Journal of the American Statistical Association

ISSN: 0162-1459 (Print) 1537-274X (Online) Journal homepage: https://www.tandfonline.com/loi/uasa20

AdaBoost semiparametric model averaging prediction for multiple categories

Jialiang Li, Jing Lv, Alan T.K. Wan & Jun Liao

To cite this article: Jialiang Li, Jing Lv, Alan T.K. Wan & Jun Liao (2020): AdaBoost semiparametric model averaging prediction for multiple categories, Journal of the American Statistical Association, DOI: 10.1080/01621459.2020.1790375

To link to this article: https://doi.org/10.1080/01621459.2020.1790375

AdaBoost semiparametric model averaging prediction for multiple categories

Jialiang Li

Department of Statistics & Applied Probability, National University of Singapore, Singapore

Jing Lv *

School of Mathematics and Statistics, Southwest University, Chongqing, China

Alan T.K. Wan

Department of Management Sciences, City University of Hong Kong, Hong Kong

Jun Liao

School of Statistics, Renmin University of China, Beijing, China

*Corresponding author. (lvjing@swu.edu.cn)

Abstract

Model average techniques are very useful for model-based prediction. However most earlier works in this field focused on parametric models and continuous responses. In this paper, we study varying coefficient multinomial logistic models and propose a semiparametric model averaging prediction (SMAP) approach for multi-category outcomes. The proposed procedure does not need any artificial specification of the index variable in the adopted varying coefficient sub-model structure in order to forecast the response. In particular this new SMAP method is more flexible and robust against model mis-specification. To improve the practical predictive performance, we combine SMAP with the AdaBoost algorithm to obtain more accurate estimations of class probabilities and model averaging weights. We compare our proposed methods with all existing model averaging approaches and a wide range of popular classification methods via extensive simulations. An automobile classification study is included to illustrate the merits of our methodology.

Keywords: Boosting, Model averaging, Model misspecification, Prediction accuracy, Smoothing, Vary coefficient structure identification.

1 Introduction

As the business world embraces the concept of big data with the hope of maximizing prediction accuracy, many are seeking to understand this new movement and how it may impact the current landscape. Sophisticated learning tools based on massive data information are rapidly developed for many real problems in the last several years, driven by advances in computer science and statistics. It is increasingly recognized that the traditional model-based regression and classification strategy can be enhanced with new data mining devices. In this paper we intend to carry out a predictive study which involves new statistical modelling methodology coupled with state-of-the-art machine learning tools.

Often we face a large number of candidate models constructed from the available data information. One popular approach is model selection which aims to choose an optimal model to be used for making a prediction. Popular model selection strategies include subset selection (Akaike, 1974; Akaike, 1979; Findley, 1991; Breiman, 1995), regularization method (Fan and Li, 2001; Yuan and Lin, 2006; Zou, 2006), sure independence screening technique (Fan and Lv, 2008; Fan and Song, 2010; Barut et al., 2016) and dimension reduction (Ma and Zhu, 2013; Li et al., 2014; Guo et al., 2015). However, the selected model, though sharing good theoretical properties, could still miss useful information and consequently yield poor prediction performance when variables absent from the final selected model have significant influences on predicting the outcome.

Unlike model selection, model averaging is an alternative which combines predictions by a weighted average over all potential candidate models. Thus this strategy effectively deals with the uncertainty across different models and usually leads to a lower model misspecification risk. We have seen growing research interests in this field in recent years. Hansen (2007) utilized the Mallows criterion to choose the optimal model weights and demonstrated that their method could asymptotically achieve the lowest possible squared error. Wan et al. (2010) contributed an alternative proof for asymptotic optimality of

the Mallows model averaging estimation under a general non-nested set-up. Thus their work provided an even stronger theoretical justification for the use of the Mallows criterion. Liang et al. (2011) developed a new model weighting mechanism that involves minimizing the trace of an unbiased estimator of the model average estimation covariance matrix. Hansen and Racine (2012) developed a jackknife model averaging estimate with weights selected by minimizing a cross-validation criterion. Under high-dimensional regression setting, Ando and Li (2014) proposed a delete-one-out cross-validation model averaging estimation and demonstrated their proposal can achieve the lowest possible prediction loss asymptotically in theory. Zhang and Liu (2018) studied the asymptotic properties of nested least squares model averaging estimators including the Mallows model averaging estimator (Hansen, 2007) and the jackknife model averaging estimator (Hansen and Racine, 2012). Other related literature includes Hansen (2008), Liu (2015) and Gao et al. (2016) and references therein.

Almost all previous research findings are based on averaging a set of parametric models (e.g. linear regression models), which apparently limit the model space of all likely sub-model candidates. Although parametric models are widely used in many applications such as medicine, economics and sociology, they might produce misleading results when the unknown data structure deviates from assumed parameter forms. In contrast, semiparametric models with less structural restriction may provide improved prediction results with more accurate characterization of the relationship between the response and the predictors. Thus the semiparametric model averaging prediction (SMAP) methods are flexible and able to yield a lower model misspecification risk. Recently, Li et al. (2015) proposed a nonparametric model averaging procedure in which the multivariate mean regression function was approximated by an affine combination of one-dimensional marginal smooth functions. Chen et al. (2018a) further generalized the method of Li et al. (2015) and developed two marginal approaches for ultra-high dimensional nonlinear dynamic time series regression models. However, these works do not incorporate interaction

terms. The varying coefficient model is more useful to model complicated interaction effects (Hastie and Tibshirani, 1993; Fan and Zhang, 1999; Fan and Huang, 2005; Fan and Zhang, 2008). Recently, Li et al. (2018a) proposed a Mallows model averaging estimation for semiparametric varying coefficient models. Zhu et al. (2018) developed a model averaging estimation for semiparametric varying-coefficient partially linear model and proved that model weights selected by the Mallows distance criterion could lead to an asymptotically optimal model average estimation under certain regularity conditions. Li et al. (2018b) proposed varying-coefficient SMAP which outperforms the traditional varying coefficient model in terms of out-of-sample prediction. More discussion on recent development in SMAP can refer to Zhang and Liang (2011), Huang and Li (2018) and Zhang and Wang (2018).

The aforementioned model averaging approaches can only deal with the continuous response and may not be directly applied to the categorical response. Logistic regression models as an important class of generalized linear models (McCullagh and Nelder, 1989) are widely used to analyze binary data. When the response takes more than two discrete values, multinomial logistic regression models can be used for modeling the multi-category data (Venable and Ripey, 2002; Hosmer and Lemeshow, 2004; Li et al., 2013). Although some research results about model averaging for the discrete response were developed, they are mainly focused on parametric models (Wan et al., 2014; Zhang et al. 2016; Ando and Li, 2017; Chen et al., 2018b). Most existing works considered the asymptotic optimality for the estimate of model weights but casted little insight on the prediction performance. If only parametric model average results are adopted, they may be hardly comparable to the cutting-edge classifiers from machine learning and computer sciences. We thus consider averaging a set of varying-coefficient semiparametric models to classify the multiple categories.

In addition to the basic model construction, we further refine our classification procedure with the well-known adaptive boosting (AdaBoost) algorithm. Boosting is one of the off-the-shelf machine learning techniques, which iteratively evolves weak classifiers to a strong panel. Originally proposed in

the computational learning field, it has now received much attention over the last two decades (Freund and Schapire, 1997; Schapire, 1997; Schapire and Singer, 1999). Particularly, Friedman et al. (2000) developed a LogitBoost for classification through combining additive models with exponential loss. Domingo and Watanabe (2000) modified the weighting system of AdaBoost and developed MadaBoost to reduce the estimation noise. For multi-category classification, Zhu et al. (2009) extended the AdaBoost algorithm to the multi-class problem without simplifying the multi-class problem to multiple two-class problems and showed that their proposal achieves lower misclassification rate than the traditional AdaBoost algorithm. Li and Bradic (2018) utilized nonconvex loss functions to design ArchBoost and adaptive robust boosting, and they also considered breakdown point analysis and influence function analysis to prove the robustness of their proposals. Other related literature on boosting can refer to Freund (1995), Freund (2001a), Friedman (2001b) and Bühlmann and Yu (2006). In this article, we consider a similar boosting framework as Zhu et al. (2009) and implement it in varying coefficient SMAP for further enhancing the prediction performance.

The methodology development in this paper was motivated by an automobile classification problem. Accurate classification for automobiles is very important in industry applications such as surveillance, security framework, traffic congestion prevention and accidents avoidance. As the number of automobiles on the road is increasing tremendously in modern society, vehicle classification is not only regarded as one of the essential parts in intelligent traffic system but also plays an irreplaceable role for transportation planning, facility design, and operations. Operating agencies typically employ the vehicle classification technique to monitor heavy vehicle usage such as restricting the number of heavy vehicles on highway bridges, which prolongs lifespan of the highway bridge. In addition, with the advancement of traffic surveillance, visual vehicle images can be easily collected to measure all kinds of numerical characteristics of vehicles according to many different shooting angles. So far, machine learning approaches including adaptive boosting (Rahim et al., 2013) and support vector machine (Sun and Ban,

2013) are used to classify vehicle items into appropriate categories based on the observed appearance features of vehicles. In this paper, we focus on the task of vehicle classification and consider to apply our proposed SMAP approach to predict the class membership of observed vehicles.

Compared with existing works, this paper generates the following research merits. Firstly, it is usually a challenging task to select a suitable index variable when one fits the varying coefficient model. In fact, any continuous covariate variable can be taken as the index variable. Our proposed model averaging procedure effectively overcome this problem as we average multiple varying coefficient sub-models with different index variables, and the model weights automatically adjust the relative importance of these sub-models and naturally accommodate the complicated interaction effects. Thus we avoid the criticism of artificially choosing the index variables in varying-coefficient studies. Secondly, our proposed method is more robust against model mis-specification and works much more satisfactorily than traditional varying coefficient multinomial logit models. The SMAP can be applied without assuming any true model form which is harder and harder to postulate in this big data era. Basically, *all assumed models are wrong*. But aggregating candidate models with a weighted average effectively provides a close approximation to the reality and offers great potentials for future prediction. Thirdly, published research findings of improving model average via the boosting algorithm are rather limited. As far as we reviewed, no work yet applied the AdaBoost algorithm to the SMAP setting. Therefore, another fold of contribution of this article is an investigation of the augmented forecasting performance of the model averaging approach for categorical data.

The rest of the paper is organized as follows. In Section 2, the varying coefficient SMAP approach is introduced where we consider the varying coefficient multinomial logistic models as candidate models, and we further incorporate the AdaBoost algorithm in the prediction. In Section 3, extensive simulation studies are conducted to evaluate the proposed methods and compare with existing competitive approaches. In Section 4, a real automobile data set is analyzed to further illustrate our proposed methodology. Section 5

ends with conclusion and discussion. Finally, the proof of Theorem is given in the Appendix.

2 Methodology

2.1 Varying coefficient multinomial logit model

Let Y be a categorical response variable with J levels, taking value from the discrete set $\{1,2,...,J\}$ and $\boldsymbol{X}=\left(X_1,...,X_p\right)^T$ be a p-dimensional covariate. Suppose that we have a set of independent and identically distributed samples $\{\boldsymbol{X}_i,Y_i\}_{i=1}^n$ from $\{\boldsymbol{X},Y\}$, where $\boldsymbol{X}_i=\left(X_{i1},...,X_{ip}\right)^T$, $Y_i\in\{1,2,...,J\}$ and $Y_i = r$ means that the ith individual is from the rth category for $r\in\{1,2,...,J\}$. Furthermore, let $p_{ij}=\Pr\{Y_i=j\mid\boldsymbol{X}_i\}$ be the conditional probability that the ith individual belongs to the jth category. Without loss of generality, we often assume that the response categories are mutually exclusive and exhaustive. This means that the probabilities add up to one for each individual, i.e. $\sum_{j=1}^J p_{ij}=1$ for each i. In practice, a usual model-based predicted value of Y_i is $\hat{Y}_i=\text{argmax}_{r=1,\cdots,J}\{\hat{p}_{ir}\}$, where $\hat{p}_{ij}$ is an estimate of p_{ij} evaluated from a fitted statistical model (classifier).

A reliable prediction of Y boils down to a reliable modelling of the conditional probability p_{ij}. The multinomial logistic model (MLM) is widely adopted for this purpose and introduced in elementary texts (McCullagh and Nelder (1989), Venable and Ripley (2002), Hosmer and Lemeshow (2004) and Chen et al. (2018b)). Although this model is easy to understand and widely accepted by many practitioners, its linear form may invite criticism for being too restrictive. Many nonparametric multi-category classifiers have been proposed over the years and may improve the performance of logistic regression. In this article, we focus on the varying coefficient multinomial logistic model (VCMLM).

