The Korean Communication
in Statistics Vol. 12 No. 3, 2005
pp. 713-727

Adaptive Regression by Mixing for Fixed Design

Jong Chul Oh[1], Yun Lu[2], and Yuhong Yang[3]

Abstract

Among different regression approaches, nonparametric procedures perform well under different conditions. In practice it is very hard to identify which is the best procedure for the data at hand, thus model combination is of practical importance. In this paper, we focus on one dimensional regression with fixed design. Polynomial regression, local regression, and smoothing spline are considered. The data are split into two parts, one part is used for estimation and the other part is used for prediction. Prediction performances are used to assign weights to different regression procedures. Simulation results show that the combined estimator performs better or similarly compared with the estimator chosen by cross validation. The combined estimator generates a similar risk to the best candidate procedure for the data.

Keywords : ARM, Model Selection, Performance Criteria

1. Introduction

Regression analysis is a popular statistical technique. Regression estimation includes parametric and nonparametric approaches. Parametric approaches are simple, better interpretable, and highly efficient when the chosen models are appropriate. However, parametric methods can fail when the true form of the regression function f is wrongly specified. Parametric approaches also have the disadvantage of lacking flexibility. Another collection of procedures is nonparametric regression techniques including smoothing (smoothing spline, etc.) and parametric approximation (in terms of polynomials, etc.). Nonparametric regressions are more flexible compared with parametric methods. In this paper, we focus on nonparametric regressions.

Since various methods are available, we need to decide what is the right method for the data at hand. Several model selection criteria have been proposed, including Akaike information

1) Associate Professor, Department of Informatics and Statistics, Kunsan National University, Kunsan 573-701, Korea
E-mail : ohjc@kunsan.ac.kr
2) Department of Statistics, Iowa State University, Ames, Iowa, USA
3) Professor, School of Statistics, University of Minnesota, Minneapolis, Minnesota, USA

criterion (Akaike 1973), Baysian information criterion (Schwartz 1978), cross-validation (Stone, 1974), etc. However, model selection can generate a rather unstable estimator. A very different model can be selected because of a small perturbation of the data. Estimators of the regression function based on model selection often have large variance due to the unstableness of the model selection.

An alternative to model selection is model combination. Yang (2000) has shown that given several regression procedures, a properly combined procedure behaves asymptotically as well as the best procedure in terms of rate of convergence under Gaussian errors. Yang (2001) proposed a practical algorithm with theoretically proven properties. The combination method is adaptive regression by mixing (ARM), which is used to combine estimators of a regression function based on the same data. The algorithm can be used when there are multiple candidate error distributions and does not require normality. Yang's results showed that under mild conditions, the combined estimator performs optimally in rates of convergence.

The work of Yang (2001) focused on random design. The goal of this creative component is to provide a practically feasible weighting method for one-dimensional regression with fixed design. Three nonparametric regression approaches are considered (polynomial regression, smoothing spline, and local regression). The creative component is organized as follows. In next section, we present the preliminary knowledge about different regression procedures. Then, the proposed combing method is compared to the model selection methods such as cross-validation and generalized cross-validation in simulations. Section 4 is the result of a real data set. Finally, a conclusion follows in section 5.

2. Some Preliminaries

2.1 Regression Analysis

Let us suppose n observations are taken on a random variable Y at n predetermined values of independent variable X. Let (x_i, Y_i), $i=1,\cdots,n$, be the values of X and Y which result from this sampling scheme and assume that the dependent variable Y_i differs from $f(x_i)$ by a random quantity $\sigma(x_i)\cdot\varepsilon_i$. The equation $Y_i=f(x_i)+\sigma(x_i)\cdot\varepsilon_i$, $i=1,\cdots,n$, is a regression model in which f is the unknown regression function. ε is an uncorrelated random error with zero mean, and the unknown function $\sigma(x)$ controls the variance of the random error given $X=x$. If we assume $\sigma(x_i)$ are the same for all the x_i's, the regression model can be written as $Y_i=f(x_i)+\varepsilon_i$, $i=1,\cdots,n$, where ε has zero mean and a common variance σ^2.

Regression estimation includes parametric and nonparametric approaches. A parametric

regression model assumes that the form of f is known except for finitely many unknown parameters. In nonparametric regression, it is not assumed that f takes parametric form which gives us great flexibility. Nonparametric regression techniques rely more heavily on the data for information about f than their parametric counterparts. Nonparametric estimators are less efficient than parametric estimators when the parametric models are valid and simple. However, if an incorrect parametric model is used, the nonparametric regression techniques will be much better than parametric methods.

In this paper, we will focus on the three regression methods: polynomial regression, local regression, and smoothing spline.

