COBRA: A Nonlinear Aggregation Strategy

Gérard Biau
Université Pierre et Marie Curie[1] *& Ecole Normale Supérieure*[2]*, France*
gerard.biau@upmc.fr

Aurélie Fischer
Université Paris Diderot, France
aurelie.fischer@univ-paris-diderot.fr

Benjamin Guedj
Université Pierre et Marie Curie, France
benjamin.guedj@upmc.fr

James D. Malley
National Institute of Health, U.S.A.
jmalley@mail.nih.gov

Abstract

A new method for combining several initial estimators of the regression function is introduced. Instead of building a linear or convex optimized combination over a collection of basic estimators $r_1, \dots, r_M$, we use them as a collective indicator of the distance between the training data and a test observation. This local distance approach is model-free and extremely fast. Most importantly, the resulting collective estimator is shown to perform asymptotically at least as well in the L^2 sense as the best basic estimator in the collective. Moreover, it does so without having to declare which might be the best basic estimator for the given data set. A companion R package called `COBRA` (standing for COmBined Regression Alternative) is presented (downloadable on `http://cran.r-project.org/web/packages/COBRA/index.html`). Numerical evidence is provided on both synthetic and real data sets to assess the excellent performance of our method in a large variety of prediction problems.

Index terms — Regression estimation, aggregation, nonlinearity, consistency, prediction.

2010 Mathematics Subject Classification: 62G05, 62G20.

---

[1]Research partially supported by the French National Research Agency (grant ANR-09-BLAN-0051-02 "CLARA") and by the Institut universitaire de France.

[2]Research carried out within the INRIA project "CLASSIC" hosted by Ecole Normale Supérieure and CNRS.

1 Introduction

Recent years have witnessed a growing interest in aggregated statistical procedures, supported by a considerable research and thorough empirical evidence. Indeed, the increasing number of available estimation and prediction methods (hereafter also denoted as machines) in a wide range of modern statistical problems naturally suggests using some efficient method for combining procedures and estimators. If the combined strategy is known to be optimal in some sense and relatively free of assumptions that are hard to evaluate, then such a model-free strategy is a valuable research tool.

In this regard, numerous contributions have enriched the aggregation literature with various approaches, such as model selection aggregation (select the optimal single estimator from a list), convex aggregation (searching for the optimal convex combination of given estimators, such as exponentially weighted aggregates) and linear aggregation (selecting the optimal linear combination).

Model selection, linear-type aggregation strategies and related problems have been studied by Catoni (2004), Juditsky and Nemirovski (2000), Nemirovski (2000), Yang (2000, 2001, 2004), Györfi et al. (2002) and Wegkamp (2003). Minimax results have been derived by Nemirovski (2000) and Tsybakov (2003), leading to the notion of optimal rates of aggregation. Similar results can be found in Bunea et al. (2007a). Further upper bounds for the risk in model selection and convex aggregation have been established for instance by Audibert (2004), Birgé (2006) and Dalalyan and Tsybakov (2008). An interesting feature is that such aggregation problems may be treated within the scope of L^1-penalized least squares, as performed in Bunea et al. (2006, 2007a,b). This kind of framework is also considered by van de Geer (2008) and Koltchinskii (2009), with the L^2 loss replaced by another convex loss. More recently, specific models such as single-index in Alquier and Biau (2013) and additive models in Guedj and Alquier (2013) have been studied in the context of aggregation under a sparsity assumption.

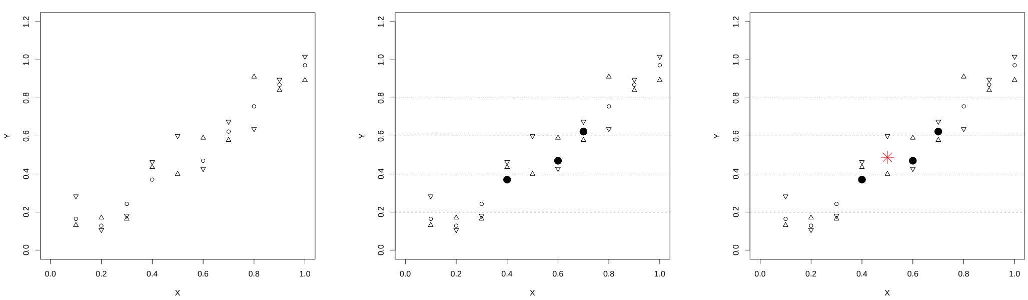
The present article investigates a distinctly different point of view, motivated by the sense that nonlinear, data-dependent techniques are a source of analytic flexibility and might improve over current aggregation procedures. In this regard, consider the following example classification problem: If the ensemble of machines happens to have a strong one, lurking but unnamed in the collection of which many might be very weak machines, it might make sense to consider a more sophisticated method than the previously cited methods for pooling the information across the machines. Thus, if one machine has an error rate of 5%, say, while most of the other machines have error rates close to 35%, then the ensemble approach might reduce the error rate to 25% or even 15%, but these are still significantly worse than the strong machine rate. Choosing to set aside some of the machines, on some data-dependent criteria, seems only weakly motivated, since the performance of the collective, retaining those suspect machines, might be quite good on a nearby data set. Similarly, searching for some phantom strong machine in the collective could also be ruinous when presented with new and different data.

Instead of choosing either of these options—selecting out weak performers, searching for a hidden, universally strong performer—we propose an original nonlinear method for combining the outcomes over some list of plausibly good procedures. We call this combined method a regression collective over the given basic machines. More specifically, we consider the problem of building a new estimator by combining M estimators of the regression function, thereby exploiting an idea proposed in the context of classification by Mojirsheibani (1999). In words, given a set of preliminary estimators $r_1, \ldots, r_M$, the idea behind the resulting aggregation method is a "unanimity" concept, in that it is based on the values predicted by $r_1, \ldots, r_M$ for the data and for a new observation $\mathbf{x}$. More specifically, a data point is considered to be "close" to $\mathbf{x}$, and consequently, reliable for contributing to the estimation of this new observation, if all estimators predict values which are close to each other for $\mathbf{x}$ and this data item, i.e., not more distant than a prespecified threshold ε. The predicted value corresponding to this query point $\mathbf{x}$ is then set to the average of the responses of the selected observations.

To make the concept clear, consider the following toy example illustrated by Figure 1. Assume we are given the observation plotted in circles, and the values predicted by two known machines f_1 and f_2 (triangles pointing up and down, respectively). The goal is to predict the response for the new point $\mathbf{x}_0 = 0.5$. Set a threshold $\varepsilon = 0.2$, the black solid circles are the data points $(\mathbf{x}_i, y_i)$ within the two dotted intervals, i.e.such that for $m = 1, 2$, $|f_m(\mathbf{x}_i) - f_m(\mathbf{x}_0)| < \varepsilon$. Averaging the corresponding y_i yields the prediction for $\mathbf{x}_0$ (black star).

Figure 1: A toy example.

(a) Data points. (b) The collective operates. (c) Predicted value for $\mathbf{x}_0$.



We stress that the central and original idea behind our approach is that the resulting regression predictor is a nonlinear, data-dependent function of the basic predictors $r_1, \dots, r_M$. To the best of our knowledge there exists no formalized procedure in the learning machine and aggregation literature that operates as does ours. However, we note that our approach has a conceptual link with the framework described in van der Laan et al. (2007), where several estimators are combined using a cross-validation scheme. Since their strategy—called Super Learner, SL—is motivated by research concerns similar to our own it is reasonable to deploy SL as a benchmark in our study of regression collectives.

Along with this paper, we release the software `COBRA` (Guedj, 2013) which implements the method as an additional package to the statistical software R (see R Core Team, 2012). `COBRA` is freely downloadable on the CRAN website[3]. As detailed in Section 3, we undertook a lengthy series of numerical experiments, over which `COBRA` proved extremely and surprisingly successful. These stunning results lead us to believe that regression collectives can provide valuable insights on a wide range of prediction problems. Finally, these same results demonstrate that `COBRA` has remarkable speed in terms of CPU timings. In the context of high-dimensional or genomic data, such velocity is critical, and in fact `COBRA` can natively take advantage of multi-core parallel environments.

The paper is organized as follows. In Section 2, we describe the combined estimator—the regression collective—and derive a non-asymptotic risk bound. Next we present the main result, that the collective is asymptotically at least as good as any of the basic estimators. Section 3 is devoted to the companion R package `COBRA` and presents benchmarks of its excellent performance on both simulated and real data sets. We also show that `COBRA` compares favorably with the SL, the `SuperLearner` R package, in that it performs similarly in most situations, much better in some, while it is consistently much faster in every case. Finally, for ease of exposition, proofs are collected in Section 4.

2 The combined estimator

2.1 Notation

Throughout the article, we assume to be given a training sample denoted by $\mathcal{D}_n = ((\mathbf{X}_1, Y_1), \dots, (\mathbf{X}_n, Y_n))$ composed of i.i.d. random variables taking their values in $\mathbb{R}^d \times \mathbb{R}$, distributed as an independent prototype pair $(\mathbf{X}, Y)$ satisfying $\mathbb{E}Y^2 < \infty$ (with the notation $\mathbf{X} = (X_1, \dots, X_d)$). The space $\mathbb{R}^d$ is equipped with the standard Euclidean metric. For fixed $\mathbf{x} \in \mathbb{R}^d$, our goal is to consistently estimate the regression function $r^\star(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$ using the data $\mathcal{D}_n$.

[3] http://cran.r-project.org/web/packages/COBRA/index.html

To begin with, the original data set $\mathcal{D}_n$ is split into two data sequences $\mathcal{D}_k = ((\mathbf{X}_1, Y_1), \dots, (\mathbf{X}_k, Y_k))$ and $\mathcal{D}_\ell = ((\mathbf{X}_{k+1}, Y_{k+1}), \dots, (\mathbf{X}_n, Y_n))$, with $\ell = n - k \geq 1$. For ease of notation, the elements of $\mathcal{D}_\ell$ are renamed $((\mathbf{X}_1, Y_1), \dots, (\mathbf{X}_\ell, Y_\ell))$. There is a slight abuse of notation here, as the same letter is used for both subsets $\mathcal{D}_k$ and $\mathcal{D}_\ell$—however, this should not cause any trouble since the context is clear.

Now, suppose that we are given a collection of $M \geq 1$ competing candidates $r_{k,1}, \dots, r_{k,M}$ to estimate $r^\star$.
These basic estimators—basic machines—are assumed to be generated using only the first subsample $\mathcal{D}_k$. These machines can be any among the researcherâĂŹs favorite tool kit, such as linear regression, kernel smoother, SVM, Lasso, neural, naive Bayes, or random forests. They could equally well be any ad hoc regression rules suggested by the experimental context. The essential idea is that these basic machines can be parametric or nonparametric, or indeed semi-parametric, with possible tuning rules. All what is asked for is that each of the $r_{k,m}(\mathbf{x})$, $m = 1, \dots, M$, is able to provide an estimation of $r^\star(\mathbf{x})$ on the basis of $\mathcal{D}_k$ alone. Thus, any collection of model-based or model-free machines are allowed, and the collection is here called the regression collective.

