On Selecting a Probabilistic Classifier for Appointment No-show Prediction

Shannon L Harris
School of Business, Virginia Commonwealth University, Richmond, VA 23284, USA
{harriss10@vcu.edu}

Michele Samorani
Leavey School of Business, Santa Clara University, Santa Clara, CA 95053, USA
{msamorani@scu.edu}

Appointment no-shows are disruptive to healthcare clinics, and may increase patient waiting time and clinic overtime, resulting in increased clinic costs. Appointment scheduling models typically mitigate the negative effects of no-shows through appointment overbooking. Recent work has proposed a predictive overbooking framework, where a probabilisitic classifier predicts the no-show probability of individual appointment requests, and a scheduling algorithm uses those predictions to optimally schedule appointments. Because predicting no-shows is typically an imbalanced classification problem, the preferred classifier is often chosen based upon the area under the receiver operator characteristic curve (AUC), which is a commonly used metric for many other imbalanced classification problems. Contrary to intuition, in this paper we show that employing the AUC to select a classifier results in significantly lower schedule efficiency than using other metrics such as Log Loss or Brier Score. Our computational experiments, validated on large real-world appointment data, suggest that by using Log Loss or Brier Score instead of AUC, practitioners can improve the schedule quality by 3-7%.

Keywords: healthcare; appointment no-show; appointment scheduling; classification; classification performance

1 Introduction

A healthcare appointment is labeled as a no-show if the patient fails to show up for a scheduled clinic appointment. No-shows are disruptive to a clinic, may cause access and scheduling issues, and may increase the cost of clinic operation. No-show rates vary, but have been reported to average 23% based upon recent studies [1]. This variability in patient attendance behavior makes it difficult for a healthcare provider to offer high-quality service to its patients at a low cost. Many articles in the literature show that clinics may mitigate the negative affect of patient no-shows by overbooking appointment slots, that is, scheduling more than one person in an appointment slot [2,3,4,5,6].

The first step in determining how to overbook slots in a clinic schedule is to train a model to predict the no-show probability of each individual appointment request. Those no-show probabilities can then be used as input to a scheduling model to determine the optimal schedule. Whereas early work on overbooking assumed that all patients had the same no-show probability, recent work [7,8] has found that utilizing individualized no-show probabilities results in better quality clinic schedules. Due to the number of modeling techniques that are available to build a no-show prediction model (e.g. logistic regression, decision trees, artificial neural networks, etc), it is often best practice to build several predictive models and then determine which is the preferred model that should be utilized when building a clinic schedule. Choosing the preferred model is typically done by measuring the performance of multiple models on available data, for example through cross validation, and then selecting the model with the best classification performance.

The focus of this study is on how to select a preferred no-show prediction model, where the probabilities from the model will be used as input to a scheduling model. Intuitively, the greater the classification performance of a no-show prediction model, the smaller the cost of the schedules obtained using those predictions as input [7,8]. However, how to measure classification performance in this particular domain is unclear. In the majority of the existing work in literature, the quality of a model for no-show prediction is measured with the area under the receiver operator characteristic curve (AUC). Because predicting no-shows is typically an imbalanced classification problem (i.e., patients are much more likely to show up for their appointment), evaluating the classification performance based upon the AUC may seem reasonable, as AUC is commonly used to evaluate models for other imbalanced classification problems (see [14,15]). However, we will show that selecting the model with the greatest AUC does not necessarily lead to the minimum scheduling cost.

In the literature, several types of models have been utilized to predict no-shows. Those models include decision tree models [9], association rule mining [3], and functional approximation modeling [12]. The most common model utilized is the logit model [10,11,13,16]. Li et al. [7] extend the standard logit

model and develop a Bayesian nested logit model. Similarly, Alaeddini et al. [17] construct a hybrid probabilistic model based on multinomial logistic regression and Bayesian inference. The metric that is used in all of those studies to evaluate the classification performance is the AUC.

In this paper, however, we challenge the widespread assumption that AUC is an appropriate metric to use in the context of choosing a preferred no-show prediction model. Our research question is as follows: *Which performance metric should be used to select a probabilistic classifier that estimates the no-show probabilities used as inputs to a scheduling model?* In order to answer that question, we employ seven common models – artificial neural networks (NN), logistic regression (LR), decision trees (DT), random forest (RF), discriminant analysis (DA), AdaBoost (AB), and Naïve Bayes (NB) – to calculate no-show probabilities. In the remainder of this paper, we refer to those models, and any model that can be utilized to model no-show probabilities, as probabilistic classifiers, as they are all capable of providing not only a binary outcome, but also a no-show probability. Models such as logistic regression are probabilistic by nature, and the output of all other methods can be translated into probabilities using Platt's method [21]. We also design a schedule optimizer which takes as input individual no-show probabilities. To design our schedule optimizer, we improve the model introduced by Samorani & Harris [8] by adding constraints that result in reducing solution times by as much as 90%. That reduction in solution time enables us to perform large computational studies to rigorously answer our research question. We conduct two computational experiments to analyze three performance metrics – AUC, Brier score, and Log Loss – and determine which metric leads to the best clinic performance, as measured by the lowest schedule cost. Detailed definitions of the classifiers and performance metrics can be found in Section 3.3 and 3.4, respectively.

Samorani & Harris [8] introduce a framework that integrates individual no-show probabilities with a scheduling model, but do not provide guidelines on which metric is appropriate to use when selecting the prediction model. Our aim is to assess which metric among AUC, Brier Score, and Log Loss is most highly correlated with the schedule cost, and will therefore lead to the schedule with the lowest cost. After analyzing the difference in correlations between cost and performance metric in two studies – one study

with real-world data and another study with generated data – we find that the cost associated with the schedule based upon the classifier chosen with Brier score or Log Loss can range from 3% to 7% less than the cost of the schedule when AUC is used to select the probabilistic classifier. Additionally, we find that when Brier score or Log Loss are utilized to choose a probabilistic classifier, the resulting schedule is more correlated with a lower schedule cost. There is a slight advantage to using Log Loss as a performance metric when analyzing the correlations, but it is not always statistically significant. Thus, based upon the results of our experiments, we suggest Brier score or Log Loss be used as a performance metric to choose a probabilistic classifier, as opposed to AUC.

The contributions of this paper are as follows. First, we analyze how the prediction performance of a classifier impacts scheduling outcomes and choice of performance metric. In the literature, papers typically focus only on the prediction performance of a classifier, without investigating whether that performance translates into a reduction in the schedule cost, which is the outcome that ultimately matters to practitioners. Second, to the best of our knowledge, this is the first paper to conduct a statistical study on the correlation between performance metrics and schedule cost to measure the quality of a data mining model. We contribute to the body of knowledge of healthcare scheduling and analysis in this paper, and provide results for both real data and generated data, which allows our results to be more generalizable. Lastly, we present a fast and easy-to-implement scheduling model that allows for individual no-show probability predictions as input, and that can be a valuable resource to practitioners interested in solving a scheduling problem in real-time.

The rest of the paper is organized as follows. In Section 2 we review related literature, in Section 3 we describe our Methodology. Our Results and Discussion are in Section 4, and we conclude and discuss our limitations and future research directions in Section 5.

2 Related literature

In this section we review literature related to no-show prediction using analytical models, the performance metrics used to evaluate no-show prediction models, and papers that have integrated no-show prediction within a scheduling framework.

Dantas [1] reviewed no-shows in appointment scheduling, and found that over half of the studies between 1980 and 2016 modeled no-show probabilities using logistic regression. Machine learning methods have also become more prevalent with naïve bayes [29], artificial neural networks [28,29,31,33], support vector machines [28], decision trees [9], random forest [28,31,33], association rule mining [3], and Bayesian models [7,17,32,30] often being used to understand patient attendance behavior. The objective of those papers is typically to develop robust models to predict no-shows, compare models to determine a preferred model, and oftentimes, develop feature selection methods to improve estimates and create a more parsimonious model [28,29,30,32,33]. Ferro et al. [31] develop a no-show prediction model with the purpose of integrating the model into a larger decision support system to encourage patient attendance. Nasir et al. [28] present a scheduling model that takes as input specific measurements from the predictive model. All of those papers focus on predicting no-shows, and do not investigate whether a "good-quality" prediction translates into a good-quality schedule. In contrast, in this paper we study how the classification performance of different classifiers, as measured by different metrics, translates into schedule quality.

Features commonly used to predict patients' no-show probability are their demographics, their past no-show history, and appointment characteristics such as lead time (i.e., the time between when an appointment is made and when it is to occur), day of the week, and whether the appointment was a follow-up [9,10,11,12,13,28,29,30,31,32,33]. Lead time and prior no-show history are often identified as key drivers of future patient behavior. Simsek et al. [30] develop a Tree Augmented Naïve Bayes (TAN) based model which is able to identify relationships between predictors. They find that lead time is the most directly influential variable, with age, whether the patient received a text reminder, the time a patient called for an appointment, and time between appointments all being dependent on lead time. Ferro et al. [31]

study data from underserved communities in Bogota, Colombia, and find that income and neighborhood crime statistics affect no-show probabilities. For the models built in this paper, we utilize features based upon the variables shown to be relevant in prior work and upon the availability of the variables in our dataset.

