# Two-stage super learner for predicting healthcare expenditures

**Abstract**
Healthcare utilization and associated costs have increased rapidly in recent years, making the study of healthcare expenditures an important area of public health research. Analysis of healthcare expenditure data is challenging due to heavily skewed distributions and zero-inflation. Myriad methods have been developed for analyzing cost data; however, a priori determination of an appropriate method is often difficult. Super-learning, a technique that considers an ensemble of methods for cost estimation, provides an interesting alternative for modeling healthcare expenditures. The super learner has demonstrated benefits over a single method in recent studies across many disciplines. In this work, we propose a two-stage super learner specifically designed for predicting zero-inflated expenditures. We demonstrate that the two-stage super learner has strong performance in predicting healthcare costs across a variety of cost distributions, in both real and simulated data.

**Keywords:** healthcare expenditure, zero-inflation, two-part model, super learning, cross-validation

## Introduction

The study of healthcare expenditures has become an important area in epidemiological and public health research. This is motivated by the growing interest in cost control and program evaluation, in view of adopting treatments and policies on the basis of cost-effectiveness. Statistical models are often used for modeling these expenditures either for the purpose of prediction or to examine interventions' effects of expenditures [1]. However, specific characterizations of expenditures, such as healthcare costs, length of stay, and utilization of health care services, often pose great challenges to statistical modeling [2]. Such data can exhibit a non-negligible point mass at zero due to the non-consumption of healthcare, in addition to substantial positive skewness arising because more severe impact patients often require myriad services due to clinical complications and comorbidities [3]. For example, in the United States, a small minority of the individuals account for a high proportion of health care costs. Berk & Monheit [4] report that 5% of the population accounts for the majority of health expenditures, a trend that has remained stable for decades.

A common goal in studies of healthcare utilization is to model the conditional mean for a measure of utilization $Y$ given demographic and diagnostic covariates $X$. In the remainder, for simplicity, we will take $Y$to be cost. This is useful not only as an end in itself for generating cost predictions for future subjects, but also as a means to drawing inference on the causal effect of a new health policy on costs [5, 6, 7]. Here, we focus on conditional means as opposed to medians given it's more sensitive to outliers and thus more sensitive to how estimators treat the skewness in the outcome and other statistical problems that are common in such data, e.g., zero-inflation [8]. Robust modeling of conditional mean costs is often challenging due to the presence of heavy right tails in the cost distribution. A common approach involves regression of log-transformed cost using ordinary least squares. The log-transformed outcome has decreased skewness and is often better approximated by a normal distribution than costs on the original scale. Duan's smearing estimator [9] can be used to map estimates from the log-scale back to the original cost scale. However, this approach is limited in settings with zero inflation, since the log of zero is undefined. An alternative approach is generalized linear models (GLM), which can accommodate skewness which extend the normal linear model framework to allow response variables that are not normally distributed. GLMs provide considerable flexibility and are often

found to fit healthcare expenditures well [10]. Other parametric and semi-parametric models have also been proposed including the Cox proportional hazards model [4, 10], the Tobit model, the Tweedie model and quantile-based models [11]. See Jones [3] for a detailed overview and comparison of several regression models for healthcare costs.

To better account for zero-inflation, researchers have utilized two-part models. In this approach, two models are specified: a model for the conditional probability of any cost, and a model for the conditional mean of costs amongst individuals with positive costs. Two-part GLMs with either a logit or probit link for the binary component and a Gamma distribution with log-link for the continuous component have been used in a variety of empirical work in health service research [12, 13, 14]. Often, the specification of the binary model is less important than the model for positive expenditures, which can alter results dramatically with different models [15]. Consequently, recent research has mainly focused on developing new models for the continuous part of the two-part model. The generalized Gamma distribution is a flexible choice with one scale and two shape parameters that has been shown to be relatively robust compared to the alternatives [8].

Despite the proliferation of sophisticated statistical methods for modeling healthcare expenditures, there is, unfortunately, no "one-size-fits-all" model [15]. Moreover, many proposed methods rely on parametric or semi-parametric models, which require the proper specification of the regression formula. This can be quite challenging in many contexts where there is limited prior information as to how patient characteristics relate to expenditures. This has motivated a movement towards machine learning techniques, accelerated by the ubiquity of "big data", e.g., through electronic medical records [16]. These techniques make fewer assumptions about underlying relationships and can flexibly adapt to the data, thereby providing more accurate cost predictions.

One particularly appealing approach in machine learning is super learning, also known as regression stacking [17]. Super Learning originated from “Stacked Generalization” (Wolpert, 1992 [18]), which is an approach to combine “lower-level” predictive algorithms into a “higher-level” model to increase predictive accuracy. Breiman later applied stacking in a regression context (“Stacked Regression”) and least squares with positive constraints on the higher-level model [17]. Super learning provides a generalization of these frameworks and strong theoretical guarantees have been established for the approach (van der Laan and Dudoit, 2003, [19]). Super learning entails positing a collection of potential methods and using cross-validation to learn the optimal way to combine predictions from these methods to achieve the most accurate prediction. The collection of candidate models could include parametric, semiparametric, or machine learning-based estimators. Under regularity conditions, the super learner has essentially the same large-sample predictive performance as an oracle estimator (i.e., the unknown, best-possible weighted combination among included algorithms). In this way, super learning provides a theoretically optimal means of combining estimators in the face of model uncertainty. Recently, Super Learner has shown benefits over using a single method in several healthcare studies including prediction of post-traumatic stress disorder based on traumatic experiences (Kessler and others, 2014, [20]), prediction of mortality, both in the elderly population [21] and in intensive care units (Pirracchio and others, 201, [22]), as well as prediction of plan payment risk adjustment for total annual healthcare expenditures (Rose, 2016, [16]).

The goal of the present work is to extend the super learner approach to better accommodate zero-inflated cost data. We propose a two-stage super learner that implements the super-learning technique under a two-part model framework: we define a set of candidate

methods for predicting the presence of any costs, as well as for the positive portion of the cost distribution. The full super learner library then consists of all pairwise combinations of the two. In this way, we are able to learn the optimal model for both components in terms of arriving at the most accurate predictions of costs. We compare the two-stage super learner with the super learner, along with individual algorithms designed in other studies of healthcare cost and evaluate the benefits of using this Two-stage Super Learner via Monte Carlo simulations under various data generating processes. In addition, two empirical analyses are performed with the data from 2016-2017 Medical Expenditure Panel Survey (MEPS) and Back pain Outcomes using Longitudinal Data (BOLD) project. In both cases, we find that the two-stage super learner improves not only on individual algorithms for predicting costs, but also on a typical super learner that combines these algorithms.

## Methods

### *MEPS Data*

The expenditure data for empirical analysis were drawn from the Medical Expenditure Panel Survey (MEPS) from 2016 to 2017. The MEPS is a national survey on the financing and use of medical care of families and individuals, their medical providers (doctors, hospitals, pharmacies, etc.) and employers across the United States. The Agency for Healthcare Research and Quality (AHRQ) has collected MEPS data every year since 1996. The MEPS provides an accurate measure of healthcare expenditures, as well as detailed measures of health status and other observable characteristics correlated with expenditures. Participating household components, containing families and individuals, were drawn from a nationally representative subsample of households that participated in the prior year's National Health Interview Survey. Household respondents provided demographic information, health status, self-reported medical conditions, medical expenditure and utilization, health insurance coverage and access to care for medical events. For some individuals, self-reported medical expenditures are supplemented with information from medical providers and insurers. MEPS uses a complex survey design including weighting, stratification, clustering and disproportionate sampling to create nationally representative annual estimates for U.S. civilian and noninstitutionalized population.

In this study, 2016 MEPS data were used as a training sample to fit each candidate estimator and super learner while 2017 MEPS data were used as a testing sample to evaluate the performance of each estimator and super learner. The total annual healthcare expenditures were the outcome of interest and the covariates involve demographics, medical conditions and insurance characteristics collected on participants. The total annual healthcare expenditures include out-of-pocket payments and third-party payments from all sources but exclude insurance premiums. Considering the purpose of the study was to evaluate the benefits of the two-stage super learner rather than estimating national statistics, we disregard the sampling weights and survey design information of MEPS data to reduce data complexity. Additionally, we restrict the sample to include only adults and exclude observations with missing data in terms of outcome and considered covariates (refused, not asked, etc.). The final sample contained 10925 observations for training and 10815 observations for testing.

### *BOLD Data*

The Back Pain Outcomes using Longitudinal Data (BOLD) project established a large, community-based registry of patients aged 65 years and older who presented with primary care visits for a new episode of back pain (no prior visits to a health care provider for back pain care within 6 months) during March 2011 to March 2013 at three integrated healthcare systems: Harvard Vanguard, Henry Ford Health System, and Kaiser Permanente Northern

California. Details of BOLD registry are described in the BOLD Study Protocol [23]. The BOLD data comes primarily from patient self-reported questionnaires and electronic medical records (EMR). Expenditures in BOLD were calculated as total relative value units (RVU), a measure of value used in the US medicare reimbursement formula for physician services [24].

In this study, patient self-reported questionnaire responses were used to predict future expenditures (as measured by RVUs) in several categories. Specifically, the covariates include the following measures from patient self-reported questionnaires collected at baseline: (1) Socio-demographics (age, sex, race, ethnicity, education, employment status, etc.); (2) Pain-related characteristics (back/leg pain duration, back/leg pain intensity, modified Roland-Morris Disability Questionnaire [25], Brief Pain Inventory Activity Interference Scale [26]); (3) PHQ-4 measure of anxiety and depressive symptoms [27]; (4) European Quality of Life 5 Dimension (EQ5D) index and Visual Analog Scale [28]; (5) Number of falls [29]; and (6) Recovery expectation [30, 31]. Besides, we also include the Quan comorbidity score [32], baseline diagnosis and total relative value units (RVU) at one year before index visit from EMR as covariates. In addition to covariates, 4 spine-related total relative value units (RVUs) calculated in the 365 days after index visit from EMR data were used as outcomes, including: 1) Sum of spine-related RVUS; 2) Sum of spine-related physical therapy RVUS; 3) Sum of spine-related injection RVUs; and 4) Sum of spine-related imaging RVUs. The 4 spine-related RVUs varied in both scale and level of zero-inflation. The BOLD data used in this study were served as both training and validation samples through the V-fold cross-validation. We excluded observations with missing data in terms of outcomes and considered covariates. The final sample contained 4397 observations for training and validation.

***Two-part model***

The two-part model is a flexible statistical model specifically designed to deal with continuous variables with a point mass at a specific value, in this case, zero. The model consists of a binary model for the probability of the outcome being positive and a regression model applied to the positive subsample. The justification for this model is that $E[Y|X]$ can be written as

$$E[Y|X] = Pr(Y > 0|X)E[Y|Y > 0, X] + Pr(Y = 0|X)E[Y|Y = 0, X]$$
$$= Pr(Y > 0|X)E[Y|Y > 0, X]$$

The two pieces of this equation can be estimated separately. The first part $Pr(Y > 0 \mid X)$ is commonly modeled using logistic regression indexed by a finite-dimensional regression parameter $\theta_1$, e.g.,

$$logit\{Pr(y > 0; \theta_1|x)\} = \theta_1' x.$$

The second component $E[Y|Y > 0, X]$ is commonly modeled using OLS based on log-transformed outcome with smearing estimator or Gamma GLMs with log link functions to appropriately account for the high skewness in the data. Other approaches such as accelerated failure time regression, hazard-based regressions, and quantile-based regressions may also be used. In principle any binary regression technique could be used for the first component and any regression technique for the second. This includes the usage of modern machine learning techniques. Given the dizzying array of possible ways of building a predictor of expenditures, it could be quite useful for practitioners to have a formal framework for selecting amongst the many choices.

***Super Learning***

Super learning provides one such framework. Super learning is a general ensembling framework that provides a formal means of selecting a combination of algorithms that best

fits the true cost regression function. Here, the notion of "best" refers to a cross-validated risk criterion. Suppose we have a dataset of independent observations $O_i = (Y_i, X_i), i = 1, \ldots, n$, where $Y$ is the outcome of interest and $X$ is a p-dimensional set of covariates. Suppose we have access to a candidate prediction function $\hat{Q}$, e.g., $\hat{Q}(x)$ could describe a prediction obtained on a new observation $x$ based on a fitted two-stage model, where a logistic regression is used in the first stage and a Gamma GLM in the second stage. We introduce the notion of the *risk of* $\hat{Q}$, which provides a global summary of how well $\hat{Q}$ predicts outcomes $Y$based on covariates $X$. Often (Benkeser and others, 2019, [33]) we rely on risk measures that can be expressed as the *average discrepancy* between $Y$and the prediction made by $\hat{Q}$. In other words, we can define a *loss function* $L(\hat{Q})$ , that takes as input a particular data point $(x, y)$and returns a real number measuring the discrepancy between $\hat{Q}(x)$ and $Y$. A larger value of the loss indicates a further gap between prediction and truth. For example, in the sequel we explicitly consider the squared error loss function $L(\hat{Q})(x, y) = \{y - \hat{Q}(x)\}^2$; and consider mean squared error $R(\hat{Q}) = E\{L(\hat{Q})(x, y)\}$ as our risk criterion. Given a risk criterion, we can define the optimal prediction function, say $Q_0$ as the function that minimizes risk over all possible prediction functions.

