Electronic Journal of Statistics
Vol. 9 (2015) 1583–1607
ISSN: 1935-7524
DOI: 10.1214/15-EJS1035

Computationally efficient confidence intervals for cross-validated area under the ROC curve estimates

Erin LeDell[*], **Maya Petersen**[†] **and Mark van der Laan**[‡]

Division of Biostatistics
University of California, Berkeley
Berkeley, CA 94720
USA

e-mail: ledell@berkeley.edu; mayaliv@berkeley.edu; laan@berkeley.edu

Abstract: In binary classification problems, the area under the ROC curve (AUC) is commonly used to evaluate the performance of a prediction model. Often, it is combined with cross-validation in order to assess how the results will generalize to an independent data set. In order to evaluate the quality of an estimate for cross-validated AUC, we obtain an estimate of its variance. For massive data sets, the process of generating a single performance estimate can be computationally expensive. Additionally, when using a complex prediction method, the process of cross-validating a predictive model on even a relatively small data set can still require a large amount of computation time. Thus, in many practical settings, the bootstrap is a computationally intractable approach to variance estimation. As an alternative to the bootstrap, we demonstrate a computationally efficient influence curve based approach to obtaining a variance estimate for cross-validated AUC.

MSC 2010 subject classifications: Primary 62G15, 62G05; secondary 62G20.
Keywords and phrases: AUC, binary classification, confidence intervals, cross-validation, influence curve, influence function, machine learning, model selection, ROC, variance estimation.

Received December 2014.

Contents

[*]Corresponding author.
[†]Supported by the Doris Duke Charitable Foundation under Grant No. 2011042.
[‡]Supported by the National Institutes of Health under Grant No. R01 AI074345.

1. Introduction

The area under the ROC curve, or AUC, is a ranking-based measure of performance in binary classification problems. Its value can be interpreted as the probability that a randomly selected positive sample will rank higher than a randomly selected negative sample. AUC is a more discriminating performance measure than accuracy [1], and is invariant to relative class distributions [2].

In practice, we are generally concerned with how well our results will generalize to new data. Cross-validation is a means of obtaining an estimate that is generalizable to data drawn from the same distribution but not used in the training set. Common types of cross-validation procedures include k-fold [3], leave-one-out [21, 7, 3], and leave-p-out [20] cross-validation. Given the advantages of AUC as a performance measure, along with the desire to produce generalizable results, cross-validated AUC is frequently used in binary classification problems.

An important task in any estimation procedure is rigorously quantifying the uncertainty in the estimates. In many cases, specification of a parametric model known to contain the truth is not possible, and approaches to inference which are robust to model misspecification are therefore needed. Two approaches to robust inference include inference based on resampling methods, and inference based on influence curves (also known as influence functions). In practice, resampling methods such as the nonparametric bootstrap [11, 12], are commonly used due to their generic nature and simplicity. However, when data sets are large or when methods for training a prediction model are complex, bootstrapping can quickly become a computationally prohibitive procedure.

Although cross-validation lends itself well to parallelization, it can still take a very long time to generate a cross-validated performance measure, such as cross-validated AUC, depending on the complexity of the algorithm used to train the prediction model or the size of the training set. In machine learning, ensemble methods are prediction methods that make use of, or combine, several or many candidate learning algorithms to obtain better predictive performance. This boost in performance is often accompanied by an increase in the time it takes to generate cross-validated predictions. Alternatively, given massive data sets, even simple prediction methods can be computationally expensive. In cases where obtaining a single estimate of cross-validated AUC requires a significant amount of time and/or resources, the bootstrap is either not an option, or at the very least, a undesirable option for obtaining variance estimates.

As a response to the computational costs of the bootstrap, variations of the bootstrap have been developed that achieve a more desirable computational footprint, such as the "m out of n bootstrap" [9] and subsampling [19]. Another recent advancement that has been made in this area is the "Bag of Little Bootstraps" (BLB) method [4]. Unlike previous variations, BLB simultaneously addresses computational costs, statistical correctness and automation, which appears to be a promising generalized method for variance estimation on massive data sets.

Regardless of the reduction in computation that different variations of the bootstrap offer, all bootstrapping variants require repeated estimation on at least some subset of the original data. By using influence curves for variance estimation, we avoid the need to re-estimate our parameter of interest, which in the case of cross-validated AUC, requires fitting additional models. In order to estimate variance using influence curves, you must first, unsurprisingly, calculate the influence curve for your estimator. For complex estimators, it can be a difficult task to derive the influence curve. However, once the derivation is complete, variance estimation is reduced to a simple and computationally negligible calculation. This is the main motivation for our use of influence curves as a means of variance estimation.

The main goal of this paper is to establish an influence curve based approach for estimating the asymptotic variance of the cross-validated area under the ROC curve estimator. We first define true cross-validated AUC along with a corresponding estimator and then provide a brief overview of influence curve based variance estimation. We derive the influence curve for the AUC of both i.i.d. data and pooled repeated measures data (multiple observations per independent sampling unit, such as a patient), and demonstrate the construction of influence curve based confidence intervals. We conclude with a simulation that evaluates the coverage probability of the confidence intervals and provide a comparison to bootstrapped based confidence intervals. The methods are implemented in a publicly available R package called **cvAUC** [16].

2. Cross-validated AUC as a target parameter

In this section, we formally introduce AUC. We then define the estimator for cross-validated AUC, as well as the target that it is estimating, the true cross-validated AUC.

Consider some probability distribution, P_0, that is known to be an element of a statistical model, $\mathcal{M}$. Let $O = (X, Y) \sim P_0 \in \mathcal{M}$, where Y is a binary outcome variable, and $X \in \mathbb{R}^p$ represents one or more covariates or predictor variables ($p \geq 1$). Without loss of generality, we will denote $Y = 1$ as the positive class and $Y = 0$ as the negative class, and ψ as a function that maps X into $(0, 1)$. The quantity, $\psi(X)$, is the predicted value or score of a sample. The Area Under the ROC curve can be defined as the following:

$$AUC(P_0, \psi) = \int_0^1 P_0\left(\psi(X) > c \mid Y = 1\right) P_0\left(\psi(X) = c \mid Y = 0\right) dc. \qquad (2.1)$$

Alternatively, we can define AUC as

$$AUC(P_0, \psi) = P_0\left(\psi(X_1) > \psi(X_2) \mid Y_1 = 1, Y_2 = 0\right), \tag{2.2}$$

where (X_1, Y_1) and (X_2, Y_2) are i.i.d. samples from P_0. The quantity, $AUC(P_0, \psi)$, the true AUC, equals the probability, conditional on sampling two independent observations where one is positive ($Y_1 = 1$) and the other is negative ($Y_2 = 0$), that the predicted value (or rank) of the positive sample, $\psi(X_1)$, is higher than the predicted value (or rank) of the negative sample, $\psi(X_2)$.

Consider $O_1, \ldots, O_n$, i.i.d. samples from P_0, such that $O_i = (X_i, Y_i)$ for each i, and let P_n denote the empirical distribution. Let n_0 be the number of observations with $Y = 0$ and let n_1 be the number of observations with $Y = 1$. In machine learning, the ψ function is what is learned by a binary prediction algorithm using the training data. The AUC of the empirical distribution can be written as follows:

$$\begin{aligned} AUC(P_n, \psi) &= \frac{1}{n_0 n_1} \sum_{i=1}^{n} \sum_{j=1}^{n} I\left(\psi(X_j) > \psi(X_i)\right) I\left(Y_i = 0, Y_j = 1\right) \\ &= \frac{1}{n_0 n_1} \sum_{i=1}^{n_0} \sum_{j=1}^{n_1} I\left(\psi(X_j) > \psi(X_i)\right), \end{aligned}$$

where I is the indicator function.

We focus on estimating cross-validated AUC. We do not require that the cross-validation be any particular type; however, in practice, k-fold is common. We will use a generalized notation to encode the data splitting procedure, where a binary indicator vector is used to specify which observations belong to the validation set at each iteration of the cross-validation process. Let $B_n \in \{0, 1\}^n$ be a random split of the observations into a training and validation set, and define P^1_{n,B_n} and P^0_{n,B_n} as the empirical distributions of the validation set, $\{i : B_n(i) = 1\}$, and training set, $\{i : B_n(i) = 0\}$, respectively. Let $B^1_n, \ldots, B^V_n$ be the collection of random splits that define our cross-validation procedure, where $B^v_n \in \{0, 1\}^n$. In the case of k-fold cross-validation, $k = V$, and each of the B^v_n encodes a single fold; the v^{th} validation fold is the set of observations indexed by $\{i : B^v_n(i) = 1\}$, and the remaining observations belong to the v^{th} training set, $\{i : B^v_n(i) = 0\}$.

