

Ending the War Between Model Selection and Combination: A Cross-Validation Approach

Elio Zhang[1(✉)] and Craig Rolling[2]

[1] Seattle University, Seattle, WA 98112, USA
ezhang2@seattleu.edu
[2] HNTB, Kansas City, MO 64105, USA
crolling@hntb.com

Abstract. Model averaging is a powerful alternative to model selection, often yielding improved prediction accuracy by combining the strengths of multiple models. However, in high-dimensional settings, the large number of candidate models poses serious challenges: many existing averaging methods become computationally infeasible, and indiscriminate combination can lead to poor performance. Furthermore, model averaging may underperform model selection when a single well-specified model exists. To address these challenges, we propose Cross-Validated Procedure Averaging (CVPA), a data-driven method that uses cross-validation to adaptively determine both which models to include and how many to combine. A key innovation of CVPA is its ability to select the optimal number of models in the ensemble, balancing variance reduction with robustness to model misspecification. When appropriate, CVPA reduces to model selection, providing a seamless bridge between selection and averaging. We establish theoretical guarantees for CVPA in high-dimensional linear regression, including selection consistency. Empirical studies further confirm its strong predictive performance across various settings with differing signal strengths and sparsity levels. Our results highlight the importance of controlling ensemble size and demonstrate that adaptive model combination—guided by cross-validation—is essential for reliable performance in high-dimensional applications.

Keywords: model averaging · forecast combination · cross-validation · high-dimensional prediction · risk minimization

1 Introduction

High-dimensional prediction problems arise in many modern applications, including genetics, genomics, and many other fields, where the number of potential predictors can be enormous while the sample size remains limited. In such settings, analysts often face multiple plausible modeling pipelines, differing in

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2027
O. Gervasi et al. (Eds.): ICCSA 2026, LNCS 16768, pp. 52–69, 2027.
https://doi.org/10.1007/978-3-032-30488-9_4

screening rules, tuning criteria, or selection methods, and predictive performance can depend strongly on which pipeline is used. Relying on a single selected model may therefore be unstable, yet combining many candidate models indiscriminately can reduce interpretability and may harm prediction. Motivated by this practical dilemma, we focus on two fundamental challenges for high-dimensional model combination.

Challenge 1: Selection vs. *Combination.* Model combination should not be applied blindly. In many scientific applications, identifying a small set of relevant features is as important as prediction (e.g., biomarker discovery), and averaging across many models can harm interpretability. Even from a purely predictive standpoint, if one candidate model adequately captures the data-generating process, combining it with inferior alternatives cannot improve and may even degrade performance [11]. In such cases, the optimal number of models to combine is one—that is, combination should reduce to model selection. A practical high-dimensional method should therefore adaptively decide, based on the observed data, whether selection or combination is preferred.

Challenge 2: Which and How Many Models to Combine. When combination is beneficial, the size of the candidate model space typically grows exponentially with the number of features p, making exhaustive averaging infeasible even for moderately large p [9]. In practice, analysts rarely attempt to average over all conceivable models; instead, they work with a finite pool of candidates chosen using scientific judgment, interpretability requirements, or computational feasibility. Classical approaches such as Bayesian model averaging [13] and frequentist model averaging [12] are designed to combine a prespecified collection of candidate models, but direct application is still prohibitive in high-dimensional regression when even that practical pool is large. Consequently, any viable procedure must include a screening mechanism to identify a manageable subset of models to combine [2,15]. Despite long-standing recognition of the importance of "selecting forecasts prior to combination," this problem remains unresolved in the forecast combination literature [18]. A key goal is therefore to determine the number of models to combine adaptively from the data, rather than fixing it *a priori*.

Limitations of Existing Work. Relative to model selection, model combination has received limited attention in the high-dimensional regression literature [2]. Moreover, existing methods primarily target the second challenge while treating selection and combination as separate tasks. For instance, [2] trims the model space by filtering features using a correlation test with a user-chosen p-value threshold, while [15,22] averages the best m models (with m fixed *a priori*) ranked by a criterion such as AIC. Such approaches reduce computation, but rely on ad hoc thresholds and do not provide a principled mechanism to choose between selection and combination.

Our Approach: Procedure Averaging Upstream. We address both challenges with a *method-centric* strategy that combines *procedures* rather than models. Here a

procedure refers to a modeling pipeline (algorithm and tuning/selection rule) that maps data to a fitted model—for example, CART [6], logistic regression, or penalized regression with distinct information criteria such as AIC [1] and BIC [16]. This distinction is important in practice: a model is the fitted predictive object ultimately used for prediction, while a procedure is the rule that produces such a model from the data. We use the labels "upstream" and "downstream" only as informal descriptors of *when* combination occurs in the workflow, not in the sense of abstraction layers. Conventional model averaging is performed *downstream*, meaning at the model level after a procedure has produced fitted models: a single preferred procedure is chosen first, then a collection of models produced by that procedure is averaged [2,15]. In contrast, we perform "procedure combination" *upstream*, meaning at the procedure level before the final model-level combination, by first assessing and selecting a set of qualified procedures and then combining the models they produce. Candidate procedures in this framework may include screening-based, threshold-based, or penalized-regression pipelines, so the method is compatible with common dimension-reduction practices rather than opposed to them.

Our method, Cross-Validated Procedure Averaging (CVPA), uses delete-n_v cross-validation to repeatedly select a qualified procedure from a candidate set. Here the validation set is simply the held-out portion of the data used to compare candidate procedures through an out-of-sample prediction criterion; throughout the paper, "validation" refers to this held-out predictive assessment step rather than to external outcome validation in an application-specific sense. Delete-n_v cross-validation is known to be consistent for selecting the risk-optimal procedure under suitable conditions [21,24]. Repeating the cross-validation B times yields B selected (not necessarily distinct) qualified procedures. Each selected procedure is then refit on the full dataset to produce a predictive model, and predictions for a new point are obtained by simple averaging across the resulting set of B models.

A distinctive feature of CVPA is that it adapts between selection and combination. If there exists a sufficiently dominant procedure in predictive risk, it will be selected by essentially all cross-validation repetitions, so CVPA reduces to model selection and combines a single model. When signal-to-noise is lower or misspecification is more prevalent, multiple procedures are selected across repetitions, and CVPA averages their corresponding fitted models. Thus, the number of models combined is estimated from the data rather than chosen *a priori*.