We pause to remark that this optimization problem can be equivalently defined as a statistical estimation problem. For example, it is straightforward to show that for squared error loss, $Q_0 = E(Y|X)$. That is, the optimal function for predicting $Y$ from $X$is the conditional mean. Thus, the task of learning the optimal prediction function is equivalent to the task of estimating the conditional mean of cost given covariates. Super learner provides one strategy for accomplishing this task. The equivalence between the pure prediction task and the estimation problem of estimating the conditional mean means that super learning is potentially useful *both* for (i) predicting health care utilization *and* (ii) analyses that examine the impact of interventions on expenditures, where a key step often involves learning the conditional mean outcome in order to de-confound the relationship between intervention and outcome.

In a given problem, there are often many different approaches to developing a prediction function. In super learning, we call these approaches an *algorithm* and refer to a pre-specified collection of algorithms as a *library.* Here, "*algorithm"* is used in a general sense as any means of mapping a given data set into a prediction function and we use *training* to refer to the process of applying an algorithm to data. Examples of algorithms include: (i) fitting ordinary least squared regression and returning the linear predictor; (ii) performing variable screening based on a univariate significance threshold, *then* applying ordinary least squares regression; (iii) training a random forest, where tuning parameters are selected via cross-validation. The super learner library should be, to the greatest extent possible, informed by subject-area expertise, but could also utilize data-driven, machine learning approaches as well [34].

Denote the library $L$ and the cardinality of $L$ as $K$. The implementation of super learner involves using cross-validation to split the data into several distinct training and validation samples. Each algorithm is applied in each training sample, while predictions are obtained in the respective validation samples. The fit of the validation sample predictions is evaluated to determine an ensemble of algorithms that constitute the super learner. The process is described in these several steps:

1. Fit each algorithm in $L$ on the entire dataset $O_i = (Y_i, X_i), i = 1, \ldots, n$ to estimate $\hat{Q}_k(X), k = 1,2, \ldots, K$.
2. Split the data into V mutually exclusive and exhaustive blocks of approximately equal

size. Let the $v^{-th}$ block be the validation sample, and the remaining $V-1$ groups be the training sample, $v = 1,2, \ldots, V$. Define $T(v)$ as the indices of the data in the $v^{-th}$ training set and $V(v)$ as the corresponding validation set.

3. For $v = 1, \ldots, V$, do: train each algorithm using observations $T(v)$ to generate $\hat{Q}_{k,v}$. Obtain a prediction for each observation in $V(v)$, $\hat{Q}_{k,v}(X_i),\ i \in V(v)$.
4. Propose a family of weighted combinations of candidate estimators indexed by $K$-length weight vector $\alpha$. For example, we might consider all $\alpha$that are non-negative and sum to 1:

$$m(x;\ \alpha) = \sum_{k=1}^{K} \alpha_k \hat{Q}_k(x)\,, \alpha_k \geq 0, \sum_{k=1}^{K} \alpha_k = 1$$

5. Using squared error loss, determine the $\alpha$ that minimize the cross-validated risk (CV-MSE) of the ensemble estimator. Let $\hat{m}(X_i;\ \alpha) = \sum_{k=1}^{K} \alpha_k \hat{Q}_{k,v}(X_i)$ and find

$$\hat{\alpha} = argmin_{\alpha} \sum_{i=1}^{n} \{Y_i - \hat{m}(X_i;\ \alpha)\}^2$$

6. The final super learner is

$$\hat{Q}_{SL} = \sum_{k=1}^{K} \hat{\alpha}_k \hat{Q}_k$$

In step 4, many choices of weights could be considered. One particular choice is to select weights from amongst weight vectors that assign a weight of 1 to a particular algorithm and 0 weight to all others. This estimator is known as the cross-validation selector or discrete super learner, since it represents the single algorithm with lowest cross-validated risk. The super learner has strong theoretical guarantees of performance [16] and has shown strong performance in simulation and real data across many different settings [35].

***Two-stage Super Learning***

The super learner framework provides a natural approach to handling two-stage modeling problems. We propose a two-stage Super Learner model, wherein we propose a library for $Pr(Y > 0|X)$ *and for* $E[Y|Y > 0, X]$. The overall super learner library consists of all pairwise combinations of these two models, thereby providing great flexibility in building a robust super learning library to simultaneously handle zero inflation and skewed outcomes. Assuming the stage-1 library $L_1$ includes $K_1$ estimators and the stage-2 library $L_2$ includes $K_2$ estimators, then the two-stage super learner's "whole library" $L_1 \times L_2$ would contain $K_1 \times K_2$ candidate estimators with each one representing a specific combination of algorithms from the first stage and second stage.

The steps for implementing the two-stage super learner is similar to the super learner. To compute prediction function $\hat{Q}$, it involves fitting two models, one for each stage, and taking the product of their prediction. Models at stage 1 should be limited to classification models for binary outcomes and models at stage 2 should be limited to regression models for continuous outcomes. This yields the candidate estimators constituting the two-stage super learner library. The two-stage super learner is then constructed by taking the weighted combination of estimators in the library that minimize the cross-validated risk (CV-MSE). By basic applications of theoretical results from the super learner, the two-stage super learner is believed to perform asymptotically as well as the best possible weighted combination of estimators and outperform any of the single estimators in the two-stage super learner library.

We developed a R package that implements the two-stage super learner and overcame some of the additional challenges associated with super learning in the context of healthcare expenditure data. The package uses a modified assigning scheme for cross-validation that ensures approximately equal split of zeros and outliers over the folds. Secondly, we proposed a scaled quadratic programming to calculate the best convex combination of weights, which avoids the problems of overflow and constraints inconsistency by shrinking the matrix and vectors in quadratic functions. For more details see the supplementary material.

**Simulation studies**

To evaluate the performance of the two-stage super learner, a Monte Carlo simulation was used to show how each estimator behaves under a wide variety of data circumstances that are common in health economics and health service studies.

***Data generating process***

For simulation studies, the participants' covariates were generated under a design that was similar to MEPS data and comprehensive to include as many scenarios as possible. Specifically, the participants' characteristics were simulated as follows:

$$X_1 \sim Bernoulli(0.5); \quad X_6 \sim Bernoulli(0.2);$$
$$X_2 \sim Uniform(-1, 1); \quad X_7 \sim Uniform(0, 1);$$
$$X_3 \sim Normal(0, 1); \quad X_8 \sim Normal(0, 3)$$
$$X_4 \sim Gamma(1, 0.5); \quad X_9 \sim Gamma(0.5, 1)$$
$$X_5 \sim Poisson(1); \quad X_{10} \sim Poisson(2)$$

$X_1 \sim X_5$ were used for generating the simulation data while $X_6 \sim X_{10}$ were served as confounding variables during the model fitting of each estimator. To assess how the sample size, zero percentage, non-zero distribution, and model complexity affects the estimations, we consider four data generating settings for tuning in simulation:

Setting 1: sample size – small (500) vs. large (2000);

Setting 2: zero percentage – low (5%) vs. high (70%);

Setting 3: non-zero distribution – Log-normal, Gamma, Tweedie, Mixture;

Setting 4: model complexity (two-way interactions among covariates) – Yes vs. No

The combination of four settings above result in a total of 32 different data scenarios and covers a broad range of situations that occur in the real world. The healthcare cost was simulated using a two-stage procedure to allow for point mass at zero. Specifically, in the first stage, the sample was first generated from a Bernoulli distribution with the probability of zero determined by a logistic model:

$$logit\{Pr(Y = 0|x)\} = log\left(\frac{Pr(Y = 0|x)}{1 - Pr(Y = 0|x)}\right) = X^T\beta$$

Where $X = (1, X_1, \dots, X_5)$ and $\beta = (\beta_0, \beta_1, \dots, \beta_5)$. The value of estimated coefficients $\beta$ was controlled by data generating settings 2 & 4 listed above. In the second stage, we focus on the subsample with non-zero value in the Bernoulli sample. To determine the effect of skewness level on the estimation, we studied the performance of estimators under four different skewed probability density functions (PDF) that yield positive outcomes skewed to the right. We allowed these skewed distributions to depend on covariates $X$ through a linear combination $X^T\gamma$ with $\gamma$ corresponding to estimated coefficients and remained unchanged across four different distributions.

**Lognormal Distribution:** The true model assumed is as follows:

$$ln\,(y) = X^T\gamma + \varepsilon$$

Where $X = (1, X_1, \dots, X_5), \gamma = (\gamma_0, \gamma_1, \dots, \gamma_5), \varepsilon \sim N(0,\ \sigma^2), E(X^T\varepsilon) = 0$. The conditional expectation of $y$ is given by $E(y|x) = exp\,(X^T\gamma + 0.5\sigma^2)$. For log-normal distribution, the raw-scale mean, variance, skewness and kurtosis are all increasing functions of variance on the log-scale. Specifically, the raw-scale skewness (S) is $S_{raw} = (exp(\sigma^2) + 2)(exp\,(\sigma^2) - 1)^{0.5}$. The logarithm of outcome $ln\,(y)$ was normally distributed with mean determined by $\mu = X^T\gamma$ and variance set to $\sigma^2 = 0.3$.

**Gamma Distribution:** Gamma distribution has a PDF that can be either monotonically declining throughout the support or bell-shaped, but skewed right. The PDF of Gamma distribution is:

$$f(y) = \frac{1}{\Gamma(\alpha)b^a} y^{\alpha-1} e^{-\frac{y}{b}}$$

Where $b$ is the shape parameter and $\alpha$ is the scale parameter. The mean is equal to $\alpha b$ and the skewness (S) is a decreasing function of the shape parameter $S = \frac{2}{\sqrt{\alpha}}$. The shape parameter was determined with $b = exp\,(X^T\gamma)$ and the scale parameter was set to $\alpha = 1.3$ in the simulation study.

**Tweedie Distribution:** Tweedie distributions are defined as a subfamily of exponential dispersion models (ED) with a special mean-variance relationship. A random variable $Y$ is Tweedie distributed $Tw_p(\mu,\ \sigma^2)$, if its mean $\mu = E(Y)$ and $var(Y) = \sigma^2\mu^p$, where $\sigma^2$ is the dispersion parameter and $p \in R$ is the power parameter. The PDF for the Tweedie family is complex and cannot be expressed in closed form, but the Tweedie family includes common distributions like Normal ($p = 0$), Poisson ($p = 1$) and Gamma ($p = 2$). In the simulation, the mean was determined with $\mu = X^T\gamma$, the scale and power were set to $\sigma^2 = 1.8$, and $p = 1.5$, respectively.

**Mixture distribution:** The mixture distribution is a mixture of log-normal and gamma distribution. Specifically, it was generated by first drawing a binary random variable with a pre-specified probability, $G \sim Bernoulli(p)$. We subsequently generated a gamma distribution if $G \leq p$, or a log-normal distribution if $G > p$. In the simulation, the probability was set to $p = 0.5$. We wanted to evaluate the two-stage super learner in situations where a simple parametric model does not capture the true distribution, as we expect to be the case in practice.

***Estimators***

In a two-stage super learner, the probability of the outcome being positive was estimated in stage 1 and the positive level of outcome was estimated in stage 2. Previous study [15] has shown that alternative specifications of the binary choice model (stage-1) yield nearly identical results but the choice of model for the distribution of the outcome conditional on it being positive (stage-2) can yield quite different results with different models. As a result, in this study, we mainly focus on the specification of estimators in stage 2. We used a diverse library for the two-stage super learner, with the library in stage 1 including 3 estimators and the library in stage 2 including 10 estimators. Additionally, we are interested in how the performance of a two-stage super learner compared to that of a super learner, thus we also fit a super learner with a library of 8 estimators, resulting in a total of 38

estimators. The details are shown in table 1 below.