2.2 Performance Criteria

Suppose there are several families of estimators for the regression function. Since the selection of an estimator can be made subjectively, an objective choice will usually be preferred. Suppose we consider a class of estimators for f, $C(\Lambda) = \{f_\lambda : \lambda \in \Lambda\}$, with Λ representing some index set. The problem to be considered is selection of a best estimator $\hat{f}_\lambda$ of f from among the elements of $\{f_\lambda : \lambda \in \Lambda\}$. There are certain criteria which are widely accepted and used.

The *loss* in estimating f on $[0,1]$ is defined as

$$L(\lambda) = \int_0^1 (f(x) - \hat{f}_\lambda(x))^2 dx$$

$L(\lambda)$ represents a natural measure of the closeness of $\hat{f}_\lambda$ to f. The expected value of $L(\lambda)$ is called the *risk*, i.e.

$$R(\lambda) = E\,L(\lambda)$$

Both $L(\lambda)$ and $R(\lambda)$ provide assessments of an estimator's performance with smaller values of the criteria being indicative of better estimation. A value of λ that minimizes the loss provides a best estimate of f among those considered λ's for the particular data set in question while the value of that minimizes the risk can be viewed as the best for prediction of future responses or estimation of f in repeated sampling.

Of course, the risk is unknown since it depends on the unknown regression function f. Monte Carlo methods can be used to simulate the risk. In our case, a large number of new x values (n.xnew = 500), say X_i, $1 \le i \le 500$, are generated independently from the uniform distribution on $[0,1]$. The loss of the estimator can be calculated by

$$L = \frac{\sum_{i=1}^{n.xnew} (f(X_i) - \hat{f}_{n,\lambda}(X_i))^2}{500}.$$

The procedure is repeated *runct* times and the risk is estimated by the average of the loss

$$\hat{R}=\frac{\sum_{k=1}^{runct} L_k}{runct}.$$

2.3 Polynomial Regression

Polynomial regression estimators represent an important cornerstone in the theory of nonparametric regression. To estimate f , the model $Y_i=f(x_i)+\varepsilon_i$, $i=1,\cdots,n$, $0\le x_1<\ldots<x_n\le 1$ can be rewritten in an alternative form (Eubank, 1988)

$$Y_i=\sum_{j=0}^{m-1}\beta_j x_i^j+rem(x_i)+\varepsilon_i$$

with $rem(x)=[(m-1)!]^{-1}\int_0^t f^{(m)}(t)(x-t)_+^{m-1}dt.$

If the $rem(x_i)$ can be assumed to be small, we can estimate f using $\hat{f}=\sum_{j=0}^{m-1} b_j x_i^j$, and the estimator $\hat{f}$ will be called an mth order polynomial regression estimator.

2.4 Smoothing Spline

Splines are generally defined as piecewise polynomials (Eubank, 1988) in which curve (or line) segments are constructed individually and then pieced together. In a spline model, a turning point is represented by a spline knot. There are different types of splines. A smoothing spline results, generally, from minimizing

$$S(f)=\sum_{i=1}^{n}[y_i-f(x_i)]^2+\lambda\int_0^1\left[\frac{df^m(x)}{dx^m}\right]^2dx$$

where $0\le x_1\le x_2\le\cdots\le x_n\le 1$.

The minimization of $S(f)$ produces the function $\hat{f}$. Before that minimization can be obtained, however, the smoothing parameter λ and m must be selected. The value of m determines the order of the smoothing spline. The value of λ determines the amount of smoothing and thus governs the tradeoff between smoothness and goodness-of-fit. If λ=0, we would then be attempting to minimize the residual sum of squares. If $\lambda\to\infty$, the smooth approaches the least squares line. The value of λ is often selected by cross validation.

2.5 Local Regression

Local regression was called locally weighted regression (Stone, 1977; Cleveland, 1979). In local regression the size of a neighborhood is referred to as the bandwidth and the neighborhoods are overlapping. The objective of local regression is to identify the model that

is appropriate for each data segment. Local regression is used to model a relation between a predictor variable (or variables) X and response variable Y, which is related to the predictor variables. For a fitting point x and a bandwidth $h(x)$, only observations within window $(x-h(x), x+h(x))$ are used to estimate $f(x)$. The weights for the x_i depend on their distance from x. Specifically, the weight assigned to x_i for obtaining the predicted value at x is

$$w_i(x) = W\left(\frac{x_i - x}{h(x)}\right),$$

where $W(.)$ is a weight function that assigns largest weights to observations close to x and assigns zero weights for observations outside the window. Within the smoothing window, $f(x)$ is approximated by polynomial.