The rest of the paper is organized as follows. Section 2 develops the CVPA algorithm. Section 3 presents the theoretical properties of CVPA. Simulation results are reported in Sect. 4, and a real-data application is discussed in Sect. 5. Section 6 concludes with a discussion.

2 Proposed Method: Cross-validated Procedure Averaging (CVPA)

This section defines our high-dimensional regression framework and describes the CVPA algorithm to estimate the regression function.

2.1 Problem Setup and Definitions

Our work is done in a random-design regression setting, where

$$Y = f(\boldsymbol{X}) + \varepsilon.$$

In this framework, Y is the response, $\boldsymbol{X}$ is comprised of p features $(X_1, \cdots, X_p)$ with unknown distribution P_X having support $\mathcal{X}$, $f(\boldsymbol{X}) = E(Y|\boldsymbol{X})$ is the true regression function, and ε is the random error with $E(\varepsilon|\boldsymbol{X}) = 0$ and $E(\varepsilon^{4k}|\boldsymbol{X}) < \infty$ for a positive integer k. The data $\mathbf{Z} = (\mathbf{X}, \mathbf{Y})$ is comprised of n independent and identically distributed observations on $(X_1, \cdots, X_p, Y)$, where $\mathbf{Y}$ is an n-dimensional vector and $\mathbf{X}$ is an $n \times p$ feature matrix. For our theoretical results, we assume that $(\log p)/n \to 0$ as $n \to \infty$; in other words, p is not considered fixed but is allowed to grow with n at a controlled rate.

The true regression function f may be linear or nonlinear; however, as is typical in high-dimensional regression analysis, throughout this article only linear predictive models of the form

$$\widehat{f}(\boldsymbol{X}) = \sum_{j=1}^{p} \widehat{\beta}_j X_j \tag{1}$$

are considered as candidates. If the true regression function is linear,

$$f(\boldsymbol{X}) = E(Y|\boldsymbol{X}) = \sum_{j=1}^{p} \beta_j X_j,$$

then we define the "true model" M_0 as the set of variables whose regression coefficient $\beta_j \neq 0$. We formally define a linear modeling procedure as follows:

Definition 1. *A linear modeling procedure δ is a mapping from a sample $\mathbf{Z} = (\mathbf{X}, \mathbf{Y})$ to a p-dimensional vector $\widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z}) = \left(\widehat{\beta}_1(\delta, \mathbf{Z}), \cdots, \widehat{\beta}_p(\delta, \mathbf{Z})\right)^T$, i.e., $\mathbf{Z} :\to \widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z})$.*

For ease of notation, we sometimes denote $\widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z})$ by $\widehat{\boldsymbol{\beta}}_\delta$. A linear modeling procedure applied to $\mathbf{Z}$ results in a predictive model, defined as follows:

Definition 2. *A predictive model is a functional relationship between a set of inputs $\boldsymbol{X} = (X_1, \cdots, X_p)$ and a predicted outcome $\widehat{f}$.*

For example, the linear modeling procedure δ applied to $\mathbf{Z}$ produces the estimates $\widehat{\boldsymbol{\beta}}_\delta$, and the predictive model estimated by this modeling procedure is

$$\widehat{f}_\delta(\mathbf{X}^0) = \sum_{j=1}^{p} \widehat{\beta}_j(\delta, \mathbf{Z}) X_j^0$$

where $\mathbf{X}^0$ is a new observation on all p features and independent of $\mathbf{Z}$.

2.2 Procedure, Model and Feature Selection

In the setting of high-dimensional regression, there are on the order of billions of candidate models, so combining all of them is obviously infeasible. A filter step, which aims to select a set of "good" models, needs to be done before combination. Thus, model selection is an indispensable step in any method of model combination for high-dimensional regression. However, confusion can occur when discussing the distinct but intertwined concepts of model selection, procedure selection and feature selection in regression [24]. In this section, we clarify the differences and connections between these concepts before introducing our method.

In the general regression setting, a predictive model (1) is a mathematical function that is able to produce a point forecast $\widehat{f}$ for a given input $\boldsymbol{X} = \{X_1, \ldots, X_p\}$. The ultimate goal of our article is to develop and combine a set of such predictive models by making use of various selection and estimation methods. As soon as a predictive model is obtained, a set of active features (those whose values affect $\widehat{f}$ in the model) are defined. In other words, feature selection is a byproduct of model selection.

While feature selection is an important goal of many academic efforts, especially where questions of causation are concerned, optimal feature selection does not guarantee optimal model selection. To illustrate this point, consider the following example. Suppose $X_1 \sim N(0,1)$, $X_2 \sim N(0,1)$ and $\epsilon \sim N(0,1)$ are mutually independent. Let $Y = X_1 + X_2 + \epsilon$ and $X_3 = X_1 + 2X_2$. Then $\rho(Y, X_1) = 1/\sqrt{3}$, $\rho(Y, X_2) = 1/\sqrt{3}$, and $\rho(Y, X_3) = \sqrt{3/5}$. Thus, the variable X_3 is more strongly correlated with Y than X_1 or X_2. However,

$$E\big(Y - (X_1 + X_2)\big)^2 = 1 \tag{2}$$

$$E(Y - (\beta X_3))^2 = 1 + (1-\beta)^2 + (1-2\beta)^2 \tag{3}$$

where $1 + (1-\beta)^2 + (1-2\beta)^2 \geq 1.2$ (when $\beta = 0.6$). Thus, the model defined by (2), which does not use the most important feature X_3 in the feature set, is preferred to any model that uses only X_3. As exhibited by the example, a feature selection approach does not necessarily perform the best in model selection when the ultimate goal is prediction.

Moreover, model selection is fundamentally different from procedure selection. A modeling procedure is a method that is able to produce (1) from a sample of data. This definition implies that a general methodology that involves one or more tuning parameters is *not* a modeling procedure. For example, LASSO [17] as formulated as

$$\|Y - \sum_j X_j \beta_j\|_2^2 + \lambda \sum_{j=1}^{p} |\beta_j|, \tag{4}$$

when λ is unknown, is not a modeling procedure because (4) cannot produce a predictive model. However, for a given $\lambda = 2$, which is either selected by cross-validation or just chosen according to personal flavor, (4) becomes a modeling procedure. In other words, LASSO (4) represents a family of modeling procedures except when λ is known.