Table 1. Candidate estimators for the simulation study

| Super Learner | Stage | Method |
|---|---|---|
| Two-stage | 1 | GLM (logistic regression) |
| | 1 | Lasso (logistic regression) |
| | 1 | Random Forest |
| | 2 | GLM (log link, Gamma family) |
| | 2 | GLM (identity link, Gamma family) |
| | 2 | Log OLS + smearing |
| | 2 | Lasso (OLS) |
| | 2 | Adaptive GLM |
| | 2 | AFT (generalized Gamma) |
| | 2 | Cox hazard |
| | 2 | Adaptive hazard |
| | 2 | Quantile regression |
| | 2 | Random forest |
| Standard (one-stage) | | GLM intercept only (mean) |
| | | OLS |
| | | Lasso (OLS) |
| | | Zero-inflated Poisson |
| | | Zero-inflated Negative Binomial |
| | | Tobit |
| | | Tweedie |
| | | Random Forest |

***Evaluation metrics***

Each of the estimators is evaluated on 1000 simulated replicate samples from each of the data generating processes. This allows us to reduce the Monte Carlo simulation variance by holding the specific draws of the underlying random numbers constant when comparing alternative estimators. Each estimator was evaluated based on two evaluation metrics:

(1) The mean squared error (MSE). The MSE uses a squared difference for error calculation and indicates how well the estimator minimized the residual error on the raw-scale of the replicate sample. For each replicate r with sample size n,

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_{ri} - \hat{y}_{ri})^2$$

Different data settings have a different scale for the outcome. To compare the performance of the prediction algorithms across diverse data settings we used the relative mean squared error where the denominator is the mean squared error of a pre-specified baseline model:

$$relMSE(k) = \frac{MSE(k)}{MSE(baseline)}, \quad k = 1,2,\dots,K$$

(2) The coefficient of determination, or $R^2$. $R^2$ is another metric to evaluate the model and it is closely related to MSE but has the advantage of being scale-free. $R^2$ is always between $[-\infty, 1]$ no matter how large or small the MSE is. For each replicate r with sample size n,

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_{ri} - \hat{y}_{ri})^2}{\sum_{i=1}^{n}(y_{ri} - \bar{y}_{ri})^2}$$

When $R^2$ is negative it means the model is worse than predicting the mean. Besides $R^2$ itself, we also calculated the relative efficiency (RE) for each algorithm, which we defined as the ratio of $R^2$ for an algorithm to the $R^2$ for the two-stage Super Learner. The larger the RE, the better the performance of an algorithm compared to two-stage Super Learner.

$$RE = \frac{R^2(k)}{R^2(Two - stage\ SL)},\ \ k = 1,2,\dots,K$$

All the evaluation metrics listed above are evaluated via cross-validation to more accurately assess the out-of-sample performance.

**Results**

***Simulation results***

The numerical results of top 10 estimators, summarized in Table 2, were based on 1000 times replication for each of the 32 data generation processes. Within each data generating mechanism, the outcomes were skewed to the right and heavy-tailed with a point mass at zero. Each estimator was evaluated via average MSE and $R^2$, along with their corresponding standard error across 32 different data settings. The relative MSE, calculated using the MSE of S1: GLM + S2: GLM-Gamma-Log as a reference, was provided to better visualize the performance of various estimators relative to a assumed default model for analyzing healthcare expenditure. The boxplots of MSE for each estimator across 32 data settings were shown in figure 1, with the diamond inside the box indicating the mean MSE over 1000 simulations. For all the boxplots following, the estimators were ordered descendingly according to their mean value. In general, with the given library, the two-stage super learner was able to adapt to the underlying structure of different data generating functions with the smallest average MSE and largest average $R^2$. As expected, the super learner behaved worse than the two-stage super learner. The two-stage super learner also appeared to improve on the discrete super learner with better prediction. The selection of a single algorithm based on cross-validated risk minimization was unstable and changed every time under different data settings, while the two-stage super learner can average a few of the best algorithms in the library to give a more stable estimator to model misspecification. Other estimators with favorable performances included combinations of logit model and Lasso at stage 1 with GLM (log-link & gamma distribution), quantile regression, and log-transformed OLS at stage 2, together with zero-inflated Negative Binomial model. By contrast, estimators based on cox hazard, GLM (identity link & gamma distribution) at stage 2 as well as Tweedie and Tobit model were less efficient with larger MSE, indicating poor predictions.

Table 2. The MSE, relative MSE, and $R^2$, averaged over 1000 repetitions of top 10 estimators across 32 data generating processes

| Algorithm | MSE ($10^8$) | Relative MSE | $R^2$ |
|---|---|---|---|
| Two-stage Super Learner | 3.251 | 0.907 | 0.622 |
| Discrete Super Learner | 3.307 | 0.923 | 0.616 |
| Super Learner | 3.382 | 0.944 | 0.611 |
| S1: Lasso + S2: GLM-Gamma-Log | 3.563 | 0.994 | 0.609 |
| S1: Lasso + S2: Quantile regression | 3.567 | 0.996 | 0.608 |
| S1: Lasso + S2: Log OLS-smearing | 3.570 | 0.996 | 0.608 |
| Zero-inflated Negative Binomial (ZINB) | 3.582 | 1.000 | 0.607 |

| | | | |
|---|---|---|---|
| S1: GLM + S2: GLM-Gamma-Log | 3.583 | 1.000 | 0.607 |
| S1: GLM + S2: Quantile regression | 3.589 | 1.002 | 0.606 |
| S1: GLM + S2: Log OLS-smearing | 3.590 | 1.002 | 0.606 |

Note: Algorithms are presented in ascending order according to average MSE. S1 refers to stage-1 and S2 refers to stage-2. GLM in S1 refers to logistic regression and Lasso in S1 refers to logistic Lasso regression. GLM-Gamma-Log refers to GLM with Gamma family and Log link function. The relative MSE is calculated using the MSE of S1: GLM + S2: GLM-Gamma-Log as a reference.

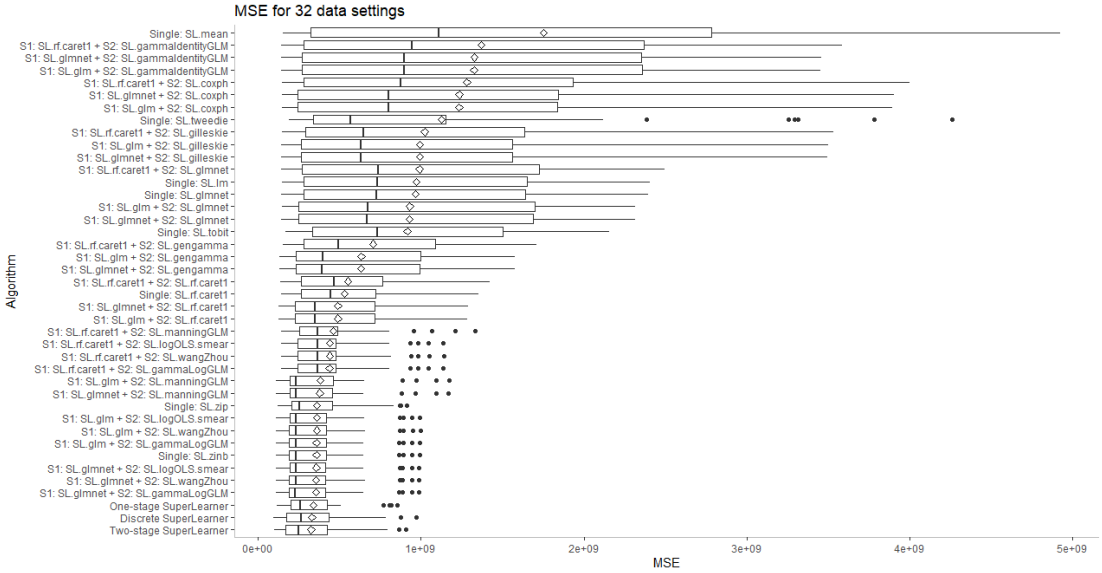


Figure 1. Simulation – Boxplot of MSE across 32 data settings

Figure 2 contains the MSE from all estimators on small vs. large datasets as well as low vs. high zero percentage. Similar results were seen for small (figure 2a) and large (figure 2b) sample sizes, where the super learner-based methods all out-performed the parametric/semiparametric methods due to the latter being probably mis-specified. For a small sample size, we found the standard super learner yielded the minimal average MSE, followed by the two-stage super learner, though their observed differences were trivial within the bounds of Monte Carlo error. For a large sample size, the two-stage super learner surpassed the standard super learner and their difference in MSE is more evident. The MSE of the two-stage super learner in small sample size was smaller than that of the discrete super learner compared to a large sample size, justified the improved performance of super learner under the finite-sample setting. Investigation of the estimator ranking showed that the performances of each estimator remained fairly consistent regardless of the sample size. Nonetheless, the results from low zero percentage (figure 2c) and high zero percentage were quite different. Generally, the MSE for data with high zero percentages was smaller than that for data with low zero percentages. When the number of zeros was low (5%), the standard super learner did reasonably well and outperformed the two-stage super learner. This was owing to the limited number of zeros that obviated the need to specify an additional ensemble for zero-point mass issues, as can be seen in figure 2c where the Tweedie and zero-inflated model beat all the two-part models in the library. When the zero percentage was high (70%), the two-stage super learner started to exceed the standard super learner. This was in line with expectations as the presence of excess zeros was serious and an extra layer was in need to deal with zero-point mass issues, as can be seen in figure 2d where the performances of the one-part models in high zero-percentage samples were not quite as good as that in low-zero-percentage samples.

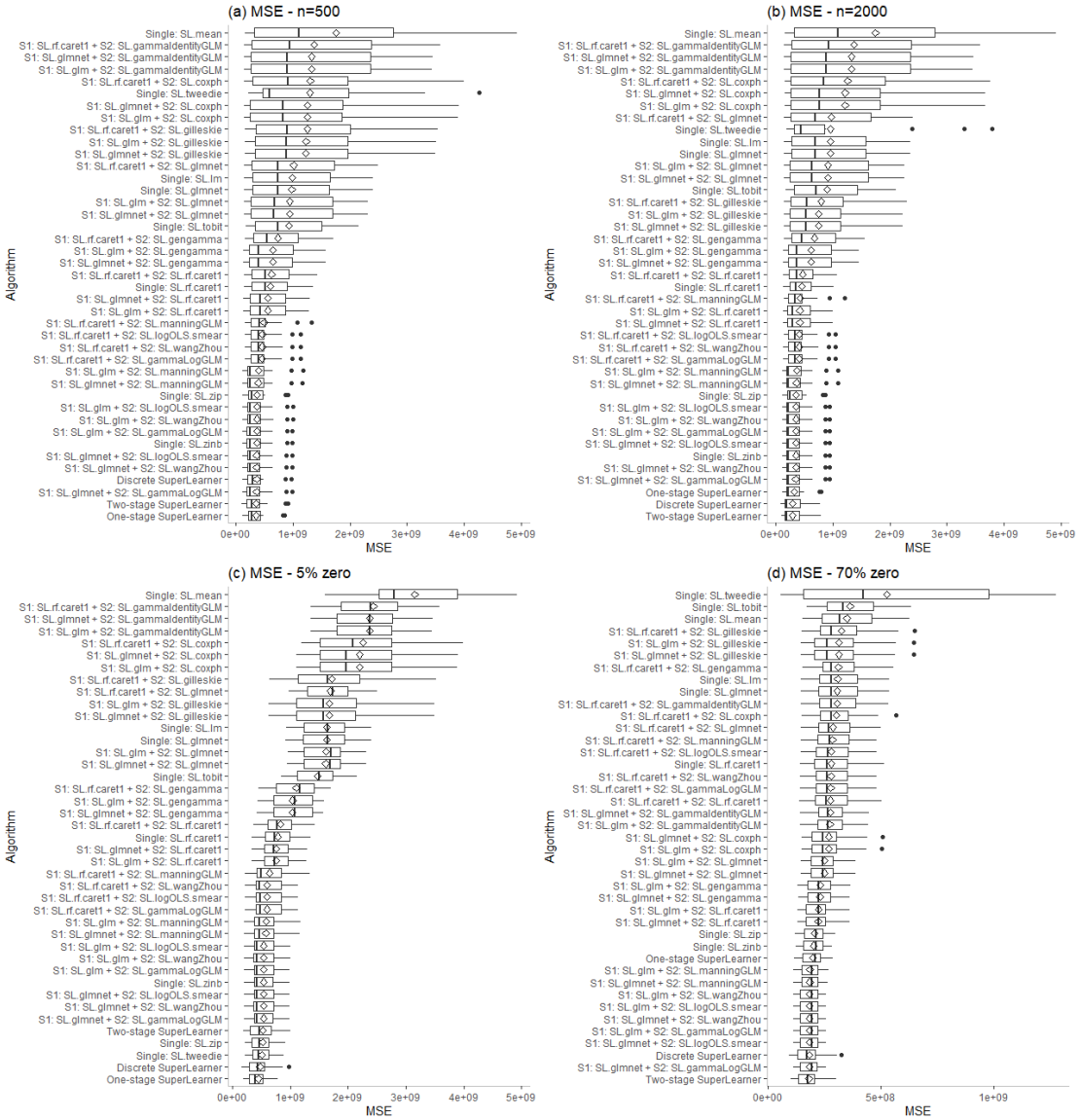


Figure 2. Simulation – Boxplot of MSE for small vs. large sample size and low vs. high zero percentage

Figure 3 contains the MSE for the top 10 estimators under four different distributions. We additionally added the MSE of the Tweedie model in Figure 3c for illustration. It's shown that the estimators behaved pretty differently under different data generating distributions, leading to slightly disparate performance of the super learner. For log-normal and gamma generated data, the log-OLS with smearing retransformation and GLM/adaptive GLM were able to achieve the minimum MSE among all single algorithms respectively since these algorithms were able to well approximate the true underlying data structure. However, under the mixture distribution, the quantile regression by Wang & Zhou had a better model fit than other single algorithms. This is probably because this algorithm was distribution-free and was able to flexibly estimate the outcome through regressing on the quantiles of the outcome, while other algorithms were not enough to approach the underlying distribution. Surprisingly, the Tweedie model exhibited relatively bad performances even under the Tweedie data-generating scenario. This may be due to the instability of the

Tweedie model, as can be seen by the large interquartile and extreme outlier in figure 3c. A comparison of different distributions revealed that the two-stage super learner was always best behaved and had the minimum MSE across almost all data generating distributions. The only exception occurred at the gamma distribution, where the best results were obtained on the discrete super learner, perhaps indicating a deleterious effect in finite samples of including mis-specified methods in the library of a two-stage super learner. The Cox hazard model was generally very bad, possibly because the proportional hazards assumption was violated in all four data generating distributions.