Specifically, we take the last category J as a baseline reference and set the log-odds of category j relative to the baseline be a varying-coefficient model

$$\mu_{ij} = \log \frac{p_{ij}}{p_{iJ}} = \xi_j\left(X_{it}\right) + \boldsymbol{X}_{i,\,t}^{T}\boldsymbol{\theta}_j\left(X_{it}\right), i = 1,...,n,\ j = 1,...,J-1, \qquad (1)$$

where $\xi_j(\cdot)$ is an univariate function, $\boldsymbol{\theta}_j(\cdot) = \left(\theta_{j1}(\cdot),...,\theta_{j,p-1}(\cdot)\right)^T$ is a $(p-1)$ dimensional function vector and $\boldsymbol{X}_{i,\,t}$ is a $(p-1)$ dimensional covariate excluding the tth predictor X_{it}. In fact, any continuous elements of $\boldsymbol{X}$ can be chosen as the index variable and some practical recommendations are often made to select an index variable. For the simplicity of presentation, we choose the tth predictor X_{it} as the index variable in this section. More general treatment on this issue will be given in the next subsection. Clearly model (1) may produce more flexible prediction than MLM and other additive nonparametric models as it allows nonlinear interaction effects between X_{it} and the other $(p-1)$ variables $\boldsymbol{X}_{i,\,t}$. Following this model, the conditional probability of $Y_i = j$ given $\boldsymbol{X}_i$ can be written as

$$\begin{cases} p_{ij} = \Pr\left\{Y_i = j \mid \boldsymbol{X}_i\right\} = \dfrac{\exp\left(\mu_{ij}\right)}{1 + \sum_{l=1}^{J-1}\exp\left(\mu_{il}\right)}, i = 1,...,n,\ j = 1,...,J-1, \\ p_{iJ} = \Pr\left\{Y_i = J \mid \boldsymbol{X}_i\right\} = \dfrac{1}{1 + \sum_{l=1}^{J-1}\exp\left(\mu_{il}\right)}, i = 1,...,n. \end{cases} \qquad (2)$$

To estimate the unknown functions, we apply the B-splines basis approximation method (de Boor (2001)). Specifically, $\xi_j(\cdot)$ and $\boldsymbol{\theta}_j(\cdot)$ can be approximated respectively by

$$\xi_j(\cdot) \approx \boldsymbol{B}(\cdot)^T\boldsymbol{\lambda}_j, \theta_{jk}(\cdot) \approx \boldsymbol{B}(\cdot)^T\boldsymbol{\varsigma}_{jk},\ j = 1,...,J-1, k = 1,...,p-1, \qquad (3)$$

where $\boldsymbol{B}(u) = \left(B_s(u) : 1 \le s \le K_n\right)^T$ is a K_n – vector of B-spline basis functions of order q ($q \ge 2$), $K_n = N_n + q$, N_n is the number of interior knots, $\boldsymbol{\lambda}_j = \left(\lambda_{js} : 1 \le s \le K_n\right)^T$ and $\boldsymbol{\varsigma}_{jk} = \left(\varsigma_{jks} : 1 \le s \le K_n\right)^T$ are unknown loading vectors. Let $\boldsymbol{\gamma}_j = \left(\boldsymbol{\lambda}_j^T, \boldsymbol{\varsigma}_{j1}^T,...,\boldsymbol{\varsigma}_{jp-1}^T\right)^T$ and

$$\boldsymbol{\Pi}_i \square \left(\boldsymbol{B}(X_{it})^T, X_{i1}\boldsymbol{B}(X_{it})^T,..., X_{i,t-1}\boldsymbol{B}(X_{it})^T, X_{i,t+1}\boldsymbol{B}(X_{it})^T,..., X_{ip}\boldsymbol{B}(X_{it})^T\right)^T.$$

Then VCMLM can thus be approximated by MLM and rewritten as

$$\mu_{ij} = \log \frac{p_{ij}}{p_{iJ}} \approx \boldsymbol{\Pi}_i^T \boldsymbol{\gamma}_j, i = 1,..., n;\ j = 1,..., J-1. \quad (4)$$

We can now estimate the unknown parameter $\boldsymbol{\gamma} = \left(\boldsymbol{\gamma}_1^T,..., \boldsymbol{\gamma}_{J-1}^T\right)^T$ in (4) through the maximum likelihood approach. The estimator $\hat{\boldsymbol{\gamma}} = \left(\hat{\boldsymbol{\gamma}}_1^T,..., \hat{\boldsymbol{\gamma}}_{J-1}^T\right)^T$ of $\boldsymbol{\gamma}$ in model (4) can be obtained by maximizing the following log-likelihood function

$$l(\boldsymbol{\gamma}) = \sum_{i=1}^{n} \left\{ \sum_{j=1}^{J-1} \mathrm{I}\left(Y_i = j\right) \boldsymbol{\Pi}_i^T \boldsymbol{\gamma}_j - \log\left(1 + \sum_{l=1}^{J-1} \exp\left(\boldsymbol{\Pi}_i^T \boldsymbol{\gamma}_l\right)\right) \right\}, \quad (5)$$

where $\mathrm{I}(A)$ is an indicator function for event *A*. Although there is no explicit form of $\hat{\boldsymbol{\gamma}}$, maximizing (5) with respect to $\boldsymbol{\gamma}$ can be implemented by the familiar Newton-Raphson algorithm or other nonlinear solver. There are also quite a few packages in R for this purpose. We employ the "multinom" function in the package "nnet" in this paper.

Let $\boldsymbol{x}_0 = \left(\mathrm{x}_{01},..., \mathrm{x}_{0p}\right)^T$ be the covariates for a new subject with an unknown response Y_0. Then the conditional probability $p_{0j} = \Pr\left\{Y = j \mid \boldsymbol{x}_0\right\}$ can be estimated by

$$\begin{cases} \hat{p}_{0j} = \exp\left(\hat{\mu}_{0j}\right) / \left(1 + \sum_{l=1}^{J-1} \exp\left(\hat{\mu}_{0l}\right)\right), j = 1,..., J-1, \\ \hat{p}_{0J} = 1 / \left(1 + \sum_{l=1}^{J-1} \exp\left(\hat{\mu}_{0l}\right)\right), \end{cases}$$

where $\hat{\mu}_{0j} = \boldsymbol{\Pi}_0^T \hat{\boldsymbol{\gamma}}_j$ and

$$\boldsymbol{\Pi}_0 \equiv \boldsymbol{\Pi}\left(\boldsymbol{x}_0\right) = \left(\boldsymbol{B}(\mathrm{x}_{0t})^T, \mathrm{x}_{01}\boldsymbol{B}(\mathrm{x}_{0t})^T,..., \mathrm{x}_{0,t-1}\boldsymbol{B}(\mathrm{x}_{0t})^T, \mathrm{x}_{0,t+1}\boldsymbol{B}(\mathrm{x}_{0t})^T,..., \mathrm{x}_{0p}\boldsymbol{B}(\mathrm{x}_{0t})^T\right)^T.$$

We can forecast the response Y_0 by $\hat{\mathrm{Y}}_0 = \mathrm{argmax}_{j=1,\cdots,J}\left\{\hat{p}_{0j}\right\}$.

In practice we do not always know the true model. VCMLM is more sophisticated than MLM and support vector machine, but still may not be the true model. Therefore we are often faced with a large number of candidate

models and not sure which model should be adopted. Using a single model to predict the future outcome may therefore be risky if the assumed model form differs remarkably from the true model. An effective approach to avoid model mis-specification is the model averaging method which combines more useful information attained from different candidate models. The resulting model averaging prediction may suffer less model specification bias. We consider a VCMLM on the basis of the semiparametric model averaging prediction (SMAP) in the following. Our procedure also naturally avoid the problem of selecting index variables in varying coefficient functions.

To be more specific, we consider a sequence of approximating candidate models $\mathrm{M}_1,\cdots,\mathrm{M}_p$, where the *s*th candidate model is a VCMLM and can be approximated by MLM based on B-spline basis functions:

$$\begin{aligned}\mathrm{M}_s : \mu_{ij}^{(s)} &= \log\frac{p_{ij}^{(s)}}{p_{iJ}^{(s)}} = \xi_j^{(s)}\left(X_{is}\right) + \boldsymbol{X}_{i,\,s}^{T}\boldsymbol{\theta}_j^{(s)}\left(X_{is}\right) \\ &\approx \boldsymbol{\Pi}_i^{(s)T}\boldsymbol{\gamma}_j^{(s)}, i=1,...,n,\ j=1,...,J-1, s=1,...,p,\end{aligned} \tag{6}$$

where $\boldsymbol{X}_{i,\,s}$ is a $(p-1)$ dimensional covariate by removing the *s*th predictor X_{is}, $\boldsymbol{\gamma}_j^{(s)} = \left(\boldsymbol{\lambda}_j^{(s)T}, \boldsymbol{\varsigma}_{j1}^{(s)T},...,\boldsymbol{\varsigma}_{jp-1}^{(s)T}\right)^T$ is the unknown spline coefficient vector of the *s*th candidate model and

$$\boldsymbol{\Pi}_i^{(s)} \square \left(\boldsymbol{B}(X_{is})^T, X_{i1}\boldsymbol{B}(X_{is})^T,...,X_{i,s-1}\boldsymbol{B}(X_{is})^T, X_{i,s+1}\boldsymbol{B}(X_{is})^T,...,X_{ip}\boldsymbol{B}(X_{is})^T\right)^T.$$

Let $\boldsymbol{\gamma}^{(s)} = \left(\boldsymbol{\gamma}_1^{(s)T},...,\boldsymbol{\gamma}_{J-1}^{(s)T}\right)^T$. The maximum likelihood estimator $\hat{\boldsymbol{\gamma}}^{(s)}$ of $\boldsymbol{\gamma}^{(s)}$ under the *s*th candidate model can be obtained by maximizing the following objective function

$$l_s(\boldsymbol{\gamma}^{(s)}) = \sum_{i=1}^{n}\left\{\sum_{j=1}^{J-1}\mathrm{I}(Y_i = j)\boldsymbol{\Pi}_i^{(s)T}\boldsymbol{\gamma}_j^{(s)} - \log\left(1+\sum_{l=1}^{J-1}\exp\left(\boldsymbol{\Pi}_i^{(s)T}\boldsymbol{\gamma}_l^{(s)}\right)\right)\right\}.$$

Then, the conditional probability $p_{ij}^{(s)} = \Pr\left\{Y = j \mid \boldsymbol{X}_i\right\}$ under the *s*th candidate model can be estimated by

$$\begin{cases} \hat{p}_{ij}^{(s)} = \exp\left(\hat{\mu}_{ij}^{(s)}\right) / \left(1 + \sum_{l=1}^{J-1} \exp\left(\hat{\mu}_{il}^{(s)}\right)\right), i = 1, ..., n, \; j = 1, ..., J-1, \\ \hat{p}_{iJ}^{(s)} = 1 / \left(1 + \sum_{l=1}^{J-1} \exp\left(\hat{\mu}_{il}^{(s)}\right)\right), i = 1, ..., n, \end{cases} \quad (7)$$

where $\hat{\mu}_{ij}^{(s)} = \boldsymbol{\Pi}_i^{(s)T} \hat{\boldsymbol{\gamma}}_j^{(s)}$. Furthermore, let $\boldsymbol{w} = \left(w_1, ..., w_p\right)^T$ be the weight vector that belongs to the following unit hypercube

$$\mathrm{W} = \left\{ \boldsymbol{w} \in [0,1]^p : \sum_{s=1}^{p} w_s = 1 \right\}.$$

Then, the model averaging estimator of the conditional probability $p_{ij} = \Pr\left\{Y = j \mid \boldsymbol{X}_i\right\}$ is given by

$$\hat{p}_{ij}^{w} = \sum_{s=1}^{p} w_s \hat{p}_{ij}^{(s)}, i = 1, ..., n, \; j = 1, ..., J.$$

The weight vector $\boldsymbol{w}$ plays a central role in producing good and reliable prediction performance. We consider the following least squares criterion to determine the weights

$$\Delta\left(\boldsymbol{w}\right) = \sum_{i=1}^{n} \sum_{j=1}^{J} \left(\sum_{s=1}^{p} w_s \hat{p}_{ij}^{(s)} - \mathrm{I}\left(Y_i = j\right) \right)^2, \quad (8)$$

where $\hat{p}_{ij}^{(s)}, j = 1, ..., J$ are defined in (7). Then the optimal weight vector is obtained by choosing $\boldsymbol{w} \in \mathrm{W}$ to minimize $\Delta\left(\boldsymbol{w}\right)$, i.e., $\boldsymbol{w} = \arg\min_{\boldsymbol{w} \in \mathrm{W}} \Delta\left(\boldsymbol{w}\right)$.

We now establish the asymptotic optimality of the estimated weight vector. Let $\boldsymbol{\gamma}^{(s)*} = \left(\boldsymbol{\gamma}_1^{(s)*T}, ..., \boldsymbol{\gamma}_{J-1}^{(s)*T}\right)^T = \mathrm{argmin}_{\boldsymbol{\gamma}^{(s)}} E\left\{-l_s(\boldsymbol{\gamma}^{(s)})\right\}$. In the following presentation, let C be a generic constant, and $\lambda_{\min}(\boldsymbol{A})$ and $\lambda_{\max}(\boldsymbol{A})$ be the minimum and maximum singular values of a real matrix $\boldsymbol{A}$ respectively. Denote $\mathbf{P} = \left(\mathrm{P}_1^T, ..., \mathrm{P}_n^T\right)^T$ with $\mathbf{P}_i = \left(\mathbf{P}_{i1}, ..., \mathbf{P}_{iJ}\right)^T$ and $\mathbf{P}_{ij} = \left(\hat{p}_{ij}^{(1)}, ..., \hat{p}_{ij}^{(p)}\right)^T$ for $i = 1, ..., n, \; j = 1, ..., J$, $\boldsymbol{P} = \left(\boldsymbol{P}_1^T, ..., \boldsymbol{P}_n^T\right)^T$ with $\boldsymbol{P}_i = \left(p_{i1}, ..., p_{iJ}\right)^T$, and define the risk function as $R(\boldsymbol{w}) = E\left\|\boldsymbol{P} - \mathbf{P}\,\boldsymbol{w}\right\|^2$. The optimal weight vector is $\boldsymbol{w}^0 = \mathrm{argmin}_{\boldsymbol{w} \in \mathrm{W}} R(\boldsymbol{w})$. Write $\psi_n = \min_{\boldsymbol{w} \in \mathrm{W}} R(\boldsymbol{w})$. We let $\boldsymbol{w}^0$ be an interior point of W.

