorithm to generate the clusters through a Gibbs sampler when M, G_0 and Σ are given:

$$\text{P}(s_i = j|\mathbf{s}_{-i}, X_1, \ldots, X_n) \propto \begin{cases} n_{-i,j} \int \phi(X_i;\theta,\Sigma) dH_{-i,j}(\theta), & j = 1, \ldots, k_{-i}, \\ M \int \phi(X_i;\theta,\Sigma) dG_0(\theta), & j = k_{-i} + 1, \end{cases} \tag{18.6}$$

where $H_{-i,j}$ is the posterior distribution based on the prior G_0 and all observations X_l for which $l \in I_j$ and $l \neq i$, that is,

$$dH_{-i,j}(\theta) \propto \prod_{l\in I_j, l\neq i} \phi(X_l;\theta,\Sigma) dG_0(\theta),$$

$n_{-i,j} = \#I_j \setminus \{i\}$, $j = 1,\ldots,k_{-i}$, k_{-i} is the number of distinct elements in $\{\theta_j : j \neq i\}$ and $\mathbf{s}_{-i} = (s_1,\ldots,s_{i-1},s_{i+1},\ldots,s_n)$. Letting G_0 have density $\phi(\cdot;\xi,\Phi)$, it follows that

$$\mathrm{P}(s_i = j|s_{-i},X_1,\ldots,X_n) \propto M\phi(X_i;\xi,\Sigma+\Phi), \quad j = k_{-i}+1.$$

For $j = 1,\ldots,k_{-i}$, the corresponding expression can be simplified using

$$\begin{aligned}
&\int \phi(X_i;\theta,\Sigma)dH_{-i,j}(\theta) \\
&= \frac{\int \phi(X_i;\theta,\Sigma)\prod_{l\in I_j,l\neq i}\phi(X_l;\theta,\Sigma)\phi(\theta;\xi,\Phi)d\theta}{\int \prod_{l\in I_j,l\neq i}\phi(X_l;\theta,\Sigma)\phi(\theta;\xi,\Phi)d\theta} \\
&= \frac{n_{-i,j}}{n_{-i,j}+1}\phi(X_i;\bar{X}_{+i,j},\frac{n_{-i,j}}{n_{-i,j}+1}\Sigma) \times \frac{\phi(\xi;\bar{X}_{+i,j},\frac{1}{n_{-i,j}+1}\Sigma+\Phi)}{\phi(\xi;\bar{X}_{-i,j},\frac{1}{n_{-i,j}}\Sigma+\Phi)},
\end{aligned} \tag{18.7}$$

where

$$\bar{X}_{-i,j} = \frac{1}{n_{-i,j}}\sum_{l\in I_j,l\neq i} X_l \quad \text{and} \quad \bar{X}_{+i,j} = \frac{1}{n_{-i,j}+1}\left(\sum_{l\in I_j,l\neq i} X_l + X_i\right). \tag{18.8}$$

The above expressions are controlled by the hyper-parameters M, ξ, Φ and Σ. In a Bayesian setting, it is natural to put priors on them too and update during the MCMC stage. However, as we have integrated out $\varphi_1,\ldots,\varphi_k$, the complicated form of the marginal likelihood of ξ, Φ and Σ based on only $(X_1,\ldots,X_n)$ and $(s_1,\ldots,s_n)$ rules out existence of natural conjugate priors. One may update the hyperparameters using the Metropolis-Hastings algorithm, but it slows down the process. To avoid that, we take an empirical Bayes approach and estimate these parameters within an MCMC step by taking advantage of the knowledge of the partition. As ξ is the marginal expectation of X, we estimate it by $\hat{\xi} = \bar{X}$. Note that Σ is the dispersion matrix within group, and hence can be estimated by

$$\hat{\Sigma} = n^{-1}\sum_{j=1}^{k}\sum_{l\in I_j}(X_l-\bar{X}_j)(X_l-\bar{X}_j)^T, \qquad \bar{X}_j = n_j^{-1}\sum_{l\in I_j} X_l. \tag{18.9}$$

The matrix Φ stands for the between variation and hence can be estimated by

$$\hat{\Phi} = n^{-1}\sum_{j=1}^{k} n_j(\bar{X}_j-\bar{X})(\bar{X}_j-\bar{X})^T. \tag{18.10}$$

Clearly both $\hat{\Sigma}$ and $\hat{\Phi}$ are positive definite provided that $n_j > 1$ for all j. Note that, as we are dealing with a relatively few parameters, for large sample size, the difference between the empirical and full Bayesian estimates are likely to be small. The precision parameter M can be updated by Gibbs sampling using a clever data

augmentation technique [Escobar and West,1995]. However, to reduce computational time, we shall work with a fixed moderate value of M such as 1 or 2.

It may be noted that each step of MCMC changes the membership of at most one observation. Since sample means and dispersions can be calculated recursively when an observation is added to or deleted from a group, it is clear that one can avoid calculation from the scratch every time in 18.2.6.

Also it follows that we could form location-scale mixtures in 18.2.3 by allowing Σ to vary in each group as Σ_j, $j = 1,\ldots,k$. The same methodology will work if Σ_j's are estimated separately, for instance, by $\hat{\Sigma}_j = n_j^{-1}\sum_{l\in I_j}(X_l - \bar{X}_j)(X_l - \bar{X}_j)^T$.

