Published in final edited form as:

Stat Med. 2015 January 15; 34(1): 106–117. doi:10.1002/sim.6322.

Ensemble learning of inverse probability weights for marginal structural modeling in large observational datasets

Susan Gruber^{a,*}, Roger W. Logan^a, Inmaculada Jarrín^b, Susana Monge^b, and Miguel A. Hernán^{a,c,d}

^aDepartment of Epidemiology, Harvard School of Public Health, Boston, MA

^bNational Center of Epidemiology, Instituto de Salud Carlos III, Madrid, Spain

^cDepartment of Biostatistics, Harvard School of Public Health, Boston, MA

^dHarvard-MIT Division of Health Sciences and Technology, Boston, MA

Abstract

Inverse probability weights used to fit marginal structural models are typically estimated using logistic regression. However a data-adaptive procedure may be able to better exploit information available in measured covariates. By combining predictions from multiple algorithms, ensemble learning offers an alternative to logistic regression modeling to further reduce bias in estimated marginal structural model parameters. We describe the application of two ensemble learning approaches to estimating stabilized weights: super learning (SL), an ensemble machine learning approach that relies on V -fold cross validation, and an ensemble learner (EL) that creates a single partition of the data into training and validation sets. Longitudinal data from two multicenter cohort studies in Spain (CoRIS and CoRIS-MD) were analyzed to estimate the mortality hazard ratio for initiation versus no initiation of combined antiretroviral therapy among HIV positive subjects. Both ensemble approaches produced hazard ratio estimates further away from the null, and with tighter confidence intervals, than logistic regression modeling. Computation time for EL was less than half that of SL. We conclude that ensemble learning using a library of diverse candidate algorithms offers an alternative to parametric modeling of inverse probability weights when fitting marginal structural models. With large datasets, EL provides a rich search over the solution space in less time than SL with comparable results.

Keywords

ensemble learning; super learning; marginal structural model; inverse probability weighting; data-adaptive; longitudinal data

1. Introduction

Inverse probability (IP) weighting is a method to adjust for measured confounders. In practice, IP weighting is often used to estimate the parameters of marginal structural models (MSM) [1]. Estimating the correct IP weights is crucial for reducing bias, but poses two challenges: 1) identifying a set of baseline and time-dependent covariates sufficient to control for confounding, and 2) accurately modeling the distribution of treatment at each time point conditional on past treatment and covariate history. In this paper we evaluate the use of data-adaptive ensemble learning to improve the estimation of the conditional distribution of treatment given a fixed set of covariates without imposing distributional assumptions, or assumptions on the functional form. The goal is to harness additional information available in the covariates to further reduce bias in the effect estimate by performing a rich search over the solution space using a diverse library of prediction algorithms.

Unlike previous papers on this topic [2, 3], we explore trade-offs in the specification of the ensemble library by examining super learning, an ensemble predictive machine learning algorithm that relies on V-fold cross validation [4, 5], and an ensemble learning algorithm that avoids cross validation by partitioning the data into training and validation sets one time only. Understanding and mitigating the computational challenges associated with these choices are the main focus of the work presented here.

The paper is organized as follows. Section 2 describes a marginal structural model for hazard ratio estimation, and motivates the application of ensemble learning. Section 3 introduces super learning (SL), discusses computational tradeoffs associated with different user-specified options, and introduces a closely-related alternative ensemble approach, EL, that lessens the computational burden associated with a rich search over the same solution space. A simulation study presented in Section 4 demonstrates the impact of using SL and EL to estimate stabilized IP weights. The study illustrates that the potential for improving estimation depends upon the complexity of the true data generating process, the degree of confounding, and implementation choices that define the ensemble learner. In Section 5 to estimate the mortality hazard ratio for initiation versus no initiation of antiretroviral therapy in CoRIS, a cohort of HIV-positive individuals [6]. Parameter estimates and confidence intervals are compared with those obtained using logistic regression-based weights. The paper concludes with a discussion and suggestions for related research.

2. IP weighting and marginal structural models

We have previously fit a pooled logistic MSM to estimate the mortality hazard ratio for initiation vs. no initiation of combined antiretroviral therapy in a large multinational collaboration of cohort studies from Europe and the United States [7]. We use the following pooled logistic MSM to model the hazard ratio within strata defined by baseline covariate vector V , $\text{logit Pr}[Y(t+1) = 1 \mid Y(t) = 0, V] = \beta_0(t) + \beta_1 a(t) + \beta' V$; where $a(t)$ is 1 if treatment has been initiated by time t (zero otherwise), Y is a counterfactual indicator of death at time t under regime $\gamma = (a_0, a_1, \dots)$. When the conditional risk of the outcome is low in each time period, the odds ratio of a pooled logistic regression model closely

approximates the hazard ratio [8]. When the MSM is correctly specified the hazard ratio has a causal interpretation, otherwise the parameter can be interpreted as the projection of the truth onto this working MSM.

Under the assumptions of exchangeability and positivity [9], we can consistently estimate the parameters of this MSM by fitting a weighted logistic model in which each subject receives the time-varying stabilized IP weight

$$SW(t) = \prod_{m=0}^t \frac{f[A(m)|\bar{A}(m-1), V]}{f[A(m)|\bar{A}(m-1), \bar{L}(m)]}$$

where $A(m)$ is an indicator of initiation of treatment by time m , $(m-1)$ is the treatment history through time $m-1$, $(A(0), A(1), \dots, A(m-1))$, V is a vector of baseline covariates, and $\bar{L}(m)$ is the covariate history through time m .