Let $\mathcal{M}_{NP}$ denote a nonparametric model that includes the empirical distribution, P_n, and let $\hat{\Psi} : \mathcal{M}_{NP} \to \mathbb{R}$ be an estimator of target parameter, ψ_0, true cross-validated AUC. We assume that $\hat{\Psi}(P_0) = \psi_0$.

For each B^v_n, we define $\psi_{B^v_n} = \hat{\Psi}(P^0_{n,B^v_n})$, where P^0_{n,B^v_n} is the empirical distribution of the observations contained in the v^{th} training set. The function $\psi_{B^v_n}$, which is learned from the v^{th} training set, will be used to generate predicted values for the observations in the v^{th} validation fold. We define n^v_1 and n^v_0 to be the number of positive and negative samples in the v^{th} validation fold, respectively. Formally, $n^v_1 = \sum_{i=1}^{n} I\left(Y_i = 1\right) I\left(B^v_n(i) = 1\right)$ and $n^v_0 = \sum_{i=1}^{n} I\left(Y_i = 0\right) I\left(B^v_n(i) = 1\right)$. We note that n^v_1 and n^v_0 are random variables that depend on the value of both B^v_n and $\{Y_i : B^v_n(i) = 1\}$. The AUC for

a single validation fold, $\{i : B_n^v(i) = 1\}$, is:

$$AUC(P^1_{n,B_n^v}, \psi_{B_n^v}) =$$
$$\frac{1}{n_0^v n_1^v} \sum_{i=1}^{n} \sum_{j=1}^{n} I\left(\psi_{B_n^v}(X_j) > \psi_{B_n^v}(X_i)\right) I\left(Y_i = 0, Y_j = 1\right) I\left(B_n^v(i) = B_n^v(j) = 1\right).$$

Then the V-fold cross-validated AUC estimator is defined as:

$$\begin{aligned} E_{B_n} AUC(P^1_{n,B_n}, \psi_{B_n}) &= \frac{1}{V} \sum_{v=1}^{V} AUC(P^1_{n,B_n}, \psi_{B_n^v}) \\ &= \frac{1}{V} \sum_{v=1}^{V} \frac{1}{n_0^v n_1^v} \sum_{i=1}^{n} \sum_{j=1}^{n} I\left(\psi_{B_n^v}(X_j) > \psi_{B_n^v}(X_i)\right) \\ &\times \; I\left(Y_i = 0, Y_j = 1\right) I\left(B_n^v(i) = B_n^v(j) = 1\right). \end{aligned} \tag{2.3}$$

The target, ψ_0, of the V-fold cross-validated AUC estimator is defined as:

$$\begin{aligned} E_{B_n} AUC(P_0, \psi_{B_n}) &= \frac{1}{V} \sum_{v=1}^{V} AUC(P_0, \psi_{B_n^v}) \\ &= \frac{1}{V} \sum_{v=1}^{V} P_0\left(\psi_{B_n^v}(X_1) > \psi_{B_n^v}(X_2) \mid Y_1 = 1, Y_2 = 0\right), \end{aligned} \tag{2.4}$$

where (X_1, Y_1) and (X_2, Y_2) are i.i.d. samples from P_0. In other words, our target parameter, the true cross-validated AUC, corresponds to fitting the prediction function on each training set, evaluating its true performance (or true probability of correctly ranking two randomly selected observations, where one is a positive sample and the other a negative sample) in the corresponding validation set, and finally, taking the average over the validation sets. The true value of this target parameter is random, in that it depends on the split of the sampled data into training sets and corresponding fits of the prediction function. We now wish to construct confidence intervals for our estimator of cross-validated AUC, $E_{B_n} AUC(P^1_{n,B_n}, \psi_{B_n})$.

3. Influence curves for variance estimation

We provide a brief overview of influence curves and their relation to variance estimation. We outline the general procedure for obtaining confidence intervals using the influence curve of an estimator. This section serves as a gentle introduction to concepts and notation used throughout the paper.

Suppose that $O \equiv O_1, \ldots, O_n$ are i.i.d. samples from a probability distribution, P_0, that is known to be an element of a statistical model, $\mathcal{M}$. Let $\mathcal{F}$ be some class of functions of O. Throughout this paper, we will use the notation Pf, where P is a probability distribution, to denote $\int f(x) dP(x)$. We consider the empirical process, $(P_0 f : f \in \mathcal{F})$, which is a "vector" of true means. Let

$\Psi : \mathcal{M} \to \mathbb{R}^d$ be a parameter of interest, and let $\psi_0 = \Psi(P_0) \equiv \Psi(P_0 f : f \in \mathcal{F})$ be the true parameter value; ψ_0 is a function of true means. Now let $\mathcal{M}_{NP}$ denote a nonparametric model that includes the empirical distribution, P_n, of $O_1, \ldots, O_n$. We consider the empirical process, $(P_n f : f \in \mathcal{F})$, which is a "vector" of empirical means. Let $\hat{\Psi} : \mathcal{M}_{NP} \to \mathbb{R}^d$ be an estimator of ψ_0 that maps the empirical distribution, P_n, or rather, a "vector" of empirical means, into $\mathbb{R}^d$. Let $\hat{\Psi}(P_n) \equiv \hat{\Psi}(P_n f : f \in \mathcal{F})$. We assume that $\hat{\Psi}(P_0) = \psi_0$, so that the estimator targets the desired target parameter, ψ_0. This estimate is *asymptotically linear* at P_0 if

$$\hat{\Psi}(P_n) - \hat{\Psi}(P_0) = \frac{1}{n} \sum_{i=1}^{n} IC(P_0)(O_i) + o_P(1/\sqrt{n}) \tag{3.1}$$

for some zero-mean function, $IC(P_0)$, of O (i.e. $P_0 IC(P_0) = 0$). The function, $IC(P_0)$, that results from demonstrating asymptotic linearity is called the *influence curve* (or *influence function*) of the estimator, $\hat{\Psi}$. The main task in the process of constructing influence curve based confidence intervals is demonstrating the asymptotic linearity of your estimator.

By the Central Limit Theorem, we find that $\sqrt{n}\,(\hat{\Psi}(P_n) - \hat{\Psi}(P_0)) \xrightarrow{d} \mathcal{N}(0, \Sigma_0)$, where $\Sigma_0 = P_0 IC(P_0) IC(P_0)^T$. This covariance matrix can be estimated with the empirical covariance matrix $\widehat{IC}(O_i)$, $i = 1, \ldots, n$ where $\widehat{IC}$ is an estimate of $IC(P_0)$. When our target parameter is one-dimensional, as in cross-validated AUC, we can write the following:

$$\sqrt{n}\,\left(\hat{\Psi}(P_n) - \hat{\Psi}(P_0)\right) \xrightarrow{d} \mathcal{N}\left(0, \Phi^2(P_0)\right), \tag{3.2}$$

where $\Phi^2(P_0) = \int IC(P_0)(x)^2 dP_0(x)$. We can estimate $\Phi^2(P_0)$ as

$$\Phi_n^2 \equiv \Phi^2(P_n) = \frac{1}{n} \sum_{i=1}^{n} IC(P_n)(O_i)^2, \tag{3.3}$$

however, other estimators of the variance of the influence curve can be considered. Letting z_r denote the r^{th} quantile of the standard normal distribution, it follows that for any estimate $\Phi_n^2 \equiv \Phi^2(P_n)$ of $\Phi^2(P_0)$, we have that

$$\left(\hat{\Psi}(P_n) - z_{1-\alpha/2} \frac{\Phi_n}{\sqrt{n}}\ ,\ \ \hat{\Psi}(P_n) + z_{1-\alpha/2} \frac{\Phi_n}{\sqrt{n}}\right) \tag{3.4}$$

forms an approximate $100 \times (1-\alpha)\%$ confidence interval for $\psi_0 \equiv \hat{\Psi}(P_0)$.

In order to assume that asymptotically linear estimators of ψ_0 exist, we must assume that the parameter Ψ is pathwise differentiable [10]. This method for establishing the asymptotic linearity and normality of the estimator is called the *functional delta method* [22, 14], which is a generalization of the classical delta method for finite dimensional functions of a finite set of estimators.

4. Confidence intervals for cross-validated AUC

In this section, we establish the influence curve for AUC and show that the empirical AUC is an asymptotically linear estimator of the true AUC. Using these results, we follow the methodology from Section 3 to derive confidence intervals for cross-validated AUC. Then we provide a description of the practical construction of the confidence intervals from an i.i.d. data sample.