Although a modeling procedure should be evaluated by the predictive accuracy of the model selected by it, procedure selection should not be viewed as redundant or be equated to model selection. They are fundamentally different, and there is a clear-cut border between them. In the above sample, when we make a choice between two competing models $Y = X_1 + X_2 + \epsilon$ and $Y = 0.6X_3 + \epsilon$, we have to turn to a modeling procedure or model selection criterion such as AIC [1]. However, why not BIC [16]? Obviously, procedure selection has to be done first. In summary, if we want to develop a predictive model, then two steps are necessary. The first (upstream) step is defined as procedure selection in this paper. The second (downstream) step is model selection, which is achieved by applying the procedure selected in the first step to a set of sample data.

2.3 Cross-validated Procedure Averaging (CVPA) Algorithm

The example in Sect. 2.2 illustrates the disadvantage of methods originating from feature selection such as the method of [2], which ranks the correlation of each feature with Y and selects only features above some threshold. As for model selection, a single method is not the ultimate solution because various procedures such as AIC and BIC often select different models, and there are irreconcilable differences between them [20]. Overall, methods based on feature selection or model selection have severe limitations when the ultimate goal is prediction from high dimensional data. In our view, the most useful methods of selection or combination, generally speaking, are those that select or combine candidate *procedures*, each of which produces a model when applied to the data sample $\mathbf{Z}$. Thus our approach, which we describe next, uses a procedure selection approach to combine selected promising models. The most important characteristic of our proposed method is to select a set of promising modeling procedures first and then combine the models estimated by the set of "selected" procedures, which distinguishes our method from other model combination methods. As demonstrated below, this double-selection method (procedure selection followed by model selection) has substantial advantages over competing approaches.

Although our method of Cross-Validated Procedure Averaging (CVPA) is a very general idea that could be applied to many types of procedures and models, in this article we focus on the procedures of penalized linear regression that are most common in the study and practice of high-dimensional regression. Specifically, we consider modeling procedures that minimize an objective function $\mathcal{C}_{\delta,\mathbf{Z}}(\boldsymbol{\beta})$ in the form of

$$\mathcal{C}_{\delta,\mathbf{Z}}(\boldsymbol{\beta}) \equiv \frac{1}{n}\|\mathbf{X}\boldsymbol{\beta} - \mathbf{Y}\|_2^2 + P_\delta(\boldsymbol{\beta}), \tag{5}$$

where $P_\delta(\boldsymbol{\beta})$ is a positive penalty function on $\boldsymbol{\beta}$ serving to impose sparsity. For example, the methods of LASSO [17], SCAD [10], and MCP [23] are all modeling procedures that minimize an objective function of the form (5).

As mentioned in the introduction, the inherently large set of candidate models in high-dimensional regression creates two questions for model averaging:

which and how many models to combine? To answer these questions, Cross-Validated Procedure Averaging identifies a set of promising modeling procedures and uses them to select and estimate a set of predictive models to combine. The set of utilized procedures $\widehat{\mathbf{\Delta}}^*_{\mathcal{B}} = \{\widehat{\delta}^*_b; b = 1, \cdots, B\}$ is determined by applying procedure selection (via cross-validation) B times. This is akin to conducting selection by applying CV to each member of a collection of B pseudo-samples $\mathbf{Z}^*_{\mathcal{B}} = \{\mathbf{Z}^*_b; b = 1, \cdots, B\}$, each created from the original sample data $\mathbf{Z}$ by resampling without replacement (i.e., permutation). Because modeling procedure selection, when applied to different pseudo-samples, may select the same procedure, the number of predictive models to combine is determined by the number of unique procedures in $\widehat{\mathbf{\Delta}}^*_{\mathcal{B}}$ and thus is adaptive to the data. The set of promising procedures $\widehat{\mathbf{\Delta}}^*_{\mathcal{B}}$, which may consist of only one unique procedure, select and estimate a set of promising models $\widehat{f}_{\mathcal{B}} = \{\widehat{f}_1, \cdots, \widehat{f}_B\}$, which may consist of only one unique model. As a consequence, the CVPA procedure is a model combination method that includes model selection as a special case.

In summary, given a practitioner-specified set of candidate procedures $\mathbf{\Delta} = \{\delta_1, \cdots, \delta_D\}$, the Cross-Validated Procedure Averaging (CVPA) algorithm includes three steps. In applications, $\mathbf{\Delta}$ is intended to be a finite pool of scientifically and operationally reasonable procedures rather than an exhaustive list of all possible models:

1. *Selection: Apply the delete-n_v cross-validation to* $\mathbf{Z}$ *to select the best procedure* $\widehat{\delta}^*$. *Repeat the selection* B *times to get* B *procedures* $\widehat{\delta}^*_b$; $b = 1, \cdots, B$.
2. *Estimation: Apply* $\widehat{\mathbf{\Delta}}^*_{\mathcal{B}} = \{\delta^*_b; b = 1, \cdots, B\}$ *to* $\mathbf{Z}$ *to produce* B *predictive models,* $\{\widehat{\boldsymbol{\beta}}(\widehat{\delta}^*_b, \mathbf{Z}); b = 1, \cdots, B\}$.
3. *Combination: Given a new observation* $\mathbf{X}^0$, *the forecast of* Y^0 *by* CVPA *is*

$$\widehat{Y}^0_{\text{CVPA}} = \frac{1}{B} \sum_{b=1}^{B} (\mathbf{X}^0)^T \widehat{\boldsymbol{\beta}}(\widehat{\delta}^*_b, \mathbf{Z}). \tag{6}$$

In words, CVPA does repeated selection of modeling procedures and ultimately averages the selected models for prediction. Although the algorithm just described is relatively simple, to the best of our knowledge it has not been proposed before now in the literature. As we will show in the following section, the CVPA algorithm possesses attractive theoretical properties with regard to variable selection.