When model selection is involved with no-show prediction, the typical metrics used to compare models are AUC, accuracy, sensitivity, and specificity [28,29,30,31,32,33]. Topuz et al. use the G-Mean, which attempts to balance accuracies in both the positive and negative class while also maximizing them, in model selection. Brier score is used in Simsek et al. [30] to validate the probabilistic consistency and reliability of the estimate.

Lastly, we review papers that have integrated a no-show prediction model within a scheduling framework. Recent work [6,7,8] has promoted the inclusion of individual no-show probabilities into a scheduling model. The goal of those papers was to minimize the clinic cost (typically, the patients' waiting time and the provider's overtime and idle time) using individual no-show probabilities, and found that better classification performance leads to a better-quality schedule. Given that greater classification performance of a no-show prediction model has been shown to lead to smaller the cost of a schedules [7,8], it is important to choose the appropriate model.

3 Methodology

In this section, we first describe a scheduling framework that takes as input individual no-show probabilities, and present a scheduling model that can generate optimal schedules. We then discuss the probabilistic classifiers and classification performance metrics that we study in the paper. Finally, we describe the computational experiments used to answer our research question.

3.1 Predictive overbooking framework

In order to mitigate the negative affect of appointment no-shows, clinics may overbook appointment slots. Overbooking must be done strategically in order to not result in excess patient waiting time and clinic overtime. Thus, a predictive overbooking scheduling framework may be utilized, which

take as input patient no-show probabilities, to optimize a clinic schedule. The design of a predicting overbooking framework are shown in Figure 1.

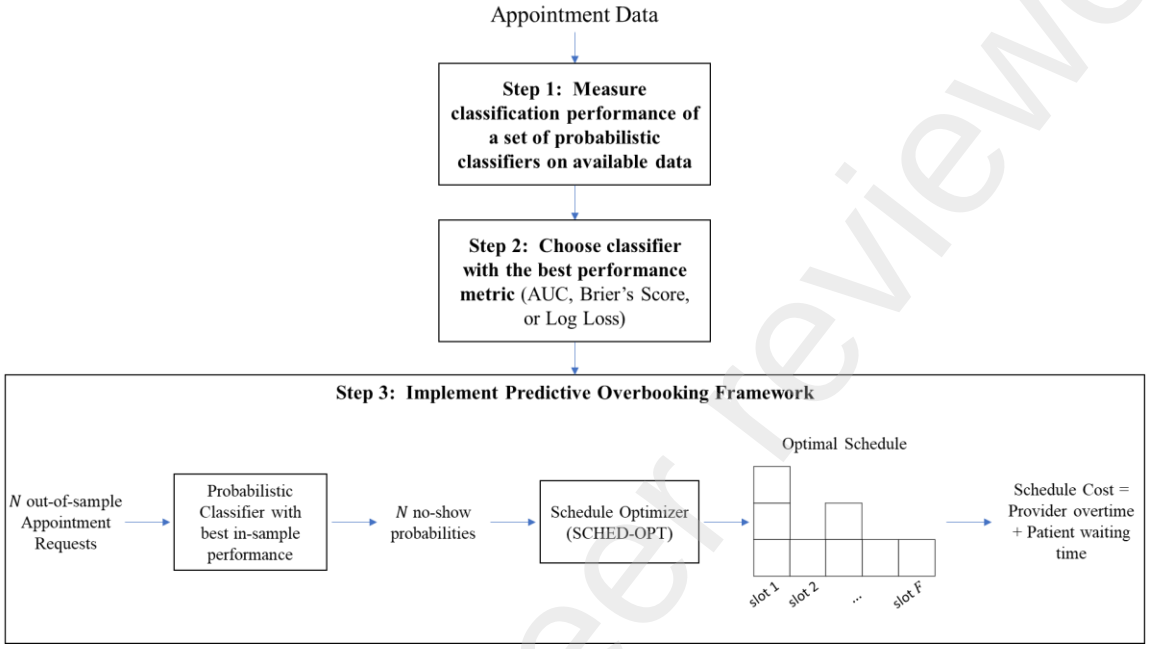

Figure 1: Design of the predictive overbooking framework

The first step when designing a predictive overbooking framework is to choose a probabilistic classifier to predict appointment no-show probabilities. That is typically done by measuring the performance of multiple classifiers on available data (Step 1 in Figure 1), and then selecting the classifier with the best performance (Step 2 in Figure 1). After a classifier is chosen, the predictive overbooking framework can be implemented (Step 3 in Figure 1). The framework is composed of two components: the probabilistic classifier which predicts the no-show probability of N incoming appointment requests, and a schedule optimizer which takes those predictions as input to optimally schedule the requests in F appointment slots. We assume $F < N$, that is, some requests are scheduled in the same appointment slot (i.e., they are overbooked). The schedule optimizer assigns an appointment slot to each appointment request with the objective of minimizing the expected patients' waiting time and the provider's overtime. This paper focuses on the scheduling problem where N appointment requests arrive at the same time,

because this problem is the building block to developing more realistic models where appointment requests arrive sequentially.

A patient will experience waiting time if their scheduled appointment does not start at the predefined appointment time because the provider is seeing another patient, and the provider will experience overtime if there are patients waiting to be seen at the end of the clinic session. As an illustration, consider the schedule depicted in Figure 2, where a set of $N = 8$ appointment requests are scheduled in $F = 5$ appointment slots.

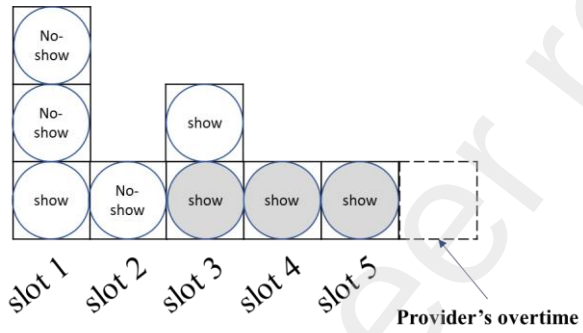

Figure 2: Example schedule with $N = 8$ appointment requests scheduled in $F = 5$ appointment slots. Grey circles indicate the patients that will experience waiting time. The dashed slot indicates a slot needed for provider's overtime.

Overbooked slots (slots 1 and 3 in Figure 2) are typically depicted as a vertical stack of appointment slots. When patients are scheduled, the only information available is their individual no-show probabilities; whether the patient actually shows or not is known only when the appointment is to occur. Suppose that three of the eight patients depicted in Figure 2 do not show up (two in slot 1 and one in slot 2). In slot 1, the provider sees the only patient who shows up in that slot. The provider becomes available at the beginning of slot 2, but will stay idle during that slot because no other patient shows up. In slot 3, two patients show up. However, the provider can only see one patient during each appointment slot, causing one patient to be delayed to slot 4. Because two more patients show up in slot 4 and slot 5, the delay propagates until the end of the clinic session, causing the three patients depicted in grey to experience

waiting time. Finally, the provider also experiences one slot of overtime, because the last patient will have to be seen after the clinic nominal end time. Thus, in this example, the total patients' waiting time is three time units and the provider's overtime is one time unit.

3.2 Schedule optimizer model

In this section, we illustrate our solution method for the scheduling problem with individual no-show probabilities, which is an extension of the model in Samorani & Harris [8]. We consider an outpatient clinic with a single provider who sees patients sequentially. We assume there are N appointment requests per day that must be scheduled in F appointment slots. Each slot is of equal length, and appointments times are constant and equal to the length of an appointment slot. The output of the scheduling model is an assignment of the N appointment requests into the F appointment slots. We assume that $F < N$ such that each slot is booked, and some slots may be overbooked. The i-th appointment request, $i = 1, \dots, N$, is assumed to no-show with probability q_i. Patients who do show are assumed to be punctual.

A common objective of the scheduling problem is to schedule appointment requests in order to minimize a weighted average of the patients' waiting time and the provider's overtime [24]. Waiting time is incurred by a patient when their appointment does not begin at the scheduled time, and overtime is incurred when the provider must see patients past the nominal end of the clinic day. We assume a cost of ω dollars for every unit of waiting time incurred by each patient, and a cost of τ dollars for unit of overtime incurred by the provider. Without loss of generality, we fix the waiting time cost to 1, and we vary the overtime cost τ. Our notation is reported in Table 1.

Table 1: Notation

Parameter	Description
F	Number of appointment slots
N	Number of appointment requests to schedule
q_i	The no-show probability of appointment request i, with $i = 1, \dots, N$
ω	The cost per time unit that each patient incurs for starting their appointment late
τ	The cost per time unit of overtime

q_i^s	A binary scalar that indicates whether request i will not show under scenario s, with $i = 1, \dots, N$ and $s = 1, \dots, S$
$v_{i,i+1}$	A binary scalar that indicates whether patient i's no-show probability is equal to patient $i+1$'s no-show probability, i.e., $v_{i,i+1} = 1$ iff $q_i = q_{i+1}$, for $i = 1, \dots, N-1$
x_{ij}	A binary decision variable indicating whether the i-th appointment request is scheduled in the j-th slot, with $i = 1, \dots, N$ and $j = 1, \dots, F$
b_j^s	The number of patients that under scenario s overflow from slot j to slot $j+1$, with $j = 1, \dots, F$ and $s = 1, \dots, S$

We model the scheduling problem as a stochastic mixed integer linear program. Without loss of generality, we assume that the no-show probabilities, q_i, are pre-sorted by increasing probability of no-show, that is, $q_i \leq q_{i+1}$ for $i = 1, \dots, N-1$. That assumption assists in speeding up computation time, a concept which we discuss in detail at the end of this section.