Given the collection of basic machines $\mathbf{r}_k = (r_{k,1}, \dots, r_{k,M})$, we define the collective estimator T_n be

$$T_n\left(\mathbf{r}_k(\mathbf{x})\right) = \sum_{i=1}^{\ell} W_{n,i}(\mathbf{x}) Y_i, \quad \mathbf{x} \in \mathbb{R}^d,$$

where the random weights $W_{n,i}(\mathbf{x})$ take the form

$$W_{n,i}(\mathbf{x}) = \frac{\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x}) - r_{k,m}(\mathbf{X}_i)| \leq \varepsilon_\ell\}}}{\sum_{j=1}^{\ell} \mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x}) - r_{k,m}(\mathbf{X}_j)| \leq \varepsilon_\ell\}}}. \tag{2.1}$$

In this definition, ε_ℓ is some positive parameter and, by convention, $0/0 = 0$.

The weighting scheme used in our regression collective is distinctive but not obvious. Starting from (Györfi et al., 2002), we see that T_n is a local averaging estimate in the following sense: The value for $r^\star(\mathbf{x})$, that is, the estimated outcome at the query point $\mathbf{x}$, is the unweighted average over those Y_i's such that $\mathbf{X}_i$ is "close" to the query point. More precisely, "close" means that the output at the query point, generated from each basic machine, is within an ε_ℓ distance of the output generated by the same basic machine at each $\mathbf{X}_i$ in the training data. If a basic machine evaluated at $\mathbf{X}_i$ is close to the basic machine evaluated at the query point $\mathbf{x}$, then the corresponding outcome Y_i is included in the average, and not otherwise. Also, as a further note of clarification: "closeness" of the $\mathbf{X}_i$ is not here in the Euclidean sense of close to any other point in the training data, or of the query point to any point in the training data. It refers to closeness of the basic machine outputs at the query point compared with basic machine outputs over all points in the training data. Training points $\mathbf{X}_i i$'s that are close, in the basic machine sense, to the corresponding basic machine output at the query point contribute to the indicator function for the corresponding outcome Y_i.

In this context, ε_ℓ plays the role of a smoothing parameter: Put differently, in order to retain Y_i, all basic estimators $r_{k,1}, \dots, r_{k,M}$ have to deliver predictions for the query point $\mathbf{x}$ which are in a ε_ℓ-neighborhood of the predictions $r_{k,1}(\mathbf{X}_i), \dots, r_{k,M}(\mathbf{X}_i)$. Note that the greater ε_ℓ, the more tolerant the process. It turns out that the practical performance of T_n strongly relies on an appropriate choice of ε_ℓ. This important question will thoroughly be discussed in Section 3, where we devise an automatic (i.e., data-dependent) selection strategy of ε_ℓ.

Next, we note that the subscript n in T_n may be a little confusing, since T_n is a weighted average of the Y_i's in $\mathcal{D}_\ell$ only. However, T_n depends on the entire data set $\mathcal{D}_n$, as the rest of the data is used to set up the original machines $r_{k,1}, \dots, r_{k,M}$. Finally, and most importantly, it should be noticed that the combined estimate T_n is nonlinear with respect to the basic estimators $r_{k,m}$'s. This makes it very different from more model selection, convex and linear aggregation. As such, it is inspired by the preliminary work of Mojirsheibani (1999) in the supervised classification context. It is also close in spirit to the "Super Learner" strategy developed by van der Laan et al. (2007), as mentioned earlier.

Let us finally mention that, in the weights definition (2.1), all original estimators are asked to have the same opinion on the importance of the observation $\mathbf{X}_i$ (within the range of ε_ℓ) for the corresponding Y_i to be integrated in the combinaison T_n. However, this unanimity constraint may be relaxed by imposing, for example, that a fixed fraction $\alpha \in (0, 1]$ of the machines agree on the importance of $\mathbf{X}_i$. In that case, the weights take the more sophisticated form

$$W_{n,i}(\mathbf{x}) = \frac{\mathbf{1}_{\{\sum_{m=1}^{M} \mathbf{1}_{\{|r_{k,m}(\mathbf{x}) - r_{k,m}(\mathbf{X}_i)| \le \varepsilon_\ell\}} \ge M\alpha\}}}{\sum_{j=1}^{\ell} \mathbf{1}_{\{\sum_{m=1}^{M} \mathbf{1}_{\{|r_{k,m}(\mathbf{x}) - r_{k,m}(\mathbf{X}_j)| \le \varepsilon_\ell\}} \ge M\alpha\}}}.$$

It turns out that adding the parameter α does not change the asymptotic properties of T_n, provided $\alpha \to 1$. Thus, to keep a sufficient degree of clarity in the mathematical statements and subsequent proofs, we have decided to consider only the case $\alpha = 1$ (i.e., unanimity). We leave as an exercise the possibility to extend the results to more general values of α. On the other hand, as highligthed by Section 3, α has a non-negligible impact on the performance of the combined estimate. Accordingly, we will discuss in Section 3 an automatic procedure to select this extra parameter.

2.2 Theoretical performance

This section is devoted to the study of some asymptotic and non-asymptotic properties of the combined estimate T_n, whose quality will be assessed by the quadratic risk

$$\mathbb{E}\left|T_n\left(\mathbf{r}_k(\mathbf{X})\right)-r^\star(\mathbf{X})\right|^2.$$

Here and later, $\mathbb{E}$ denotes the expectation with respect to both $\mathbf{X}$ and the sample $\mathcal{D}_n$. Throughout, we let

$$T\left(\mathbf{r}_k(\mathbf{X})\right)=\mathbb{E}\left[Y|\mathbf{r}_k(\mathbf{X})\right]$$

and note that, by the very definition of the L^2 conditional expectation,

$$\mathbb{E}\left|T(\mathbf{r}_k(\mathbf{X}))-Y\right|^2 \leq \inf_f \mathbb{E}\left|f(\mathbf{r}_k(\mathbf{X}))-Y\right|^2, \tag{2.2}$$

where the infimum is taken over all square integrable functions of $\mathbf{r}_k(\mathbf{X})$.

Our first result is a non-asymptotic inequality, which states that the combined estimator behaves as well as the best one in the original list, within a term measuring how far T_n is from T.

Theorem 2.1. *Let $\mathbf{r}_k=(r_{k,1},\ldots,r_{k,M})$ be the collection of basic estimators, and let $T_n(\mathbf{r}_n(\mathbf{x}))$ be the combined estimate. Then*

$$\begin{aligned}\mathbb{E}\left|T_n\left(\mathbf{r}_k(\mathbf{X})\right)-r^\star(\mathbf{X})\right|^2 \leq & \min_{m=1,\ldots,M}\mathbb{E}\left|r_{k,m}(\mathbf{X})-r^\star(\mathbf{X})\right|^2\\ &+\mathbb{E}\left|T_n\left(\mathbf{r}_k(\mathbf{X})\right)-T\left(\mathbf{r}_k(\mathbf{X})\right)\right|^2,\end{aligned}$$

for all distributions of $(\mathbf{X},Y)$ with $\mathbb{E}Y^2<\infty$.

Theorem 2.1 reassures us on the performance of T_n with respect to the basic machines, whatever the distribution of $(\mathbf{X},Y)$ is and regardless of which individual estimate is actually the best. The term $\mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X}))-T(\mathbf{r}_k(\mathbf{X}))|^2$ is a variance-type term, which can be asympotically controlled.

Proposition 2.1. *Assume that*

$$\varepsilon_\ell\to 0 \quad \textit{and} \quad \ell\varepsilon_\ell^M\to\infty \quad \textit{as } \ell\to\infty.$$

Then

$$\mathbb{E}\left|T_n\left(\mathbf{r}_k(\mathbf{X})\right)-T\left(\mathbf{r}_k(\mathbf{X})\right)\right|^2\to 0 \quad \textit{as } \ell\to\infty,$$

for all distribution of $(\mathbf{X},Y)$ with $\mathbb{E}Y^2<\infty$.

Thus, combining Theorem 2.1 and Proposition 2.1, we obtain

$$\limsup_{\ell\to\infty}\mathbb{E}\left|T_n\left(\mathbf{r}_k(\mathbf{X})\right)-r^\star(\mathbf{X})\right|^2\leq\min_{m=1,\ldots,M}\mathbb{E}\left|r_{k,m}(\mathbf{X})-r^\star(\mathbf{X})\right|^2.$$

This result is remarkable, for at least two reasons. Firstly, it shows that, in terms of predictive quadratic risk, the combined estimate does asymptotically at least as well as the best primitive machine. Secondly, the result is universal, in the sense that it is true for all distributions of $(\mathbf{X}, Y)$, without exceptions. This is especially interesting because the performance of any estimation procedure eventually depends upon some model and smoothness assumptions on the observations. For example, a linear regression fit performs well if the distribution is truly linear, but may behave poorly otherwise. Similarly, the Lasso procedure is known to do a good job for non-correlated designs (see van de Geer, 2008), with no clear guarantee however in adversarial situations. Likewise, rates of convergence of nonparametric procedures such as the k-nearest neighbor method, kernel estimates and random forests dramatically deteriorate as the ambient dimension increases, but may be significantly improved if the true underlying dimension is reasonable. This phenomenon is thoroughly analyzed for the random forests algorithm in Biau (2012). The crux is that model and smoothness assumptions are usually unverifiable, especially in modern, high-dimensional and large scale data sets. To circumvent this difficulty, people often try many different methods and retain the one exhibiting the best empirical results. Our aggregation strategy offers a nice alternative, in the sense that if one of the initial estimators is consistent for a given smoothness class $\mathcal{M}$ of distributions, then T_n inherits the same property. Our procedure therefore allows the statistician to consider model-free prediction. This is formalized in the following corollary.

Corollary 2.1. *Assume that one of the original estimators, say r_{k,m_0}, satisfies*

$$\mathbb{E}\left|\mathbf{r}_{k,m_0}(\mathbf{X}) - r^\star(\mathbf{X})\right|^2 \to 0 \quad \textit{as } k \to \infty$$

for all distribution of $(\mathbf{X}, Y)$ in some smoothness class $\mathcal{M}$. Then, if

$$\varepsilon_\ell \to 0 \quad \textit{and} \quad \ell\varepsilon_\ell^M \to \infty \quad \textit{as } \ell \to \infty,$$

one has

$$\mathbb{E}\left|T_n\left(\mathbf{r}_k(\mathbf{X})\right) - r^\star(\mathbf{X})\right|^2 \to 0 \quad \textit{as } k, \ell \to \infty,$$

for all distribution of $(\mathbf{X}, Y)$ in $\mathcal{M}$.

3 Implementation and numerical studies

This section is devoted to the implementation of the described method. Its excellent performance is then assessed in a series of benchmarks. The companion R package `COBRA` (standing for COmBined Regression Alternative) is available on the CRAN website http://cran.r-project.org/web/packages/COBRA/index.html, for Linux, Mac and Windows platforms, see Guedj (2013). `COBRA` includes a `parallel` option, allowing for improved performance on multi-core computers (see Knaus, 2010).

As raised in the previous section, a fine calibration of the smoothing parameter ε_ℓ is crucial. Clearly, a too small value will discard many machines and most weights will be zero. Conversely, a large value sets all weights to $1/\Sigma$ with

$$\Sigma = \sum_{j=1}^{\ell} \mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x})-r_{k,m}(\mathbf{X}_j)|\le\varepsilon_\ell\}},$$

giving the naive predictor that does not take into account any new data point and predict the mean over sample $\mathcal{D}_\ell$. We also consider a relaxed version of the unanimity constraint: Instead of requiring global agreement over the implemented machines, consider some $\alpha \in (0,1]$ and keep observation Y_i in the construction of T_n if and only if at least a proportion α of the machines agree on the importance of $\mathbf{X}_i$. This parameter requires as well a fine calibration. To understand better, consider the following toy example: On some data set, assume most machines but one have nice predictive performance. For any new data point, requiring global agreement will fail since the pool of machines is heterogeneous. In this regard, α should be seen as a measure of homogeneity: If a small value is selected, it should be seen as an indicator that some machines perform (possibly much) better than some others. Conversely, a large value indicates that the predictive abilities of the machines are close.