Theorem 4.1. *Let $O = (W, Y) \sim P_0$, where W represents one or more variables and Y is binary. Without loss of generality, assume $Y \in \{0, 1\}$ and that ψ is a function that maps W into (0,1). Define $AUC(P_0, \psi)$ as*

$$\int_0^1 P_0\left(\psi(X) > c \mid Y = 1\right) P_0\left(\psi(X) = c \mid Y = 0\right) dc.$$

The efficient influence curve of $AUC(P_0, \psi)$, evaluated at a single observation, $O_i = (X_i, Y_i)$, for a nonparametric model for P_0 is given by

$$\begin{aligned} IC_{AUC}(P_0, \psi)(O_i) = {} & \frac{I(Y_i = 1)}{P_0(Y = 1)} P_0\left(\psi(X) < w \mid Y = 0\right)\Big|_{w=\psi(X_i)} \\ & + \frac{I(Y_i = 0)}{P_0(Y = 0)} P_0\left(\psi(X) > w \mid Y = 1\right)\Big|_{w=\psi(X_i)} \\ & - \left\{ \frac{I(Y_i = 0)}{P_0(Y = 0)} + \frac{I(Y_i = 1)}{P_0(Y = 1)} \right\} AUC(P_0, \psi). \end{aligned}$$

For each ψ, the empirical $AUC(P_n, \psi)$ is asymptotically linear with influence curve $IC_{AUC}(P_0, \psi)$. Let $B_n \in \{0, 1\}^n$ be a random split of the observations into a training and validation set. Let P^1_{n,B_n} and P^0_{n,B_n} be the empirical distributions of the validation set, $\{i : B_n(i) = 1\}$, and training set, $\{i : B_n(i) = 0\}$, respectively. We assume that B_n has only a finite number of values uniformly in n, as in V-fold cross-validation. We assume that $p = \sum_i B_n(i)/n$ is bounded away from a $\delta > 0$, with probability 1. Define the cross-validated area under the ROC curve as

$$\hat{R}(\hat{\Psi}, P_n) = E_{B_n} AUC\left(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n})\right). \tag{4.1}$$

We also define the target of this cross-validated area under the ROC curve as

$$\tilde{R}(\hat{\Psi}, P_n) = E_{B_n} AUC\left(P_0, \hat{\Psi}(P^0_{n,B_n})\right). \tag{4.2}$$

We assume that there exists a $\psi_1 \in \mathbf{\Psi}$ so that

$$P_0\left\{ IC_{AUC}\left(P_0, \hat{\Psi}(P_n)\right) - IC_{AUC}(P_0, \psi_1) \right\}^2$$

converges to zero in probability as $n \to \infty$. We also assume that

$$\sup_{\psi \in \mathbf{\Psi}} \sup_O |IC_{AUC}(P_0, \psi)(O)| < \infty,$$

where the supremum over O is over a support of P_0. Then,

$$\hat{R}(\hat{\Psi}, P_n) - \tilde{R}(\hat{\Psi}, P_n) = \frac{1}{n}\sum_{i=1}^{n} IC_{AUC}(O_i) + o_P(1/\sqrt{n}). \tag{4.3}$$

In particular, $\sqrt{n}(\hat{R}(\hat{\Psi}, P_n) - \tilde{R}(\hat{\Psi}, P_n))$ converges to a normal distribution with mean zero and variance, $\sigma^2 = P_0\{IC_{AUC}(P_0, \psi_1)\}^2$. Thus, one can construct an asymptotically 0.95-confidence interval for $\tilde{R}(\hat{\Psi}, P_n)$ given by $\hat{R}(\hat{\Psi}, P_n) \pm 1.96\frac{\sigma_n}{\sqrt{n}}$, where σ_n^2 is a consistent estimator of σ^2. A consistent estimator of σ^2 is obtained as

$$\sigma_n^2 = E_{B_n} P^1_{n,B_n} \left\{ IC_{AUC}\left(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n})\right)\right\}^2. \tag{4.4}$$

Proof. In order to derive influence curve based confidence intervals for cross-validated AUC, we must first derive the influence curve for AUC and show that $AUC(P_n, \psi)$ is an asymptotically linear estimator of $AUC\,(P_0, \psi)$ with influence curve as specified in the theorem. For that purpose we use the functional delta method [22, 14]. The asymptotic linearity of $AUC(P_n, \psi)$ is an immediate consequence of the compact differentiability of functionals $(F_1, F_2) \to \int F_1(x)dF_2(x)$ for cumulative distribution functions (F_1, F_2) in [14] so that the functional delta-method can be applied here as well. Therefore it only remains to determine the actual influence curve which is defined in terms of the Gateaux derivative of $P \to AUC(P, \psi)$ in the direction of the empirical distribution for a single observation O. We will do that now.

We define $F_a(c) = P_0(\psi(X) < c \mid Y = a)$ for $a \in \{0, 1\}$. Therefore, we can alternatively express true AUC as

$$AUC\,(P_0, \psi) = \Phi(F_0, F_1) = \int (1 - F_1(c))dF_0(c). \tag{4.5}$$

The Gateaux derivative of $\Phi(F_0, F_1)$ in direction (h_0, h_1) is given by:

$$\frac{d}{d\epsilon}\Phi(F_0 + \epsilon h_0, F_1 + \epsilon h_1)\Big|_{\epsilon=0} = \int -h_1(c)dF_0(c) + \int (1 - F_1(c))dh_0(c)$$

Therefore, we have the following linear approximation:

$$\Phi(F_{0n}, F_{1n}) - \Phi(F_0, F_1) = \int -(F_{1n} - F_1)dF_0 + \int (1 - F_1)d(F_{0n} - F_0)$$

Let F_{0n}, F_{1n} be the empirical distributions of F_0, F_1. Next we derive the linear approximations of $F_{1n} - F_1$ and $F_{0n} - F_0$. Note that for $a \in \{0, 1\}$,

$$F_{an}(c) = P_n(\psi(X) < c \mid Y = a) = \frac{P_n(\psi(X) < c, Y = a)}{P_n(Y = a)} \tag{4.6}$$

It follows that $F_{an}(c) - F_a(c) \sim$

$$\frac{(P_n - P_0)(\psi(X) < c, Y = a)}{P_0(Y = a)} - \frac{P_0(\psi(X) < c, Y = a)}{[P_0(Y = a)]^2}(P_n(Y = a) - P_0(Y = a))$$
$$= \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{I(\psi(X_i) < c, Y_i = a)}{P_0(Y = a)} - \frac{F_a(c)I(Y_i = a)}{P_0(Y = a)}\right\}$$

So the influence curve of $F_{an}(c)$ for a single observation, $O_i = (X_i, Y_i)$, is:

$$\frac{I(\psi(X_i) < c, Y_i = a)}{P_0(Y = a)} - \frac{F_a(c)I(Y_i = a)}{P_0(Y = a)} \tag{4.7}$$

We can substitute this for h_a in the linear approximation above resulting in the desired influence curve $IC_{AUC}(P_0, \psi)$ as presented in the theorem. For that, it is helpful to observe that:

$$\int I(\psi(X_i) < c, Y_i = 1)dF_0(c) = I(Y_i = 1)\int_{\psi(X_i)<c} dF_0(c) \tag{4.8}$$
$$= I(Y_i = 1)(1 - F_0(\psi(X_i)) \tag{4.9}$$

This is the influence curve for $AUC(P_n, \psi)$, and, since the model $\mathcal{M}$ for P_0 is nonparametric, this is also the efficient influence curve of parameter $AUC(P_0, \psi)$ on a nonparametric model.

Using the notation that was defined in Section 2, it follows that

$$\begin{aligned}
&E_{B_n}AUC(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n})) - E_{B_n}AUC(P_0, \hat{\Psi}(P^0_{n,B_n}))\\
&\quad= E_{B_n}(P^1_{n,B_n} - P_0)IC_{AUC}(P_0, \hat{\Psi}(P^0_{n,B_n})) + E_{B_n}R(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n}))\\
&\quad\approx E_{B_n}(P^1_{n,B_n} - P_0)IC_{AUC}(P_0, \hat{\Psi}(P^0_{n,B_n})) + o_P(1/\sqrt{n})\\
&\quad= E_{B_n}(P^1_{n,B_n} - P_0)IC_{AUC}(P_0, \psi_1)\\
&\qquad+ E_{B_n}(P^1_{n,B_n} - P_0)\left\{IC_{AUC}(P_0, \hat{\Psi}(P^0_{n,B_n})) - IC_{AUC}(P_0, \psi_1)\right\}\\
&\qquad+ o_P(1/\sqrt{n})\\
&\quad= E_{B_n}(P^1_{n,B_n} - P_0)IC_{AUC}(P_0, \psi_1) + o_P(1/\sqrt{n})\\
&\quad= (P_n - P_0)IC_{AUC}(P_0, \psi_1) + o_P(1/\sqrt{n}).
\end{aligned}$$

At the first equality we apply the previously established asymptotic linearity of $AUC(P_{n,B^1_n}, \psi)$, conditional on the training sample, which proves that each B_n-specific remainder $R(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n}))$ is $o_P(1/\sqrt{n})$. Since there are only a finite number of possible B_n, this also proves the next equivalence stating that the average across the different B_n-splits of the remainder is also $o_P(1/\sqrt{n})$. In the third equality, we just carry out a simple split of the empirical process in two terms. In the statement of the theorem, we assume that $P_0\{IC_{AUC}(P_0, \hat{\Psi}(P_n)) - IC_{AUC}(P_0, \psi_1)\}^2$ converges to zero in probability as $n \to \infty$ for some ψ_1. Using a result from [23] involving the application

of empirical process theory (specifically Lemma 2.14.1 in [22]), the term, $E_{B_n}(P^1_{n,B_n} - P_0)\{IC_{AUC}(P_0, \hat{\Psi}(P^0_{n,B_n})) - IC_{AUC}(P_0, \psi_1)\}$, is shown to be $o_P(1/\sqrt{n})$, which results in the fourth equality.