The individualized no-show probabilities act as input to the scheduling problem. Given the no-show probabilities, q_1, q_2, …, q_N, we build S no-show scenarios, each corresponding to one possible realization of patient attendance. That is, each scenario corresponds to a binary sequence of 0's and 1's to indicate if a patient will show (0) or no-show (1). Because there are 2^N subsets of N requests, the total number of scenarios is $S = 2^N$. Each scenario s, ($s = 1, \dots, S$) is represented by a binary vector [q_1^s, q_2^s, …, q_N^s] of length N. The probability of scenario s occurring, p^s, is given in Equation (1). It is calculated as the joint probability of observing all requests such that $q_i^s = 1$ and all requests such that $q_i^s = 0$, for all i.

$$p^s = \prod_{i=1,\dots,N}\left(q_i^s q_i + (1 - q_i^s)(1 - q_i)\right), s = 1, \dots, S \quad (1)$$

We demonstrate the calculation of p^s with an example. Consider a small problem with $N = 3$ requests with no-show probabilities q_1, q_2, and q_3. In that problem, there are 8 scenarios, each indicating which patients will show or no show. For example, scenario $[q_1^s, q_2^s, q_3^s] = [0,0,1]$, indicates that the first two patients will show up and the third will not. The probability of this scenario occurring is equal to $(1 - q_1)(1 - q_2)q_3$.

The cost incurred by the clinic for each scenario is computed by summing the waiting time and overtime costs incurred in each slot. Waiting time and overtime costs are incurred based upon how events unfold throughout the clinic day for each attendance scenario. Waiting time is incurred as follows. The number of people available for service at the beginning of slot j under scenario s is equal to the number of patients scheduled in slot j who show up under scenario s, and the number of patients waiting at the end of the previous slot, slot $j-1$. The number of patients waiting at the end of slot $j-1$ who overflow to slot j under scenario s is stored by a decision variable b_{j-1}^s. As long as there is at least one patient available to be seen, the provider begins seeing one patient at the beginning of slot j, and finishes seeing the patient at the end of slot j. Thus, there will be one patient being served and b_j^s patients waiting for the duration of the slot. In slot j under scenario s, the clinic incurs a waiting cost equal to ωb_j^s (i.e., a cost of ω for every patient waiting during slot j under scenario s). At the end of slot j, the patient being served exits the system. If other patients have been waiting during slot j without being served, they overflow to the next slot.

Overtime is incurred based upon the following sequence. At the end of the last slot, slot F, the provider must see all patients waiting for service at a rate of one patient per time unit. Thus, if there are b_F^s patients at the end of the last slot under scenario s, the clinic will incur an overtime cost of τb_F^s. During overtime, the provider also incurs a waiting time cost of $\omega\big((b_F^s-1)+(b_F^s-2)+\cdots+1\big)$, as the number of patients waiting in the first overtime slot is b_F^s-1, and b_F^s-2 in the second overtime slot, etc. The last overtime slot to be considered is slot F^{max}, which we will show is equal to N.

We now present a stochastic mixed integer linear programming model to find the optimal schedule for a given a set of appointment requests. The assignment of requests to slots is determined by a set of binary decision variables, x_{ij}, which are equal to 1 if request i is scheduled to slot j (for i = 1,…,N and j = 1,…,F).

SCHED-OPT = $min \sum_{s=1\ldots S} p^s\big(\omega \sum_{j=1\ldots F^{max}} b_j^s + \tau b_F^s\big)$

st:

$$\sum_{j=1,\dots,F} x_{ij} = 1 \; \forall i = 1 \dots N \tag{2}$$

$$b_j^s \geq b_{j-1}^s + \sum_{i=1\dots N} x_{ij}(1 - q_i^s) - 1 \; \forall s = 1 \dots S, j = 1 \dots F^{max} \tag{3}$$

$$\sum_i x_{ij} \geq 1 \forall j = 1 \dots F \tag{4}$$

$$\sum_{j=j'+1,\dots,F} x_{ij} \leq 1 - v_{i,i+1} \sum_{j=0,\dots,j'} x_{i+1,j}\,, \; \forall i = 1 \dots N-1 \; \forall j' = 1 \dots F-1 \tag{5}$$

$$x_{ij} \in \{0,1\} \; \forall s = 1 \dots S, j = 1 \dots F \tag{6}$$

$$b_j^s \geq 0 \; \forall s = 1 \dots S, j = 1 \dots F^{max} \tag{7}$$

The objective is to minimize the expected total cost of the schedule. Under each scenario, a given assignment of requests to slot (i.e., a schedule) will result in a scenario-dependent cost equal to the sum of the waiting time costs and the overtime cost. Our model computes the expected cost of a schedule as the average cost obtained under each scenario s, weighted by the probabilities p^s, $s = 1, \dots, S$.

Constraint set (2) states that each patient i must be assigned to exactly one slot. Constraint set (3) assigns the value of b_j^s, i.e., the number of patients overflowing from slot j to slot $j+1$. It is at least equal to the number of patients overflowing into slot j, b_{j-1}^s plus the number of arrivals in slot j ($\sum_{i=1\dots N} x_{ij} q_i^s$) minus one to account for the fact that one patient will be seen in slot j. Constraint set (4) prevents the presence of empty slots in the schedule. As proved by LaGanga and Lawrence [4], this is an optimality condition, and we include it to reduce the solution space and, consequently, the computation time. Because of this condition, the maximum overtime incurred by the provider will be equal to $N - F$ time units, which may be incurred by assigning one patient to each of the regular F slots, and the remaining $N - F$ patients to slots as overbooked patients. If all patients show up, the provider will see the last patient in slot N (i.e., $N - F$ slots overtime). Thus, we set $F^{max} = N$.

Constraint set (5) aids in reducing the model computation time, and extends the mathematical model proposed by [8]. The solution procedure for large scheduling problems may be simplified by mapping the original input no-show probabilities into a small number of unique values. For example, if

the original no-show probabilities are [0.1, 0.2, 0.6, 0.7, 0.8], a set of simplified no-show probabilities with only two unique values is [0.15, 0.15, 0.7, 0.7, 0.7]. The simplified no-show probabilities can be obtained by clustering the original no-show probabilities using, for example, *K*-means with $K = 2$ [18]. In the case where patient i's and patient $i + 1$'s no-show probabilities are equal, any two schedules obtained by swapping these two patients have the same cost. To avoid considering all these equivalent schedules, constraint set (5) forces patient $i + 1$ to be scheduled "not before" patient i if their no-show probabilities are equal. This constraint is aided by the fact that we restrict probabilities to be pre-sorted by increasing probability of no-show. It is easy to verify that this constraint set reduces computation time. Constraint sets (6) and (7) restrict the slot assignment variables to be binary and the patient overflow variables to be non-negative, respectively.

3.3 Classifiers

Predicting no-shows is a binary classification problem, where the positive class is composed of the appointment requests that resulted in a no-show, and the negative class is composed of the appointment requests that resulted in a show. To solve the classification problem, a classifier must be trained on past no-show data, and employed to predict the no-show outcome of future appointment requests. In order to obtain individual no-show probabilities, we utilize seven common models: artificial neural networks (NN), logistic regression (LR), decision trees (DT), random forest (RF), discriminant analysis (DA), AdaBoost (AB), and Naïve Bayes (NB). We chose those analytical models due to their prevalence in the classification literature, and because clinics should have easy access to those types of models if they choose to analyze no-show probabilities. We refer to the analytical models chosen as probabilistic classifiers, where a probabilistic classifier is defined as a classifier that is capable of providing not only a binary no-show outcome, but also a no-show probability that may be utilized in a scheduling model.

Some classifiers are probabilistic by nature (e.g., logistic regression), in that their raw output is a set of probabilities. Other classifiers are capable of labeling each appointment request with a numeric no-show score which can be used to generate a no-show probability. As explained by Niculescu-Mizil &

Caruana [20], a common technique to translate those numeric scores into probabilities is Platt's method [21]. Using Platt's method, a logistic regression model is tuned that predicts the binary no-show outcome given the no-show score. We utilize Platt's method to generate no-show probabilities for all classifiers in this paper, aside from logistic regression. We provide a brief description of each of the models below; for a more detailed description the authors refer the reader to the following texts [18,19,26].

3.3.1 Artificial neural networks

The concept behind artificial neural networks (NN) was inspired by biological neurons, and the technique is widely recognized for its ability to accurately model complex, nonlinear relationships. A NN is "a set of connected input/output units in which each connection has a weight associated with it." During the learning phase, the network learns by adjusting the weights" to be able to predict the class label [18]. Although NNs provide reliable estimates, they are considered a black-box model due to the lack of interpretability of the relationship between the inputs and outputs.