A natural measure of the risk in the prediction context is the empirical quadratic loss, namely

$$r(\hat{\mathbf{Y}}) = \frac{1}{\ell}\sum_{j=1}^{\ell}(\hat{Y}_i - Y_i)^2,$$

where $\hat{\mathbf{Y}} = (\hat{Y}_1, \ldots, \hat{Y}_\ell)$ is the vector of predicted values for the responses $Y_1, \ldots, Y_\ell$.

We adopted the following protocol: Using a simple data-splitting device, ε_ℓ and α are chosen by minimizing the empirical risk r over the set $\{\varepsilon_{\ell,\min}, \ldots, \varepsilon_{\ell,\max}\} \times \{1/M, \ldots, 1\}$, where $\varepsilon_{\ell,\min} = 10^{-9}$ and $\varepsilon_{\ell,\max}$ is the largest difference between two predictions of the pool of machines. In the package, $\#\{\varepsilon_{\ell,\min}, \ldots, \varepsilon_{\ell,\max}\}$ may be modified by the user, otherwise the default value 100 is chosen. Figure 2 illustrates the discussion about the choice of ε_ℓ and α.

By default, `COBRA` includes the following classical packages dealing with regression estimation and prediction. However, note that the user has the choice to modify this list to her/his own convenience.

- Lasso (R package `lars`, see Hastie and Efron, 2012),
- Ridge regression (R package `ridge`, see Cule, 2012),
- k-nearest neighbors (R package `FNN`, see Li, 2012),
- CART algorithm (R package `tree`, see Ripley, 2012),

- Random Forest algorithm (R package `randomForest`, see Liaw and Wiener, 2002).

First, `COBRA` is benchmarked on synthetic data. For each of the following eight models, two designs are considered: Uniform over $(-1, 1)$ (referred to as "Uncorrelated" in Table 1, Table 2 and Table 3), and Gaussian with mean 0 and covariance matrix Σ with $\Sigma_{ij} = 2^{-|i-j|}$ ("Correlated"). Models considered cover a wide spectrum of contemporary regression problems. Indeed, Model 2 comes from van der Laan et al. (2007), Model 3 and Model 4 appear in Meier et al. (2009). Model 1 and Model 5 are classic settings. Model 6 is about predicting labels, Model 7 is inspired by high-dimensional sparse regression problems. Finally, Model 8 deals with probability estimation, linking with nonparametric model-free approaches such as in Malley et al. (2012). In the sequel, we let $\mathcal{N}(\mu, \sigma^2)$ denote a Gaussian random variable with mean μ and variance σ^2. In the simulations, the training data set was usually set to 80% of the whole sample, then split into two equal parts corresponding to $\mathcal{D}_k$ and $\mathcal{D}_\ell$.

Model 1. $n = 800$, $d = 50$, $Y = X_1^2 + \exp(-X_2^2)$.

Model 2. $n = 600$, $d = 100$, $Y = X_1X_2 + X_3^2 - X_4X_7 + X_8X_{10} - X_6^2 + \mathcal{N}(0, 0.5)$.

Model 3. $n = 600$, $d = 100$, $Y = -\sin(2X_1) + X_2^2 + X_3 - \exp(-X_4) + \mathcal{N}(0, 0.5)$.

Model 4. $n = 600$, $d = 100$, $Y = X_1 + (2X_2 - 1)^2 + \sin(2\pi X_3)/(2 - \sin(2\pi X_3)) + \sin(2\pi X_4) + 2\cos(2\pi X_4) + 3\sin^2(2\pi X_4) + 4\cos^2(2\pi X_4) + \mathcal{N}(0, 0.5)$.

Model 5. $n = 700$, $d = 20$, $Y = \mathbf{1}_{\{X_1>0\}} + X_2^3 + \mathbf{1}_{\{X_4+X_6-X_8-X_9>1+X_{14}\}} + \exp(-X_2^2) + \mathcal{N}(0, 0.5)$.

Model 6. $n = 500$, $d = 30$, $Y = \sum_{k=1}^{10} \mathbf{1}_{\{X_k^3<0\}} - \mathbf{1}_{\{\mathcal{N}(0,1)>1.25\}}$.

Model 7. $n = 600$, $d = 300$, $Y = X_1^2 + X_2^2 X_3 \exp(-|X_4|) + X_6 - X_8 + \mathcal{N}(0, 0.5)$.

Model 8. $n = 600$, $d = 50$, $Y = \mathbf{1}_{\{X_1+X_4^3+X_9+\sin(X_{12}X_{18})+\mathcal{N}(0,0.1)>0.38\}}$.

Table 1 presents the mean quadratic error and standard deviation over 100 independent replications, for each model and design. Bold number identifies the lowest error, i.e., the best competitor. Boxplots of errors are presented in Figure 3 and Figure 4. Further, Figure 5 and Figure 6 shows the predictive capacities of `COBRA`, and Figure 7 depicts its ability to reconstruct the functional dependence over the covariates when this dependence is additive, assessing the striking performance of our approach in a wide spectrum of statistical settings. A remarkable fact is that `COBRA` performs at least as well as the best machine, and improves even significantly in Model 3, Model 5 and Model 6.

Next, we compare COBRA to the SuperLearner algorithm (Polley and van der Laan, 2012). This widespread algorithm was first described in van der Laan et al. (2007). SuperLearner is used in this section as the key competitor to our method: In a spirit close to ours, the main idea lies on a nonlinear way to combine basic estimators based on cross-validation. We feel close to the approach used in the SuperLearner package, allowing the user to add as many machines as desired, then blending them to deliver predictive outcomes.

Table 2 summarizes the performance of COBRA and SuperLearner (used with SL.randomForest, SL.ridge and SL.glmnet, so that both methods compete on equal terms) through the described protocol. Both methods compete on similar terms in most models, although COBRA proves much more efficient on correlated design in Model 2 and Model 4. This already remarkable result is to be stressed by the flexibility and velocity showed by COBRA. Indeed, as emphasized in Table 3, without even using the parallel option, COBRA obtains similar or better results than SuperLearner roughly five times faster.

Next, COBRA is used to process the following real-life data sets.

- Concrete Slump Test[4] (see Yeh, 2007),
- Concrete Compressive Strength[5] (see Yeh, 1998),
- Wine Quality[6] (see Cortez et al., 2009). Note that this data set involves supervised classification and opens a line for future research since COBRA is mainly devoted to regression.

The good predictive performance of COBRA is summarized in Figure 8 and errors are presented in Figure 9. For every data set, the sample is divided into a training set (90%) and a testing set (10%) on which the predictive performance is evaluated.
As a conclusion to this thorough experimental protocol, COBRA sets a new gold standard for prediction-oriented problems in the context of regression.

[4] http://archive.ics.uci.edu/ml/datasets/Concrete+Slump+Test.
[5] http://archive.ics.uci.edu/ml/datasets/Concrete+Compressive+Strength.
[6] http://archive.ics.uci.edu/ml/datasets/Wine+Quality.

Table 1: Quadratic errors of the implemented machines and COBRA. Means and standard deviations over 100 independent replications.

Uncorrelated		lars	ridge	fnn	tree	rf	COBRA
Model 1	m.	0.1561	0.1324	0.1585	0.0281	0.0330	**0.0259**
	sd.	0.0123	0.0094	0.0123	0.0043	0.0033	0.0036
Model 2	m.	0.4880	0.2462	0.3070	0.1746	**0.1366**	0.1645
	sd.	0.0676	0.0233	0.0303	0.0270	0.0161	0.0207
Model 3	m.	0.2536	0.5347	1.1603	0.4954	0.4027	**0.2332**
	sd.	0.0271	0.4469	0.1227	0.0772	0.0558	0.0272
Model 4	m.	7.6056	6.3271	10.5890	3.7358	3.5262	**3.3640**
	sd.	0.9419	1.0800	0.9404	0.8067	0.3223	0.5178
Model 5	m.	0.2943	0.3311	0.5169	0.2918	0.2234	**0.2060**
	sd.	0.0214	0.1012	0.0439	0.0279	0.0216	0.0210
Model 6	m.	0.8438	1.0303	2.0702	2.3476	1.3354	**0.8345**
	sd.	0.0916	0.4840	0.2240	0.2814	0.1590	0.1004
Model 7	m.	1.0920	0.5452	0.9459	0.3638	0.3110	**0.3052**
	sd.	0.2265	0.0920	0.0833	0.0456	0.0325	0.0298
Model 8	m.	0.1308	0.1279	0.2243	0.1715	0.1236	**0.1021**
	sd.	0.0120	0.0161	0.0189	0.0270	0.0100	0.0155

Correlated		lars	ridge	fnn	tree	rf	COBRA
Model 1	m.	2.3736	1.9785	2.0958	0.3312	0.5766	**0.3301**
	sd.	0.4108	0.3538	0.3414	0.1285	0.1914	0.1239
Model 2	m.	8.1710	4.0071	4.3892	**1.3609**	1.4768	1.3612
	sd.	1.5532	0.6840	0.7190	0.4647	0.4415	0.4654
Model 3	m.	6.1448	6.0185	8.2154	4.3175	4.0177	**3.7917**
	sd.	11.9450	12.0861	13.3121	11.7386	12.4160	11.1806
Model 4	m.	60.5795	42.2117	51.7293	**9.6810**	14.7731	9.6906
	sd.	11.1303	9.8207	10.9351	3.9807	5.9508	3.9872
Model 5	m.	6.2325	7.1762	10.1254	3.1525	4.2289	**2.1743**
	sd.	2.4320	3.5448	3.1190	2.1468	2.4826	1.6640
Model 6	m.	1.2765	1.5307	2.5230	2.6185	1.2027	**0.9925**
	sd.	0.1381	0.9593	0.2762	0.3445	0.1600	0.1210
Model 7	m.	20.8575	4.4367	5.8893	3.6865	**2.7318**	2.9127
	sd.	7.1821	1.0770	1.2226	1.0139	0.8945	0.9072
Model 8	m.	0.1366	0.1308	0.2267	0.1701	0.1226	**0.0984**
	sd.	0.0127	0.0143	0.0179	0.0302	0.0102	0.0144

Table 2: Quadratic errors of `SuperLearner` and `COBRA`. Means and standard deviations over 100 independent replications.

