Bayesian Analysis (2017) **12**, Number 3, pp. 807–829

A Bayes Interpretation of Stacking for $\mathcal{M}$-Complete and $\mathcal{M}$-Open Settings

Tri Le[*,‡] and Bertrand Clarke[†]

Abstract. In $\mathcal{M}$-open problems where no true model can be conceptualized, it is common to back off from modeling and merely seek good prediction. Even in $\mathcal{M}$-complete problems, taking a predictive approach can be very useful. Stacking is a model averaging procedure that gives a composite predictor by combining individual predictors from a list of models using weights that optimize a cross-validation criterion. We show that the stacking weights also asymptotically minimize a posterior expected loss. Hence we formally provide a Bayesian justification for cross-validation. Often the weights are constrained to be positive and sum to one. For greater generality, we omit the positivity constraint and relax the 'sum to one' constraint.

A key question is 'What predictors should be in the average?' We first verify that the stacking error depends only on the span of the models. Then we propose using bootstrap samples from the data to generate empirical basis elements that can be used to form models. We use this in two computed examples to give stacking predictors that are (i) data driven, (ii) optimal with respect to the number of component predictors, and (iii) optimal with respect to the weight each predictor gets.

Keywords: stacking, cross-validation, Bayes action, prediction, problem classes, optimization constrains.

1 Introduction

Stacking is a model averaging procedure for generating predictions first introduced by Wolpert (1992). The basic idea is that if J candidate signal plus noise models of the form $Y = f_j(x) + \epsilon$ for $j = 1, \dots, J$ are available then they can be usefully combined to give the predictor

$$\hat{Y}(x) = \sum_{j=1}^{J} \hat{w}_j \hat{f}_j(x),$$

where $\hat{f}_j$ is an estimate of f_j. Usually, $f_j(x) = f_j(x, \beta_j)$ so $\hat{f}_j(x) = f_j(x, \hat{\beta}_j)$ where $\hat{\beta}_j$ is an estimate of β_j. The weights $\hat{w} = (\hat{w}_1, \dots, \hat{w}_J)$ satisfy

$$\hat{w} = \arg\min_{w} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{J} w_j \hat{f}_{j,-i}(x_i) \right)^2 \tag{1}$$

[*] Department of Statistics, University of Nebraska-Lincoln, tle20@unl.edu

[†] Department of Statistics, University of Nebraska-Lincoln, bclarke3@unl.edu

[‡] Le gratefully acknowledges the extensive support provided by Holland Computing Center and Clarke gratefully acknowledges support from NSF grant # DMS-1419754.

DOI: 10.1214/16-BA1023

where $\hat{f}_{j,-i}$ is the estimate of f_j using the $n-1$ of the n data points by dropping the i-th one, i.e., $(x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n)$. The Y_i's are assumed independent and the x_i's are deterministic design points. Often the w_j's are assumed to be non-negative and sum to one. The properties of stacking as a predictor have been explored in numerous contexts such as regression Breiman (1996), Clarke (2003), Sill et al. (2009), classification and distance learning Ting and Witten (1999), Ozay and Vural (2012), density estimation Smyth and Wolpert (1999), and estimating bagging's error rate Rokach (2010), Wolpert and Macready (1999).

These earlier contributions treated stacking as a frequentist procedure. However, more recently, Clyde and Iversen (2013) brought stacking into the Bayesian paradigm. They recalled the tripartite partition of statistical problems into three classes namely $\mathcal{M}$-closed, $\mathcal{M}$-complete, $\mathcal{M}$-open, see Bernardo and Smith (2000), and suggested that outside the $\mathcal{M}$-closed setting the posterior risk could be approximated by a cross-validation (CV) error (for the same loss function). Hence the action minimizing the posterior risk could be approximated by the stacking predictor that minimizes (1). More precisely, given models M_j for $j = 1, \ldots, J$, a loss function ℓ, a vector of responses $\boldsymbol{Y} = \boldsymbol{y} = (y_1, \ldots, y_n)$, and an element $a(\boldsymbol{y})$ in the action appropriate for a collection of models, say $\mathcal{M}$, Clyde and Iversen (2013) used

$$\int \ell(y_{n+1}, a(\boldsymbol{y})) p(y_{n+1} \mid \boldsymbol{y}) \mathrm{d}y_{n+1} \approx \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, a(\boldsymbol{y}_{-i})) \tag{2}$$

in an $\mathcal{M}$-open context, where y_{n+1} represents a future outcome at a future design point x_{n+1}, $\boldsymbol{y}_{-i}$ is the data vector $\boldsymbol{y}$ with the i-th entry deleted, and $p(\cdot \mid \boldsymbol{y})$ is the predictive distribution for Y_{n+1}. Here and elsewhere, the design points $x_1, \ldots, x_{n+1}$ are suppressed in the notation unless consideration of them is essential for a step in a proof. Hence, Clyde and Iversen (2013) observed that minimizing the left hand side of (2) over $a(\boldsymbol{y})$ and the right hand side over $a(\boldsymbol{y}_{-i})$ leads to two actions that are asymptotically identical. Otherwise put, the stacking predictor is the asymptotic Bayes action for $\mathcal{M}$-complete problems. It is not the Bayes action in the $\mathcal{M}$-open case because the mode of convergence is undefined. Nevertheless, Clyde and Iversen (2013) used (2) in an $\mathcal{M}$-open context to good effect. It should be noted that (2) seems to have been initially conjectured in Bernardo and Smith (2000) and a non-cross-validatory version of (2) for individual models M_j, namely

$$E_{Y_{n+1}|\boldsymbol{Y}, M_j} \ell(Y_{n+1}, a_{M_j}(\boldsymbol{Y})) - \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, a_{M_j}(\boldsymbol{Y})) \xrightarrow{P} 0$$

is established in Walker and Gutierrez-Pena (1999) where $a_{M_j}(\boldsymbol{Y})$ is in the action space associated with M_j.

Aside from the applications of these results to the stacking predictor, the results – if proved formally as below – establish that leave-one-out CV is asymptotically a Bayes optimal procedure under some conditions. It can be verified that the proofs below extend to leave-k-out CV as well. That is, our results provide a Bayesian justification for using

CV as a way to choose a model from which to generate predictions outside of $\mathcal{M}$-closed problems.

For the sake of completeness, we recall that Bernardo and Smith (2000) define $\mathcal{M}$-closed problems as those for which a true model can be identified and written down but is one amongst finitely many models from which an analyst has to choose. By contrast, $\mathcal{M}$-complete problems are those in which a true model (sometimes called a belief model) exists but is inaccessible in the sense that even though it can be conceptualized it cannot be written down or at least cannot be used directly. Effectively this means that other surrogate models must be identified and used for inferential purposes. $\mathcal{M}$-open problems according to Bernardo and Smith (2000) are those problems where a true model exists but cannot be specified at all.

Here however, we make a stronger distinction between $\mathcal{M}$-complete and $\mathcal{M}$-open problems by taking the view that in the $\mathcal{M}$-open case no true model can even be conceptualized. Hence it is inappropriate to assume the existence of a true model. We prefer this stronger distinction because it ensures that $\mathcal{M}$-complete and $\mathcal{M}$-open are disjoint classes. In both $\mathcal{M}$-complete and $\mathcal{M}$-open classes the status of the prior is unclear because none of the models under consideration are taken is true. However, a weighting function ostensibly indistinguishable from a prior can be regarded as a sort of pseudo-belief in the sense that it is the weight one would pre-experimentally assign to the model if it were an action for predicting the outcomes of a data generator. More generally, the weights can only be interpreted as an index for a class of actions, provided the weighted combination of predictors from the J models is regarded as an action.

The three main contributions of this paper are (i) a formal proof that (2) holds for several loss functions in $\mathcal{M}$-complete settings, (ii) explicit formulae for the stacking weights for various choices of constraints on the w_j's, and (iii) a way to choose optimal basis expansions to stack so as to clarify the suggestion in Breiman (1996) that the models be chosen as different from each other as possible. In our examples, we choose two data generators, one $\mathcal{M}$-complete and one $\mathcal{M}$-open, to see how stacking performs.