The bandwidth $h(x)$ has a very important impact on the local regression fit. If $h(x)$ is too small, insufficient data fall within the smoothing window, unnecessarily large variance will result. On the other hand, if $h(x)$ is too large, important features of the mean function $f(x)$ may be distorted or loss completely due to over smoothing. The fit will have large bias. The bandwidth must be chosen to compromise this bias-variance trade-off. The simplest case is to choose a constant bandwidth $h(x)=h$. Another way is to choose $h(x)$ so that the local neighborhood contains a specified number of points na, and this method is called nearest neighbor bandwidth. This method can reduce the problems caused by data sparsity.

2.6 Regression Functions Considered

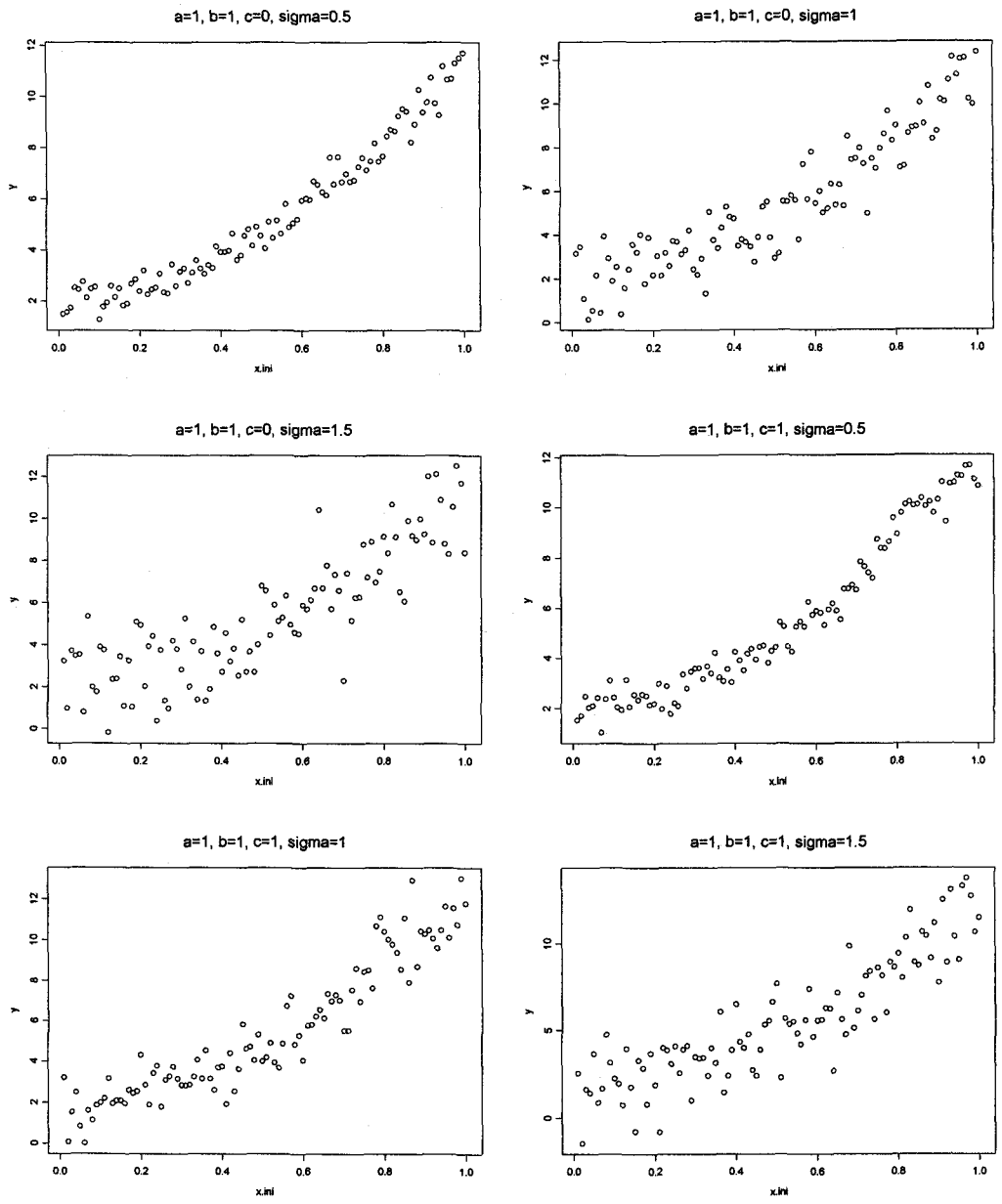
In this creative component, the regression setting $Y_i = f(x_i) + \varepsilon_i$ with fixed design points $x_i = i/n$ are considered, where $i = 1, 2, \ldots, n$. The error term ε is assumed to follow normal distribution with mean 0 and variance σ^2 (unknown). Our goal is to estimate the regression function f based on data $Z^n = (x_i, Y_i)_{i=1}^n$. Suppose there are J different regression procedures to estimate f. Adaptive regression by mixing (ARM) is used for combining different procedures. In the next two sections, we demonstrate applications of ARM for simulation data and real data.