Our development of an asymptotic theory of the optimal weight vector requires the following technical conditions:

Condition 1. $\lambda_{max}\left\{\left(\frac{1}{n}\frac{\partial^2 l_s(\boldsymbol{\gamma}^{(s)})}{\partial\boldsymbol{\gamma}^{(s)}\partial\boldsymbol{\gamma}^{(s)T}}\right)^{-1}\right\}=O_p(1)$ *uniformly for* $s(1\le s\le p)$ *and* $\boldsymbol{\gamma}^{(s)}$ *lies between* $\hat{\boldsymbol{\gamma}}^{(s)}$ *and* $\boldsymbol{\gamma}^{(s)*}$.

Condition 2. $E\left|\Pi_{ik}^{(s)}\right|^2<C$ *uniformly in* $s(1\le s\le p)$ *and* $k(1\le k\le pK_n)$, *where* $\Pi_{ik}^{(s)}$ *is the kth element of* $\boldsymbol{\Pi}_i^{(s)}$.

Condition 3. $Pr\left(\lambda_{min}\left(\mathbf{P}^T\mathbf{P}/n\right)>C>0\right)$ *tends to 1, and* $\lambda_{max}\left(\mathbf{P}^T\mathbf{P}/n\right)=O_p(p)$.

Condition 4. $\psi_n^{1/2}/(K_n p^{3/2}n^{\delta})=o(1)$ *for some positive constant δ.*

Condition 1 restricts the minimum singular value of the second derivative matrix of $l_s(\boldsymbol{\gamma}^{(s)})$. This is similar to Condition C.4 of Zhang et al. (2016). Condition 2 is a mild condition that constrains the moment of $\Pi_{ik}^{(s)}$. Condition 3, which is commonly used in the literature (Fan and Peng, 2004; Ravikumar et al., 2009). Condition 4 provides the relationship between $\{\psi_n, K_n, p, n\}$, where ψ_n is allowed to increase at the rate that does not exceed $K_n^2 p^3 n^{2\delta}$.

Theorem 1. *Let Conditions 1 - 4 be satisfied. Then there exists a local minimiser* $\boldsymbol{w}$ *of* $\square(\boldsymbol{w})$ *such that*

$$\|\boldsymbol{w}-\boldsymbol{w}^0\|=O_p\left(K_n p^2 n^{-1/2+\delta}\right), \quad (9)$$

where δ is the positive constant given under Condition 4.

Theorem 1 indicates that the weight estimator $\boldsymbol{w}$ approaches the optimal weight $\boldsymbol{w}^0$ at the rate of $K_n p^2 n^{-1/2+\delta}$. For the case where p is fixed, and ψ_n and K_n have the orders of $n^{1-\alpha}(0<\alpha\le 1)$ and $n^{\beta}(\beta\ge 0)$ respectively, $\|\boldsymbol{w}-\boldsymbol{w}^0\|\xrightarrow{p}0$ if $1/2-\alpha/2-\delta<\beta<1/2-\delta$.

The proof of the theorem is given in the Appendix. The convergence rate in Theorem 1 may not be an optimal rate for estimating the weights. One can

impose stronger technical conditions to improve the rates. For example, we can use similar arguments as in the Appendix to derive a relatively faster rate by assuming that

$$\lambda_{max}\left(1/n\sum_{i=1}^{n}\mathbf{\Pi}_i^{(s)}\mathbf{\Pi}_i^{(s)T}\right)=O_p(1) \qquad (10)$$

uniformly in *s*. This condition is actually quite strong because the dimension of $\mathbf{\Pi}_i^{(s)}$ diverges. Similar conditions are adopted in Fan and Peng (2004) and Bickel and Levina (2008).

Remark 1. In the *s*th candidate model M_s, we use the *s*th component X_{is} as the index variable and its nonlinear interaction with all other covariates can be incorporated in this model. If some components are discrete such as gender or ethnicity, then the varying coefficients for these terms are automatically regarded as linear functions in our implementation.

Remark 2. Non-negative constraints on w_j, $j=1,...,p$ are required in the proposed procedure since $\boldsymbol{w}$ serves as the model weights. On the other hand, we should ensure $\sum_{j=1}^{J}\hat{p}_{0j}^{w}=1$ that requires $\sum_{s=1}^{p}w_s=1$. These requirements lead to $\boldsymbol{w}\in\mathrm{W}$.

Remark 3. The objective function $\square(\boldsymbol{w})$ has the following quadratic matrix form

$$\square(\boldsymbol{w})=\left(\mathbf{Y}-\mathbf{P}\boldsymbol{w}\right)^T\left(\mathbf{Y}-\mathbf{P}\boldsymbol{w}\right),$$

where $\mathbf{Y}=\left(\mathbf{Y}_1^T,...,\mathbf{Y}_n^T\right)^T$, with $\mathbf{Y}_i=\left(\mathrm{Y}_{i1},...,\mathrm{Y}_{iJ}\right)^T$ and $\mathrm{Y}_{ij}=\mathrm{I}\left(Y_i=j\right)$ for $i=1,...,n$ and $j=1,...,J$. Thus we may obtain a closed-form solution of the weight vector $\boldsymbol{w}$ directly. However, the resulting estimator does not satisfy $\boldsymbol{w}\in\mathrm{W}$. In order to ensure these constraints on parameter estimation, we have to adopt a quadratic programming technique. In this paper we use the “solve.QP” function in the R package “quadprog” developed by Berwin Turlach and Andreas Weingessel.

Then, for a future observation $\boldsymbol{x}_0 = \left(\boldsymbol{x}_{01},..., \boldsymbol{x}_{0p}\right)^T$, the conditional probability $p_{0j}^{(s)} = \Pr\left\{Y = j \mid \boldsymbol{x}_0\right\}$ under the *s*th candidate model can be obtained by

$$\begin{cases} \hat{p}_{0j}^{(s)} = \exp\left(\hat{\mu}_{0j}^{(s)}\right) / \left(1 + \sum_{l=1}^{J-1} \exp\left(\hat{\mu}_{0l}^{(s)}\right)\right), j = 1,., J-1, \\ \hat{p}_{0J}^{(s)} = 1 / \left(1 + \sum_{l=1}^{J-1} \exp\left(\hat{\mu}_{0l}^{(s)}\right)\right), \end{cases}$$

where $\boldsymbol{\Pi}_0^{(s)} = \left(\boldsymbol{B}(\mathrm{x}_{0s})^T, \mathrm{x}_{01}\boldsymbol{B}(\mathrm{x}_{0s})^T,..., \mathrm{x}_{0,s-1}\boldsymbol{B}(\mathrm{x}_{0s})^T, \mathrm{x}_{0,s+1}\boldsymbol{B}(\mathrm{x}_{0s})^T,..., \mathrm{x}_{0p}\boldsymbol{B}(\mathrm{x}_{0s})^T\right)^T$ and $\hat{\mu}_{0j}^{(s)} = \boldsymbol{\Pi}_0^{(s)T} \hat{\boldsymbol{\gamma}}_j^{(s)}$. Based on the estimated $\hat{p}_{0j}^{(s)}$ and $\boldsymbol{w}$, one can predict the future response Y_0 by $\hat{\mathrm{Y}}_0 = \mathrm{argmax}_{j=1,\cdots,J}\left\{\hat{p}_{0j}^{w}\right\}$ with $\hat{p}_{0j}^{w} = \sum_{s=1}^{p} \hat{w}_s \hat{p}_{0j}^{(s)}$ for $j = 1,..., J$.

Remark 4. For the sake of comparison, we also consider the adaptive varying coefficient model (Fan et al. (2003)) which has the following structure

$$E\left(Y \mid \boldsymbol{X} = \boldsymbol{x}\right) = G\left(\boldsymbol{x}\right) = g_0\left(\boldsymbol{x}^T\boldsymbol{\beta}\right) + \sum_{j=1}^{p} g_j\left(\boldsymbol{x}^T\boldsymbol{\beta}\right) x_j$$

where the index coefficient $\boldsymbol{\beta} = \left(\beta_1,..., \beta_p\right)^T \in \mathbb{R}^p$ satisfies $\|\boldsymbol{\beta}\| = 1$ and $\beta_r > 0$ with $r \in \{1,..., p\}$ and $g_0(\cdot),..., g_p(\cdot)$ are unknown functions. Many popular approaches have been developed to estimate $\boldsymbol{\beta}$ and $g_j(\cdot)$ (e.g., (Powell et al. (1989), Xia et al. (2002), Huang et al. (2004), Fan and Zhang (2008) and Cui et al. (2011)). However, the above model only deals with continuous response. Based on the idea of Ke et al. (2016) and our earlier notations, an adaptive varying coefficient multinomial logistic model (AVCMLM) can be constructed by

$$\mu_{ij} = \log\frac{p_{ij}}{p_{iJ}} = g_{0j}\left(\boldsymbol{X}_i^T\boldsymbol{\beta}\right) + \sum_{l=1}^{p} g_{lj}\left(\boldsymbol{X}_i^T\boldsymbol{\beta}\right) X_{il}, i = 1,..., n, j = 1,..., J-1,$$

where $\boldsymbol{X}_i$ is a covariate vector and $\boldsymbol{g}_j(\cdot) = \left(g_{0j}(\cdot),..., g_{pj}(\cdot)\right)^T$ is a vector of unknown functions. Here we can also utilize B-splines to approximate nonparametric functions and apply existing methods to make a prediction for the response.

For an adequate comparison, we implement the AVCMLM in simulation studies and empirical application. Observing from such numerical results, we briefly summarize the merits of our model averaging approaches over the AVCMLM. On the one hand, it is easy to see that the AVCMLM has more complicated model structure than our concerned varying coefficient multinomial logistic model (VCMLM). Thus, it is more difficult to obtain efficient estimators $\boldsymbol{\beta}$ and $\boldsymbol{g}_j(\cdot)$ in the AVCMLM for a moderate sample size (e.g., n = 200 or 400), hence leading to unsatisfactory prediction performance. On the other hand, the AVCMLM focuses on using only one semi-parametric model to forecast the response, which may still suffer the model mis-specification biases. In contrast, our proposed approaches aim to optimally combine a series of VCMLMs to make a prediction, which effectively reduces reliance on a particular model and results in relatively better out-of-sample prediction performance.

2.2 AdaBoost semiparametric model averaging

Boosting is one of the most powerful learning tools and widely used for improving the classification performance by taking a weighted majority vote of weak classifiers. To enhance the practical performance of SMAP proposed in the preceding section, we will incorporate a boosting algorithm in the prediction. Specifically we apply the adaptive boosting (AdaBoost) algorithm to improve the estimation accuracy of $\boldsymbol{w}$ and $\hat{p}_{0j}^{(s)}$, and consequently achieve more accurate model averaging estimate $\hat{p}_{0j}^{\boldsymbol{w}}$.

First, we propose a new multi-class AdaBoost algorithm to improve the estimation of $p_{ij}^{(s)}$ and $p_{0j}^{(s)}$ in the sth candidate model M_s with $s = 1,..., p$. For the sth candidate model, let $\tilde{\boldsymbol{\gamma}}_m^{(s)} = \left(\tilde{\gamma}_{1,m}^{(s)T},..., \tilde{\gamma}_{J-1,m}^{(s)T}\right)^T$ be the weighted maximum likelihood estimate by maximizing the following weighted conditional log-likelihood function

$$\begin{aligned}\tilde{\boldsymbol{\gamma}}_m^{(s)} &= \arg\max_{\boldsymbol{\gamma}^{(s)}} l_{\boldsymbol{\tau}}\left(\boldsymbol{\gamma}^{(s)}\right) \\ &\Leftrightarrow \arg\max_{\boldsymbol{\gamma}^{(s)}} \sum_{i=1}^{n} \tau_i^{(m-1)} \left\{ \sum_{j=1}^{J-1} \mathrm{I}\left(Y_i = j\right) \boldsymbol{\Pi}_i^{(s)T} \boldsymbol{\gamma}_j^{(s)} - \log\left(1 + \sum_{l=1}^{J-1} \exp\left(\boldsymbol{\Pi}_i^{(s)T} \boldsymbol{\gamma}_l^{(s)}\right)\right)\right\},\end{aligned} \tag{11}$$

where the weight vector $\boldsymbol{\tau}^{(m-1)} = \left(\tau_1^{(m-1)}, ..., \tau_n^{(m-1)}\right)^T$ ($m = 1, ..., M$) is updated according to Algorithm 1:

Algorithm 1 AdaBoost algorithm for estimating $p_{ij}^{(s)}$ and $p_{0j}^{(s)}$ ($j = 1, ..., J$) for the *s*th candidate model.

---

1. Initialize the equal weights $\tau_i^{(0)} = 1/n, i = 1, ..., n$.

2. For *m* = 1 to *M*:

(a) Based on the training data, we obtain a prediction value of Y_i by $\tilde{Y}_i^{(m)} = \text{argmax}_{j=1,\cdots,J}\left\{\tilde{p}_{ij,m}^{(s)}\right\}$ with $\tilde{p}_{ij,m}^{(s)} = \exp\left(\tilde{\mu}_{ij,m}^{(s)}\right) / \left(1 + \sum_{l=1}^{J-1} \exp\left(\tilde{\mu}_{il,m}^{(s)}\right)\right)$, $\tilde{p}_{iJ,m}^{(s)} = 1 / \left(1 + \sum_{l=1}^{J-1} \exp\left(\tilde{\mu}_{il,m}^{(s)}\right)\right)$, $\tilde{\mu}_{ij,m}^{(s)} = \boldsymbol{\Pi}_i^{(s)T} \tilde{\boldsymbol{\gamma}}_{j,m}^{(s)}$ and $\tilde{\boldsymbol{\gamma}}_{j,m}^{(s)}$ is obtained from (11) for $i = 1, ..., n$ and $j = 1, ..., J-1$.