3.3.2 Logistic regression

Logistic regression (LR) is the most common modeling technique to predict no-show probabilities. It is a standard regression technique, where the dependent variable is a binary variable. In LR, the log odds of the positive class (e.g. appointment no-show) is modeled as a linear combination of the independent variables in the model. The relationship between the independent variables and the output probability are easily interpretable, and the model output is the probability of an observation being in the positive class.

3.3.3 Decision trees

Decision Trees (DT) are an intuitive classification method that are capable of discovering nonlinear relationships. There are several algorithmic variations of decision trees, each with the goal of partitioning data to obtain a set of rules or an equation to classify the data. Tree classifiers split observations into classes based upon the independent variables in order to create maximum separation [18]. DTs are an attractive modeling technique because the final result is a tree-like structure from which a set of simple

rules can be easily generated and evaluated to determine the class assignment of each observation. Predictions for each observation in a leaf-node are made by the majority rule.

3.3.4 Random forest

Random forest (RF) is an example of an ensemble method, which combines a series of decision trees to create an improved classification model [18]. Each tree in the forest is generated using a random sample of independent variables and observations. Random forests are efficient because they rely on fewer independent variables, avoid overfitting due to the ensemble nature, and the individual trees do not need to be pruned [18].

3.3.5 Discriminant analysis

The objective of linear discriminant analysis (DA) is to perform dimensionality reduction while preserving as much of the class discriminatory information as possible [26]. A linear combination of predictor variables is modeled to create maximal separation of the observations; the linear combinations are used to assign observations to a specific class.

3.3.6 AdaBoost

AdaBoost (AB) is another example of an ensemble method, where each observation is assigned a weight. In the AB algorithm, a sample of data is used to generate a model. After a model is created, the algorithm assesses the accuracy of the classification and updates the weight of each observation in order to improve accuracy in the next model. The class label is identified based upon a weighted sum of the result of each classifier.

3.3.7 Naïve bayes

The naïve bayes (NB) algorithm assumes that, given the class assignment of an observation, assumes all independent variables are conditionally independent [18]. That assumption simplifies computation, and, when it holds true, allows NB to be a reliable classifier. Observations are assigned a class based upon the class with the highest posterior probability.

3.4 Performance metrics

The choice of the best classifier (step 2 in Figure 1) is determined by measuring the performance obtained by different classifiers on the available data, for example, in cross-validation. As illustrated in Section 2, the performance metric most commonly used for no-show data is the area under the ROC curve (AUC), with the classifier obtaining the greatest AUC chosen as the preferred one. AUC is particularly common when data are imbalanced. Other metrics include the Log Loss and the Brier's score, which measure the deviation between predicted no-show probabilities and actual no-show outcomes. In this paper, we evaluate each classifier listed in Section 3.3 using the cross-validated AUC, Brier score, and Log Loss. Note that the classifier that performs best according to one metric may not be the one that performs best according to the other metrics. Also, each classifier may result in different schedule costs, as measured by waiting time and overtime. Brief descriptions of the metrics are given in the subsections below.

3.4.1 Area under the ROC curve (AUC)

A ROC curve is a plot of the true positive rate (TPR) versus the false positive rate (FPR) for a classifier at multiple cutoff values between 0 and 1 [18]. The area under the generated curve is denoted as the AUC. AUC lies between 0 and 1, with a greater value being preferred. It is a metric based upon how well a classifier ranks the probabilities against actual outcome. When the ranking of the probabilities aligns with all of the smaller probabilities being assigned a show (0) and all larger probabilities being assigned a no-show (1), then the AUC will increase. This metric is the most commonly used to select a probabilistic classifier.

3.4.2 Brier score

The Brier score was introduced by Glenn Brier in 1950 [22] as a verification for weather forecasts. The Brier score measures the mean squared error between the predicted probabilities of a classifier and the actual outcome, and penalizes predictions that are further away from actual. It ranges from 0 to 1, with a lesser value being preferred. The Brier score can be used to evaluate any categorical outcome that can be structured as true or false.

3.4.3 Log Loss

Log Loss is also referred to as logistic loss or cross-entropy loss [23]. It is calculated by measuring the difference between each predicted probability of a classifier and the actual outcome, with the penalty increasing exponentially with the difference from actual. The final metric is an average of the Log Loss for each estimated probability. A classifier that predicts perfectly would achieve a Log Loss of 0.

In order to analyze the performance metrics, we completed two computational studies. The first study is conducted with real-world data and focuses on all steps of the predictive overbooking framework illustrated in Figure 1, and the second study is conducted with simulated data and focuses on Step 3. A description of the studies is provided in the following subsections.

3.5 Computational study one

In our first computational study, we consider a real-world data set and design a predictive overbooking framework as illustrated in Figure 1. The goal of this study is to compare the quality of the schedules obtained resulting by employing the different performance metrics to choose the classifier at step 2. The data set is a sample of fully anonymized, administrative records of 108,515 scheduled, in-person, weekday appointments from a local primary care clinic. The features include past no-show history, age, appointment lead time, day of the week, month of the year, and flags for whether or not the appointment was a follow-up or if the patient had multiple appointments scheduled in the same day. Of the 108,515 appointment requests, 30% resulted in a no-show.

3.5.1 Step 1: Measure the classification performance on available data

First, we partition the dataset into a calibration set (80% of the data) and a test set (20% of the data). Next, we measure the performance obtained on the calibration set based upon seven common probabilistic classifiers: artificial neural networks (NN), logistic regression (LR), decision trees (DT), random forest (RF), discriminant analysis (DA), AdaBoost (AB), and Naïve Bayes (NB). For each classifier, we execute a 10-fold cross validation on the calibration set. At each iteration of the cross validation, the classifier is trained on 90% of the calibration set, which is subsequently used to label the remaining 10%

with a predicted no-show score. To perform a fair comparison among the classification techniques, we employed the default parameters provided by the scikit-learn package in Python 3 for each classifier. The no-show scores are translated into probabilities using Platt's method, in order to build a probabilistic classifier. This step is performed for every classification technique except LR, as the output of LR is a probability. We then calculate and record the cross-validated AUC, Brier score, and Log Loss metrics for each classifier obtained on the calibration set. Finally, we train each classifier on the entire calibration set, because we will need to predict the no-show probabilities of the test set. We repeat this process for 20 iterations, each time using a different random seed to partition the data set into calibration set and test set. At the end of this process, for each of the 20 calibration-set-test set partitions, we obtain seven trained classifiers, as well as their cross-validated AUC, Brier, and Log Loss values obtained on the calibration set.

3.5.2 Step 2: Choose the classifier with the best performance metric

The next step is to select the classifier to use in the predictive overbooking framework based upon their cross-validated performance, as measured by AUC, Brier's score, or Log Loss. However, in order to assess which metric correlates most strongly to the quality of the schedules obtained, we keep all seven classifiers considered in Step 1, instead of keeping only the best one and discarding the rest.

3.5.3 Step 3: Implement the predictive overbooking framework

Next, we implement a separate predictive overbooking framework for each of the seven classifiers. That is, we use each classifier to predict the no-show probabilities of the test set, and then use those probabilities to schedule appointments. We proceed as follows for each of the seven classifiers. For each of the 20 calibration-test-set partitions generated during Step 1, we construct a set of 50 scheduling problems by randomly sampling with replacement from the test set 50 times N appointment requests to be scheduled into F slots. We study four parameter combinations: $N = 7$ and $F \in \{4,5\}$, $N = 10$ and $F = 7$, and $N = 12$ and $F = 8$. Those choices were based upon the no-show rate of the data set of 30%, in order to obtain a patient load similar to the clinic capacity. The scheduling problems with $N = 10$ and $N = 12$ are solved with the following heuristic: we discretize the original no-show probabilities into three

bins using *K*-means, and use a random sample of 300 scenarios as opposed to 2^N scenarios. Thus, for study one we construct and solve a total of 28,000 scheduling problems (7 classifiers x 20 partitions x 50 schedule samples x 4 parameter combinations).

For each scheduling problem, we estimate the no-show probabilities using each probabilistic classifier trained earlier. For each set of predictions, we employ SCHED-OPT to find the optimal schedule given the estimated no-show probabilities and use the actual attendance values from the dataset to evaluate the schedule. That is, the schedule is found from the no-show probabilities predicted by the classifier, but its quality is evaluated using the patients' real no-show outcomes from the test data.

In summary, in this set of experiments, we considered a real-world data set and collected the schedule quality obtained by seven different classifiers on 4,000 scheduling problems. We analyze the relationship between the classifier metrics and the schedule cost obtained, with the goal of determining which metric should guide the choice of classifier in Step 2 of the predictive overbooking method depicted in Figure 1.

3.6 Computational study two

In reference to Step 3 of Figure 1, it is intuitive to assume that higher-quality no-show probability predictions are associated with a higher-quality schedule generated by SCHED-OPT. The goal of study two is to determine how the quality of the predicted no-show probabilities, as measured through the AUC, Brier's score, or Log Loss, correlates with schedule cost.