Once the clusters have been identified, we can use a standard regression method, such as the method of least squares for linear regression, within each cluster giving estimates $\hat{m}_j(x)$ of regression functions $m_j(x)$, $j = 1,\ldots,k$. Polynomial regression may be used if we suspect the lack of linearity in the model. Variants of the least square estimate such as the ridge regression estimate [Hoerl and Kennard,1970] or the LASSO [Tibshirani,1996] may be used in place of ordinary least squares to lower the variation in the estimate if there is a lot of multicolinearity in the data. The latter is particularly very useful in higher dimension where typically a variable selection is desirable before estimation. Further, as the clusters produced by the Dirichlet mixture will invariably differ from the true clusters, the use of a robust alternative to the least square estimate is desirable to prevent misclassified observations altering the estimates significantly. These robust estimates require some tuning and are generally harder to compute compared to the least squares estimate. For the sake of simplicity, we do not pursue these modifications in the present paper.

To complete the estimation of $m(x)$, it remains to estimate $\pi_j(x)$, $j = 1,\ldots,k$. In order to do that, we consider a Bayesian model based on the given observations and the clusters. The prior probability of a new observation coming from the jth subgroup is given by the empirical fraction of observations falling in the group, that is n_j/n, $j = 1,\ldots,k$. Under the empirical model, the jth subpopulation is assumed to be normal with center at $\bar{X}_j$ and dispersion matrix $\hat{\Sigma}$. If the value of the new observation is x, then the likelihood of the jth subgroup is $\phi(x;\bar{X}_j,\hat{\Sigma})$. Applying the Bayes theorem, the posterior probability of x coming from the jth subpopulation, based on the empirical model, is proportional to $n_j\phi(x;\bar{X}_j,\hat{\Sigma})$. To estimate the regression function at x given the clusters, one may do a "model averaging" with respect to the empirical posterior probabilities to yield the estimate

$$\hat{m}(x) = \frac{\sum_{j=1}^k n_j\phi(x;\bar{X}_j,\hat{\Sigma})\hat{m}_j(x)}{\sum_{j=1}^k n_j\phi(x;\bar{X}_j,\hat{\Sigma})}. \tag{18.11}$$

Alternatively, one may do a "model selection" to select the most plausible sub-

model and then use the estimate within that submodel to yield the estimate

$$\hat{m}(x) = \hat{m}_{\hat{j}}(x), \quad \hat{j} = \arg\max\{j : n_j \phi(x; \bar{X}_j, \hat{\Sigma})\}. \tag{18.12}$$

Both of the above estiamtes 18.2.11 and 18.2.12 are based on a given value of k and the given break-up of the clusters. Of course, all of these are unknown. In order to remove the dependence on a particular cluster formation, we form an ensemble of estimates given by 18.2.11 or 18.2.12 — a fresh estimate in each MCMC iteration of the DM process. The final estimate may thus be formed by a simple averaging of these $\hat{m}(x)$. We call these ensemble based estimates corresponding to 18.2.11 and 18.2.12 as DM-AVE (Dirichlet mixture — average) and DM-ML (Dirichlet mixture — most likely) respectively.
Piecing all the previous discussions together, the complete algorithm to calculate our estimate may be described as follows:

- Choose a sufficiently fine grid of x-values.
- Start with an initial configuration $(k, s_1, \ldots, s_n)$.
- Let $\mathbf{s}_{-i}$, k_{-i} and $n_{-i,j}$ be as defined above. Given $\mathbf{s}_{-i}$ and $X_1, \ldots, X_n$, sample s_i from $\{1, \ldots, k_{-i}, k_{-i}+1\}$ with probabilities

$$\begin{cases} c\frac{n_{-i,j}}{n_{-i,j}+1}\phi(X_i; \bar{X}_{+i,j}, \frac{n_{-i,j}}{n_{-i,j}+1}\hat{\Sigma}) \times \frac{\phi(\bar{X}; \bar{X}_{+i,j}, \frac{1}{n_{-i,j}+1}\hat{\Sigma}+\hat{\Phi})}{\phi(\bar{X}; \bar{X}_{-i,j}, \frac{1}{n_{-i,j}}\hat{\Sigma}+\hat{\Phi})}, & j = 1, \ldots, k_{-i}, \\ cM\phi(X_i; \bar{X}, \hat{\Sigma}+\hat{\Phi}), & j = k_{-i}+1, \end{cases}$$

  where $\bar{X}_j$, $\bar{X}_{+i,j}$, $\bar{X}_{-i,j}$, $\hat{\Sigma}$ and $\hat{\Phi}$ are defined in equations 18.2.8, 18.2.9 and 18.2.10, and c is the reciprocal of the following expression:

$$M\phi(X_i; \bar{X}, \hat{\Sigma}+\hat{\Phi}) + \sum_{j \neq i} \frac{n_{-i,j}^2}{n_{-i,j}+1} \phi(X_i; \bar{X}_{+i,j}, \frac{n_{-i,j}}{n_{-i,j}+1}\hat{\Sigma}) \times \frac{\phi(\bar{X}; \bar{X}_{+i,j}, \frac{1}{n_{-i,j}+1}\hat{\Sigma}+\hat{\Phi})}{\phi(\bar{X}; \bar{X}_{-i,j}, \frac{1}{n_{-i,j}}\hat{\Sigma}+\hat{\Phi})}.$$

- Repeat the above procedure for $i = 1, \ldots, n$ to complete an MCMC cycle.
- Compute DM-AVE and DM-ML as defined by 18.2.11 and 18.2.12 for each cycle.
- Repeat the cycle sufficiently large number of times to let the chain mix well.
- Compute the averages of the estimates DM-AVE or DM-ML for each cycle after a burn-in for each point on a chosen grid.
- Obtain the estimated function by joining the average values over the grid points.