(b) Compute $\varepsilon^{(m)} = \sum_{i=1}^{n} \tau_i^{(m-1)} \mathrm{I}\left(Y_i \neq \tilde{Y}_i^{(m)}\right) / \sum_{i=1}^{n} \tau_i^{(m-1)}$.

(c) Compute $\nu^{(m)} = \log\left(\frac{1-\varepsilon^{(m)}}{\varepsilon^{(m)}}\right) + \log(J-1)$.

(d) Set $\tau_i^{(m)} \leftarrow \tau_i^{(m-1)} \cdot \exp\left(\nu^{(m)} \mathrm{I}\left(Y_i \neq \tilde{Y}_i^{(m)}\right)\right)$ for $i = 1, ..., n$.

(e) Renormalize $\boldsymbol{\tau}^{(m)} = \left(\tau_1^{(m)}, ..., \tau_n^{(m)}\right)^T$ such that $\sum_{i=1}^{n} \tau_i^{(m)} = 1$ and $\tau_i^{(m)} \geq 0$.

3. Renormalize $\boldsymbol{\nu} = \left(\nu^{(1)}, ..., \nu^{(M)}\right)^T$ such that $\sum_{m=1}^{M} \nu^{(m)} = 1$ and $\nu^{(m)} \geq 0$ and output $\bar{p}_{ij}^{(s)} = \sum_{m=1}^{M} \nu^{(m)} \tilde{p}_{ij,m}^{(s)}$. Similarly for out-of-sample prediction we have $\bar{p}_{0j}^{(s)} = \sum_{m=1}^{M} \nu^{(m)} \tilde{p}_{0j,m}^{(s)}$, where $\tilde{p}_{0j,m}^{(s)}$ is obtained similarly by replacing $\boldsymbol{\Pi}_i^{(s)}$ with $\boldsymbol{\Pi}_0^{(s)}$.

Algorithm 2 AdaBoost algorithm for estimating the weight vector $\boldsymbol{w}$.

---

1. Initialize $\delta_i^{(0)} = 1/n, i = 1, ..., n$.

2. For *m* = 1 to *M*:

(a) Using the outputs from Algorithm 1, we define $\overline{\mathrm{P}} = \left(\overline{\mathrm{P}}_1^T, ..., \overline{\mathrm{P}}_n^T\right)^T$, $\overline{\mathrm{P}}_i = \left(\overline{\mathrm{P}}_{i1}, ..., \overline{\mathrm{P}}_{iJ}\right)^T$, $\overline{\mathrm{P}}_{ij} = \left(\bar{p}_{ij}^{(1)}, ..., \bar{p}_{ij}^{(p)}\right)^T$, $\mathbf{W} = diag\left\{\mathbf{W}_1, ..., \mathbf{W}_n\right\}$

and $\mathbf{W}_i = \delta_i^{(m-1)}\mathbf{I}_J$ with $\mathbf{I}_J$ being a $J \times J$ unit matrix. Let $\square_1(\boldsymbol{w}) = \sum_{i=1}^{n}\delta_i^{(m-1)}\sum_{j=1}^{J}\left(\sum_{s=1}^{p} w_s \bar{p}_{ij}^{(s)} - \mathrm{I}(Y_i = j)\right)^2 = \left(\mathbf{Y} - \overline{\mathbf{P}}\boldsymbol{w}\right)^T \mathbf{W}\left(\mathbf{Y} - \overline{\mathbf{P}}\boldsymbol{w}\right)$. Denote by $\overline{\boldsymbol{w}}^{(m)} = \arg\min_{\boldsymbol{w}\in\mathrm{W}} \square_1(\boldsymbol{w})$. Then we obtain a prediction value of Y_i by $\bar{Y}_i^{(m)} = \mathrm{argmax}_{j=1,\cdots,J}\left\{\bar{p}_{ij,m}^{\bar{w}}\right\}$ with $\bar{p}_{ij,m}^{\bar{w}} = \sum_{s=1}^{p}\bar{w}_s^{(m)}\bar{p}_{ij}^{(s)}$.

(b) Compute $\mathrm{D}^{(m)} = \sum_{i=1}^{n}\delta_i^{(m-1)}\mathrm{I}\left(Y_i \neq \bar{Y}_i^{(m)}\right) / \sum_{i=1}^{n}\delta_i^{(m-1)}$.

(c) Compute $\varsigma^{(m)} = \log\left(\frac{1-\mathrm{D}^{(m)}}{\mathrm{D}^{(m)}}\right) + \log(J-1)$.

(d) Set $\delta_i^{(m)} \leftarrow \delta_i^{(m-1)} \cdot \exp\left(\varsigma^{(m)}\mathrm{I}\left(Y_i \neq \bar{Y}_i^{(m)}\right)\right)$ for $i = 1,...,n$.

(e) Renormalize $\boldsymbol{\delta}^{(m)} = \left(\delta_1^{(m)},...,\delta_n^{(m)}\right)^T$ such that $\sum_{i=1}^{n}\delta_i^{(m)} = 1$.

3. Renormalize $\varsigma = \left(\varsigma^{(1)},...,\varsigma^{(M)}\right)^T$ such that $\sum_{m=1}^{M}\varsigma^{(m)} = 1$ and output $\overline{\boldsymbol{w}} \square \left(\bar{w}_1,...,\bar{w}_p\right)^T = \sum_{m=1}^{M}\varsigma^{(m)}\overline{\boldsymbol{w}}^{(m)}$.

In the above algorithm 1, individual weights of the observations are updated for the next iteration at step 2(*d*). It is easy to see that observations misclassified by the prediction procedure at the previous iteration will be allocated more weights through a factor $\exp\left(\nu^{(m)}\right)$ in the next iteration. In addition to improve the individual probability estimates, we may also use similar AdaBoost to improve the weight estimates for $\boldsymbol{w}$ in the model averaging prediction, which can be seen in Algorithm 2.

Finally, an AdaBoost model averaging prediction value of Y_0 can be obtained by $\bar{\mathrm{Y}}_0 = \mathrm{argmax}_{j=1,\cdots,J}\left\{\bar{p}_{0j}^{\bar{w}}\right\}$ with $\bar{p}_{0j}^{\bar{w}} = \sum_{s=1}^{p}\bar{w}_s\bar{p}_{0j}^{(s)}$ for $j = 1,...,J$, where $\bar{p}_{0j}^{(s)}$ and $\bar{w}_s$ are outputs given in Algorithms 1 and 2, respectively.

Remark 5. One usually needs to choose a sensible *M* value to stop the boosting iteration. Zhang and Yu (2005) advised to stop early in order to achieve consistency. Mease and Wyner (2008) argued via experiments that AdaBoost should be run for a longer time to guarantee convergence. Bennett (2008) pointed out that AdaBoost could converge faster with a larger

hypothesis space and a rapidly converged problem might enter an overtraining phase. Buhlmann and Yu (2008) also supported early stopping from their own experiments for estimating conditional class probabilities. In the following numerical analysis we have considered different M values and observed quite similar results when $M \geq 10$. We thus only present the findings by setting M = 10 for all the cases. We also suggest practitioners to investigate the convergence issue carefully for their applications.

3 Simulation studies

In this section, we investigate the finite sample performance of proposed model averaging procedures using simulations. We compare our proposed method with the following model averaging and classification strategies:

- (1) the random forest algorithm (Breiman et al., 2001; Biau, 2012; Scornet et al., 2015), denoted as ranfor, which can be implemented by the R function “randomForest” in the package “randomForest”;
- (2) the classification tree method (Breiman et al., 1984; Ripley, 1996), denoted as tree, which can be implemented by the R function “tree” in the package “tree”;
- (3) the support vector machines (Crammer and Singer, 2001; Lee et al., 2004) are an excellent tool for classification, denoted as svm, which can be implemented by the R function “ksvm” in the package “kernlab”;
- (4) the multi-class AdaBoost classification method proposed by Zhu et al. (2009), denoted as boost, which can be obtained by the R function “ boosting” in the package “adabag”.
- (5) the parametric multinomial logistic model (Venable and Ripley, 2002, Hosmer and Lemeshow, 2004), denoted as mlm, which can be implemented by the R function “multinom” in the package “nnet”;
- (6) the model averaging approach for parametric multinomial logistic model proposed by Chen et al. (2018b), denoted as mamlm;
- (7) the varying coefficient multinomial logistic model (VCMLM), where we use X_1 as the index variable, denoted as vcmlm;

- (8) the adaptive varying coefficient multinomial logistic model (AVCMLM), denoted as avcmlm;
- (9) the proposed varying coefficient semiparametric model averaging prediction approach given in subsection 2.1, denoted as SMAP;
- (10) As the local polynomial regression is also a popular nonparametric estimation method, we employ local constant (and linear) regression technique to estimate varying coefficient functions involved in model (6) according to a referee's suggestion. Corresponding varying coefficient multinomial logistic model and semiparametric model averaging are defined as vcmlm-LC (and vcmlm-LL) and SMAP-LC (and SMAP-LL), where "LC" and "LL" represent that varying coefficient functions are estimated by local constant and linear regression respectively. Please note that we use X_1 as the index variable for vcmlm-LC and vcmlm-LL. Similar to Ke et al. (2016), the kernel function involved in local constant and linear regression was chosen as the Gaussian kernel, and the bandwidth was selected as $h = \left[(J-1)\, p / n \right]^{0.2}$.
- (11) the proposed AdaBoost semiparametric model averaging approach given in subsection 2.2, denoted as abSMAP;
- (12) in order to make a sufficient comparison, we also consider the oracle model, where the true model and true coefficients were known for us. However, the oracle method is of no practical utility and can only be used as a benchmark in simulations.

In this article, we suggest adopting lower order splines, such as linear (q = 2) for all simulation examples. As far as we know, the effect of the splines on the model is multiplicative, so high order splines would lead to complicated interactions and collinearity among the variables. Furthermore, linear splines have an optimal property (Huang et al. (2004) and Kim. (2007)). The number of interior knots is set as $N_n = \left[n^{1/(2q+1)} \right]$, where $[a]$ stands for the largest integer not greater than a.

Example 1. We first consider a binary classification problem and the data are generated from the following multinomial varying-coefficient model

$$\mu_{ij} = \log \frac{p_{ij}}{p_{iJ}} = \xi_j\left(X_{i1}\right) + \boldsymbol{X}_{i,\ 1}^T \boldsymbol{\theta}_j\left(X_{i1}\right) + \boldsymbol{X}_{i,\ 2}^T \boldsymbol{\theta}_j\left(X_{i2}\right), i = 1,...,n,\ j = 1,...,J-1, \quad (12)$$

where $J = 2$, $\xi_1(u) = \sin(2\pi u)$ and

$$\boldsymbol{\theta}_1(u) = \left(2u(1-u), \exp(u-0.5), \cos(2\pi u), 2\exp(-0.5u^2)/(\exp(-0.5u^2)+1), 0, ..., 0\right)^T.$$

The covariate $\boldsymbol{X}_i = \left(X_{i1}, ..., X_{ip}\right)^T$ are generated from mean zero normal distributions with $Cor\left(X_{ij}, X_{il}\right) = 0.5^{|j-l|}$ for $1 \le j, l \le p$. We consider $p = 5, 7,$ and 9, corresponding to different sparsity levels. We note that this true model involves two types of varying coefficients induced by two index variables.

In this example, we generate a training data set of sample size n to estimate the conditional probability $p_{ij} = \Pr\{Y = j \mid \boldsymbol{X}_i\}$ and the model weight vector $\boldsymbol{w}$. Then two types of test samples are generated to calculate prediction performances of all strategies: Type (A): we generate 500 extra observations (the first test data set) from (12) to calculate prediction performances; Type (B): following Ke et al. (2016), we generate a test data from (12) such that there is one category into which subjects are classified with conditional probabilities of 80%–90% (or 90%–100%) and keep simulating until each class has a sample size of 200. Type B test data can ensure that the oracle method can identify correctly with probability 80%–90% (or 90%–100%). All simulation results are based on $R = 300$ replications. Mean squared prediction error (MSPE) and hit rate (HR) are used to evaluate the out-of-sample performances of various methods, given by

$$\text{MSPE} = \frac{1}{n_* \times R} \sum_{r=1}^{R} \sum_{i=1}^{n_*} \sum_{j=1}^{J} \left(p_{ij,r} - \hat{p}_{ij,r}\right)^2, \text{HR} = \frac{1}{n_* \times R} \sum_{r=1}^{R} \sum_{i=1}^{n_*} \mathrm{I}\left(Y_{i,r} = \hat{Y}_{i,r}\right),$$

where n_* is the test sample size, $p_{ij,r}$ is the probability of the ith individual selecting category j in the rth replication and $\hat{p}_{ij,r}$ is its estimate. The hit rate can be regarded as the percentage of correctly forecasting the responses in the test sample.

Example 2. In this example, we consider a three-category classification problem. The responses are generated based on (12) with the following specifications:

$$J = 3, \xi_1(u) = \xi_2(u) = \sin(2\pi u),$$
$$\boldsymbol{\theta}_1(u) = \left(2u(1-u), \exp(u-0.5), 0, ..., 0\right)^T,$$

and

$$\boldsymbol{\theta}_2(u) = \left(\cos(2\pi u), 2\exp(-0.5u^2)/(\exp(-0.5u^2)+1), 0, ..., 0\right)^T.$$

In this example, we set p = 3, 5, 7, and other settings are the same as that in example 1.