In order for our findings to be generalizable rather than specific to the data set considered in Section 2.4, in this set of experiments we randomly generate a large number of scheduling problems where patients are assigned random no-show probabilities, and, for each problem, we randomly generate a large number of probabilistic classifiers with a random classification performance. Each classifier produces a different set of N predicted no-show probabilities, which we subsequently employ as an input for SCHED-OPT to find a schedule. Because each set of no-show probabilities is characterized by a value of AUC,

Log loss, and Brier score, we can assess which one among those three metrics correlates most strongly with higher-quality schedules.

Study two is outlined in Figure 3. First, we generate scheduling problems which consist of assigning N appointment requests to F slots. We set $N = 7$ and $F \in \{4,5\}$. That number of appointment requests is common in outpatient clinics, where appointments are generally scheduled in two sessions per day – one in the morning with N appointments and one in the afternoon with N additional appointments. To simulate different classification performances, we assume that each appointment has a true no-show probability, which we refer to as the ground-truth probability, and that a classifier provides a predicted no-show probability that may deviate from the ground truth. The ground truth no-show probability, q_i, of an appointment request, is generated from a beta distribution with parameters q and σ^2, where q is the population no-show rate and $\sigma^2 \in (0.01, 0.02)$. For each value of F, we fix the mean no-show probability to $q = 1 - F/N$ (i.e., 42.9% and 28.6%). The expression for q results in an expected number of shows equal to the number of slots. We chose the beta distribution to generate no-show probabilities because it has been found to fit well with no-show rates calculated from real data [8]. For all four combinations of number of slots (i.e. average no-show probability) and variance we replicate 50 ground truth probability vectors of length N.

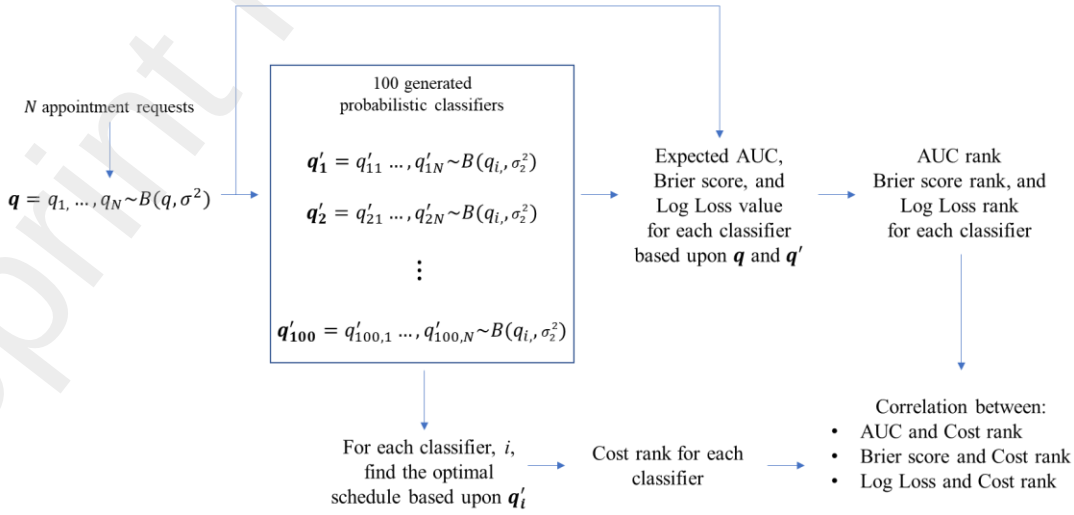

Figure 3: Diagram of study two

For a given scheduling problem, we generate 100 different probabilistic classifiers. We then measure the cost obtained by optimally scheduling appointments based on the no-show probabilities predicted by each of the classifiers. Each classifier is built by randomly generating a vector of predicted no-show probabilities, $\boldsymbol{q}'$, as follows. For each classifier, the predicted probability for patient i is generated using another beta distribution with parameters q_i and σ_2^2, where q_i is the ith patient's true no-show probability and σ_2^2 is a parameter which determines the prediction quality of the classifier and whose value is $\sigma_2^2 \in \{0.01, 0.02\}$. We refer to σ_2^2 as the predicted probability variance. That is, the no-show probability predicted by a classifier for the ith patient is distributed around the patient's ground truth no-show probability, q_i; the deviation from q_i (i.e., the prediction error made by the classifier on that patient) is proportional to σ_2^2. As mentioned above, classifiers are generated by using $\sigma_2^2 = 0.01$ and $\sigma_2^2 = 0.02$.

For each scheduling problem and each randomly generated classifier, we use the vectors of actual and predicted no-show probabilities, $\boldsymbol{q}$ and $\boldsymbol{q}'$ respectively, to compute the classification performance in terms of AUC, Brier score and Log Loss. However, in order to calculate those metrics for a given classifier, the actual attendance – show or no-show – must be known. Because only the no-show probabilities, $\boldsymbol{q}$, not the actual attendance, are known, we calculate the "expected" AUC, Brier score, and Log Loss, as follows. First, we enumerate all possible 2^N no-show scenarios obtained from the vector of actual no-show probabilities, $\boldsymbol{q}$, (i.e., ($s_1 = (0,0,\dots,0,0)$, $s_2 = (0,0,\dots,0,1)$, $s_3 = (0,0,\dots,1,0)$, …, $s_{2^N} = (1,1,\dots,1,1)$). Then, for each scenario, s_i ($i = 1, \dots, 2^N$), we compute the AUC, Brier score, and Log Loss obtained by the predicted no-show vector, $\boldsymbol{q}'$. The probability of each scenario occurring is calculated as in Equation (1). Finally, based upon the probability of each scenario occurring, we calculate the average AUC, Brier score, and Log Loss obtained by every classifier.

After calculating the metric values for all classifiers for a given scheduling problem, each classifier is assigned a rank for each performance metric, with a rank of 1 being the best classification performance

obtained within that scheduling problem. For example, if for a given scheduling problem a classifier obtained the greatest AUC, lowest Brier score, and the lowest Log Loss, it would be assigned a rank of one for all three metrics. Ties are assigned the minimum rank value.

For each scheduling problem, after computing the classification metrics that the 100 classifiers obtain on that problem, we input the generated probabilities into SCHED-OPT in order to find the optimal schedule. In this way, we obtain 100 schedules for each scheduling problem – each obtained using a different classifier to predict the no-show probabilities. For each schedule, we use the ground-truth no-show probabilities to compute its expected cost. Each probabilistic classifier is assigned a rank based upon the expected cost of the resulting schedule, with a rank of one being the best (lowest cost). Ties are assigned a minimum rank value.

To determine which metric correlates best with schedule cost, within each set of 100 probabilistic classifiers, we calculate a Spearman rank correlation between cost rank and AUC rank, cost rank and Brier score rank, and cost rank and Log Loss rank and determine which correlation is statistically greater.

In summary, in computational study two, we investigate the correlation between the AUC, Brier score, and Log Loss and the resulting schedule cost.

4 Results and Discussion

In this section we present our results based upon the computational studies outlined in Sections 3.5 and 3.6.

4.1 Results of study one

In study one, which is outlined in Section 3.5, seven classifiers were trained on real-world data and used to predict the no-show probabilities of a test set; those probabilities were then used to schedule appointments. Our goal is to assess which classifier results in the highest-quality schedules, as measured by the schedule with the largest in-sample AUC, Brier Score, or Log Loss.

To assess that goal, for all 4,000 replications of our experiments, we compare the schedule costs obtained when an optimal schedule is generated based upon the predicted probabilities of the classifier

with the largest cross-validated AUC, the classifier with the smallest cross-validated Brier's score, and the classifier with the smallest cross-validated Log Loss.

For each replication, the classifier with the smallest Brier's score was also the one with the smallest Log Loss, and thus the same cost. Table 2 lists the mean relative deviation of the average costs for each N, F parameter combination, where a positive value indicates that the average schedule cost obtained by choosing the classifier based upon AUC is greater than the schedule cost obtained by choosing the classifier based upon either Brier score or Log Loss. The differences in cost for all combinations are statistically significant at $\alpha = 0.05$. Because the costs for the schedules based upon the classifier chosen with Brier score and Log Loss are equal, we only report one deviation and *p*-value which relates AUC to Brier score and Log Loss.

Table 2: Mean relative cost increase when using the classifier with the best AUC vs using the classifier with either the best Brier score or Log Loss

N, *F*	Cost Increase of Schedule when AUC is Used to Choose Classifier	*p*-value
7, 4	4.50%	<0.0000
7, 5	3.20%	0.0290
10, 7	5.76%	0.0003
12, 8	7.32%	<0.0000

The result for all four parameter combinations indicates that the average cost of the schedule when Brier score and Log Loss are used to choose the classifier is less than the cost when AUC is utilized. The difference in cost can be up to 7% and statistically significant.