Finally, $E_{B_n}(P^1_{n,B_n} - P_0)IC_{AUC}(P_0, \psi_1) = (P_n - P_0)IC_{AUC}(P_0, \psi_1)$, proving the asymptotic linearity of the cross-validated AUC estimator as stated in the final equality. In particular,

$$\sqrt{n}\left(E_{B_n}AUC(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n})) - E_{B_n}AUC(P_0, \hat{\Psi}(P^0_{n,B_n}))\right)$$

converges to a normal distribution with mean zero and variance, $\sigma^2 = P_0\left\{IC_{AUC}(P_0, \psi_1)\right\}^2$. A consistent estimator of σ^2 is obtained as

$$\sigma_n^2 = E_{B_n}P^1_{n,B_n}\left\{IC_{AUC}\left(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n})\right)\right\}^2.$$

For σ_n^2, we estimate the unknown conditional probabilities of the influence curve IC_{AUC} with the empirical distribution of the validation set, so that $P^1_{n,B_n}(\psi(X) > w \mid Y = 0)$ will be consistent at $\psi = \hat{\Psi}(P^0_{n,B_n})$ under no conditions on the estimator $\hat{\Psi}$. This is why we replaced P_0 in $IC_{AUC}(P_0, \psi)$ by the empirical distribution of the validation set. However, the probabilities $P_0(Y = 1)$ and $P_0(Y = 0)$ can be estimated using the whole sample.

Thus, one can construct an asymptotically 0.95-confidence interval for $E_{B_n}AUC(P_0, \hat{\Psi}(P^0_{n,B_n}))$ given by $E_{B_n}AUC(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n})) \pm 1.96\frac{\sigma_n}{\sqrt{n}}$.

□

4.1. A practical implementation for i.i.d. data

For further clarity, we provide a description of the practical construction of the confidence intervals from an i.i.d. data set, as implemented in our software package. Consider an i.i.d. sample of size n with a binary outcome Y. For each observation, $O_i = (X_i, Y_i)$, we have a d-dimensional numeric vector X_i (design matrix) and a binary outcome, Y_i. Without loss of generality, let $Y_i \in \{0, 1\}$, for all $i = 1, \ldots, n$, however, Y can be any ordered two-class variable. In this example, we will use k-fold cross-validation for $k = V > 1$ and define the splits as $B^1_n, \ldots, B^V_n$, as defined previously. Calculating the V-fold cross validated AUC estimate corresponds to:

1. Building or fitting the prediction function on each of V validation sets.
2. Generating a predicted outcome for each observation in the v^{th} validation set. The predictions are generated using a fit that was trained on the $\{1, \ldots, V\} \setminus v$ folds.
3. For each validation fold, using these predicted values, together with the observed outcomes for each observation, to generate an estimate of the AUC for that validation fold.
4. Average these estimates across the V validation folds to calculate the cross-validated AUC.

Recall that P^1_{n,B^v_n} and P^0_{n,B^v_n} are the empirical distributions of the v^{th} validation and training set, respectively and P_n is the empirical distribution of the whole data sample. The V-fold cross-validated AUC estimate, denoted $\hat{R}(\hat{\Psi}, P_n)$, is given by $\frac{1}{V}\sum_{v=1}^{V} AUC(P^1_{n,B_n}, \psi_{B^v_n})$. In order to construct influence curve based confidence intervals for $\hat{R}(\hat{\Psi}, P_n)$, we estimate the asymptotic variance as:

$$\sigma_n^2 = E_{B_n} P^1_{n,B_n} \left\{ IC_{AUC}\left(P^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n})\right)\right\}^2 \tag{4.10}$$

$$= \frac{1}{V}\sum_{v=1}^{V}\left\{\frac{1}{n}\sum_{i=1}^{n}\left\{IC_{AUC}\left(P^1_{n,B^v_n}, \hat{\Psi}(P^0_{n,B^v_n})\right)(O_i)\right\}^2 I\left(B^v_n(i)=1\right)\right\}, \tag{4.11}$$

where $\psi_{B^v_n} = \hat{\Psi}(P^0_{n,B^v_n})$, and for each $v \in \{1,\ldots,V\}$ and $i \in \{1,\ldots,n\}$, we have

$$\begin{aligned} &IC_{AUC}(P^1_{n,B^v_n}, \hat{\Psi}(P^0_{n,B^v_n}))(O_i) \\ &\qquad = \frac{I(Y_i=1)}{P_n(Y=1)} P^1_{n,B^v_n}\left(\psi_{B^v_n}(X) < w \mid Y=0\right)\Big|_{w=\psi_{B^v_n}(X_i)} \\ &\qquad\quad + \frac{I(Y_i=0)}{P_n(Y=0)} P^1_{n,B^v_n}\left(\psi_{B^v_n}(X) > w \mid Y=1\right)\Big|_{w=\psi_{B^v_n}(X_i)} \\ &\qquad\quad - \left\{\frac{I(Y_i=0)}{P_n(Y=0)} + \frac{I(Y_i=1)}{P_n(Y=1)}\right\} AUC\left(P^1_{n,B_n}, \psi_{B^v_n}\right). \end{aligned}$$

Despite the density of the notation above, each of the components in the influence curve can be calculated very easily from the data. The terms, $P_n(Y=1) \equiv \frac{1}{n}\sum_{j=1}^{n} I(Y_j=1)$ and $P_n(Y=0) \equiv \frac{1}{n}\sum_{j=1}^{n} I(Y_j=0)$, are the proportions of positive and negative samples, respectively, in the empirical distribution. Let $n_1^v = \sum_{j=1}^{n} I(Y_j=1) I(B^v_n(j)=1)$ be the number of positive samples in the v^{th} validation set and let $n_0^v = \sum_{j=1}^{n} I(Y_j=0) I(B^v_n(j)=1)$ be the number of negative samples in the v^{th} validation set. Also, recall that $\psi_{B^v_n}$ is the function learned by the v^{th} training set, which maps a vector, W, of covariates, to a predicted value, $\psi_{B^v_n}(X) \in (0,1)$. For a given sample, $O_i = (X_i, Y_i)$, we calculate the predicted value, $\psi_{B^v_n}(X_i)$, and note whether Y_i is labeled as positive ($Y_i = 1$) or negative ($Y_i = 0$). Above, each of the terms in the expression for the influence curve contains an indicator function, conditional on the value of Y_i. Therefore, given the value of Y_i, we need only to evaluate the non-zero part of the expression.

When $Y_i = 1$, we need to evaluate:

$$\begin{aligned} &P^1_{n,B^v_n}\left(\psi_{B^v_n}(X) < w \mid Y=0\right)\Big|_{w=\psi_{B^v_n}(X_i)} \\ &\qquad = \frac{1}{n^v_0}\sum_{j=1}^{n} I\left(X_j < \psi_{B^v_n}(X_i)\right) I\left(Y_j=0\right) I\left(B^v_n(j)=1\right) \end{aligned}$$

This sum counts the number of *negative* samples in the validation set that have a predicted value *less than* $\psi_{B_n^v}(X_i)$, the predicted value for sample i. Then, we divide by the total number of negative samples in the validation set. Similarly, when $Y_i = 0$, we need to evaluate:

$$P^1_{n,B_n^v}\left(\psi_{B_n^v}(X) > w \mid Y = 1\right)\Big|_{w=\psi_{B_n^v}(X_i)} = \frac{1}{n_0^v}\sum_{j=1}^{n} I\left(X_j > \psi_{B_n^v}(X_i)\right) I\left(Y_j = 1\right) I\left(B_n^v(j) = 1\right)$$