Constructing the stabilized IP weights, $SW(t)$, requires separately estimating the numerator and denominator at each time point $m = 0$ to t , and calculating the cumulative product of this ratio. Consider the denominator of the treatment term, $f[A(m)|\bar{A}(m-1), \bar{L}(m)]$. This quantity can be estimated via a pooled logistic model for the probability of treatment initiation at or before time m conditional on past treatment and covariate history. When, as in our example, we know or assume that treatment is never discontinued once it has been started, the model can be restricted to person-times with $A(m-1) = 0$ (all other person-times have probability 1 of receiving treatment), and thus covariates for past treatment history are not needed in the model. To model covariate history, which grows over time, one needs to decide which components of the covariate history are sufficient to adjust for time-varying confounding. In this paper we will assume that the baseline values and the most recent values of the time-varying confounders are sufficient to adjust for history when included as covariates in the model. The numerator is estimated analogously by regressing treatment at time m on baseline covariates.

One also needs to decide the functional form of the covariates in the logistic model. Because the functional form of the relationship between confounders and treatment is typically unknown, any a priori specified parametric model is likely to be misspecified. An alternative is to estimate the probability using data-adaptive approaches such as recursive partitioning algorithms [10], boosted regression [11], or ensemble learning to simultaneously considers a collection, or library, of approaches [12, 13].

3. Ensemble learning in large datasets

Because data-adaptive methods perform differently under different data-generating scenarios, the single best estimation procedure to apply to data whose true underlying distribution is unknown cannot be pre-specified. An alternative that has a long history in the machine learning literature is to combine predictions from multiple models [14–16], or more generally from multiple predictive algorithms [12, 13]. Algorithms included in a prediction algorithm library can be any mixture of non-parametric, parametric and semi-parametric procedures, any of which could itself be data-adaptive. Commonly used

algorithms include logistic regression models, neural nets, and classifiers such as k -nearest neighbor algorithms [17], and boosted classification and regression trees (CART) [18–20]. When the response variable is a highly non-linear function of the covariates, procedures that avoid modeling the functional form directly might achieve more success than those that do not [10, 21].

A rich library of prediction algorithms weakens modeling assumptions and may improve estimation of the true conditional distribution of treatment when, as we will assume, the relevant portion of the treatment and confounder history is captured by measured covariates. However, the computational cost associated with a rich search over a high-dimensional solution space quickly becomes prohibitive when there are many observations. Restricting the library of algorithms to those that run quickly, e.g., a collection of parametric regression models, may be unfavorably trading off search quality and computation time. For example, including only logistic regression models that regress the dependent variable on different subsets of suspected confounders imposes parametric assumptions on a complex response surface, and precludes obtaining better results from a predictor that avoids those assumptions. Including an algorithm because it provides results that are interpretable from a domain perspective is reasonable. Should this algorithm happen to predict treatment assignment well, it will contribute greatly to the combination. However, restricting the library entirely to algorithms that are easily interpreted by domain experts squanders the opportunity to conduct a broader search over the solution space that includes non-intuitive transforms of the data.

In summary, in large datasets there is tension between carrying out a computationally intensive search over the solution space and the feasibility of such a search in practice. Computation time and space increase with the dimension of the data, the size of the dataset, and the number of algorithms in the library. We describe two approaches to ensemble learning, discuss strategies for resolving this tension, and demonstrate the utility of ensemble learning in an applied data analysis.

3.1. Super learning

SL is a general loss-based estimation approach to ensemble prediction that uses V -fold cross validation to assess the individual performance of prediction algorithms in an ensemble library, and combines these algorithms to produce an asymptotically optimal combination. The optimal combination converges to the true data model when the prediction algorithms search over the correct portion of the solution space, and otherwise to the minimizer of a loss function-based dissimilarity measure, $\mathcal{J}(\cdot)$, with respect to target parameter ψ , $E[\hat{\mathcal{J}}(\psi) - \mathcal{J}(\psi)]$ [4, 5, 22]. If one of the prediction algorithms converges at a parametric rate to the truth, SL will converge at almost the parametric rate of $\log n/n$. If none of the prediction algorithms converge at a parametric rate to the true model, SL will be asymptotically equivalent with an oracle algorithm that can accurately evaluate the loss function with respect to the truth [4].

The SL algorithm presented in [5] is summarized here in the context of estimating the conditional expectation, $E[A | X]$, where X is a set of covariates, A is a dependent variable,

and the data consists of n observations $O = (A, X)$. For example, in our setting A is the vector of treatment indicators, $A_i(t)$, across all subjects i , and all times t .

SL Algorithm—Initializations: Define the ensemble library $\mathcal{L} = \{Alg_1, \dots, Alg_K\}$, where K is the number of algorithms in the library

Step 1: Train each algorithm ($Alg_1, \dots, Alg_K$) using all n observations, and obtain predicted values,

$$Z' = \begin{bmatrix} \hat{A}_{11} & \dots & \hat{A}_{1K} \\ \vdots & \ddots & \vdots \\ \hat{A}_{n1} & \dots & \hat{A}_{nK} \end{bmatrix}_{n \times K}.$$

Step 2: Create V equal-sized partitions of the data indexed by $v \in 1, \dots, V$. For each v this defines two disjoint sets, training set $Tr(v)$ contains all observations *except* those in partition v , and corresponding validation set $Val(v)$, containing observations in partition v .

