2019

An Unbiased Variance Estimator of a K-sample U-statistic with Application to AUC in Binary Classification

Alexandria Guo
aguo@wellesley.edu

An Unbiased Variance Estimator of a K-sample U-statistic with Application to AUC in Binary Classification

Alexandria Guo
Under the Advisement of
Professor Qing Wang

A Thesis Submitted in Partial Fulfillment
of the Prerequisite for Honors
in the Department of Mathematics,
Wellesley College

May 2019

Abstract

Many questions in research can be rephrased as binary classification tasks, to find simple yes-or-no answers. For example, does a patient have a tumor? Should this email be classified as spam? For classifiers trained to answer these queries, area under the ROC (receiver operating characteristic) curve (AUC) is a popular metric for assessing the performance of a binary classification method, where a larger AUC score indicates an overall better binary classifier. However, due to sampling variation, the model with the largest AUC score for a given data set is not necessarily the optimal model. Thus, it is important to evaluate the variance of AUC. We first recognize that AUC can be estimated unbiasedly in the form of a two-sample U-statistic. We then propose a new method, an unbiased variance estimator of a general K-sample U-statistic, and apply it to evaluate the variance of AUC. We suggest choosing the most parsimonious model whose AUC score is within 1 standard error of the maximum AUC. The developed procedure improves model selection algorithms that weigh complexity and performance.

To realize the proposed unbiased variance estimator of AUC, we propose to use a partition resampling scheme that yields high computational efficiency. We conduct simulation studies to investigate the performance of the developed method in comparison to bootstrap and jackknife variance estimators. The simulations suggest that the proposal yields comparable or even better results in terms of bias and mean squared error. In addition, it has significantly improved computational efficiency compared to its resampling-based counterparts. Moreover, we also discuss the generalization of the devised method to estimating the variance of a general K-sample U-statistic ($\mathrm{K} \geq 2$), which has broad applications in practice.

Acknowledgments

I am indebted to my thesis advisor, Professor Qing Wang, without whom it would not have been possible for me to complete this work. Thank you to Professor Wang, for her endless patience and detailed guidance at every step of my thesis this year. I would like to express my gratitude to my major advisors, Professor Mala Radhakrishnan and Professor Brian Tjaden, for their support of me in finding a research topic and throughout, and Professor Eni Mustafaraj for her continued advice on my thesis and more. Thank you also to Professor Ellen Hildreth and Professor Megan Kerr, and Professor Radhakrishnan and Professor Tjaden again, for their time and energy spent serving on my thesis defense committee.

I am also grateful for my friends who supported me, wrote with me, and inspired me — Rachel Kim, Daniela Kreimerman, Regine Ong, Kavindya Thennakoon, Havannah Tran, and many others. Finally, I want to thank my family for their constant stream of love and encouragement.

Contents

List of Figures

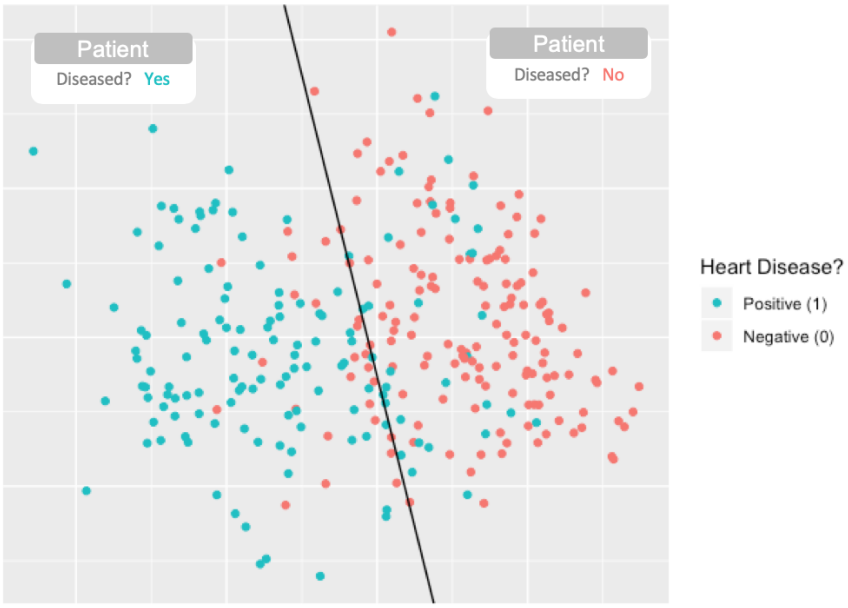
List of Tables

1 Introduction

Classification is one of the pattern recognition problems in statistics, where the larger task of pattern recognition uses algorithms to identify regularities in the data and creates a mapping from a given set of input values to an output space (Bishop, 2006). More specifically, classification maps input values to a class c_i from a set of a finite number of classes $\{c_1, \cdots, c_k\}$. In this case the output variable, say Y, is categorical with k different levels, $c_1, \cdots, c_k$. Binary classification is the specific case where there are only two possible classes, $k = 2$, and each instance is associated with one and only one label.

In practice, the binary labels are encoded as ones and zeros and can each be interpreted as a "yes" or "no," or positive or negative response, to a yes-or-no question. Research questions in many fields, such as biological applications, can be phrased in terms of a binary classification problem. For example, Figure 1 visualizes a binary classifier trained on the well-known and open-source *Heart Disease* data set that we explore more deeply in Chapter 7. In this medical context, each data point is a patient undergoing angiography, who may (colored blue) or may not (colored red) be diagnosed with heart disease. Using the data of observed characteristics for all the patients along these two features visualized along the axes, a binary classifier was trained to try and predict the health of the patients in the data set, and possibly that of other patients (Detrano et al., 1989).

Figure 1: Example of binary classification for the *Heart Disease* data set.



In general, binary classification can be used to answer the question of whether or not an observed data point falls into a certain set or under a particular label. In medical research, we can consider whether or not a patient expresses a certain phenotype or has a particular disease, based on available data such as health records (Hammond et al., 2018; Gupta et al., 2019; Brisimi et al., 2018). Alternatively, on the microscopic level, it can be used to characterize molecules like proteins and whether or not a label such as "intrinsically disordered" is accurate (Fukuchi et al., 2011). Binary classification also has applications in any other field of research asking similarly structured questions such as computer science, as statistical and machine learning models are applied to identify sockpuppets (i.e. fake accounts) or spam email (Kumari & Srivastava, 2017; Carreras & Marquez, 2001).

In recent research, classification has expanded from binary classification into multi-class as well as multi-label problems. In multi-class classification, there are $k > 2$ mutually exclusive possible classes, while in multi-label classification, a data point might be assigned more than one label. However, binary classifications remain an important piece to constructing classifiers for multi-class and multi-label problems, as proposed techniques to solve these questions often involve either layering binary classifiers or converting the data set into multiple binary data sets (Herrera et al., 2016).

The rest of the thesis is organized as follows: in Chapter 2, we summarize several commonly used binary classification methods, which provide solutions to the problem of binary classification introduced above. In Chapter 3, we discuss evaluation metrics of the performance of a binary classifier. In particular, we focus on the popular graphical tool called the ROC (receiver operating characteristic) curve and the area under the ROC curve (AUC). We also show that AUC has a probabilistic interpretation, which leads to an unbiased estimator of AUC in the form of a two-sample U-statistic. In later discussions we consider using AUC as the criterion to select the best binary classifier and choose the model with the highest AUC score. We note that AUC score, a sample statistic, suffers from sampling variation. Therefore, a model with the highest AUC for a given data set may not be optimal when one changes to a different data set. To account for the variability of AUC and potentially select a more parsimonious model, we consider implementing the one-standard-error (1-SE) rule (Hastie, Tibshirani, & Friedman, 2009), which is explained in detail in Chapter 4. In Chapter 5, we propose an unbiased U-statistic variance estimator, and

compare its performance to the existing variance estimators using simulation studies in Chapter 6. In Chapter 7, we demonstrate the use of our proposed unbiased estimator on a real data set, and discuss future work and applications in Chapter 8.

2 Binary Classification

In this thesis, we focus on binary classification to demonstrate our two-sample U-statistic variance estimator for a binary classification performance metric called AUC (area under receiver operating characteristic curve), that can be written in the form of a U-statistic. In the simulations realized in this thesis, we focus on logistic regression, but the metric and our proposed estimator can be applied to all binary classifiers that assign a probability to input data points. In this chapter, we review the formulations of logistic regression in Section 2.2 and probit regression in Section 2.3 (Neter et al., 2005).

2.1 Definitions

Given multivariate data of n observations described along k features, we represent the data set as an n by k matrix $\mathcal{D}$, where $\mathcal{D} = [X_1, \cdots, X_i, \cdots, X_k]$ and each column $X_i = [x_{1i}, \cdots, x_{ji} \cdots, x_{ni}]$. x_{ji} denotes an observed value, while X_i denotes the i^{th} feature or x variable. Y denotes a response variable, which in binary classification takes values from $\{0, 1\}$.

2.2 Logistic Regression

For a simple linear regression, we can write a simple linear regression model as follows:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i \tag{1}$$

where ϵ_i are often assumed to be independent, identically distributed normal random variables with a mean of 0 and constant variance. However, when the response variable of interest is discrete, a number of these assumptions in linear regression are violated. In the context of binary classification, the response variable has only two possible outcomes. In this case, Y_i follows a Bernoulli distribution $Y_i \sim \text{Bern}(\pi_i)$ with probability mass function defined as

$$p(y_i) = \begin{cases} \pi_i & \text{if} \quad y_i = 1 \\ 1 - \pi_i & \text{if} \quad y_i = 0 \end{cases}$$

Under a Bernoulli distribution, it is easy to show that in this case $E(Y_i) = \pi_i$ and $\text{Var}(Y_i) = \pi_i(1 - \pi_i)$. Given Equation (1) and $E(\epsilon_i) = 0$, the expected value of Y_i is:

$$E(Y_i) = \beta_0 + \beta_1 X_i = \pi_i$$

where π_i is the probability of $Y_i = 1$ when the predictor variable is X_i. From the axioms of probability, π_i is restricted to the range of 0 to 1. Thus, we have three major violations of the usual regression assumptions: 1) non-normal error terms, 2) non-constant error variance, and 3) constraints on response function (Neter et al., 2005). A simple linear regression model, like in Equation (1), could not satisfy all the listed constraints.