This sum counts the number of *positive* samples in the validation set that have a predicted value *greater than* $\psi_{B_n^v}(X_i)$, the predicted value for sample i. Then, we divide by the total number of positive samples in the validation set. The remaining term in the expression for the influence curve is simply $AUC(P^1_{n,B_n}, \psi_{B_n^v})$, given in Section 3, multiplied by inverse probability of $P_n(Y = 1)$ or $P_n(Y = 0)$, depending on the value of the indicator function at Y_i. Thus, for fixed $v \in \{1, \ldots, V\}$ and $i \in \{1, \ldots, n\}$, we have demonstrated how to calculate the quantity, $IC_{AUC}(P^1_{n,B_n^v}, \hat{\Psi}(P^0_{n,B_n^v}))(O_i)$, from an i.i.d. data set. Then we square this term and sum over i.i.d. samples, i, and cross-validation folds, v, to get

$$\sigma_n^2 = \frac{1}{V}\sum_{v=1}^{V}\left\{\frac{1}{n}\sum_{i=1}^{n}\left\{IC_{AUC}\left(P^1_{n,B_n^v}, \hat{\Psi}(P^0_{n,B_n^v})\right)(O_i)\right\}^2 I\left(B_n^v(i) = 1\right)\right\},$$

an estimate for the asymptotic variance of $\hat{R}(\hat{\Psi}, P_n)$, our V-fold cross-validated AUC estimator. The target of this estimator is

$$\tilde{R}(\hat{\Psi}, P_n) = E_{B_n} AUC\left(P_0, \hat{\Psi}(P^0_{n,B_n})\right) = \frac{1}{V}\sum_{v=1}^{V} AUC\left(P_0, \hat{\Psi}(P^0_{n,B_n^v})\right),$$

the true V-fold cross-validated AUC. Then, as in Theorem 4.1, one can construct an asymptotically 0.95-confidence interval for $\tilde{R}(\hat{\Psi}, P_n)$ as $\hat{R}(\hat{\Psi}, P_n) \pm 1.96\frac{\sigma_n}{\sqrt{n}}$.

5. Generalization to pooled repeated measures data

Above, we derived a consistent influence curve based estimator of the asymptotic variance of cross-validated AUC for the simple setting in which there are n i.i.d. observations. Each of these observations, O_i has a predictor variable, X_i, coupled with a binary outcome variable, Y_i, that we wish to predict. Now we consider the common setting in which there are repeated measures for each observation. This data structure arises frequently in medical studies, where each patient is measured at multiple time points. We focus on the case where the order of these measures is not meaningful, and one simply wishes to obtain a single summary of classifier performance pooled over all measures. We begin by providing a formal definition of the target parameter, the pooled cross-validated AUC, for

such cases. We then extend the results presented in the previous sections to derive an influence curve based variance estimator for the cross-validated AUC of a pooled repeated measures data set.

As before, we let $P_0 \in \mathcal{M}$ and $\Psi : \mathcal{M} \to \mathbf{\Psi}$. We denote the target parameter $\Psi(P_0)$ as ψ_0. Let $O = (X(t), Y(t) : t \in \tau) \sim P_0$ for a possibly random index set $\tau \subset \{1, \ldots, T\}$, where t corresponds to a single time-point observation. Here $Y(t)$ is binary for each t. We observe n i.i.d. copies $O_i = (X_i(t), Y_i(t) : t \in \tau_i)$, $i = 1, \ldots, n$ of O. Let $\mathcal{M}_{NP}$ denote a nonparametric model that includes the empirical distribution, P_n, of $O_1, \ldots, O_n$ and let $\hat{\Psi} : \mathcal{M}_{NP} \to \mathbb{R}$ be an estimator of ψ_0. We assume that $\hat{\Psi}(P_0) = \psi_0$. We consider the case where t is not a meaningful index, and that either $\psi_0(t, x) = E_0(Y(t) \mid X(t) = x)$ does not depend on t, or that the investigator has no interest in understanding the dependence on t. Consider the distribution,

$$\bar{P}_0(x, y) = \frac{1}{E_0|\tau|} \sum_{t=1}^{T} P_0(t \in \tau) P_0(X(t) = x, Y(t) = y \mid t \in \tau).$$

This represents the limit distribution of the empirical distribution $\bar{P}_n$ of the pooled sample:

$$\bar{P}_n(x, y) = \frac{1}{\sum_{i=1}^{n} |\tau_i|} \sum_{i=1}^{n} \sum_{t \in \tau_i} I(X_i(t) = x, Y_i(t) = y).$$

One could define as a measure of interest for evaluation a predictor ψ, the area under the ROC curve one would obtain if one treats the pooled sample as N i.i.d. observations. That is, we define

$$\overline{AUC}\left(\bar{P}_0, \psi\right) = \int_0^1 \bar{P}_0(\psi(X) > c \mid Y = 1) \bar{P}_0(\psi(X) = c \mid Y = 0) \, dc, \qquad (5.1)$$

where, without loss of generality, we let the positive class be represented by $Y = 1$ and the negative class be represented by $Y = 0$. The pooled repeated measures AUC can be interpreted as the probability that, after pooling over all independent sampling units and all time points, a randomly sampled positive outcome will be ranked more highly than a randomly sampled negative outcome.

The AUC for the empirical distribution of the pooled sample can be expressed explicitly as follows. Let $n_0 = \sum_{i=1}^{n} \sum_{t \in \tau_i} I(Y_i(t) = 0)$ and let $n_1 = \sum_{j=1}^{n} \sum_{s \in \tau_j} I(Y_j(s) = 1)$. Then we have

$$\overline{AUC}\left(\bar{P}_n, \psi\right) = \frac{1}{n_0 n_1} \sum_{i=1}^{n} \sum_{t \in \tau_i} \sum_{j=1}^{n} \sum_{s \in \tau_j} I(\psi(X_j(s)) > \psi(X_i(t))) I(Y_i(t) = 0, Y_j(s) = 1).$$

Now we consider the cross-validated AUC of a pooled repeated measures data set. Let $B_n \in \{0, 1\}^n$ be a random split of the n independent observations into a

training and validation set. Let $\bar{P}^1_{n,B_n}$ and $\bar{P}^0_{n,B_n}$ be the empirical distributions of the pooled data within the validation set, $\{i : B_n(i) = 1\}$, and training set, $\{i : B_n(i) = 0\}$, respectively. We assume that B_n has only a finite number of values uniformly in n, as in k-fold cross-validation. Given a random split, B_n, we define $\psi_{B_n} \equiv \hat{\Psi}(\bar{P}^0_{n,B_n})$.

As in the i.i.d. example in the previous section, we will walk through the case of V-fold cross-validation. Let $B^1_n, \ldots, B^V_n$ be the collection of random splits that define our cross-validation procedure. In the case of V-fold cross-validation, each of the B^v_n encodes a single fold; the v^{th} validation fold is $\{i : B^v_n(i) = 1\}$, and the remaining samples belong to the v^{th} training set, $\{i : B^v_n(i) = 0\}$. Note that since our independent units are collections of pooled time points, $O_i = (X_i(t), Y_i(t) : t \in \tau_i)$, that all pooled samples from each i.i.d. sample, O_i will be contained within the same validation fold.

For each B^v_n, we define $\psi_{B^v_n} \equiv \hat{\Psi}(\bar{P}^0_{n,B^v_n})$, where $\bar{P}^0_{n,B^v_n}$ is the empirical distribution of the pooled data contained in the v^{th} training set. The function $\psi_{B^v_n}$, which is learned from the v^{th} training set, will be used to generate predicted values for the observations in the v^{th} validation fold. We define n^v_1 and n^v_0 to be the number of positive and negative samples in the v^{th} validation fold, respectively. Formally, $n^v_1 = \sum_{i=1}^n \sum_{t\in\tau_i} I\left(Y_i(t) = 1\right) I\left(B^v_n(i) = 1\right)$ and $n^v_0 = \sum_{i=1}^n \sum_{t\in\tau_i} I\left(Y_i(t) = 0\right) I\left(B^v_n(i) = 1\right)$. We note that n^v_1 and n^v_0 are random variables that depend on the value of both B^v_n and $\{Y_i : B^v_n(i) = 1\}$.

The AUC for a single validation fold, $\{i : B^v_n(i) = 1\}$, for pooled repeated measures data, is

$$\overline{AUC}\left(\bar{P}^1_{n,B^v_n}, \psi_{B^v_n}\right) = \frac{1}{n^v_0 n^v_1} \sum_{i=1}^n \sum_{t\in\tau_i} \sum_{j=1}^n \sum_{s\in\tau_j} h(n, v, i, t, j, s) \tag{5.2}$$

where $h(n, v, i, t, j, s) =$

$$I\left(\psi_{B^v_n}(X_j(s)) > \psi_{B^v_n}(X_i(t))\right) I\left(Y_i(t) = 0, Y_j(s) = 1\right) I\left(B^v_n(i) = B^v_n(j) = 1\right).$$

In other words, it is the probability that, after pooling over units and time, a randomly drawn positive sample will be assigned a higher predicted value than a randomly drawn negative sample in the same validation fold by the prediction model fit using the corresponding training set.