The structure of this paper is as follows. In Section 2 we present the formal proof of using CV to approximate posterior risk and hence derive stacking as an approximation to the Bayes action. In Section 3, we use the approximation to the posterior expected risk to derive stacking weights under several sets of constraints on the weights, observing that relaxing the non-negativity and the sum to one constraints improves prediction. In Section 4, we show how to get optimal data-driven basis expansions to stack. These bases should be different from each other in the sense of being independent; orthogonality does not seem to be helpful. In Section 5, we give two real data examples where the $\mathcal{M}$-complete or $\mathcal{M}$-open assumption is reasonable. We use them to show the effect of the sum of the coefficients and to suggest desirable properties of basis element generation. Some concluding remarks are made in Section 6.

2 Approximating posterior risk

Let $\mathcal{M} = \{M_1, \ldots, M_J\}$ be a class of models and $\boldsymbol{y} = (y_1, \cdots, y_n)$ be the vector of outcomes of $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, where the Y_i's are independently distributed with probability density function (pdf) $p_j(y \mid \theta_j)$ for $j = 1\dots, J$ equipped with a prior $w_j(\theta_j)$. Consider a loss function $\ell : \mathbb{R} \times \mathcal{A} \to \mathbb{R}$ where $\mathcal{A}$ is the action space of a predictive decision problem. In this setting $\ell(y_{n+1}, a(\boldsymbol{y}))$ is the cost of taking action $a(\boldsymbol{y})$, where y_{n+1} is a future observation. Although the language of utility functions is more common in our context, we prefer the language of loss functions because it is more suggestive of decision theory. The posterior risk under model j is

$$\int \ell(y_{n+1}, a(\boldsymbol{y}))p_j(y_{n+1} \mid \boldsymbol{y})dy_{n+1}, \tag{3}$$

where $p_j(\cdot \mid \boldsymbol{y})$ is the predictive density from model j. Given a set of convex weights $\pi(j)$ for use over the models, the overall posterior risk is

$$\int \ell(y_{n+1}, a(\boldsymbol{y}))p(y_{n+1} \mid \boldsymbol{y})dy_{n+1}, \tag{4}$$

where $p(y_{n+1} \mid \boldsymbol{y})$ is the predictive density marginalizing out over j as well as the θ_j's.

The relationship between the notation in (1) and the above is that if $Y = f_j(x) + \epsilon$ we can write f_j in a generic parametric form $f_j(x) = f_j(x, \beta_j)$ so that $Y \sim p_j(y \mid \theta_j)$ means $Y \sim p_j(y \mid x, \theta_j)$ where θ_j is the concatenation of β_j and the parameters in the distribution of ϵ. We also assume without further comment that (i) the explanatory variable x and the parameter θ_j are of fixed dimension, and, for simplicity of notation, (ii) $(x, \theta_j) \in K_1 \times K_2$ where K_1 is a compact set in the space of explanatory variables and K_2 is a compact set. Strictly speaking, K_2 depends on j, but we assume that a single K_2 can be found and used for all j. This latter regularity condition can be relaxed at the cost of more notation. As a separate issue, because the Bayes predictors require integration over θ, the compactness of K_2 is only needed for the frequentist results.

In the results below we establish six versions of (2) using three different loss functions (squared error, absolute error, and logarithmic loss – also sometimes called a logarithmic scoring rule) and two different classes of predictor (Bayes and plug-in). Bayes predictors are of the form $E_j(Y_{n+1} \mid \boldsymbol{Y})$ and plug-in predictors are of the form $E_{\hat{\theta}_j} Y_{n+1}$ where $\hat{\theta}_j$ is an estimator of the true value of θ_j using $\boldsymbol{Y}$. To an extent the proofs of these results are similar: All of them use multiple steps of the form 'add and subtract the right extra terms, apply the triangle inequality, and bound the result term-by-term', and conclude by invoking a uniform integrability condition. One difference is that the results for the Bayes predictors invoke a martingale convergence theorem whereas plug-in predictors add an extra step based on the consistency of the $\hat{\theta}_j$'s.

We begin by giving conditions under which we can state and prove (2) for squared error and Bayes predictors; this provides formal justification for the methodology in Clyde and Iversen (2013) in $\mathcal{M}$-complete problems.

Theorem 2.1. *Let $\ell(z, a) = (z - a)^2$ denote squared error loss. Assume*

(i) For any $j = 1, \dots, J$ and any pre-assigned $\epsilon > 0$,

$$E_j(Y^{4+\epsilon}) = \int\int y^{4+\epsilon} p_j(y \mid x, \theta_j)w_j(\theta_j)d\theta_j dy < \infty,$$

and $E_j(Y^{4+\epsilon})$ is continuous for $x \in K_1$,

(ii) For each $j = 1, \ldots, J$, the conditional densities $p_j(y \mid x, \theta_j)$ are equicontinuous for $x \in K_1$ for each y and $\theta_j \in K_2$, and,

(iii) For each $j = 1, \ldots, J$, the Bayes predictor

$$\hat{Y}_j = E_j(Y_{n+1} \mid \boldsymbol{Y}) = \int\int y_{n+1} p_j(y_{n+1} \mid x_{n+1}, \theta_j) w_j(\theta_j \mid \boldsymbol{Y}) d\theta_j dy_{n+1}$$

is used to generate predictions at the $n+1$ step.

Then, for any action $a(\boldsymbol{Y}) = \sum_{j=1}^{J} w_j \hat{Y}_j$, $w_j \in \mathbb{R}$ for all j, we have

$$\int \ell(y_{n+1}, a(\boldsymbol{Y})) p(y_{n+1} \mid \boldsymbol{Y}) dy_{n+1} - \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, a(\boldsymbol{Y}_{-i})) \overset{L^2}{\to} 0 \text{ as } n \to \infty.$$

Proof. See Supplemental Appendix A (Le and Clarke, 2016). □

The result in Theorem 2.1 remains true for squared error loss if we use plug-in predictors of the form

$$\hat{Y}_j = E_{\hat{\theta}_j(\boldsymbol{Y})}(Y_{n+1}) = \int y_{n+1} p_{\hat{\theta}_j(\boldsymbol{Y})}(y_{n+1}) dy_{n+1},$$

where $\hat{\theta}_j$ is any consistent estimator for θ_j in M_j rather than Bayes predictors as in Assumption (iii). This assertion is in the following.

Theorem 2.2. *Let $\ell(z, a) = (z - a)^2$ denote squared error loss. Assume*

(i) For any $j = 1, \ldots, J$ and any pre-assigned $\epsilon > 0$,

$$E_j(Y^{4+\epsilon}) = \int\int y^{4+\epsilon} p_j(y \mid x, \theta_j) w_j(\theta_j) d\theta_j dy < \infty,$$

and $E_j(Y^{4+\epsilon})$ is continuous for $x \in K_1$,

(ii) For each $j = 1, \ldots, J$, the conditional densities $p_j(y \mid x, \theta_j)$ are equicontinuous for $x \in K_1$ for each y and $\theta_j \in K_2$, and,

(iii) For each $j = 1, \ldots, J$, let the plug-in predictor

$$\hat{Y}_j = E_{\hat{\theta}_j(\boldsymbol{Y})}(Y_{n+1}) = \int y_{n+1} p_{\hat{\theta}_j(\boldsymbol{Y})}(y_{n+1}) dy_{n+1}$$

be used to generate predictions at the $n+1$ step, where $\hat{\theta}_j(\boldsymbol{Y})$ is a consistent estimator for θ_j, and

(iv) For each $j = 1, \ldots, J$, $E_{\theta_j}(Y_{n+1})$ and $E_{\theta_j}(Y_{n+1}^4)$ are continuous for $\theta_j \in \Theta_j$, where $\Theta_j \subset K_2$ is a compact parameter space for M_j.

Then, for any action $a(\boldsymbol{Y}) = \sum_{j=1}^{J} w_j \hat{Y}_j$, $w_j \in \mathbb{R}$ *for all* j, *we have*

$$\int \ell(y_{n+1}, a(\boldsymbol{Y}))p(y_{n+1} \mid \boldsymbol{Y})dy_{n+1} - \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, a(\boldsymbol{Y}_{-i})) \overset{L^2}{\to} 0 \text{ as } n \to \infty.$$

Proof. See Supplemental Appendix B (Le and Clarke, 2016). □

Unsurprisingly, the conclusions of Theorems 2.1 and 2.2 continue to hold if the squared error is replaced by the absolute error $\ell(z, a) = |z - a|$. We state the Bayes and plug-in versions for absolute error in the following.