Step 3: For each v , train each algorithm on $Tr(v)$ and obtain predicted values for observations in $Val(v)$, yielding a matrix of predictions containing one predicted value per observation for each algorithm,

$$Z = \begin{bmatrix} \hat{A}_{11}^{Val} & \dots & \hat{A}_{1K}^{Val} \\ \vdots & \ddots & \vdots \\ \hat{A}_{n1}^{Val} & \dots & \hat{A}_{nK}^{Val} \end{bmatrix}_{n \times K}$$

Step 4: Propose a family of weighted combinations of candidate estimators indexed by

$\alpha, m(z|\alpha) = \sum_{k=1}^K \alpha_k \hat{A}_k$, subject to the constraints $\alpha_k \geq 0 \forall k$ and $\sum_{k=1}^K \alpha_k = 1$. Calculate the contribution of each algorithm to the final prediction ($\hat{\alpha}_1, \dots, \hat{\alpha}_K$) by determining $\hat{\alpha}$ that minimizes the cross-validated risk.

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^n (A_i - m(z_i|\alpha))^2$$

subject to the constraints, $\alpha_k \geq 0 \forall k$, and $\sum_{k=1}^K \alpha_k = 1$, where z_i is the vector of predicted values ($\hat{A}_{i1}, \dots, \hat{A}_{iK}$) for subject i from the i th row of prediction matrix Z .

Step 5: Evaluate the SL predictions

$$\hat{A}_{SL} = \sum_{k=1}^K \hat{\alpha}_k z'_k,$$

where z'_k is column k of the Z' prediction matrix obtained in Step 1.

Theory encourages a rich library containing many algorithms (polynomial in sample size) [4], yet running each of these algorithms V times on subsets of size $n - n/V$ can be time-consuming. Several alternatives exist when the empirical distribution of a subset of the data closely resembles that of all the data, as in many of today's "big data" problems. One obvious approach is to obtain fitted values for all observations from an SL predictor trained on only a subset of observations. Alternatively, V -fold cross validation can be avoided altogether by partitioning the data into a training set used to fit the algorithms in the library, and a single validation set (hold-out set), used to assess the goodness-of-fit of the models and establish the convex combination when estimating α in Step 4. We describe the second of these alternative ensemble learners (EL) next.

3.2. Ensemble learning using one hold-out set

Given unlimited time and computing resources one could exhaustively search the solution space to achieve an optimal estimate of the treatment probability for calculating IP weights. In practice, however, conducting a relatively sparse search over the solution space using a concise prediction algorithm library is the only realistic option. In large datasets the trade-off between the richness of the search and computation time required for V -fold cross validation can be improved upon by using only a single partition of the data into training and validation sets. Recall the data consists of n observations $O = (A, X)$, where A is the dependent variable and X is a set of covariates. The algorithm for an EL approach to estimating $E(A | X)$ is the following.

EL Algorithm—Initializations:

- Define the ensemble library $\mathcal{L} = \{Alg_1, \dots, Alg_K\}$, where K is the number of algorithms in the library
- Partition the data into a training set, Tr containing pn observations, and validation set, V , containing the remaining $(1 - p)n$ observations, where $0 < p < 1$ is a user-specified proportion.

Step 1: Train each algorithm ($Alg_1, \dots, Alg_K$) on observations in Tr

Step 2: Obtain predicted values from each algorithm for all observations in V .

$$Z = \begin{bmatrix} \hat{A}_{11} & \dots & \hat{A}_{1K} \\ \vdots & \ddots & \vdots \\ \hat{A}_{(1-p)n1} & \dots & \hat{A}_{(1-p)nK} \end{bmatrix}_{(1-p)n \times K}$$

Step 3: Propose a family of weighted combinations of candidate estimators indexed by

$\alpha, m(z|\alpha) = \sum_{k=1}^K \alpha_k \hat{A}_k$, subject to the constraints $\alpha_k \geq 0 \forall k$ and $\sum_{k=1}^K \alpha_k = 1$. Calculate the contribution of each algorithm to the final prediction ($\alpha_1, \dots, \alpha_K$) by determining $\hat{\alpha}$ that minimizes the cross-validated risk.

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^{(1-p)n} (A_i - m(z_i|\alpha))^2$$

subject to the constraints, $\alpha_k \geq 0 \forall k$, and $\sum_{k=1}^K \alpha_k = 1$, where z_i is the vector of predicted values $(\hat{A}_{i1}^{Val}, \dots, \hat{A}_{iK}^{Val})$ for subject i from the i th row of prediction matrix Z .

Step 4: Obtain predicted values for all n observations after re-training each algorithm having a non-zero contribution. There are $M = \sum_{k=1}^K I(\hat{\alpha}_k > 0)$ such algorithms.

$$Z' = \begin{bmatrix} \hat{A}_{11}^{Val} & \dots & \hat{A}_{1M}^{Val} \\ \vdots & \ddots & \vdots \\ \hat{A}_{n1}^{Val} & \dots & \hat{A}_{nM}^{Val} \end{bmatrix}_{n \times M}$$

Step 5: Evaluate the EL predictions

$$\hat{A}_{EL} = \sum_{m=1}^M \hat{\alpha}_m z'_m,$$

where z'_m is column m of the Z' prediction matrix.

The optimal choice of partition size, p , is not known. Increasing the size of the validation set by increasing p provides better estimates of the loss function, but decreasing the size of the training set reduces the goodness of the fit. As with selecting the number of folds for V -fold cross validation, the optimal partition size depends on the underlying data distribution, and becomes more robust as n approaches infinity [13].