Then the V-fold cross-validated AUC estimator, for pooled repeated measures data, is defined as

$$E_{B_n}\overline{AUC}\left(\bar{P}^1_{n,B_n}, \psi_{B_n}\right) = \frac{1}{V}\sum_{v=1}^V \overline{AUC}\left(\bar{P}^1_{n,B^v_n}, \psi_{B^v_n}\right) \tag{5.3}$$

$$= \frac{1}{V}\sum_{v=1}^V \left\{ \frac{1}{n^v_0 n^v_1} \sum_{i=1}^n \sum_{t\in\tau_i} \sum_{j=1}^n \sum_{s\in\tau_j} h(n, v, i, t, j, s) \right\}. \tag{5.4}$$

We also define the target, ψ_0, of the V-fold cross-validated AUC estimate as

$$E_{B_n}\overline{AUC}\left(\bar{P}_0, \psi_{B_n}\right) = \frac{1}{V}\sum_{v=1}^{V}\overline{AUC}\left(\bar{P}_0, \psi_{B_n^v}\right) \tag{5.5}$$

$$= \frac{1}{V}\sum_{v=1}^{V}\bar{P}_0\left(\psi_{B_n^v}(X_1) > \psi_{B_n^v}(X_2) \mid Y_1 = 1, Y_2 = 0\right), \tag{5.6}$$

where $(X_1, Y_1) \equiv (X_1(t), Y_1(t))$ and $(X_2, Y_2) \equiv (X_2(t), Y_2(t))$ are single time-point observations. The following theorem is the pooled repeated measures analogue to Theorem 4.1.

Analogous to i.i.d. data version, this target represents the average across validation folds of the true probability (under P_0) that a randomly sampled positive observation would be ranked higher than a randomly sampled negative observation in the same validation fold by the prediction function fit in the corresponding training set. Again, the true value of this target parameter is random – it depends on the random split of the sample into V folds and corresponding fits of the prediction function. However, it nonetheless provides a meaningful measure of the performance of the prediction function on independent data.

Theorem 5.1. *The efficient influence curve of* $\overline{AUC}\left(\bar{P}_0, \psi\right)$*, evaluated at* $O_i = (X_i(t), Y_i(t)) : t \in \tau_i)$*, for a nonparametric model for* P_0 *is given by:*

$$IC_{\overline{AUC}}\left(\bar{P}_0, \psi\right)(O_i) = \frac{1}{E_0|\tau|}\sum_{t\in\tau} IC_{AUC}\left(\bar{P}_0, \psi\right)(X_i(t), Y_i(t)),$$

where

$$\begin{aligned} &IC_{AUC}\left(\bar{P}_0, \psi\right)(X_i(t), Y_i(t)) \\ &\quad= \frac{I(Y_i(t)=1)}{\bar{P}_0(Y=1)}\bar{P}_0\left(\psi(X) < w \mid Y(t) = 0\right)\Big|_{w=\psi(X_i(t))} \\ &\qquad+ \frac{I(Y_i(t)=0)}{\bar{P}_0(Y=0)}\bar{P}_0\left(\psi(X) > w \mid Y(t) = 1\right)\Big|_{w=\psi(X_i(t))} \\ &\qquad- \left\{\frac{I(Y_i(t)=0)}{\bar{P}_0(Y=0)} + \frac{I(Y_i(t)=1)}{\bar{P}_0(Y=1)}\right\} AUC\left(\bar{P}_0, \psi\right), \end{aligned}$$

Directly above, $(W, Y) \equiv (W(s), Y(s))$ *represents a single time-point observation. For each* ψ*, the estimator* $\overline{AUC}\left(\bar{P}_n, \psi\right)$ *obtained by plugging in the pooled empirical distribution* $\bar{P}_0$*, is asymptotically linear with influence curve* $IC_{\overline{AUC}}\left(\bar{P}_0, \psi\right)$*.*

Let $B_n \in \{0,1\}^n$ *be a random split and let* P^1_{n,B_n} *and* P^0_{n,B_n} *be the empirical distributions of the validation* $\{i : B_n(i) = 1\}$ *and training set* $\{i : B_n(i) = 0\}$*, respectively. Let* $\bar{P}^1_{n,B_n}$ *be the empirical distribution of the pooled data within the validation set. We assume that* B_n *has only a finite number of values uniformly in* n*, as in* k*-fold cross-validation. We assume that* $p = \sum_i B_n(i)/n$ *is bounded*

away from a $\delta > 0$, with probability 1. Define the cross-validated area under the ROC curve as

$$\hat{R}(\hat{\Psi}, P_n) = E_{B_n} \overline{AUC} \left(\bar{P}^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n}) \right). \tag{5.7}$$

We also define the target of this cross-validated area under the ROC curve as

$$\tilde{R}(\hat{\Psi}, P_n) = E_{B_n} \overline{AUC} \left(\bar{P}_0, \hat{\Psi}(P^0_{n,B_n}) \right). \tag{5.8}$$

We assume that there exists a $\psi_1 \in \mathbf{\Psi}$ so that

$$P_0 \left\{ IC_{\overline{AUC}} \left(\bar{P}_0, \hat{\Psi}(P_n) \right) - IC_{\overline{AUC}} \left(\bar{P}_0, \psi_1 \right) \right\}^2$$

converges to zero in probability as $n \to \infty$. We also assume that

$$\sup_{\psi \in \mathbf{\Psi}} \sup_{O} \left| IC_{\overline{AUC}} \left(\bar{P}_0, \psi \right) (O) \right| < \infty,$$

where the supremum over O is over a support of P_0. Then,

$$\hat{R}(\hat{\Psi}, P_n) - \tilde{R}(\hat{\Psi}, P_n) = \frac{1}{n} \sum_{i=1}^{n} IC_{\overline{AUC}} \left(\bar{P}_0, \psi_1 \right) (O_i) + o_P(1/\sqrt{n}). \tag{5.9}$$

In particular, $\sqrt{n}(\hat{R}(\hat{\Psi}, P_n) - \tilde{R}(\hat{\Psi}, P_n))$ converges to a normal distribution with mean zero and variance, $\sigma^2 = P_0\{IC_{\overline{AUC}}(\bar{P}_0, \psi_1)\}^2$. Thus, one can construct an asymptotically 0.95-confidence interval for $\tilde{R}(\hat{\Psi}, P_n)$ given by $\hat{R}(\hat{\Psi}, P_n) \pm 1.96 \frac{\sigma_n}{\sqrt{n}}$ where σ_n^2 is a consistent estimator of σ^2. A consistent estimator of σ^2 is obtained as

$$\sigma_n^2 = E_{B_n} P^1_{n,B_n} \left\{ IC_{\overline{AUC}} \left(\bar{P}^1_{n,B_n}, \hat{\Psi}(P^0_{n,B_n}) \right) \right\}^2. \tag{5.10}$$

Proof. This is the pooled repeated measures analogue of Theorem 4.1, so the proof follows the exact same format and arguments as the proof of Theorem 4.1. □

6. Software

We implemented the influence curve based confidence intervals for cross-validated AUC for i.i.d. data as well as for pooled repeated measures data, as an R package. The package, called **cvAUC** [16], has the same function interface as the popular **ROCR** package [5].

For each observation, the user provides a cross-validated predicted value, as generated by a binary prediction algorithm, and a corresponding binary class label. If the user has pooled repeated measures data instead of i.i.d. data, then the user must also provide an id for each observation. The user must also indicate which observations belong to each cross-validation fold. To be clear, the user must provide for each observation, i:

1. The value of the outcome, Y_i.
2. The validation fold, $v \in \{1, \ldots, V\}$, that observation, i, is associated with.
3. The predicted probability of the outcome, $\psi(X_i)$, based on plugging in that observation's covariates, X_i, into a fit trained on the observations associated with folds: $\{1, \ldots, V\} \backslash v$.

The main functions of the package calculate the confidence intervals (confidence level supplied by the user; defaults to 95%) for cross-validated AUC and AUC estimates calculated using i.i.d. and pooled repeated measures training data. The package also includes utility functions to compute AUC and cross-validated AUC from a set of predicted values and associated true labels.

To provide some context to the computational efficiency of our methods, the influence curve based CV AUC variance calculation for i.i.d. data takes less than half a second to execute for a sample of 100,000 observations on a 2.3 GHz Intel Core i7 processor (package version 1.0.3). For 1 million observations, it currently takes 13 seconds. More information and code examples can be found in the user manual for the package, and we provide a simple code example in Appendix A. The **cvAUC** R package is available on CRAN and GitHub. More information and code examples can be found in the user manual for the package.