Theorem 2.3. *Let* $\ell(z, a) = |z - a|$ *denote absolute error loss. Assume*

(i) For any $j = 1, \dots, J$ *and any pre-assigned* $\epsilon > 0$,

$$E_j(Y^{2+\epsilon}) = \int\int y^{2+\epsilon} p_j(y \mid x, \theta_j)w_j(\theta_j)d\theta_j dy < \infty,$$

and $E_j(Y^{2+\epsilon})$ *is continuous for* $x \in K_1$,

(ii) For each $j = 1, \dots, J$, *the conditional densities* $p_j(y \mid x, \theta_j)$ *are equicontinuous for* $x \in K_1$ *for each* y *and* $\theta_j \in K_2$, *and,*

(iii) For each $j = 1, \dots, J$, *let either the Bayes or the plug-in predictor from Theorem 2.1 or Theorem 2.2, respectively, be used to generate predictions at the* $n + 1$ *time step.*

(iv) If plug-in predictors are chosen in Assumption (iii), then assume in addition that for each $j = 1, \dots, J$, $E_{\theta_j}(Y_{n+1})$ *and* $E_{\theta_j}(Y_{n+1}^2)$ *are continuous as functions of* $\theta_j \in \Theta_j$, *where* $\Theta_j \subset K_2$ *is a compact parameter space for* M_j.

Then, for any action $a(\boldsymbol{Y}) = \sum_{j=1}^{J} w_j \hat{Y}_j$, $w_j \in \mathbb{R}$ *for all* j, *we have*

$$\int \ell(y_{n+1}, a(\boldsymbol{Y}))p(y_{n+1} \mid \boldsymbol{Y})dy_{n+1} - \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, a(\boldsymbol{Y}_{-i})) \overset{L^2}{\to} 0 \text{ as } n \to \infty.$$

Proof. See Supplemental Appendix B (Le and Clarke, 2016). □

The log-loss is qualitatively different from squared error or absolute error because log-loss can be positive or negative. Nevertheless, we extend our results to the log-loss to verify that CV continues to remain an asymptotically Bayes procedure. Now, an action a is of the form

$$a(Y_{n+1} \mid \boldsymbol{Y}) = \sum_{j=1}^{J} w_j p_j(Y_{n+1} \mid \boldsymbol{Y}), \tag{5}$$

and the corresponding log-loss is

$$\ell(Y_{n+1}, a(Y_{n+1} \mid \boldsymbol{Y})) = -\log\left[\sum_{j=1}^{J} w_j p_j(Y_{n+1} \mid \boldsymbol{Y})\right]. \tag{6}$$

As shown in our next result, CV approximates the posterior expected loss of the Bayes action of the form (5) or the plug-in action of the form $a(Y_{n+1} \mid \boldsymbol{Y}) = \sum_{j=1}^{J} w_j p_{\hat{\theta}_j(\boldsymbol{Y})}(Y_{n+1})$ where $\hat{\theta}_j(\boldsymbol{Y})$ is a consistent estimator of $\theta_j \in K_2$ in M_j.

Theorem 2.4. *Let* $\ell(Y_{n+1}, a(\boldsymbol{Y})) = -\log[\sum_{j=1}^{J} w_j p_j(Y_{n+1} \mid \boldsymbol{Y})]$ *denote log-loss. Assume*

(i) For each $j = 1, \ldots, J$, *there is a function* $B_j(\cdot)$ *so that*

$$\sup_{\boldsymbol{Y}} |\log p_j(Y_{n+1} \mid \boldsymbol{Y})| \leq B_j(Y_{n+1}) < \infty,$$

$B_j(\cdot)$ *is independent of* $x_1, x_2, \ldots,$ *and*

$$E[g(Y_{n+1})] < \infty,$$

where

$$g(Y_{n+1}) = \max\left\{\left(\log \sum_{j=1}^{J} w_j e^{-B_j(Y_{n+1})}\right)^4, \left(\log \sum_{j=1}^{J} w_j e^{B_j(Y_{n+1})}\right)^4\right\}.$$

(ii) For each $j = 1, \ldots, J$, *the conditional densities* $p_j(y \mid x, \theta_j)$ *are equicontinuous in* x *for each* y *and* $\theta_j \in \Theta_j \subset K_2$, *and the predictive densities* $p_j(y \mid \boldsymbol{Y})$ *within the* j*-th model are uniformly equicontinuous in* y.

(iii) For each $j = 1, \ldots, J$, *let the Bayes action* (5) *be used to generate predictions at the* $n+1$ *time step.*

Then, we have

$$\int \ell(y_{n+1}, a(\boldsymbol{Y}))p(y_{n+1} \mid \boldsymbol{Y})dy_{n+1} - \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, a(\boldsymbol{Y}_{-i})) \overset{L^2}{\to} 0 \text{ as } n \to \infty.$$

If $p_{\hat{\theta}_j(\boldsymbol{Y})}(y_{n+1})$ *where* $\hat{\theta}_j(\boldsymbol{Y})$ *is a consistent estimator of* θ_j *is used instead of* $p_j(y_{n+1} \mid \boldsymbol{Y})$, *the result still holds.*

Proof. See Supplemental Appendix B (Le and Clarke, 2016). □

3 Derivation of stacking weights

Suppose we have J predictors $\hat{Y}_1, \cdots, \hat{Y}_J$ from distinct models. Then we might seek weights $\hat{w}_j$, using the training data, so as to form a model average prediction at x_{new} of the form

$$\hat{y}(x_{new}) = \sum_{j=1}^{J} \hat{w}_j \hat{y}_j(x_{new}). \tag{7}$$

From a Bayesian point of view, one should find the action that minimizes the posterior risk (or maximizes the posterior expected utility) given the data $\boldsymbol{y}$. Theorem 2.1 shows that the posterior risk is asymptotically equivalent to

$$\frac{1}{n}\sum_{i=1}^{n} \ell(y_i, a(\boldsymbol{y}_{-i})) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i}(x_i)\right)^2,$$

when ℓ is squared error loss. Ignoring the $(1/n)$ and minimizing over the $\hat{w}_j$'s gives the same expression as (1). That is, the stacking weights are asymptotically Bayes optimal – the precise form of optimality given by the constraints imposed on the w_j's – and can be used in (7) to give the stacking predictor. This formalizes the heuristic approximations used in Clyde and Iversen (2013).

Although the w_j's are often assumed to be positive and sum to one e.g., Clyde and Iversen (2013), Breiman (1996) only assumed the weights were positive and some remarks in Clyde and Iversen (2013) consider the case that the weights only satisfy a 'sum to one' constraint thereby permitting negative weights. We can see the effect of the sum to one constraint in a simple example. Following Clyde (2012), consider the two models $M_1 : Y = x_1\beta_1 + \epsilon$ and $M_2 : Y = x_2\beta_2 + \epsilon$ where the explanatory variables are orthogonal i.e., $x_1'x_2 = 0$. As shown in Supplemental Appendix C (Le and Clarke, 2016), if we stack these two models with the sum to one constraint we get $\hat{w}_1 = \hat{w}_2 = 1/2$. That is, predictions are generated from $Y_W = (1/2)x_1\hat{\beta}_1 + (1/2)x_2\hat{\beta}_2$ where the $\hat{\beta}_k$ are found from model M_k for $k = 1, 2$. On the other hand, if we stack M_1 and M_2 without the sum to one constraint but with, say, a sum to two constraint we get $Y_{WO} = x_1\hat{\beta}_1 + x_2\hat{\beta}_2 + \epsilon$, i.e., $\hat{w}_1 = \hat{w}_2 = 1$. Obviously, $Y_{WO} = 2Y_W$ so Y_W is half the size it should be. This extends to three or more models and shows that the sum to one constraint can be too restrictive. In addition, permitting w_j's to be negative increases the range of the stacking predictors and can only result in better predictions. Consequently in most of our results below we do not impose either the sum to one constraint or the non-negativity constraint. Indeed, removing the non-negativity constraint and relaxing the sum-to-one constraint to a sum-to-m constraint give the following.