Uncorr.		SL	COBRA
Model 1	m.	0.0541	**0.0320**
	sd.	0.0053	0.0104
Model 2	m.	**0.1765**	0.3569
	sd.	0.0167	0.8797
Model 3	m.	**0.2081**	0.2573
	sd.	0.0282	0.0699
Model 4	m.	4.3114	**3.7464**
	sd.	0.4138	0.8746
Model 5	m.	**0.2119**	0.2187
	sd.	0.0317	0.0427
Model 6	m.	**0.7627**	1.0220
	sd.	0.1023	0.3347
Model 7	m.	**0.1705**	0.3103
	sd.	0.0260	0.0490
Model 8	m.	0.1081	**0.1075**
	sd.	0.0121	0.0235

Corr.		SL	COBRA
Model 1	m.	0.8733	**0.3262**
	sd.	0.2740	0.1242
Model 2	m.	2.3391	**1.3984**
	sd.	0.4958	0.3804
Model 3	m.	**3.1885**	3.3201
	sd.	1.5101	1.8056
Model 4	m.	25.1073	**9.3964**
	sd.	7.3179	2.8953
Model 5	m.	5.6478	**4.9990**
	sd.	7.7271	9.3103
Model 6	m.	**0.8967**	1.1988
	sd.	0.1197	0.4573
Model 7	m.	**3.0367**	3.1401
	sd.	1.6225	1.6097
Model 8	m.	0.1116	**0.1045**
	sd.	0.0111	0.0216

Table 3: Average CPU-times in seconds. No parallelization. Means and standard deviations over 10 independent replications.

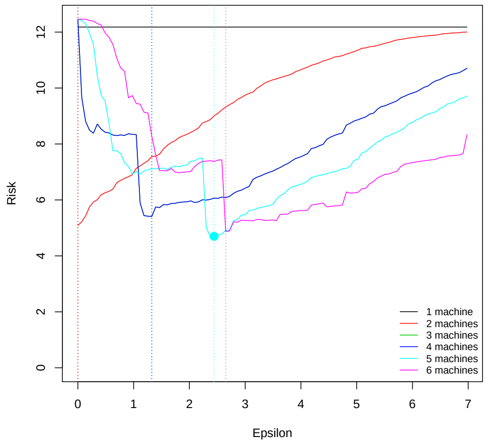
Uncorr.		SL	COBRA
Model 1	m.	53.92	**10.92**
	sd.	1.42	0.29
Model 2	m.	57.96	**11.90**
	sd.	0.95	0.31
Model 3	m.	53.70	**10.66**
	sd.	0.55	0.11
Model 4	m.	55.00	**11.15**
	sd.	0.74	0.18
Model 5	m.	28.46	**5.01**
	sd.	0.73	0.06
Model 6	m.	22.97	**3.99**
	sd.	0.27	0.05
Model 7	m.	127.80	**35.67**
	sd.	5.69	1.91
Model 8	m.	32.98	**6.46**
	sd.	1.33	0.33

Corr.		SL	COBRA
Model 1	m.	61.92	**11.96**
	sd.	1.85	0.27
Model 2	m.	70.90	**14.16**
	sd.	2.47	0.57
Model 3	m.	59.91	**11.92**
	sd.	2.06	0.41
Model 4	m.	63.58	**13.11**
	sd.	1.21	0.34
Model 5	m.	31.24	**5.02**
	sd.	0.86	0.07
Model 6	m.	24.29	**4.12**
	sd.	0.82	0.15
Model 7	m.	145.18	**41.28**
	sd.	8.97	2.84
Model 8	m.	31.31	**6.24**
	sd.	0.73	0.11

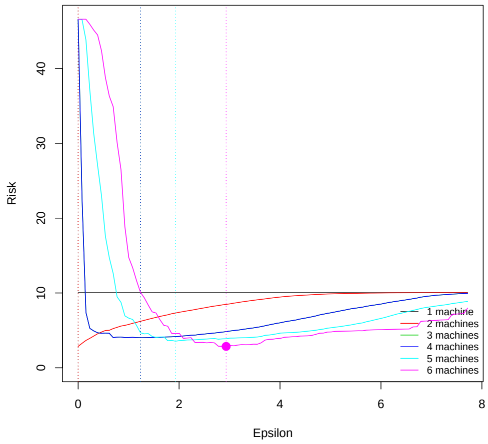
Figure 2: Examples of calibration of parameters ε_ℓ and α. The bold point is the minimum.

(a) Model 3, correlated design.

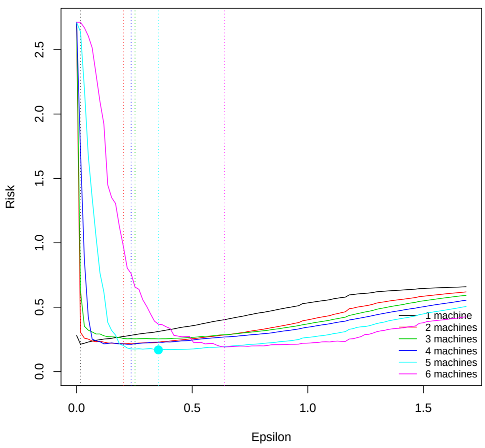
s



(b) Model 4, uncorrelated design.



(c) Model 5, uncorrelated design.



(d) Model 5, correlated design.

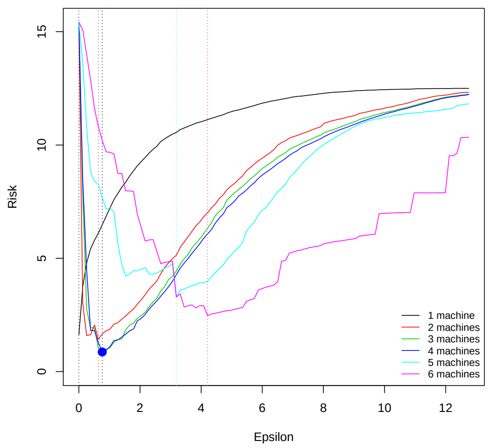
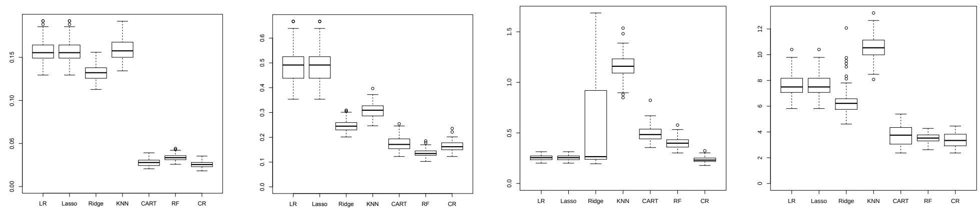


Figure 3: Boxplots of quadratic errors, uncorrelated design. From left to right: `lars`, `ridge`, `fnn`, `tree`, `randomForest`, `COBRA`.

(a) Model 1. (b) Model 2. (c) Model 3. (d) Model 4.



(e) Model 5. (f) Model 6. (g) Model 7. (h) Model 8.

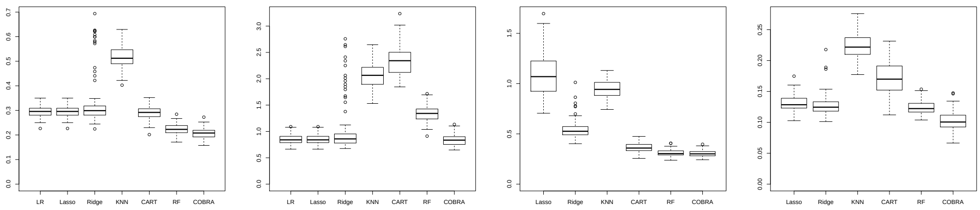
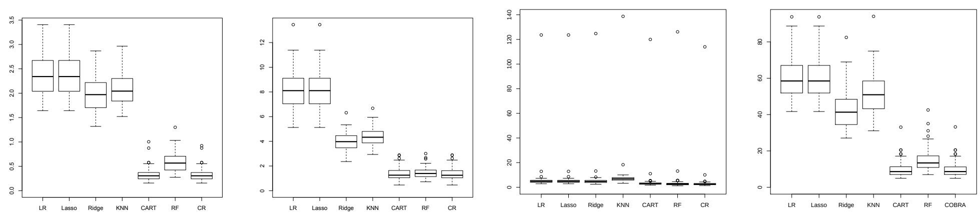


Figure 4: Boxplots of quadratic errors, correlated design. From left to right: `lars`, `ridge`, `fnn`, `tree`, `randomForest`, `COBRA`.

(a) Model 1. (b) Model 2. (c) Model 3. (d) Model 4.



(e) Model 5. (f) Model 6. (g) Model 7. (h) Model 8.

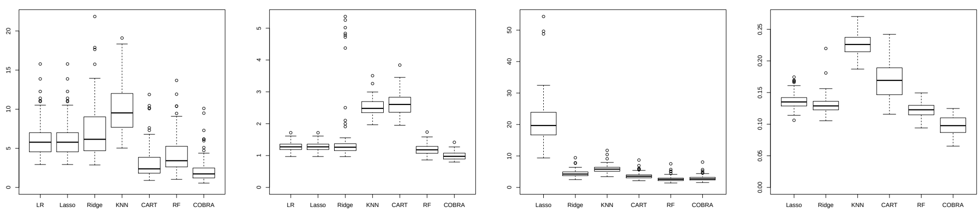
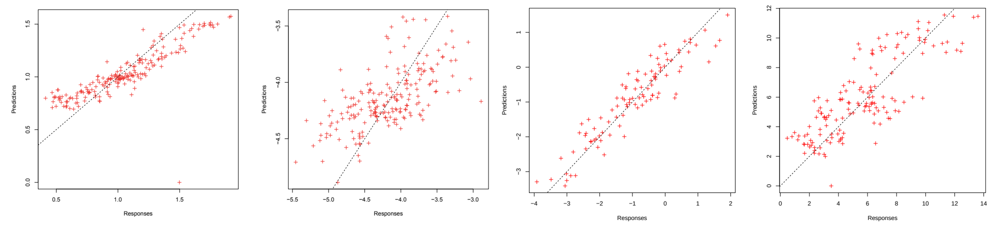


Figure 5: Prediction over the testing set, uncorrelated design. The more points on the first bissectrix, the better the prediction.

(a) Model 1. (b) Model 2. (c) Model 3. (d) Model 4.



(e) Model 5. (f) Model 6. (g) Model 7. (h) Model 8.

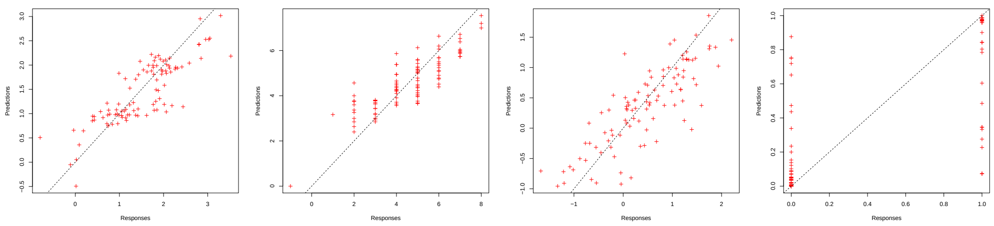
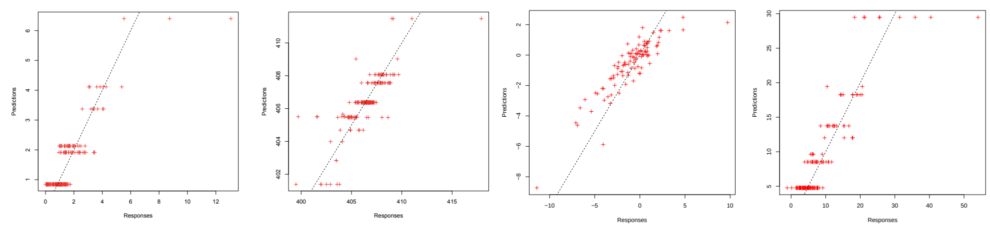


Figure 6: Prediction over the testing set, correlated design. The more points on the first bissectrix, the better the prediction.