One common model to analyze binary outcome data is the logistic regression model. Assume the binary response variable Y_i arises from dichotomizing a continuous response Y_i^c. The formulation of logistic regression is motivated by fitting a linear regression model on Y_i^c by assuming that the random errors ϵ_i's follow a logistic distribution. For a logistic random variable ε_L with a mean of 0 and standard deviation of $\frac{\pi}{\sqrt{3}}$, the cumulative density formula is:

$$F_L(\varepsilon_L) = \frac{\exp(\varepsilon_L)}{1 + \exp(\varepsilon_L)}$$

A binary variable Y_i can be created from a continuous variable $Y_i^c = \beta_0^c + \beta_1^c X_i + \epsilon_i^c$ compared to a threshold value of z. We can then rewrite $P(Y_i = 1)$ using the continuous variable, formulating a standardized expression for the continuous response:

$$\begin{aligned} P(Y_i = 1) = \pi_i &= P(Y_i^c \leq z) \\ &= P(\beta_0^c + \beta_1^c X_i + \varepsilon_i^c \leq z) \\ &= P(\frac{\varepsilon_i^c}{\sigma_c} \leq \frac{z - \beta_0^c}{\sigma_c} - \frac{\beta_1^c}{\sigma_c} X_i) \\ &= P(Z \leq \beta_0^* + \beta_1^* X_i) \end{aligned}$$

For a logistic random error with standard deviation σ_c, we can use the above equation to arrive at

the form of a logistic regression.

$$\begin{aligned} P(Y_i = 1) &= P(\frac{\varepsilon_i^c}{\sigma_c} \leq \beta_0^* + \beta_1^* X_i) \\ &= P(\sigma_c \frac{\varepsilon_i^c}{\sigma_c} \leq \sigma_c \beta_0^* + \sigma_c \beta_1^* X_i) \\ &= P(\varepsilon_L \leq \beta_0 + \beta_1 X_i) \\ &= F_L(\beta_0 + \beta_1 X_i) \\ &= \frac{\exp(\beta_0 + \beta_1 X_i)}{1 + \exp(\beta_0 + \beta_1 X_i)} \end{aligned}$$

The final form of the logistic mean response function and its inverse, the logit link transformation, are:

$$E(Y_i) = \pi_i = F_L(\beta_0 + \beta_1 X_i) = \frac{\exp(\beta_0 + \beta_1 X_i)}{1 + \exp(\beta_0 + \beta_1 X_i)} = \frac{1}{1 + \exp(-\beta_0 - \beta_1 X_i)}$$

$$F_L^{-1}(\pi_i) = \pi_i' = \beta_0 + \beta_1 X_i = \ln\left(\frac{\pi_i}{1 - \pi_i}\right)$$

As β_1^* increases, the shape of the curve becomes more S-like, and reversing its sign causes the curve to monotonically decrease rather than increase. Changing β_0^* shifts the curve along the horizontal axis, with the direction of the shift depending on both beta coefficients. The logistic curve also possesses the symmetry property, which means that if all 0's are reversed to 1's and vice versa, and $Y_i' = 1 - Y_i$, the curve would be symmetric across the vertical axis, due to the signs of all coefficients being switched.

In general, a logistic regression model with k predictor variables and a binary response Y_i is defined in the following form:

$$\log\left(\frac{\pi_i}{1 - \pi_i}\right) = \beta_0 + \beta_1 X_{i,1} + \cdots + \beta_k X_{i,k}$$

2.3 Probit Regression

Instead of the logistic mean response function, another sigmoidal response function that can be used is the probit mean response function. Any binary variable Y_i can be reconsidered as a discretization of a continuous variable Y_i^c, rewritten into two inequalities of greater than or equal to some z and less than z. For probit regression, we assume the error associated with the underlying continuous

and linear response model ε_i^c is normally distributed, with a mean of 0 and variance of σ_c^2. Hence:

$$\begin{aligned} P(Y_i = 1) = \pi_i &= P(Y_i^c \leq z) \\ &= P(\beta_0^c + \beta_1^c X_i + \varepsilon_i^c \leq z) \\ &= P(\frac{\varepsilon_i^c}{\sigma_c} \leq \frac{\beta_0^c}{\sigma_c} + \frac{\beta_1^c}{\sigma_c} X_i) \\ &= P(Z \leq \beta_0^* + \beta_1^* X_i) \end{aligned}$$

where Z, β_0^*, and β_1^*, were standardized to follow a normal distribution. If $P(Z \leq w) = \Phi(w)$, we define the probit mean response function and its inverse, the probit link transformation to be:

$$E(Y_i) = \pi_i = \phi(\beta_0^* + \beta_1^* X_i)$$

$$\phi^{-1}(\pi_i) = \pi_i' = \beta_0^* + \beta_1^* X_i$$

The transformation expression can also be called the probit response function or linear predictor. The probit regression shares all the properties previously mentioned for the logistic regression, from the symmetry property to the effect of coefficient changes on the curve.

3 Evaluation of a Binary Classifier

In this chapter, we first define a series of performance metrics in Section 3.1, and focus on defining the receiver operating characteristic curve (ROC) and its area under the curve (AUC) in Section 3.2. In Section 3.3, we demonstrate the probabilistic interpretation of AUC, which allows us in Section 3.4, when reviewing methods of calculating AUC, to highlight the Mann-Whitney U-statistic. The Mann-Whitney test statistic can be equivalently written in a two-sample U-statistic form.

3.1 Performance Metrics

Table 1: General confusion matrix for two classes, positive and negative.

		Actual Label: +	Actual Label: -
Predicted Label	+	True Positives (TP)	False Positives (FP)
	-	False Negatives (FN)	True Negatives (TN)

We can create a 2 by 2 confusion matrix to name the 4 possible combinations of correct and incorrect predictions by a binary classifier, and use these quantities to construct the following metrics.

Definition 3.1. *Accuracy* is the proportion of correctly labeled data overall, and is defined as

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

Accuracy is a popular overall metric, but can be misleading in particular applications and when the classes are imbalanced. For instance, in medical contexts, a specific disease may be rare and occur in 1% of the population. A classifier that contains no information from the data at all and naively diagnoses every patient as healthy, or negative for the disease, will have an accuracy of 99%. However, this classifier would not have been able to identify any of the diseased people, which is

often the critical piece in medical applications to ensure those patients would undergo treatment.

Definition 3.2. *Precision* is the proportion of correctly labeled positive data points out of all the data points labeled as positive. It is defined as

$$\text{Precision} = \frac{TP}{TP + FP}$$

Definition 3.3. Also known as *recall* and the *true positive rate (TPR)*, *sensitivity* is the proportion of correctly labeled positive data points out of all the actual positive data points. It is defined as

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

In the above example in medical diagnostics, the same classifier that scored a 99% for accuracy would have also scored a 0% for precision and sensitivity.

Definition 3.4. Also known as the *true negative rate (TNR)*, *specificity* is the proportion of correctly labeled negative data points out of all the actual negative data points. It is defined as

$$\text{Specificity} = \frac{TN}{TN + FP}$$

All the above metrics take values in $[0, 1]$, and the closer to 1 the value, then generally the better the performance of the binary classifier. Often times, precision and recall are paired together either as a precision-recall curve or a F-measure metric.

Definition 3.5. The *F-measure* or F_1-*Score* is the harmonic average of precision and recall, and is defined as

$$\text{F-measure} = 2 \times \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

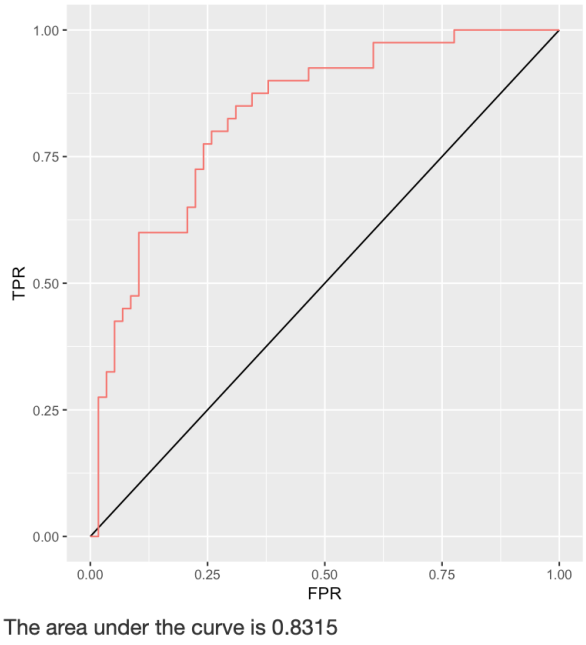
Sensitivity and specificity are also frequently paired together, as displayed in an ROC curve.

3.2 ROC Curve

A receiver operating characteristic (ROC) curve is a plot of the true positive rates against false positive rates (i.e. 1 - specificity) of a binary classifier, across different threshold values. Overall,

this diagram captures the compromise made between sensitivity to detect all positive labels, and specificity to avoid false alarms. The axes of the plot can also be described as sensitivity vs. 1 – specificity. A sample ROC curve is displayed in Figure 2, which was created from the popular, open-source data set *Pima Indians diabetes*[1] (Smith et al., 1988).

Figure 2: Example ROC curve for the *Pima Indians diabetes* data set.



In addition to the curve itself, an ROC plot will often also include the $y = x$ line for reference. Any point along this line reflects a classifier that is performing as well as random guessing. The further left and above the $y = x$ line a curve is, the better the respective classifier is. Performing worse than random chance presents the interesting case of capturing useful information for prediction, but implementing it incorrectly. By flipping the signs of the coefficients of the regression, the curve should be able to flip across the $y = x$ line.

If we consider all n data points, each with a true class of positive or negative, we assign each data point a probability $\hat{p}(y) \in [0, 1]$ of belonging to the positive class, based on a statistical model such as the logistic regression model. We then can rank them from least to greatest in magnitude as $\hat{p}_1, \cdots, \hat{p}_i, \cdots, \hat{p}_n$. For a given binary classifier (e.g. logistic regression, probit regression, etc.), we then choose a discrimination threshold t to assign the i^{th} data point with $\hat{p}_i \leq t$ to the negative

[1]The *Pima Indians diabetes* data set was obtained from the package `mlbench` in `R`

class or $\hat{p}_i > t$ to the positive class. After obtaining the classifier predictions, we can generate a confusion matrix from which we calculate TPR and FPR per t. The domain of possible values for t is $[0, 1]$. On the curve, the point on the bottom left always represents $t = 1$ and the point on the top right represents $t = 0$. Note that although this interval has infinite possible discrimination thresholds, there are, at most, only $n + 1$ unique pairs of TPR and FPR values, assuming there are no ties in the probabilities (of different true classes). As n is finite, the ROC curve is a step function.