Reduced computation time is the key difference between EL and SL. However, note that the Z matrix of predicted values has dimension $(1-p)n \times K$ in the EL algorithm, versus $n \times K$ in the SL algorithm. This decrease in efficiency is part of the trade-off that we accept in exchange for an overall reduction in computation time. The time savings realized by avoiding an additional $V-1$ repetitions of model fitting required by SL's V -fold cross validation scheme can be spent exploring a larger portion of the solution space by running additional prediction algorithms in a more diversified library. Notice, too, that Step 1 of the SL procedure has been re-ordered as Step 4 in the EL algorithm. Instead of fitting each algorithm in the library on all observations, EL was implemented to fit only the (typically small) subset of algorithms contributing to the final fit, the M algorithms having a non-zero contribution to the convex combination on all observations. The SL algorithm could be similarly modified without any loss in performance.

While SL's V -fold cross validation scheme is preferred to EL's hold-out set approach for small datasets, the hold-out set approach is valid precisely when a large number of

observations in the dataset might discourage an analyst planning to use SL from adding computationally intensive algorithms to the library. We describe the integration of ensemble learning into the analysis of longitudinal data, and examine the performance of these algorithms in simulation studies and when used to estimate the mortality hazard ratio for initiation of antiretroviral therapy in a cohort of HIV-positive subjects.

4. Simulation study

A simulation study was conducted to illustrate the impact that using ensemble learning to estimate stabilized weights can have on the bias and variance of hazard ratio estimates. The directed acyclic graph in Figure 1 depicts causal relationships among an unmeasured baseline covariate, U , a continuous time-varying covariate, L , measured at baseline and again at time $t = 1$, binary treatment indicators, $A(0)$ and $A(1)$, and binary event indicators $Y(1)$ and $Y(2)$. For simplicity, we simulated data that differs from actual HIV settings in two important ways: treatment $A(0)$ does not affect subsequent treatment $A(1)$, and the confounders $L(0)$ and $L(1)$ are not affected by previous treatment. The lack of an effect of treatment on confounders implies that IP weighting is not necessary for an appropriate adjustment for time-varying confounding, but this point is tangential to the problems discussed in this paper.

Data were generated under two scenarios that differ only in the formulas used to generate treatment probabilities. Data for Scenario I were generated as follows,

$$\begin{aligned} U &\sim \text{Bern}(0.5) \\ L(0) &= 0.1 + 0.6U + \varepsilon_1 \\ P[A(0)=1|L(0)] &= \text{expit}(-0.4 + 0.6L(0)) \\ P[Y(1)=1|U] &= \text{expit}(-3.5 - 0.9U) \\ L(1) &= 0.5 + 0.02L(0) + 0.5U + 1.5A(0) + \varepsilon_2 \\ P[A(1)=1|L(0), L(1)] &= \text{expit}(-0.4 + 0.02L(0) + 0.58L(1)) \\ P[Y(2)=1|Y(1)=0, U] &= \text{expit}(-2.5 - 0.9U) \end{aligned}$$

where ε_1 and ε_2 are i.i.d. $N(0, 1)$. The treatment mechanism was modified for Scenario II, $P_{\text{ScenarioII}}[A(t) = 1 | L(0), L(t)] = \text{expit}(-0.5 - 0.1L(0) + 2I(L(t) < 0.5) - 2I(L(t) > 1.5))$. All remaining variables were generated the same as in Scenario I. As the DAG indicates, under both scenarios treatment has no effect on the outcome. Scenario I illustrates relative performance when the parametric main terms logistic regression model included in the prediction algorithm library is the correct model, and Scenario II illustrates relative performance when it is not.

Hazard ratios were estimated for 250 datasets containing $n = 500$ observations, and an additional 250 datasets containing $n = 10,000$ observations by fitting a weighted pooled logistic regression of the outcome at time $t + 1$ on the baseline value of L , $A(t)$, and the most recent value of treatment, $L(t)$. Six approaches to pooled estimation of the stabilized weights were compared, logistic regression using a main terms model, SL with $V = 3$, and three variants of EL, with the proportion of observations in the validation sample set to $p = (0.10, 0.25, 0.50)$. Analyses were run in R v2.15.2 [23] using the SuperLearner package [24]. R code is provided in a supplementary web file. A library of diverse prediction algorithms was

made available to SL and EL for estimating the numerator and denominator of the stabilized weights: a main terms logistic regression model, k-nearest neighbors, with $k = 21$, [17], a neural net algorithm [17] with one hidden layer containing 2 nodes, and boosted CART, with the number of trees set to 5,000, shrinkage set to 0.005, no interactions, and the best tree selected using 3-fold cross validation [19]. Covariate U was not made available to any prediction algorithm. $L(0)$ was available to procedures estimating the numerator of the stabilized weights, and $(L(0), L(t))$ was available for estimating the denominator. This data generation scheme allows us to examine the effect of sample size on estimator performance, in particular the effect of partition size, p , on EL estimates, and the choice of V for super learning. There is no single best choice of p or V . Performance varies with the underlying data distribution and the stability of prediction algorithms in the library [13].

4.1. Simulation study results

Table 1 shows the bias, variance, and mean squared error of the hazard ratio estimate for Scenarios I and II at sample sizes, $n = 500$ and $n = 10,000$. For Scenario I, using main terms logistic regression to estimate the weights provides unbiased estimation of the hazard ratio. Data adaptive procedures for estimating the stabilized weights perform equally well at both sample sizes, with minor differences. EL estimates are robust to the choice of p .

For Scenario II, a comparison of the unweighted estimates with those obtained using a misspecified logistic regression model for the IP weights shows that at the smaller sample size, the use of incorrect weights increased variance without reducing bias. At the larger sample size the increase in variance was offset by partial bias reduction, but the estimator is asymptotically biased due to misspecification. In contrast, SL reduces bias at both sample sizes, with variance approximately three and a half times greater than that of the unweighted estimator, but only 30% of that of the logistic regression-weighted estimator (15% when $n = 10,000$). Although the correct regression model was not explicitly included in the prediction algorithm library, the combination of algorithms used to predict the IP weights reduced bias to levels comparable to those of Scenario I. When $n = 500$ EL performed on a par with SL, and showed small sensitivity to the choice of p . At the larger sample size, EL with $p = 0.50$ is more biased and has larger variance than the other ensemble learners. The information gain at the larger sample size is evident in the performance of all the weighted estimators.