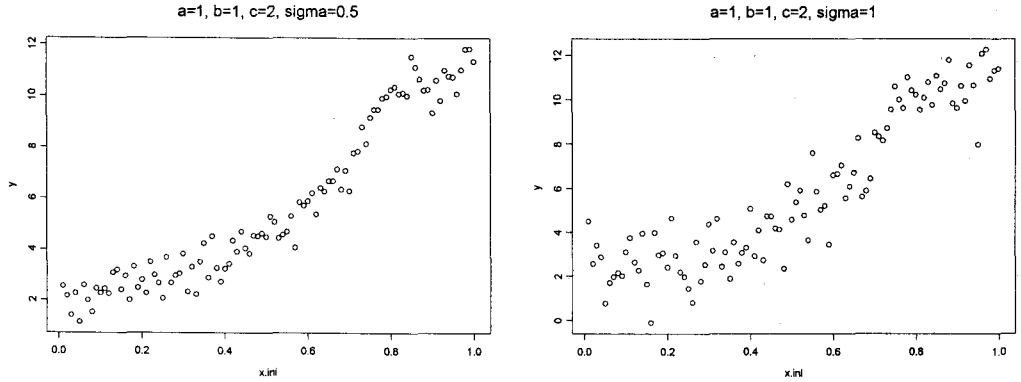
The true underlying regression function is the following functions on $[0,1]$:

$$f(x) = 1 + 8x^2 + ae^{bx} + ce^{-200(x-0.8)^2}.$$

The sample size is taken to be $n=100$, $a=0$ and 1, $b=0.1$ and 1, $c=0$, 0.1, 1, and 2, and $\sigma=0.5$, 1, and 1.5. The squared L_2 risk is used as a measure of discrepancy in estimating the regression function. The squared L_2 loss is simulated using $n.xnew=500$ new independent X_{new}'s from the uniform distribution between $[0,1]$. The squared L_2 risk is computed based on $runct=200$ replications. All the simulations are conducted using Splus 2000. Typical

realizations of data are plotted in Figure 1.





<Figure 1> A typical realization of data for $f(x) = 1 + 8x^2 + ae^{bx} + ce^{-200(x-0.8)^2}$

3. ARM by Systematic Half-half Splitting

In this section, we demonstrate applications of ARM for simulation data for

$$f(x) = 1 + 8x^2 + ae^{bx} + ce^{-200(x-0.8)^2}$$

where a=*0* and *1*, b=*0.1* and *1*, c=*0, 0.1, 1*, and *2*, σ=*0.5, 1*, and *1.5*. Our goal is to estimate the regression function f based on data $Z^n = (x_i, Y_i)_{i=1}^n$. We consider three regression methods: polynomial regression (j=*1*), smoothing spline (j=*2*), and local regression (j=*3*). The order of polynomial regression is chosen by AIC. We use the default choice of generalized cross-validation for choosing the smoothing parameter for smoothing spline. The bandwidth of local regression is selected by generalized cross-validation.

In this section, adaptive regression by mixing (ARM) is used for combining different procedures. Generalized cross-validation and cross-validation are used as criteria to do model selection. The following sections explained the detailed procedures.

3.1 ARM Using Common $\widehat{\sigma^2}$(*ARMC*)

In this section, we systematically split the data into two parts according to the x values. The first part is used for estimation by each regression procedure and the second part is used to assess the prediction performance and assign weights to regression procedures. In order to assign weights to different procedures, we need to estimator σ^2. In this section, we use one common estimator $\widehat{\sigma^2}$ for all different regression procedures. We assume the observations are ordered in x. The detailed combining procedures are the following:

Step 0. Obtain estimates $\hat{f}_{n,j}(x; Z)$ based on data $Z^n = (x_i, Y_i)_{i=1}^n$, using regression

procedure j, $j=1, 2, 3$.

Step 1. Split the data into two parts

$$Z^{(1)} = (x_{2l-1}, Y_{2l-1})_{l=1}^{n/2}, \quad Z^{(2)} = (x_{2l}, Y_{2l})_{l=1}^{n/2}$$

Rearrange the data so that $Z' = (x_i', Y_i')_{i=1}^{n}$, where $Z^{(1)} = (x_i', Y_i')_{i=1}^{n/2}$, and $Z^{(2)} = (x_i', Y_i')_{i=n/2+1}^{n}$.

Step 2. Obtain estimates $\hat{f}_{n/2,j}(x; Z^{(1)})$ based on $Z^{(1)}$ for $j=1, 2, 3$.

Estimate the variance function σ^2 by

$$\hat{\sigma}^2_{n/2,c} = \frac{1}{2(n/2-1)} \sum_{i=1}^{n/2-1} (Y'_{(i+1)} - Y'_{(i)})^2,$$

where $Y_{(i)}$ denotes the observed response at the ith smallest x value (Rice, 1984).