3.3 AUC and the Probabilistic Interpretation of AUC

An associated metric with ROC curves is total area under the curve. In addition to the primary geometric definition, AUC also has a probabilistic interpretation. To derive this, we think about binary classification in terms of conditional probabilities. For a binary classifier, each data point y is associated with a predicted probability $\hat{p}(y) \in [0, 1]$ of y being in the positive class, ultimately assigning a label based upon a decision threshold $t \in [0, 1]$, where 0 represents the negative class and 1 represents the positive class. The ROC curve plots TPR vs. FPR, two values which can be rewritten as:

$$T(t) = P[\hat{p}(y^+) > t \mid class(y^+) = 1]$$

$$F(t) = P[\hat{p}(y^- > t \mid class(y^-) = 0]$$

Using the notation denoting that y^+ is from the positive class and that y^- is from the negative class, AUC can be re-expressed as a definite integral across the $x-$axis, where $T(t)$ is a function of

$F(t)$ and F_0 is some specific value for FPR along the $x-$axis:

$$\begin{aligned}
AUC &= \int_0^1 T(F_0)dF_0 \\
&= \int_0^1 P[\hat{p}(y^+) > F^{-1}(F_0) \mid class(y^+) = 1]\ dF_0 \\
&= \int_0^1 P[\hat{p}(y^+) > F^{-1}(F(t)) \mid class(y^+) = 1] \cdot \frac{dF(t)}{dt} dt \\
&= \int_0^1 P[\hat{p}(y^+) > t \mid class(y^+) = 1] \cdot P[\hat{p}(y^-) = t \mid class(y^-) = 0]\ dt \\
&= \int_0^1 P[\hat{p}(y^+) > \hat{p}(y^-)\ \&\ \hat{p}(y^-) = t \mid class(y^+) = 1\ \&\ class(y^-) = 0]\ dt \\
&= P[\hat{p}(y^+) > \hat{p}(y^-) \mid class(y^+) = 1\ \&\ class(y^-) = 0]
\end{aligned}$$

where in the partial derivative above, we use the fact that the cumulative distribution function $1 - F(t) = P[\hat{p}(y^-) \leq t \mid class(y^-) = 0]$ has a derivative with respect to t of $f(t) = P[\hat{p}(y^-) = t \mid class(y^-) = 0]$.

The final line in the above derivation of AUC's probabilistic interpretation demonstrates that AUC is equivalent to the chance that probabilities calculated by the binary classifier will be higher for y^+ than for y^-, given y^+ is from the positive class and y^- is from the negative class.

3.4 Estimations of AUC

AUC is usually numerically estimated, given that ROC curves generally are not expressible as closed-form functions. Three common methods for estimating AUC are:

1. **Geometric estimation**

   Because the ROC curve is a step function, it is often easy to break up the area under the curve into simple shapes, such as rectangles. The trapezoidal method is one of the more accurate and popular methodologies, and is implemented by the `R` package `pROC`. The height of each trapezoid piece is the interval along the $x-$axis (number of points of estimation^{-1}), and the base lengths are the TPR values corresponding to x_i and x_{i+1}.

2. **Mann-Whitney U-Statistic**

   The non-parametric Mann-Whitney U-statistic represents the empirical probability that y^+ is assigned a rank higher than y^-. Thus, due to AUC's probabilistic interpretation, this

statistic returns a result very similar to the geometric estimation of AUC. The `R` package `ROCR` uses the equivalence of the Mann-Whitney test and AUC in its function to calculate the latter, and a more detailed proof of this equivalence is given by Yan et al. (2003). The Mann-Whitney U-statistic is calculated using:

$$U_{MW} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbb{1}\big(\hat{p}(y_i^+), \hat{p}(y_j^-)\big) \tag{2}$$

where

$$\mathbb{1}\big(\hat{p}(y_i^+), \hat{p}(y_j^-)\big) = \begin{cases} 1 & \text{if} \quad \hat{p}(y_i^+) > \hat{p}(y_j^+) \\ 0 & \text{otherwise} \end{cases}$$

and m and n are the number of observations in the positive and negative classes respectively.

3. **Smoothing**

Smoothing is another parametric AUC estimation method, and is given as an option in the fitting of the ROC curve for the `R` package `pROC`. A smooth ROC curve is fit using kernel smoothing, and then integration is used to estimate the AUC (Faraggi & Reiser 2002). There are various methods of curve smoothing, using some density function. However, if semi-parametric binomial smoothing is used, we assume that both populations of $\hat{p}(y^+)$ and $\hat{p}(y^-)$ follow Gaussian distributions $\hat{p}(y^+) \sim N(\bar{y}^+, \sigma^+)$ and $\hat{p}(y^-) \sim N(\bar{y}^-, \sigma^-)$, where $\bar{y}^+$ and $\bar{y}^-$ represent the means of $\hat{p}(y^+)$ and $\hat{p}(y^-)$ respectively. We can then derive the formula:

$$AUC = \Phi\left(\frac{\bar{y}^+ - \bar{y}^-}{\sqrt{\sigma^{+2} + \sigma^{-2}}}\right) = \Phi\left(\frac{a}{\sqrt{1 + b^2}}\right)$$

where $a = \frac{\bar{y}^+ - \bar{y}^-}{\sigma^{-2}}$, $b = \frac{\sigma^{-2}}{\sigma^{+2}}$, and Φ is the cumulative normal distribution function (Promjiraprawat & Wongseree, 2016).

4 1-SE Rule and Its Implementation

In this chapter, we first review a method for model selection for one-standard error in Section 4.1. In order to implement this algorithm, we need an estimate of the variance of a performance metric, which we can obtain by using resampling-based variance estimation techniques such as bootstrap and jackknife, which are explored in Sections 4.2 and 4.3 respectively.

4.1 1-SE rule

The one-standard error rule was first proposed by Hastie, Tibshirani, and Friedman (2009). It aims to overcome the overfitting problem and seek the most parsimonious model whose performance is close to optimum. It is a general algorithm for selecting an optimal model from a set of similar models trained on the same data and cross-validation algorithm. The overall intuition is to pick the most parsimonious model that still scores similarly to the optimal model. In practice, this results in calculating a metric of model performance, such as AUC or cross-validation error, and identifying the optimal score. We then select the smallest model whose score for the metric of interest falls within the range of one standard error from the optimal score.

In other words, we calculate a metric score, say M, for each model from M_1 to M_K, ordered in increasing size, where K is the number of predictors in the largest model size tested. If a higher value in the performance metric is better, given that model j is found to have the highest metric score M_j, we choose the smallest model i such that $M_i \geq M_j - SE$ and $i \leq j$, where SE is the standard error of the metric for the optimal model. Alternatively, if a lower metric score is better, we want to find the smallest model i such that $M_i \leq M_j + SE$.

4.2 Bootstrap Variance Estimation

The bootstrap method was proposed by Efron (1979) as a "more primitive" method than its predecessor, the jackknife method. Assuming the population sampling distribution is similar to the distribution of the data, bootstrap estimates the sampling distribution by randomly selecting from the n observations in the data with replacement to generate bootstrap samples of size n, in order to approximate the sampling distribution of the statistic of interest. There are n^n possible bootstrap samples, which can quickly become impossible to compute exhaustively, and so we choose

to generate and calculate over $B << n^n$ random bootstrap samples.

Let θ be the population parameter of interest. Given a sample of size n, denote the sample estimate as $\hat{\theta}$. To understand the sampling distribution of $\hat{\theta}$ or to estimate the variance of $\hat{\theta}$, one can generate a large number B of bootstrap samples, say $D_1^*, \cdots, D_B^*$, each of size m and obtained from sampling with replacement from the original data set D. Then, one can calculate B bootstrap statistics, say $\hat{\theta}_1^*, \cdots, \hat{\theta}_B^*$, where $\hat{\theta}_b^*$ is the statistic computed based on the b^{th} bootstrap sample D_b^*. Thus, we can estimate the standard deviation of our statistic $\hat{\theta}$ as:

$$SE_{boot}(\hat{\theta}) = \sqrt{\frac{1}{B}\sum_{b=1}^{B}(\hat{\theta}_b^* - \bar{\hat{\theta}}^*)^2}$$

where $\bar{\hat{\theta}}^*$ is the average of all calculated $\hat{\theta}_b^*$. It is recommended to choose a value of B at which the calculated variance begins to converge.

4.3 Jackknife Variance Estimation

The Quenouille-Tukey jackknife method was an early resampling technique first proposed by Quenouille in 1949, and later expanded and given its current name by Tukey in 1958. The jackknife method also attempts to approximate the sampling population distribution with the observed data distribution by resampling from the data, but unlike bootstrap, without replacement. In Efron's proposal for bootstrap, he demonstrated that jackknife is a linear approximation of bootstrap (1979). For the *delete-one* version of the jackknife method, we take some data $D = [X_1, \cdots, X_{i-1}, X_i, X_{i+1}, \cdots, X_n]$ and remove the i^{th} data entry to generate samples $D_{(-1)}^*, \cdots D_{(-n)}^*$, where each $D_i = [X_1, \cdots, X_{i-1}, X_{i+1}, \cdots, X_n]$ is of size $n-1$. In general, the delete-one jackknife method's runtime is shorter than newer and more popular methods such as bootstrap, because it generates n rather than B samples, where $B << n^n$. However, the jackknife method may also take longer to converge, and is also biased.

To estimate the variance of the sample estimate $\hat{\theta}$, one generates n jackknife samples, say $D_{(-1)}^*, \cdots, D_{(-n)}^*$, each of size $n-1$ and obtained from sampling with replacement from the original data set D. Then, one calculates n bootstrap statistics, say $\hat{\theta}_{(-1)}^*, \cdots, \hat{\theta}_{(-n)}^*$, where $\hat{\theta}_{(-i)}^*$ is the statistic computed based on jackknife sample $D_{(-i)}^*$ with the i^{th} observation removed. Thus, we

can estimate the standard deviation of our statistic $\hat{\theta}$ as:

$$SE_{jack-1}(\hat{\theta}) = \sqrt{\frac{n-1}{n}\sum_{i=1}^{n}(\hat{\theta}^*_{(-i)} - \hat{\theta}^*_{(.)})^2}$$

where $\hat{\theta}^*_{(.)}$ is the average of all n calculated $\hat{\theta}^*_{(-i)}$ (Efron & Stein, 1981).