(a) Model 1. (b) Model 2. (c) Model 3. (d) Model 4.



(e) Model 5. (f) Model 6. (g) Model 7. (h) Model 8.

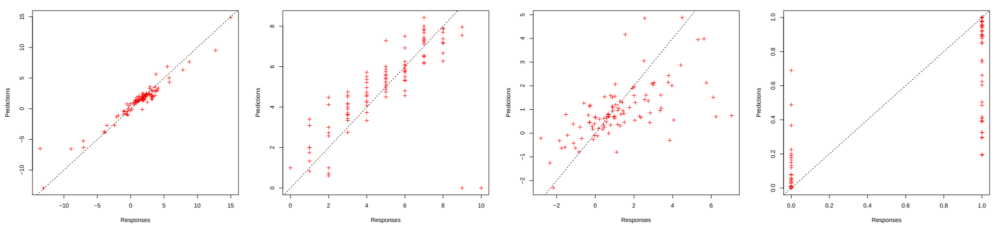
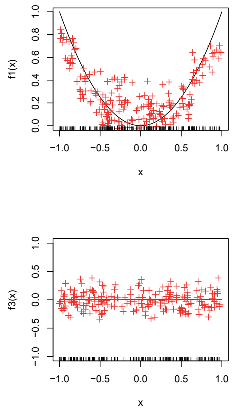
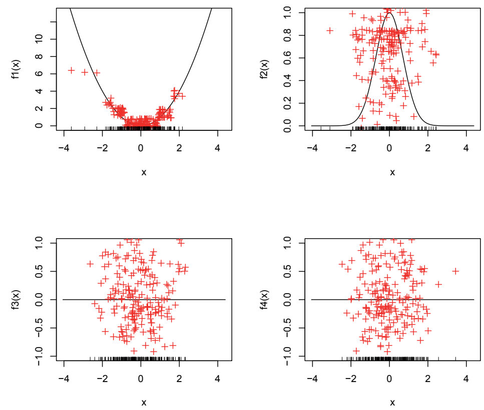


Figure 7: Examples of reconstruction of the functional dependencies, for covariates 1 to 4.

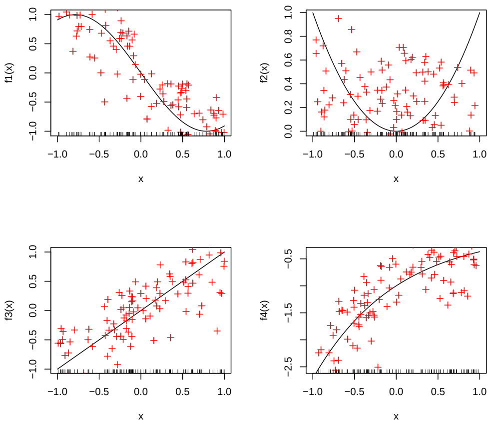
(a) Model 1, uncorrelated design.



(b) Model 1, correlated design.



(c) Model 3, uncorrelated design.



(d) Model 3, correlated design.

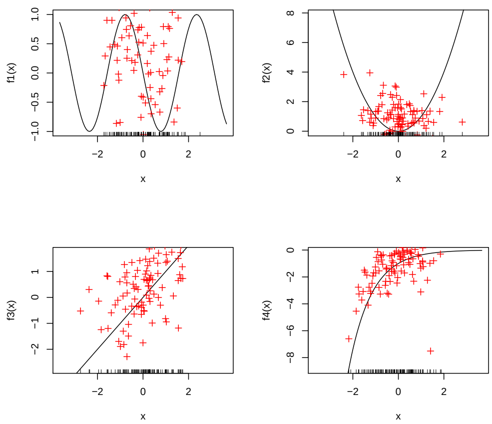


Figure 8: Prediction over the testing set, real-life data sets.

(a) Concrete Slump Test. (b) Concrete Compressive Strength. (c) Wine Quality, red wine. (d) Wine Quality, white wine.

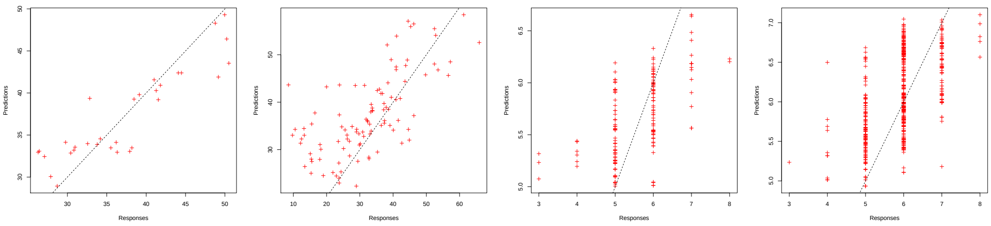
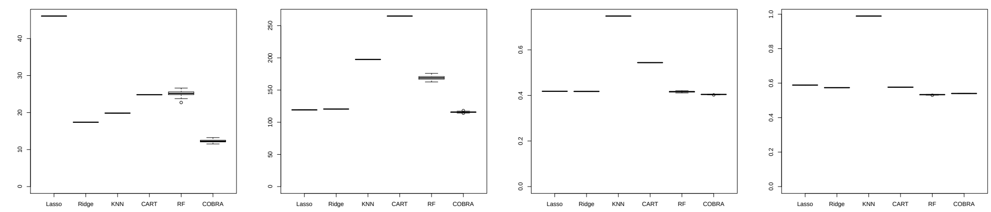


Figure 9: Boxplot of quadratic errors, real-life data sets.

(a) Concrete Slump Test. (b) Concrete Compressive Strength. (c) Wine Quality, red wine. (d) Wine Quality, white wine.



4 Proofs

4.1 Proof of Theorem 2.1

For each $m = 1, \ldots, M$, we have

$$\begin{aligned} 0 &\leq \mathbb{E}|r_{k,m}(\mathbf{X}) - Y|^2 - \mathbb{E}|T(\mathbf{r}_k(\mathbf{X})) - Y|^2 \\ &= \mathbb{E}|r_{k,m}(\mathbf{X}) - Y|^2 - \mathbb{E}|r^\star(\mathbf{X}) - Y|^2 + \mathbb{E}|r^\star(\mathbf{X}) - Y|^2 \\ &\quad - \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - Y|^2 + \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - Y|^2 - \mathbb{E}|T(\mathbf{r}_k(\mathbf{X})) - Y|^2, \end{aligned} \tag{4.1}$$

where we used that $\mathbb{E}\,|T(\mathbf{r}_k(\mathbf{X})) - Y|^2 \leq \inf_f \mathbb{E}\,|f(\mathbf{r}_k(\mathbf{X})) - Y|^2$. Observe now that

$$\mathbb{E}|r_{k,m}(\mathbf{X}) - Y|^2 = \mathbb{E}|r_{k,m}(\mathbf{X}) - r^\star(\mathbf{X})|^2 + \mathbb{E}|r^\star(\mathbf{X}) - Y|^2, \tag{4.2}$$

since

$$\begin{aligned} &\mathbb{E}[(r_{k,m}(\mathbf{X}) - r^\star(\mathbf{X}))(r^\star(\mathbf{X}) - Y)] \\ &\quad = \mathbb{E}[\mathbb{E}[(r_{k,m}(\mathbf{X}) - r^\star(\mathbf{X}))(r^\star(\mathbf{X}) - Y)|\mathcal{D}_k, \mathbf{X}]] \\ &\quad = \mathbb{E}[(r_{k,m}(\mathbf{X}) - r^\star(\mathbf{X}))\mathbb{E}[r^\star(\mathbf{X}) - Y|\mathbf{X}]] \\ &\quad = \mathbb{E}[(r_{k,m}(\mathbf{X}) - r^\star(\mathbf{X}))(r^\star(\mathbf{X}) - r^\star(\mathbf{X}))] \\ &\quad = 0. \end{aligned}$$

Likewise,

$$\mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - Y|^2 = \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - r^\star(\mathbf{X})|^2 + \mathbb{E}|r^\star(\mathbf{X}) - Y|^2$$

and

$$\mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - Y|^2 = \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - T(\mathbf{r}_k(\mathbf{X}))|^2 + \mathbb{E}|T(\mathbf{r}_k(\mathbf{X})) - Y|^2.$$

Combining these equalities reveals that the expression in (4.2) equals

$$\mathbb{E}|r_{k,m}(\mathbf{X}) - r^\star(\mathbf{X})|^2 - \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - r^\star(\mathbf{X})|^2 + \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - T(\mathbf{r}_k(\mathbf{X}))|^2.$$

It follows that

$$\mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - r^\star(\mathbf{X})|^2 \leq \mathbb{E}|r_{k,m}(\mathbf{X}) - r^\star(\mathbf{X})|^2 + \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - T(\mathbf{r}_k(\mathbf{X}))|^2.$$

Taking the infimum over $m = 1, \ldots, M$ leads to

$$\begin{aligned} \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - r^\star(\mathbf{X})|^2 \leq &\min_{m=1,\ldots,M} \mathbb{E}|r_{k,m}(\mathbf{X}) - r^\star(\mathbf{X})|^2 \\ &+ \mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - T(\mathbf{r}_k(\mathbf{X}))|^2. \end{aligned}$$

This is the desired result.

4.2 Proof of Proposition 2.1

We start with a technical lemma, whose proof can be found in the monograph by Györfi et al. (2002).

Lemma 4.1. *Let $B(n,p)$ be a binomial random variable with parameters $n \geq 1$ and $p > 0$. Then*

$$\mathbb{E}\left[\frac{1}{1+B(n,p)}\right] \leq \frac{1}{p(n+1)}$$

and

$$\mathbb{E}\left[\frac{\mathbf{1}_{\{B(n,p)>0\}}}{B(n,p)}\right] \leq \frac{2}{p(n+1)}.$$

For all distribution of $(\mathbf{X}, Y)$, using the elementary inequality $(a+b+c)^2 \leq 3(a^2+b^2+c^2)$, note that

$$\begin{aligned}
&\mathbb{E}|T_n(\mathbf{r}_k(\mathbf{X})) - T(\mathbf{r}_k(\mathbf{X}))|^2 \\
&\quad = \mathbb{E}\left|\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})\left(Y_i - T(\mathbf{r}_k(\mathbf{X}_i)) + T(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X})) + T(\mathbf{r}_k(\mathbf{X}))\right)\right. \\
&\qquad \left. - T(\mathbf{r}_k(\mathbf{X}))\right|^2 \\
&\quad \leq 3\mathbb{E}\left|\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})(T(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X})))\right|^2 \qquad (4.3) \\
&\qquad + 3\mathbb{E}\left|\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})(Y_i - T(\mathbf{r}_k(\mathbf{X}_i)))\right|^2 \qquad (4.4) \\
&\qquad + 3\mathbb{E}\left|\sum_{i=1}^{\ell} (W_{n,i}(\mathbf{X}) - 1) T(\mathbf{r}_k(\mathbf{X}))\right|^2. \qquad (4.5)
\end{aligned}$$

Consequently, to prove the proposition, it suffices to establish that (4.3), (4.4) and (4.5) tend to 0 as ℓ tends to infinity. This is done, respectively, in Proposition 4.1, Proposition 4.2 and Proposition 4.3 below.