We now investigate the correlation between the performance of all seven classifiers, as measured by AUC, Brier Score, and Log Loss, and the quality of the schedule obtained. Within each replication, we rank the seven classifiers based upon four criteria: AUC, Brier score, Log Loss, and cost of the schedule obtained. Then, we compute the correlation between each performance metric and the schedule cost within each replication. Table 3 reports the average correlation obtained across the 1,000 replications for

each N, F combination. Because several correlations are negative, we also list the median correlation for each metric. The greatest average and median for each parameter combination are indicated in bold-face type. AUC is consistently correlated the least with cost, with Log Loss having the greatest mean and Brier score the greatest median. The negative correlations in the Cost vs AUC columns indicate that as the rank of the AUC increases for each classifier, the schedule cost associated with that classifier decreases.

Table 3: Average and median spearman rank correlation between each performance metric and the schedule cost for each parameter combination

	Average Rank Correlation			Median Rank Correlation		
N, F	Cost vs AUC	Cost vs Brier Score	Cost vs Log Loss	Cost vs AUC	Cost vs Brier Score	Cost vs Log Loss
7,4	-0.046	0.232	**0.239**	-0.037	**0.238**	0.231
7,5	0.028	0.168	**0.174**	0.134	**0.223**	0.219
10, 7	0.144	**0.211**	0.206	0.185	**0.267**	0.255
12, 8	0.197	**0.243**	0.233	0.252	**0.291**	0.286

Table 3 suggests that a better classification performance, as measured by Brier Score and Log Loss, tends to lead to better-quality schedules. In contrast, a classifier's AUC ranking is not significantly correlated with the schedule cost obtained, if that classifier is adopted in the predictive overbooking framework.

To determine if the correlations between cost and the three performance metrics are statistically different for each parameter combination, we conducted a One-Way repeated measures ANOVA with metric type as a factor and the three performance metrics as the factor levels, or treatments. We chose a repeated measures ANOVA so we could account for the fact that the performance metrics for each parameter combination are calculated from the same set of probabilities. Each replication is considered a group within the repeated measures analysis. If the mean of the correlations was determined to be different, we conducted paired, multiple-comparisons analysis with Bonferroni correction. At a significance level of 0.05, none of the data sets of the four parameter combinations were found to be normal, and none of the combinations passed Mauchly's test for sphericity [25]. Due to that, the results of the ANOVA were compared against a non-parametric Friedman's test and Wilcoxon signed-rank test; the results were found to be consistent. The *p*-values of all statistical tests can be found in Appendix A.

Table 4 reports the results of the ANOVA and 95% Bonferroni CIs. The statistically significant CIs when $\alpha = 0.05$are depicted in bold type. For all parameter combinations, the Brier score and Log Loss are statistically significantly more correlated with schedule cost than AUC. The magnitude of the difference is greatest when $F = 4$. For the smaller clinic sizes, the average correlation between Log Loss and cost is statistically significantly greater Brier score and cost, while the opposite is true when $N = 10$ and $N = 12$.

Table 4: One-Way repeated measures ANOVA results and paired, multiple-comparisons analysis with 95% Bonferroni CIs for study one

N, F	ANOVA *p*-value	Multiple Comparison with Bonferroni Correction		
		Comparison	Difference	95% CI
7,4	< 0.0000	Brier – Log Loss	-0.008	**(-0.012, -0.003)**
7,4		Brier – AUC	0.278	**(0.253, 0.303)**
7,4		Log Loss – AUC	0.285	**(0.26, 0.311)**
7,5	< 0.0000	Brier – Log Loss	-0.006	**(-0.01, -0.002)**
7,5		Brier – AUC	0.140	**(0.11, 0.171)**
7,5		Log Loss – AUC	0.146	**(0.115, 0.178)**
10,7	0.0009	Brier – Log Loss	0.005	**(0.001, 0.009)**
10,7		Brier – AUC	0.067	**(0.039, 0.095)**
10,7		Log Loss – AUC	0.062	**(0.033, 0.09)**
12, 8	0.0413	Brier – Log Loss	0.009	**(0.005, 0.013)**
12, 8		Brier – AUC	0.045	**(0.019, 0.072)**
12, 8		Log Loss – AUC	0.036	**(0.009, 0.063)**

The scheduling problems with $N = 10$ and $N = 12$ are solved with a heuristic, and were able to be solved due to our inclusion of Constraint set 5 in SCHED-OPT. On average, solving the scheduling problem with $N, F = 10, 7$ without Constraint set 5 takes 29.33 seconds; with the constraint set the time reduces to 1.81 seconds on average. Thus, reducing the computation time by 93.8%. When $N, F = 12, 8$, the average times to solve the problem are 1.60 and 861.41 seconds, with and without the constraint set, respectively, which results in a 99.8% time reduction. Because of the long time needed to solve the problems in the $N, F = 12, 8$ case, the time reduction was computed only in a sample of 400 problems, rather than the entire set of 7,000 problems. The reduction in solution time for the larger instances allows our model to be more valuable to a clinic who may need to solve the scheduling problem in real-time.

The results from study one have implications for clinics deciding on which classifier to use as input into their scheduling system. It is in the interest of the clinic to choose the classifier that will result in the lowest expected clinic schedule cost. Based upon our computational study, a clinic should not use AUC to choose the preferred classifier, because AUC does not correlate with the resulting schedule cost as well as a classifier chosen based upon Brier score or Log Loss. We extend our discussion by analyzing the results of study two, where data sets are simulated.

4.2 Results of study two

We report our results of study two outlined in Section 3.6 for all eight factor combinations of number of slots, variance of no-show probability, and predicted probability variance, where, for each factor combination there are 50 ground truth probability vectors, each estimated with 100 probabilistic classifiers.

The average rank correlation between the clinic cost obtained and each factor combination is listed in Table 5. The greatest correlation in each row is depicted in bold-face type. For six out of the eight combinations, the correlation between expected schedule cost and Log Loss is the greatest. The correlation between cost and AUC is never the greatest. These results suggest that the AUC of the predicted no-show probabilities is not the best indicator of the schedule cost obtained using those no-show probabilities.

Table 5: Average spearman rank correlation between each performance metric and the schedule cost for each parameter combination

			Average Rank Correlation		
N, F	Variance of No-show Probability	Predicted Probability Variance	Cost vs AUC	Cost vs Brier Score	Cost vs Log Loss
7, 4	0.01	0.01	0.413	0.586	**0.618**
	0.01	0.02	0.258	0.572	**0.637**
	0.02	0.01	0.312	0.553	**0.623**
	0.02	0.02	0.247	0.538	**0.628**
7, 5	0.01	0.01	0.306	0.619	**0.696**
	0.01	0.02	0.233	0.591	**0.652**
	0.02	0.01	0.387	**0.573**	0.545
	0.02	0.02	0.347	**0.564**	0.473

Figure 4 depicts side-by-side box plots of the correlations for each parameter combination for all 50 iterations of the experiment. All of the correlations between cost and Brier score are positive, which

indicates that as a classifier's Brier score improves (its rank decreases), the cost of the schedule also improves (decreases). For both N, F combinations, there is a similar relationship among the correlations. When the variance of the predicted no-show probabilities is greater, the correlations for each metric are more dispersed, but the mean and median of the AUC correlations is always less than the Brier score and Log Loss. The predicted probability variance is either 0.01 or 0.02, with a lesser variance indicating a more precise prediction. Thus, we would assume classifiers with a predicted probability variance of 0.01 to have a greater average correlation with cost than the probabilities with a variance of 0.02. This is true for the majority of combinations. The exceptions occur for $N, F = 7, 4$ for the Log Loss metric. Figure 4 provides visual evidence that the average correlation of AUC and cost for all parameter combinations is less than that of the average correlation of Brier score and Log Loss with cost.

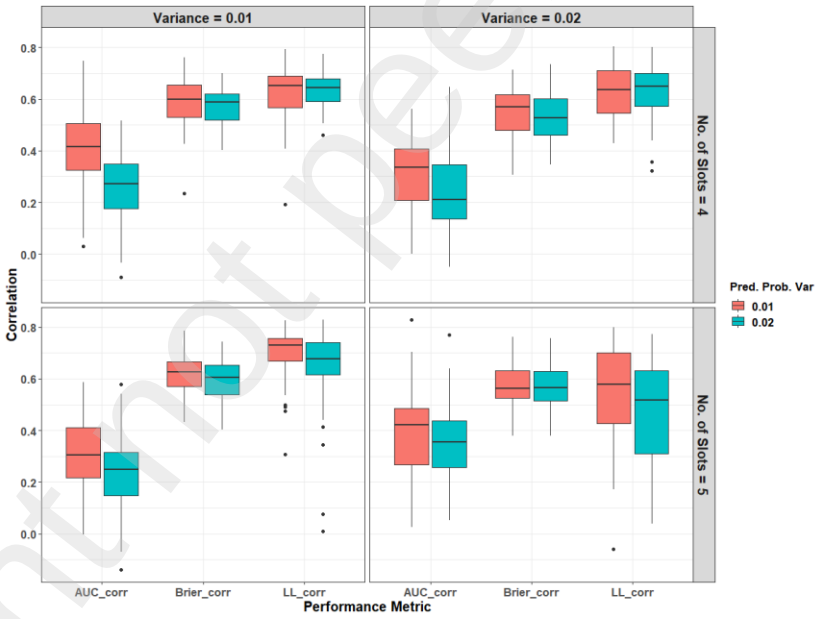

Figure 4: Side-by-Side box plots of the spearman rank correlations for each factor combination

Similar to study one, we conducted a One-Way repeated measures ANOVA with metric type as a factor and the three performance metrics as the factor levels, or treatments. The results are found to be consistent with non-parametric tests. The *p*-values of all statistical tests can be found in Appendix B.