This formula can be generalized into the *delete-d* version of the jackknife method, which takes data D and removes any d entries to generate samples $D^*_1, \cdots, D^*_z$, where $z = \binom{n}{d}$ and each D^* of size $n-d$. Delete-d is preferable to delete-one jackknife in cases where the θ is not a sufficiently smooth function, such as sample quantiles. Both methods are consistent and asymptotically unbiased for sufficiently smooth functions, but when this condition is not true, delete-one becomes inconsistent while delete-d can maintain both consistency and asymptotic unbiasedness depending on d, where d goes to infinity based on a smoothness function. It is generally recommended to select d such that $\sqrt{n} < d < n$ (Shao & Wu, 1989). The variance estimation of $\hat{\theta}$ is similar to that of the leave-one out jackknife method, but has a different normalization constant:

$$SE_{jack-d}(\hat{\theta}) = \sqrt{\frac{n-d}{dz}\sum_{j=1}^{z}(\hat{\theta}^*_{(j)} - \hat{\theta}^*_{(.)})^2}$$

delete-d jackknife, however, does have the problem of $z = \binom{n}{d}$ also growing very large as n increases. In such cases, we sample over z to generate B data sets of size $n-d$, and estimate the variance of $\hat{\theta}$ as:

$$SE_{jack-d}(\hat{\theta}) = \sqrt{\frac{n-d}{dB}\sum_{j=1}^{B}(\hat{\theta}^*_{(j)} - \hat{\theta}^*_{(.)})^2}$$

5 Our Proposal: An Unbiased Variance Estimator

AUC can be written as a two-sample U-statistic, as shown in Equation (2). This allows us to extend the previous work in Wang and Lindsay (2014), who devised an unbiased variance estimator of a general one-sample U-statistic, to K samples so that one can estimate the variance of AUC (i.e. a two-sample U-statistic) unbiasedly. In this chapter, we review basic concepts and properties of a one-sample U-statistic (Lee, 1990) in Sections 5.1 and 5.2, as well as the unbiased variance estimator proposed in Wang and Lindsay (2014) in Section 5.3. Then, we formulate the two- and K-sample U-statistic variance estimator in Sections 5.4 and 5.5, which are the main contributions of this thesis.

5.1 One-sample U-statistic

Proposed by Halmos and Hoeffding (1948), *unbiased statistics* or U-statistics are a subset of the group of all unbiased estimators, restricted to those with the minimum possible variance. An statistic is unbiased if its expected value, or mean of its sampling distribution, is equal to the target parameter. A foundational theorem states that a functional θ, defined for a distribution function F of random variables $X_1, \cdots, X_n$, admits an unbiased estimator if and only if it can be written in the form of a function ϕ of k variables such that:

$$\theta(F) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(x_1, \cdots, x_k) dF(x_1) \cdots dF(x_k) \tag{3}$$

Here, we call θ a *regular statistical functional* of *degree* k, and ϕ as the *kernel* of the functional. We use the property of unbiasedness to conclude that:

$$\theta(F) = E[\phi(X_1, \cdots, X_k)]$$

Note that ϕ is a symmetric function, which means given a set of input parameters, any permutation of those k components will yield the same output. However, in the case that ϕ is not symmetric,

we can average over all the possible permutations of the inputs to obtain a symmetric function ϕ^*:

$$\phi^*(X_1, \cdots, X_k) = \frac{1}{k!} \sum_{(n,k)} \phi(X_{i_1}, \cdots, X_{i_k})$$

where (n, k) refers to the set of all possible permutations of a subset of size k from x. This is all permutations of $\{i_1, \cdots, i_k\}$, where each i was chosen from $\{1, \cdots, n\}$ of x.

However, note that Equation (3) only uses k sampling parameters instead of all n samples from F. Given that $x_1, \cdots, x_n$ are independent and identically distributed random variables, we intuitively would want to use all available information to best estimate a parameter. In order to do this, we want to consider all the possible k subsets we could choose from n, and average over the calculated values. Thus, given $n \geq k$:

$$U_n = \frac{1}{\binom{n}{k}} \sum_{(n,k)} \phi(X_{i_1}, \cdots, X_{i_k}) = \mathbb{N}^{-1} \sum_{i=1}^{\mathbb{N}} \phi(S_i) \tag{4}$$

where $\mathbb{N}$ is the number of samples of size k denoted S_i, taken from $X_1, \cdots, X_n$. The above is the definition for a one-sample U-statistic. It is simple to show that U_n is an unbiased estimator, i.e. $E(U_n) = \theta$. Examples of common statistics with one-sample U-statistics forms, such as sample mean and sample variance, follow.

Example: Sample mean

When constructing the U-statistic for sample mean, we define the kernel of the form defined in Equation (1), which is the general continuous definition for mean:

$$\theta(F) = \int_{-\infty}^{\infty} \} f(x_1) dx_1$$

This functional is of degree 1. Using Equation (4), the U-statistic is the sample mean $\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$.

Example: Sample variance

The kernel for the U-statistic sample variance is of degree 2, and is defined as:

$$\theta(F) = \frac{1}{2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x_1 - x_2)^2 df(x_1) df(x_2)$$

The U-statistic for calculating sample mean is $s_n^2 = \frac{1}{2\binom{n}{2}}\sum_{1\leq i<j\leq n}(X_i - X_j)^2$.

5.2 Asymptotic Normality for U-statistics

In general, U-statistics are often written in terms of dependent variables and not suitable for the application of the central limit theorem (CLT) or law of large numbers (LLN). Instead, we create a *projection*, denoted by $\hat{U}_n$, that is often the sum of i.i.d. random variables to approximate the U-statistic of interest asymptotically.

5.2.1 One-sample U-statistics

For a one-sample U-statistic of degree k:

$$U_n = \frac{1}{\binom{n}{k}}\sum_{(n,k)}\phi(X_{i_1}, \cdots, X_{i_k})$$

the projection is $\hat{U}_n$:

$$\hat{U}_n = \sum_{i=1}^{n} E(U_n|X_i) - (n-1)\theta$$

where $\theta = E[\phi(X_{i_1}, \cdots, X_{i_k})]$ is the parameter of interest and $\hat{U}_n$ is an i.i.d. sum because of the first term $E(U_n|X_i)$, which is a function of θ, and the second term that acts as a normalizing factor. Thus, we can interpret $\hat{U}_n$ as projecting U_n onto each of the n i.i.d. drawn samples of X. It can be proven using standard asymptotic methods that U_n and $\hat{U}_n$ have the same asymptotic distribution, but the proof (applying the Central Limit Theorem, Slutsky Theorem, etc.) is not included here.

5.2.2 Two-sample and K-sample U-statistics

We can generalize to the two-sample case by considering two independent sets of i.i.d. random variables of sizes n_1 and n_2. Consider a kernel function ϕ of degree $k = k_1 + k_2$, where k_1 of its

components come from the first set of random variables and k_2 components come from the other set. For a two-sample U-statistic:

$$U_n = [\frac{1}{\binom{n_1}{k_1}\binom{n_2}{k_2}}] \sum_{l=1}^{2} \sum_{(n_l,k_l)} \phi(X_{1,i_1}, \cdots, X_{1,i_{k_1}}; X_{2,i_1}, \cdots, X_{2,i_{k_2}}) \tag{5}$$

their associated projection $\hat{U}_n$ is:

$$\hat{U}_n = \sum_{l=1}^{2} \sum_{i=1}^{n_l} E(U_n|X_{li}) - [(n_1 + n_2) - 1]\theta$$

where $\theta = E[\phi(X_{1,i_1}, \cdots, X_{1,i_{k_1}}; X_{2,i_1}, \cdots, X_{2,i_{k_2}})]$.

Analogously, a K-sample U-statistic ($K \geq 2$) is defined by

$$U_n = [\prod_{l=1}^{K} \binom{n_l}{k_l}]^{-1} \sum_{l=1}^{K} \sum_{(n_l,k_l)} \phi(X_{1,i_1}, \ldots, X_{1,i_{k_1}}; \ldots; X_{K,i_1}, \ldots, X_{K,i_{k_K}})) \tag{6}$$

where the degree of $k = k_1 + \cdots + k_K$. Its associated projection $\hat{U}_n$ is:

$$\hat{U}_n = \sum_{l=1}^{K} \sum_{i=1}^{n_l} E(U_n|X_{li}) - [\sum_{l=1}^{K} n_l - 1]\theta$$

where $\theta = E[\phi(X_{1,i_1}, \ldots, X_{1,i_{k_1}}; \ldots; X_{K,i_1}, \ldots, X_{K,i_{k_K}})]$.

Remark 1. Recall that one commonly used estimator for AUC is of a two-sample U-statistic form with degree $k = 2$, where $k_1 = k_2 = 1$, as discussed in Section 3.4. In practice, the AUC estimate is data dependent, and so is subject to sampling variation. In later discussions, we focus on variance estimation of a general K-sample U-statistic, which can be applied to assessing the variability of AUC in binary classification.

5.3 One-sample U-statistic Variance Estimator

In Wang and Lindsay (2014), an unbiased variance estimator $\hat{V}_u$ for a one-sample U-statistic is proposed, provided $k \leq \frac{n}{2}$. Recall the complete one-sample U-statistic, as defined in Equation (4).

Definition 5.1. The unbiased variance estimator of a general one-sample U-statistic, denoted by

$\hat{V}_u$, is defined for a kernel function ϕ of degree k in the following form:

$$\hat{V}_u = Q(k) - Q(0) \tag{7}$$

where $Q(c)$ is defined as:

$$Q(c) = N_c^{-1} \sum_{P_c} \phi(S_i)\phi(S_j)$$

Denote the overlaps of two samples of size k as $O(S_i, S_j)$, from which Wang and Lindsay define $P_c = \{(S_i, S_j)|O(S_i, S_j) \leq c\}$, where $0 \leq c \leq k$, to consider all possible pairs of samples that have c or fewer elements in common. Let N_c be the number of pairs in P_c. It is easy to see that $Q(k) = U_n^2$, so $\hat{V}_u$ can be equivalently written as $\hat{V}_u = U_n^2 - Q(0)$.

Remark 2. One weakness of the unbiased variance estimator (7) is that $\hat{V}_u = Q(k) - Q(0)$ occasionally yields negative values, which do not make sense for variance. In Wang and Lindsay, a fix-up is as follows:

Definition 5.2. A strictly positive variance estimator $\hat{V}_u^+$ is:

$$\hat{V}_u^+ = \max\{\hat{V}_u, S_U^2\}$$

where

$$S_U^2 = \frac{1}{\mathbb{N}(\mathbb{N}-1)} \sum_{i=1}^{\mathbb{N}} \{\phi(S_i) - U_n\}^2, \quad \mathbb{N} = \binom{n}{k}$$

Remark 3. In practice, the calculation of $Q(k)$ and $Q(0)$ each can be computationally expensive, especially for large sample size n and kernel size k. To overcome this drawback, Wang and Lindsay (2014) proposed an equivalent expression of $\hat{V}_u$ based on partition resampling.

Definition 5.3. The complete variance estimator based on partitions $\hat{V}_{partition}$ is:

$$\hat{V}_{partition} = \frac{1}{\mathbb{B}} \sum_{a=1}^{\mathbb{B}} \Big[\frac{1}{m} \sum_{j=1}^{m} \big\{ \frac{1}{m-1} (\phi(S_{a,j}) - \bar{\phi}_a)^2 - (\bar{\phi}_a - \bar{\phi})^2 \big\} \Big]$$

where

$$\bar{\phi}_a = \sum_{j=1}^{m} \frac{\phi(S_{a,j})}{m} \quad \text{and} \quad \bar{\phi} = \sum_{a=1}^{\mathbb{B}} \frac{\bar{\phi}_a}{\mathbb{B}} = U_n$$

To realize the partition resampling, the data is partitioned into a maximal number of subsamples of size k, say $S_1, \cdots, S_m$, where $mk \leq n$. We let $\mathbb{B}$ be the number of different ways one can partition the data set, and for $a = 1, \cdots, \mathbb{B}$, let the a^{th} partition be a unique sequence of non-overlapping size-k samples $S_{a,1}, \cdots, S_{a,m}$.