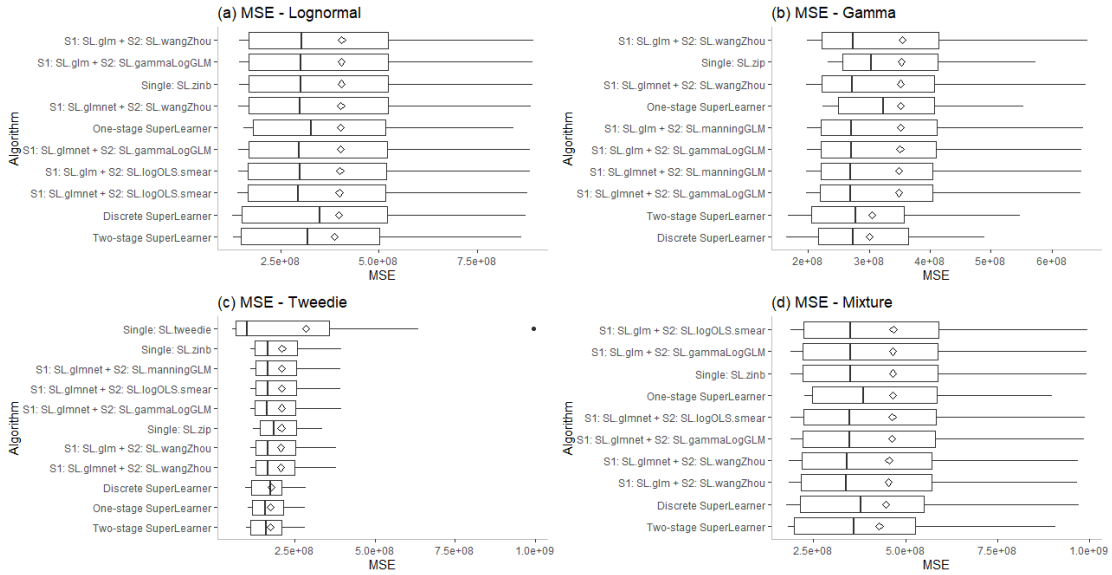


Figure 3. Simulation – Boxplot of MSE for top 10 estimators under four distributions

***Empirical data analysis***

***MEPS Data***

We used data from the 2016-2017 MEPS to evaluate the two-stage super learner in real world situations. We used an identical set of algorithms as in the simulation study, with all candidate algorithms in the two-stage super learner library modeled under a 10-fold cross-validation. The final sample had 10925 observations for training and 10815 observations for testing. Expenditures are measured in nominal US dollars. The distribution of total expenditures is highly skewed with a large mass at zero and heavy upper tails (Figure 4). The skewness is 7.2 and 6.7 for 2016 & 2017 MEPS data, respectively (compare to 0 for symmetric data). For MEPS data, almost 20% of observations have zero expenditures and in a very small fraction of observations, 2% to be precise, had expenditures over $50,000. Although it is tempting to drop extreme observations, we are reluctant to do so because we cannot be sure that they are outliers in any real sense. Summary statistics of total health expenditures are reported in Table 3.

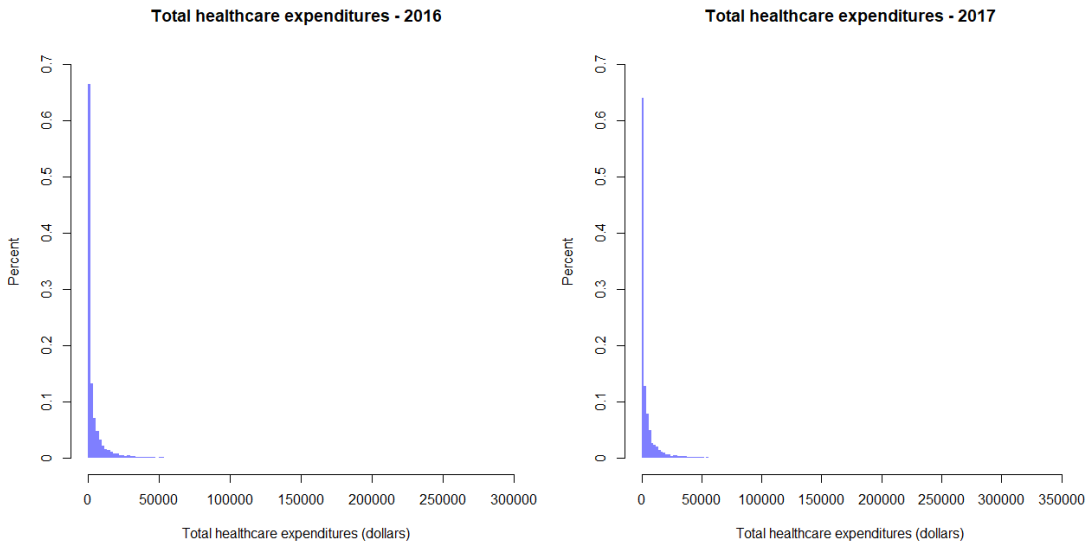


Figure 4. Distributions of total healthcare expenditures

Table 3. Summary statistics of total health expenditures for MEPS data

| Measures | 2016 MEPS N=10925 | 2017 MEPS N=10815 |
|---|---|---|
| 0.05 Percentile | 0.0 | 0.0 |
| 0.25 Percentile | 157.0 | 130.0 |
| 0.50 Percentile | 1076.0 | 1156.0 |
| 0.75 Percentile | 4521.0 | 4902.0 |
| 0.95 Percentile | 24277.8 | 28189.3 |
| Mean | 5368.4 | 6099.7 |
| SD | 13815.1 | 15992.3 |
| Skewness | 7.2 | 6.7 |
| Zero % | 17.8 | 19.6 |

Covariates include demographics, education, poverty, disease conditions and insurance coverage. Descriptive statistics of covariates are shown in Table 4. A smaller proportion of females (45.5% and 45.6%) appeared in the MEPS data and the mean age was approximately 45.5 years. The 2016 MEPS data had similar education levels and distributions of race and region as the 2017 MEPS data. People in 2017 MEPS had a better insurance and medical care coverage than people in 2016 MEPS — likely because they are richer with a higher average family income as % of the poverty line (369.5% vs. 361.1%). Participants of 2017 MEPS are also healthier compare to participants of 2016 MEPS with a slightly fewer presence of diabetes (11.9% vs. 12.0%), hypertension (33.9% vs. 34.1%), cancer (8.7% vs. 9.0%) and heart disease (12.8% vs. 13.1%).

Table 4. Descriptive statistics of covariates

| Covariate | | 2016 MEPS N=10925 | 2017 MEPS N=10815 |
|---|---|---|---|
| Age | | 46.6 | 46.4 |
| Sex | Female (%) | 45.6 | 45.5 |
| | Male (%) | 54.4 | 54.5 |
| Race | Hispanic (%) | 28.8 | 28.9 |

| | | | |
|---|---|---|---|
| | White (%) | 42.5 | 42.4 |
| | Black (%) | 18.4 | 18.3 |
| | Asian (%) | 7.5 | 7.5 |
| | Other (%) | 2.9 | 2.9 |
| Region | Northeast (%) | 16.3 | 16.2 |
| | Midwest (%) | 19.4 | 19.3 |
| | South (%) | 37.9 | 38.0 |
| | West (%) | 26.4 | 26.5 |
| Education (yrs) | | 12.9 | 12.9 |
| Income as percent poverty (%) | | 361.1 | 369.5 |
| Private insurance (%) | | 59.5 | 61.0 |
| Medicare (%) | | 22.6 | 23.5 |
| Public insurance (%) | | 22.9 | 23.6 |
| Uninsured (%) | | 12.3 | 11.0 |
| Diabetes (%) | | 12.0 | 11.9 |
| Hypertension (%) | | 34.1 | 33.9 |
| Cancer (%) | | 9.0 | 8.7 |
| Heart disease (%) | | 13.1 | 12.8 |

For the prediction of annual healthcare expenditure of MEPS 2017, the two-stage super learner performed better than the super learner and all the single algorithms considered with the lowest cv-MSE (Figure 5a) and the highest cv-$R^2$ (Figure 5b). See table 6 for details of MSE, $R^2$, and relative efficiency (RE) for top 15 estimators. As expected, for estimators that were used both alone and at stage-2 (Random Forest, Lasso), the two-stage models are always better than the one-stage model. Additionally, the choices of models at stage-2 matters more compared to the choices of models at stage-1 as estimators with the same stage-2 model but different stage-1 models shared analogous behavior. Efficiency losses for the single algorithms compared to the two-stage super learner, with respect to cross-validated $R^2$, ranged from 0 to 85%. Not surprisingly, predicting with just the mean was the worst-performing algorithm with $R^2$ and RE being approximately 0. Log-transformed OLS with smearing retransformation and Adaptive GLM used at stage 2 also performed poorly with less than 20% RE compared to two-stage Super Learner, no matter which algorithms were used at stage 1. Random forests used at stage-2 along with any candidate algorithms at stage 1 performed nearly as well as the two-stage super learner, capturing over 95% of the efficiency of two-stage super learner. Any of these three algorithms could be chosen as the discrete super learner in practice given the minor absolute differences in performance, although the random forest used at both stages had the highest $R^2$. The super learner was among the best performed estimators with $R^2$ and RE higher than any single algorithm as anticipated. For stage-2 algorithms, Lasso performed better than the GLM and quantile regression, however, its improvement over these algorithms was trivial. The hazard-based models (adaptive hazard, cox hazard) had better performances compared to the accelerated failure time (AFT) model. What remains consistent is that, given each algorithm used at stage 2, the performance is always better when the random forest is used at stage 1 compared to the logit model and Lasso.

Table 6. Results of MSE, $R^2$ and Relative Efficiency (RE) for top 15 estimators

| Algorithm | MSE ($10^9$) | $R^2$ | RE |
|---|---|---|---|
| Two-stage Super Learner | 2.180 | 0.147 | 1.000 |
| Discrete Super Learner | 2.192 | 0.143 | 0.969 |

| | | | |
|---|---|---|---|
| S1: RF + S2: RF | 2.192 | 0.143 | 0.969 |
| S1: GLM + S2: RF | 2.193 | 0.142 | 0.965 |
| S1: Lasso + S2: RF | 2.194 | 0.141 | 0.961 |
| Super Learner | 2.221 | 0.132 | 0.893 |
| RF | 2.236 | 0.126 | 0.852 |
| Zero-inflated Poisson (ZIP) | 2.257 | 0.119 | 0.810 |
| S1: GLM + S2: Lasso (OLS) | 2.260 | 0.118 | 0.803 |
| S1: Lasso + S2: Lasso (OLS) | 2.260 | 0.117 | 0.798 |
| S1: RF + S2: Lasso (OLS) | 2.261 | 0.117 | 0.796 |
| OLS | 2.264 | 0.116 | 0.789 |
| Lasso (OLS) | 2.265 | 0.115 | 0.782 |
| S1: RF + S2: GLM-Gamma-Log | 2.267 | 0.114 | 0.776 |
| S1: GLM + S2: GLM-Gamma-Log | 2.268 | 0.114 | 0.770 |

Note: Estimators are presented in ascending order based on MSE. S1 refers to stage-1 and S2 refers to stage-2. RF refers to Random Forest. GLM in S1 refers to logistic regression and Lasso in S1 refers to logistic Lasso regression. GLM-Gamma-Log refers to GLM with Gamma family and Log link function.