Theorem 3.1. *The weights $w_1, \ldots, w_J$ achieving*

$$\min_{w} \sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i}(x_i)\right)^2 \quad \textit{subject to} \quad \sum_{j=1}^{J} w_j = m$$

are of the form

$$\hat{w} \propto U^{-1} 1_J,$$

where

$$\begin{aligned} U &= (u_{lj})_{J\times J}, \\ u_{lj} &= \sum_{i=1}^{n} \left(\frac{y_i}{m} - \hat{y}_{j,-i}\right) \hat{y}_{l,-i} - \sum_{i=1}^{n} \left(y_i - \hat{y}_{j,-i}\right) y_i, \qquad (8) \\ 1_J &= (1, \cdots, 1)'. \end{aligned}$$

Proof. This is a standard Lagrange multipliers problem. Write the Lagrangian as

$$L = -\sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i}\right)^2 - \lambda_0 (\sum_{j=1}^{J} w_j - m).$$

Then $\hat{w}$ is the solution of the following system,

$$\begin{aligned} \frac{\partial L}{\partial w_l} &= 2\sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i}\right) \hat{y}_{l,-i} - \lambda_0 = 0 \text{ for } l = 1, \cdots, J, \qquad (9) \\ \frac{\partial L}{\partial \lambda_0} &= \sum_{j=1}^{J} w_j - m = 0. \qquad (10) \end{aligned}$$

From (9) and (10), we have

$$\begin{aligned} &\sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i}\right) \hat{y}_{l,-i} = \frac{\lambda_0}{2} \\ &\Rightarrow \sum_{i=1}^{n} y_i \hat{y}_{j,-i} - \sum_{j=1}^{J} w_j \sum_{i=1}^{n} \hat{y}_{j,-i} \hat{y}_{l,-i} = \frac{\lambda_0}{2} \\ &\Rightarrow \frac{1}{m} \sum_{i=1}^{n} y_i \hat{y}_{j,-i} \sum_{j=1}^{J} w_j - \sum_{j=1}^{J} w_j \sum_{i=1}^{n} \hat{y}_{j,-i} \hat{y}_{l,-i} - \sum_{j=1}^{J} \sum_{i=1}^{n} (y_i - \hat{y}_{j,-i}) y_i w_j \\ &\quad = \frac{\lambda_0}{2} - \sum_{j=1}^{J} \sum_{i=1}^{n} (y_i - \hat{y}_{j,-i}) y_i w_j \quad \text{ for } l = 1, \cdots, J. \end{aligned}$$

Since the right hand side does not depend on l, we have

$$\frac{1}{m} \sum_{i=1}^{n} y_i \hat{y}_{j,-i} \sum_{j=1}^{J} w_j - \sum_{j=1}^{J} w_j \sum_{i=1}^{n} \hat{y}_{j,-i} \hat{y}_{l,-i} - \sum_{j=1}^{J} \sum_{i=1}^{n} (y_i - \hat{y}_{j,-i}) y_i w_j \propto 1.$$

Rearranging gives

$$\begin{aligned}
&w_1 \left(\frac{1}{m} \sum_{i=1}^{n} y_i \hat{y}_{l,-i} - \sum_{i=1}^{n} \hat{y}_{1,-i} \hat{y}_{l,-i} - \sum_{i=1}^{n} (y_i - \hat{y}_{1,-i}) y_i \right) \\
&+ w_2 \left(\frac{1}{m} \sum_{i=1}^{n} y_i \hat{y}_{l,-i} - \sum_{i=1}^{n} \hat{y}_{2,-i} \hat{y}_{l,-i} - \sum_{i=1}^{n} (y_i - \hat{y}_{2,-i}) y_i \right) \\
&\vdots \\
&+ w_J \left(\frac{1}{m} \sum_{i=1}^{n} y_i \hat{y}_{l,-i} - \sum_{i=1}^{n} \hat{y}_{J,-i} \hat{y}_{l,-i} - \sum_{i=1}^{n} (y_i - \hat{y}_{J,-i}) y_i \right) \propto 1,
\end{aligned}$$

for $l = 1, \cdots, J$.

In matrix form, this system of equations is

$$Uw \propto 1_J,$$

where U and 1_J are defined as in (8). Therefore, the solution is

$$\hat{w} \propto U^{-1} 1_J,$$

which can be rescaled to satisfy the sum to m constraint. □

Corollary 3.1. *If $m = 1$, then the weights $w_1, \ldots, w_J$ achieving*

$$\min_w \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i}(x_i) \right)^2 \text{ subject to } \sum_{j=1}^{J} w_j = 1$$

are of the form

$$\hat{w} \propto (\hat{e}'\hat{e})^{-1} 1_J, \tag{11}$$

where

$$\hat{e} = (y_i - \hat{y}_{j,-i})_{n \times J} \text{ and } 1_J = (1, \cdots, 1)'.$$

Remark 3.1. *This corollary is the result Clyde and Iversen (2013) used.*

For contrast, let us solve (1) but without any sum constraint (and without the non-negativity constraint). Now, the Lagrangian is

$$L = -\sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i} \right)^2,$$

and $\hat{w}$ is the solution of the system of equations

$$\frac{\partial L}{\partial w_l} \quad = \quad 2\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i}\right)\hat{y}_{l,-i} = 0 \text{ for } l = 1, \cdots, J.$$

Therefore,

$$\begin{aligned} &\sum_{i=1}^{n}\hat{y}_{l,-i}\sum_{j=1}^{J} w_j \hat{y}_{j,-i} = \sum_{i=1}^{n} y_i \hat{y}_{l,-i} \\ &\Leftrightarrow \sum_{j=1}^{J}\left(\sum_{i=1}^{n}\hat{y}_{l,-i}\hat{y}_{j,-i}\right) w_j = \sum_{i=1}^{n} y_i \hat{y}_{l,-i}, \text{ for } l = 1, \cdots, J, \end{aligned} \tag{12}$$

or, in matrix form,

$$Tw = c,$$

where

$$\begin{aligned} T &= \left(\sum_{i=1}^{n}\hat{y}_{l,-i}\hat{y}_{j,-i}\right)_{J\times J}, \\ c &= \left(\sum_{i=1}^{n} y_i \hat{y}_{1,-i}, \cdots, \sum_{i=1}^{n} y_i \hat{y}_{J,-i}\right)'. \end{aligned} \tag{13}$$

Hence the solution to (1) without the sum to one constraint and without the non-negativity constraint is

$$\hat{w} = T^{-1}c.$$

We summarize this in the following theorem.

Theorem 3.2. *The weights $w_1, \ldots, w_J$ achieving*

$$\min_{w}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{J} w_j \hat{y}_{j,-i}(x_i)\right)^2$$

are of the form

$$\hat{w} = T^{-1}c, \tag{14}$$

where T and c are given in (13). In addition, if the J predictors are orthonormal,

$$\sum_{i=1}^{n}\hat{y}_{l,-i}\hat{y}_{j,-i} = \delta_{l\neq j} \quad (1 \textit{ if } l \neq j \textit{ and } 0 \textit{ otherwise}),$$

then $T = I$ and the solution becomes

$$\hat{w}_j = \sum_{i=1}^{n} y_i \hat{y}_{j,-i}, \textit{ for } j = 1, \cdots, J. \tag{15}$$

Note that the minimum in Corollary 3.1 with the sum to one constraint is taken over a smaller set than that of Theorem 3.2 without any sum constraint. So, when the stacking weights from the two cases both exist, we expect the latter to give better predictive performance because the minimum in Theorem 3.2 can only be smaller than the minimum in Corollary 3.1. Hence we do not favor imposing the sum to one constraint. Indeed, we find in our computed examples that when a sum to one constraint gives better prediction, it is merely a happenstance from the more general optimization. This is straightforward because if we find the optimal weights from Theorem 3.2 then we can use them to find $m = \sum_{j=1}^{J} w_j$ for use in Theorem 3.1.

Using arguments similar to those used in the proof of Theorem 3.2, the following result extends Theorem 3.2 to a Hilbert space $\mathcal{H}$ equipped with an empirical inner product

$$\langle g, h \rangle_n = \frac{1}{n} \sum_{i=1}^{n} g(x_i) h(x_i) \quad \forall g, h \in \mathcal{H}.$$

Theorem 3.3. *The weights $w_1, \ldots, w_J$ achieving*

$$\min_{w} \sum_{i=1}^{n} \left(y(x_i) - \sum_{j=1}^{J} w_j \hat{f}_{j,-i}(x_i) \right)^2 ,$$

where y and $\hat{f}_{j,-i}$, $j = 1, \cdots, J$, belong to $\mathcal{H}$, are of the form

$$\hat{w} = T^{-1} c,$$

where T and c are of the same form as (13).

As $n \to \infty$, there are conditions that ensure the empirical inner product $\langle g, h \rangle_n$ converges uniformly to the inner product $\langle g, h \rangle = \int g(x) h(x) dx$ of the $\mathcal{H}$ space, see van de Geer (2014). Therefore, as n increases we can approximate the empirical inner product by the $\mathcal{H}$ inner product and the results in Theorem 3.3 will remain true.