5.4 Two-sample U-statistic variance estimator

In this thesis, we aim to generalize the proposal in Wang and Lindsay (2014) to the K-sample scenario. The extension of the above estimator to two samples does not involve a manipulation of the above equations, but a redefinition of how the subsamples are selected with two independent samples of sizes n_1 and n_2, where $n = n_1 + n_2$. Given a two-sample U-statistic defined in Equation (5), the unbiased variance estimator of a general two-sample U-statistic, denoted by $\hat{V}_u$, is defined for a kernel function $\phi(x_{i_1}, \cdots, x_{i_{k_1}}; y_{j_1}, \cdots, y_{j_{k_2}})$ with k_1 observations from sample 1 ($k_1 < n_1$) and k_2 observations from sample 2 ($k_2 < n_2$), in the following form:

$$\hat{V}_u = Q(k) - Q(0) \tag{8}$$

where $Q(c)$ is defined as:

$$Q(c) = N_c^{-1} \sum_{P_c} \phi(x_{i_1}, \cdots, x_{i_{k_1}}; y_{j_1}, \cdots, y_{j_{k_2}}) \phi(x_{s_1}, \cdots, x_{s_{k_1}}; y_{t_1}, \cdots, y_{t_{k_2}})$$

Compared to Equation (7), both S_i and S_j are now size $k = k_1 + k_2$ drawing k_1 from the first sample and k_2 from the second sample. The N_c of the new, larger P_c is now calculated considering these two samples; for example, for $c = 0$ in $Q(0)$, the two samples may not overlap at all. Thus $N_0 = \binom{n_1}{k_1}\binom{n_1-k_1}{k_1}\binom{n_2}{k_2}\binom{n_2-k_2}{k_2}$. For $c = k$ in $Q(k)$, the two samples have no restrictions, as the number of overlapping elements can be the total number of set elements k or less, and $N_k = \binom{n_1}{k_1}^2\binom{n_2}{k_2}^2$. Thus, for $\hat{V}_u$ of the two-sample U-statistic, the equations in Definition 5.1 still apply but with the new definitions for the above variables.

Remark 4. For the two-sample U-statistic, the strictly positive variance estimator is still $\hat{V}_u^+ =$

$\max\{\hat{V}_u, S_U^2\}$. S_U^2 is slightly modified from Definition 5.2 to be:

$$S_U^2 = \frac{1}{\mathbb{N}(\mathbb{N}-1)} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \{\phi(S_{1i}; S_{2j}) - U_n\}^2, \quad \mathbb{N} = n_1 n_2$$

Remark 5. Consider a two-sample U-statistic of degree $k = k_1 + k_2$, and two independent samples of size n_1 and n_2 respectively. Let $m_1 = n_1/k_1$ and $m_2 = n_2/k_2$ (here we assume m_1 is divisible by k_1, and m_2 is divisible by k_2). Without loss of generality, assume $m_1 = m_2$, so we denote it as m. Otherwise, take $m = \min\{m_1, m_2\}$. Given a two-sample data set of size n, one could partition it into m blocks of data subsets, each of size $k = k_1 + k_2$, denoted by $S_1, \ldots, S_m$. Then, one could compute the corresponding incomplete U-statistic

$$U_n^{\text{inc}} = \frac{1}{m} \sum_{i=1}^{m} \phi(S_i)$$

which approximates the exact two-sample U-statistic U_n. We use it as a building block to construct a partition-resampling-based realization of the unbiased variance estimator for a two-sample U-statistic.

Let $\mathbb{B}$ be the total number of partitions of the size n data set such that the data set is divided into m blocks of subsets of size k, of which k_1 observations are from sample 1 and k_2 observations are from sample 2. When $m_1 = m_2$, $\mathbb{B} = \binom{n_1}{k_1}\binom{n_1-k_1}{k_1} \cdots \binom{k_1}{k_1}\binom{n_2}{k_2}\binom{n_2-k_2}{k_2} \cdots \binom{k_2}{k_2}$. Given partition a ($1 \le a \le \mathbb{B}$), denote the m data subsets of size k as $S_{a,1}, \ldots, S_{a,m}$. The kernel function ϕ takes values of $\phi(S_{a,1}), \ldots, \phi(S_{a,m})$. We define the partition variance estimator as

$$\hat{V}_{\text{partition}} = \frac{1}{\mathbb{B}} \sum_{\text{all partitions}} \left\{ \frac{1}{m(m-1)} \sum_{i=1}^{m} (\phi(S_{a,i}) - \bar{\phi}_a)^2 - (\bar{\phi}_a - \bar{\phi})^2 \right\} \tag{9}$$

In the case of AUC estimation, the two-sample U-statistic estimator is of degree $k = 2$ ($k_1 = k_2 = 1$), and the kernel function $\phi(\hat{p}_1, \hat{p}_2) = \mathbb{I}\{\hat{p}_1 > \hat{p}_2\}$ is an indicator function. Thus, the partition variance estimator in Equation (9) can be defined accordingly. It can be shown that with the exhaustive number of partitions $\mathbb{B}$, the $\hat{V}_{\text{partition}}$ is equivalent to the $\hat{V}_u$, the proposed unbiased variance estimator.

In practice, the total number of partitions $\mathbb{B}$ is enormous. Thus, there is no computational

advantage of calculating the partition variance estimator as defined in Equation (9). However, one could draw B ($B << \mathbb{B}$) random partitions with replacement from the exhaustive set, and approximate the unbiased variance estimator efficiently.

$$\hat{V}_{\text{partition},B} = \frac{1}{B} \sum_{B \text{ random partitions}} \left\{ \frac{1}{m(m-1)} \sum_{i=1}^{m} (\phi(S_{a,i}) - \bar{\phi}_a)^2 - (\bar{\phi}_a - \bar{\phi})^2 \right\}$$

In our simulation studies we notice that when $n = 100$ or 500 with $B = 10^2$ or 10^3, the partition-resampling realization of the proposed variance estimator yields comparable performance to bootstrap, and it is at least 20 times faster to compute than bootstrap and jackknife variance estimators.

5.5 K-sample U-statistic Variance Estimator

In the most general case, the K-sample U-statistic is provided in Equation (6). The unbiased variance estimator of a general K-sample U-statistic, denoted by $\hat{V}_u$, is defined for a kernel function $\phi(X_{1,s_1}, \ldots, X_{1,s_{k_1}}; \ldots; X_{K,s_1}, \ldots, X_{K,s_{k_K}})$ with k_j observations from the j_{th} sample, in the following form:

$$\hat{V}_u = Q(k) - Q(0) \tag{10}$$

where $Q(c)$ is defined as:

$$Q(c) = N_c^{-1} \sum_{P_c} \phi(X_{1,s_1}, \ldots, X_{1,s_{k_1}}; \ldots; X_{K,s_1}, \ldots, X_{K,s_{k_K}}) \phi(X_{1,t_1}, \ldots, X_{1,t_{k_1}}; \ldots; X_{K,t_1}, \ldots, X_{K,t_{k_K}})$$

As for the two-sample variance estimator, we must redefine S_i, S_j, N_c, P_c, and $Q(c)$ for the K-sample case. Any S_i are of size $k = \sum_{j=1}^{K} k_j$ drawing k_j from the j^{th} sample. The N_c of P_c is now much larger, such that $N_0 = \prod_{j=1}^{K} \binom{n_j}{k_j}\binom{n_j - k_j}{k_j}$ and $N_k = \prod_{j=1}^{K} \binom{n_j}{k_j}^2$. $\hat{V}_u$ can then be formulated in the K-sample case based on Definition 5.1.

6 Simulation Studies

In this chapter, we present simulation studies that evaluate the performance of the proposed variance estimator in comparison to bootstrap and jackknife variance estimators.

To simulate binary-outcome data in the context of binary classification, we started with generating a continuous outcome for Y^c through a linear relationship with a set of x-variables. Then, we converted the continuous response Y^c to binary outcome Y via dichotomization. More specifically, we simulated $R = 1000$ independent data sets, each of size n ($n = 100, 500$). The continuous response variable was obtained based on the following multiple linear regression model:

$$Y_i^c = 1 + X_{i,1} + X_{i,2} + X_{i,3} + X_{i,4} + 0.1X_{i,5} + 0X_6 + \epsilon_i$$

where $\epsilon_i \sim \text{logistic}(\text{location} = 0, \text{scale} = 5)$. Thus, the true model was only composed of 5 of the 6 total available predictors. Each set of n x-variables was independently simulated from the uniform distribution to obtain values between 0 and 1, inclusive. The binary response Y in each data set was determined by comparison of Y^c to a threshold fixed across simulations of the same sample size n, calculated to yield approximately a 50-50, 60-40, 70-30, or 80-20 split between the negative and positive classes. For each simulated data set of a given size, we fitted the following six logistic regression models, each of a different number of predictors, as shown in Table 2. Then, we computed the U-statistic estimate of AUC based on the fitted logistic regression model, and estimated the variance of the AUC score by different methods. In total, we implemented four variance estimators for AUC, including the bootstrap (B=10^3), the jackknife (B=10^3), and our proposed variance estimator based on partition resampling with B=10^2 or B=10^3. The bootstrap and jackknife variance estimators were realized based on 10^3 bootstrap or jackknife samples for each given data set. The specific form of jackknife used was delete-d ($d = 10, 25$), as AUC comes from the ROC curve step function and thus does not satisfy the delete-one jackknife requirement of smoothness. Additionally, the proposed estimator implemented in this simulation used the positive fix-up as well as the partition resampling scheme — note that this is not the *exact* unbiased variance estimator, as the total number of partitions is much larger than $B = 10^2$ or $B = 10^3$.

The true variance of AUC was simulated based on 10^6 AUC scores obtained by 10^6 independently

Table 2: Models under comparison in the simulation studies.

p	X_1	X_2	X_3	X_4	X_5	X_6
1	✓					
2	✓	✓				
3	✓	✓	✓			
4	✓	✓	✓	✓		
5	✓	✓	✓	✓	✓	
6	✓	✓	✓	✓	✓	✓

generated data sets. The simulations were realized using the statistical language `R`, and the code was parallelized over 4 cores using Professor Wang's computational machine "poweredge," composed of Intel Xeon E5-2680 v3 processors, in order to speed up the computational process.