Table 6: One-Way repeated measures ANOVA results and paired, multiple-comparisons analysis with 95% Bonferroni CIs for study two

				Multiple Comparison with Bonferroni Correction		
N, F	Variance of No-show Probability	Predicted Probability Variance	ANOVA p-value	Comparison	Difference	95% CI
7, 4	0.01	0.01		Brier – Log Loss	-0.032	**(-0.046, -0.018)**
7, 4	0.01	0.01	< 0.0000	Brier – AUC	0.173	**(0.121, 0.224)**
7, 4	0.01	0.01		Log Loss – AUC	0.205	**(0.143, 0.267)**
7, 4	0.01	0.02		Brier – Log Loss	-0.065	**(-0.081, -0.048)**
7, 4	0.01	0.02	< 0.0000	Brier – AUC	0.314	**(0.275, 0.354)**
7, 4	0.01	0.02		Log Loss – AUC	0.379	**(0.329, 0.429)**
7, 4	0.02	0.01		Brier – Log Loss	-0.071	**(-0.087, -0.054)**
7, 4	0.02	0.01	< 0.0000	Brier – AUC	0.241	**(0.204, 0.277)**
7, 4	0.02	0.01		Log Loss – AUC	0.311	**(0.263, 0.36)**
7, 4	0.02	0.02		Brier – Log Loss	-0.090	**(-0.12, -0.059)**
7, 4	0.02	0.02	< 0.0000	Brier – AUC	0.292	**(0.253, 0.33)**
7, 4	0.02	0.02		Log Loss – AUC	0.381	**(0.319, 0.444)**
7, 5	0.01	0.01		Brier – Log Loss	-0.077	**(-0.106, -0.048)**
7, 5	0.01	0.01	< 0.0000	Brier – AUC	0.312	**(0.271, 0.354)**
7, 5	0.01	0.01		Log Loss – AUC	0.389	**(0.331, 0.447)**
7, 5	0.01	0.02		Brier – Log Loss	-0.061	**(-0.11, -0.012)**
7, 5	0.01	0.02	< 0.0000	Brier – AUC	0.358	**(0.321, 0.396)**
7, 5	0.01	0.02		Log Loss – AUC	0.419	**(0.341, 0.497)**
7, 5	0.02	0.01		Brier – Log Loss	0.028	(-0.022, 0.078)
7, 5	0.02	0.01	< 0.0000	Brier – AUC	0.186	**(0.141, 0.231)**
7, 5	0.02	0.01		Log Loss – AUC	0.158	**(0.075, 0.241)**
7, 5	0.02	0.02		Brier – Log Loss	0.090	**(0.025, 0.156)**
7, 5	0.02	0.02	< 0.0000	Brier – AUC	0.217	**(0.18, 0.253)**
7, 5	0.02	0.02		Log Loss – AUC	0.126	**(0.042, 0.21)**

The results of the ANOVA along with the multiple-comparisons tests are listed in Table 6. The statistically significant 95% Bonferroni-adjusted confidence intervals (CIs) are depicted in bold type. For all parameter combinations, at least one correlation is found to be statistically different with the p-value of each ANOVA test being < 0.0001. The average AUC correlation is always statistically less than the Brier score correlation and the Log Loss correlation. Thus, Brier score and Log Loss are more highly correlated with the cost of a schedule than AUC. The difference between Brier score and Log Loss is statistically significant for all but one parameter combinations. When it is statistically significant, the correlation between Log Loss and cost is greater than the correlation between Brier score and cost, except for when $N, F = 7,5$ and both variances equal 0.02.

The results of both of our studies indicate that Brier score or Log Loss should be used to select a probabilistic classifier that is used as input to a scheduling model as part of executing the predicting overbooking framework. Our results are based upon ranking AUC, Brier score, and Log Loss values across probabilistic classifiers. The ranking of the classifiers within each metric does not take into account statistically significant differences among the values. Because of this, we ran our studies rounding the metric values to two and three decimal places. That rounding would cause values that were "close" to now be equal in rank. After re-running both studies with the rounded metrics, we found consistent results for both studies.

5 Conclusions, limitations, and future research directions

5.1 Conclusions

In this paper, we sought to answer the question of which performance metric should be used to select a probabilistic classifier that estimates the no-show probabilities used as inputs to a scheduling model. The insights from this paper can assist analysts in healthcare clinics make better decisions about the inputs to their scheduling models, and therefore, create a schedule that will support patients in a more efficient manner. To achieve our objective, we design a schedule optimizer which takes as input the individual no-show probabilities of appointment requests, and schedules the requests into appointment slots. Using this model, we conduct two computational experiments – one with real-world data and one with generated data –to analyze three performance metrics – AUC, Brier score, and Log Loss – and determine which metric leads to the best clinic performance, as measured by the lowest schedule cost.

The quality of no-show predictions in healthcare scheduling is typically measured with the AUC metric. The results of both of our studies indicate that Brier score or Log Loss are the preferred performance metrics when choosing a probabilitistic classifier to predict patient no-show probabilities. Both metrics are more highly correlated with lower schedule cost, as compared to AUC. When Brier score and Log Loss were statistically different, Log Loss was preferred over Brier score by a margin. Additionally, we found that the cost of a schedule when Brier score or Log loss are used to choose a classifier can be

4.5% less than the cost of a schedule when AUC is used to choose a classifier for models we solved without a heuristic, and 7% for larger scheduling models with required a heuristic. The choice of probabilistic classifier is an important one, as the probabilities generated by the classifier drive the scheduling model. The superiority of Brier score and Log Loss over AUC suggests that accurately predicting the magnitude of the appointment requests' no-show probabilities is more important than accurately ranking them by their risk of no-show. An additional contribution is our enhanced SCHED-OPT model which leads to dramatic reductions in solution times for large scheduling problems (up to 99.8%). That time reduction makes its adoption suitable in a real-time implementation.

5.2 Limitations

The task of scheduling patients under uncertainty is a difficult one, and in this paper, we shed light on how to make that process more efficient. Nonetheless, our research has its limitations. One of the limitations of our model lies in the need to solve larger problems using a heuristic. While the addition of Constraint set (5) allowed us to solve large scheduling problems more quickly, it has its drawbacks in not allowing for all probabilities to be introduced individually. An additional limitation is the use of a dataset that did not contain factors that have been found to influence no-shows such as new/established patients.

5.3 Future research directions

There are several other topics that stem from this study that can be considered as future work to continue contributing to healthcare scheduling and planning. First, while we demonstrated that producing no-show probabilities that are numerically inaccurate is detrimental to schedule quality, we have not investigated whether it is more detrimental to underestimating or overestimating these probabilities. While it is important to identify patients who will no-show due to the wasted resources attributed to no-shows [28], it is also costly to improperly identity a patient as a no-show and have the patient show up for the appointment. The trade-off of between over- and underestimating should be carefully evaluated by a clinic, while considering the cost of patient waiting time and clinic overtime. Second, because there are several performance metrics available to an analyst (AUC, Brier's score, Log Loss, etc), using just one may not be

preferable; future work may focus on developing a model that incorporates several different metrics in choosing a classifier. A third area of future work to consider would be adding more complexities to the scheduling problem, such as different patient priorities, stochastic service times, or unpunctuality, to determine how those factors affect the choice of performance metric. While a schedule may be analytically optimal, if a patient is visiting a physician and requires additional time, the schedule should also be able to adapt to those additional factors that affect the medical delivery model.

Funding

Michele Samorani's research was partly funded by the 2020 Leavey Research Grant.

References

[1] Dantas, L.F., Fleck, J.L., Oliveira, F.L.C. and Hamacher, S., 2018. No-shows in appointment scheduling–a systematic literature review. *Health Policy*, *122*(4), pp.412-421.

[2] Muthuraman, K., & Lawley, M. (2008). A stochastic overbooking model for outpatient clinical scheduling with no-shows. *IIE Transactions*, *40*(9), 820-837.

[3] Glowacka, K. J., Henry, R. M., & May, J. H. (2009). A hybrid data mining/simulation approach for modelling outpatient no-shows in clinic scheduling. *Journal of the Operational Research Society*, *60*(8), 1056-1068.

[4] LaGanga, L. R., & Lawrence, S. R. (2012). Appointment overbooking in health care clinics to improve patient service and clinic performance. *Production and Operations Management*, *21*(5), 874-888.

[5] Robinson, L. W., & Chen, R. R. (2010). A comparison of traditional and open-access policies for appointment scheduling. *Manufacturing & Service Operations Management*, *12*(2), 330-346.

[6] Zacharias, C., & Pinedo, M. (2014). Appointment scheduling with no-shows and overbooking. *Production and Operations Management*, *23*(5), 788-801.