4 What models should we put in the stack?

Here, we show that the intuition of Breiman (1996) that the models to be stacked should be as different as possible is only partially correct. In $\mathcal{M}$-complete problems what matters about the models to be stacked is that they be independent. The extra 'difference' amongst models from imposing orthogonality is not actually helpful in terms of reducing the error criterion (1). We show this for models constructed in general Hilbert spaces of functions and then provide one possible answer for how to construct the models in a Hilbert space to be stacked. By contrast, in $\mathcal{M}$-open problems one cannot assume the regression function is in a Hilbert space. In our $\mathcal{M}$-open example in Section 5.2 we did not find orthonormality of a basis gave better predictions than merely requiring independence but this need not hold in general. That is, in general, using models that are different but not necessarily independent give asymptotically optimal performance. This is seen by analogy in an example from Minka (2002). He uses a finite model list for an $\mathcal{M}$-complete problem and finds that distinct models that are not independent can give asymptotically optimal performance. By the same logic it is reasonable to conjecture that independence will likewise not help in $\mathcal{M}$-open problems, let alone orthogonality. That is, in $\mathcal{M}$-open problems, it is likely enough in general for models to be genuinely different and for independence or orthogonality not to be useful.

4.1 The error depends only on the span of the model list

In the last section, we saw that releasing the sum to one constraint can only reduce the error criterion and from our example in Section 3 we saw that this constraint can often be genuinely harmful. This argument is particularly strong outside $\mathcal{M}$-closed settings where model mis-specification is always present.

Our first result shows that given a set of models to stack in an $\mathcal{M}$-complete problem, the error depends only on the span of the models; requiring that the models to be stacked be orthogonal as well as independent does not reduce the error. Our result is the following.

Theorem 4.1. *Let $\mathcal{H}$ be a Hilbert space with inner product denoted $\langle \cdot, \cdot \rangle$. Let $\mathcal{M} = \{f_1, \cdots, f_J\}$ and $\mathcal{M}' = \{f'_1, \cdots, f'_{J'}\}$ be sets of elements from $\mathcal{H}$ with minima $Q^{\mathcal{M}}_{\min}$ and $Q^{\mathcal{M}'}_{\min}$ for (1), respectively. Denote the span of a set of elements in $\mathcal{H}$ by $\langle \cdot \rangle$. Then, if $\langle \mathcal{M} \rangle = \langle \mathcal{M}' \rangle$,*

$$Q^{\mathcal{M}}_{\min} = Q^{\mathcal{M}'}_{\min},$$

i.e., the stacking error only depends on the span of the predictors.

Proof. This involves routine manipulations with Hilbert spaces, see Supplemental Appendix C (Le and Clarke, 2016) for details. □

Theorem 4.1 means that given a fixed subspace $S \subset \mathcal{H}$, any basis for S is as good as any other for forming a stacking predictor. So, we are free to choose whichever basis is most convenient. In Section 4.2 we will choose f_j's through bootstrapping function estimators that lie in a Hilbert space. Consequently, Theorem 4.1 holds when the stacking weights, the w_j's, are constrained to be positive. If a given w_j is negative, an extra negative sign can be put on the function estimator without changing the span of the model list. In Section 5, we will show that the behavior of the sum of the w_j's can change character when they are required to be non-negative.

Note that Theorem 4.1 does not hold in the presence of the 'sum to one' constraint on the w_j's. Indeed, in this case the conclusion may be false. Let $J = J'$, $\mathcal{M} = \{\hat{y}_j = (\hat{y}_{j,-1}, \cdots, \hat{y}_{j,-n})', j = 1, \cdots, J\}$ be an orthogonal basis, and $\mathcal{M}' = \{\hat{y}'_j = (\hat{y}'_{j,-1}, \cdots, \hat{y}'_{j,-n})', j = 1, \cdots, J\}$ be any basis of $\langle \mathcal{M}' \rangle = \langle \mathcal{M} \rangle$. Then, under the sum to one constraint on $\mathcal{M}$ we have

$$\begin{aligned} Q^{\mathcal{M}}_{\min} &= \left\| y - \sum_{j=1}^{J} \hat{w}_j \hat{y}_j \right\|^2 \\ &= \left\| \sum_{j=1}^{J} \langle y, \hat{y}_j \rangle \hat{y}_j + \sum_{j=J+1}^{n} \langle y, e_j \rangle e_j - \sum_{j=1}^{J} \hat{w}_j \hat{y}_j \right\|^2 \\ &= \left\| \sum_{j=1}^{J} \left(\langle y, \hat{y}_j \rangle - \hat{w}_j \right) \hat{y}_j \right\|^2 + \|y_2\|^2, \end{aligned} \tag{16}$$

where $\hat{w}$ is now the solution in Corollary 3.1, $\{e_j, j = J+1, \cdots, n\}$ are complement vectors of $\{\hat{y}_j, j = 1, \cdots, J\}$ to form an orthonormal basis of $\mathbb{R}^n$, and $y = y_1 + y_2 = \sum_{j=1}^{J} \langle y, \hat{y}_j \rangle \hat{y}_j + \sum_{j=J+1}^{n} \langle y, e_j \rangle e_j$. Similarly, for $\mathcal{M}'$ we have

$$Q^{\mathcal{M}'}_{\min} = \left\| \sum_{j=1}^{J} \left(\alpha_j - \hat{w}'_j \right) \hat{y}'_j \right\|^2 + \|y_2\|^2, \tag{17}$$

where $\hat{w}'$ is the solution in Corollary 3.1 and $y = y_1 + y_2 = \sum_{j=1}^{J} \alpha_j \hat{y}'_j + \sum_{j=J+1}^{n} \langle y, e_j \rangle e_j$. Obviously, from (16) and (17), it is possible for $Q^{\mathcal{M}}_{\min} < Q^{\mathcal{M}'}_{\min}$ or $Q^{\mathcal{M}}_{\min} > Q^{\mathcal{M}'}_{\min}$. This can be seen from the following example. Let $J = J' = 1$ and $\hat{y}' = k\hat{y}$, then $\hat{w} = \hat{w}' = 1$ and $\alpha = \langle y, \hat{y} \rangle / k$. Hence $Q^{\mathcal{M}}_{\min} = (\langle y, \hat{y} \rangle - 1)^2 + \|y_2\|^2$ and $Q^{\mathcal{M}'}_{\min} = (\langle y, \hat{y} \rangle - k)^2 + \|y_2\|^2$. So, by careful choice of k, $Q^{\mathcal{M}}_{\min}$ can be larger than $Q^{\mathcal{M}'}_{\min}$ or the reverse.

To reinforce Theorem 4.1, we observe that reducing the dimension of the span of the predictors can only increase the error criterion.

Theorem 4.2. *Let $\mathcal{M} = \{f_1, \cdots, f_J\}$ be a basis and $\mathcal{N} = \{f_1, \cdots, f_{J-1}\}$. Let $Q^{\mathcal{M}}_{\min}$ and $Q^{\mathcal{N}}_{\min}$ be the minima of (1) corresponding to $\mathcal{M}$ and $\mathcal{N}$, respectively. Then,*

$$Q^{\mathcal{M}}_{\min} \leq Q^{\mathcal{N}}_{\min}. \tag{18}$$

Proof. This involves relatively routine manipulations with Hilbert spaces, see Supplemental Appendix C (Le and Clarke, 2016) for details. □

Taken together, Theorem 4.1 and Theorem 4.2 tell us that the predictors being stacked should be different from one another in the sense of being independent (but not necessarily orthogonal) and that the stacking error (1) is a non-increasing function of the span of the predictors. Thus, when choosing predictors to stack, there is a tradeoff between the number of predictors and their proximity to a true model assuming one exists. That is, using more predictors will generally be helpful, but using fewer, better predictors can easily outperform many, weaker predictors.

4.2 Optimal choice of predictors to stack

Having seen that both the number of basis vectors and the proximity of a linear combination of them to a true function (if a true function exists) can be important we want to choose the basis vectors effectively. The results in Section 4.1 mean that, without loss of generality, we can limit our search to orthogonal bases. Hence, in this subsection, we propose a data-driven method to choose an optimal number of basis vectors even if the set of basis vectors is not unique.