A simple example may help illustrate the criteria and rules used in practice. Suppose an analyst specifies the candidate set $\Delta = \{$lasso with AIC, lasso with BIC, MCP with BIC, SCAD with BIC$\}$. For one delete-n_v split, each candidate procedure is fit on the training portion and evaluated on the held-out validation portion using prediction error. The procedure with the smallest validation error is the winner for that split. Repeating this comparison B times produces a sequence of selected procedures. Each selected procedure is then refit on the full dataset, and the final CVPA prediction is the simple average of the resulting fitted-model predictions. In this way, the criterion is held-out prediction error,

the selection rule is "choose the procedure with the lowest validation error" within each repetition, and the combination rule is "average the predictions from the selected procedures across repetitions."

From an implementation perspective, the two main tuning choices in CVPA are the number of repetitions B and the validation size n_v in delete-n_v cross-validation. The role of B is to stabilize the empirical distribution of selected procedures: when B is too small, the final average may reflect Monte Carlo variability from only a few resamples, whereas larger B values give a clearer picture of whether the data support one dominant procedure or several competitive ones. The role of n_v is to control the bias-variance trade-off in procedure comparison. A larger validation set yields a more reliable out-of-sample comparison of candidate procedures, while a sufficiently large training set remains necessary so that each candidate procedure can still be fit well. Thus, in practice, B governs stability of the averaging step and n_v governs stability of the selection step.

It is worth noting that delete-n_v cross-validation plays an important role in our method. As shown in [24], when n_v satisfies certain conditions, delete-n_v CV is consistent for procedure selection, which in turn yields optimality properties for our proposed procedure.

The computational cost of CVPA is also transparent. If there are D candidate procedures, then each repetition requires comparing those D procedures on one training-validation split, and the final step refits the selected procedures on the full data. Hence the dominant cost is driven by B repeated procedure-selection problems, followed by B refits. This is more expensive than choosing a single procedure once, but it remains practical when the candidate collection is moderate, which is the intended regime of the method. Moreover, the B repetitions are embarrassingly parallel, so modern multicore or cluster environments can reduce wall-clock time substantially.

3 Properties of CVPA

3.1 CVPA for Prediction

The selected B procedures $\{\widehat{\delta}_1^*, \cdots, \widehat{\delta}_B^*\}$ are independent and identically distributed conditional on $\mathbf{Z}$. As a consequence,

$$E(\widehat{Y}^0_{\text{CVPA}} - Y^0)^2|\mathbf{Z} = \sigma^2 + Var\big(\widehat{Y}^0_{\text{CVPA}}|\mathbf{Z}\big) + \big(E(\widehat{Y}^0_{\text{CVPA}}|\mathbf{Z}) - \mu^0\big)^2$$

where $\mu^0 = E(Y^0)$ and $\sigma^2 = Var(Y^0)$.

$$Var\big(\widehat{Y}^0_{\text{CVPA}}|\mathbf{Z}\big) = \frac{1}{B} Var\big((\mathbf{X}^0)^T \widehat{\boldsymbol{\beta}}(\widehat{\delta}_b^*, \mathbf{Z})|\mathbf{Z}\big),$$

which indicates that CVPA accomplishes lower variance than a single selected procedure $\widehat{\delta}_b^*$. As a result, the method of Cross-Validated Procedure Averaging (CVPA) is guaranteed to have lower risk than selection. The favorable predictive performance of CVPA compared to other model selection and combination methods also is demonstrated through numerical studies in Sects. 4 and 5.

3.2 CVPA for Variable Selection

A key contribution of Cross-Validated Procedure Averaging (CVPA) is that it bridges the gap between model combination and model selection, so we focus on the selective performance of CVPA in this section. As shown by the equation (6), the prediction by CVPA typically is produced by the average of many different models. However, if one modeling procedure stands out and is selected by every pseudo-sample $\mathbf{Z}_b^*$, then $\widehat{\boldsymbol{\beta}}_b^* = \widehat{\boldsymbol{\beta}}(\widehat{\delta}_b^*, \mathbf{Z})$ will be the same for every b. In this case, the prediction $\widehat{Y}^0_{\text{CVPA}} = (\mathbf{X}^0)^T \widehat{\boldsymbol{\beta}}_1^*$ from CVPA is based on only one predictive model. In other words, model selection is performed and an active set of variables $\widehat{M}_0 \equiv \{j : \hat{\beta}_j \neq 0\}$ is identified by CVPA in this case. The following result shows that if the true regression function is linear with active set M_0, and if the best procedure among the candidates is able to consistently identify the true active set of variables, then CVPA also will perform consistent variable selection.

For a modeling procedure δ, let $\widehat{M}_\delta$ denote the active set in $\widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z})$ and $\widehat{M}_{\widehat{\delta}_b^*}$ denote the active set in $\widehat{\boldsymbol{\beta}}(\widehat{\delta}_b^*, \mathbf{Z})$. Define the risk of a modeling procedure δ as

$$R_n(\delta) = E\|\mathbf{X}^0 \widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z}) - Y^0\|_2^2, \tag{7}$$

where Y^0 represents a new observation on the response and the expectation is taken over the joint distribution of $\mathbf{Z}$ and $\mathbf{Z}^0 = ((\mathbf{X}^0)^T, Y^0)^T$. We use δ_0 to denote the procedure with the lowest risk $R(\delta)$ among the candidates in Δ, i.e.,

$$\delta_0 = \arg\min_{\delta \in \Delta} R_n(\delta) \tag{8}$$

For every b, a procedure is selected by delete-n_v CV [24], where n_v denotes the number used in cross-validation and n_t the number of observations used for parameter estimation with $n_v + n_t = n$. The delete-n_v CV error of a modeling procedure δ is calculated by

$$CV(\delta) = \sum_{i=1}^{n_v} E\big(\mathbf{X}_{v_i} \widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z}_t) - Y_{v_i}\big)^2 = \|\mathbf{X}_v \widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z}_t) - \mathbf{Y}_v\|_2^2, \tag{9}$$

where $\mathbf{Z}_t$ are the n_t observations allocated to the training set and $\{\mathbf{Z}_{v_i} : i = 1, \cdots, n_v\}$ are the n_v observations allocated to the validation set. In other words, the validation criterion used throughout is out-of-sample prediction error measured on held-out observations.