Step 3. For each j, evaluate predictions. For $Z^{(2)}$, predict Y_i' by $\hat{f}_{n/2,j}(x_i)$. Compute

$$E_j = \frac{(2\pi)^{-n/4} \exp(-\sum_{i=n/2+1}^{n} ((Y_i' - \hat{f}_{n/2,j}(x_i'))^2 / (2\hat{\sigma}^2_{n/2,c}))}{\hat{\sigma}^{n/2}_{n/2,c}}$$

Step 4. Compute the current weight for procedure j. Let $W_j = \frac{E_j}{\sum_{l=1}^{3} E_l}$.

Step 5. The final estimator is

$$\hat{f}(x) = \sum_{j=1}^{3} W_j \hat{f}_{n,j}(x)$$

3.2 Model Selection Using Cross-validation

Instead of using ARM to obtain combined estimator, we can use cross-validation to do model selection. The data are split into two parts. We use the first part of the data to do estimation and use the second part of the data to assess prediction. Prediction residual sum of squares is used as a criterion, and the model generates the smallest prediction residual sum of squares is selected.

3.2.1 Cross-validation with Systematic Half-half Splitting (CV)

In this section, half of the data is used for estimation and the other half for prediction.

Step 0. Obtain estimates $\hat{f}_{n,j}(x; Z)$ based on data $Z^n = (x_i, Y_i)_{i=1}^{n}$, using regression procedure j, $j=1, 2, 3$.

Step 1. Split the data into two parts as described in Step 1 of section 3.1.

Step 2. Obtain estimates $\hat{f}_{n/2,j}(x; Z^{(1)})$ based on $Z^{(1)}$ for *j=1, 2, 3.*

Step 3. For each *j*, evaluate predictions. For $Z^{(2)}$, predict Y_i' by $\hat{f}_{n/2,j}(x_i)$.
CV is used as a criterion to select the best regression procedure.

$$CV(\hat{f}_j) = \sum_{i=n/2+1}^{n} (Y_i' - \hat{f}_{n/2,j}(x_i'))^2$$

Step 4. The final estimator is $\hat{f}_{n,j}(x; Z)$, with procedure *j* generating the smallest CV.

3.2.2 Cross-validation with 1/4 of The Data for Estimation (CV1)

In this section, one the fourth of the data is used for estimation and 3/4 of the data is used for prediction. The model selection procedure is the same as in section 3.2.1 except Step 1. The new Step 1 is as follows:

Step 1. Split the data into two parts $Z^{(1)} = (x_{4l-3}, Y_{4l-3})_{l=1}^{n/4}$, $Z^{(2)}$ are the rest of the data. Rearrange the data so that $Z' = (x_i', Y_i')_{i=1}^{n}$, where $Z^{(1)} = (x_i', Y_i')_{i=1}^{n/4}$, and $Z^{(2)} = (x_i', Y_i')_{i=n/4+1}^{n}$.

3.2.3 Cross-validation with 3/4 of The Data for Estimation (CV2)

In this section, three the fourth of the data is used for estimation and 1/4 of the data is used for prediction. The model selection procedure is the same as in section 3.2.1 except Step 1. The new Step 1 is as follows:

Step 1. Split the data into two parts $Z^{(2)} = (x_{4l}, Y_{4l})_{l=1}^{n/4}$, $Z^{(1)}$ are the rest of the data. Rearrange the data so that $Z' = (x_i', Y_i')_{i=1}^{n}$, where $Z^{(1)} = (x_i', Y_i')_{i=1}^{3n/4}$, and $Z^{(2)} = (x_i', Y_i')_{i=3n/4+1}^{n}$.

3.3 Model Selection Using Generalized Cross-validation (GCV)

The generalized cross validation criterion was first proposed in the context of smoothing splines by Craven and Wahba (1979). This provides an approximation to cross validation and is easier to compute.

Step 0. Obtain estimates $\hat{f}_{n,j}(x; Z)$ based on data $Z^n = (x_i, Y_i)_{i=1}^{n}$, using regression procedure *j*, *j=1, 2, 3.*

Step 1. GCV is used as a criterion to select the best regression procedure.

$$GCV(\hat{f}_{n,j}) = n\frac{\sum_{i=1}^{n}(Y_i - \hat{f}_{n,j}(x_i))^2}{(n-df)^2}$$

For polynomial regression, *df=polynomial order+1* for smoothing spline, *df=tr(S)*, S is the implicit smoother matrix; for local regression, $df = \nu_1 = tr(L)$, where $n \times n$ matrix L maps the data to the fitted values.