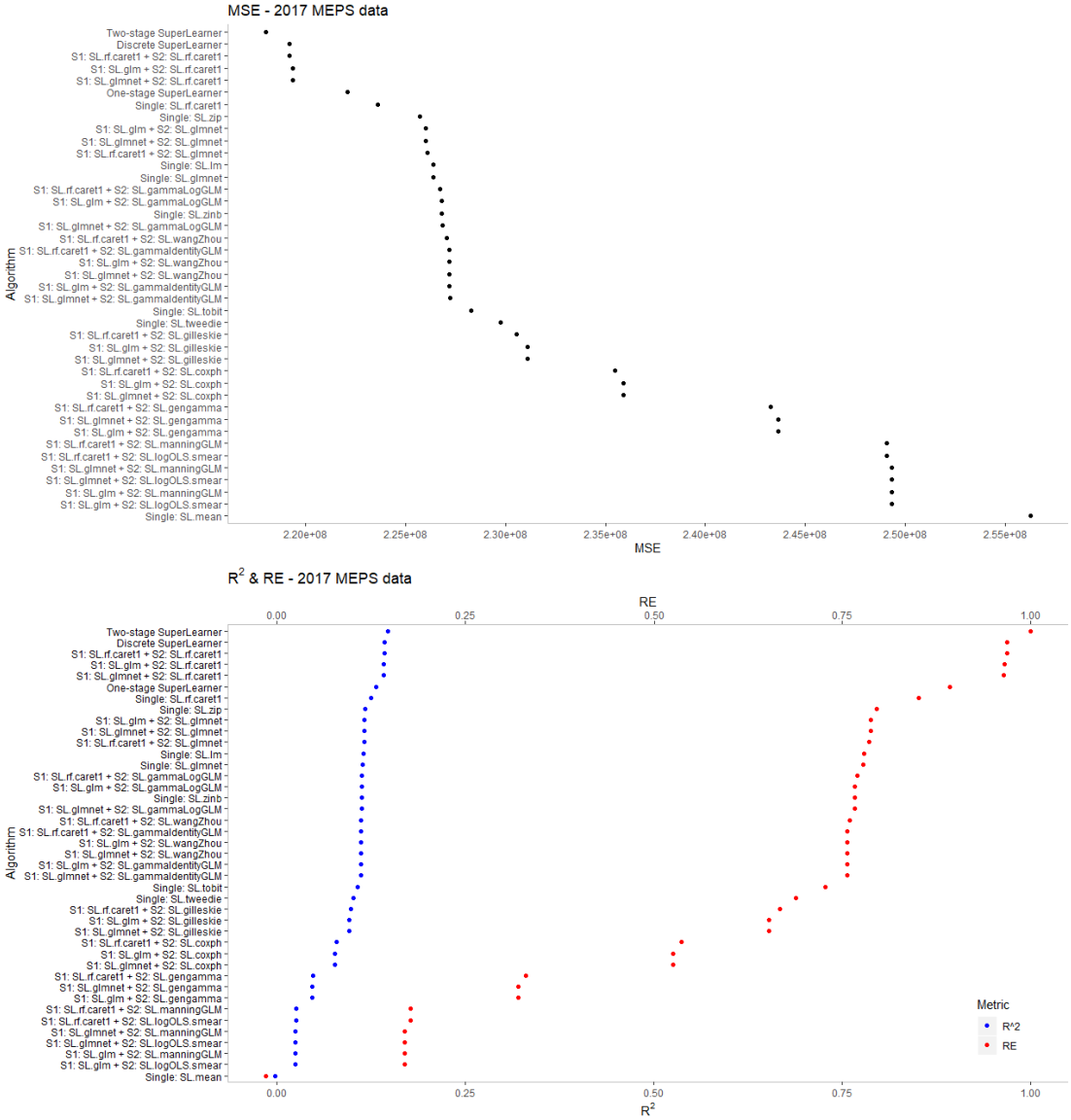


Figure 5. MEPS results – MSE, $R^2$ and relative efficiency (RE)

***BOLD Data***

We further applied the two-stage super learner to analyze the BOLD data. The final dataset contained 4397 observations including 4 spine-related total RVUs as outcomes and 24 variables measured at baseline as covariates. A summary of 4 outcomes is presented in Table 7. Despite the difference in scale, the distributions of 4 spine-related RVUs are all highly skewed with heavy upper tails (Figure 4). Their skewness are all above 6, compared to 0 of absolute symmetry. Conversely, different spine-related RVUs have different zero-inflation, with spine-related RVU having the lowest zero-mass (5%), then comes the spine-related imaging RVU (55%), spine-related physical therapy RVU (85%) and spine-related injection RVU (91%).

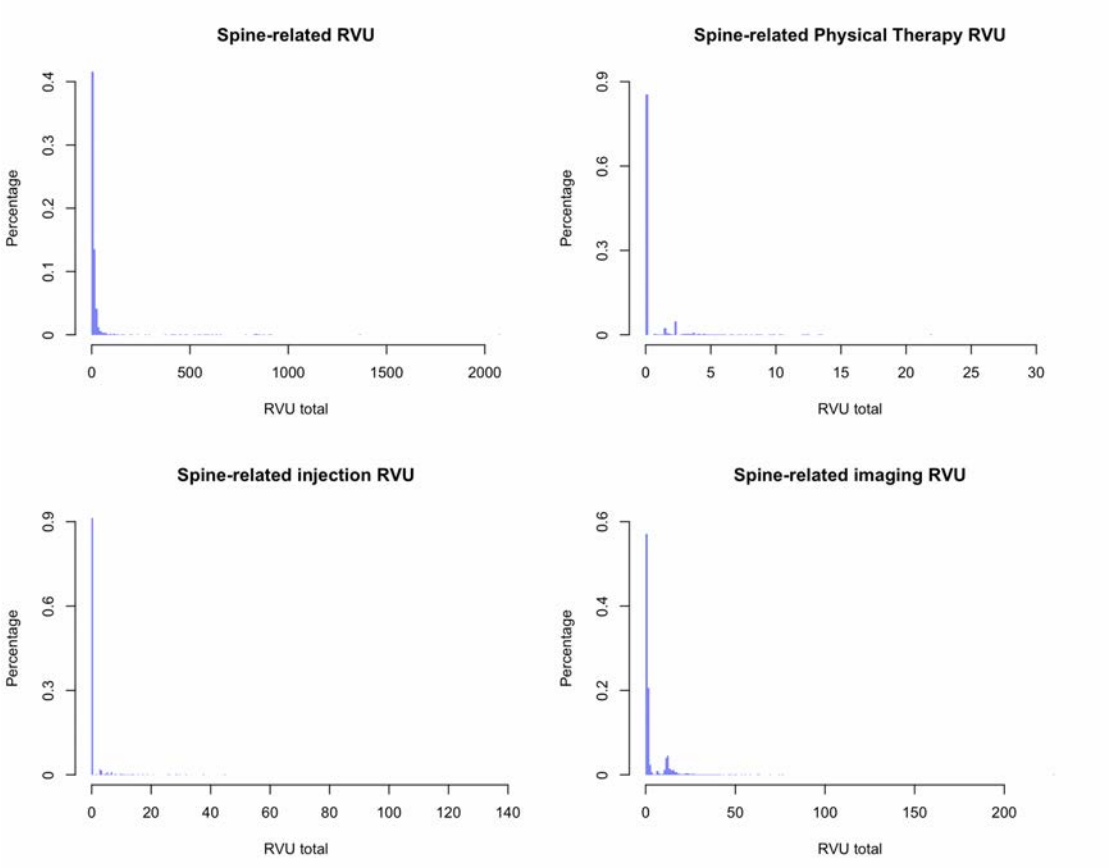


Figure 6. Distributions of 4 total spine-related RVUs

Table 7. Summary statistics of total spine-related RVUs for BOLD data

| Measures | Spine-related RVUS | Spine-related physical therapy RVU | Spine-related injection RVU | Spine-related imaging RVU |
|---|---|---|---|---|
| 0.05 Percentile | 0.10 | 0.00 | 0.00 | 0.00 |
| 0.25 Percentile | 1.48 | 0.00 | 0.00 | 0.00 |
| 0.50 Percentile | 2.45 | 0.00 | 0.00 | 0.00 |
| 0.75 Percentile | 7.38 | 0.00 | 0.00 | 0.00 |
| 0.95 Percentile | 28.32 | 3.39 | 4.28 | 15.40 |
| Mean | 14.83 | 0.54 | 0.75 | 3.30 |
| SD | 80.60 | 1.85 | 4.20 | 7.69 |

| | | | | |
|---|---|---|---|---|
| Skewness | 12.01 | 6.20 | 13.96 | 8.20 |
| Zero % | 5.00 | 85.35 | 91.31 | 55.22 |

Descriptive statistics of BOLD baseline covariates are shown in Table 8. To reduce the sparsity in data, excess levels in certain baseline categorical covariates were grouped together to facilitate the modeling process. This grouping procedure was consistent with previous BOLD studies [36, 37, 38]. Regarding sociodemographics, most patients in BOLD study were females (64.9%), Caucasians (73.4%), recruited from Kaiser site (66.6%), high school graduate (54.6%), non-smokers (55.1%), retired (81.8%) and lived with spouse or partner (59.5%) with a mean age of 74 years old. With respect to back pain related measures, patients in BOLD study were more likely to be diagnosed as back pain only (67.6%), have no falls and injuries in last 3 weeks (92.5% & 96.6%) and have had back pain for less than 3 months (53.8%). They rated their back and leg pain as moderate intense (5.0 & 3.5), they reported minor pain interference with activities (BPI: 3.4), slight psychological distress (PHQ-4: 1.6), modest level of disability (RMDQ: 9.7), medium back pain recovery expectation (5.5) and good quality of life, as measured by the EQ-5D index (0.76) and EQ-5D VAS (74.4). The mean total RVUs in a year before index visit is 39.1.

Table 8. Descriptive statistics of BOLD baseline covariates

| Covariate | | No. (%) of Patients N=4397 |
|---|---|---|
| Study Site | Harvard Vanguard | 682 (15.5%) |
| | Henry Ford | 787 (17.9%) |
| | Kaiser | 2928 (66.6%) |
| Age, mean (SD) | | 73.7 (6.8) |
| Gender - Female | | 2852 (64.9%) |
| Hispanic - Yes | | 259 (5.9%) |
| Race | Black | 671 (15.3%) |
| | Asian | 185 (4.2%) |
| | White | 3229 (73.4%) |
| | Mixed race | 312 (7.1%) |
| Education | < High school | 252 (5.7%) |
| | >= High school | 2399 (54.6%) |
| | College graduate | 972 (22.1%) |
| | Graduate degree | 774 (17.6%) |
| Living with Spouse or partner | | 2616 (59.5%) |
| Smoking status | Never Smoked | 2424 (55.1%) |
| | Quit > 1 year ago | 1712 (38.9%) |
| | Current smoker/quit < 1 year ago | 261 (5.9%) |
| Employment | Working Full-time/Part-time | 483 (11.0%) |
| | Retired (not due to ill health) | 3598 (81.8%) |
| | Retired/disabled because of ill health | 125 (2.8%) |
| | Other | 191 (4.3%) |
| Lawyer - Yes | | 26 (0.6%) |
| Back Pain duration | < 1 month | 1488 (33.8%) |
| | 1 - 3 months | 879 (20.0%) |
| | 3 - 6 months | 296 (6.7%) |
| | 6 - 12 months | 257 (5.8%) |
| | 1 - 5 years | 645 (14.7%) |
| | > 5 years | 832 (18.9%) |

| | | |
|---|---|---|
| Back Pain intensity (0-10), mean (SD) | | 5.0 (2.8) |
| Leg Pain intensity (0-10), mean (SD) | | 3.5 (3.3) |
| Back Pain Recovery Expectations in 3 months (0–10), mean (SD) | | 5.5 (3.7) |
| Patients with one or more fall in last 3 weeks | | 328 (7.5%) |
| Patients with an injury* caused by falls | | 150 (3.4%) |
| RMDQ score (0-24), mean (SD) | | 9.7 (6.3) |
| BPI interference (0-10), mean (SD) | | 3.4 (2.5) |
| EQ-5D index (0-1), mean (SD) | | 0.76 (0.17) |
| EQ-5D VAS (0-100), mean (SD) | | 74.4 (18.3) |
| PHQ-4 score (0-12), mean (SD) | | 1.6 (2.5) |
| Baseline diagnosis | Back pain only | 2972 (67.6%) |
| | Back and leg pain | 954 (21.7%) |
| | Spinal Stenosis | 217 (4.9%) |
| | Other | 254 (5.8%) |
| Quan comorbidity score | 0 | 527 (12.0%) |
| | 1 | 1726 (39.3%) |
| | 2 and more | 2144 (48.8%) |
| RVUs in a year before index, mean (SD) | | 39.1 (89.1) |

*The injury was defined as limiting regular activities for at least a day or requiring a visit to a doctor.