Theorem 1. *Suppose resampling without replacement and modeling procedure selection by delete-n_v CV are performed for each b and the following assumptions are satisfied:*

1. *For any $\delta \in \Delta$ such that $\delta \neq \delta_0$, $\liminf_{n\to\infty} n\big(R_n(\delta) - R_n(\delta_0)\big) \geq \xi$, where ξ is a positive constant;*
2. *$\lim_{n\to\infty} n_t/n_v = 0$;*
3. *The number of resamplings B satisfies $\lim_{n\to\infty} B(n_t/n_v)^k = 0$.*

Then $\lim_{n\to\infty} P\left(\cap_{b=1}^{B}\left\{\widehat{M}_{\widehat{\delta}_b^*} = \widehat{M}_{\delta_0}\right\}\right) = 1$.

Proof. First, by Bonferroni's inequality,

$$P\left(\cap_{b=1}^{B}\{\widehat{\delta}_b^* = \delta_0\}\right) \geq 1 - B\,P(\widehat{\delta}_b^* \neq \delta_0).$$

For any $\delta \neq \delta_0$, write $CV_b(\delta) = \|\mathbf{X}_v\widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z}_t) - \mathbf{Y}_v\|_2^2$ and let

$$R_{n_t}(\delta) = E\big(\mathbf{X}'_{v_1}\widehat{\boldsymbol{\beta}}(\delta, \mathbf{Z}_t) - Y_{v_1}\big)^2.$$

Then, by Markov's inequality,

$$\begin{aligned} & P(CV_b(\delta) < CV_b(\delta_0)) \\ \leq\ & \frac{E[CV_b(\delta_0) - CV_b(\delta) + n_v(R_{n_t}(\delta) - R_{n_t}(\delta_0))]^{2k}}{n_v^{2k}[R_{n_t}(\delta) - R_{n_t}(\delta_0)]^{2k}} \\ \leq\ & \frac{2E[CV_b(\delta_0) - n_v R_{n_t}(\delta_0)]^{2k} + 2E[CV_b(\delta) - n_v R_{n_t}(\delta)]^{2k}}{n_v^{2k}[R_{n_t}(\delta) - R_{n_t}(\delta_0)]^{2k}}. \end{aligned}$$

Using the same argument as in Theorem 1 of [21] together with Whittle's inequality [19], the numerator is of order $(n_v/n_t)^k$, while Assumption 1 bounds the denominator away from zero. Hence $P(\widehat{\delta}_b^* \neq \delta_0) = O((n_t/n_v)^k)$, so

$$P\left(\cap_{b=1}^{B}\{\widehat{\delta}_b^* = \delta_0\}\right) \geq 1 - O\big(B(n_t/n_v)^k\big) \to 1$$

by Assumption 3. Therefore $P\left(\cap_{b=1}^{B}\{\widehat{M}_{\widehat{\delta}_b^*} = \widehat{M}_{\delta_0}\}\right) \to 1$.

As implied by Theorem 1, if the best procedure δ_0 is able to consistently identify the true active set (that is, $P(\widehat{M}_{\delta_0} = M_0) \to 1$), then CVPA also performs consistent variable selection when appropriate values of B, n_t and n_v are used. Notably, even if the best procedure δ_0 is not able to consistently identify the true active set, CVPA still has a chance to do so. Indeed, the probability CVPA selects the true active set is at least

$$\begin{aligned} & P(\{\cap_{b=1}^{B}\widehat{M}_{\widehat{\delta}_b^*}\} = M_0) \\ & \geq P(\cap_{b=1}^{B}\{\widehat{M}_{\widehat{\delta}_b^*} = \widehat{M}_{\delta_0}\}; \widehat{M}_{\delta_0} = M_0) + P(\{\cap_{b=1}^{B}\widehat{M}_{\widehat{\delta}_b^*}\} = M_0; \widehat{M}_{\delta_0} \neq M_0), \end{aligned} \tag{10}$$

where the second term may be greater than zero in some cases, as shown by [3] and [14]. Consequently, CVPA may surpass model selection by a single procedure in terms of variable selection accuracy. This is also demonstrated in the simulation results of Sect. 4.2.

Assumptions 1 and 2 are standard in delete-n_v CV theory [21,24], and Assumption 3 ensures that the number of pseudo-samples does not grow too quickly. Together they imply that, when an outstanding procedure exists in Δ, CVPA inherits its selection consistency.

3.3 CVPA for Variable Selection in Practice

For practical variable selection, let

$$\widehat{p}_j^* = \frac{1}{B}\sum_{b=1}^{B} I_{\{\widehat{\beta}_j(\widehat{\delta}_b^*,\mathbf{Z})\neq 0\}}$$

denote the selection frequency of variable X_j across the B selected procedures. Theorem 1 implies that, if the best candidate δ_0 consistently identifies M_0, then $\widehat{p}_j^* \to 1$ for active variables and $\widehat{p}_j^* \to 0$ otherwise. In practice we select variables with $\widehat{p}_j^* \geq p_0$, as in related stability-based procedures [3,14]. The case $p_0 = 1$ is asymptotically consistent under Theorem 1, while smaller values such as 0.9 or 0.95 may be useful in finite samples. The results in Sect. 4.2 show that this approach can work well in practice.

Our proposed procedure results in two important advantages of CVPA relative to selection. First, the predictions from CVPA will have lower mean squared error than those from selection. Second, if there is an outstanding modeling procedure, CVPA will reduce to model selection.

4 Simulations

This section presents simulation studies that compare CVPA against other modeling procedures in high-dimensional linear regression settings with different degrees of sparsity and correlation between predictors.

We adopt the simulation settings of [2] and make some minor adjustments to accommodate a wider variety of patterns. Each row of the $n \times p$ design matrix $\mathbf{X} = (X_{i,j})$ is independently generated from $N(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{0}$ denotes the p-dimensional vector of 0 and the $p \times p$ covariance matrix $\mathbf{\Sigma} = (\rho^{|j_1 - j_2|})$, where j_1 and j_2 $(j_1, j_2 = 1, \cdots, p)$ are the row and column indices, respectively. Two values of ρ (0.5 and 0.9) are examined. The responses are generated from

$$Y_i = \sum_{j=1}^{p} \beta_j X_{i,j} + \varepsilon_i; \quad i = 1, \cdots, n \tag{11}$$

where $\varepsilon_i \sim N(0, 0.2^2)$. We let $n = 100$ and $p = 2000$. Among $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p)^T$, $q = 3$, 6 or 12 randomly selected coefficients are assigned nonzero values, which are generated by $N(0, 0.5^2)$ independently. The set of candidate procedures $\mathbf{\Delta}$ consists of LASSO, MCP and SCAD with the 100 different λ values that are the default candidates in the R package `ncvreg` [4], so $|\mathbf{\Delta}| = 300$. For each setting, 200 replications (realizations of sample data) are performed.