Proposition 4.1. *Under the assumptions of Proposition 2.1,*

$$\lim_{\ell\to\infty} \mathbb{E}\left|\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})(T(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X})))\right|^2 = 0.$$

Proof of Proposition 4.1. By the Cauchy-Schwarz inequality,

$$
\begin{aligned}
&\mathbb{E}\left|\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})(T(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X})))\right|^2 \\
&= \mathbb{E}\left|\sum_{i=1}^{\ell} \sqrt{W_{n,i}(\mathbf{X})}\sqrt{W_{n,i}(\mathbf{X})}\,(T(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X})))\right|^2 \\
&\leq \mathbb{E}\left[\sum_{j=1}^{\ell} W_{n,j}(\mathbf{X}) \sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})\,|T(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X}))|^2\right] \\
&= \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})\,|T(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X}))|^2\right] \\
&:= A_n.
\end{aligned}
$$

The function T is such that $\mathbb{E}[T^2(\mathbf{r}_k(\mathbf{X}))] < \infty$. Therefore, it can be approximated in an L^2 sense by a continuous function with compact support, say $\tilde{T}$. This result may be found in many references, amongst them Györfi et al. (2002, Theorem A.1). More precisely, for any $\eta > 0$, there exists a function $\tilde{T}$ such that

$$
\mathbb{E}\left|T(\mathbf{r}_k(\mathbf{X})) - \tilde{T}(\mathbf{r}_k(\mathbf{X}))\right|^2 < \eta.
$$

Consequently, we obtain

$$
\begin{aligned}
A_n &= \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|T(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X}))|^2\right] \\
&\leq 3\mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|T(\mathbf{r}_k(\mathbf{X}_i)) - \tilde{T}(\mathbf{r}_k(\mathbf{X}_i))|^2\right] \\
&\quad + 3\mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|(\tilde{T}\mathbf{r}_k(\mathbf{X}_i)) - \tilde{T}(\mathbf{r}_k(\mathbf{X}))|^2\right] \\
&\quad + 3\mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|\tilde{T}(\mathbf{r}_k(\mathbf{X})) - T(\mathbf{r}_k(\mathbf{X}))|^2\right] \\
&:= 3A_{n1} + 3A_{n2} + 3A_{n3}.
\end{aligned}
$$

Computation of A_{n3}. Thanks to the approximation of T by $\tilde{T}$,

$$
\begin{aligned}
A_{n3} &= \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|T(\mathbf{r}_k(\mathbf{X})) - \tilde{T}(\mathbf{r}_k(\mathbf{X}))|^2\right] \\
&\leq \mathbb{E}\left|T(\mathbf{r}_k(\mathbf{X})) - \tilde{T}(\mathbf{r}_k(\mathbf{X}))\right|^2 < \eta.
\end{aligned}
$$

Computation of A_{n1}. Denote by μ the distribution of $\mathbf{X}$. Then,

$$\begin{aligned}
A_{n1} &= \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|\tilde{T}(\mathbf{r}_k(\mathbf{X}_i)) - T(\mathbf{r}_k(\mathbf{X}_i))|^2\right] \\
&= \ell\,\mathbb{E}\left[\frac{\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_1)|\le\varepsilon_\ell\}}}{\sum_{j=1}^{\ell}\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_j)|\le\varepsilon_\ell\}}}|\tilde{T}(\mathbf{r}_k(\mathbf{X}_1)) - T(\mathbf{r}_k(\mathbf{X}_1))|^2\right]. \\
&= \ell\int|\tilde{T}(\mathbf{r}_k(\mathbf{u})) - T(\mathbf{r}_k(\mathbf{u}))|^2 \\
\times\mathbb{E}&\left[\int\frac{\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x})-r_{k,m}(\mathbf{u})|\le\varepsilon_\ell\}}}{\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x})-r_{k,m}(\mathbf{u})|\le\varepsilon_\ell\}}+\sum_{j=2}^{\ell}\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x})-r_{k,m}(\mathbf{X}_j)|\le\varepsilon_\ell\}}}\mathrm{d}\mu(\mathbf{x})\right]\mathrm{d}\mu(\mathbf{u}).
\end{aligned}$$

Let us prove that

$$\begin{aligned}
A'_{n1} &= \mathbb{E}\left[\int\frac{\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x})-r_{k,m}(\mathbf{u})|\le\varepsilon_\ell\}}}{\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x})-r_{k,m}(\mathbf{u})|\le\varepsilon_\ell\}}+\sum_{j=2}^{\ell}\mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{x})-r_{k,m}(\mathbf{X}_j)|\le\varepsilon_\ell\}}}\mathrm{d}\mu(\mathbf{x})\right] \\
&\le \frac{2^M}{\ell}.
\end{aligned}$$

To this aim, observe that

$$\begin{aligned}
A'_{n1} &= \mathbb{E}\left[\int\frac{\mathbf{1}_{\{\mathbf{x}\in\bigcap_{m=1}^{M}r_{k,m}^{-1}([r_{k,m}(\mathbf{u})-\varepsilon_\ell,r_{k,m}(\mathbf{u})+\varepsilon_\ell])\}}}{1+\sum_{j=2}^{\ell}\mathbf{1}_{\{\mathbf{X}_j\in\bigcap_{m=1}^{M}r_{k,m}^{-1}([r_{k,m}(\mathbf{x})-\varepsilon_\ell,r_{k,m}(\mathbf{x})+\varepsilon_\ell])\}}}\mathrm{d}\mu(\mathbf{x})\right] \\
&= \mathbb{E}\left[\int\frac{\mathbf{1}_{\{\mathbf{x}\in\bigcup_{(a_1,\ldots,a_M)\in\{1,2\}^M}r_{k,1}^{-1}(I_{n,1}^{a_1}(\mathbf{u}))\cap\ldots\cap r_{k,M}^{-1}(I_{n,M}^{a_M}(\mathbf{u}))\}}}{1+\sum_{j=2}^{\ell}\mathbf{1}_{\{\mathbf{X}_j\in\bigcap_{m=1}^{M}r_{k,m}^{-1}([r_{k,m}(\mathbf{x})-\varepsilon_\ell,r_{k,m}(\mathbf{x})+\varepsilon_\ell])\}}}\mathrm{d}\mu(\mathbf{x})\right] \\
&\le \sum_{p=1}^{2^M}\mathbb{E}\left[\int\frac{\mathbf{1}_{\{\mathbf{x}\in R_n^p(\mathbf{u})\}}}{1+\sum_{j=2}^{\ell}\mathbf{1}_{\{\mathbf{X}_j\in\bigcap_{m=1}^{M}r_{k,m}^{-1}([r_{k,m}(\mathbf{x})-\varepsilon_\ell,r_{k,m}(\mathbf{x})+\varepsilon_\ell])\}}}\mathrm{d}\mu(\mathbf{x})\right].
\end{aligned}$$

Here, $I_{n,m}^1(\mathbf{u}) = [r_{k,m}(\mathbf{u}) - \varepsilon_\ell, r_{k,m}(\mathbf{u})]$, $I_{n,m}^2(\mathbf{u}) = [r_{k,m}(\mathbf{u}), r_{k,m}(\mathbf{u}) + \varepsilon_\ell]$, and $R_n^p(\mathbf{u})$ is the p-th set of the form $r_{k,1}^{-1}(I_{n,1}^{a_1}(\mathbf{u})) \cap \ldots \cap r_{k,M}^{-1}(I_{n,M}^{a_M}(\mathbf{u}))$ assuming that they have been ordered using the lexicographic order of $(a_1, \ldots, a_M)$.

Next, note that

$$\mathbf{x} \in R_n^p(\mathbf{u}) \Rightarrow R_n^p(\mathbf{u}) \subset \bigcap_{m=1}^{M} r_{k,m}^{-1}([r_{k,m}(\mathbf{x}) - \varepsilon_\ell, r_{k,m}(\mathbf{x}) + \varepsilon_\ell]).$$

To see this, just observe that, for all $m = 1, \ldots, M$, if $r_{k,m}(\mathbf{z}) \in [r_{k,m}(\mathbf{u}) - \varepsilon_\ell, r_{k,m}(\mathbf{u})]$, i.e., $r_{k,m}(\mathbf{u}) - \varepsilon_\ell \le r_{k,m}(\mathbf{z}) \le r_{k,m}(\mathbf{u})$, then, as $r_{k,m}(\mathbf{u}) - \varepsilon_\ell \le r_{k,m}(\mathbf{x}) \le r_{k,m}(\mathbf{u})$, one has $r_{k,m}(\mathbf{x}) - \varepsilon_\ell \le r_{k,m}(\mathbf{z}) \le r_{k,m}(\mathbf{x}) + \varepsilon_\ell$. Similarly, if $r_{k,m}(\mathbf{u}) \le r_{k,m}(\mathbf{z}) \le r_{k,m}(\mathbf{u}) + \varepsilon_\ell$, then $r_{k,m}(\mathbf{u}) \le r_{k,m}(\mathbf{x}) \le r_{k,m}(\mathbf{u}) + \varepsilon_\ell$ implies $r_{k,m}(\mathbf{x}) - \varepsilon_\ell \le r_{k,m}(\mathbf{z}) \le r_{k,m}(\mathbf{x}) + \varepsilon_\ell$. Consequently,

$$\begin{aligned}
A'_{n1} &\le \sum_{p=1}^{2^M} \mathbb{E}\left[\int \frac{\mathbf{1}_{\{\mathbf{x}\in R_n^p(\mathbf{u})\}}}{1+\sum_{j=2}^{\ell}\mathbf{1}_{\{\mathbf{X}_j\in R_n^p(\mathbf{u})\}}}\mathrm{d}\mu(\mathbf{x})\right]\\
&= \sum_{p=1}^{2^M} \mathbb{E}\left[\mathbb{E}\left[\frac{\mu\{R_n^p(\mathbf{u})\}}{1+\sum_{j=2}^{\ell}\mathbf{1}_{\{\mathbf{X}_j\in R_n^p(\mathbf{u})\}}}\Big|\mathcal{D}_k\right]\right]\\
&\le \sum_{p=1}^{2^M} \mathbb{E}\left[\frac{\mu\{R_n^p(\mathbf{u})\}}{\ell\mu\{R_n^p(\mathbf{u})\}}\right]\\
&\le \frac{2^M}{\ell}
\end{aligned}$$

(by the first statement of Lemma 4.1). Thus, returning to A_{n1}, we obtain

$$A_{n1} \le 2^M \mathbb{E}\left|\tilde{T}(\mathbf{r}_k(\mathbf{X}) - T(\mathbf{r}_k(\mathbf{X})))\right|^2 < 2^M\eta.$$

Computation of A_{n2}. For any $\delta > 0$, write

$$\begin{aligned}
A_{n2} &= \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|\tilde{T}(\mathbf{r}_k(\mathbf{X}_i)) - \tilde{T}(\mathbf{r}_k(\mathbf{X}))|^2\right]\\
&= \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|\tilde{T}(\mathbf{r}_k(\mathbf{X}_i)) - \tilde{T}(\mathbf{r}_k(\mathbf{X}))|^2 \mathbf{1}_{\bigcup_{m=1}^M\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_i)|>\delta\}}\right]\\
&\quad + \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|\tilde{T}(\mathbf{r}_k(\mathbf{X}_i)) - \tilde{T}(\mathbf{r}_k(\mathbf{X}))|^2 \mathbf{1}_{\bigcap_{m=1}^M\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_i)|\le\delta\}}\right]\\
&\le 4\sup_{\mathbf{u}\in\mathbb{R}^d}|\tilde{T}(\mathbf{r}_k(\mathbf{u}))|^2 \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})\mathbf{1}_{\bigcup_{m=1}^M\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_i)|>\delta\}}\right] && (4.6)\\
&\quad + \left(\sup_{\mathbf{u},\mathbf{v}\in\mathbb{R}^d, \bigcap_{m=1}^M\{|r_{k,m}(\mathbf{u})-r_{k,m}(\mathbf{v})|\le\delta\}} |\tilde{T}(\mathbf{r}_k(\mathbf{v})) - \tilde{T}(\mathbf{r}_k(\mathbf{u}))|\right)^2. && (4.7)
\end{aligned}$$