Considering the unknown true underlying distribution of RVUs in reality and potential bias caused by model misspecification, in BOLD analysis we replaced certain algorithms used previously in super learner and stage-2 of two-stage super learner with machine learning algorithms including regression splines**,** regression tree, bootstrap aggregating, Gradient Boosting and Neural Network, which are more flexible and require fewer assumptions. These algorithms have been used for cost estimation in previous researches of healthcare expenditures [39, 40]. The details are shown in table 9. Performances of all estimators were evaluated based on 10-fold cross-validation. Within each training fold, the candidate algorithms in the two-stage super learner library were also modeled under a 10-fold cross-validation.

Table 9. Candidate estimators for the BOLD data

| Super Learner | Stage | Method |
|---|---|---|
| Two-stage | 1 | GLM (logistic regression) |
| | 1 | Lasso (logistic regression) |
| | 1 | CV-Random Forest |
| | 2 | GLM (log link, Gamma family) |
| | 2 | GLM (identity link, Gamma family) |
| | 2 | Log OLS + smearing |
| | 2 | Lasso (OLS) |
| | 2 | Multivariate Adaptive Regression Spline (MARS) |
| | 2 | Regression Tree |
| | 2 | Bootstrap Aggregating (Bagging) |
| | 2 | CV-Random Forest |
| | 2 | Gradient Boosting Machine |
| | 2 | Neural Network |
| | | OLS |
| | | Lasso (OLS) |
| | | Multivariate Adaptive Regression Spline (MARS) |

| | |
|---|---|
| Standard (one-stage) | Regression Tree |
| | Bootstrap Aggregating (Bagging) |
| | CV-Random Forest |
| | Gradient Boosting Machine |
| | Neural Network |

For prediction of spine-related RVUs, the improvements of two-stage super learner over super learner and best single algorithms are modest but prevalent. See table 10 for cross-validated MSE, $R^2$ and relative efficiency (RE) of top 10 algorithms. Specifically, the two-stage super learner had the best overall performance under all different zero-inflation levels, with the smallest cross-validated MSE and the largest cross-validated $R^2$ among all algorithms considered. However, its improvement over the super learner and best single algorithm was humble, where the super learner had a relative efficiency ranging from 95% to 99% and the best single algorithm had a relative efficiency about 98% in modeling 4 spine-related RVUs. Algorithms that perform well in one setting will not necessarily perform well in other settings, as can be seen in figure 7 where the top 15 algorithms in modeling 4 spine-related RVUs changed dramatically. Super learner worked better in modeling outcomes with low to medium zero-inflation, with performance ranked 2th in modeling spine-related RVUs (5% zero) and 6th in modeling spine-related imaging RVUs (55% zero). One-stage models had good behavior likewise when zero-mass were not serious, especially in low zero-inflation situations where the one-stage Random Forest beat all the two-stage models (figure 7). Overall, the regression tree and regression splines (MARS) had worst performance either used alone or combined with estimators at stage-1 (appendix table). Neural Network and bootstrap aggregating also performed poorly compared to parametric regressions (appendix table).

In terms of probability estimation at stage-1, parametric regressions were recommended under high zero-inflation while machine learning algorithms were recommended under medium zero-inflation, as suggested in figure 7 where Random Forest at stage-1 performed perfectly in modeling spine-related imaging RVUs while GLM and Lasso at stage-1 performed perfectly in modeling spine-related physical therapy RVUs and injection RVUs. When zero-inflation was low, the choice of algorithms at stage-1 was relatively less important compared to that at stage-2. GLM, Lasso and Random Forest at stage-1 performed similarly good in modeling spine-related RVUs when Random Forest or Lasso were used at stage-2 (figure 7). Similar recommendations applied for positive cost estimation at stage 2. In particular, Random Forest is good at modeling spine-related RVUs and spine-related imaging RVUs, Gradient Boosting is good at modeling spine-related imaging RVUs, while log-LOS smearing, GLM with Gamma family and identity or log link are good at modeling spine-related physical therapy RVUs and injection RVUs. We found that baseline socio-demographics and patient-reported outcomes may have signals for predicting various spine-related RVUs in a year after index, with a cross-validated $R^2$ ranging from 6.2% to 26.6% for two-stage super learner. The performance was especially great for modeling spine-related physical therapy RVUs. MSE varied significantly in modeling different spine-related RVUs, partly due to the difference in terms of scale of RVUs. Our results also suggest that there may exist redundancy in the covariates as Lasso is always among the best performed estimators in modeling spine-related RVUs with different zero-inflations. Spine-related RVU is responsible for approximately 20% of the total RVU and around 5% of patients in our sample have no spine-related consumption, so, while a cross-validated $R^2$ of 6.6% may seem low, this is a stronger signal than we should ideally see.

Table 10. Top 10 algorithms + Super Learner for modeling 4 spine-related RVUs

| Algorithm | MSE | $R^2$ | RE |
|---|---|---|---|
| Spine-related RVUs (5% zero) | | | |
| Two-stage Super Learner | 6291.351 | 0.0618 | 1.0000 |
| Super Learner | 6307.790 | 0.0611 | 0.9899 |
| Discrete Super Learner | 6316.867 | 0.0608 | 0.9844 |
| Single: RF | 6320.680 | 0.0607 | 0.9820 |
| S1: RF + S2: RF | 6336.996 | 0.0600 | 0.9720 |
| S1: GLM + S2: RF | 6344.708 | 0.0597 | 0.9673 |
| S1: Lasso + S2: RF | 6344.743 | 0.0597 | 0.9673 |
| S1: RF + S2: Lasso | 6394.274 | 0.0579 | 0.9369 |
| S1: Lasso + S2: Lasso | 6395.829 | 0.0578 | 0.9359 |
| SL: GLM + S2: Lasso | 6396.604 | 0.0578 | 0.9355 |
| Spine-related imaging RVUs (55% zero) | | | |
| Two-stage Super Learner | 55.6323 | 0.0872 | 1.0000 |
| Discrete Super Learner | 55.7390 | 0.0855 | 0.9799 |
| S1: RF + S2: GBM | 55.7779 | 0.0849 | 0.9726 |
| S1: RF + S2: Lasso | 55.7976 | 0.0845 | 0.9689 |
| S1: RF + S2: SL.RF | 55.8768 | 0.0832 | 0.9540 |
| Super Learner | 55.8939 | 0.0830 | 0.9508 |
| S1: RF + S2: GLM-Gamma-Identity | 55.9810 | 0.0815 | 0.9344 |
| S1: RF + S2: GLM-Gamma-Log | 55.9972 | 0.0813 | 0.9314 |
| S1: RF + S2: Log-OLS smearing | 56.0175 | 0.0809 | 0.9276 |
| S1: GLM + S2: Lasso | 56.0735 | 0.0800 | 0.9170 |
| Spine-related physical therapy RVUs (85% zero) | | | |
| Two-stage Super Learner | 2.5094 | 0.2659 | 1.0000 |
| Discrete Super Learner | 2.5191 | 0.2630 | 0.9893 |
| S1: Lasso + S2: Log-OLS smearing | 2.5213 | 0.2624 | 0.9869 |
| S1: GLM + S2: Log-OLS smearing | 2.5233 | 0.2618 | 0.9847 |
| S1: Lasso + S2: GLM-Gamma-Identity | 2.5260 | 0.2610 | 0.9817 |
| S1: GLM + S2: GLM-Gamma-Identity | 2.5265 | 0.2609 | 0.9812 |
| S1: GLM + S2: Lasso | 2.5270 | 0.2607 | 0.9806 |
| S1: Lasso + S2: Lasso | 2.5294 | 0.2600 | 0.9779 |
| S1: Lasso + S2: GBM | 2.5298 | 0.2599 | 0.9775 |
| S1: GLM + S2: GBM | 2.5314 | 0.2594 | 0.9758 |
| Super Learner | 2.5331 | 0.2589 | 0.9739 |
| Spine-related injection RVUs (91% zero) | | | |
| Two-stage Super Learner | 17.2069 | 0.1191 | 1.0000 |
| Discrete Super Learner | 17.2446 | 0.1175 | 0.9866 |
| S1: Lasso + S2: Lasso | 17.2480 | 0.1173 | 0.9853 |
| S1: Lasso + S2: GLM-Gamma-Identity | 17.2543 | 0.1171 | 0.9831 |
| S1: GLM + S2: SL.Lasso | 17.2564 | 0.1170 | 0.9823 |
| S1: GLM + S2: Log-OLS smearing | 17.2571 | 0.1169 | 0.9821 |

| | | | |
|---|---|---|---|
| S1: GLM + S2: GLM-Gamma-Identity | 17.2575 | 0.1169 | 0.9820 |
| S1: Lasso + S2: GLM-Gamma-Log | 17.2596 | 0.1168 | 0.9812 |
| S1: Lasso + S2: Log-OLS smearing | 17.2605 | 0.1168 | 0.9809 |
| S1: GLM + S2: GLM-Gamma-Log | 17.2637 | 0.1167 | 0.9797 |
| Super Learner | 17.2763 | 0.1161 | 0.9752 |

Note: S1 refers to stage-1 and S2 refers to stage-2. RF refers to Random forest. GLM in S1 refers to logistic regression and Lasso in S1 refers to logistic Lasso regression. GBM refers to gradient boosting machine. GLM-Gamma-Identity refers to GLM with Gamma family and Identity link function.

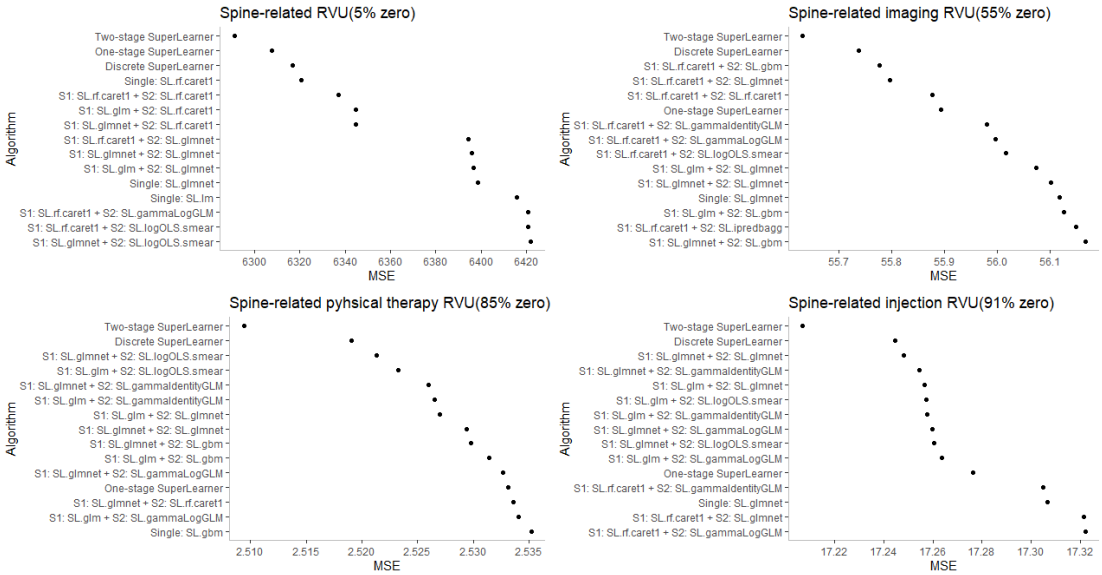


Figure 7. BOLD results – MSE of top 15 algorithms for 4 spine-related RVUs

**Discussion**

Modeling healthcare expenditures frequently have challenges related to the distribution of outcome. Healthcare expenditure data, for those with any healthcare use, are generally right skewed. In the United States, the majority have zero healthcare expenditure while a small fraction of the population accounts for a substantial fraction of total expenditures. Numerous methods had been proposed to address the issues in healthcare cost analysis and super learner was proved to be an appealing paradigm that leverages various tools developed in the health economics literature for modeling health costs. However, those researches have mainly focused on the continuous part of the cost. In this study, I demonstrated that the Binomial part deserves more attention, as it is obvious that a one-stage model cannot adequately distinguish the non-users and users if they share similar characteristics (i.e. they are highly correlated) and the classes are very unbalanced (i.e. the number of non-users is low). Specifically, a two-stage version of the super learner (two-stage super learner) was designed where the first stage predicts whether an individual is likely to have healthcare expenditure and the second stage calculates the expected spending given the individual has healthcare services. The two-stage super learner was evaluated under a Monte Carlo simulation, with hypothetical cost data drawn randomly through different data generating processes, along with two empirical analyses where the data are from 2016-2017 MEPS and BOLD study. The former allows the performance of each

estimator to be assessed against known parameter values. The latter allows the predictive performance to be assessed when the estimators are confronted with the idiosyncrasies of the distribution of actual cost data, rather than textbook parametric distributions.