Simulation results of examples 1–2 for the type (A) are summarized in Tables 1–2. The following conclusions may be drawn from the simulation results. Firstly, it is clear from the results that the out-of-sample prediction performances of our proposed semiparametric model average prediction SMAP, SMAP-LC, SMAP-LL and abSMAP are superior to the multinomial logistic models (mlm, vcmlm, vcmlm-LC and vcmlm-LL) and existing parametric model average approach (mamlm). On the one hand, mlm and mamlm apply parametric multinomial logistic models to obtain predicted values of the response, which completely ignores the nonlinear functional relationship between the response and covariates. This is the main reason that mlm and mamlm perform badly in simulations. On the other hand, based on the underlying data generation mechanism, vcmlm, vcmlm-LC and vcmlm-LL) use misspecified model sructure to make predictions, leading to poor prediction performance. Although our proposed model averaging approaches (SMAP, SMAP-LC, SMAP-LL and abSMAP) may use misspecified candidate models, they combine more useful information from different candidate models, and thus produce more precise prediction outcomes. Secondly, our proposed methods SMAP, SMAP-LC, SMAP-LL and abSMAP outperform the popular machine learning classification approaches including ranfor, tree, svm and boost in most of cases. Finally, our proposed AdaBoost approach

abSMAP has slightly better discrete prediction outcomes than SMAP as it attains the larger HR in the majority of cases, which indicates that it is meaningful to apply the popular AdaBoost algorithm to model averaging. Simulation results of examples 1–2 for the type (B) are similar and given in Tables S1 and S2 of the supplementary material. In addition, we also add another complicated three-category classification simulation example in the supplementary material, which also shows that our proposed model average methods have an obvious advantage over other methods.

In summary, the proposed model average methods SMAP, SMAP-LC, SMAP-LL and abSMAP appear to perform very well and are competitive compared with the existing methods.

4 Analysis of vehicle silhouettes data

The vehicle silhouettes data to be analyzed was originally collected by Siebert J.P. at the Turing Institute of Glasgow, Scotland in 1986–1987. Evgenia Dimitriadou had converted this data set to R format, which is available in the R package “mlbench”. The original research goal is to identify the types of vehicles within a two-dimension image by application of an ensemble of shape feature extractors to the two-dimension silhouettes of the vehicles. The images were collected by a camera looking downwards at the model vehicle from a fixed angle of elevation, and all kinds of numerical features of vehicles were extracted from the images by the Hierarchical Image Processing System (HIPS). In this experiment, four “Corgie” model vehicles were considered including a double decker bus (***bus***), an Opel Manta 400 (***opel***), Saab 9000 (***saab***) and Cheverolet van (***van***). The main reason that we consider combination of these vehicles is that the bus, the van and either of the cars would be readily distinguishable. The groups sizes are roughly balanced with proportions being 25.8% (***bus***), 25.1% (***opel***), 25.7% (***saab***) and 23.5% (***van***).

This is a four-category classification problem and the response variable (Y) is taken as the auto category. This data set had been studied by Hsu and Lin (2002) and Zhong and Fukushima (2007) by using multi-class support vector machines. The purpose is to classify a given silhouette as one of the

four types according to numerical features extracted from the collected images. The raw data consists of 846 samples with a vehicle labeled response Y and 18 vehicle features extracted from the silhouettes and summarized in Table 3. In this data set, all covariates are continuous predictors and thus any covariate from $\{X_1,...,X_{18}\}$ may serve as the index variable for the varying coefficients, resulting in 18 varying coefficient candidate models for our proposed SMAP approaches.

Before implementing the estimation procedure, we standardize all covariates such that they are mean zero and variance one. Figures 1 and 2 display the means and box-plots of the 18 features for the four types of vehicles. We may notice that *opel* and *saab* have similar vehicle silhouettes and thus are harder to distinguish than the other two types. After performing the semiparametric model averaging over the 18 sub-models, the estimated model weights for semiparametric model averaging prediction methods including SMAP, SMAP-LC and SMAP-LL are all summarized in Table 3. Although the three model averaging approaches obtained by different smoothers yield different model weights, they all assign relatively large weights for submodels with X_1 being the index of varying coefficient function.

To compare with traditional varying coefficient multinomial logistic models, X_1 is chosen in the following as the index variable for vcmlm, vcmlm-LC and vcmlm-LL since it has the greatest model weights based on Table 3. The estimated varying coefficient functions from vcmlm-LL are displayed in Figure S1 of the supplementary material. We can clearly see that almost all estimated functions are nonlinear, which suggests that varying coefficient model structure is more appropriate for this data set. The estimated varying coefficient functions for vcmlm and vcmlm-LC are also similar and not reported for saving space.

To evaluate the predictive performance of various classification methods, the data set is randomly divided into a training sample and a test sample with size n_{train} and n_{test} ($n = n_{train} + n_{test}$), respectively. We consider the test sample size as $n_{test} = 50, 100, 200$ and 300 in the following. We repeat the random splitting

procedure for 200 times and report the means, medians and sample standard deviations (sd) of the hit rates in Table 4. In this empirical application, we also consider the popular single index model (Cui et al. (2011)) and additive model (Huang et al. (2010)) in addition to the approaches examined in simulations. Based on existing literatures, single index multinomial logistic model (simlm) and additive multinomial logistic model (amlm) for multiple categories have the following structure

$$\mathbf{SIMLM}: \mu_{ij} = \log \frac{p_{ij}}{p_{iJ}} = g_j\left(\boldsymbol{X}_i^T \boldsymbol{\beta}\right), i = 1,...,n,\ j = 1,...,J-1,$$

$$\mathbf{AMLM}: \mu_{ij} = \log \frac{p_{ij}}{p_{iJ}} = \mu_j + \sum_{l=1}^{p} f_{lj}\left(X_{il}\right), i = 1,...,n,\ j = 1,...,J-1,$$

where $\boldsymbol{\beta} = \left(\beta_1,...,\beta_p\right)^T \in \mathbb{R}^p$ is an index coefficient and satisfies $\|\boldsymbol{\beta}\| = 1$ and $\beta^{(r)} > 0$ for some $r \in \{1,...,p\}$, μ_j is the intercept term, $g_j(\cdot)$ and $f_{lj}(\cdot)$ are unknown nonparametric functions. Compared with the conventional parametric multinomial logistic model (mlm), semiparametric multinomial logistic models (vcmlm, vcmlm-LC, vcmlm-LL, simlm, amlm and avcmlm), parametric model averaging method (mamlm) and popular machine learning classification approaches (ranfor, tree, svm and boost), our proposed model averaging methods (SMAP, SMAP-LC, SMAP-LL and abSMAP) achieve higher out-of-sample HR. In addition, AdaBoost appears to improve the hit rate on top of SMAP, especially when n_{test} is large. We note that vcmlm, vcmlm-LC, and vcmlm-LL work well in this example when X_1 is selected as the index variable. However, when we replace the index with any other covariate, the HR drops below 80% for almost all cases. The performance of such varying coefficient models is heavily dependent on the selected index variable.

It is of practical interest to investigate the probabilities that individual categories are correctly identified. In the supplementary material, Tables S5–S8 report the corresponding results with different test sample sizes over 200 repetitions. For **bus** and ***van***, we can see that the probabilities of correct identification are higher than 90% for all methods. However, the probabilities

for correctly classifying **opel** and ***saab*** are much lower and the two categories are easily mixed up by all classifiers. One can see clearly that our proposed SMAP, SMAP-LC, SMAP-LL and abSMAP attain the highest rates of correct classification among all competing classifiers for ***opel*** and ***saab***, indicating stronger discrimination ability for such difficult categories with high similarity in physical appearance.

5 Conclusion and discussion

Varying coefficient models have become popular in many fields such as applied econometrics, biomedical informatics and sociology. They not only allow a more flexible specification compared to linear regression models but also retain some of the desirable properties such as the interpretability. In the context of varying coefficient multinomial logit models, we develop new varying coefficient semiparametric model averaging approaches (SMAP, SMAP-LC and SMAP-LL), which is more robust against model mis-specification. Furthermore, coupled with the AdaBoost algorithm, we propose a more effective classification procedure (abSMAP). Extensive numerical studies suggest that our proposed methods have excellent prediction ability compared to traditional multinomial logistic models, parametric model averaging methods and popular machine learning classification approaches. We believe that SMAP, SMAP-LC, SMAP-LL and abSMAP represent qualified alternatives and deserve discussions in greater detail in both theory and application.

Recently, analysis of ultra-high dimensional varying coefficient models has attracted much attention. But most works were focused on variable selection and feature screening under the signal sparseness condition (Fan et al., 2014; Liu et al., 2014; Cheng et al., 2016). Despite the enthusiasm for investigation of semiparametric model averaging, how well it performs in ultra-high dimensional setting is still of limited understanding. It is an interesting task to consider ultra-high dimensional varying coefficient multinomial logistic model averaging for categorical data. Though Theorem 1 allows p to diverge slowly at a polynomial order of the sample size n, in general the convergence only

holds for $p = o(n)$. When $p > n$, it is difficult or infeasible to implement the maximum likelihood estimation (5) at the sub-model building stage. Some authors considered regularized estimation for multinomial logistic regression in their recent works and suggested possible computational tools, including Friedman et al. (2010), Vincent and Hansen (2014), Tutz et al. (2015), Power et al. (2018), among others. Integrating such computational components in the model average framework increases the complexity level and demands additional theoretical and empirical support. Such an intriguing extension of the current analysis needs to be solved in future research.

Acknowledgments

Jing Lv's work is partially supported by the National Natural Science Foundation of China Grant 11801466, the Basic and Frontier Research Program of Chongqing Grant cstc2017jcyjAX0182 and the Fundamental Research Funds for the Central Universities Grant XDJK2019C105. Jialiang Li's work is partially supported by Academic Research Funds R-155-000-174-114, R-155-000-195-114 and Tier 2 Ministry of Education funds in Singapore MOE2017-T2-2-082: R-155-000-197-112 (Direct cost) and R-155-000-197-113 (IRC).

Appendix

Proof of Theorem 1.

Let $\delta_n = K_n p^2 n^{-1/2+\delta}$. Following Fan and Peng (2004) and Chen et al. (2018a), to prove Theorem 1, it suffices to show that there exists a constant C_0 such that for the $p \times 1$ vector $\boldsymbol{u} = (u_1, ..., u_p)^T$,

$$\lim_{n\to\infty} Pr\left(\inf_{\|\boldsymbol{u}\|=C_0, (\boldsymbol{w}^0+\delta_n\boldsymbol{u})\in W} \ell(\boldsymbol{w}^0 + \delta_n \boldsymbol{u}) > \ell(\boldsymbol{w}^0) \right) = 1. \qquad (13)$$

Note that

$$
\begin{aligned}
&\ell(\boldsymbol{w}^0 + \delta_n \boldsymbol{u}) - \ell(\boldsymbol{w}^0) \\
&= \left(\mathbf{Y} - \mathbf{P}\boldsymbol{w}^0 - \delta_n \mathbf{P}\boldsymbol{u}\right)^T \left(\mathbf{Y} - \mathbf{P}\boldsymbol{w}^0 - \delta_n \mathbf{P}\boldsymbol{u}\right) - \ell(\boldsymbol{w}^0) \\
&= \delta_n^2 \boldsymbol{u}^T \mathbf{P}^{*T} \mathbf{P}\boldsymbol{u} - 2\delta_n \left(\boldsymbol{P} - \mathbf{P}\boldsymbol{w}^0\right)^T \mathbf{P}\boldsymbol{u} \qquad (14) \\
&-2\delta_n \left(\mathbf{Y} - \boldsymbol{P}\right)^T \mathbf{P}^{*}\boldsymbol{u} - 2\delta_n \left(\mathbf{Y} - \boldsymbol{P}\right)^T \left(\mathbf{P}^{*} - \mathbf{P}\right)\boldsymbol{u} \\
&\equiv \Lambda_1 + \Lambda_2 + \Lambda_3 + \Lambda_4,
\end{aligned}
$$

where $\mathbf{P}^*$ is the same as $\hat{\mathbf{P}}$ except that $\hat{\gamma}_j^{(s)}$ is replaced by $\gamma_j^{(s)*}$ for all $1 \le s \le p$ and $1 \le j \le J-1$.

Also, by Condition 3, we have

$$\delta_n^2 \boldsymbol{u}^T \mathbf{P}^{*T} \mathbf{P}\boldsymbol{u} > Cn\delta_n^2 \|\boldsymbol{u}\|^2 > 0 \quad (15)$$

with probability tending to 1. Hence (13) holds if $|\Lambda_2|, |\Lambda_3|$, and $|\Lambda_4|$ are dominated by Λ_1 asymptotically.