With respect to the term (4.6), if $\delta > \varepsilon_\ell$, then

$$\begin{aligned}
&\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})\mathbf{1}_{\bigcup_{m=1}^M\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_i)|>\delta\}}\\
&\quad = \sum_{i=1}^{\ell} \frac{\mathbf{1}_{\bigcap_{m=1}^M\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_i)|\le\varepsilon_\ell\}}\mathbf{1}_{\bigcup_{m=1}^M\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_i)|>\delta\}}}{\sum_{j=1}^{\ell}\mathbf{1}_{\bigcap_{m=1}^M\{|r_{k,m}(\mathbf{X})-r_{k,m}(\mathbf{X}_j)|\le\varepsilon_\ell\}}}\\
&\quad = 0.
\end{aligned}$$

It follows that, for all $\delta > 0$, this term converges to 0 as ℓ tends to infinity. On the other hand, letting $\delta \to 0$, we see that the term (4.7) tends to 0 as well, by uniform continuity of $\tilde{T}$. Hence, A_{n2} tends to 0 as ℓ tends to infinity. Letting finally η go to 0, we conclude that A_n vanishes as ℓ tends to infinity. □

Proposition 4.2. *Under the assumptions of Proposition 2.1,*

$$\lim_{\ell\to\infty} \mathbb{E}\left|\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})(Y_i - T(\mathbf{r}_k(\mathbf{X}_i)))\right|^2 = 0.$$

Proof of Proposition 4.2.

$$\begin{aligned}
&\mathbb{E}\left|\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})(Y_i - T(\mathbf{r}_k(\mathbf{X}_i)))\right|^2 \\
&\quad = \sum_{i=1}^{\ell}\sum_{j=1}^{\ell} \mathbb{E}[W_{n,i}(\mathbf{X})W_{n,j}(\mathbf{X})(Y_i - T(\mathbf{r}_k(\mathbf{X}_i)))(Y_j - T(\mathbf{r}_k(\mathbf{X}_j)))] \\
&\quad = \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}^2(\mathbf{X})|Y_i - T(\mathbf{r}_k(\mathbf{X}_i))|^2\right] \\
&\quad = \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}^2(\mathbf{X})\sigma^2(\mathbf{r}_k(\mathbf{X}_i))\right],
\end{aligned}$$

where

$$\sigma^2(\mathbf{r}_k(\mathbf{x})) = \mathbb{E}[|Y - T(\mathbf{r}_k(\mathbf{X}))|^2|\mathbf{r}_k(\mathbf{x})].$$

For any $\eta > 0$, using again Györfi et al. (2002, Theorem A.1), σ^2 can be approximated in an L^1 sense by a continuous function with compact support $\tilde{\sigma}^2$, i.e.,

$$\mathbb{E}|\tilde{\sigma}^2(\mathbf{r}_k(\mathbf{X})) - \sigma^2(\mathbf{r}_k(\mathbf{X}))| < \eta.$$

Thus

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}^2(\mathbf{X})\sigma^2(\mathbf{r}_k(\mathbf{X}_i))\right] \\
&\quad \le \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}^2(\mathbf{X})\tilde{\sigma}^2(\mathbf{r}_k(\mathbf{X}_i))\right] + \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}^2(\mathbf{X})|\sigma^2(\mathbf{r}_k(\mathbf{X}_i)) - \tilde{\sigma}^2(\mathbf{r}_k(\mathbf{X}_i))|\right] \\
&\quad \le \sup_{\mathbf{u}\in\mathbb{R}^d} |\tilde{\sigma}^2(\mathbf{r}_k(\mathbf{u}))|\mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}^2(\mathbf{X})\right] \\
&\qquad + \mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|\sigma^2(\mathbf{r}_k(\mathbf{X}_i)) - \tilde{\sigma}^2(\mathbf{r}_k(\mathbf{X}_i))|\right].
\end{aligned}$$

With the same argument as for A_{n1}, we obtain

$$\mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X})|\sigma^2(\mathbf{r}_k(\mathbf{X}_i)) - \tilde{\sigma}^2(\mathbf{r}_k(\mathbf{X}_i))|\right] \leq 2^M \eta.$$

Therefore, it remains to prove that $\mathbb{E}\left[\sum_{i=1}^{\ell} W_{n,i}^2(\mathbf{X})\right] \to 0$ as $\ell \to \infty$. To this aim, fix $\delta > 0$, and note that

$$\begin{aligned}
\sum_{i=1}^{\ell} W_{n,i}^2(\mathbf{X}) &= \frac{\sum_{i=1}^{\ell} \mathbf{1}_{\bigcap_{m=1}^M \{|r_{k,m}(\mathbf{X}) - r_{k,m}(\mathbf{X}_i)| \leq \varepsilon_\ell\}}}{\left(\sum_{j=1}^{\ell} \mathbf{1}_{\bigcap_{m=1}^M \{|r_{k,m}(\mathbf{X}) - r_{k,m}(\mathbf{X}_j)| \leq \varepsilon_\ell\}}\right)^2} \\
&\leq \min\left\{\delta, \frac{1}{\sum_{i=1}^{\ell} \mathbf{1}_{\bigcap_{m=1}^M \{|r_{k,m}(\mathbf{X}) - r_{k,m}(\mathbf{X}_i)| \leq \varepsilon_\ell\}}}\right\} \\
&\leq \delta + \frac{\mathbf{1}_{\left\{\sum_{i=1}^{\ell} \mathbf{1}_{\bigcap_{m=1}^M \{|r_{k,m}(\mathbf{X}) - r_{k,m}(\mathbf{X}_i)| \leq \varepsilon_\ell\}} > 0\right\}}}{\sum_{i=1}^{\ell} \mathbf{1}_{\bigcap_{m=1}^M \{|r_{k,m}(\mathbf{X}) - r_{k,m}(\mathbf{X}_i)| \leq \varepsilon_\ell\}}}.
\end{aligned}$$

To complete the proof, we have to establish that the expectation of the right-hand term tends to 0. Denoting by I an arbitrary interval on the real line, we have

$$\begin{aligned}
&\mathbb{E}\left[\frac{\mathbf{1}_{\left\{\sum_{i=1}^{\ell} \mathbf{1}_{\left\{\mathbf{X}_i \in \bigcap_{m=1}^M r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell])\right\}} > 0\right\}}}{\sum_{i=1}^{\ell} \mathbf{1}_{\left\{\mathbf{X}_i \in \bigcap_{m=1}^M r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell])\right\}}}\right] \\
&\leq \mathbb{E}\left[\frac{\mathbf{1}_{\left\{\sum_{i=1}^{\ell} \mathbf{1}_{\left\{\mathbf{X}_i \in \bigcap_{m=1}^M r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell])\right\}} > 0\right\}} \mathbf{1}_{\left\{\mathbf{X} \in \bigcap_{m=1}^M r_{k,m}^{-1}(I)\right\}}}{\sum_{i=1}^{\ell} \mathbf{1}_{\left\{\mathbf{X}_i \in \bigcap_{m=1}^M r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell])\right\}}}\right] \\
&\quad + \mu\Big(\bigcup_{m=1}^M r_{k,m}^{-1}(I^c)\Big) \\
&= \mathbb{E}\left[\mathbb{E}\left[\frac{\mathbf{1}_{\left\{\sum_{i=1}^{\ell} \mathbf{1}_{\left\{\mathbf{X}_i \in \bigcap_{m=1}^M r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell])\right\}} > 0\right\}} \mathbf{1}_{\left\{\mathbf{X} \in \bigcap_{m=1}^M r_{k,m}^{-1}(I)\right\}}}{\sum_{i=1}^{\ell} \mathbf{1}_{\left\{\mathbf{X}_i \in \bigcap_{m=1}^M r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell])\right\}}} \Bigg| \mathcal{D}_k, \mathbf{X}\right]\right] \\
&\quad + \mu\Big(\bigcup_{m=1}^M r_{k,m}^{-1}(I^c)\Big) \\
&\leq \frac{2}{(\ell+1)} \mathbb{E}\left[\frac{\mathbf{1}_{\left\{\mathbf{X} \in \bigcap_{m=1}^M r_{k,m}^{-1}(I)\right\}}}{\mu(\bigcap_{m=1}^M r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell]))}\right] \\
&\quad + \mu\Big(\bigcup_{m=1}^M r_{k,m}^{-1}(I^c)\Big).
\end{aligned}$$

The last inequality arises from the second statement of Lemma 4.1. By an appropriate choice of I, the second term on the right-hand side can be made as small as desired. Regarding the first term, there exists a finite number N_ℓ of points $\mathbf{z}_1, \dots, \mathbf{z}_{N_\ell}$ such that

$$\bigcap_{m=1}^{M} r_{k,m}^{-1}(I) \subset \bigcup_{(j_1,\dots,j_M)\in\{1,\dots,N_\ell\}^M} r_{k,1}^{-1}(I_{n,1}(\mathbf{z}_{j_1})) \cap \dots \cap r_{k,M}^{-1}(I_{n,M}(\mathbf{z}_{j_M})),$$

where $I_{n,m}(\mathbf{z}_j) = [\mathbf{z}_j - \varepsilon_\ell/2, \mathbf{z}_j + \varepsilon_\ell/2]$. Suppose, without loss of generality, that the sets

$$r_{k,1}^{-1}(I_{n,1}(\mathbf{z}_{j_1})) \cap \dots \cap r_{k,M}^{-1}(I_{n,M}(\mathbf{z}_{j_M}))$$

are ordered, and denote by R_n^p the p-th among the $N_\ell^M = (\lceil |I|/\varepsilon_\ell \rceil)^M$ sets. Here $|I|$ denotes the length of the interval I and $\lceil x \rceil$ denotes the smallest integer greater than x. For all p,

$$\mathbf{x} \in R_n^p \Rightarrow R_n^p \subset \bigcap_{m=1}^{M} r_{k,m}^{-1}([r_{k,m}(\mathbf{x}) - \varepsilon_\ell, r_{k,m}(\mathbf{x}) + \varepsilon_\ell]).$$