5. Data analysis

We estimated the mortality hazard ratio of initiation versus no initiation of combined antiretroviral therapy in two prospective studies conducted by The Cohort of the Spanish HIV Research Network (CoRIS, CoRIS-MD) (2013) [6]. The analyses were restricted to 7974 subjects, 18 years and older, not pregnant, who had baseline CD4 and RNA measurements within six months of each other. During 276,493 months of follow-up the mortality rate was 0.6 per 100 person-years. Approximately 45% of subjects initiate treatment during the observation period, with average follow-up of 2.8 years. A second product term was incorporated into the IP weights to account for a small amount of right censoring.

$$SW(t) = \prod_{m=0}^t \frac{f[A(m)|\bar{A}(m-1), V]}{f[A(m)|\bar{A}(m-1), \bar{L}(m)]} \prod_{m=0}^t \frac{f[C(m)=0|C(m-1)=0, \bar{A}(m-1), V]}{f[C(m)=0|C(m-1)=0, \bar{A}(m-1), \bar{L}(m)]}, \quad (1)$$

where $C(m)$ indicates monotone censoring status at time m , ($C(m) = 0$ indicates uncensored at m). For these data the contribution to the weights from censoring was approximately 1 for 98% of observations, with a maximum contribution of 2.1. Because the impact of the contribution from censoring on the stabilized weights is so small, we used fixed logistic regression models to estimate the numerator and denominator of the censoring term, and allocate all of our computational resources to estimating the contribution from treatment. We investigate several approaches to estimating this contribution, using a pre-specified logistic regression model, SL, and three variations of EL with partition size p set to (0.10, 0.25, 0.50).

The covariates used in the logistic regression model were carefully pre-selected based on domain knowledge [7]. These include a time-varying indicator for ever use of combined antiretroviral therapy, month of follow-up (restricted cubic splines with five knots) and the following baseline covariates: CD4 cell count (< 100, 100 to < 200, 200 to < 350, 350 to < 500, 500 cells/ μ l), HIV RNA level(< 10000, 10000 – 100000, > 100000 copies/ml), sex, transmission group (heterosexual, homosexual/ bisexual, injection drug use, other or unknown), calendar year (1997 – 1998, 1999 – 2000, 2001 – 2006, > 2006), age (< 35; 35 – 50, > 50 years), geographic origin (western developed countries, other or unknown), years since HIV diagnosis (< 1 year, 1 to < 5 years, > 5 years, or unknown), and an indicator of diagnosis with AIDS. These same covariates were subjected to a preliminary dimension reduction step before being made available to EL and SL. Because binary covariates that take on the same value for almost all observations are likely to contribute little to the fit, those having fewer than 1% variation in the sub-sampled data were dropped. To guard against excluding a covariate that might be highly predictive of a rare treatment, we simultaneously required that the correlation between an excluded covariate and treatment be less than 0.1. In these data no covariates met the exclusion criteria.

Analyses were run in R v2.15.2 [23] using the SuperLearner package [24]. R code is provided in a supplementary web file. A library containing five diverse prediction algorithms was made available to SL and EL for estimating the denominator: 1) a parsimonious main terms logistic regression model that included baseline age category and sex, current AIDS status, most recent CD4 and viral load category, and splines for time (glm.parsim), 2) a larger main terms logistic regression model that included all baseline and time-varying covariates (glm), 3) k-nearest neighbors, with $k = 81$ [17], 4) a neural net algorithm [17] with one hidden layer containing 25 nodes, and 5) boosted CART, with the number of trees set to 5000, shrinkage set to 0.005, no interactions, and the best tree selected using 3-fold cross validation. [19] The same library was used to estimate the numerator, substituting a parsimonious logistic regression model for the numerator that excluded time-dependent covariates. The number of SL cross validation folds was fixed at $V = 3$, following a rule of thumb suggesting $2 \leq V \leq 10$ [25], and was also set to $V = 3$ for a second SL estimator that uses a limited library consisting of the following collection of logistic regression models: 1) the numerator-specific (denominator-specific) parsimonious model, 2)

a main terms model with all covariates, 3–5) three additional main terms model that includes covariates that survive an associated pre-screening algorithms, namely a p -value on the test statistic associated with Pearson correlation between treatment and covariate less than α , with α taking on values (0.1, 0.2, 0.3), 6–9) four additional logistic regression models constructed using a forward selection procedure that searched over all main and second order terms, using Akaike Information Criteria (AIC) [17] to select the model. All covariates were supplied to the first of these four algorithms. The remaining three forward selection algorithms were given covariates that passed through the same three pre-screening algorithms just described. Although this library contains a total of nine prediction algorithms, in practice pre-screening at different p -value levels could yield identical sets of covariates, and the stepwise procedure could end up selecting identical models that produce identical fitted values. For this reason, and also because it searches over a restricted space of logistic regression models, we refer to this predictor as SL_{naive} , and the more diverse SL as SL_{rich} .