7. Coverage probability of the confidence intervals

In this section, we describe and present results from a simulation which demonstrates the coverage probability of our influence curve based confidence intervals as implemented in our R package, **cvAUC** [16]. The *coverage probability* of a confidence interval is the proportion of the time, over repetitions of the identical experiment, that the interval contains the true value of interest. Our true value of interest is true cross-validated AUC, defined in equation 2.4. In the simulation below, we consider a variety of training set sizes. We show that when n is small, the coverage probability of the influence curve based confidence interval may drop below the specified rate. Therefore, if you have a small sample size, bootstrapping may serve as a computationally-reasonable alternative variance estimation technique. To quantify the computational advantage of the influence curve approach, we calculate the number of bootstrap replicates that are required in order to achieve 95% coverage.

7.1. Simulation to evaluate coverage probability

Let $n \times p$ represent the dimensions of our training set design matrix, $\boldsymbol{X}$. We considered training sets where $n = \{500, 1000, 5000, 10000, 20000\}$ and $p = \{10, 50, 100, 200\}$. The number of covariates that are correlated with the outcome is fixed at 10. The remaining $p-10$ covariates are random noise. For the 10 informative covariates, we generate 100,000 points from $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, and for each these observations, we let $Y = 0$. Similarly, we generate 100,000 observations from $\mathcal{N}(\boldsymbol{\nu}, \boldsymbol{\Sigma})$ and let $Y = 1$ for all these observations. For this simulation, we

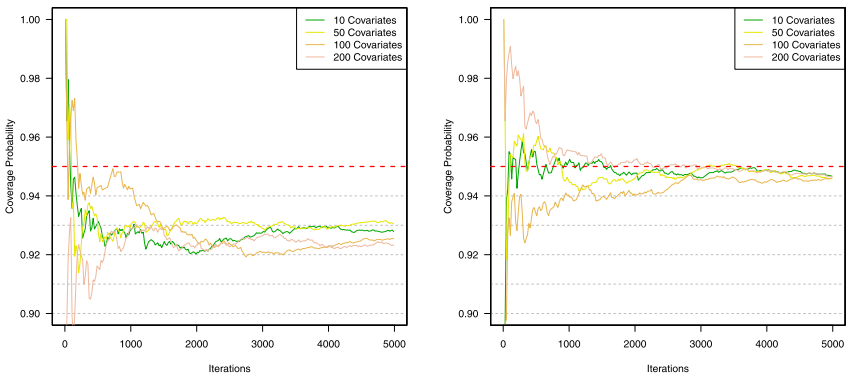

FIG 1. *Plots of the coverage probabilities for 95% confidence intervals generated by our simulation for training sets of 1,000 (left) and 5,000 (right) observations. In the case of a 95% confidence interval, the coverage probability should be close to 0.95. For the smaller dataset of* $n = 1,000$ *observations, we see that the coverage is slightly lower (92-93%) than specified, whereas for* $n = 5,000$*, the coverage is closer to 95%.*

let $\mu_i = 0$ and $\nu_i = 0.3$, for $i \in \{1, \ldots, 10\}$ and we let $\mathbf{\Sigma}$ represent the identity covariance matrix. These combined 200,000 observations represent our true data distribution, P_0. We note that our target parameter, true cross-validated AUC, is itself random, but that it represents a true target. We are interested in the confidence interval that contains this random target 95% of the time. The samples were generated using the `mvrnorm` function of the R package, **MASS** [6].

To calculate the coverage probability of our influence curve based confidence intervals, we generate the CV AUC and corresponding confidence intervals 5,000 times and report the proportion of times that the confidence interval contains the true CV AUC. For each iteration, we sample n points from the same distribution as our population data and use that as a training set.

We perform 10-fold cross-validation by splitting these n observations into 10 validation folds, stratifying by outcome, Y. For each validation fold, we train a Lasso-regularized logistic regression fit using the **glmnet** R package [13] using the observations from the remaining 9 folds. Using the fit model, we then generate predictions for each of the samples in the validation fold and calculate the empirical AUC. We will call this the "fold AUC." We also calculate the "true AUC" by generating predicted values for all of the 200,000 data points in our population data and calculating the empirical AUC among this population.

This process is repeated for each of the 10 validation folds, at which point we average the fold AUCs to get the estimate for cross-validated AUC. We also average the 10 true AUCs to get the true cross-validated AUC. We then calculate a 95% confidence interval for our CV AUC estimate and note whether or not the true CV AUC falls within the confidence interval.

For each value of $p \in \{10, 50, 100, 200\}$, this process is repeated 5,000 times to obtain an estimate of the coverage probability of our confidence intervals. The

TABLE 1
Coverage probability for influence curve based confidence intervals for CV AUC using training sets of various dimension

	$n = 500$	$n = 1,000$	$n = 5,000$	$n = 10,000$	$n = 20,000$
$p = 10$	0.909	0.928	0.946	0.943	0.943
$p = 50$	0.891	0.931	0.946	0.950	0.941
$p = 100$	0.885	0.925	0.946	0.946	0.949
$p = 200$	0.878	0.923	0.947	0.937	0.940

TABLE 2
Influence curve based standard errors for CV AUC for training sets of various dimensions

	$n = 500$	$n = 1,000$	$n = 5,000$	$n = 10,000$	$n = 20,000$
$p = 10$	0.023	0.015	0.007	0.005	0.003
$p = 50$	0.023	0.016	0.007	0.005	0.003
$p = 100$	0.024	0.016	0.007	0.005	0.003
$p = 200$	0.024	0.016	0.007	0.005	0.003

TABLE 3
Standard deviation of 5,000 CV AUC estimates for training sets of various dimensions

	$n = 500$	$n = 1,000$	$n = 5,000$	$n = 10,000$	$n = 20,000$
$p = 10$	0.028	0.017	0.007	0.005	0.003
$p = 50$	0.033	0.018	0.007	0.005	0.003
$p = 100$	0.034	0.019	0.007	0.005	0.003
$p = 200$	0.038	0.019	0.007	0.005	0.003

coverage probability is the proportion times that the true CV AUC fell within our confidence interval. For 95% confidence intervals, we expect the coverage probability to be close to 0.95. The coverage probabilities for each training set is shown in Table 1.

The results of the simulation indicate that for a relatively small sample size (e.g. $n = 1,000$), the coverage probability of the confidence intervals are slightly lower (92-93%) than specified (95%). However, when $n \geq 5,000$, we have coverage between 94-95%. These simulations use just one particular data generating distribution, but the results can serve as a rough benchmark of coverage probability rates over various n.

In Table 2, we summarize the standard errors estimated using the influence curve based variance estimation technique, as implemented in the **cvAUC** package. For comparison, in Table 3 we report the standard deviation of the CV AUC estimates across the 5,000 iterations of the simulation. We see that for $n \geq 5,000$, the standard errors and standard deviations are identical, however, for smaller n, the influence curve based standard errors are slightly conservative compared to the standard deviation across the 5,000 iterations. This is expected, based on the coverage probabilities reported in Table 1.

For reference, we provide the average CV AUC estimate across 5,000 iterations for training sets of various dimensions in Table 4. A total of $20 \times 5,000 = 500,000$ cross validated AUC estimates were generated for the entire simulation. The number of individual models that were trained across all 10 folds was $500,000 \times 10 = 5$ million.

TABLE 4
Average CV AUC across 5,000 iterations for training sets of various dimensions

	$n = 500$	$n = 1,000$	$n = 5,000$	$n = 10,000$	$n = 20,000$
$p = 10$	0.720	0.737	0.747	0.748	0.748
$p = 50$	0.706	0.733	0.747	0.748	0.748
$p = 100$	0.699	0.731	0.747	0.748	0.748
$p = 200$	0.689	0.728	0.747	0.748	0.748

TABLE 5
Bootstrap confidence interval coverage probability using B bootstrapped replicates of a training set of $n = 1,000$ observations

	$B = 100$	$B = 200$	$B = 300$	$B = 400$
$p = 10$	0.906	0.930	0.929	0.958

7.2. Comparison to bootstrapped confidence intervals

We implemented quantile (or percent) bootstrapped confidence intervals in Julia [8] (version 0.0.3) to compare the coverage probability of bootstrap derived confidence intervals to influence curve derived confidence intervals. The same data generating distributions [18] as the influence curve based simulations were used, and again we used Lasso-regularized logistic regression [15]. For each iteration of the experiment, we generate an original training set and B bootstrapped replicates of the this training set. Using the B training sets, we generate B cross-validated AUC estimates [17]. We use the 0.025 and 0.975 quantiles of the B cross-validated AUCs to estimate the 95% confidence intervals. In this simulation, the computation time for bootstrapped confidence intervals is $O(B)$ times greater than the runtime of the influence curve based confidence intervals since each bootstrap replicate requires a complete re-calculation of CV AUC. Some methods of bootstrapping (e.g. m of out n bootstrap [9] and "Bag of Little Bootstraps" [4]) make computational improvements on $o(B)$, however all bootstrapping methods require you to make repeated estimations of CV AUC.