6.1 Results

Table 3: Summary statistics of estimated variance of AUC at a 50-50 split.

		n = 100			n = 500		
		Mean	StDev	MSE	Mean	StDev	MSE
k = 1	Truth	2.98e-03	NA	NA	6.59e-04	NA	NA
	Bootstrap	3.26e-03	8.92e-04	**8.73e-07**	7.12e-04	1.84e-04	3.67e-08
	Jackknife	4.70e-03	5.08e-03	2.88e-05	7.58e-04	9.31e-04	8.76e-07
	Unbiased (B=10^2)	3.09e-03	1.50e-03	2.26e-06	6.50e-04	1.89e-04	3.58e-08
	Unbiased (B=10^3)	3.08e-03	1.46e-03	2.15e-06	6.50e-04	1.73e-04	**3.00e-08**
k = 2	Truth	2.79e-03	NA	NA	6.03e-04	NA	NA
	Bootstrap	3.24e-03	5.30e-04	**4.82e-07**	6.69e-04	9.44e-05	1.33e-08
	Jackknife	4.41e-03	3.89e-03	1.78e-05	7.27e-04	8.19e-04	6.87e-07
	Unbiased (B=10^2)	3.03e-03	1.63e-03	2.72e-06	6.43e-04	7.75e-05	7.64e-09
	Unbiased (B=10^3)	3.02e-03	1.63e-03	2.71e-06	6.45e-04	5.62e-05	**4.91e-09**
k = 3	Truth	2.59e-03	NA	NA	5.82e-04	NA	NA
	Bootstrap	3.02e-03	4.13e-04	**3.53e-07**	6.13e-04	5.10e-05	3.59e-09
	Jackknife	3.95e-03	2.46e-03	7.94e-06	6.54e-04	1.54e-04	2.88e-08
	Unbiased (B=10^2)	3.02e-03	6.11e-04	5.58e-07	6.32e-04	6.16e-05	6.31e-09
	Unbiased (B=10^3)	3.02e-03	5.59e-04	4.98e-07	6.33e-04	3.14e-05	**3.57e-09**
k = 4	Truth	2.45e-03	NA	NA	5.71e-04	NA	NA
	Bootstrap	2.78e-03	3.57e-04	**2.34e-07**	5.88e-04	3.89e-05	**1.82e-09**
	Jackknife	3.61e-03	1.59e-03	3.87e-06	6.38e-04	4.54e-05	6.63e-09
	Unbiased (B=10^2)	2.94e-03	4.19e-04	4.18e-07	6.21e-04	5.94e-05	6.05e-09
	Unbiased (B=10^3)	2.95e-03	3.21e-04	3.50e-07	6.19e-04	3.24e-05	3.45e-09
k = 5	Truth	2.32e-03	NA	NA	5.60e-04	NA	NA
	Bootstrap	2.61e-03	2.98e-04	**1.75e-07**	5.79e-04	3.70e-05	**1.74e-09**
	Jackknife	3.55e-03	1.49e-03	3.73e-06	6.45e-04	1.53e-04	3.06e-08
	Unbiased (B=10^2)	2.87e-03	7.45e-04	8.62e-07	6.15e-04	5.92e-05	6.59e-09
	Unbiased (B=10^3)	2.87e-03	6.93e-04	7.84e-07	6.18e-04	3.27e-05	4.41e-09
k = 6	Truth	2.22e-03	NA	NA	5.50e-04	NA	NA
	Bootstrap	2.48e-03	2.61e-04	**1.32e-07**	5.70e-04	3.44e-05	**1.60e-09**
	Jackknife	3.45e-03	9.39e-04	2.39e-06	6.46e-04	7.37e-05	1.46e-08
	Unbiased (B=10^2)	2.84e-03	4.31e-04	5.66e-07	6.12e-04	5.97e-05	7.37e-09
	Unbiased (B=10^3)	2.84e-03	3.35e-04	4.86e-07	6.15e-04	3.20e-05	5.33e-09

Table 4: Summary statistics of estimated variance of AUC at a 60-40 split.

		n = 100			n = 500		
		Mean	StDev	MSE	Mean	StDev	MSE
k = 1	Truth	3.07e-03	NA	NA	6.74e-04	NA	NA
	Bootstrap	3.33e-03	9.27e-04	**9.24e-07**	7.25e-04	1.89e-04	3.84e-08
	Jackknife	4.76e-03	4.99e-03	2.77e-05	8.25e-04	1.40e-03	1.98e-06
	Unbiased (B=10^2)	3.19e-03	1.27e-03	1.63e-06	6.67e-04	1.73e-04	3.00e-08
	Unbiased (B=10^3)	3.20e-03	1.20e-03	1.46e-06	6.67e-04	1.47e-04	**2.16e-08**
k = 2	Truth	2.87e-03	NA	NA	6.21e-04	NA	NA
	Bootstrap	3.33e-03	5.32e-04	**4.93e-07**	6.90e-04	1.00e-04	1.47e-08
	Jackknife	4.52e-03	3.53e-03	1.52e-05	7.54e-04	7.47e-04	5.76e-07
	Unbiased (B=10^2)	3.19e-03	1.15e-03	1.42e-06	6.67e-04	7.79e-05	8.15e-09
	Unbiased (B=10^3)	3.18e-03	1.14e-03	1.39e-06	6.65e-04	3.63e-05	**3.22e-09**
k = 3	Truth	2.67e-03	NA	NA	5.99e-04	NA	NA
	Bootstrap	3.09e-03	4.48e-04	**3.74e-07**	6.30e-04	5.46e-05	**3.96e-09**
	Jackknife	4.16e-03	3.22e-03	1.26e-05	6.66e-04	6.32e-05	8.46e-09
	Unbiased (B=10^2)	3.09e-03	1.15e-03	1.49e-06	6.51e-04	7.70e-05	8.68e-09
	Unbiased (B=10^3)	3.09e-03	1.12e-03	1.42e-06	6.50e-04	3.85e-05	4.10e-09
k = 4	Truth	2.53e-03	NA	NA	5.86e-04	NA	NA
	Bootstrap	2.85e-03	3.93e-04	**2.60e-07**	6.03e-04	3.95e-05	**1.85e-09**
	Jackknife	3.81e-03	1.94e-03	5.44e-06	6.55e-04	4.73e-05	7.08e-09
	Unbiased (B=10^2)	3.02e-03	6.54e-04	6.74e-07	6.37e-04	7.69e-05	8.54e-09
	Unbiased (B=10^3)	3.02e-03	5.55e-04	5.51e-07	6.37e-04	3.88e-05	4.15e-09
k = 5	Truth	2.39e-03	NA	NA	5.74e-04	NA	NA
	Bootstrap	2.67e-03	3.30e-04	**1.86e-07**	5.93e-04	3.76e-05	**1.74e-09**
	Jackknife	3.67e-03	1.45e-03	3.76e-06	6.59e-04	4.86e-05	9.45e-09
	Unbiased (B=10^2)	2.97e-03	5.24e-04	6.09e-07	6.36e-04	7.64e-05	9.58e-09
	Unbiased (B=10^3)	2.97e-03	4.02e-04	4.97e-07	6.35e-04	3.83e-05	5.08e-09
k = 6	Truth	2.29e-03	NA	NA	5.64e-04	NA	NA
	Bootstrap	2.54e-03	2.95e-04	**1.48e-07**	5.82e-04	3.60e-05	**1.62e-09**
	Jackknife	3.60e-03	1.23e-03	3.21e-06	6.62e-04	5.00e-05	1.20e-08
	Unbiased (B=10^2)	2.93e-03	5.09e-04	6.69e-07	6.36e-04	7.58e-05	1.10e-08
	Unbiased (B=10^3)	2.91e-03	4.08e-04	5.47e-07	6.33e-04	3.79e-05	6.19e-09

Table 5: Summary statistics of estimated variance of AUC at a 70-30 split.

		n = 100			n = 500		
		Mean	StDev	MSE	Mean	StDev	MSE
k = 1	Truth	3.46e-03	NA	NA	7.69e-04	NA	NA
	Bootstrap	3.72e-03	1.05e-03	**1.17e-06**	8.20e-04	2.10e-04	**4.68e-08**
	Jackknife	5.65e-03	6.92e-03	5.26e-05	1.01e-03	2.20e-03	4.88e-06
	Unbiased (B=10^2)	3.60e-03	2.28e-03	5.23e-06	7.39e-04	4.36e-04	1.91e-07
	Unbiased (B=10^3)	3.60e-03	2.26e-03	5.14e-06	7.41e-04	4.22e-04	1.79e-07
k = 2	Truth	3.26e-03	NA	NA	7.12e-04	NA	NA
	Bootstrap	3.79e-03	6.95e-04	**7.63e-07**	7.88e-04	1.23e-04	**2.09e-08**
	Jackknife	5.60e-03	5.32e-03	3.38e-05	9.07e-04	1.24e-03	1.57e-06
	Unbiased (B=10^2)	3.58e-03	1.91e-03	3.74e-06	7.45e-04	3.22e-04	1.05e-07
	Unbiased (B=10^3)	3.58e-03	1.67e-03	2.88e-06	7.48e-04	2.83e-04	8.14e-08
k = 3	Truth	3.03e-03	NA	NA	6.82e-04	NA	NA
	Bootstrap	3.48e-03	6.02e-04	**5.63e-07**	7.21e-04	7.39e-05	**6.96e-09**
	Jackknife	5.01e-03	4.29e-03	2.24e-05	7.76e-04	2.70e-04	8.18e-08
	Unbiased (B=10^2)	3.49e-03	1.90e-03	3.81e-06	7.46e-04	1.19e-04	1.84e-08
	Unbiased (B=10^3)	3.49e-03	1.78e-03	3.37e-06	7.45e-04	6.15e-05	7.82e-09
k = 4	Truth	2.85e-03	NA	NA	6.64e-04	NA	NA
	Bootstrap	3.19e-03	5.46e-04	**4.12e-07**	6.85e-04	5.67e-05	**3.65e-09**
	Jackknife	4.57e-03	3.01e-03	1.20e-05	7.55e-04	7.75e-05	1.42e-08
	Unbiased (B=10^2)	3.42e-03	1.51e-03	2.59e-06	7.30e-04	1.25e-04	1.99e-08
	Unbiased (B=10^3)	3.41e-03	1.41e-03	2.29e-06	7.29e-04	6.15e-05	7.96e-09
k = 5	Truth	2.69e-03	NA	NA	6.50e-04	NA	NA
	Bootstrap	2.98e-03	4.74e-04	**3.06e-07**	6.70e-04	5.35e-05	**3.28e-09**
	Jackknife	4.26e-03	1.83e-03	5.81e-06	7.58e-04	7.11e-05	1.67e-08
	Unbiased (B=10^2)	3.41e-03	8.48e-04	1.23e-06	7.24e-04	1.24e-04	2.08e-08
	Unbiased (B=10^3)	3.39e-03	6.59e-04	9.23e-07	7.26e-04	6.22e-05	9.64e-09
k = 6	Truth	2.58e-03	NA	NA	6.36e-04	NA	NA
	Bootstrap	2.81e-03	4.36e-04	**2.42e-07**	6.57e-04	4.98e-05	**2.92e-09**
	Jackknife	4.15e-03	1.72e-03	5.42e-06	7.62e-04	8.56e-05	2.32e-08
	Unbiased (B=10^2)	3.23e-03	1.69e-03	3.27e-06	7.19e-04	1.23e-04	2.20e-08
	Unbiased (B=10^3)	3.23e-03	1.57e-03	2.88e-06	7.23e-04	6.19e-05	1.14e-08