In density estimation applications, one usually works with the initial configuration that $k = n$, that is, all clusters are singleton. Here one cannot use that because $\hat{\Sigma}$ will then be the zero matrix, and thus the formula will break down. As an alternative, one may split the domain into small cubes and form the initial clusters by looking at the sample values falling into these regions, where singleton clusters will have to be merged with some neighboring one.

Note that we substitute ξ, Σ, Φ, and $m_1(\cdot), \ldots, m_k(\cdot)$, as well as the group mean values $\theta_1, \ldots, \theta_k$ within an MCMC iteration by the corresponding empirical estimates in the analysis, thus adopting an empirical Bayes approach. Implementation of a fully hierarchical Bayesian approach is possible provided that we do full updating of the Dirichlet mixture process. However, the fully Bayesian approach takes much longer time to run and does not seem to lead to more accurate estimates.

It may be observed that we used the conjugate normal center measure G_0 for the Dirichlet process. In principle, it is possible to use any center measure, but the equation 18.2.7 and the formula preceding that will not have any closed-form expression. One can carry out the fully Bayesian method of updating each $\varphi_1, \ldots, \varphi_k$ by using specially devised algorithms for the non-conjugate case like the "no gaps" algorithm [MacEachern and Müller]. However, the simplification resulting from integrating out the latent variables $\varphi_1, \ldots, \varphi_k$ and working only with the configuration vector $(s_1, \ldots, s_n)$ will not be possible, unless one calculates integrals numerically in equation 18.2.6. Numerical integration at each iteration of the Gibbs sampling scheme will be a daunting task and will result in extremely slow and inefficient computing.

Our method can be expanded further to construct a pointwise band for the regression function. Let x be a fixed point in the space of the regressor. A confidence interval for $m(x) = \mathrm{E}(Y|X = x)$ is easily obtained from the method of least squares given the subgrouping of the samples and the group to which the new observation $X = x$ belongs. However, since both the group to which x belongs and the break-up of clusters are unknown, we can integrate them out according to their posterior distributions, leading to a "posterior expected pointwise confidence band" as described below. Treat $X = x$ as a new observation and suppose that J stands for the label of the cluster where x belongs. Given the clusters, let $[L_j, U_j]$ be $100(1-\alpha_0)\%$ confidence limits for a $m_j(x)$, $j = 1, \ldots, k$. This confidence interval may be thought of as a posterior credible interval for $m_j(x)$. Especially, in large sample sizes the two notions will tend to agree. Let L be the $\gamma/2$-quantile of $L_1, ..., L_k$ and U be the $1-\gamma/2$-quantile of $U_1, \ldots, U_k$, respectively, where $\gamma = \alpha - \alpha_0$ and the associated probabilities are $\pi_1(x), \ldots, \pi_k(x)$. Let

$m^*(x) = m_j(x)$ if $J = j$, $j = 1,\ldots,k$. Then

$$\mathrm{P}(m^*(x) \le L) = \sum_{j:L_j \le L} \mathrm{P}(m_j(x) \le L)\pi_j(x) + \sum_{j:L_j > L} P(m_j(x) \le L)\pi_j(x) \le \frac{\gamma}{2} + \frac{\alpha_0}{2}.$$

Similarly $\mathrm{P}(m^*(x) \ge U) \le (\gamma + \alpha_0)/2$, and so $\mathrm{P}(L \le m^*(x) \le U) \ge 1 - \alpha$, $\alpha = \gamma + \alpha_0$. Thus by the convexity of $[L,U]$, $m(x)$, being an average of $m^*(x)$, has the same credible bound given the clusters. Finally, the bounds L and U may be averaged out over MCMC iterations leading the final posterior expected pointwise confidence band. The obtained band may not have frequentist validity though, and it more closely resembles a posterior credible band. Asymptotically, Dirichlet mixtures estimate densities consistently [cf. Ghosal et al,1999], which implies that the mixing distribution is estimated consistently in the weak topology. In view of a result of Ref. 5, consistency essentially implies that the number of clusters in the pattern obtained by the posterior is asymptotically at least as much as the true number of groups. In general, the obtained number can be considerably bigger, which is also confirmed by the simulation study. However, the superfluous breaking up of groups causes only a minor loss of precision because of smaller usable sample size. On the other hand, an inappropriate merging of two groups introduces serious bias in the estimates.

18.3. Simulation Study

In this section we do several simulations to evaluate the performance of our method. First, data are generated from a mixture of normal distributions in the univariate and the multivariate case, and in different groups different regression functions are given with several parameters. We proceed to estimate the fitted regression function using our proposed method and also some other methods such as those based on kernel and spline smoothing in the univariate case and MARS and GAM in the multivariate case.