Step 2. The final estimator is $\hat{f}_{n,j}(x; Z)$, with procedure j generating the smallest GCV.

3.4 Simulation

In this section, simulation is conducted for

$$f(x) = 1 + 8x^2 + ae^{bx} + ce^{-200(x-0.8)^2}$$

where *a=0* and *1*, *b=0.1* and *1*, *c=0*, *0.1*, *1*, and *2*. When *a=0* and *c=0*, the function is quadratic; when $a \neq 0$ and *c=0*, the function is a combination of quadratic and exponential; when $c \neq 0$, the function has one hump, and the size of the hump is proportional to the value of *c*. The above *a*, *b*, and *c* values are chosen in order to compare the performance of different regression procedures and to compare model combination with model selection for different functions.

The following are some simulation.

<Table 1> Ratios of the risks of CV, CV1, CV2, GCV to ARMC. The means and medians are calculated over all the 36 combinations of various a, b, c and sigma.

	cv/ARMC	cv1/ARMC	cv2/ARMC	gcv/ARMC
mean	1.0741	1.0895	1.0971	1.0733
s.e.	0.0040	0.0106	0.0062	0.0164
median	1.0723	1.1053	1.0908	1.1018

From Table 1, we can see that the means and medians are bigger than 1. The results suggest that in average ARMC generates smaller risks compared with model selection criteria CV, CV1, CV2, and GCV. It seems clear that ARMC is a better method above the averagely.

Table 2 is listed below to illustrate the performance of model selection criteria and ARMC for different functions. "Poly" stands for polynomial regression, "locreg" stands for local regression, and "smsp" stands for smoothing spline.

From Table 2, we can see that the performance of CV, CV1, CV2, and GCV is not very consistent. The result supports that model selection can generate unstable estimators.

ANOVA is conducted for the ratios of the risks of CV, CV1, CV2, GCV to ARMC and the results are listed below in Table 3. Because the effects of a and b values on the ratios are not significant, we only include the output for c, σ, and the interaction of c and σ.

<Table 2> Comparing model selection with model combination for different functions and different σ values.

a=1, b=1, c=0, runct=200

σ		poly	locreg	smsp	cv	cv1	cv2	gcv	ARMC
0.5	risk	.0094	.0131	.0158	.0117	.0124	.0122	.0132	.0109
	s.e.	.0006	.0008	.0009	.0007	.0007	.0007	.0009	.0006
1	risk	.0376	.0507	.0596	.0482	.0483	.0463	.0536	.0432
	s.e.	.0020	.0031	.0038	.0029	.0028	.0028	.0038	.0023
1.5	risk	.0864	.1121	.1171	.1117	.1126	.1048	.1070	.1010
	s.e.	.0052	.0055	.0062	.0063	.0060	.0057	.0062	.0054

a=1, b=1, c=0.1, runct=200

σ		poly	locreg	smsp	cv	cv1	cv2	gcv	ARMC
0.5	risk	.0089	.0124	.0139	.0113	.0119	.0111	.0117	.0102
	s.e.	.0005	.0007	.0007	.0006	.0007	.0006	.0007	.0005
1	risk	.0355	.0502	.0536	.0418	.0489	.0439	.0494	.0402
	s.e.	.0021	.0029	.0031	.0027	.0030	.0027	.0030	.0023
1.5	risk	.0774	.1073	.1058	.0926	.1008	.0962	.0973	.0883
	s.e.	.0056	.0064	.0061	.0055	.0064	.0067	.0063	.0053

a=1, b=1, c=1, runct=200

σ		poly	locreg	smsp	cv	cv1	cv2	gcv	ARMC
0.5	risk	.0582	.0293	.0285	.0331	.0302	.0358	.0290	.0303
	s.e.	.0006	.0008	.0008	.0012	.0009	.0011	.0009	.0009
1	risk	.0929	.0866	.0839	.0873	.0890	.0864	.0877	.0796
	s.e.	.0021	.0027	.0034	.0030	.0026	.0029	.0033	.0025
1.5	risk	.1400	.1505	.1429	.1429	.1452	.1450	.1510	.1334
	s.e.	.0056	.0059	.0065	.0058	.0060	.0065	.0067	.0053

a=1, b=1, c=2, runct=200

σ		poly	locreg	smsp	cv	cv1	cv2	gcv	ARMC
0.5	risk	.1913	.0363	.0357	.0379	.0362	.0412	.0357	.0357
	s.e.	.0012	.0009	.0010	.0016	.0009	.0024	.0010	.0010
1	risk	.2322	.1199	.1121	.1386	.1256	.1429	.1124	.1274
	s.e.	.0029	.0034	.0035	.0049	.0041	.0052	.0035	.0039
1.5	risk	.2939	.2314	.2229	.2449	.2420	.2412	.2320	.2313
	s.e.	.0054	.0061	.0064	.0063	.0064	.0059	.0066	.0059

The output in Table 3 below suggests that the effects of c, σ, and the $c*\sigma$ on the ratios are not consistent for CV, CV1, CV2, and GCV. The result again supports our previous claim that the estimators generated by model selection criteria are rather unstable.