4.1 Prediction Accuracy

The following four methods are compared by their predictive performances:

(1) The Model selection method (denoted by SEL in the following figures and tables): Using delete-n_v CV to select the optimal procedure $\widehat{\delta}$ from $\mathbf{\Delta}$. Then apply $\widehat{\delta}$ to $\mathbf{Z} = (\mathbf{X}, \mathbf{Y})$ to develop the model for prediction.
(2) CVPA: Select the best procedure from $\mathbf{\Delta}$ by delete-n_v CV and perform the CV $B = n$ times. This generates B selected procedures $\widehat{\mathbf{\Delta}}^*_{\mathcal{B}} = (\widehat{\delta}^*_1, \cdots, \widehat{\delta}^*_B)$. Apply each procedure in $\widehat{\mathbf{\Delta}}^*_{\mathcal{B}}$ to $\mathbf{Z}$ to produce B predictive models and make predictions by (6).
(3) MCV2 [2]: In this method, candidate models are least-squares fits of the response on various subsets of predictors with similar marginal correlation with the outcome. Within each predictor subset, the number of regressors is determined by cross-validation. Then the candidate models are averaged, with the weights optimized by cross-validation and without the constraint that the weights sum to unity. [2] propose three algorithms, and MCV2 has the best overall performance of these in their simulation studies.
(4) Bagging (denoted by BAGG in the following figures and tables) [5]: Obtain $\widehat{\mathbf{\Delta}}^*_{\mathcal{B}}$ by the same method as above. For each $b = 1, \cdots, B$, apply $\widehat{\delta}^*_b$ to $\mathbf{Z}^*_b$ to yield B different predictive models, and make predictions by averaging the B predictions.

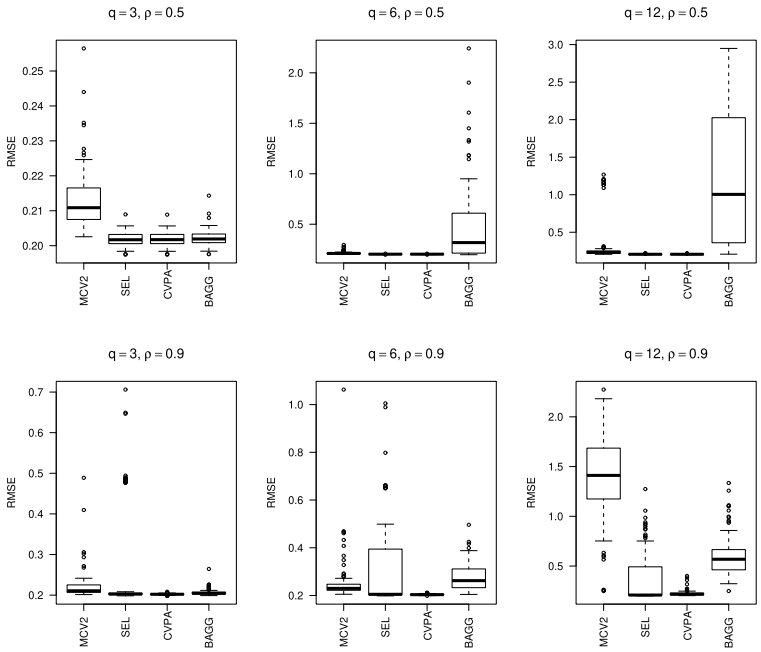

Fig. 1. RMSE of MCV2, selection (SEL), CVPA and bagging (BAGG). Simulation setting: $n = 100$, $p = 2000$; the q nonzero β_j are drawn from $N(0, 0.5^2)$. The feature matrix is generated as $N(\mathbf{0}, \mathbf{\Sigma})$ with $\mathbf{\Sigma} = \rho^{|j_1 - j_2|}$.

In this comparison, SEL, CVPA and BAGG all use delete-n_v cross-validation for tuning parameter λ selection where $n_v = n \log_{10} n/(\log_{10} n + 1)$ as suggested by [24]. The cross-validation error is calculated by (9). This choice makes the validation sample large enough to compare procedures on prediction performance while still leaving enough observations for estimation in each split. We also tried other forms of CV, such as 5-fold and 10-fold, and the relative results were similar. In MCV2, leave-one-out CV is employed to estimate the weights as suggested by [2]. For each competitor, we establish predictions $\widehat{Y}_k^0$ at 10,000 independent new observations $(\mathbf{X}_k^0, Y_k^0)$ from the same distribution as $(X_1, \cdots, X_p, Y)$ and the model (11). The RMSE $= \sqrt{\sum_{k=1}^{10000} \left(Y_k^0 - \widehat{Y}_k^0\right)^2 / 10000}$ for each competitor over the 200 replications is presented in Fig. 1.

As shown in Fig. 1, when the correlation among features is moderate ($\rho = 0.5$) and the true model is sparse ($q \leq 6$), CVPA and SEL outperform the other two methods in predictive accuracy. CVPA performs slightly better than SEL. As the true model size increases to $q = 12$, the advantage of CVPA over SEL becomes more pronounced due to the increased likelihood of model misidentification. The results for $q = 12$ are consistent with Figures 1–2 of [2], which show that, as the true model becomes dense, model selection methods can perform poorly by omitting important features. When the correlation among features is strong ($\rho = 0.9$), CVPA outperforms all competitors, and model selection becomes unstable even when the true model is sparse.