We compare our method with other methods by the empirical L_1-error and L_2-error between estimates and the true regression function, defined respectively by $\mathrm{EL}_1(\hat{f}) = n^{-1}\sum_{i=1}^n |\hat{f}(x_i) - f(x_i)|$ and $\mathrm{EL}_2(\hat{f}) = \{n^{-1}\sum_{i=1}^n (\hat{f}(x_i) - f(x_i))^2\}^{1/2}$.

Since the success of our method is intimately linked with the ability to discover the correct patterns in the data as well as the number of clusters, in the simulation study, we shall monitor the distribution of the number of groups found by the posterior and the similarity of the obtained groupings with the actual ones.

In order to monitor how close is the partition induced by the Dirichlet mixture process to the true partition, we shall use Rand's measure [Rand,1971] of similarity of clusterings. A more refined measure is provided by Ref. 14. Given n points,

$\theta_1,\ldots,\theta_n$ and two clustering vectors, $\mathbf{s} = \{s_1,\ldots,s_{k_1}\}$ and $\mathbf{s}' = \{s'_1,\ldots,s'_{k_2}\}$, Rand's measure d is defined by

$$d(\mathbf{s},\mathbf{s}') = \frac{\sum_{i=1}^{n}\sum_{j=i+1}^{n}\gamma_{ij}}{\binom{n}{2}}, \tag{18.13}$$

where

$$\gamma_{ij} = \begin{cases} 1, \text{ if there exist } k \text{ and } k' \text{ such that both } \theta_i \text{ and } \theta_j \text{ are in both } s_k \text{ and } s'_{k'}, \\ 1, \text{ if there exist } k \text{ and } k' \text{ such that } \theta_i \text{ is in both } s_k \text{ and } s'_{k'} \text{ while } \theta_j \\ \quad \text{is in neither } s_k \text{ or } s'_{k'}, \\ 0, \text{ otherwise.} \end{cases} \tag{18.14}$$

In practice, we use a simple computational formula for d:

$$d(\mathbf{s},\mathbf{s}') = \Big[\binom{n}{2} - \frac{1}{2}\Big\{\sum_{i=1}^{n}(\sum_{j=i+1}^{n} n_{i,j})^2 + \sum_{j=1}^{n}(\sum_{i=j+1}^{n} n_{i,j})^2\Big\} + \sum_{i=1}^{n}\sum_{j=i+1}^{n} n_{i,j}^2\Big] \Big/ \binom{n}{2}, \tag{18.15}$$

where $n_{i,j}$ is the number of points simultaneously classified in the ith cluster of $\mathbf{s}$ and the jth cluster of $\mathbf{s}'$. Note that d ranges from 0 to 1. When $d = 0$, the two clusterings have no similarities, and when $d = 1$ the clusterings are identical. Rand's measure will be computed in each MCMC step. A boxplot showing its distribution will also be displayed in one instance.

Simulation studies are conducted under several different combinations of the true model and different sample sizes. Throughout this section and in the next, we shall work with the choice $M = 1$ for the precision measure of associated Dirichlet process. To see how accurate the clustering is, we also monitor Rand's measure and the number of clusters formed by the Dirichlet mixture process. All of simulation work is done by the package R (version 2.0.1).

18.3.1. *One dimension*

To give a simple example we consider a univariate predictor X having the following distribution.

$$X \sim 0.3N(-0.9,\sigma^2) + 0.2N(-0.3,\sigma^2) + 0.4N(0.4,\sigma^2) + 0.1N(1.0,\sigma^2),$$

where $\sigma^2 = 0.01, 0.02, 0.03, 0.04$ are considered.

First generate 100 samples of X from above mixture of normals and generate Y using four linear functions with different coefficients are given within each group.

$$f_1(x) = 2.3 + 1.1x, \quad f_2(x) = 1.5 - 0.6x, \quad f_3(x) = 0.8 + 0.9x, \quad f_4(x) = 1.7 - 0.2x,$$

with an additive noise term having mean 0 and variance $\tau^2 = 0.03$.

In the MCMC sampling step, we generate 5000 samples and ignore the first 1000

as a burn-in period. For appropriate selection of burn-in period, the trace plot of the number of distinct clusters was used and 1000 steps were found to be sufficient. This simulation work is repeated 100 times. Each simulation took about 5 minutes. It is found that the performances of our two methods DM-AVE and DM-ML are almost identical in every occasion.