5.1. Results

Hazard ratio estimates based on untruncated stabilized weights ($\hat{\lambda}$), and stabilized weights truncated at 10 ($\hat{\lambda}_{trunc}$), are reported in Table 2. Closed-form variance estimates are not available for arbitrary prediction algorithm libraries, thus we provide bootstrapped quantile-based 95% confidence intervals (500 bootstrap samples), and computation time associated with estimating the contribution to the stabilized weights from treatment. The hazard ratio estimate from an unweighted outcome pooled logistic model was 2.58 (1.76 – 3.95) when the model included no covariates, and 1.11 (0.67 – 1.89) when the model included the baseline covariates. The hazard ratio estimated using a weighted outcome pooled logistic regression model was 0.98 whether a single logistic regression model or SL_{naive} was used to fit weights. However, the hazard ratio estimates from weighted outcome pooled logistic models were less than or equal to 0.8 when SL_{rich} and EL were used to fit the weights. Because of external evidence of the beneficial effect of combined antiretroviral therapy on mortality [26, 27], our findings suggest that even without modifying the adjustment set, weakening modeling assumptions further reduced bias due to residual confounding. Truncation had little effect on the hazard ratio estimates.

Computation time reported in Table 2 is installation-specific, but clearly illustrates that there are different ways to trade-off computation time and the quality of the search. In comparison with SL_{rich} using the five disparate learners (15.8 hours), computations involving the less aggressive SL_{naive} library were completed in less than half the time, but offered no improvement over the single logistic regression model in terms of reducing bias in the hazard ratio estimate. In contrast, EL with $p = 0.25$ or 0.5 ran in less than half the time of SL_{rich} , without sacrificing search quality.

Table 3 shows that differences among the disparate approaches manifest primarily in the tails of the distribution; mean weights are close to 1, and there is little difference in the 1st and 99th percentiles of the empirical distributions. The boxplots of the log of the stabilized weights in Figure 2 show that while each distribution is centered around 1 (0 on the log scale), the distributions estimated by SL_{rich} and each of the three EL algorithms are more

symmetric and have smaller variance than those resulting from logistic regression and SL_{naive} .

Figure 3 depicts the relative contributions to the SL_{rich} and EL predictions from each algorithm included in the diverse library, (*glm.parsim*, *glm*, *knn81*, *nnet20*, *CART*). There is some variation, but overall good agreement on the selection of algorithms with non-zero contributions and the relative size of each contribution. Two algorithms *glm*, *CART* clearly dominate the procedure for estimating the numerator, while *glm*, *knn81* make the largest contribution to the estimation of the denominator. In the company of these other algorithms neither *glm.parsim* nor *nnet20* are major contributors.

6. Discussion

We have shown that ensemble learning to model the conditional probability of treatment given known confounders can be advantageous. Hazard ratio estimates of the effect of initiation of combined antiretroviral therapy on mortality obtained using EL and SL_{rich} were further from the null than those obtained using the carefully crafted pooled logistic regression model, without increasing the width of the quantile-based 95% bootstrap confidence interval. These results suggest that future analyses of similar datasets would benefit from going beyond logistic regression when estimating inverse probability weights. In these data, EL with partition size $p = 0.25$ produced a hazard ratio estimate in good agreement with that obtained using a partition size of $p = 0.5$ and with SL_{rich} using all the data for 3-fold cross validation in approximately 40% of the time, and also in less time than SL_{naive} using a less diverse prediction algorithm library. This cannot be interpreted as evidence that 0.25 is the best partition size in all datasets. The degree of heterogeneity in the data plays a role in stability of estimates of the parameter of interest across a range of choices for tuning parameter p . As the data analysis results illustrate, algorithms having zero contribution to the convex combination (*glm.parsim*, *neural net* when estimating numerator or denominator, additionally *knn81* for the denominator) are consistently identified across all partition sizes.

In the simulation study SL 's V-fold cross validation approach proved slightly superior to EL at two out of three partition sizes at the smaller sample size. In Scenario I of the simulation study, where the main terms logistic regression model is correct, using SL or EL instead had little impact on the variance. At the larger sample size, SL was superior to EL with $p = 0.10$ and $p = 0.50$ in Scenario II. Scenario II shows that although ensemble learning sometimes increases the variance of an estimator, at other times the improved fit of the model leads to a reduction in variance.

The data analysis illustrated that at large sample sizes EL can reduce computation cost without sacrificing search quality. In the analysis of the CoRIS data the EL approach showed stability in the choice of algorithms across a range of partition sizes. Because typically only a few of the algorithms in the ensemble make non-zero contributions to the convex combination ($\hat{\alpha}_k > 0$), re-fitting these few selected learners on all observations is much less time consuming than fitting all learners on all observations. Optionally, the condition $\hat{\alpha}_k = 0$ can be relaxed by replacing 0 with some minimum threshold (ϵ), and either

1) equally distributing the algorithm's contribution across the remaining algorithms,

$\hat{\alpha}'_k = I(\hat{\alpha}_k \leq \varepsilon)0 + I(\hat{\alpha}_k > \varepsilon)\hat{\alpha}_k / \sum_{k=1}^K I(\hat{\alpha}_k > \varepsilon)\hat{\alpha}_k$, or 2) using the non-negative least squares algorithm to estimate the M' non-zero elements of $\hat{\alpha}'$ using a sub-matrix of Z restricted to columns indexed by k such that $\hat{\alpha}_k > \varepsilon$.