Let us first consider Λ_2. As $\psi_n = E\left\|\boldsymbol{P} - \mathbf{P}\boldsymbol{w}^0\right\|^2$, we have $\left\|\boldsymbol{P} - \mathbf{P}\boldsymbol{w}^0\right\| = O_p(\psi_n^{1/2})$. Moreover, by Condition 3,

$$\left\|\mathbf{P}\right\| = \lambda_{max}^{1/2}\left(\mathbf{P}^T\mathbf{P}\right) = O_p(n^{1/2}p^{1/2}). \qquad (16)$$

Hence we obtain

$$|\Lambda_2| \le 2\delta_n \left\|\boldsymbol{P} - \mathbf{P}\boldsymbol{w}^0\right\| \left\|\mathbf{P}\right\| \|\boldsymbol{u}\| = O_p(\delta_n \psi_n^{1/2} n^{1/2} p^{1/2}). \quad (17)$$

Now, to consider Λ_3, let $e_{ij} = \mathrm{Y}_{ij} - p_{ij}$, and we rewrite $\hat{p}_{ij}^{(s)}$ as $p_{ij}^{(s)*}$ when $\hat{\gamma}_j^{(s)}$ in $\hat{p}_{ij}^{(s)}$ is replaced by $\gamma_j^{(s)*}$. Observe that

$$
\begin{aligned}
& E\left\|\mathbf{P}^{*T}(\mathbf{Y}-\boldsymbol{P})\right\|^2 \\
&= \sum_{s=1}^{p} E\left(\sum_{i=1}^{n}\sum_{j=1}^{J} p_{ij}^{(s)*} e_{ij}\right)^2 \\
&\le J\sum_{j=1}^{J}\sum_{s=1}^{p} E\left(\sum_{i=1}^{n} p_{ij}^{(s)*} e_{ij}\right)^2 \qquad (18) \\
&= J\sum_{j=1}^{J}\sum_{s=1}^{p}\sum_{i=1}^{n} E\left(p_{ij}^{(s)*} e_{ij}\right)^2 \\
&= O(np),
\end{aligned}
$$

where the second equality follows from the fact that $E\left(p_{ij}^{(s)*} e_{ij}\right)=0$, and the equality on the last line is true because $p_{ij}^{(s)*}$ and $|e_{ij}|$ are bounded uniformly in *i*, *s*, and *j*. This implies that

$$
|\Lambda_3| \le 2\grave{o}_n \|\boldsymbol{u}\| \left\|\mathbf{P}^{*T}(\mathbf{Y}-\boldsymbol{P})\right\| = O_p(\grave{o}_n n^{1/2} p^{1/2}). \qquad (19)
$$

Finally, let us consider Λ_4. Let
$\mu_{ij}^{(s)*} = \boldsymbol{\Pi}_i^{(s)T}\boldsymbol{\gamma}_j^{(s)*}$, $\boldsymbol{\mu}_i^{(s)*} = \left(\mu_{ij}^{(s)*} : 1 \le j \le J-1\right)^T$, $\boldsymbol{\mu}_i = \left(\mu_{ij} : 1 \le j \le J-1\right)^T$, and
$\boldsymbol{\mu}_i^{(s)} = \left(\hat{\mu}_{ij}^{(s)} : 1 \le j \le J-1\right)^T$. Write

$$
\hat{p}_{ij}^{(s)} - p_{ij}^{(s)*} = \left(\boldsymbol{\mu}_i^{(s)} - \boldsymbol{\mu}_i^{(s)*}\right)^T \frac{\partial p_{ij}}{\partial \boldsymbol{\mu}_i}\Big|_{\boldsymbol{\mu}_i = \boldsymbol{\mu}_i^{(s)}}, \qquad (20)
$$

where $\boldsymbol{\mu}_i^{(s)}$ lies between $\boldsymbol{\mu}_i^{(s)}$ and $\boldsymbol{\mu}_i^{(s)*}$. Note that when $1 \le j \le J-1$,

$$
\frac{\partial p_{ij}}{\partial \boldsymbol{\mu}_i} = \frac{e^{\mu_{ij}}\left(1+\sum_{j=1}^{J-1} e^{\mu_{ij}}\right)\boldsymbol{l}_j - e^{\mu_{ij}}\left(\sum_{j=1}^{J-1} e^{\mu_{ij}}\boldsymbol{l}_j\right)}{\left(1+\sum_{j=1}^{J-1} e^{\mu_{ij}}\right)^2}, \qquad (21)
$$

and when *j* = *J*,

$$
\frac{\partial p_{ij}}{\partial \boldsymbol{\mu}_i} = \frac{-\sum_{j=1}^{J-1} e^{\mu_{ij}}\boldsymbol{l}_j}{\left(1+\sum_{j=1}^{J-1} e^{\mu_{ij}}\right)^2}, \qquad (22)
$$

where $\boldsymbol{l}_j$ is the jth column of the identity matrix $\boldsymbol{I}_{J-1}$. Hence we have

$$\left\| \frac{\partial p_{ij}}{\partial \boldsymbol{\mu}_i} |_{\boldsymbol{\mu}_i = \boldsymbol{\mu}_i^{(s)}} \right\|^2 \le C \qquad (23)$$

uniformly in i, s, and j. Furthermore, denoting $\hat{\boldsymbol{\gamma}}^{(s)} = \left(\hat{\boldsymbol{\gamma}}_1^{(s)T}, ..., \hat{\boldsymbol{\gamma}}_{J-1}^{(s)T} \right)^T$, and by (20) and Condition 2, we find that

$$\begin{aligned}
&\left\| \mathbf{P} - \mathbf{P}^* \right\|^2 \\
&\le \sum_{s=1}^{p} \sum_{j=1}^{J} \sum_{i=1}^{n} \left(\hat{p}_{ij}^{(s)} - p_{ij}^{(s)*} \right)^2 \\
&\le C \sum_{s=1}^{p} \sum_{j=1}^{J} \sum_{i=1}^{n} \left\| \boldsymbol{\mu}_i^{(s)} - \boldsymbol{\mu}_i^{(s)*} \right\|^2 \\
&\le C \sum_{s=1}^{p} \sum_{i=1}^{n} \left(\hat{\boldsymbol{\gamma}}^{(s)} - \boldsymbol{\gamma}^{(s)*} \right)^T \begin{pmatrix} \boldsymbol{\Pi}_i^{(s)} \boldsymbol{\Pi}_i^{(s)T} & & \\ & \ddots & \\ & & \boldsymbol{\Pi}_i^{(s)} \boldsymbol{\Pi}_i^{(s)T} \end{pmatrix} \left(\hat{\boldsymbol{\gamma}}^{(s)} - \boldsymbol{\gamma}^{(s)*} \right) \\
&\le C \sum_{s=1}^{p} \lambda_{max} \left(\sum_{i=1}^{n} \boldsymbol{\Pi}_i^{(s)} \boldsymbol{\Pi}_i^{(s)T} \right) \left\| \hat{\boldsymbol{\gamma}}^{(s)} - \boldsymbol{\gamma}^{(s)*} \right\|^2 \\
&\le Cn \sum_{s=1}^{p} \sum_{k=1}^{pK_n} \left(\frac{1}{n} \sum_{i=1}^{n} \left| \Pi_{ik}^{(s)} \right|^2 \right) \max_{1 \le s \le p} \left\| \hat{\boldsymbol{\gamma}}^{(s)} - \boldsymbol{\gamma}^{(s)*} \right\|^2 \\
&= \max_{1 \le s \le p} \left\| \hat{\boldsymbol{\gamma}}^{(s)} - \boldsymbol{\gamma}^{(s)*} \right\|^2 O_p(np^2 K_n). \qquad (24)
\end{aligned}$$

Note that

$$\hat{\boldsymbol{\gamma}}^{(s)} - \boldsymbol{\gamma}^{(s)*} = -\left(\frac{\partial^2 l_s(\boldsymbol{\gamma}^{(s)})}{\partial \boldsymbol{\gamma}^{(s)} \partial \boldsymbol{\gamma}^{(s)T}} |_{\boldsymbol{\gamma}^{(s)} = \tilde{\boldsymbol{\gamma}}^{(s)}} \right)^{-1} \frac{\partial l_s(\boldsymbol{\gamma}^{(s)})}{\partial \boldsymbol{\gamma}^{(s)}} |_{\boldsymbol{\gamma}^{(s)} = \boldsymbol{\gamma}^{(s)*}}, \qquad (25)$$

where $\tilde{\boldsymbol{\gamma}}^{(s)}$ lies between $\hat{\boldsymbol{\gamma}}^{(s)}$ and $\boldsymbol{\gamma}^{(s)*}$. Also, from the definition of $\boldsymbol{\gamma}^{(s)*}$, we have

$$
\begin{aligned}
& E \max_{1\le s\le p} \left\| \frac{1}{\sqrt{n}} \frac{\partial l_s(\boldsymbol{\gamma}^{(s)})}{\partial \boldsymbol{\gamma}^{(s)}} \Big|_{\boldsymbol{\gamma}^{(s)}=\boldsymbol{\gamma}^{(s)*}} \right\|^2 \\
& \le \sum_{s=1}^{p} E \left\| \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \sum_{j=1}^{J-1} \left\{ \mathrm{Y}_{ij} - e^{\boldsymbol{\Pi}_i^{(s)T} \boldsymbol{\gamma}_j^{(s)*}} \Big/ \left(1 + \sum_{j=1}^{J-1} e^{\boldsymbol{\Pi}_i^{(s)T} \boldsymbol{\gamma}_j^{(s)*}} \right) \right\} [\boldsymbol{\Pi}_i^{(s)}]^j \right\|^2 \\
& = \sum_{s=1}^{p} \sum_{j=1}^{J-1} \sum_{k=1}^{pK_n} E \left[\frac{1}{\sqrt{n}} \sum_{i=1}^{n} \left\{ \mathrm{Y}_{ij} - e^{\boldsymbol{\Pi}_i^{(s)T} \boldsymbol{\gamma}_j^{(s)*}} \Big/ \left(1 + \sum_{j=1}^{J-1} e^{\boldsymbol{\Pi}_i^{(s)T} \boldsymbol{\gamma}_j^{(s)*}} \right) \right\} \Pi_{ik}^{(s)} \right]^2 \qquad (26) \\
& \le C \sum_{s=1}^{p} \sum_{k=1}^{pK_n} E \left| \Pi_{ik}^{(s)} \right|^2 \\
& = O(p^2 K_n),
\end{aligned}
$$

where $[\boldsymbol{\Pi}_i^{(s)}]^j$ consists of $(J-1)$ pK_n dimensional vectors, with the jth part being $\boldsymbol{\Pi}_i^{(s)}$ and all others zeroes. Hence by (26) and Condition 1, it follows that

$$
\begin{aligned}
& \max_{1\le s\le p} \left\| \hat{\boldsymbol{\gamma}}^{(s)} - \boldsymbol{\gamma}^{(s)*} \right\|^2 \\
& \le n^{-1} \max_{1\le s\le p} \lambda_{max}^2 \left\{ \left(\frac{1}{n} \frac{\partial^2 l_s(\boldsymbol{\gamma}^{(s)})}{\partial \boldsymbol{\gamma}^{(s)} \partial \boldsymbol{\gamma}^{(s)T}} \Big|_{\boldsymbol{\gamma}^{(s)}=\tilde{\boldsymbol{\gamma}}^{(s)}} \right)^{-1} \right\} \max_{1\le s\le p} \left\| \frac{1}{\sqrt{n}} \frac{\partial l_s(\boldsymbol{\gamma}^{(s)})}{\partial \boldsymbol{\gamma}^{(s)}} \Big|_{\boldsymbol{\gamma}^{(s)}=\boldsymbol{\gamma}^{(s)*}} \right\|^2 \qquad (27) \\
& = O_p(p^2 K_n n^{-1}),
\end{aligned}
$$

which, together with (24), leads to

$$
\left\| \mathbf{P} - \mathbf{P}^* \right\|^2 = O_p(n p^2 K_n p^2 K_n n^{-1}) = O_p(K_n^2 p^4). \qquad (28)
$$

Recognising that $\|\mathbf{Y} - \boldsymbol{P}\| = O_p(n^{1/2})$, we have

$$
|\Lambda_4| \le 2\dot{\alpha}_n \left\| \mathbf{Y} - \boldsymbol{P} \right\| \left\| \mathbf{P} - \mathbf{P}^* \right\| \|\boldsymbol{u}\| = O_p(\dot{\alpha}_n n^{1/2} K_n p^2). \qquad (29)
$$

By (15), (17), (19), (29), and Condition 4, we can see that $|\Lambda_2|$, $|\Lambda_3|$, and $|\Lambda_4|$ are dominated by Λ_1 asymptotically. That completes the proof.

References

Akaike, H. (1974). A new look at the statistical model identification. IEEE Transactions on Automatic Control, 19, 716–723.

Akaike, H. (1979). A Bayesian extension of the minimum AIC procedure of autoregressive model fitting. Biometrika, 66, 237–242.

Ando, T., and Li, K-C. (2014). A model-averaging approach for high-dimensional regression. Journal of the American Statistical Association, 109, 254–265.

Ando, T., and Li, K-C. (2017). A weight-relaxed model averaging approach for high-dimensional generalized linear models. The Annals of Statistics, 45, 2654–2679.

Barut, E., Fan, J., and Verhasselt, A. (2016). Conditional sure independence screening. Journal of the American Statistical Association, 111, 1266–1277.

Bennett, K. (2008) Response to “Mease and Wyner, Evidence Contrary to the Statistical View of Boosting”. Journal of Machine Learning Research, 9, 157-164.

Biau, G. (2012). Analysis of a random forests model. Journal of Machine Learning Research, 13, 1063–1095.

Bickel, P.J. and Levina, E. (2008). Regularized estimation of large covariance matrices. The Annals of Statistics, 36, 199-227.

Breiman, L. (1995). Better subset regression using the nonnegative garotte. Technometrics, 37, 373–384.

Breiman, L. (2001). Random forests. Machine Learning, 45, 5–32.

Breiman L., Friedman J. H., Olshen R. A., and Stone, C. J. (1984). Classification and regression trees. Wadsworth.

Bühlmann, P., and Yu, B. (2006). Sparse boosting. Journal of Machine Learning Research, 7, 1001–1024.

Bühlmann, P., and Yu, B. (2008). Response to ”Mease and Wyner, Evidence Contrary to the Statistical View of Boosting”. Journal of Machine Learning Research, 9, 187-194.