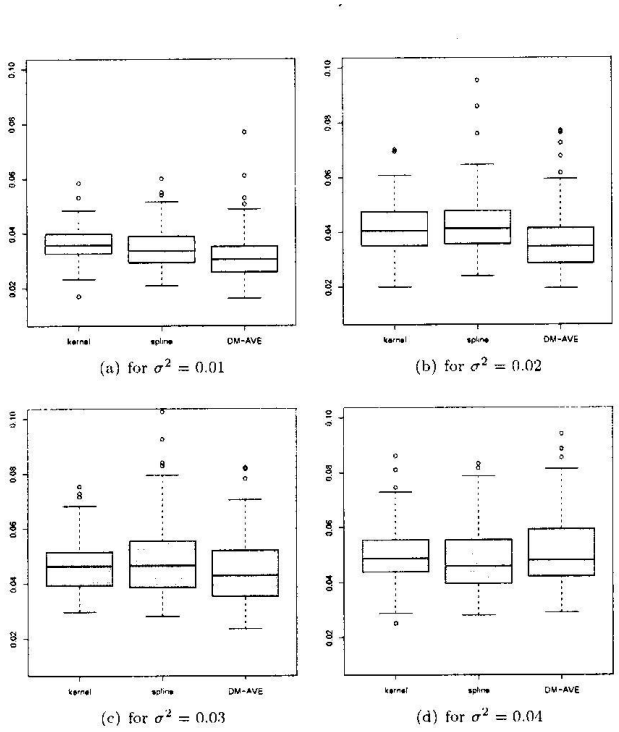


Fig. 18.1. (One dimension) : The boxplots of L_2-error. Rand's measures on average are 0.94, 0.87, 0.83 and 0.81, respectively.

For kernel estimation, we use a normal kernel with the help of the R function "ksmooth" and choose bandwidth by the default mechanism in that program. We also monitor the empirical L_2-error and L_1-error. It shows that our method has good performance for all the cases we investigated. Larger the value of σ^2

is, more overlapped the groups are, and smaller is the value of Rand's measure. This is expected because if some of clusters are severely overlapped, it is hard to separate them.

Comparing the errors by means of Wilcoxon rank test in the cases $\sigma^2 = 0.01$ and 0.02, we conclude that the median error of DM-AVE method is significantly smaller than that of other two methods at the 5% level of significance. For $\sigma^2 = 0.03$, the difference between DM-AVE and spline is significant but the difference with the kernel estimate is not significant. For $\sigma^2 = 0.04$, none of the differences are significant.

For a typical case, we monitor the number of clusters is given which in Figure 18.2. The graph of the confidence band for one such instance is shown in Figure 18.3.

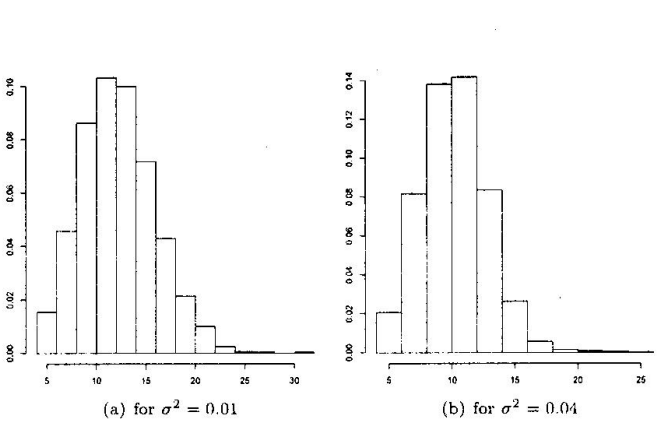


Fig. 18.2. (One dimension) : Histograms of the distribution of k

Now let us consider the case that the true regression function is not linear. Let X be generated by the following distribution

$$X \sim 0.3N(-1.9, \sigma^2) + 0.3N(-0.3, \sigma^2) + 0.4N(1.4, \sigma^2),$$

where $\sigma^2 = 0.01, 0.02, 0.03, 0.04$ are considered. The regression functions in each