Theory indicates that SL and EL perform asymptotically as well as the optimal combination of algorithms comprising the library of prediction algorithms [25, 28], but how to choose algorithms to include in the library is a question that involves theoretical and practical considerations that restrict the search to a portion of the solution space. This paper presents an alternative to super learning that reduces the computational burden by creating only one partition of the data into training and validation sets in lieu of V -fold cross validation. In big data problems, for a fixed computational cost it is feasible to enrich the SL library and replace V -fold cross validation with a hold-out set approach to search more broadly over the solution space at a fixed computational cost. A drawback of both EL and SL is that using the bootstrap to estimate valid confidence intervals is quite time consuming. Confidence intervals for just one of the EL analyses presented here based on 500 bootstrap iterations requires approximately $500 * 7$ hours, or 146 straight days of continuous computing. Running the bootstraps in parallel will substantially reduce calendar time, but just how much time can be saved depends on the availability of computing nodes with sufficient memory, and is therefore installation-dependent.

This paper did not address alternative weighting schemes for combining treatment probability predictions that could reduce bias by dampening the effect of a few influential observations. We also did not discuss covariate balance as a method for choosing between competing models for treatment (propensity score models). How to carry out assessment of many time dependent covariates across a large number of time points is not yet well established in the literature. A previously proposed method, covariate balancing propensity score [29], begins to break down as support in the data diminishes over time. The lack of support is complete when treatment is continuous, and also when the MSM conditions on continuous baseline covariates, as in this paper. We also did not address the important question of confounder selection. The ideal confounder selection strategy would yield unbiased estimates with optimal variance, but how to do this in finite samples continues to be the subject of debate. An important open question is what aspects of the covariate history are relevant for addressing confounding, and what functional form best captures the relationships in the data. References [30, 31] present parametric data-adaptive procedures that combine propensity score and outcome regression modeling. The semi-parametric collaborative targeted maximum likelihood estimator includes a confounder selection procedure that attempts to optimally trade off bias and variance in the parameter estimate [32, 33]. An important next step in this research will be to combine a principled confounder selection strategy with ensemble learning.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgements

We thank Jamie Robins for helpful discussions. This research was supported by NIH grant R01-AI073127

References

1. Hernan MA, Brumback B, Robins JM. Marginal structural models to estimate the causal effect of zidovudine on the survival of HIV-positive men. *Epidemiology*. 2000; 11(5):561–570. [PubMed: 10955409]
2. Neugebauer R, Fireman B, Roy J, Raebel M, Nichols G, O'Connor P. Super learning to hedge against incorrect inference from arbitrary parametric assumptions in marginal structural modeling. *Journal of Clinical Epidemiology*. 2013; 66:S99–S109. [PubMed: 23849160]
3. Neugebauer R, Chandra M, Paredes A, Graham D, McCloskey C, Go A. A marginal structural modeling approach with super learning for a study on oral bisphosphonate therapy and atrial fibrillation. *Journal of Causal Inference*. 2013; 1:21–50.
4. van der Laan M, Polley E, Hubbard A. Super Learner. *Statistical Applications in Genetics and Molecular Biology*. 2007; 6(25)
5. Polley, E.; van der Laan, M. Technical Report. Berkeley: Division of Biostatistics, University of California; 2010. Super learner in prediction.
6. Caro-Murillo A, et al. Spanish cohort of naive HIV-infected patients (CoRIS): rationale, organization and initial results. *Enferm Infecc Microbiol Clin*. 2007; 25:3–4. [PubMed: 17261239]
7. Collaboration TH. The effect of combined antiretroviral therapy on the overall mortality of HIV-infected individuals. *AIDS*. 2010; 24(1)
8. D'Agostino R, Lee M, Belanger A, Cupples L, Anderson K, Kannel W. Relation of pooled logistic regression to time dependent Cox regression analysis: the Framingham Heart Study. *Statistics in Medicine*. 1990; 12:1501–1515. [PubMed: 2281238]
9. JM, R.; MA, H. Estimation of the causal effects of time-varying exposures. In: G, F.; M, D.; G, V.; G, M., editors. *Advances in longitudinal data analysis*. Boca Raton: Chapman and Hall/CRC Press; 2009. p. 553–599.
10. Lee B, Lessler J, Stuart E. Improved propensity score weighting using machine learning. *Statistics in Medicine*. 2009; 29:337–346. [PubMed: 19960510]
11. McCaffrey D, Ridgeway G, Morral A. Propensity score estimation with boosted regression for evaluating causal effects in observational studies. *Psychological Methods*. 2004; 9(4):403–425. [PubMed: 15598095]
12. Polikar R. Ensemble learning. *Scholarpedia*. 2009; 4(1):2776.
13. Arlot S, Celisse A. A survey of cross-validation procedures for model selection. *Statistic Surveys*. 2010; 4:40–79.
14. Ting K, Witten IH. Issues in stacked generalization. *Journal of Artificial Intelligence Research*. 1999; 10:271–289.
15. Wolpert D. Stacked generalization. *Neural Networks*. 1992; 5:241–259.
16. Breiman L. Stacked regression. *Machine Learning*. 1996; 24:49–64.
17. Venables, WN.; Ripley, BD. *Modern Applied Statistics with S*. 4th edn.. New York: Springer; 2002. URL <http://www.stats.ox.ac.uk/pub/MASS4>.
18. Breiman, L.; Friedman, JH.; Olshen, R.; Stone, CJ. *Classification and regression trees*. The Wadsworth statistics/probability series, Wadsworth International Group; 1984.
19. Ridgeway G, McCaffrey D, Morral A, Burgette L, Griffin B. Toolkit for Weighting and Analysis of Nonequivalent Groups: A Tutorial for the Twang Package. 2013 URL <http://CRAN.R-project.org/package=gbm>, R package.
20. McCaffrey D, Burgette L, Griffin B, Martin C. Toolkit for Weighting and Analysis of Nonequivalent Groups: A Tutorial for the Twang SAS Macro. 2014
21. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*. Springer Series in Statistics, Springer; 2002.