Table 6: Summary statistics of estimated variance of AUC at a 80-20 split.

		n = 100			n = 500		
		Mean	StDev	MSE	Mean	StDev	MSE
k = 1	Truth	4.39e-03	NA	NA	1.01e-03	NA	NA
	Bootstrap	4.76e-03	1.48e-03	**2.33e-06**	1.06e-03	3.07e-04	**9.71e-08**
	Jackknife	7.40e-03	7.89e-03	7.12e-05	1.35e-03	1.98e-03	4.04e-06
	Unbiased (B=10^2)	4.92e-03	2.95e-03	8.98e-06	1.00e-03	3.87e-04	1.50e-07
	Unbiased (B=10^3)	4.90e-03	2.60e-03	7.01e-06	9.93e-04	3.17e-04	1.01e-07
k = 2	Truth	4.21e-03	NA	NA	9.49e-04	NA	NA
	Bootstrap	4.85e-03	1.22e-03	**1.89e-06**	1.05e-03	1.89e-04	**4.62e-08**
	Jackknife	7.46e-03	6.30e-03	5.02e-05	1.39e-03	1.93e-03	3.90e-06
	Unbiased (B=10^2)	4.83e-03	2.38e-03	6.05e-06	9.95e-04	3.43e-04	1.20e-07
	Unbiased (B=10^3)	4.81e-03	2.12e-03	4.87e-06	9.94e-04	2.78e-04	7.94e-08
k = 3	Truth	3.92e-03	NA	NA	8.96e-04	NA	NA
	Bootstrap	4.41e-03	1.07e-03	**1.40e-06**	9.62e-04	1.32e-04	**2.18e-08**
	Jackknife	6.81e-03	5.21e-03	3.55e-05	1.12e-03	1.07e-03	1.20e-06
	Unbiased (B=10^2)	4.68e-03	2.05e-03	4.78e-06	9.80e-04	2.20e-04	5.55e-08
	Unbiased (B=10^3)	4.68e-03	1.84e-03	3.98e-06	9.87e-04	1.22e-04	2.32e-08
k = 4	Truth	3.68e-03	NA	NA	8.65e-04	NA	NA
	Bootstrap	4.01e-03	1.01e-03	**1.14e-06**	9.02e-04	1.06e-04	**1.26e-08**
	Jackknife	6.25e-03	4.26e-03	2.47e-05	1.07e-03	8.77e-04	8.09e-07
	Unbiased (B=10^2)	4.52e-03	1.73e-03	3.71e-06	9.69e-04	2.30e-04	6.34e-08
	Unbiased (B=10^3)	4.52e-03	1.50e-03	2.96e-06	9.65e-04	1.21e-04	2.47e-08
k = 5	Truth	3.45e-03	NA	NA	8.41e-04	NA	NA
	Bootstrap	3.69e-03	9.11e-04	**8.84e-07**	8.77e-04	9.83e-05	**1.09e-08**
	Jackknife	5.86e-03	3.14e-03	1.56e-05	1.05e-03	6.28e-04	4.37e-07
	Unbiased (B=10^2)	4.39e-03	1.63e-03	3.53e-06	9.63e-04	2.23e-04	6.47e-08
	Unbiased (B=10^3)	4.36e-03	1.40e-03	2.80e-06	9.59e-04	1.22e-04	2.86e-08
k = 6	Truth	3.30e-03	NA	NA	8.20e-04	NA	NA
	Bootstrap	3.46e-03	8.58e-04	**7.60e-07**	8.52e-04	9.10e-05	**9.33e-09**
	Jackknife	5.76e-03	3.35e-03	1.73e-05	1.05e-03	7.31e-04	5.87e-07
	Unbiased (B=10^2)	4.24e-03	1.71e-03	3.80e-06	9.61e-04	2.25e-04	7.06e-08
	Unbiased (B=10^3)	4.23e-03	1.51e-03	3.15e-06	9.54e-04	1.21e-04	3.26e-08

Table 7: Runtimes (in seconds) of each variance estimator, real elapsed time per job.

		50-50	60-40	70-30	80-20
n = 100	Bootstrap (B=10^3)	77.47	79.65	81.47	77.12
	Jackknife (B=10^3)	85.01	80.90	66.32	78.70
	Unbiased (B=10^2)	0.43	0.82	0.41	0.58
	Unbiased (B=10^3)	3.56	6.84	3.63	4.54
n = 500	Bootstrap (B=10^3)	116.03	113.26	112.68	110.63
	Jackknife (B=10^3)	126.68	121.98	124.65	122.53
	Unbiased (B=10^2)	3.08	2.76	1.90	1.60
	Unbiased (B=10^3)	17.76	25.03	9.61	13.58

In Tables 3, 4, 5, and 6, we show the mean, standard deviation, and mean squared error (MSE) of the estimated variances over all R simulations. From those Tables, we can see that bootstrap often performs the best in terms of the MSE. Jackknife can be significantly biased upwards compared to the simulated truth, especially at smaller sample sizes such as $n = 100$. As its variance is also generally higher than both the bootstrap and unbiased methods, it is unable to compete with either, even as its performance improves with increased n.

Although bootstrap performs the best in most of the simulations, the $B = 10^3$ unbiased estimator often achieves MSEs close to those achieved by bootstrap. In general, it seems that in cases with more evenly split classes, the unbiased estimator is more likely to even outperform bootstrap — this may be due to the fact that the partition form of the unbiased U-statistic estimator is dependent on the size of the smaller class n_1 and thus makes the estimator less powerful in these cases. Overall, we note that the performance of the proposed variance estimator improves with a larger value of B in achieving a smaller bias, smaller sd, and smaller MSE, as the estimator implemented was not the exact unbiased variance estimator. Theoretically, as B goes to infinity, the partition-based realization would be equivalent to the exact unbiased variance estimator and show a smaller bias than the bootstrap method.

Additionally, we can also see from Table 7 that with the help of the partition resampling scheme, one can realize the unbiased variance estimator of a two-sample U-statistic quite efficiently. With $B = 10^2$, the performance of the proposal is already fairly good and has very comparable means, standard deviations, and MSEs compared to bootstrap, but with much improved efficiency.

With $B = 10^3$, which is the same number of replications as used in bootstrap and jackknife, the computational cost of the proposed method is about 15 to 20 times faster at $n = 100$, and about 10 times faster at $n = 500$, compared to its counterparts.

7 Real Data Analysis

In this chapter, we illustrate the practical application of our proposed variance estimator (10), in comparison to bootstrap and jackknife methods, using a real data set called *Heart Disease*. Because AUC is estimated based on the given data and is subject to sampling error, in the context of model comparison it is questionable whether the model with the largest AUC score is truly optimal or not. To account for variability of AUC and select a possibly more parsimonious model, we consider implementing the one-standard-error rule in choosing the optimal model (Hastie et al., 2009). Our results show that with our proposed variance estimator, the one-standard error rule selects a model that is comparable to the ones selected using bootstrap or jackknife variance estimators. However, the realization of our method is much more efficient than bootstrap and jackknife, using the partition resampling scheme.

The rest of this chapter is structured as follows. In Section 7.1, we briefly introduce the *Heart Disease* data set. In Section 7.2, we then discuss the results and our findings.

7.1 Data

The full *Heart Disease*[2] data set was first assembled by Detrano et al. (1989), and contains 76 attributes to predict up to 4 classes of heart disease from patients in hospitals in Hungary, Switzerland, California, and Ohio. However, most published papers using this data have used a cleaned subset of the initial data, collected from patients undergoing angiography at a Cleveland, Ohio hospital. Additionally, the 4 classes of heart disease are often reduced to a binary prediction problem of whether or not the patient has heart disease. After removing missing data entries, 297 data entries remained of the original 303, with 13 possible predictor variables. There are 160 patients without heart disease and 137 patients with heart disease, resulting in an approximate 54-46 ratio between the negative and positive classes.

In Wang and Lindsay (2017), the same data set was used to illustrate model selection based on various variance estimators using the one-standard error rule. Out of the 2^{13} total possible models, we consider a total of 13 models, each being the optimal one of a given size p ($1 \leq p \leq 13$), where p represents the number of predictors in the model. The BIC criterion was used to

[2]The *Heart Disease* data set was downloaded from the University of California, Irvine (UCI) Machine Learning Repository at https://archive.ics.uci.edu/ml/datasets/heart+Disease

Table 8: Variable meanings in the *Heart Disease* data set.

variable	description	type (# of levels)
thal	exercise thallium scintigraphic defects	categorical (3)
exang	if the patient suffered an exercise-induced angina	categorical (2)
ca	the number of major vessels containing calcium	categorical (4)
slope	slope of the peak exercise ST segment	categorical (3)
cp	chest pain type	categorical (3)
sex	sex	categorical (2)
trestbps	resting blood pressure (mmHg)	quantitative
thalach	maximum heart rate (bpm)	quantitative
chol	serum cholesterol (mg/dl)	quantitative
fbs	if fasting blood sugar > 120 mg/dl	categorical (2)
restecg	resting electrocardiographic results	categorical (3)
oldpeak	exercise-induced ST depression	quantitative
age	age (years)	quantitative

Table 9: Models under comparison in the *Heart Disease* data set.

p	thal	exang	ca	slope	cp	sex	trestbps	thalach	chol	fbs	restecg	oldpeak	age
1	✓												
2	✓	✓											
3	✓	✓	✓										
4	✓	✓	✓	✓									
5	✓	✓	✓	✓	✓								
6	✓	✓	✓	✓	✓	✓							
7	✓	✓	✓	✓	✓	✓	✓						
8	✓	✓	✓	✓	✓	✓	✓	✓					
9	✓	✓	✓	✓	✓	✓	✓	✓	✓				
10	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
12	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
13	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