Chen, J., Li, D., Linton, O., and Lu, Z. (2018a). Semiparametric ultra-high dimensional model averaging of nonlinear dynamic time series. Journal of the American Statistical Association, 113, 919–932.

Chen, L., Wan, A.T.K., Tso, G., and Zhang, X. (2018b). A model averaging approach for the ordered probit and nested logit models with applications. Journal of Applied Statistics, 45, 3012–3052.

Cheng, M. Y., Honda, T., and Zhang, J. T. (2016). Forward variable selection for sparse ultra-high dimensional varying coefficient models. Journal of the American Statistical Association, 111, 1209–1221.

Cui, X., Hardle, W.K., Zhu, L. (2011) The EFM approach for single-index models. The Annals of Statistics, 39, 1658–1688.

Crammer, K., and Singer, Y. (2001). On the algorithmic implementation of multiclass kernel-based vector machines. Journal of Machine Learning Research, 2, 265–292.

de Boor, C. (2001). A practical guide to splines. Springer, New York.

Domingo, C., and Watanabe, O. (2000). Madaboost: a modified version of adaboost. In Proceedings of the Thirteenth Annual Conference on Computational Learning Theory, 180–189.

Fan, J., and Huang, T. (2005). Profile likelihood inferences on semiparametric varying-coefficient partially linear models. Bernoulli, 11, 1031–1057.

Fan, J., and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. Journal of the American Statistical Association, 96, 1348–1360.

Fan, J., and Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. Journal of the Royal Statistical Society, Series B, 70, 849–911.

Fan, J., Ma, Y., and Dai, W. (2014). Nonparametric independence screening in sparse ultra-high-dimensional varying coefficient models. Journal of the American Statistical Association, 109, 1270–1284.

Fan, J. and Peng, H. (2004). Nonconcave penalized likelihood with a diverging number of parameters. The Annals of Statistics, 32, 928-961.

Fan, J., and Song, R., (2010). Sure independence screening in generalized linear models with NP-dimensionality. The Annals of Statistics, 38, 3567–3604.

Fan, J., Yao, Q., Cai, Z. (2003). Adaptive varying-coefficient linear models. Journal of the Royal Statistical Society, Series B, 65, 57–80.

Fan, J., and Zhang, W. (1999). Statistical estimation in varying coefficient models. The Annals of Statistics, 27, 1491–1518.

Fan, J., and Zhang, W. (2008). Statistical methods with varying coefficient models. Statistics and Its Interface, 1, 179–195.

Findley, D.F. (1991). Counterexamples to parsimony and BIC. Annals of the Institute of Statistical Mathematics, 43, 505–514.

Freund, Y. (2001a). An adaptive version of the boost-by-majority algorithm. Machine learning, 43, 293–318.

Freund, Y., and Schapire, R. (1997). A decision theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55, 119–139.

Freund, Y. (1995). Boosting a weak learning algorithm by majority. Information and Computation, 121, 256–285.

Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting. The Annals of Statistics, 28, 337–407.

Friedman, J. (2001b). Greedy function approximation: a gradient boosting machine. The Annals of statistics, 29, 1189–1232.

Friedman, J., Hastie, T., Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. Journal of Statistical Software, 33(1), 1–22.

Gao, Y., Zhang, X., Wang, S., and Zou, G. (2016). Model averaging based on leave-subject-out cross-validation. Journal of Econometrics, 192, 139–151.

Guo, Z., Li, L., Lu, W., and Li, B. (2015). Groupwise dimension reduction via envelope method. Journal of the American Statistical Association, 110, 1515–1527.

Hansen, B.E. (2007). Least squares model averaging. Econometrica, 75, 1175–1189.

Hansen, B.E. (2008). Least-squares forecast averaging. Journal of Econometrics, 146, 342–350.

Hansen, B.E., and Racine, J.S. (2012). Jackknife model averaging. Journal of Econometrics, 167, 38–46.

Hastie, T., and Tibshirani, R. (1993). Varying-coefficient model. Journal of the Royal Statistical Society, Series B, 55, 757–796.

Hosmer, D.W., and Lemeshow, S. (2004). Applied logistic regression. John Wiley & Sons, New York.

Hsu, C-W., and Lin, C-J. (2002). A comparison of methods for multiclass support vector machines. IEEE Transactions on Neural Networks, 13, 415–425.

Huang, J.Z., Wu, C.O., Zhou, L. (2004). Polynomial spline estimation and inference for varying coefficient models with longitudinal data. Statistica Sinica 14, 763–788.

Huang, J., Horowitz, J.Z., Wei, F. (2010) Variable selection in nonparametric additive models The Annals of Statistics, 38, 4, 2282–2313.

Huang, T., and Li, J. (2018). Semiparametric model average prediction in panel data analysis. Journal of Nonparametric Statistics, 30, 125–144.

Ke, Y., Fu, B., and Zhang, W. (2016). Semi-varying coefficient multinomial logistic regression for disease progression risk prediction. Statistics in Medicine, 35, 4764–4778.

Kim, M. (2007). Quantile regression with varying coefficients. The Annals of Statistics, 35, 92–108.

Lee, Y., Lin, Y., and Wahba., G. (2004). Multicategory support vector machines: theory and application to the classification of microarray data and satellite radiance data. Journal of the American Statistical Association, 99, 67–81.

Li, A.H., and Bradic, J. (2018). Boosting in the presence of outliers: Adaptive classification with nonconvex loss functions. Journal of the American Statistical Association, 113, 660–674.

Li, C., Li, Q., Racine, J., and Zhang, D. (2018a). Optimal model averaging of varying coefficient models. McMaster University, Department of Economics Working Paper, http://dx.doi.org/10.2139/ssrn.2905268.

Li, D., Linton, O., and Lu, Z. (2015). A flexible semiparametric forecasting model for time series. Journal of Econometrics, 187, 345–357.

Li, J., Jiang, B., and Fine, J. (2013). Multicategory reclassification statistics for assessing improvements in diagnostic accuracy. Biostatistics, 14, 382–394.

Li, J., Xia, X., Wong, W., and Nott, D. (2018b). Varying-coefficient semiparametric model averaging prediction. Biometrics, 74, 1417–1426.

Li, W., Li, B., and Yin, X. (2014). On efficient dimension reduction with respect to a statistical functional of interest. The Annals of Statistics, 42, 382–412.

Liang, H., Zou, G., Wan, A.T.K., and Zhang, X. (2011). Optimal weight choice for frequentist model average estimators. Journal of the American Statistical Association, 106, 1053–1066.

Liu, C. (2015). Distribution theory of the least squares averaging estimator. Journal of Econometrics, 186, 142–159.

Liu, J., Li, R., and Wu, R. (2014). Feature selection for varying coefficient models with ultrahigh-dimensional covariates. Journal of the American Statistical Association, 109, 266–274.

Ma, Y., and Zhu, L. (2013). Efficient estimation in sufficient dimension reduction. The Annals of Statistics, 41, 250–268.

McCullagh, P., and Nelder, J.A. (1989). Generalized linear models, 2nd edn. Chapman and Hall, London, New York.

Mease, D. and Wyner,A. (2008). Evidence contrary to the statistical view of boosting. Journal of Machine Learning Research, 9, 131–156.

Powell, J. L., Stock, J. H. and Stoker, T. M. (1989). Semiparametric estimation of index coefficients. Econometrica 57, 1403–1430.

Power, S., Hastie, T., Tibshirani, R. (2018). Statistical Modelling, 18(56), 388–410.

Rahim, N.A., MP, P., and Adom, A.H. (2013). Adaptive boosting with SVM classifier for moving vehicle classification. Procedia Engineering, 53, 411–419.

Ravikumar, P. and Lafferty, J. and Liu, H. and Wasserman, L. (2009). Sparse Additive Models. Journal of the Royal Statistical Society, Series B, 71, 1009–1030.

Ripley, B. D. (1996). Pattern recognition and neural networks. Cambridge University Press, Cambridge.

Schapire, R. (1997). Using output codes to boost multiclass learning problems. Proceedings of the Fourteenth International Conference on Machine Learning. Morgan Kauffman.

Schapire, R., and Singer, Y. (1999). Improved boosting algorithms using confidence-rated prediction. Machine Learning, 37, 297–336.

Scornet, E., Biau, G., and Vert, J.P. (2015). Consistency of random forests. The Annals of Statistics, 43, 1716–1741.

Sun, Z. and Ban, X. (2013). Vehicle classification using GPS data. Transportation Research Part C, 37, 102–117.

Tutz, G., Pobnecker, W., Uhlmann, L. (2015). Variable selection in general multinomial logit models. Computational Statistics and Data Analysis, 82, 207–222.

Venables, W. N., and Ripley, B. D. (2002). Modern applied statistics with S. Fourth edition, Springer.

Vincent, M. and Hansen, N. R. (2014). Sparse group lasso and high dimensional multinomial classification. Computational Statistics and Data Analysis, 71, 771–786.

Wan, A.T.K., Zhang, X., and Zou, G. (2010). Least squares model averaging by Mallows criterion. Journal of Econometrics, 156, 277–283.

Wan, A.T.K., Zhang, X., and Wang, S. (2014). Frequentist model averaging for multinomial and ordered logit models. International Journal of Forecasting, 30, 118–128.

Xia, Y., Tong, H., Li, W. K. and Zhu, L. (2002). An adaptive estimation of dimension reduction space (with discussions). Journal of the Royal Statistical Society, Series B, 64, 363–410.

Yuan, M., and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society, Series B, 68, 49–67.

Zhang, X., and Liang, H. (2011). Focused information criterion and model averaging for generalized additive partial linear models. The Annals of Statistics, 39, 174–200.

Zhang, X., and Liu, C-A. (2018). Inference after model averaging in linear regression models. Econometric Theory, 35, 816–841.

Zhang, X., Yu, D., Zou, G., and Liang, H. (2016). Optimal model averaging estimation for generalized linear models and generalized linear mixed-effects models. Journal of the American Statistical Association, 111, 1775–1790.

Zhang, X., and Wang, W. (2018). Optimal model averaging estimation for partially linear models. Statistica Sinica, 29, 693–718.

Zhang, T. and Yu, B. (2005). Boosting with early stopping: Convergence and consistency. The Annals of Statistics, 33, 1538–1579.

Zhong, P., and Fukushima, M. (2007). Regularized nonsmooth Newton method for multi-class support vector machines. Optimization Methods and Software, 22, 225–236.

Zhu, J., Zou, H., Rosset, S., and Hastie, T. (2009). Multi-class AdaBoost. Statistics and Its Interface, 2, 349–360.

Zhu, R., Wan, A.T.K., Zhang, X., and Zou, G. (2018). A Mallows-type model averaging estimator for the varying-coefficient partially linear model. Journal of the American Statistical Association, 114, 882–892.

Zou, H., (2006). The adaptive lasso and its oracle properties. Journal of the American Statistical Association, 101, 1418–1429.

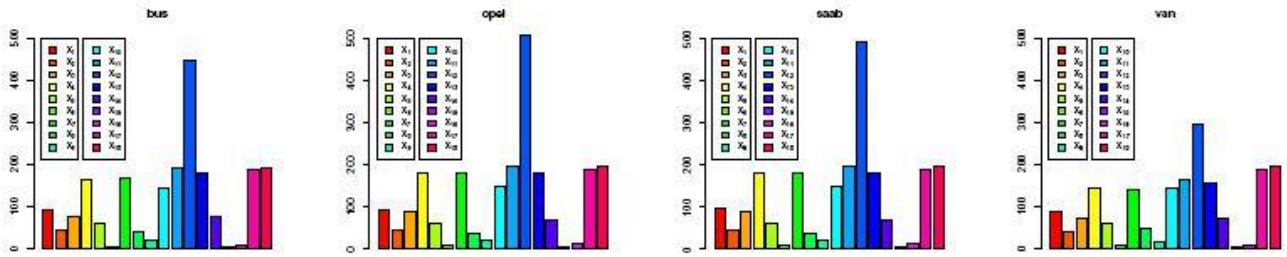

Fig. 1 The bar charts of the means of 18 original predictor variables for the four types of vehicles.