Our simulation demonstrated that there is no best model among all data conditions. However, the proposed two-stage super learner performed quite well compared to classic parametric/semiparametric models as well as super learner under a wide range of distributions. By building an ensemble of algorithms, our method was robust and adapted for situations where the underlying data is mix-distributed. Our simulations also indicated the ability of the two-stage super learner to predict the expenditures in small sample size was as good as in large sample size. The two-stage super learner especially shined with superior predictive accuracy when the proportion of zeros was high. There exist situations, particularly when analyzing inpatient utilization, where more than 70% of zeros occur. Analysis of the MEPS data illustrates consistent results to simulation studies with the optimal performance achieved by two-stage super learner. The smallest CV-MSE obtained from a two-stage super learner was about 16% less than the largest CV-MSE obtained from the model with intercept only. This discrepancy represents a clinically meaningful difference from a hospital administrative perspective. Furthermore, we demonstrated that random forests provided nontrivial improvements compared to parametric regressions. This suggests there may be complex nonlinear relationships and interactions in our MEPS data that parametric regressions were not able to capture. Results in BOLD analysis exhibited the prevailing progress of two-stage super learner over super learner and best single algorithm under all different zero-inflation conditions, although such improvements may sometimes be modest. In addition, we observed that parametric regressions were more appropriate for constructing stage-1 & 2 estimators under high zero-inflation while machine learning algorithms were a better option for stage-1 & 2 estimators under low zero-inflation. It is worth noting that in both empirical studies the CV-$R^2$ for most estimators were relatively low, which is common in health service studies since the prediction of healthcare spending is very difficult. Previous study showed that the diagnosis-based risk adjustment functions have an average $R^2$ of 6.7% (Hermann, Rollins, and Chan 2007). This estimate was based on fitting all observations and likely overestimated the performance compared to cv-$R^2$ we used here.

There are a few limitations in our methodology. Firstly, we included a small set of algorithms in our library and many used the default tuning parameters, which may not be optimal. A natural expansion would be to include a much larger set of algorithms with a range of tuning parameter specifications. It is important to note that there is increased computing time and memory required in implementing ensemble super learning compared to standard regression techniques. Secondly, our implementation of the two-stage super learner did not involve an additional layer of variable screening given the dataset we used is relatively low-dimensional, although the Lasso and random forests performed the variable selection inherently. With high dimensional data, it can be useful to reduce the number of variables considered, thus simplifying the model formula. Finally, we only consider squared-error as the loss function for the two-stage super learner. In practice, there are several choices for loss functions that could be used to evaluate regression fit and the squared-error loss is the most common choice for the continuous outcome. However, this criterion heavily penalizes regions of poor fit in the regression function. As a result, the estimated cross-validated risk based on this loss will be highly sensitive to subjects with large healthcare costs. We could consider other loss functions that are less sensitive to the heavy upper tails of the cost data, such as negative quasi-log-likelihood for bounded continuous outcomes and Huber Loss.

In conclusion, machine learning can be a useful tool for cost estimation, and it provides researchers with alternatives to parametric regressions with ever-increasing numbers of covariates, which may not provide the flexibility necessary in the age of "big data". When additional novel estimators for prediction are developed, they can be easily added to the library of the two-stage super learner, as candidate algorithms. Super learning can augment our learning from data and provide statistical guarantees that we are leveraging the information collected in the strongest possible way. Furthermore, by combining estimators with the weights based on minimizing cross-validated risk, the two-stage super learner could control for over-fitting, even when using a large collection of candidate estimators. In practice, researchers need not spend time and energy guessing which algorithm might perform the best or which variables should be included; they can now use the two-stage super learner to run many at once. The two-stage super learner would either be the best fit or near the best fit. We hope the two-stage super learner proposed here has broader implications for general cost estimation. Those applications include the analysis of expenditures on health and other commodities and services, earnings, and many other economic outcomes that are often skewed to the right.

**References:**


1. Gregori D, Petrinco M, Bo S, Desideri A, Merletti F, Pagano E. Regression models for analyzing costs and their determinants in health care: an introductory review. Int J Qual Health Care. 2011;23(3):331–41.
2. Manning WG and Mullahy J. Estimating log models: to transform or not to transform? Journal of Health Economics 2001; 20(4): 461–494.
3. Jones, A.M, 2010. Models For Health Care Health, Econometrics and Data Group (HEDG) Working Papers 10/01
4. Berk ML,Monheit AC. 2001. The concentration of health care expenditures, revisited. Health Aff. 20:9–18
5. Schuler MS, Rose S. Targeted Maximum Likelihood Estimation for Causal Inference in Observational Studies. Am J Epidemiol. 2017;185: 65–73.
6. Basu, Anirban et al. “ESTIMATING TREATMENT EFFECTS ON HEALTHCARE COSTS UNDER EXOGENEITY: IS THERE A 'MAGIC BULLET'?” Health services & outcomes research methodology vol. 11,1-2 (2011): 1-26.
7. Smith, Jeffrey and Arthur Sweetman, 2016. “Estimating the Causal Effect of Policies and Programs” Canadian Journal of Economics 49 (3): 871–905.
8. Manning WG, Basu A, Mullahy J. Generalized modeling approaches to risk adjustment of skewed outcomes data. J Health Economics. 2005;24(3):465–88.
9. Duan, N., 1983. Smearing estimate: a nonparametric retransformation method. Journal of the American Statistical Association 78, 605–610.
10. Blough DK, Madden CW, Hornbrook MC. 1999. Modeling risk using generalized linear models. J. Health Econ. 18:153–71
11. Huixia Judy Wang, Xiao-Hua Zhou, Estimation of the retransformed conditional mean in health care cost studies, Biometrika, Volume 97, Issue 1, March 2010, Pages 147–158
12. Cawley J, Meyerhoefer C. 2012. The medical care costs of obesity: an instrumental variables approach. J. Health Econ. 31:219–30
13. Finkelstein EA, Trogdon JG, Cohen JW, DietzW. 2009. Annual medical spending attributable to obesity: payer-and service-specific estimates. Health Aff. 28:w 822–31
14. Le Cook B, McGuire TG, Lock K, Zaslavsky AM. 2010. Comparing methods of racial and ethnic disparities measurement across different settings of mental health care.

Health Serv. Res. 45:825–47
15. Deb P, Norton EC. Modeling Health Care Expenditures and Use. Annu Rev Public Health. 2018 Apr 1; 39: 489-505.
16. Rose S. A Machine Learning Framework for Plan Payment Risk Adjustment. Health Serv Res. 2016 Dec;51(6):2358-2374
17. Breiman, L. (1996c). Stacked regressions. Machine Learning, 24(1), 49-64
18. Wolpert, D. (1992), "Stacked Generalization," Neural Networks, 5, 241- 259.
19. M. J. van der Laan and S. Dudoit. Unified Cross-Validation Methodology for Selection Among Estimators and a General Cross-Validated Adaptive Epsilon-Net Estimator: Finite Sample Oracle Inequalities and Examples. Technical Report 130, Division of Biostatistics, University of California, Berkeley, Nov. 2003. URL http://www.bepress.com/ucbbiostat/ paper130/
20. KESSLER, R., ROSE, S., KOENEN, K., KARAM, E., STANG, P., STEIN, D., HEERINGA, S., HILL, E., LIBERZON, I., MCLAUGHLIN, K. and others. (2014). How **well** can post-traumatic stress disorder be predicted from pre-trauma risk factors? an exploratory study in the WHO World Mental Health Surveys. World Psychiatry 13, 265–274.
21. Rose S. Mortality risk score prediction in an elderly population using machine learning. American Journal of Epidemiology 2013; 177(5): 443–452.
22. Pirracchio R, Petersen ML, Carone M et al. Mortality prediction in intensive care units with the super ICU learner algorithm (SICULA): a population-based study. The Lancet Respiratory Medicine 2015; 3(1): 42–52.
23. Jarvik JG, Comstock BA, Bresnahan BW, et al. Study protocol: the Back pain Outcomes using Longitudinal Data (BOLD) Registry. BMC Musculoskeletal Disorders 2012; 13:64.
24. Uwe Reinhardt (December 10, 2010). "The Little-Known Decision-Makers for Medicare Physician Fees". The New York Times. Retrieved July 6, 2011.
25. Roland M, Morris R. A study of the natural history of back pain. Part 1: development of a reliable and sensitive measure of disability in low back pain. Spine 1983; 8: 141–4.
26. Cleeland CS, Nakamura Y, Mendoza TR, Edwards KR, Douglas J, Serlin RC. Dimensions of the impact of cancer pain in a four country sample: new information from multidimensional scaling. Pain 1996; 67: 267–73.
27. Kroenke K, Spitzer RL,Williams JB, Lowe B. An ultra-brief screening scale for anxiety and depression: the PHQ-4. Psychosomatics 2009; 50: 613–21.
28. Brooks R. EuroQOL: the current state of play. Health Policy (New York) 1996; 37: 53–72.
29. Centers for Disease Control and Prevention (CDC). Self-reported falls and fall-related injuries among persons aged > or =65 years—United States, 2006. MMWR Morb Mortal Wkly Rep 2008; 57: 225–9.
30. Iles RA, Davidson M, Taylor NF, O'Halloran P: Systematic review of the ability of recovery expectations to predict outcomes in non-chronic non-specific low back pain. J Occup Rehabil 2009, 19:25–40.
31. Kongsted A, Vach W, Axo M, Bech RN, Hestbaek L: Expectation of recovery from low back pain: a longitudinal cohort study investigating patient characteristics related to expectations and the association between expectations and 3-month outcome. Spine (Phila Pa 1976) 2014, 39:81–90.
32. Quan H, Li B, Couris CM, et al. Updating and validating the Charlson comorbidity index and score for risk adjustment in hospital discharge abstracts using data from 6 countries. Am J Epidemiol 2011; 173: 676–82.
33. David Benkeser, Maya Petersen & Mark J. van der Laan (2019) Improved Small-

Sample Estimation of Nonlinear Cross-Validated Prediction Metrics, Journal of the American Statistical Association, DOI: 10.1080/01621459.2019.1668794
34. Benkeser, David; Cai, Weixin; van der Laan, Mark J. Rejoinder: A Nonparametric Superefficient Estimator of the Average Treatment Effect. Statist. Sci. 35 (2020), no. 3, 511--517.
35. van der Laan MJ and Polley EC. Super learner. Statistical Applications in Genetics and Molecular Biology 2007; 6(1): 1–23.
36. Jarvik JG, Comstock BA, Heagerty PJ, et al. Back pain in seniors. BMC Musculoskeletal Disorders. 2014;15: 134.
37. Jarvik JG, Gold LS, Comstock BA, et al. Association of early imaging for back pain with clinical outcomes in older adults. JAMA. 2015; 313: 1143–1153.
38. Jarvik JG, et al. Long-term outcomes of a large prospective observational cohort of older adults with back pain. Spine J. 2018; 18: 1540–1551.
39. I. Duncan, M. Loginov & M. Ludkovski (2016) Testing Alternative Regression Frameworks for Predictive Modeling of Health Care Costs, North American Actuarial Journal, 20:1, 65-87
40. Morid M.A., Kawamoto K., Ault T. et al. (2018) Supervised learning methods for predicting healthcare costs: systematic literature review and empirical evaluation. AMIA ... Annual symposium proceedings. AMIA Symposium, 2017, 1312–1321.

**Supplement Materials**

**Modifications to two-stage super learner**

We consider two modifications on (i) the weights calculation and (ii) the cross-validation scheme to improve the performance of the two-stage super learner on predicting costs. For continuous outcome, a quadratic programming algorithm designed by Goldfarb and Idnani was generally applied to calculate the best convex combination of weights that minimize the squared error loss. However, the heavy upper tails in healthcare expenditure would result in excessive huge numbers in the matrix and vector of quadratic function to be minimized. This consequently induce overflow errors and cause the quadratic programming constraints inconsistent. As a modification, we proposed a scaling scheme which divide the quadratic function by a large constant to shrink the huge matrix and vector in quadratic function. The scaling would not affect the results given the raw quadratic function is just a multiple of the scaled quadratic function. Alternatively, we consider a modification to the cross-validation scheme in the standard super learning procedure. The standard V-fold cross-validation allocates subjects randomly to each block. However, this random allocation could result in all observed subjects with zero costs or large costs in the same block of data. We might expect a better finite sample evaluation of how well methods fit zero as well as large costs by evenly splitting the zero and large costs amongst the blocks. Considering the minimum of our data is zero, we propose a snake-like assignment of subjects to each block. That is, the V lowest ordered costs are assigned to blocks 1 through V, then the next V lowest ordered cost are assigned reversely to blocks V through 1, respectively. This process is repeated until all subjects have been assigned a block. By splitting up the zero and largest costs, we ensure that every time we fit the candidate methods to V-1 of the blocks, there will be an adequate number of subjects with zero and large costs in the held-out block.