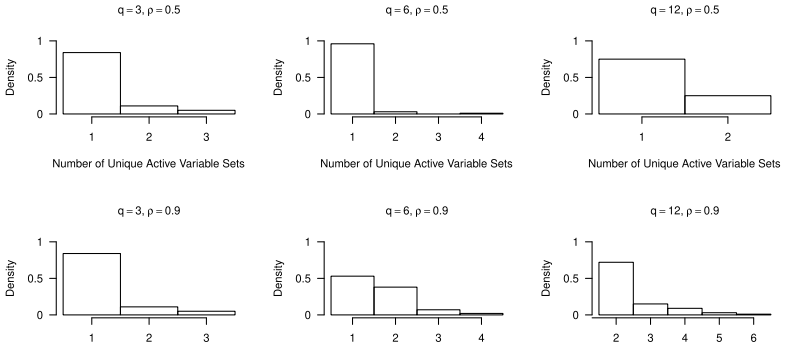

Fig. 2. Histogram of unique variable sets selected by CVPA. Simulation setting: $n = 100$, $p = 2000$. The q nonzero β_j are drawn from $N(0, 0.5^2)$, and the feature matrix is generated as $N(\mathbf{0}, \mathbf{\Sigma})$ with $\mathbf{\Sigma} = \rho^{|j_1 - j_2|}$.

As demonstrated by Fig. 2, when the correlation among features is moderate ($\rho = 0.5$) and the true model is sparse ($q = 3$), CVPA usually identifies a unique set of active variables for the regression. Moreover, as will be seen in Sect. 4.2, in this setting whenever CVPA identifies a unique active set (using $p_0 = 1$), the variables selected match those in the true model. As a consequence, CVPA and the model selection method (SEL) outperform all other methods in terms

of predictive accuracy. However, when the correlation between features is strong ($\rho = 0.9$), model misspecification becomes more likely. Therefore, the number of unique active variable sets combined by CVPA goes up with q to counter the worsening negative effect caused by omission of relevant features.

4.2 Selection Accuracy

In this subsection, the data are generated under the same setting as in Sect. 4.1, except that n=200, p=1000 and all nonzero regression coefficients are set to 1. Here we examine the performance of CVPA for variable selection. Table 1 compares the selective performance of CVPA to the model selection method (denoted by SEL) using three indices: the proportion of replications where the method selects the true active set (PROP), the average number of false positives (FP), and the average number of false negatives (FN). To implement variable selection by CVPA, we test three different threshold values ($p_0 = 0.9$, 0.95 and 1). As described in Sect. 3.3, an input variable showing up among the B models with frequency above p_0 is treated as a variable selected by CVPA.

Table 1. Comparison of selection accuracy for $n = 200$, $p = 1000$, and $B = 200$. Each nonzero $\beta_j = 1$, and $\Delta = \{$LASSO, MCP, and SCAD with 100 values of $\lambda\}$. The first column lists the true model size q. The other columns report the proportion of times the true active set is selected (PROP) and the average numbers of false negatives (FN) and false positives (FP).

q	SEL			CVPA, $p_0 = 0.9$			CVPA, $p_0 = 0.95$			CVPA, $p_0 = 1$		
	PROP	FN	FP	PROP	FN	FP	PROP	FN	FP	PROP	FN	FP
$\rho = 0.5$												
3	0.86	0	0.18	0.86	0	0.19	0.91	0	0.12	1	0	0
6	0.98	0	0.02	0.98	0	0.04	0.99	0	0.01	1	0	0
12	1.00	0	0.00	1.00	0	0.00	1.00	0	0.00	1	0	0
$\rho = 0.9$												
3	0.67	0.36	0.38	0.99	0.00	0.02	0.98	0.02	0.00	0.93	0.07	0.00
6	0.84	0.20	0.26	0.98	0.03	0.04	0.94	0.08	0.04	0.69	0.42	0.02
12	0.74	0.51	1.36	0.67	0.72	0.30	0.61	0.91	0.29	0.18	2.22	0.17

When the correlation among features is moderate ($\rho = 0.5$), both SEL and CVPA are able to identify the true model with high probability. When the correlation among features increases ($\rho = 0.9$), the frequency of selecting the true model decreases, but CVPA with $p_0 = 0.95$ achieves higher selective accuracy than SEL for $q = 3$ and $q = 6$. On the surface, it seems strange that CVPA would outperform SEL in variable selection, since CVPA is implemented by applying a selected procedure to a number of pseudo-samples. However, when a selected procedure is used to do model selection, the selection uncertainty

can be high in the sense that this procedure does not necessarily work well for another sample generated by the same population. CVPA averages out the selection uncertainty, which can improve both variable selection and prediction accuracy. The threshold p_0 controls how conservative this aggregation step is. Smaller values such as $p_0 = 0.9$ retain variables that appear frequently but not universally across the B selected models, which can reduce false negatives at the expense of admitting more false positives. Larger values such as $p_0 = 1$ keep only variables that appear in every selected model, which is more stringent and often preferable when a practitioner wants a highly stable final feature set. The intermediate choice $p_0 = 0.95$ balances these two goals and therefore works well in our simulations.

Overall, Cross-Validated Procedure Averaging (CVPA) outperforms other methods in terms of both prediction accuracy and variable selection and accomplishes adaptivity between selection and combination by mimicking whichever works better for the task at hand. In practice, $B = n$ repetitions of selections and $p_0 = 0.95$ are recommended for using CVPA when feature selection is a primary goal of the analysis. This recommendation is also computationally reasonable: using $B = n$ provides a stable estimate of selection frequencies without making the resampling burden excessively large for the moderate candidate sets considered here.

5 Real Data Application

The dataset `riboflavin` regarding vitamin B2 production is publicly available through the R package `hdi` [8]. The data is comprised of $N = 71$ observations on a single real-valued response variable, the logarithm of the B2 production rate, and $p = 4088$ predictors measuring the logarithm of the expression level of 4088 genes. The dataset is described in more detail in [7]. The goal of our analysis is to predict the logarithm of B2 production rate.

This dataset is a useful test bed for CVPA because it reflects the small-n, large-p regime that motivates the paper: the number of predictors is far larger than the sample size, procedure choice is consequential, and prediction is sensitive to how aggressively variables are screened or combined. In that sense, riboflavin is representative of the type of high-dimensional application in which neither pure model selection nor indiscriminate averaging is obviously satisfactory.