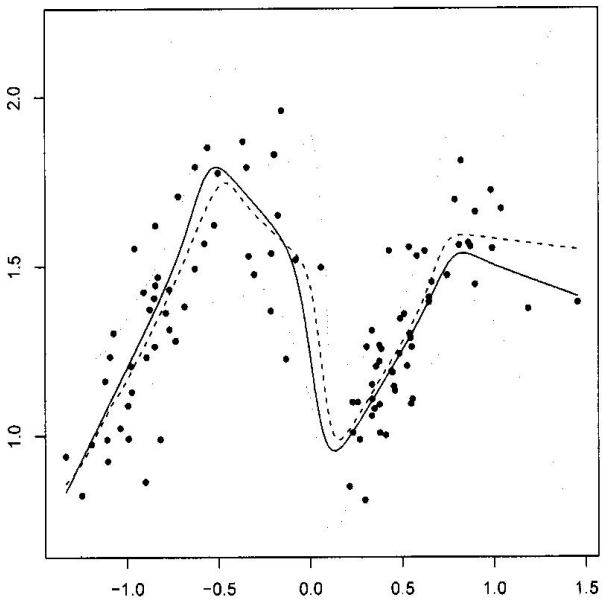


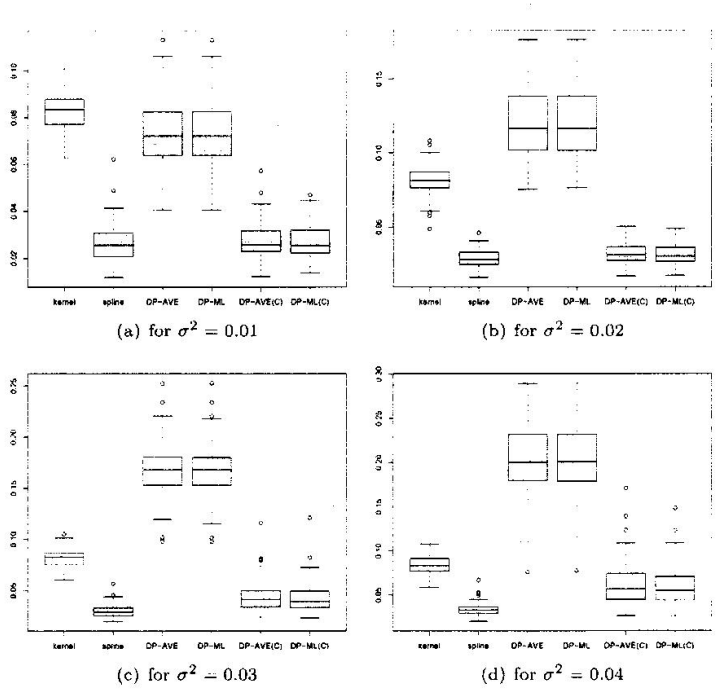
Fig. 18.3. (One dimension) : The plots of data and fitted regression function with $\sigma^2 = 0.03$ and $\tau^2 = 0.03$. The solid line is the true mean function and dotted line is fitted line using DM-AVE. 95% confidence band.

cluster are given by

$$f_1(x) = 0.3 + 0.3\sin(2\pi x), \quad f_2(x) = 1.2 - 0.6\sin(2\pi x), \quad f_3(x) = 0.5 + 0.9\sin(2\pi x).$$

Note here that we have three clusters in this case. Again we generate 5000 MCMC samples after 1000 burn-in and repeated the experiment 100 times. In this case, we use the cubic regression as well as the linear regression. Figure 18.4 tells us that the cubic regression performs better than the linear regression method and other methods. Formal rank tests shows that cubic DM-AVE method is significantly better than the kernel method. The spline method here performs slightly better than cubic DM-AVE although the difference is not significant. It may be noted that the cubic is also not the correct functional form of the regression functions in the groups. Thus the DM-AVE method performs reasonably well even under

misspecified models provided that a flexible function like a cubic is used.



(a) for $\sigma^2 = 0.01$ (b) for $\sigma^2 = 0.02$

(c) for $\sigma^2 = 0.03$ (d) for $\sigma^2 = 0.04$

Fig. 18.4. (One dimension) : The case where the true model is not linear. The plots of L_1-error are shown. Average Rand's measures are 0.97, 0.96, 0.95 and 0.94, respectively.

18.3.2. *Two dimension*

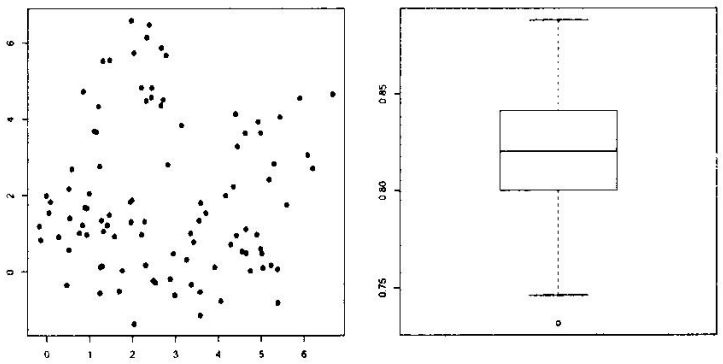
We consider $X = (X_1, X_2)^T$ distributed as a mixture of bivariate normal

$$X \sim 0.3N_2\left(\begin{pmatrix}1\\1\end{pmatrix},\Sigma\right) + 0.2N_2\left(\begin{pmatrix}2\\5\end{pmatrix},\Sigma\right) + 0.4N_2\left(\begin{pmatrix}4\\0.5\end{pmatrix},\Sigma\right) + 0.1N_2\left(\begin{pmatrix}5\\3\end{pmatrix},\Sigma\right)$$

where the regression functions in the subpopulations are given by

$$f_1(X) = 2.3 + 1.1X_1 - 2X_2, \quad f_2(X) = 1.5 - 0.6X_1 + 1.2X_2,$$
$$f_3(X) = 0.8 + 0.9X_1 - 1.1X_2, \quad f_4(X) = 1.7 - 0.2X_1 + 1.2X_2,$$

and the dispersion matrix Σ for the error given the groups label is taken of the form $\sigma^2 I_2$. We consider five different choices 0.2, 0.4, 0.6, 0.8 and 1.0 of σ^2 in the simulations. The error variance, τ^2, is taken to be 0.1. We generate $n = 100$ samples from the distribution and obtain the estimate of the regression function and the related quantities based on the sample. We generate 5000 MCMC samples ignoring the first 1000 as burn-in. The whole experiment is repeated 100 times. For one such replication, Figure 18.5 shows the data plot and the boxplot of Rand's measure obtained over different MCMC iterations.