Table 11. The MSE, relative MSE, and $R^2$, averaged over 1000 repetitions of different estimators across 32 data generating processes

| Algorithm | MSE ($10^8$) | Relative MSE | $R^2$ |
|---|---|---|---|
| Two-stage Super Learner | 3.251 | 0.907 | 0.622 |
| Discrete Super Learner | 3.307 | 0.923 | 0.616 |
| Super Learner | 3.382 | 0.944 | 0.611 |
| S1: Lasso + S2: GLM-Gamma-Log | 3.563 | 0.994 | 0.609 |
| S1: Lasso + S2: Quantile regression | 3.567 | 0.996 | 0.608 |
| S1: Lasso + S2: Log OLS-smearing | 3.570 | 0.996 | 0.608 |
| Zero-inflated Negative Binomial (ZINB) | 3.582 | 1.000 | 0.607 |
| S1: GLM + S2: GLM-Gamma-Log | 3.583 | 1.000 | 0.607 |
| S1: GLM + S2: Quantile regression | 3.589 | 1.002 | 0.606 |
| S1: GLM + S2: Log OLS-smearing | 3.590 | 1.002 | 0.606 |
| Zero-inflated Poisson (ZIP) | 3.601 | 1.005 | 0.598 |
| S1: Lasso + S2: Adaptive GLM | 3.784 | 1.056 | 0.597 |
| S1: GLM + S2: Adaptive GLM | 3.803 | 1.061 | 0.595 |
| S1: RF + S2: GLM-Gamma-Log | 4.381 | 1.223 | 0.447 |
| S1: RF + S2: Quantile regression | 4.385 | 1.224 | 0.446 |
| S1: RF + S2: Log OLS-smearing | 4.390 | 1.225 | 0.442 |
| S1: RF + S2: Adaptive GLM | 4.628 | 1.292 | 0.421 |
| S1: GLM + S2: RF | 4.898 | 1.367 | 0.518 |
| S1: Lasso + S2: RF | 4.900 | 1.368 | 0.520 |
| RF | 5.304 | 1.480 | 0.454 |
| S1: RF + S2: RF | 5.506 | 1.537 | 0.444 |
| S1: Lasso + S2: AFT (generalized Gamma) | 6.321 | 1.764 | 0.443 |
| S1: GLM + S2: AFT (generalized Gamma) | 6.339 | 1.769 | 0.441 |
| S1: RF + S2: AFT (generalized Gamma) | 7.049 | 1.967 | 0.309 |
| Tobit | 9.165 | 2.558 | 0.213 |
| S1: Lasso + S2: Lasso (OLS) | 9.295 | 2.594 | 0.349 |
| S1: GLM + S2: Lasso (OLS) | 9.313 | 2.599 | 0.346 |
| Lasso (OLS) | 9.677 | 2.701 | 0.288 |
| OLS | 9.703 | 2.708 | 0.282 |
| S1: RF + S2: Lasso (OLS) | 9.907 | 2.765 | 0.296 |
| S1: Lasso + S2: Adaptive hazard | 9.920 | 2.769 | 0.280 |
| S1: GLM + S2: Adaptive hazard | 9.933 | 2.772 | 0.279 |
| S1: RF + S2: Adaptive hazard | 10.226 | 2.854 | 0.255 |
| Tweedie | 11.273 | 3.146 | 0.246 |
| S1: GLM + S2: Cox hazard | 12.341 | 3.444 | 0.235 |
| S1: Lasso + S2: Cox hazard | 12.353 | 3.448 | 0.233 |
| S1: RF + S2: Cox hazard | 12.820 | 3.578 | 0.184 |
| S1: GLM + S2: GLM-Gamma-Identity | 13.249 | 3.698 | 0.211 |
| S1: Lasso + S2: GLM-Gamma-Identity | 13.263 | 3.702 | 0.210 |
| S1: RF + S2: GLM-Gamma-Identity | 13.717 | 3.828 | 0.166 |
| OLS intercept only (mean) | 17.512 | 4.888 | -0.002 |

Note: Estimators are presented in ascending order according to average MSE. S1 refers to stage-1 and S2 refers to stage-2. RF refers to Random forest. GLM in S1 refers to logistic regression and Lasso in S1 refers to logistic Lasso regression. GLM-Gamma-Identity refers to GLM with Gamma family and Identity link function. The relative MSE is calculated using the MSE of S1: GLM + S2: GLM-Gamma-Log as a reference. The standard error is calculated using 32 averaged metrics across 1000 repetitions.

Table 12. Results of MSE, $R^2$ and Relative Efficiency (RE) for MEPS

| Algorithm | MSE ($10^9$) | $R^2$ | RE |
|---|---|---|---|
| Two-stage Super Learner | 2.180 | 0.147 | 1.000 |
| Discrete Super Learner | 2.192 | 0.143 | 0.969 |
| S1: RF + S2: RF | 2.192 | 0.143 | 0.969 |
| S1: GLM + S2: RF | 2.193 | 0.142 | 0.965 |
| S1: Lasso + S2: RF | 2.194 | 0.141 | 0.961 |
| Super Learner | 2.221 | 0.132 | 0.893 |
| RF | 2.236 | 0.126 | 0.852 |
| Zero-inflated Poisson (ZIP) | 2.257 | 0.119 | 0.810 |
| S1: GLM + S2: Lasso (OLS) | 2.260 | 0.118 | 0.803 |
| S1: Lasso + S2: Lasso (OLS) | 2.260 | 0.117 | 0.798 |
| S1: RF + S2: Lasso (OLS) | 2.261 | 0.117 | 0.796 |
| OLS | 2.264 | 0.116 | 0.789 |
| Lasso (OLS) | 2.265 | 0.115 | 0.782 |
| S1: RF + S2: GLM-Gamma-Log | 2.267 | 0.114 | 0.776 |
| S1: GLM + S2: GLM-Gamma-Log | 2.268 | 0.114 | 0.770 |
| Zero-inflated Negative Binomial (ZINB) | 2.269 | 0.113 | 0.768 |
| S1: Lasso + S2: GLM-Gamma-Log | 2.270 | 0.113 | 0.765 |
| S1: RF + S2: Quantile regression | 2.271 | 0.112 | 0.760 |
| S1: RF + S2: GLM-Gamma-Identity | 2.272 | 0.112 | 0.758 |
| S1: GLM + S2: Quantile regression | 2.273 | 0.111 | 0.755 |
| S1: Lasso + S2: Quantile regression | 2.273 | 0.111 | 0.754 |
| S1: GLM + S2: GLM-Gamma-Identity | 2.275 | 0.110 | 0.749 |
| S1: Lasso + S2: GLM-Gamma-Identity | 2.275 | 0.110 | 0.747 |
| Tobit | 2.283 | 0.107 | 0.728 |
| Tweedie | 2.298 | 0.102 | 0.689 |
| S1: RF + S2: Adaptive hazard | 2.306 | 0.099 | 0.668 |
| S1: GLM + S2: Adaptive hazard | 2.311 | 0.096 | 0.654 |
| S1: Lasso + S2: Adaptive hazard | 2.311 | 0.096 | 0.653 |
| S1: RF + S2: Cox hazard | 2.355 | 0.079 | 0.537 |
| S1: GLM + S2: Cox hazard | 2.359 | 0.078 | 0.526 |
| S1: Lasso + S2: Cox hazard | 2.359 | 0.077 | 0.521 |
| S1: RF + S2: AFT (generalized Gamma) | 2.433 | 0.049 | 0.331 |
| S1: Lasso + S2: AFT (generalized Gamma) | 2.436 | 0.047 | 0.321 |
| S1: GLM + S2: AFT (generalized Gamma) | 2.436 | 0.046 | 0.317 |
| S1: RF + S2: Log OLS-smearing | 2.489 | 0.028 | 0.190 |
| S1: RF + S2: Adaptive GLM | 2.491 | 0.026 | 0.177 |
| S1: Lasso + S2: Log OLS-smearing | 2.492 | 0.025 | 0.170 |
| S1: Lasso + S2: Adaptive GLM | 2.493 | 0.024 | 0.165 |
| S1: GLM + S2: Log OLS-smearing | 2.494 | 0.023 | 0.157 |
| S1: GLM + S2: Adaptive GLM | 2.495 | 0.021 | 0.143 |
| OLS intercept only (mean) | 2.593 | -0.002 | -0.014 |

Note: Estimators are presented in ascending order based on MSE. S1 refers to stage-1 and S2 refers to stage-2. RF refers to Random Forest. GLM in S1 refers to logistic regression and Lasso in S1 refers to logistic Lasso regression. GLM-Gamma-Identity refers to GLM with Gamma family and Identity link function.

Table 13. Rank of algorithms for modeling 4 spine-related RVUs

| Algorithm | SR | SR | SR | SR | Over |
|---|---|---|---|---|---|

| | RVU | imaging RVU | physical therapy RVU | injection RVU | all |
|---|---|---|---|---|---|
| Two-stage Super Learner | 1 | 1 | 1 | 1 | 1 |
| Discrete Super Learner | 3 | 2 | 2 | 2 | 2.25 |
| Super Learner | 2 | 6 | 12 | 11 | 7.75 |
| S1: Lasso + S2: Lasso | 9 | 11 | 8 | 3 | 7.75 |
| SL: GLM + S2: Lasso | 10 | 10 | 7 | 5 | 8 |
| S1: RF + S2: Lasso | 8 | 4 | 22 | 14 | 12 |
| S1: Lasso + S2: GLM-Gamma-Identity | 20 | 21 | 5 | 4 | 12.5 |
| S1: Lasso + S2: RF | 7 | 17 | 13 | 18 | 13.75 |
| S1: GLM + S2: GLM-Gamma-Identity | 21 | 22 | 6 | 7 | 14 |
| S1: Lasso + S2: Log OLS smearing | 17 | 27 | 3 | 9 | 14 |
| S1: GLM + S2: Log OLS smearing | 19 | 28 | 4 | 6 | 14.25 |
| S1: RF + S2: Log OLS smearing | 14 | 9 | 19 | 16 | 14.25 |
| S1: GLM + S2: RF | 6 | 16 | 18 | 19 | 14.75 |
| S1: Lasso + S2: GLM-Gamma-Log | 16 | 24 | 11 | 8 | 14.75 |
| S1: RF + S2: RF | 5 | 5 | 25 | 25 | 15 |
| S1: RF + S2: GLM-Gamma-Identity | 19 | 7 | 24 | 12 | 15.5 |
| S1: RF + S2: GLM-Gamma-Log | 14 | 8 | 26 | 15 | 15.75 |
| S1: GLM + S2: GLM-Gamma-Log | 18 | 26 | 14 | 10 | 17 |
| S1: Lasso + S2: GBM | 24 | 15 | 9 | 22 | 17.5 |
| Single: Lasso | 11 | 12 | 34 | 13 | 17.5 |
| S1: GLM + S2: GBM | 26 | 13 | 10 | 23 | 18 |
| S1: RF + S2: GBM | 23 | 3 | 23 | 28 | 19.25 |
| Single: GBM | 27 | 19 | 15 | 20 | 20.25 |
| Single: OLS | 12 | 18 | 35 | 17 | 20.5 |
| Single: RF | 4 | 20 | 37 | 21 | 20.5 |
| S1: RF + S2: Bagging | 25 | 14 | 27 | 30 | 24 |
| S1: Lasso + S2: Bagging | 29 | 25 | 16 | 26 | 24 |
| S1: GLM + S2: Bagging | 28 | 23 | 21 | 29 | 25.25 |
| S1: GLM + S2: Neural Net | 33 | 30 | 17 | 24 | 26 |
| S1: Lasso + S2: Neural Net | 32 | 31 | 20 | 27 | 27.5 |
| Single: Bagging | 22 | 32 | 29 | 32 | 28.75 |
| S1: RF + S2: Neural Net | 31 | 29 | 28 | 36 | 31 |
| S1: RF + S2: MARS | 38 | 33 | 33 | 38 | 35.5 |
| Single: Neural Net | 30 | 40 | 41 | 31 | 35.5 |
| S1: RF + S2: Single tree | 35 | 34 | 40 | 34 | 35.75 |
| S1: Lasso + S2: MARS | 39 | 36 | 30 | 39 | 36 |
| Single: Single tree | 34 | 41 | 32 | 37 | 36 |
| S1: Lasso + S2: Single tree | 36 | 38 | 38 | 33 | 36.25 |
| S1: GLM + S2: MARS | 41 | 35 | 31 | 40 | 36.75 |
| S1: GLM + S2: Single tree | 37 | 37 | 39 | 35 | 37 |
| Single: earth | 40 | 39 | 36 | 41 | 39 |

Note: Estimators are presented in ascending order based on averaged rank. SR refers to Spine-Related. RF refers to Random forest. GLM in S1 refers to logistic regression and Lasso in S1 refers to logistic Lasso regression. MARS refers ro multivariate adaptive regression splines. Bagging refers to bootstrap aggregating and GBM refers to gradient boosting machine.