We evaluate and compare the performances of the same four procedures studied in Sect. 4 (SEL, CVPA, BAGG, and MCV2) in three steps. First, the $N = 71$ observations are divided into an evaluation set of size N_e ($N_e = 1, \cdots, 6$) and an estimation set of size $n = N - N_e$. Second, the four competitors each develop predictive models from the estimation set, which are used to make predictions on the evaluation set. Finally, for $N_e = 2, \cdots, 6$, we perform the above two steps 100 times, selecting the evaluation sets at random each time, and get 100 average prediction errors for each method. For $N_e = 1$, each record is used exactly once for evaluation, producing 71 prediction errors for each method. The average

root mean squared error (RMSE) for each of the four procedures are displayed in Table 2. In this representative example, CVPA features the best predictive accuracy among the competitors.

Table 2. For each method, the table shows the average root mean square error (RMSE) and standard error (SE) (in parentheses). Note that the SEs reported are not the standard errors in the i.i.d. case.

Comparison of Prediction Errors for Riboflavin Data				
N_e	MCV2	SEL	CVPA	BAGG
1	0.413 (0.036)	0.538 (0.041)	0.277 (0.022)	0.399 (0.032)
2	0.552 (0.050)	0.589 (0.038)	0.342 (0.022)	0.435 (0.031)
3	0.577 (0.039)	0.584 (0.034)	0.416 (0.028)	0.533 (0.029)
4	0.666 (0.045)	0.590 (0.026)	0.429 (0.022)	0.495 (0.024)
5	0.582 (0.040)	0.665 (0.026)	0.457 (0.021)	0.538 (0.022)
6	0.672 (0.051)	0.669 (0.024)	0.451 (0.023)	0.518 (0.019)

6 Discussion

In high-dimensional regression, many models may be plausible for prediction, making model selection and model combination central parts of the analysis. Model selection can yield an interpretable model and support valid inference when the model is correctly specified, but in high dimensions it often selects the wrong model, leading to poor prediction and unreliable conclusions. Model combination can improve predictive performance, but when the true model is sparse and included among the candidates, combining models may add unnecessary complexity with little—or even negative—benefit.

Our proposed method addresses this trade-off by using the data to adaptively choose between selection and combination. When the true model is linear and sparse, it consistently recovers the active set; when the true model is unlikely to be identified by selection alone, it remains competitive for prediction. In this sense, CVPA leverages the strengths of both selection and combination while mitigating their weaknesses.

This work also broadens the role of cross-validation beyond choosing a single best model. By using cross-validation to guide adaptive combination, our method can outperform a cross-validation-selected model in a range of settings (Sect. 4) and helps clarify how cross-validation can be used more effectively for prediction.

Finally, the idea of "procedure combination" appears promising for high-dimensional problems, and a natural next step is to extend it to broader machine-learning settings, including hybrid approaches that combine methods such as trees with parametric models like logistic regression.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Akaike, H.: A new look at the statistical model identification. IEEE Trans. Autom. Control **19**(6), 716–723 (1974)
2. Ando, T., Li, K.C.: A model-averaging approach for high-dimensional regression. J. Am. Stat. Assoc. **109**(505), 254–265 (2014)
3. Bach, F.R.: Bolasso: model consistent lasso estimation through the bootstrap. In: Proceedings of the 25th International Conference on Machine Learning. pp. 33–40. ACM (2008)
4. Breheny, P., Huang, J.: Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. Ann. Appl. Stat. **5**(1), 232–253 (2011)
5. Breiman, L.: Bagging predictors. Mach. Learn. **24**(2), 123–140 (1996)
6. Breiman, L.: Classification and Regression Trees. Routledge, Abingdon (2017)
7. Bühlmann, P., Kalisch, M., Meier, L.: High-dimensional statistics with a view toward applications in biology. Ann. Rev. Stat. Appl. **1**, 255–278 (2014)
8. Dezeure, R., Bühlmann, P., Meier, L., Meinshausen, N.: High-dimensional inference: confidence intervals, p-values and R-software HDI. Stat. Sci. **30**(4), 533–558 (2015)
9. Donoho, D.L.: High-dimensional data analysis: the curses and blessings of dimensionality. In: AMS Math Challenges Lecture, pp. 1–32 (2000)
10. Fan, J., Li, R.: Variable selection via nonconcave penalized likelihood and its oracle properties. J. Am. Stat. Assoc. **96**(456), 1348–1360 (2001)
11. Hibon, M., Evgeniou, T.: To combine or not to combine: selecting among forecasts and their combinations. Int. J. Forecast. **21**(1), 15–24 (2005)
12. Hjort, N.L., Claeskens, G.: Frequentist model average estimators. J. Am. Stat. Assoc. **98**(464), 879–899 (2003)
13. Hoeting, J.A., Madigan, D., Raftery, A.E., Volinsky, C.T.: Bayesian model averaging: a tutorial (with comments by m. clyde, david draper and ei george, and a rejoinder by the authors. Stat. Sci. **14**(4), 382–417 (1999)
14. Meinshausen, N., Bühlmann, P.: Stability selection. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **72**(4), 417–473 (2010)
15. Nan, Y., Yang, Y.: Variable selection diagnostics measures for high-dimensional regression. J. Comput. Graph. Stat. **23**(3), 636–656 (2014)
16. Schwarz, G.: Estimating the dimension of a model. Ann. Stat. **6**(2), 461–464 (1978)
17. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. Ser. B Stat Methodol. **58**(1), 267–288 (1996)
18. Wang, X., Hyndman, R.J., Li, F., Kang, Y.: Forecast combinations: an over 50-year review. Int. J. Forecast. (2022)
19. Whittle, P.: Bounds for the moments of linear and quadratic forms in independent variables. Theory Probabil. Appl. **5**(3), 302–305 (1960)
20. Yang, Y.: Can the strengths of AIC and BIC be shared? A conflict between model identification and regression estimation. Biometrika **92**(4), 937–950 (2005)
21. Yang, Y.: Consistency of cross validation for comparing regression procedures. Ann. Stat. **35**(6), 2450–2473 (2007)

22. Yuan, Z., Yang, Y.: Combining linear regression models: when and how? J. Am. Stat. Assoc. **100**(472), 1202–1214 (2005)
23. Zhang, C.H.: Nearly unbiased variable selection under minimax concave penalty. Ann. Stat. 894–942 (2010)
24. Zhang, Y., Yang, Y.: Cross-validation for selecting a model selection procedure. J. Econometr. **187**(1), 95–112 (2015)