Assume we have an orthonormal basis for a space $\langle\{e_1, \cdots, e_J\}\rangle$ then for each $J' \leq J$ we can form

$$\hat{y}_{J'}(x_{\sigma_k(i)}) = \sum_{j=1}^{J'} \langle e_j, \hat{f}(\cdot \mid x_{\sigma_k(1)}, \cdots, x_{\sigma_k(i-1)})\rangle e_j(x_{\sigma_k(i)}), \tag{19}$$

where σ_k for $k = 1, \ldots, K$ is a collection of independently chosen permutations of $\{1, \cdots, n\}$ and $\hat{f}$ is an estimate of the true predictor. For instance $\hat{f}$ may be a Nadaraya–Watson estimator of the form $f_{\hat{\lambda}}$, see Nadaraya (1964) and Watson (1964), in which $\hat{\lambda}$ is an estimate of the tuning parameter λ. Then, in principle, we can find, for instance,

$$\begin{aligned} &\{J_{\mathsf{opt}}, basis_{\mathsf{opt}}\} \\ &= \arg \min_{J', basis} \sum_{k=1}^{K} \sum_{i=1}^{n} \left(\hat{y}_{J',\hat{\lambda}}(x_{\sigma_k(i)}) - y(x_{\sigma_k(i)}) \right)^2, \end{aligned} \tag{20}$$

where *basis* is a variable varying over the possible orthonormal bases for subspaces of $\langle\{e_1, \cdots, e_J\}\rangle$ and $\hat{y} = \hat{y}_{J',\hat{\lambda}}$ is formed using the Nadaraya–Watson estimator.

The idea is that (20) is a sort of variance-bias expression that can be minimized to find the right number of basis vectors. Minimizing in (20) means we are preventing J_{opt} from being too high or too low. If J' is too low then the prediction is biased and can be improved by adding more basis vectors. If J' is too large then the variability of the prediction is too high and can be improved by removing the basis vectors that are least important to good prediction. While this variance-bias approach is commonly done in non-sequential settings, here we find it equally useful in sequential settings.

Expression (20) gives an approximate solution to this variance-bias optimization because it uses K independent orderings of the data and sequential predictive error (since we want $\hat{y}_{J',\hat{\lambda}}(x_{\sigma_k(i)})$ at stage $\sigma_k(i)$ to use only the data $x_{\sigma_k(1)}, \cdots, x_{\sigma_k(i-1)}$). Averaging over the permutations of the data points as they appear in the sequential predictive error means that it is reasonable to regard the empirical optimum as close to an actual optimum if it exists (the $\mathcal{M}$-complete case). In $\mathcal{M}$-open settings, it does not make sense to take limits of (20), so it is hard to prove theory. What is feasible is to seek a $\{J_{\mathsf{opt}}, basis_{\mathsf{opt}}\}$ that makes (20) small. This can be done numerically by a stochastic search provided we have a way to propose basis vectors.

Our method for data-driven random generation of basis vectors is simple. Choose J sufficiently large and draw B bootstrap samples from the data set of size n. For each bootstrap sample, define a basis vector using the Nadaraya–Watson estimator. This gives B candidates for basis vectors. In some cases, two or more of these basis vectors may be so close as to be de facto the same. When this occurs, we reject one of the basis vectors and repeat the procedure until we have J basis vectors that are satisfactorily different and apply Gram–Schmidt orthonormalization to form an orthonormal basis. Note that any technique for nonparametric regression can be used in place of Nadaraya–Watson. In Section 5 we also use Gaussian process priors, see Rasmussen and Williams (2006), to generate function estimates that can be used as basis vectors.

Next let J_{opt} be the optimal value from (20). We choose J_{opt} orthonormal basis vectors from the J orthonormal basis vectors by ordering them in terms of decreasing size of the absolute values of their Fourier coefficients with a full-data set estimate of f. More formally, let $\hat{f}$ be an estimate of f such as Nadaraya–Watson or Gaussian process priors using all the data i.e., not just a bootstrap sample, and write $\hat{e}_j$ for $j = 1, \ldots, J$ for the J empirical orthonormal basis vectors. To form a predictor we stack the J_{opt} basis vectors with the largest values of $| \langle \hat{e}_j, \hat{f} \rangle |$ for $j = 1, \ldots, J$. When we use this procedure in Section 5 we have several explanatory variables so rather than using multidimensional Fourier expansions we impose an additive model structure and form a stacking predictor from each variable individually.

5 Computed examples

In this section we apply our technique described in previous sections to one $\mathcal{M}$-complete data set and one $\mathcal{M}$-open data set. The first is a 'canned' data set that is recognized to be difficult. The second is a new data set on soil moisture graciously provided by Prof. T. Franz, see Franz et al. (2015).

5.1 Forest Fires data

Consider the Forest Fires data set publicly available from the UC Irvine Machine Learning Reposition. The sample size is $n = 517$ and there are 8 non-trivial explanatory variables, $X_1, \ldots, X_8$, related to the severity of a forest fire. The dependent variable Y is the burn area of the fire. Details and references can be found at http://archive.ics.uci.edu/ml/datasets/Forest+Fires. We regard the Forest Fires data set as $\mathcal{M}$-complete because a forest fire is a chemical reaction with a lot of randomness that cannot be quantified well. That is, there is so little about the process that is stable that it is unclear there is anything to estimate. However, it is plausible that there is a model, necessarily highly complex, that might accurately encapsulate the behavior of forest fires under a variety of environmental conditions. Of course, such a model could be so complex that even though the data generator is $\mathcal{M}$-complete it is nearly $\mathcal{M}$-open. Thus, generating predictions may be the most appropriate approach even if they have a high variability.

For our analysis, we divide the data randomly into two subsets, one for training and one for validation. The training set contains $n_1 = 267$ data points and the validation set contains $n_2 = 250$ data points. To generate a predictor, we assume an additive model

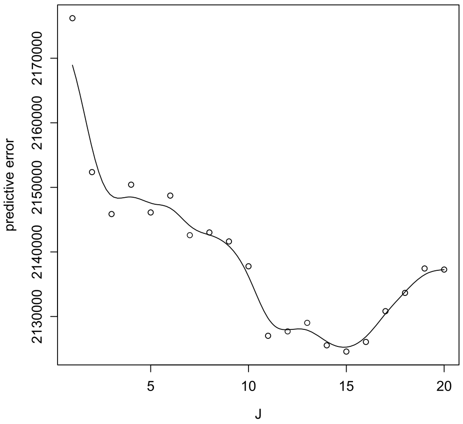


Figure 1: The individual circles indicate the values of (20) for X_1 with $J = 1, \dots, 20$, $K = 40$ and the Nadaraya–Watson estimator. The smooth curve is the result of using the Nadaraya–Watson smoother again for the 20 values of (20). To generate this graph we used a burn-in of 200 data points to ensure that the predictors from (19) would be well-defined in the sense that the predictions from the Nadaraya–Watson estimator for points relatively far from the accumulated data would not be so small as to lead to underflow problems.

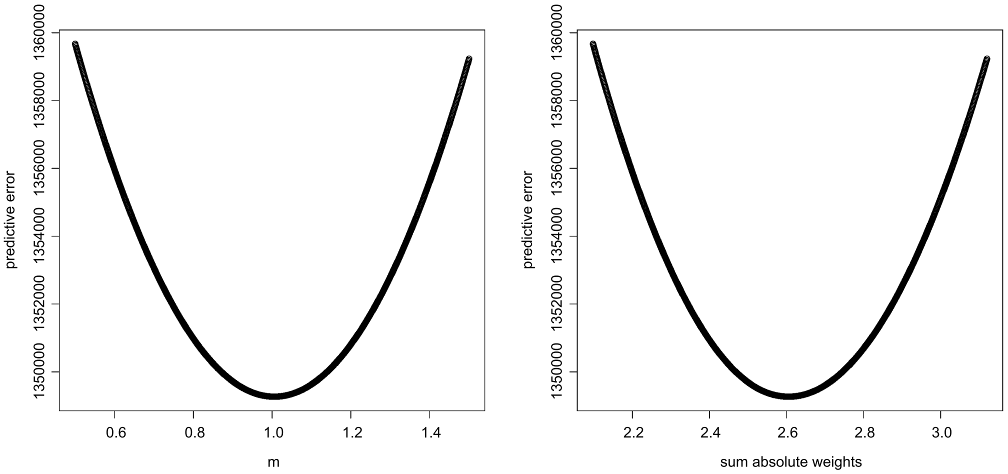
structure to generate predictions. That is, we write

$$Y = A_1(X_1) + \cdots + A_8(X_8) + \epsilon, \tag{21}$$

where ϵ is mean zero noise. So, we form eight univariate models $\hat{A}_u$ by finding $J_{\mathsf{opt}} = J_{\mathsf{opt},u}$ basis vectors for each u and stacking them. The result is $\hat{A}_u(X_u)$. Then we stack the eight univariate terms $\hat{A}_1, \dots, \hat{A}_8$ to generate a predictor for Y. So, we use stacking for both the individual functions in the additive model and to combine the individual functions to form the additive predictor.