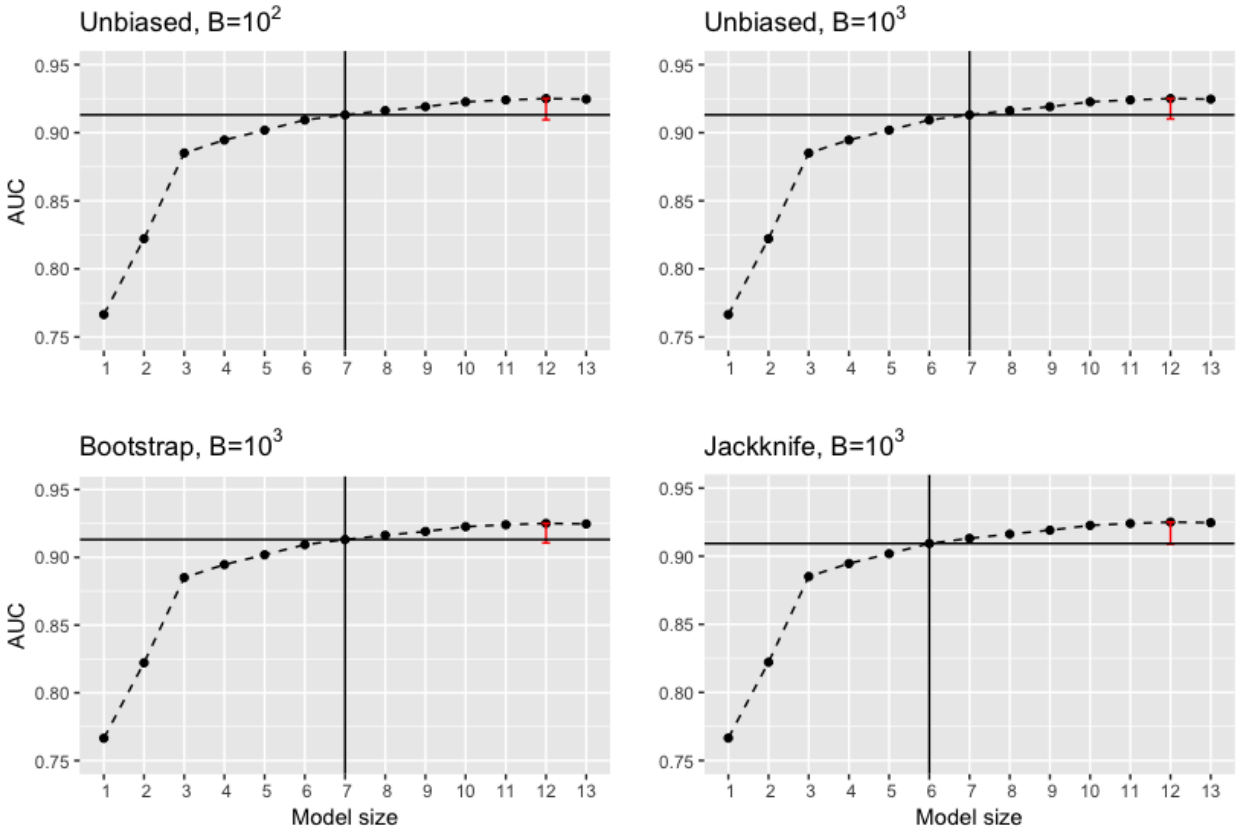
determine which model is optimal for each p, and the models are shown in Table 9. The 13 variables represent the results from submitted patient history and a series of tests administered by the research team (exercise electrocardiogram, thallium scintigraphy, and cardiac fluoroscopy). A more detailed explanation of the variable meanings and data types are given in Table 8.

We fit each of the 13 models (Table 9) on the trimmed full data set and compute their corresponding AUC scores. Model 12 turns out to be the one with the largest AUC. If one were to choose a model based on maximization of AUC, one would select model 12. However, it is highly likely that Model 12 is too complicated and overfits the current data set. In addition, many smaller models, such as Model 3 to Model 11, have AUC scores that are comparable to that of Model 12. Following the rule of parsimony, we would like to select the most parsimonious model whose AUC score is similar to that of Model 12. Here, we implement the one-standard error rule and choose the

model whose AUC score is within one standard error of Model 12's AUC. We then apply bootstrap, jackknife, and our proposed method to compute the standard error of AUC for Model 12.

7.2 Results

Figure 3: 1-SE model selection rule applied to the *Heart Disease* data set.



From Figure 3, we see a standard error bar highlighted in red for each estimator, subtracted from the AUC of the maximum AUC model, which is model size 12. Using the one-standard error rule, the smallest model with an AUC score within that one standard error range is the optimal model. For the *Heart Disease* data set, we see that both unbiased estimators (B=10^2 and B=10^3) agree with the bootstrap method to suggest that model 7 best balances complexity and model performance. The jackknife method recommends a slightly more parsimonious model of size 6; however, as we saw from the previous chapter, jackknife tends to overestimate the variance, which would result in choosing a model that may be too small. Furthermore, from our simulation studies we also know that the calculation of our proposed variance estimator is much more efficient than

the bootstrap and jackknife methods. Therefore, we believe that our developed method offers significant practical value given its computational efficiency and comparable results to existing methods.

8 Discussion and Future Work

In this thesis, we reviewed existing variance estimation methods and proposed an unbiased variance estimator for a two-sample U-statistic (with a general K-sample extension). We focused our attention on binary classification and the widely used performance metric for binary classifiers, AUC, that has a two-sample U-statistic representation with degree $k = 2$. We formulated the proposed unbiased variance estimator in the context of AUC, and designed simulation studies to compare our estimator with the bootstrap and jackknife resampling-based variance estimators. Our results showed that our unbiased estimator not only performs significantly better than jackknife, and is comparable to bootstrap method in terms of mean squared error, but also achieves computational speeds of up 10 to 20 times faster than its competitors (exact speed-up depends on the data set size). We also illustrated the practical performance of our developed variance estimator on a real data set.

Given that our unbiased estimator is competitive with bootstrap and jackknife and less computationally expensive, we recommend its usage in cases where the statistic of interest has a K-sample U-statistic form. Thus, for future work, we would find two- or K-sample U-statistic formulations for other widespread metrics and demonstrate the performance of our U-statistic for those cases. Additionally, we would like to further investigate the use of variance estimation in model selection — in our simulation, although bootstrap, jackknife, and our unbiased estimator all achieved low MSEs with regard to the simulated truth, using these estimated variances in combination with the one-standard error rule resulted in models that were too parsimonious, given that the true model we constructed was of size 5. Thus, another interesting future question would be investigating alternate applications for the proposed variance estimator, possibly using the asymptotic normality of U-statistics to construct confidence intervals or hypothesis tests to compare models of different sizes.

References

Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Brisimi, T.S., T. Wang, W. Dai, W.G., Adams, and I.C. Paschalidis (2018). Predicting chronic disease hospitalizations from electronic health records: an interpretable classification approach. *Proceedings of the IEEE Institute of Electrical and Electronics Engineers*, 106, no. 4, pp. 690-707.

Carreras X. and L. Marquez (2001). Boosting trees for anti-spam email filtering. In *4th International Conference on Recent Advances in Natural Language Processing*, Tzigov Chark, Bulgaria, pp. 58–64.

Detrano, R., A. Janosi, W. Steinbrunn, M. Pfisterer, J. Schmid, S. Sandhu, K. Guppy, S. Lee, and V. Froelicher (1989). International application of a new probability algorithm for the diagnosis of coronary artery disease. *American Journal of Cardiology*, 64, pp. 304-310.

Efron, B. (1979). Bootstrap methods: another look at the jackknife. *Annals of Statistics* 7, no. 1, pp. 1-26.

Efron, B. and C. Stein (1981). The jackknife estimate of variance. *Annals of Statistics* 9, no. 3, pp. 586-596.

Everitt, B. and T. Hothorn (2011). *An Introduction to Applied Multivariate Analysis with R*. New York: Springer.

Faraggi, D. and B. Reiser (2002). Estimation of the area under the ROC curve. *Statistics in Medicine*, 21, pp. 3093-3106.

Fukuchi, S., K. Hosoda, K. Homma, T. Gojobori, and K. Nishikawa (2011). Binary classification of protein molecules into intrinsically disordered and ordered segments. *BMC Structural Biology*, 11, no. 1, pp. 29-39.

Gupta, Y., K.H. Lee, K.Y. Choi, J.J. Lee, B.C. Kim, and G.R. Kwon (2019). Alzheimer's disease diagnosis based on cortical and subcortical features. *Journal of Healthcare Engineering*.

Hammond, R., R. Athanasiadou, S. Curado, Y. Aphinyanaphongs, C. Abrams, M.J. Messito, R. Gross, M. Katzow, M.Jay, N. Razavian (2019). Predicting childhood obesity using electronic health records and publicly available data. *PLoS One*.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning: Data*

Mining, Inference, and Prediction. Springer.

Hoeffding, W. (1948). A class of statistics with asymptotically normal distribution. *The Annals of Mathematical Statistics*, 19, no. 3, pp. 293-325.

Kumari, R. and S. Srivastava (2017). Machine learning: a review on binary classification. *International Journal of Computer Applications* 160, no. 7, pp. 11-15.

Lee, A.J. (1990). *U-statistics: Theory and Practice*. New York: Marcel Dekker.

Lobo, J.M., A. Jimémez-Valverde, and R. Real (2008). AUC: a misleading measure of the performance of predictive distribution models. *Global Ecology and Biogeography* 17, no. 2, pp. 145-151.

Neter, J., M.H. Kutner, W.J. Nachtsheim, and W. Li (2005). *Applied Linear Statistical Models* (5 ed). McGraw-Hill Irwin.

Promjiraprawat, K. and W. Wongseree (2016). Smoothing spline for the AUC estimate: simulation studies in Gaussian data. In *2016 Management and Innovation Technology International Conference (MITicon-2016)*, Bang-San, Thailand, 12-14 Oct. 2016. IEEE.

Shao, J. and C.F.J. Wu (1989). A general theory for jackknife variance estimation. *The Annals of Statistics* 17, no. 3, pp. 1176-1197.

Smith, J.W., J.E. Everhart, W.C. Dickson, W.C. Knowler, and R.S. Johannes (1988). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. *Proceedings of the Annual Symposium on Computer Application in Medical Care*, pp. 261–265.

Wang, Q. and B.G. Lindsay (2014). Variance estimation of a general U-statistic with application to cross-validation. *Statistica Sinica* 24, pp. 1117-1141.

Wang, Q. and B.G. Lindsay (2017). Pseudo-kernel method in U-statistic variance estimation with large kernel size. *Statistica Sinica* 27, no. 3, pp. 1155-1174.

Yan, L., R. Dodier, M.C. Mozer, and R. Wolniewicz (2003). Optimizing classifier performance via an approximation to the Wilcoxon-Mann-Whitney statistic. In *Proceedings of the 20th International Conference on Machine Learning (ICML-2003)*, Washington DC, 2003.