Indeed, if $\mathbf{v} \in R_n^p$, then, for all $m = 1, \dots, M$, there exists $j \in \{1, \dots, N^n\}$ such that $r_{k,m}(\mathbf{v}) \in [\mathbf{z}_j - \varepsilon_\ell/2, \mathbf{z}_j + \varepsilon_\ell/2]$, that is $\mathbf{z}_j - \varepsilon_\ell/2 \le r_{k,m}(\mathbf{v}) \le \mathbf{z}_j + \varepsilon_\ell/2$. Since we also have $\mathbf{z}_j - \varepsilon_\ell/2 \le r_{k,m}(\mathbf{X}) \le \mathbf{z}_j + \varepsilon_\ell/2$, we obtain $r_{k,m}(\mathbf{X}) - \varepsilon_\ell \le r_{k,m}(\mathbf{v}) \le r_{k,m}(\mathbf{X}) + \varepsilon_\ell$. In conclusion,

$$\begin{aligned}
&\mathbb{E}\left[\frac{\mathbf{1}_{\{\mathbf{X}\in\bigcap_{m=1}^{M} r_{k,m}^{-1}(I)\}}}{\mu(\bigcap_{m=1}^{M} r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell]))}\right] \\
&\le \sum_{p=1}^{N_\ell^M} \mathbb{E}\left[\frac{\mathbf{1}_{\{\mathbf{X}\in R_n^p\}}}{\mu(\bigcap_{m=1}^{M} r_{k,m}^{-1}([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell]))}\right] \\
&\le \sum_{p=1}^{N_\ell^M} \mathbb{E}\left[\frac{\mathbf{1}_{\{\mathbf{X}\in R_n^p\}}}{\mu(R_n^p)}\right] \\
&= N_\ell^M \\
&= \left\lceil \frac{|I|}{\varepsilon_\ell} \right\rceil^M.
\end{aligned}$$

The result follows from the assumption $\lim_{\ell\to\infty} \ell\varepsilon_\ell^M = \infty$. ☐

Proposition 4.3. *Under the assumptions of Proposition 2.1,*

$$\lim_{\ell\to\infty} \mathbb{E}\left|\sum_{i=1}^{\ell}(W_{n,i}(\mathbf{X}) - 1)T(\mathbf{r}_k(\mathbf{X}))\right|^2 = 0.$$

Proof of Proposition 4.3. Since $|\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X}) - 1| \leq 1$, one has

$$\left|\sum_{i=1}^{\ell}(W_{n,i}(\mathbf{X}) - 1)T(\mathbf{r}_k(\mathbf{X}))\right|^2 \leq T^2(\mathbf{r}_k(\mathbf{X})).$$

Consequently, by Lebesgue's dominated convergence theorem, to prove the proposition, it suffices to show that $W_{n,i}(\mathbf{X})$ tends to 1 almost surely. Now,

$$\begin{aligned}
&\mathbb{P}\left(\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X}) \neq 1\right) \\
&\quad = \mathbb{P}\left(\sum_{i=1}^{\ell} \mathbf{1}_{\bigcap_{m=1}^{M}\{|r_{k,m}(\mathbf{X}) - r_{k,m}(\mathbf{X}_i))| \leq \varepsilon_\ell\}} = 0\right) \\
&\quad = \mathbb{P}\left(\sum_{i=1}^{\ell} \mathbf{1}_{\left\{\mathbf{X}_i \in \bigcap_{m=1}^{M} r_{k,m}^{-1}\left([r_{k,m}(\mathbf{X}) - \varepsilon_\ell, r_{k,m}(\mathbf{X}) + \varepsilon_\ell]\right)\right\}} = 0\right) \\
&\quad = \int \mathbb{P}\left(\forall i = 1, \ldots, \ell, \mathbf{1}_{\left\{\mathbf{X}_i \in \bigcap_{m=1}^{M} r_{k,m}^{-1}\left([r_{k,m}(\mathbf{x}) - \varepsilon_\ell, r_{k,m}(\mathbf{x}) + \varepsilon_\ell]\right)\right\}} = 0\right) \mathrm{d}\mu(\mathbf{x}) \\
&\quad = \int \left[1 - \mu(\cap_{m=1}^{M} r_{k,m}^{-1}\left([r_{k,m}(\mathbf{x}) - \varepsilon_\ell, r_{k,m}(\mathbf{x}) + \varepsilon_\ell]\right))\right]^{\ell} \mathrm{d}\mu(\mathbf{x}).
\end{aligned}$$

Denote by I an arbitrary interval. Then,

$$\begin{aligned}
&\mathbb{P}\left(\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X}) \neq 1\right) \\
&\quad \leq \int \exp\left(-\ell\mu(\cap_{m=1}^{M} r_{k,m}^{-1}\left([r_{k,m}(\mathbf{x}) - \varepsilon_\ell, r_{k,m}(\mathbf{x}) + \varepsilon_\ell]\right))\right) \mathbf{1}_{\{\mathbf{x} \in \bigcap_{m=1}^{M} r_{k,m}^{-1}(I)\}} \mathrm{d}\mu(\mathbf{x}) \\
&\qquad + \mu\Big(\bigcup_{m=1}^{M} r_{k,m}^{-1}(I^c)\Big) \\
&\quad \leq \max_{\mathbf{u}} \mathbf{u}e^{-\mathbf{u}} \int \frac{\mathbf{1}_{\{\mathbf{x} \in \bigcap_{m=1}^{M} r_{k,m}^{-1}(I)\}}}{\ell\mu(\cap_{m=1}^{M} r_{k,m}^{-1}\left([r_{k,m}(\mathbf{x}) - \varepsilon_\ell, r_{k,m}(\mathbf{x}) + \varepsilon_\ell]\right))} \mathrm{d}\mu(\mathbf{x}) \\
&\qquad + \mu\Big(\bigcup_{m=1}^{M} r_{k,m}^{-1}(I^c)\Big).
\end{aligned}$$

Using the same arguments as in the proof of Proposition 4.2, the probability $\mathbb{P}\left(\sum_{i=1}^{\ell} W_{n,i}(\mathbf{X}) \neq 1\right)$ is bounded by $\frac{e^{-1}}{\ell}\left\lceil\frac{|I|}{\varepsilon_\ell}\right\rceil^M$. This bound vanishes as n tends to infinity since, by assumption, $\lim_{\ell \to \infty} \ell\varepsilon_\ell^M = \infty$. ☐

References

P. Alquier and G. Biau. Sparse single-index model. *Journal of Machine Learning Research*, 14:243–280, 2013.

J.-Y. Audibert. Aggregated estimators and empirical complexity for least square regression. *Annales de l'Institut Henri Poincaré (B) Probabilités et Statistique*, 40:685–736, 2004.

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

L. Birgé. Model selection via testing: An alternative to (penalized) maximum likelihood estimators. *Annales de l'Institut Henri Poincaré (B) Probabilités et Statistiques*, 42:273–325, 2006.

F. Bunea, A. B. Tsybakov, and M. H. Wegkamp. Aggregation and sparsity via ℓ_1-penalized least squares. In G. Lugosi and H. U. Simon, editors, *Proceedings of 19th Annual Conference on Learning Theory (COLT 2006), Lecture Notes in Artificial Intelligence*, volume 35, pages 379–391. Springer-Verlag, Berlin-Heidelberg, 2006.

F. Bunea, A. B. Tsybakov, and M. H. Wegkamp. Aggregation for gaussian regression. *The Annals of Statistics*, 35:1674–1697, 2007a.

F. Bunea, A. B. Tsybakov, and M. H. Wegkamp. Sparsity oracle inequalities for the Lasso. *Electronic Journal of Statistics*, 35:169–194, 2007b.

O. Catoni. *Statistical Learning Theory and Stochastic Optimization*. Lectures on Probability Theory and Statistics, Ecole d'Eté de Probabilités de Saint-Flour XXXI - 2001, Lecture Notes in Mathematics. Springer, 2004.

P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47:547–553, 2009.

E. Cule. *`ridge`: Ridge Regression with automatic selection of the penalty parameter*, 2012. URL http://CRAN.R-project.org/package=ridge. R package version 2.1-2.

A. Dalalyan and A. B. Tsybakov. Aggregation by exponential weighting, sharp PAC-Bayesian bounds and sparsity. *Machine Learning*, 72:39–61, 2008.

B. Guedj. *`COBRA`: COmBined Regression Alternative*, March 2013. URL http://cran.r-project.org/web/packages/COBRA/index.html. R package version 0.99.

B. Guedj and P. Alquier. PAC-Bayesian estimation and prediction in sparse additive models. *Electronic Journal of Statistics*, 7:264–291, January 2013.

L. Györfi, M. Kohler, A. Krzyżak, and H. Walk. *A Distribution-Free Theory of Nonparametric Regression*. Springer, 2002.

T. Hastie and B. Efron. *lars: Least Angle Regression, Lasso and Forward Stagewise*, 2012. URL http://CRAN.R-project.org/package=lars. R package version 1.1.

A. Juditsky and A. Nemirovski. Functional aggregation for nonparametric estimation. *The Annals of Statistics*, 28:681–712, 2000.

J. Knaus. *snowfall: Easier cluster computing (based on snow)*, 2010. URL http://CRAN.R-project.org/package=snowfall. R package version 1.84.

V. Koltchinskii. Sparsity in penalized empirical risk minimization. *Annales de l'Institut Henri Poincaré (B) Probabilités et Statistiques*, 45:7–57, 2009.

S. Li. *FNN: Fast Nearest Neighbor Search Algorithms and Applications*, 2012. URL http://CRAN.R-project.org/package=FNN. R package version 0.6-4.

A. Liaw and M. Wiener. Classification and regression by randomForest. *R News*, 2:18–22, 2002. URL http://CRAN.R-project.org/doc/Rnews/.

J. D. Malley, J. Kruppa, A. Dasgupta, K. G. Malley, and A. Ziegler. Probability machines: Consistent probability estimation using nonparametric learning machines. *Methods of Information in Medicine*, 51:74–81, January 2012.

L. Meier, S. A. van de Geer, and P. Bühlmann. High-dimensional additive modeling. *The Annals of Statistics*, 37:3779–3821, 2009.

M. Mojirsheibani. Combining classifiers via discretization. *Journal of the American Statistical Association*, 94:600–609, 1999.

A. Nemirovski. *Topics in Non-Parametric Statistics*. École d'Été de Probabilités de Saint-Flour XXVIII – 1998. Springer, 2000.

E. C. Polley and M. J. van der Laan. *SuperLearner: Super Learner Prediction*, 2012. URL http://CRAN.R-project.org/package=SuperLearner. R package version 2.0-9.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2012. URL http://www.R-project.org/.

B. Ripley. *tree: Classification and regression trees*, 2012. URL http://CRAN.R-project.org/package=tree. R package version 1.0-32.

A. B. Tsybakov. Optimal rates of aggregation. In B. Schölkopf and M. K. Warmuth, editors, *Computational Learning Theory and Kernel Machines*, Lecture Notes in Computer Science, pages 303–313. Springer-Verlag, 2003.

S. A. van de Geer. High dimensional generalized linear models and the Lasso. *The Annals of Statistics*, 36:614–645, 2008.

M. J. van der Laan, E. C. Polley, and A. E. Hubbard. Super learner. *Statistical Applications in Genetics and Molecular Biology*, 6, September 2007.

M. H. Wegkamp. Model selection in nonparametric regression. *The Annals of Statistics*, 31:252–273, 2003.

Y. Yang. Combining different procedures for adaptive regression. *Journal of Multivariate Analysis*, 74:135–161, 2000.

Y. Yang. Adaptive regression by mixing. *Journal of the American Statistical Association*, 96:574–588, 2001.

Y. Yang. Aggregating regression procedures to improve performance. *Bernoulli*, 10:25–47, 2004.

I.-C. Yeh. Modeling of strength of high performance concrete using artificial neural networks. *Cement and Concrete Research*, 28:1797–1808, 1998.

I.-C. Yeh. Modeling slump flow of concrete using second-order regressions and artificial neural networks. *Cement and Concrete Composites*, 29:474–480, 2007.