We consider two classes of data-driven basis vectors. The first class is generated as discussed in Section 4.2 using the Nadaraya–Watson estimator. This was found using the `npreg()` function in `R`. The second class is generated using Gaussian process priors found using the `gausspr()` function in `R`. In both cases we used the default settings for the `R` functions, e.g., cross-validation to find $\hat{\lambda}$ and radial basis functions to define the kernel in the Gaussian process prior, and we set the number of basis vectors to find to be considered to be $J = 1, \dots, 20$. This range was sufficiently large that it contained J_{opt} as an interior point in all cases. The value of J_{opt} depends on which variable X_u was being used and both stacking procedures were done independently of the m in the 'sum to m' constraint.

As an illustration, Figure 1 shows how we found $J_{\mathsf{opt}} = 15$ for X_1. The minimum sequential predictive error occurs for 15 basis vectors. Similar plots were made for $X_2, \dots, X_8$ and led to values $12, 12, 13, 9, 13, 7$, and 9 respectively. Accordingly, we



(a) (b)

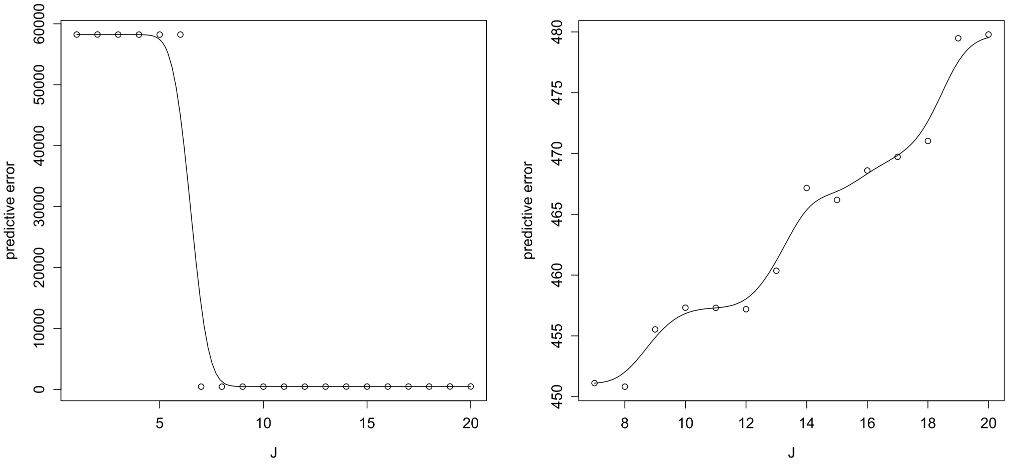
Figure 2: Plots of cumulative predictive error of stacking eight univariate predictors versus the stacking weights for the $\hat{A}_1, \ldots, \hat{A}_8$ for the Forest Fires data, with basis vectors generated by Nadaraya–Watson. Left: Cumulative predictive error on the test set versus the sum of the stacking weights, m. Right: Cumulative predictive error on the test set versus the sum of the absolute value of the stacking weights.

stacked eight univariate models that were themselves the result of stacking basis vectors found via a variance-bias trade off for each explanatory variable. Note that we could generate a figure similar to Figure 1 using Gaussian process priors. However, the differences in our examples due to using Nadaraya–Watson versus Gaussian process priors are negligible. So, we present only the results for Nadaraya–Watson.

Figure 2 shows the predictive error on the test set as a function of m in two cases. The left panel shows that $m_{\mathsf{opt}} \approx 1$ where $m = \sum_{u=1}^{8} w_u$ when the Nadaraya–Watson estimator is used. The right panel shows $m_{\mathsf{opt}} \approx 2.6$ when the absolute value of the w_u's are summed, i.e., $m = \sum_{u=1}^{8} \mid w_u \mid$. Using Gaussian process priors gives essentially the same result. The difference shows that requiring the non-negativity constraint without adjusting the signs of the models going into the stack may give similar predictive performance – the minimum predictive errors in the two panels are nearly equal – but the constraints on the w_u's can make a large difference to the optimal values.

5.2 Soil moisture data

As an $\mathcal{M}$-open example, we consider the Soil Moisture data set. The response variable Y is an interpolated form of the moisture in the topsoil on a grid of plots near Waco, Nebraska. There are six explanatory variables $X_1, \ldots, X_6$; three are for location (two for location on a grid, one for elevation), two for soil electrical resistivity, and one for a standard 'wetness index' that is a function of elevation; see Franz et al. (2015) for a



(a) (b)

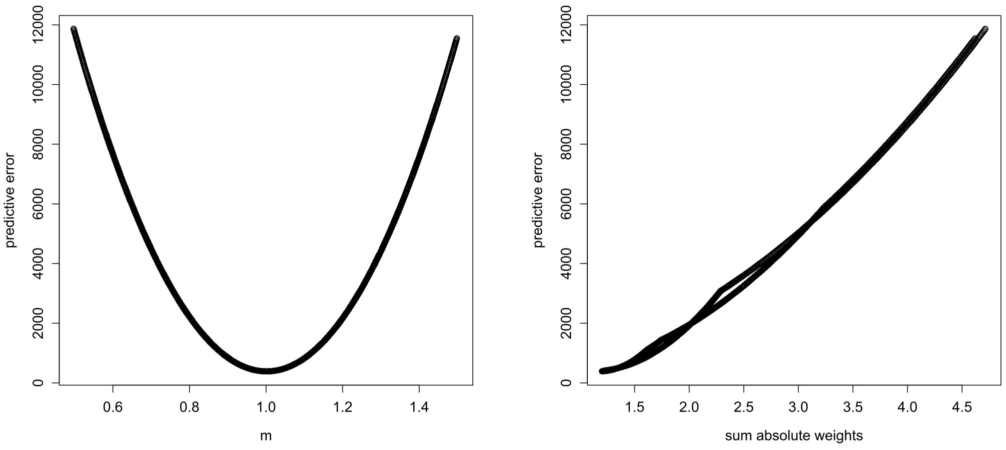
Figure 3: Parallel to Figure 1. Plots of cumulative predictive error for various J's to find the number of basis vectors to stack for X_1 in the Soil Moisture data. Numerically, the smallest value on the left hand graph occurs for $J = 8$. However, the right hand graph shows that past $J = 7$ the curve actually increases and the smoothed curve led us to choose $J_{\mathsf{opt}} = 7$ instead.

detailed description. The actual sample size is 18973 but for computational convenience, and comparison with our analysis of the Forest Fires data set, we randomly selected $n = 517$ data points, dividing them into two sets of size 267 and 250, at random, for training and validation, respectively, as in Section 5.1.

We used the model (21) but for six explanatory variables rather than eight. Thus, we found six estimates of univariate functions $\hat{A}_u(X_u)$ for $u = 1, \ldots, 6$ by stacking the basis vectors we found by using Nadaraya–Watson and Gaussian process priors on bootstrap samples. For each u we ordered the basis vectors by the absolute value of their Fourier coefficients with a full-data estimate of the unknown function (even though in an $\mathcal{M}$-open case this construct only exists in a mathematical sense). For each u we use (20) to find an optimal number of basis vectors J_{opt} to stack to get each $\hat{A}_u(X_u)$.

To identify J_{opt} for X_1 we generated Figure 3. Although the scale on the y-axis obscures the point, the left panel does show a variance-bias tradeoff. The left panel shows a sudden decrease around $J = 6$ after which the curve looks flat. In fact, past $J = 7$, the curve increases, but slowly, as indicated by the right panel which has a much finer scale on the y-axis. For A_1 we therefore chose seven basis vectors. We found the same qualitative appearance for the predictive error graphs for all $X_2, \ldots, X_6$ and for both Nadaraya–Watson and Gaussian process priors. For $X_2, \ldots, X_6$ we used $J_{\mathsf{opt}} = 11, 10, 9, 12$, and 12 respectively.