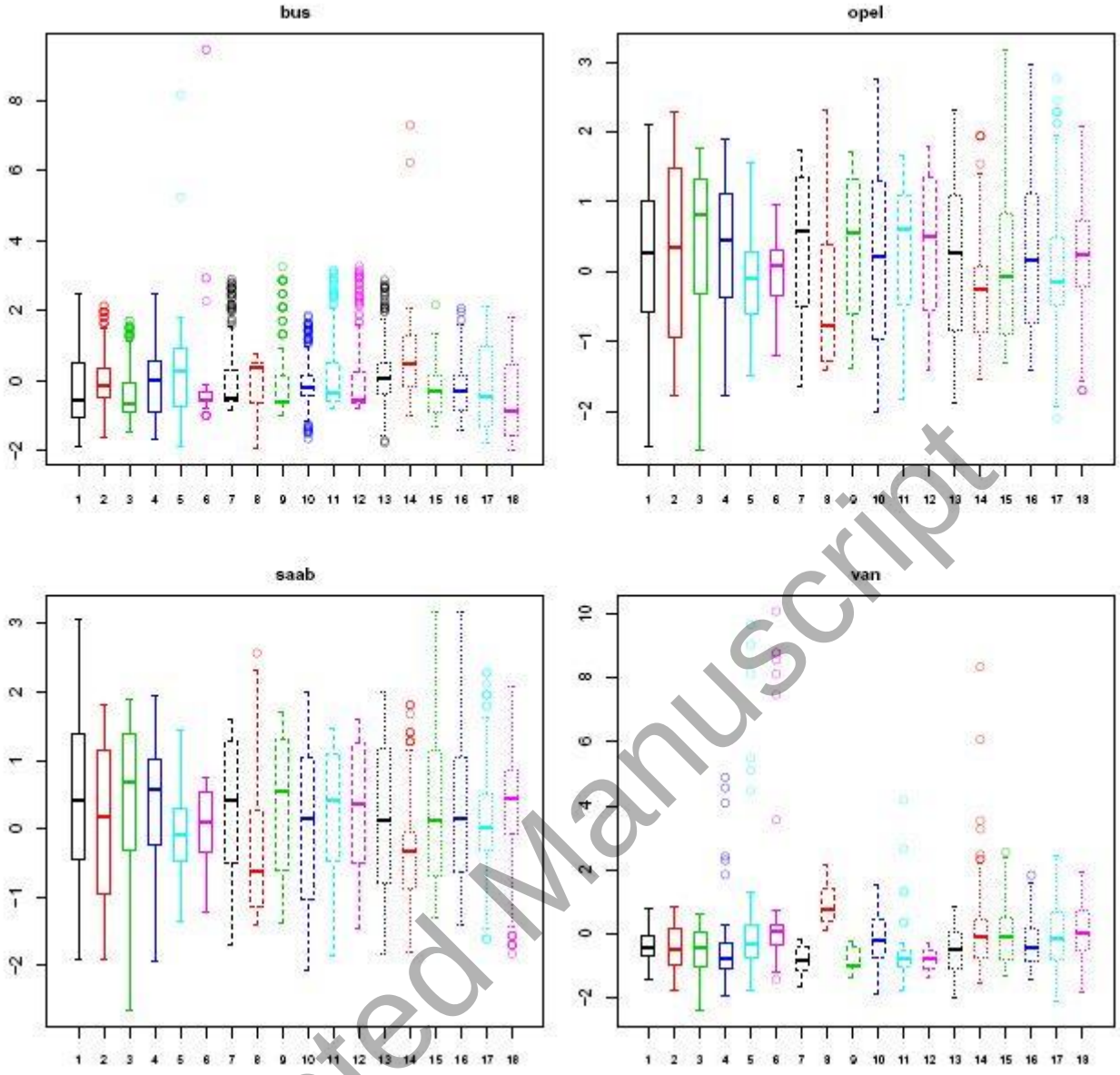

Fig. 2 The box-plots of 18 predictor variables for the four types of vehicles.

Table 1 Simulation results of HR and MSPE for the type (A) test sample of example 1 with the training sample size n = 200 and 400.

n	Method	p = 5		p = 7		p = 9	
		HR	MSPE	HR	MSPE	HR	MSPE
200	ranfor	0.78500	0.06217	0.77549	0.07101	0.77128	0.07364
	tree	0.72724	0.12645	0.72039	0.13658	0.71680	0.14345
	svm	0.79263	0.05804	0.77301	0.07064	0.75522	0.07969
	boost	0.78642	0.09259	0.78006	0.09782	0.77165	0.10255
	mlm	0.73269	0.09259	0.72814	0.09525	0.72271	0.09756
	mamlm	0.73224	0.09255	0.72834	0.09494	0.72337	0.09703
	vcmlm	0.79439	0.05959	0.78624	0.07182	0.76801	0.09285
	vcmlm-LC	0.79703	0.05218	0.79061	0.05766	0.78127	0.06391
	vcmlm-LL	0.80083	0.05505	0.79355	0.06395	0.78279	0.07431
	avcmlm	0.70635	0.1141	0.68985	0.12935	0.67446	0.14848
	SMAP	0.80744	0.04511	0.79486	0.05447	0.77324	0.06570
	SMAP-LC	0.80453	0.04745	0.79678	0.05311	0.78729	0.05903
	SMAP-LL	0.80996	0.04403	0.80108	0.05094	0.78967	0.05827
	abSMAP	0.81052	0.04685	0.80103	0.05417	0.78639	0.06151
400	ranfor	0.80393	0.04985	0.79713	0.05740	0.79141	0.06008
	tree	0.75326	0.09992	0.75109	0.10571	0.74239	0.11350
	svm	0.80938	0.04626	0.79036	0.05842	0.77320	0.06789
	boost	0.80682	0.08424	0.80333	0.08884	0.79503	0.09316
	mlm	0.73358	0.08962	0.73189	0.09141	0.73127	0.09221
	mamlm	0.73380	0.08960	0.73221	0.09125	0.73152	0.09191
	vcmlm	0.81343	0.04389	0.80721	0.05155	0.79097	0.06151
	vcmlm-LC	0.81121	0.04251	0.80897	0.04600	0.79941	0.05067
	vcmlm-LL	0.81680	0.04108	0.81343	0.04664	0.80327	0.05239

	avcmlm	0.72824	0.09698	0.71836	0.10574	0.70856	0.11378
	SMAP	0.82606	0.03281	0.81608	0.03823	0.80018	0.04374
	SMAP-LC	0.81965	0.03802	0.81489	0.04151	0.80650	0.04560
	SMAP-LL	0.82567	0.03250	0.81997	0.03676	0.81138	0.04091
	abSMAP	0.82681	0.03780	0.82109	0.04100	0.81065	0.04593
	oracle	0.87605	0.00000	0.87605	0.00000	0.87605	0.00000

Table 2 Simulation results of HR and MSPE for the type (A) test sample of example 2 with the training sample size n = 200 and 400.

n	Method	p = 3		p = 5		p = 7	
		HR	MSPE	HR	MSPE	HR	MSPE
200	ranfor	0.62097	0.03833	0.60480	0.04266	0.59340	0.04695
	tree	0.57091	0.07938	0.55532	0.09649	0.54840	0.10363
	svm	0.62644	0.03882	0.58380	0.05128	0.56238	0.05872
	boost	0.61099	0.05971	0.59591	0.06309	0.58654	0.06518
	mlm	0.50599	0.06944	0.50248	0.07164	0.50600	0.07391
	mamlm	0.50586	0.06919	0.50251	0.07098	0.50661	0.07314
	vcmlm	0.62717	0.03679	0.60525	0.05049	0.58840	0.06836
	vcmlm-LC	0.62017	0.03578	0.60298	0.04233	0.58874	0.04823
	vcmlm-LL	0.62915	0.03674	0.61154	0.04690	0.59789	0.05631
	avcmlm	0.57564	0.06401	0.54684	0.08325	0.52412	0.10115
	SMAP	0.64057	0.02949	0.61647	0.03808	0.60060	0.04697
	SMAP-LC	0.62893	0.03236	0.61077	0.03821	0.59713	0.04379
	SMAP-LL	0.63768	0.03030	0.61849	0.03757	0.60405	0.04393
	abSMAP	0.64217	0.03027	0.62262	0.03547	0.60548	0.04167
400	ranfor	0.64225	0.03105	0.63321	0.03367	0.59228	0.04756
	tree	0.59385	0.05563	0.58460	0.06185	0.54409	0.10475
	svm	0.64704	0.03080	0.61309	0.04107	0.56365	0.05875
	boost	0.62933	0.05510	0.62414	0.05756	0.58562	0.06559
	mlm	0.50416	0.06735	0.50382	0.06892	0.50141	0.07463
	mamlm	0.50402	0.06712	0.50378	0.06853	0.50164	0.07369
	vcmlm	0.63547	0.03263	0.62190	0.04002	0.58465	0.07007
	vcmlm-LC	0.63277	0.03101	0.62547	0.03465	0.59006	0.04830
	vcmlm-LL	0.64327	0.03009	0.63339	0.03569	0.59839	0.05695

	avcmlm	0.59604	0.05388	0.58281	0.05956	0.57106	0.06906
	SMAP	0.64969	0.02529	0.63602	0.02974	0.59562	0.04823
	SMAP-LC	0.64474	0.02733	0.63386	0.03102	0.59725	0.04389
	SMAP-LL	0.65313	0.02453	0.64078	0.02844	0.60429	0.04457
	abSMAP	0.65098	0.02712	0.63985	0.02977	0.60570	0.04194
	oracle	0.71919	0.00000	0.71919	0.00000	0.71919	0.00000

Table 3 Regressors for vehicle silhouettes data and the model averaging weights for the marginal sub-models corresponding to the regressors. Weights are obtained from the SMAP, SMAP-LC, SMAP-LL and their standard errors (in parenthesis) calculated by the bootstrap resampling method.

Variable	Description	SMAP	SMAP-LC	SMAP-LL
X_1	Compactness	0.37890(0.20860)	0.40469(0.15359)	0.62439(0.16103)
X_2	Circularity	0.08666(0.15713)	0.00000(0.08501)	0.17702(0.15917)
X_3	Distance Circularity	0.00000(0.07033)	0.05675(0.13992)	0.00000(0.09438)
X_4	Radius ratio	0.00000(0.06750)	0.00000(0.04536)	0.05961(0.07477)
X_5	Pr.axis aspect ratio	0.00000(0.00662)	0.00000(0.00000)	0.00000(0.00000)
X_6	Max.length aspect ratio	0.07188(0.04776)	0.00000(0.00000)	0.00000(0.00000)
X_7	Scatter ratio	0.00245(0.18274)	0.00000(0.04284)	0.00000(0.03575)
X_8	Elongatedness	0.00000(0.10851)	0.24191(0.13796)	0.00000(0.07149)
X_9	Pr.axis rectangularity	0.00000(0.10482)	0.00000(0.06749)	0.00000(0.04444)
X_{10}	Max.length rectangularity	0.00000(0.10660)	0.08541(0.14100)	0.00000(0.10242)
X_{11}	Scaled variance along major	0.00000(0.04273)	0.00000(0.02652)	0.00000(0.02483)

Variable	Description	SMAP	SMAP-LC	SMAP-LL
	axis			
X_{12}	Scaled variance along minor axis	0.23923(0.16504)	0.00000(0.01222)	0.00000(0.00837)
X_{13}	Scaled radius of gyration	0.00000(0.02722)	0.00000(0.05266)	0.00000(0.02921)
X_{14}	Skewness about major axis	0.04079(0.02764)	0.14202(0.01878)	0.00000(0.00309)
X_{15}	Skewness about minor axis	0.07589(0.05670)	0.06921(0.09652)	0.04166(0.06717)
X_{16}	Kurtosis about minor axis	0.00000(0.00459)	0.00000(0.04943)	0.00000(0.04071)
X_{17}	Kurtosis about major axis	0.00000(0.04197)	0.00000(0.04032)	0.09731(0.07629)
X_{18}	Hollows ratio	0.10421(0.05975)	0.00000(0.01466)	0.00000(0.03174)

Table 4 The means, medians and sample standard deviations (sd) of hit rates (HR) for classifying the vehicle silhouettes data with different test sample sizes.

Method	n_{test} = 50			n_{test} = 100		
	mean	median	sd	mean	median	sd
ranfor	0.74570	0.74000	0.05805	0.75105	0.75000	0.04034
tree	0.70740	0.72000	0.06023	0.70720	0.71000	0.04496
svm	0.77210	0.78000	0.05734	0.76945	0.77000	0.03969
boost	0.77870	0.78000	0.06164	0.78080	0.78000	0.04048
mlm	0.80410	0.80000	0.06322	0.79895	0.80000	0.03646
mamlm	0.80420	0.80000	0.06295	0.79900	0.80000	0.03649
vcmlm	0.78740	0.78000	0.05481	0.80090	0.80000	0.04396
vcmlm-LC	0.84600	0.84000	0.04816	0.83545	0.84000	0.03411
vcmlm-LL	0.84400	0.84000	0.04799	0.83930	0.84000	0.03562
amlm	0.66020	0.66000	0.10049	0.72940	0.73000	0.06207
simlm	0.58680	0.60000	0.06581	0.60095	0.60000	0.05782
avcmlm	0.74770	0.76000	0.06536	0.76220	0.76000	0.04601
SMAP	0.82190	0.82000	0.05694	0.83505	0.83000	0.04019
SMAP-LC	0.83940	0.84000	0.05135	0.83280	0.83000	0.03939
SMAP-LL	0.83750	0.84000	0.05096	0.84055	0.84000	0.03608
abSMAP	0.83610	0.84000	0.05642	0.83945	0.84000	0.03469
Method	n_{test} = 200			n_{test} = 300		
	mean	median	sd	mean	median	sd
ranfor	0.75025	0.75000	0.02428	0.74697	0.75000	0.02139
tree	0.70132	0.70000	0.03251	0.69055	0.69167	0.02758
svm	0.76355	0.76500	0.02404	0.75245	0.75333	0.01962
boost	0.77692	0.77750	0.02614	0.77105	0.77000	0.02137

mlm	0.79250	0.79750	0.05899	0.72285	0.78667	0.17386
mamlm	0.79240	0.79750	0.05938	0.72283	0.78667	0.17409
vcmlm	0.80552	0.80500	0.02849	0.79535	0.80333	0.06061
vcmlm-LC	0.83210	0.83250	0.02538	0.82672	0.82667	0.02126
vcmlm-LL	0.83465	0.83500	0.02543	0.83150	0.83000	0.02027
amlm	0.75628	0.76000	0.04056	0.75820	0.76000	0.03276
simlm	0.60480	0.60500	0.04573	0.59633	0.59333	0.03734
avcmlm	0.76820	0.77000	0.03621	0.76073	0.76167	0.02663
SMAP	0.83325	0.83500	0.02758	0.79053	0.81333	0.10698
SMAP-LC	0.81175	0.82000	0.04397	0.79347	0.81333	0.07193
SMAP-LL	0.83908	0.84000	0.02518	0.83540	0.83667	0.02154
abSMAP	0.84038	0.84000	0.02276	0.83365	0.83333	0.02131