22. van der Vaart A, Dudoit S, van der Laan M. Oracle inequalities for multi-fold cross-validation. *Statistics and Decisions*. 2006; 24(3):351–371.
23. R Development Core Team. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing; 2012. URL <http://www.R-project.org>.
24. Polley E. SuperLearner: Super Learner Prediction. 2012 URL <http://CRAN.R-project.org/package=SuperLearner>, R package version 2.0-9.
25. Arlot S, Lerasle M. V-fold cross-validation and V-fold penalization in least-squares density estimation. eprint arXiv:1210.5830. 2012
26. Mocroft A, Ledergerber B, Katlama C, Kirk O, Reiss P, Monforte A, et al. Decline in the AIDS and death rates in the EuroSIDA study: an observational study. *Lancet*. 2003; 362:2229.
27. Babiker A, Bhaskaran K, Darbyshire J, Pezzotti P, Porter K, Walker A. Determinants of survival following HIV-1 seroconversion after the introduction of HAART. *Lancet*. 2003; 362:1267–1274. [PubMed: 14575971]
28. van der Laan, M.; Dudoit, S. Technical report. Berkeley: Division of Biostatistics, University of California; 2003 Nov. Unified cross-validation methodology for selection among estimators and a general cross-validated adaptive epsilon-net estimator: Finite sample oracle inequalities and examples.
29. Imai, K.; Ratkovic, M. Technical Report. Princeton University; 2013 May. Robust estimation of inverse probability weights for marginal structural models. URL <http://imai.princeton.edu/research/MSM.html>.
30. Vermeulen K, Vansteelandt S. Focused estimation of nuisance parameters in doubly robust inference. *J. Am. Statist. Assoc.* 2013 (under revision).
31. Zigler C, Watts K, Yeh R, Y W, BA C, F D. Model feedback in bayesian propensity score estimation. *Biometrics*. 2013
32. van der Laan M, Gruber S. Collaborative double robust penalized targeted maximum likelihood estimation. *The International Journal of Biostatistics*. 2010 Jan.6(1)
33. Gruber S, van der Laan M. An application of collaborative targeted maximum likelihood estimation in causal inference and genomics. *The International Journal of Biostatistics*. 2010; 6(1)

Figure 1.
Directed acyclic graph representation of simulated data for Scenarios I and II.

Figure 2. Distribution of $\log(\text{SW})$ estimated using logistic regression, SL_{naive} , SL_{rich} , EL.10, EL.25, EL.50. The horizontal line is at the desired mean value.

Figure 3.

Contribution of each algorithm in the library to the convex combination for the numerator (top) and denominator (bottom) of contribution to stabilized weights from treatment as determined by SL_{rich} and EL using different partition sizes.

Table 1

Bias and variance of hazard ratio estimates obtained over 250 Monte Carlo simulations, at sample size $n = 500$ and $n = 10,000$ using different stabilized weight estimators.

	$n = 500$		$n = 10,000$	
	Bias	Var	Bias	Var
Scenario I				
unweighted	0.187	0.201	0.110	0.008
logistic reg	0.086	0.184	-0.002	0.008
SL	0.086	0.179	-0.002	0.008
EL 0.10	0.088	0.186	-0.002	0.008
EL 0.25	0.089	0.183	-0.002	0.008
EL 0.50	0.081	0.176	-0.002	0.008
Scenario II				
unweighted	-0.171	0.099	-0.220	0.005
logistic reg	0.239	1.259	0.103	0.110
SL	0.101	0.373	0.002	0.016
EL 0.10	0.087	0.343	0.006	0.018
EL 0.25	0.118	0.428	0.001	0.019
EL 0.50	0.101	0.389	0.054	0.165

Table 2

Estimated hazard ratios $\hat{\lambda}$ (95% confidence interval) based on untruncated stabilized weights, and $\hat{\lambda}_{trunc}$ based on truncated stabilized weights estimated using a fixed logistic regression model, SL_{naive} , SL_{rich} , or EL with different partition sizes, and associated computation time.

	$\hat{\lambda}$ (95% CI)	$\hat{\lambda}_{trunc}$ (95% CI)	$SW(t)$ Computation Time
Logistic reg	0.98 (0.60 – 2.48)	0.98 (0.59 – 1.84)	19 sec
SL_{naive}	0.98 (0.50 – 1.72)	0.98 (0.51 – 1.71)	7.6 hrs
SL_{rich}	0.77 (0.37 – 1.59)	0.78 (0.40 – 1.56)	15.8 hrs
EL 0.1	0.69 (0.33 – 1.64)	0.70 (0.37 – 1.64)	6.0 hrs
EL 0.25	0.80 (0.40 – 1.60)	0.80 (0.41 – 1.57)	6.5 hrs
EL 0.5	0.80 (0.37 – 1.62)	0.80 (0.40 – 1.61)	7.3 hrs

Table 3

Summary of the empirical distribution of untruncated stabilized weights obtained using logistic regression modeling, $SL_{\eta^{naive}}$, SL, and EL with four different partition sizes.

	Stabilized Weight Distribution				
	Min	1st percentile	Mean	99th percentile	Max
Logistic Reg	$7.52e-06$	0.147	1.00	3.48	481
$SL_{\eta^{naive}}$	$1.72e-06$	0.148	1.01	3.74	574
SL	$1.79e-03$	0.159	1.03	4.18	385
EL 0.10	$2.31e-03$	0.157	1.02	4.20	386
EL 0.25	$1.74e-03$	0.159	1.02	4.26	256
EL 0.50	$1.22e-03$	0.155	1.02	4.02	360