Figure 3 is qualitatively different from Figure 1. Being $\mathcal{M}$-complete, there really is a true function summarizing the Forest Fires data that the additive model can try



(a) (b)

Figure 4: Parallel to Figure 2, plots of cumulative predictive error of stacking six univariate predictors versus the stacking weights for the $\hat{A}_1, \ldots, \hat{A}_6$ for the Soil Moisture data, with basis vectors generated by Nadaraya–Watson. Left: Cumulative predictive error on the test set versus the sum of the stacking weights, m. Right: Cumulative predictive error on the test set versus the sum of the absolute value of the stacking weights. Note the 'doubling back' on the right hand panel.

to estimate. We attribute the nice, bowl-shaped appearance of Figure 1 to this clear variance-bias tradeoff. However, Figure 3 does not show a variance-bias tradeoff (except in a strictly technical sense). It shows that past a certain point the gain from adding more vectors to the stacking average of any A_u is effectively zero. Qualitatively, we see this is closer in spirit to a sparsity condition than to a variance-bias tradeoff. The importance of sparsity may be one of the key features of $\mathcal{M}$-open data such as Soil Moisture.

Figure 4 shows how the stacking coefficients for the $\hat{A}_u$'s for $u = 1, \ldots, 6$ behave with predictive error, parallel to Figure 2. The left hand panels of Figures 4 and 2 are nearly identical and show $m \approx 1$ is optimal. The right hand panels of Figures 4 and 2 are qualitatively different. The right hand panel in Figure 2 shows that as the sum of the absolute weights varies, the predictive error on the test set decreases to a minimum, approximately at $m_{\mathsf{opt}} = 1.2$ and then begins to increase again. This is a consequence of the predictive error not being a function of the sum of the absolute weights. That is, different choices of weights can have the same sum of absolute values but different predictive errors. We see this as another reflection of the complexity of $\mathcal{M}$-open data, but are unclear how to interpret it otherwise.

6 Discussion

Here we have formally established that leave-one-out CV is asymptotically the optimal action in posterior risk for a variety of loss functions. We have used this to justify the coefficients in a stacking predictor since they are based on a CV criterion. Stacking is a model averaging technique for prediction most effective when a true model is unavailable or may not even exist. We have investigated theoretically and computationally the effect of different choices of constraints on the coefficients of the stacking predictor and suggest that not imposing any leads to the best result in the sense of minimizing predictive error. In fact, our examples suggest that a 'sum to one' constraint naturally emerges when the predictive error is small. If the sum constraint is applied to the absolute values of the stacking weights the results can be very different. (Although they would have to be the same if the signs of the models being stacked were adjusted accordingly.) We comment that obvious extensions of our technique of proof show that leave-k-out CV by also be regarded as Bayes actions.

We recall that Stone (1977) showed that the Akaike information criterion (AIC) is asymptotically equivalent to leave-one-out CV and that Shao (1997) shows these further asymptotically equivalent to the Mallows' C_p criterion, the generalized CV, and the 'GIC_2' criterion. The implication from our main theorem here is that all of these methods can also be regarded as asymptotically Bayes optimal.

When the concept of a true model is problematic, it is natural to fall back on predictive methods. Indeed, it is possible that seeking a good predictor may be more useful that modeling when the model is very complex. For instance, if no simplification of the true model can be readily identified a model average predictor may give better performance in a mean squared error sense. This seems to be the case for our two examples here. In the first that we regard as $\mathcal{M}$-complete we stack orthogonal models. In the second, we used orthogonal models but found they were equivalent to independent models. We suspect that with further work we would find models that were different but not independent would do as well possibly even better cf. the Shtarkov solution in Section 3.4 of Le and Clarke (2016).

More generally, one may ask for a precise definition of $\mathcal{M}$-open. In Section 5.2 we have suggested that for $\mathcal{M}$-open problems, sparsity is more relevant than variance-bias tradeoff. We have also suggested that anomalous behavior of other quantities that would be well behaved were a true model to exist may be tip offs that a given problem is $\mathcal{M}$-open. However, these are suggestions more than defining properties and may simply reflect complexity of a data set relative to efforts to model it in a more general sense.

A logical definition is as the complement of $\{\mathcal{M}\text{-complete} \cup \mathcal{M}\text{-closed}\}$. However, this is not very useful. The best we can do at this time is to declare a problem $\mathcal{M}$-open if it defeats enough $\mathcal{M}$-complete or $\mathcal{M}$-closed efforts to address it or exhibits so little stability that such efforts seem bound to be inadequate.

Supplementary Material

Supplementary Appendices of "A Bayes interpretation of stacking for $\mathcal{M}$-complete and $\mathcal{M}$-open settings" (DOI: 10.1214/16-BA1023SUPP; .pdf).

References

Bernardo, J. and Smith, A. (2000). *Bayesian Theory*. Chichester: John Wiley & Sons. 808, 809

Breiman, L. (1996). "Stacked regressions." *Machine Learning*, 24: 49–64. 808, 809, 814, 818

Clarke, B. (2003). "Bayes model averaging and stacking when model approximation error cannot be ignored." *Journal of Machine Learning Research*, 683–712. MR2072265. doi: http://dx.doi.org/10.1162/153244304773936090. 808

Clyde, M. (2012). "Bayesian perspectives on combining models." *Slides from presentation at ISBA Kyoto*, 648–649. 814

Clyde, M. and Iversen, E. (2013). "Bayesian model averaging in the $\mathcal{M}$-open framework." In Damien, P., Dellaportas, P., Polson, N., and Stephens, D. (eds.), *Bayesian Theory and Applications*, 484–498. Oxford: Oxford University Press. MR3221178. doi: http://dx.doi.org/10.1093/acprof:oso/9780199695607.003.0024. 808, 810, 814, 816

Franz, T., Wang, T., Avery, W., Finkenbiner, C., and Brocca, L. (2015). "Combined analysis of soil moisture measurements from roving and fixed cosmic ray neutron probes for multiscale real-time monitoring." *Geophysical Research Letters*, 42: 3389–3396. 822, 824

Le, T. and Clarke, B. (2016). "Using the Bayesian Shtarkov solution for predictions." *Computational Statistics and Data Analysis*, 104: 183–196. doi: http://dx.doi.org/10.1016/j.csda.2016.06.018. 827

Le, T. and Clarke, B. (2016). "Supplementary Appendices of "A Bayes interpretation of stacking for $\mathcal{M}$-complete and $\mathcal{M}$-open settings"." *Bayesian Analysis*. doi: http://dx.doi.org/10.1214/16-BA1023SUPP. 811, 812, 813, 814, 819, 820

Minka, T. P. (2002). "Bayesian model averaging is not model combination." http://research.microsoft.com/en-us/um/people/minka/papers/minka-bma-isnt-mc.pdf. 819

Nadaraya, E. A. (1964). "On estimating regression." *Theory of Probability and Its Applications*, 9: 141–142. 821

Ozay, M. and Vural, F. T. Y. (2012). "A new fuzzy stacked generalization technique and analysis of its performance." arxiv:1204.0171. 808

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. Massachusetts: The MIT Press. MR2514435. 822

Rokach, L. (2010). "Ensemble-based classifiers." *Artificial Intelligence Review*, 33: 1–39. 808

Shao, J. (1997). "An asymptotic theory for linear model selection." *Statistica Sinica*, 7: 221–264. MR1466682. 827

Sill, J., Takacs, G., Mackey, L., and Lin, D. (2009). "Feature-Weighted Linear Stacking." arxiv:0911.0460. 808

Smyth, P. and Wolpert, D. (1999). "Linearly combining density estimators via stacking." *Machine Learning Journal*, 36: 59–83. 808

Stone, M. (1977). "Asymptotics for and against cross-validation." *Biometrika*, 64: 29–38. MR0474601. 827

Ting, K. M. and Witten, I. (1999). "Issues in stacked generalization." *Journal of Artificial Intelligent Research*, 10: 271–289. 808

van de Geer, S. (2014). "On the uniform convergence of empirical norms and inner products, with application to causal inference." *Electronic Journal of Statistics*, 8: 543–574. MR3211024. doi: `http://dx.doi.org/10.1214/14-EJS894`. 818

Walker, S. G. and Gutierrez-Pena, E. (1999). "Robustifying Bayesian procedures." *Bayesian Statistics*, 6: 685–710. MR1724876. 808

Watson, G. S. (1964). "Smooth regression analysis." *The Indian Journal of Statistics, Series A*, 26: 359–372. MR0185765. 821

Wolpert, D. (1992). "Stacked generalization." *Neural Networks*, 5: 241–259. 807

Wolpert, D. and Macready, W. (1999). "An efficient method to estimate bagging generalization error." *Machine Learning Journal*, 35: 41–55. 808