<Table 3> ANOVA output for ratios of the risks of CV, CV1, CV2, GCV to ARMC. The values in the table are Pr>F.

Pr > F	cv/ARMC	cv1/ARMC	cv2/ARMC	gcv/ARMC
c	0.0539	<.0001	0.0491	<.0001
σ	0.0301	0.0004	<.0001	0.0024
$c*\sigma$	0.0658	0.0007	0.0102	<.0001

From Table 3, we can see that for most of the cases, c, σ, and the $c*\sigma$ have significant impact on the ratios of the risks. When we look at the simulation results of the 36 combinations of various a, b, c and σ, we can see that the data can be divided into two groups. Group 1 is when $c=1$ with $\sigma=1$, 1.5 and all the cases for $c=0$, $c=0.1$. Group 2 is when $c=1$ with $\sigma=0.5$ and all the cases for $c=2$. The results are summarized in Table 4.

<Table 4> Ratios of the risks of CV, CV1, CV2, GCV to ARMC for Group 1 and Group 2.

		CV/ARMC	CV1/ARMC	CV2/ARMC	GCV/ARMC
Group 1	mean	1.0786	1.1273	1.0874	1.1331
	s.e.	0.0050	0.0074	0.0068	0.0094
	median	1.0723	1.1181	1.0862	1.1164
Group 2	mean	1.0652	1.0140	1.1166	0.9538
	s.e.	0.0059	0.0083	0.0113	0.0154
	median	1.0703	1.0162	1.1197	0.9704

The data in Table 4 suggest that for group 1, ARMC performs better than all the model selection criteria; for group 2, GCV performs better than ARMC and CV1 generates similar risks compared to ARMC.

Recall that when $c\neq 0$, the function has one hump, and the size of the hump increases when the value of c increases. When $c=1$ with $\sigma=0.5$, and $c=2$, we can clearly see there is a hump. These cases belong to group 2. For the cases in group 1, the hump can not be seen clearly due to the random error.

The data for group 2 indicate that local regression and smoothing spline generate similar risks while polynomial regression performs much worse than the other two functions. Some model selection criteria can reject polynomial regression without difficulty. Because model combination can assign some weight to polynomial regression, thus the risks generated by ARMC can be bigger than some model selection criteria.

The differences between the three regression procedures in group 1 are not as obvious as in group 2. It will make model selection more difficult thus model combination ARMC shows advantages.

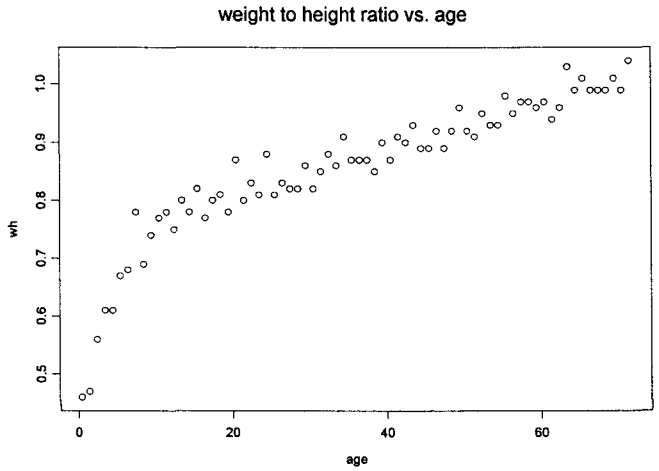
When one regression procedure performs much worse than others, ARMC can assign nonzero weight to the bad procedure thus generates bigger risk. In that case, we can assign

zero weight to the bad procedure and combine the rest procedures. We expect ARMC will perform better than model selection criteria based on our experience with the data for group 1.

In summary, ARMC performs better than model selection criteria above the averagely. ARMC has advantages especially when model selection is not very easy. When one regression procedure performs much worse than others, ARMC can assign nonzero weight to the bad procedure thus generates bigger risk. In that case, we can assign zero weight to the bad procedure and combine the rest procedures. We expect ARMC will perform better than model selection criteria based on our experience with the data for group 1.

4. Real data

In this section, we consider a real data set. The data set represents the ratios of weight to height for boys from a study by Eppright, et al. (1972) and the scatter plot is shown in Figure 2.



<Figure 2> The scatter plot for the data in Eppright, et al. (1972).