[7] Li, Y., Tang, S. Y., Johnson, J., & Lubarsky, D. A. (2019). Individualized No-show Predictions: Effect on Clinic Overbooking and Appointment Reminders. *Production and Operations Management*.

[8] Samorani, M., & Harris, S. (2019, September). The Impact of Probabilistic Classifiers on Appointment Scheduling with No-Shows. In *Fortieth International Conference on Information Systems, Munich*.

[9] Dove, H. G., & Schneider, K. C. (1981). The usefulness of patients' individual characteristics in predicting no-shows in outpatient clinics. *Medical Care*, 734-740.

[10] Gallucci, G., Swartz, W., & Hackerman, F. (2005). Impact of the wait for an initial appointment on the rate of kept appointments at a mental health center. *Psychiatric Services*, *56*(3), 344-346.

[11] Goffman, R. M., Harris, S. L., May, J. H., Milicevic, A. S., Monte, R. J., Myaskovsky, L., ... & Vargas, D. L. (2017). Modeling patient no-show history and predicting future outpatient appointment behavior in the veterans health administration. *Military medicine*, *182*(5-6), e1708-e1714.

[12] Harris, S. L., May, J. H., & Vargas, L. G. (2016). Predictive analytics model for healthcare planning and scheduling. *European Journal of Operational Research*, *253*(1), 121-131.

[13] Whittle, J., Schectman, G., Lu, N., Baar, B., & Mayo-Smith, M. F. (2008). Relationship of scheduling interval to missed and cancelled clinic appointments. *The Journal of ambulatory care management*, *31*(4), 290-302.

[14] Chen, L., Li, X., Yang, Y., Kurniawati, H., Sheng, Q. Z., Hu, H. Y., & Huang, N. (2016). Personal health indexing based on medical examinations: A data mining approach. *Decision Support Systems*, *81*, 54-65.

[15] Dag, A., Oztekin, A., Yucel, A., Bulur, S., & Megahed, F. M. (2017). Predicting heart transplantation outcomes through data analytics. *Decision Support Systems*, *94*, 42-52.

[16] Daggy, J., Lawley, M., Willis, D., Thayer, D., Suelzer, C., DeLaurentis, P. C., ... & Sands, L. (2010). Using no-show modeling to improve clinic performance. *Health informatics journal*, *16*(4), 246-259.

[17] Alaeddini, A., Yang, K., Reeves, P., & Reddy, C. K. (2015). A hybrid prediction model for no-shows and cancellations of outpatient appointments. *IIE Transactions on healthcare systems engineering*, *5*(1), 14-32.

[18] Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

[19] Berry, M. J., & Linoff, G. S. (2004). *Data mining techniques: for marketing, sales, and customer relationship management*. John Wiley & Sons.

[20] Niculescu-Mizil, A., & Caruana, R. (2005, August). Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning* (pp. 625-632). ACM.

[21] Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, *10*(3), 61-74.

[22] Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly weather review*, *78*(1), 1-3.

[23] Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

[24] Ahmadi-Javid, A., Jalali, Z., & Klassen, K. J. (2017). Outpatient appointment systems in healthcare: A review of optimization studies. *European Journal of Operational Research*, *258*(1), 3-34.

[25] Mauchly, J. W. (1940). Significance test for sphericity of a normal *n*-variate distribution. *The Annals of Mathematical Statistics*, *11*(2), 204-209.

[26] Johnson, R. A., & Wichern, D. W. (2002). *Applied multivariate statistical analysis* (Vol. 5, No. 8). Upper Saddle River, NJ: Prentice hall.

[27] He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, *21*(9), 1263-1284.

[28] Nasir, M., Summerfield, N., Dag, A., & Oztekin, A. (2020). A service analytic approach to studying patient no-shows. *Service Business*, *14*(2), 287-313.

[29] Mohammadi, I., Wu, H., Turkcan, A., Toscos, T., & Doebbeling, B. N. (2018). Data analytics and modeling for appointment no-show in community health centers. *Journal of primary care & community health*, *9*, 2150132718811692.

[30] Simsek, S., Dag, A., Tiahrt, T., & Oztekin, A. (2020). A Bayesian Belief Network-based probabilistic mechanism to determine patient no-show risk categories. *Omega*, 102296.

[31] Ferro, D. B., Brailsford, S., Bravo, C., & Smith, H. (2020). Improving healthcare access management by predicting patient no-show behaviour. *Decision Support Systems*, *138*, 113398.

[32] Topuz, K., Uner, H., Oztekin, A., & Yildirim, M. B. (2018). Predicting pediatric clinic no-shows: a decision analytic framework using elastic net and Bayesian belief network. *Annals of Operations Research*, *263*(1-2), 479-499.

[33] Simsek, S., Tiahrt, T., & Dag, A. (2020). Stratifying no-show patients into multiple risk groups via a holistic data analytics-based framework. *Decision Support Systems*, 113269.

Appendices

Appendix A – Additional Statistical Analyses for Study One

Table A.1: Shapiro-Wilk Test for Normality Results for Study One

N, *F*	*p*-value
7, 4	<0.0000
7, 5	<0.0000
10, 7	<0.0000
12, 8	<0.0000

Table A.2: Mauchly's Test for Sphericity Results for Study One

N, *F*	*p*-value
7, 4	<0.0000
7, 5	<0.0000
10, 7	<0.0000
12, 8	<0.0000

Table A.3: Friedman's Rank Sum Test Results for Study One

N, *F*	*p*-value
7, 4	<0.0000
7, 5	<0.0000
10, 7	0.0002
12, 8	0.0009

Table A.4: Wilcoxon Signed-Rank Test Results for Study One

N, *F*	Comparison	*p*-value
7, 4	Brier – Log Loss	0.009
7, 4	Brier – AUC	0.000
7, 4	Log Loss – AUC	0.000
7, 5	Brier – Log Loss	0.007
7, 5	Brier – AUC	0.000
7, 5	Log Loss – AUC	0.000
10, 7	Brier – Log Loss	0.085
10, 7	Brier – AUC	0.000
10, 7	Log Loss – AUC	0.000
12, 8	Brier – Log Loss	0.000
12, 8	Brier – AUC	0.004
12, 8	Log Loss – AUC	0.040

Appendix B – Additional Statistical Analyses for Study Two

Table B.1: Shapiro-Wilk Test for Normality Results for Study Two

		Predicted Probability	
No. of Slots	Variance	Variance	*p*-value
4	0.01	0.01	0.001
4	0.01	0.02	0.005
4	0.02	0.01	**0.260**
4	0.02	0.02	**0.350**
5	0.01	0.01	0.003
5	0.01	0.02	0.000
5	0.02	0.01	0.028
5	0.02	0.02	**0.944**

Table B.2: Mauchly's Test for Sphericity Results for Study Two

		Predicted Probability	
No. of Slots	Variance	Variance	*p*-value
4	0.01	0.01	<0.0000
4	0.01	0.02	<0.0000
4	0.02	0.01	<0.0000
4	0.02	0.02	<0.0000
5	0.01	0.01	<0.0000

5	0.01	0.02	<0.0000
5	0.02	0.01	<0.0000
5	0.02	0.02	<0.0000

Table B.3: Friedman's Rank Sum Test Results for Study Two

No. of Slots	Variance	Predicted Probability Variance	*p*-value
4	0.01	0.01	<0.0000
4	0.01	0.02	<0.0000
4	0.02	0.01	<0.0000
4	0.02	0.02	<0.0000
5	0.01	0.01	<0.0000
5	0.01	0.02	<0.0000
5	0.02	0.01	<0.0000
5	0.02	0.02	<0.0000

Table B.4: Wilcoxon Signed-Rank Test Results for Study Two

No. of Slots	Variance	Predicted Probability Variance	Comparison	*p*-value
4	0.01	0.01	Brier – Log Loss	**0.000**
4	0.01	0.01	Brier – AUC	**0.000**
4	0.01	0.01	Log Loss – AUC	**0.000**
4	0.01	0.02	Brier – Log Loss	**0.000**
4	0.01	0.02	Brier – AUC	**0.000**
4	0.01	0.02	Log Loss – AUC	**0.000**
4	0.02	0.01	Brier – Log Loss	**0.000**
4	0.02	0.01	Brier – AUC	**0.000**
4	0.02	0.01	Log Loss – AUC	**0.000**
4	0.02	0.02	Brier – Log Loss	**0.000**
4	0.02	0.02	Brier – AUC	**0.000**
4	0.02	0.02	Log Loss – AUC	**0.000**
5	0.01	0.01	Brier – Log Loss	**0.000**
5	0.01	0.01	Brier – AUC	**0.000**
5	0.01	0.01	Log Loss – AUC	**0.000**
5	0.01	0.02	Brier – Log Loss	**0.001**
5	0.01	0.02	Brier – AUC	**0.000**
5	0.01	0.02	Log Loss – AUC	**0.000**
5	0.02	0.01	Brier – Log Loss	1.000
5	0.02	0.01	Brier – AUC	**0.000**
5	0.02	0.01	Log Loss – AUC	**0.003**
5	0.02	0.02	Brier – Log Loss	0.077
5	0.02	0.02	Brier – AUC	**0.000**
5	0.02	0.02	Log Loss – AUC	**0.018**