On a training set of $n = 1,000$ observations, we evaluated how many bootstrapped replicates, B, are required to obtain 95% coverage. In this simulation, we found that at least 400 bootstrap replicates were required to obtain a coverage probability of 0.95. The coverage probabilities for increasing values of B are shown in Table 5.

Since the bootstrap confidence interval coverage probability estimate converged after approximately 1,000 iterations of the experiment, the coverage probability estimates in Table 5 are averaged over 1,000 iterations instead of 5,000.

8. Conclusion

Cross-validated AUC represents an attractive and commonly used measure of performance in binary classification problems. However, resampling based approaches to constructing confidence intervals for this quantity can be computationally expensive. In this paper, we established the asymptotical linearity of

the cross-validated AUC estimator and derived its influence curve for both the i.i.d. and pooled repeated measures cases. We then presented a computationally efficient approach to constructing confidence intervals based on estimating this influence curve, which is implemented as a publicly available R package called **cvAUC**. A simulation demonstrated that we were able to achieve the expected coverage probability for our confidence intervals, however, for small sample sizes, the coverage probability can dip below the desired rate. We have demonstrated a computationally efficient alternative to bootstrapping for estimating the variance of cross-validated AUC estimates. This technique for generating computationally efficient confidence intervals can be replicated for another estimator by following the same procedure.

Appendix A: Appendix

A.1. Code example

Below is a simple example of how to use the **cvAUC** R package. This i.i.d. data example does the following:

1. Load a data set with a binary outcome. For the i.i.d. case we use a simulated data set of 500 observations, included with the package, of graduate admissions data.
2. Divide the indices randomly into 10 folds, stratifying by outcome. Stratification is not necessary, but is commonly performed in order to create validation folds with similar distributions. Store this information in a list called `folds`.
3. Define a function to fit a model on the training data and to generate predicted values for the observations in the validation fold, for a single iteration of the cross-validation procedure. We use a logistic regression fit.
4. Apply this function across all folds to generate predicted values for each validation fold. The concatenated version of these predicted values is stored in vector called `predictions`. The outcome vector, Y, is the `labels` argument.

A code example is given on the following page.

```
# Create CV folds (stratify by outcome)
.cvFolds <- function(Y, V){
  Y0 <- split(sample(which(Y == 0)),
              rep(1:V, length = length(which(Y == 0))))
  Y1 <- split(sample(which(Y == 1)),
              rep(1:V, length = length(which(Y == 1))))
  folds <- vector("list", length = V)
  for (v in seq(V)) {folds[[v]] <- c(Y0[[v]], Y1[[v]])}
  return(folds)
}


# Train/test glm for each fold
.doFit <- function(v, folds, data){
  fit <- glm(Y~., data = data[-folds[[v]],], family = binomial)
  pred <- predict(fit, newdata = data[folds[[v]],], type = "response")
  return(pred)
}

iid_example <- function(data, V = 10){

  # Create folds
  folds <- .cvFolds(Y = data$Y, V = V)

  # CV train/predict
  predictions <- unlist(sapply(seq(V), .doFit,
                               folds = folds, data = data))
  # Re-order pred values
  predictions[unlist(folds)] <- predictions

  # Get CV AUC and confidence interval
  out <- ci.cvAUC(predictions = predictions, labels = data$Y,
              folds = folds, confidence = 0.95)
  return(out)
}


# Run example
library(cvAUC)
data(admissions)
set.seed(1)
out <- iid_example(data = admissions, V = 10)
```

The output is given as follows:

```
# > out
# $cvAUC
# [1] 0.9046473
#
# $se
# [1] 0.01620238
#
# $ci
# [1] 0.8728913 0.9364034
#
# $confidence
# [1] 0.95
```

In the i.i.d. example above, we provided cross-validated predicted values, fold indices, and class labels (0/1) to the `ci.cvAUC` function while using a default confidence level of 95%. The cross-validated AUC is shown to be approximately 0.905, with an estimated standard error of 0.016. The corresponding 0.95% confidence interval for the CV AUC is approximately [0.873, 0.936].

Acknowledgements

We would like to thank the developers of the **ROCR** R package [5].

References

[1] Ling, C., Huang, J., and Zhang, H. (2003). AUC: a statistically consistent and more discriminating measure than accuracy. *Proceedings of IJCAI 2003*.

[2] Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition 30*, 1145–1159.

[3] Geisser, S. (1975). The predictive sample reuse method with applications. *Amer. Statist. Assoc. 70*, 320–328.

[4] Kleiner, A., Talwalkar, A., Sarkar, P., and Jordan, M. (2013). A scalable bootstrap for massive data. *Journal of the Royal Statistical Society, Series B*.

[5] Sing, T., Sander, O., Beerenwinkel, N., and Lengauer, T. (2005). ROCR: Visualizing classifier performance in R. *Bioinformatics* **21**, 20, 3940–3941.

[6] Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*, Fourth ed. Springer, New York.

[7] Allen, D. M. (1974). The relationship between variable selection and data augmentation and a method for prediction. *Technometrics 16*, 125–127. MR0343481 (49 #8222)

[8] Bezanson, J., Karpinski, S., Shah, V. B., and Edelman, A. (2012). Julia: A fast dynamic language for technical computing. *CoRR abs/1209.5145*. http://arxiv.org/abs/1209.5145.

[9] Bickel, P. J., Götze, F., and van Zwet, W. R. (1997). Resampling fewer than n observations: gains, losses, and remedies for losses. *Statist. Sinica* **7**, 1, 1–31. Empirical Bayes, sequential analysis and related topics in statistics and probability (New Brunswick, NJ, 1995). MR1441142 (98g:62079)

[10] Bickel, P. J., Klaassen, C. A. J., Ritov, Y., and Wellner, J. A. (1993). *Efficient and adaptive estimation for semiparametric models*. Johns Hopkins Series in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD. MR1245941 (94m:62007)

[11] Efron, B. (1979). Bootstrap methods: another look at the jackknife. *Ann. Statist.* **7**, 1, 1–26. MR515681 (80b:62021)

[12] Efron, B. and Tibshirani, R. J. (1993). *An introduction to the bootstrap*. Monographs on Statistics and Applied Probability, Vol. **57**. Chapman and Hall, New York. MR1270903 (95h:62077)

[13] Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* **33**, 1, 1–22. http://www.jstatsoft.org/v33/i01/.

[14] Gill, R. D. (1989). Non- and semi-parametric maximum likelihood estimators and the von Mises method. I. *Scand. J. Statist.* **16**, 2, 97–128. With a discussion by J. A. Wellner and J. Præstgaard and a reply by the author. MR1028971 (91d:62042)

[15] Kornblith, S. (2014). *GLMNet.jl: Julia wrapper for fitting Lasso/Elastic-Net GLM models using glmnet*. Commit version 0526df8455, https://github.com/simonster/GLMNet.jl.

[16] LeDell, E., Petersen, M., and van der Laan, M. (2013). *cvAUC: Cross-Validated Area Under the ROC Curve Confidence Intervals*. R package version 1.0-0, http://CRAN.R-project.org/package=cvAUC.

[17] Lin, D. (2014). *A set of functions to support the development of machine learning algorithms*. v0.4.2, https://github.com/JuliaStats/MLBase.jl.

[18] Lin, D. and White, J. M. (2014). *A Julia package for probability distributions and associated functions*. v0.5.4, https://github.com/JuliaStats/Distributions.jl.

[19] Politis, D. N., Romano, J. P., and Wolf, M. (1999). *Subsampling*. Springer Series in Statistics. Springer-Verlag, New York. http://dx.doi.org/10.1007/978-1-4612-1554-7. MR1707286 (2001d:62047)

[20] Shao, J. (1993). Linear model selection by cross-validation. *J. Amer. Statist. Assoc.* **88**, 422, 486–494. MR1224373 (94k:62107)

[21] Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *J. Roy. Statist. Soc. Ser. B 36*, 111–147. With discussion by G. A. Barnard, A. C. Atkinson, L. K. Chan, A. P. Dawid, F. Downton, J. Dickey, A. G. Baker, O. Barndorff-Nielsen, D. R. Cox, S. Giesser, D. Hinkley, R. R. Hocking, and A. S. Young, and with a reply by the authors. MR0356377 (50 #8847)

[22] VAN DER VAART, A. W. and WELLNER, J. A. (1996). *Weak convergence and empirical processes*. Springer Series in Statistics. Springer-Verlag, New York. With applications to statistics. MR1385671 (97g:60035)

[23] ZHENG, W. and VAN DER LAAN, M. J. (2011). Targeted maximum likelihood estimation of natural direct effect. Tech. Rep. 288, U.C. Berkeley Division of Biostatistics Working Paper Series.