For the real data, we cannot use new independent variables and the according true values to assess the performance of regression procedures and ARM. Instead, we systematically split the data into an estimation set $Z_{(est)}$ (75%) and a test set $Z_{(pre)}$ (25%) and the average squared error in prediction was computed using the test data. There are four ways to split the data: for run 1, $Z_{(pre)}=(x_{4l+1}, Y_{4l+1})_{l=1}^{n/4}$; for run 2, $Z_{(pre)}=(x_{4l+2}, Y_{4l+2})_{l=1}^{n/4}$; for run 3, $Z_{(pre)}=(x_{4l+3}, Y_{4l+3})_{l=1}^{n/4}$; for run 4, $Z_{(pre)}=(x_{4l}, Y_{4l})_{l=1}^{n/4}$. The estimation set is treated the same as $Z^{(n)}$ in section 3. The performance of CV, and ARMC are compared for

the four runs and the results are shown in Table 5.

<Table 5> Comparing CV and ARMC for the real data set.

	poly	locreg	smsp	CV	ARMC
1	.001169	.000793	.000895	.000895	.000852
2	.001195	.000514	.000532	.000514	.000518
3	.000785	.000363	.000353	.000363	.000358
4	.001756	.001170	.001139	.001170	.001154
mean	.001226	.000710	.000730	.000736	.000720
s.e.	.000400	.000355	.000354	.000366	.000355

From Table 5, we can see that CV and ARMC have very similar performance and the pattern is similar to group 2 in Table 5. Since for this real data set, polynomial regression has higher squared error in prediction than local regression and smoothing spline, we expect it will give similar pattern compared with group 2 in Table 5. The real data set supports our findings using simulation data.

5. Conclusion

The main topic of this study is to compare the performance of model combination method ARMC and model selection criteria CV, CV1, CV2, and GCV. One-dimensional regression with fixed design is considered and the regression procedures are polynomial regression, local regression, and smoothing spline. Systematic half-half splitting is used such that the data are split into two parts: $Z^{(1)}$ and $Z^{(2)}$. The first part is used for estimation and the second part is used for prediction.

The results suggest that in average ARMC generates smaller risks compared with model selection criteria CV, CV1, CV2, and GCV. The result suggests that above the averagely, ARMC is indeed a better choice compared with model selection criteria.

When models are close in terms of a selection criterion, ARMC can be much better than model selections. However, when one regression procedure performs much worse than others, ARMC can generate bigger risk due to nonzero weight of the worst procedure. In order to improve the performance of ARMC, we can assign zero weight to the bad procedure and combine the rest procedures.

Systematic splitting is preferred in our study due to less time-consuming. If x is multi-dimensional, we expect random splitting with permutation will perform better since systematic splitting will be difficult even infeasible.

In summary, ARMC generates smaller or similar risk compared with model selection criteria and the risk is close to the risk of the best regression procedure. The result suggests that ARMC is indeed a good choice for model combination.

References

[1] Akaike, H., (1973). Information Theory and an Extension of the Maximum Likelihood Principle, in *Proceedings of the 2nd International Symposium on Information Theory*, eds. B.N. Petrov and F. Csaki, Budapest: Akademia Kiado, 267-281.

[2] Cleveland, W.S., (1979). Robust Locally Weighted Regression and Smoothing Scatterplots. *Journal of the American Statistical Association*, Vol. 74, 829-836.

[3] Craven, P. and G. Wahba, (1979). Smoothing Noisy Data with Spline Functions. *Numerisch mathematik*, Vol. 31, 377-403.

[4] Eppright, E.S., H.M. Fox, B.A. Fryer, G.H. Lamkin, V.M. Vivian, and E.S. Fuller, (1972). Nutrition of Infants and Preschool Children in the North Central Region of the United States of America. *World Review of Nutrition and Dietetics*, Vol. 14, 269-332.

[5] Eubank,R.L., (1988). *In Spline Smoothing and Nonparametric Regression*, 1st edition, Marcel Dekker, Inc.

[6] Rice, J., (1984). Bandwidth Choice for Nonparametric Regression, *The Annals of Statistics*, Vol. 12, 1215-1230.

[7] Schwartz, G., (1978). Estimating the Dimension of a Model, *The Annals of Statistics*, Vol. 6, 461-464.

[8] Stone, C.J., (1977). Consistent Nonparametric Regression, *Annals of Statistics*, Vol. 5, 595-620.

[9] Stone, M., (1974). Cross-validatory Choice and Assessment of Statistical Predictions (with discussion). *Journal of the Royal Statistical Society, Series B* Vol. 36, 111-147.

[10] Yang, Y., (2000). Combining Different Procedures for Adaptive Regression. *Journal of Multivariate Analysis*, Vol. 74, 135-161.

[11] Yang, Y., (2001). Adaptive Regression by Mixing. *Journal of American Statistical Association, Therory and Methods*, Vol. 96, 574-588

[Received June 2005, Accepted October 2005]