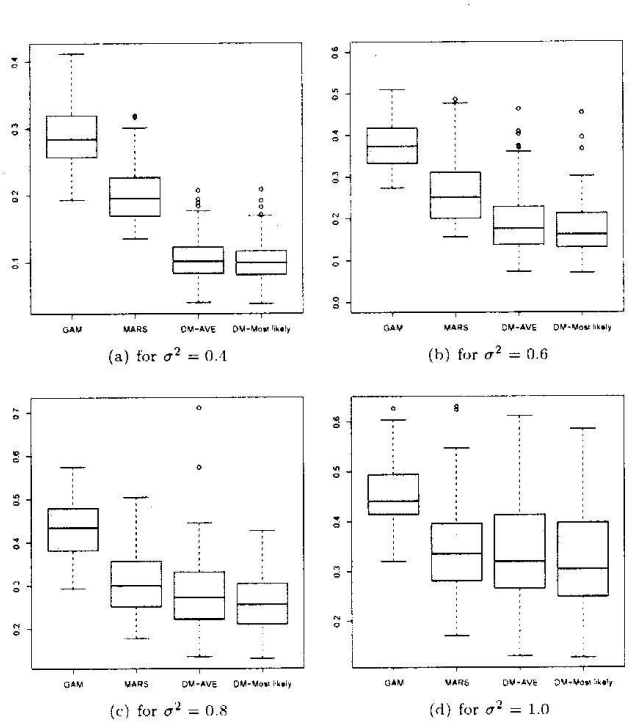


(a) The plot of X (b) Boxplot of Rand's measure

Fig. 18.5. (Two dimension): Data plot and boxplot of Rand's measure with $\sigma^2 = 0.8$.

In this case, the true number of clusters is four and those clusters are overlapped with each other. As the value of σ^2 determines how much the clusters are overlapped, we consider moderate values of σ^2.
We compare the L_2-error of DM-AVE and DM-ML with those of GAM and MARS. To allow certain flexibility, we set the degree of interaction to two, which means that it allows interaction between two variables.

Wilcoxon rank test comparing the medians of L_2-errors shows that DM-ML and DM-AVE are comparable, while those of GAM and MARS are higher at 5% level of significance except for $\sigma^2 = 1.0$, where the advantage over MARS is not statistically significant.



(a) for $\sigma^2 = 0.4$ (b) for $\sigma^2 = 0.6$

(c) for $\sigma^2 = 0.8$ (d) for $\sigma^2 = 1.0$

Fig. 18.6. (Two dimension) : The boxplots of the L_2-error.

18.3.3. *Higher dimension*

We consider the case when the predictor X has 10 variables and distributed as a mixture of four multivariate normal:

$$X \sim \sum_{j=1}^{4} \omega_j N_{10}\Big(\mu_j, \Sigma\Big).$$

We let $\Sigma = \sigma^2 I_{10}$, $\sigma^2 = 0.2, 0.3, 0.4, 0.5$ and $\tau^2 = 0.1$. The four mean vectors are given by

$$\mu_1 = (1,1,1,1,2,4,2,4,4,0.5)^T, \quad \mu_2 = (4,0.5,5,3,5,3,1,1,1,1)^T,$$
$$\mu_3 = (2,5,2,5,4,0.5,4,0.5,5,3)^T, \quad \mu_4 = (5,3,1,1,1,1,2,5,2,5)^T.$$

Let the weights be $\omega = (0.3, 0.2, 0.4, 0.1)$ and the within subpopulation regression functions are given by

$$f_j(x) = \alpha_j + \beta_j^T x, \quad j = 1, \ldots, 4,$$

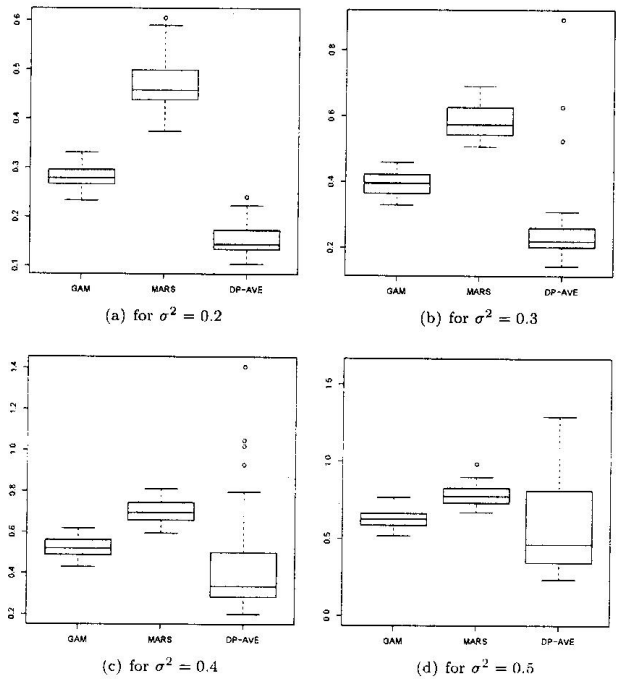
where $\alpha_1 = 2.3$, $\alpha_2 = 1.5$, $\alpha_3 = 0.8$, $\alpha_4 = 1.7$ and β_j's are the vectors of length 10 given by

$$\beta_1 = (1.1, -2, 1.1, -2, -0.6, 1.2, -0.6, 1.2, 0.9, -1.1)^T,$$
$$\beta_2 = (0.9, -1.1, -0.2, 1.2, -0.2, 1.2, 1.1, -2, 1.1, -2)^T,$$
$$\beta_3 = (-0.6, 1.2, -0.6, 1.2, 0.9, -1.1, 0.9, -1.1, -0.2, 1.2)^T,$$
$$\beta_4 = (-0.2, 1.2, 1.1, -2, 1.1, -2, -0.6, 1.2, -0.6, 1.2)^T.$$

We generated 200 sample data from the resulting population and obtain estimates of regression. The whole simulation is repeated 100 times. We consider four different σ^2 values 0.2, 0.3, 0.4 and 0.5. Our method DM-AVE is compared with GAM and MARS. In the MCMC sampling scheme, we generate 5000 samples after ignoring first 1000 as burn-in. Figure 18.7 shows that DM-AVE is better than GAM and MARS for all four choices of σ^2. The differences are significant at 5% significance level according to the Wilcoxon rank test.

18.4. Conclusions

We argued that in many natural applications, the population may be viewed as an aggregate of some hidden subpopulations, in each of which a possibly different but simple regression regime works. The number of hidden subpopulations is also considered as unknown. The overall regression function may then be estimated by identifying the missing subpopulation labels and estimating regression function parametrically in each group of data corresponding to same subpopulation labels. The approach automatically leads to estimators not affected by the curse of dimensionality problem. Moreover, assuming that the data clusters are identified fairly accurately, the method seems to enjoy nearly parametric rate of convergence. However, the uncertainty in identifying the clusters should be taken into account. In order to find the data clusters, we follow the Bayesian approach



(a) for $\sigma^2 = 0.2$ (b) for $\sigma^2 = 0.3$

(c) for $\sigma^2 = 0.4$ (d) for $\sigma^2 = 0.5$

Fig. 18.7. (Ten dimension): Boxplots of L_1-error with respect to σ^2 values. Average Rand's measures are 0.84, 0.77, 0.76 and 0.76, respectively.

and consider the Dirichlet process mixture prior on the distribution of the regressor variable, which has the ability to automatically identify clusters in MCMC iterations. By averaging over possible estimates corresponding to different clustering in these MCMC iterations, we obtain a very stable and accurate estimator of the regression function. We compare the performance of our estimator with some popular nonparametric estimators and find that when the model assumptions hold, our estimator has significantly smaller estimation error. The effect is particularly pronounced in higher dimension. There it outperforms the GAM and

MARS estimators which are not even prone to curse of dimensionality problem. The strength of our method seems to come from the assistance of the model and the near parametric rate of precision compared to other nonparametric methods.

18.5. Acknowledgments

Research of both authors are partially supported by NSF grant number DMS-0349111 awarded to the second author.

References

1. Breiman,L. J. H. Friedman, R. Olshen and C. J. Stone, (1984) *Classification and Regression Trees*, Belmont, CA: Wadsworth.
2. Breiman,L. (1996). Bagging predictors. *Machine Learning* **24**, 123–140
3. Breiman,L. (2001). Random forests. *Machine Learning* **45**, 5–32 .
4. DeSarbo W.S.and W. L. Cron, (2001)*J. Classification* **5**, 249–282 (2001).
5. Donoho,D.L. (1988). One-sided inference about functionals of a density. *Ann. Statist.* **16**, 1390–1420.
6. Efromovich,S. (1999). *Nonparametric Curve Estimation: Methods, Theory and Applications.* Springer, New York.
7. Escobar,M.D. (1995)*J. Amer. Statist. Assoc.* **90**, 557–588.
8. Ferguson,T.S. (1973). A Bayesian analysis of some nonparametric problems. *Ann. Statist.* **1**, 209–230.
9. Friedman,J.H. (1991). Multivariate adaptive regression splines (with discussion). *Ann. Statist.* **19**, 1–141.
10. Geweke,J. and M. Keane, (2005). *Smoothly Mixing Regressions* Technical Report, University of Iowa (IA, USA).
11. Ghosal,S., J. K. Ghosh and R. V. Ramamoorthi, (1999). Posterior Consistency of Dirichlet Mixtures in Density estimation. *Ann. Statist.* **27**, 143–158 .
12. Hastie,T.J. and R. I. Tibshirani, (1990). *Generalized Additive Models.* (Chapman and Hall).
13. Hoerl,A. and R. Kennard, (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **12**, 55–67 .
14. Hubert L. and P. Arabie, (1985). Comparing partitions. *J. Classification* **2**, 193–218.
15. MacEachern S. and P. Müller, *J. Comput. Graph. Statist.* **7** 223–228.
16. Müller,P. A. Erlkanli and M. West, (1996). Bayesian curve fitting using multivariate normal mixtures. *Biometrika* **83**, 67–79 .
17. Rand,W.M. (1971). Objective criteria for the evaluation of clustering methods. *J. Amer. Statist. Assoc.* **66**, 846–850 .
18. Spath,H. (1979). Algorithm 39: Clusterwise Linear Regression. *Computing* **22**, 367–373 .
19. Tibshirani,R. (1996)*J. Roy. Statist. Soc. Ser. B* **58**, 267–288.
20. Van Aelst,S. X. Wang, R. H. Zamar and R. Zhu,(2006). Linear grouping using orthogonal regression. *Computational Statistics and Data Analysis* **50**, 1287–1312.

21. Wahba,G. (1978). Improper priors, spline smoothing and the problem of guarding against model errors in regression. *J. Roy. Statist. Soc., Ser. B* **40**, 364–372.