Optimizing Clinical Early Warning Models to Meet False Alarm Constraints

Preetish Rath [1] **Michael C. Hughes** [1]

Abstract

Deployed early warning systems in clinical settings often suffer from high false alarm rates that limit trustworthiness and overall utility. Despite the need to control false alarms, the dominant classifier training paradigm remains minimizing cross entropy, a loss function that has no direct relationship to false alarms. While existing efforts often use post-hoc threshold selection to address false alarms, in this paper we build on recent work to suggest a more comprehensive solution. We develop a family of tight bounds using the sigmoid function that let us maximize recall while satisfying a constraint that holds false alarms below a specified tolerance. This new differentiable objective can be easily integrated with generalized linear models, neural networks, and any other classifier trained with minibatch gradient descent. Through experiments on toy data and acute care mortality risk prediction, we demonstrate our method can satisfy a desired constraint on false alarms interpretable to clinical staff while achieving better recall than alternatives.

1. Introduction

Recent progress in machine learning has led to several promising automated systems designed to produce early warning alerts in critical care hospital settings via near real-time processing of vital signs and laboratory events (Hyland et al., 2020; Sendak et al., 2020b; Wellner et al., 2017).

Alerts are intended to indicate at-risk patients and trigger additional attention from clinical staff, who can assess the patient and possibly provide needed interventions. Within the high-demand setting of a modern intensive care unit (ICU), clinical staff have limited time that could be put to many productive uses; it is critical that automated alerts properly identify patients who need help and do not cause staff to focus on patients who do not need further attention.

[1]Department of Computer Science, Tufts University, Medford, MA, USA. Correspondence to: P.R. <preetish.rath@tufts.edu>.

Accepted Paper at the *1st Workshop on Interpretable Machine Learning in Healthcare (IMLH)*, co-located with ICML 2021.

An unneeded alert, also known as a *false alarm* or a *false positive*, has two detrimental consequences. In the moment, it pulls resources away from where they could be better used. In the long term, too many false alarms can cause clinical staff to distrust the alerts all together, a phenomenon known as *alarm fatigue* (Cvach, 2012; Deb & Claudio, 2015; Sendelbach & Funk, 2013). Some false alarms are inevitable, especially in many clinical tasks where the adverse outcome to predict is rare. For example, for the mortality risk models we study later, only about 9% of patients die in the hospital. It is critical that alert systems are designed to avoid alarm fatigue and ensure the tool has an overall net positive benefit. If the tool fails to limit false alarms to an acceptable rate, the tool may be ignored completely.

The key challenge of designing alert systems is thus to *balance* the false alarm rate (related to *precision*) with the true positive rate (known as *recall*) (Romero-Brufau et al., 2015). In the clinic, recall measures the fraction of truly at-risk patients correctly identified by an alert. Unfortunately, common objectives for training binary classifiers, such as cross entropy, are not designed to limit false alarms. Many early warning systems (Hyland et al., 2020; Futoma et al., 2017) train using standard objectives and only balance false alarm concerns in a secondary threshold selection or early stopping stage *after* training. Such post-hoc adjustment is limited in scope and may not identify the ideal tradeoff between recall and precision. Our experiments show the deficiency of these objectives even with post-hoc adjustment.

To overcome this challenge, in this paper we show that any binary classifier trained via stochastic gradient descent can, via a simple change to its loss function, be directly steered toward solutions that directly limit false alarms while maximizing recall. Put simply, our objective seeks to maximize the number of truly at-risk patients who are helped by alerts (maximize recall), subject to meeting an interpretable guarantee on false alarm rates: ensure less than $(1 - \alpha)\%$ of all alarms are false ones. We build on previous technical methods (Eban et al., 2017), but offer tighter bounds and more rigorous empirical validation in clinical settings.

This paper makes two key contributions:

1. Advocacy of max recall under a false alarm constraint as a good objective for clinical early warning systems. While this objective has been suggested as a possibil-

ity in other applications (Eban et al., 2017), we find that for clinical early warning systems it is particularly suitable yet underutilized. Our experiments suggest that our objective can lead to absolute gains of 0.1 - 0.15 in both recall and precision on heldout data in clinical settings (see Sec. 5.4 and 5.3). Our chosen objective is further *interpretable* to clinical staff, as they can naturally specify a maximum false alarm rate that would be tolerable in their daily practice.

2. New tractable bounds for optimizing recall while constraining false alarm rate. The false alarm rate itself is not a decomposible function that can be easily optimized via minibatch stochastic gradient descent (SGD) (see Sec. 4). Previous work by Eban et al. (2017) has suggested hinge-loss based bounds that are amenable to SGD. However, we find these bounds are too loose and lead to suboptimal performance. We derive tighter bounds based on the sigmoid function that are key to obtaining high-quality performance.

We view these contributions as critical steps to achieving early warning systems that not only achieve high-quality performance on heldout data, but also actually work when prospectively deployed.

Contributions to Interpretability. Overall, we take a broad view of the need for "interpretability" in early warning prediction systems, suggesting that trust by clinical staff is of course critically needed but this does not require "whitebox" understanding of all parts of a model. Instead, following Sendak et al. (2020a) we view our work as a way to build feedback loops with stakeholders by defining clear evaluation-based criteria needed to obtain trust (limiting false alarm rates below a desired limit).

2. Background

Our goal is to develop a prediction model with parameters θ that can consume a feature vector $x \in \mathbb{R}^D$ summarizing a patient's recent history in a clinical setting, and produce a real-valued score $f_\theta(x)$ indicating confidence that some (rare) outcome will occur. Large negative scores indicate certainty the outcome will not occur and large positive scores indicate certainty it will occur. Using a threshold b, we can translate this score into a *decision* $\hat{y}(x) = f_\theta(x) > b$. In the intended use case, a positive decision *alerts* clinical staff.

To train this model, we'll assume a dataset X, Y of N labeled examples: $X = \{x_n\}_{n=1}^N, Y = \{y_n\}_{n=1}^N$, with known binary label $y_n \in \{0, 1\}$ indicating which outcome happened to patient n with features $x_n \in \mathbb{R}^D$. In many cases, the outcome of interest (such as mortality or deterioriation) is rare. We'll denote the rare label of interest as "positive" or 1, and the common label as 0. Let N_+ denote the total number of positive true labels in a dataset: $N_+ = \sum_{n=1}^N y_n$. Similarly, N_- denotes the number of negative true labels.

2.1. Evaluation metrics for binary classifiers

Many performance metrics exist for binary classifiers, each appropriate for different goals (Romero-Brufau et al., 2015).

Binary cross entropy (BCE) is defined as:

$$\text{BCE}(\theta, X, Y) = \frac{1}{N} \sum_{n=1}^{N} \log \left(\sigma(f_\theta(x_n))^{y_n} (\sigma(-f_\theta(x_n)))^{1-y_n} \right)$$

Here, $\sigma(f) = \frac{1}{1+e^{-f}}$ denotes the logistic sigmoid function, which maps real-valued inputs $f \in \mathbb{R}$ to the unit interval $0 \leq \sigma(f) \leq 1$.

BCE is by far the most common loss used to train binary classifiers. It is well-motivated as a smooth, differentiable upper bound on error rate as well as via maximum likelihood arguments. However, for problems where the positive class is rare but critical, error rate alone will not capture the key applied questions, as explained below.

True positive count. A "true positive" is an example whose true label is positive and predicted label is also positive. We define a dataset's true positive count as:

$$\text{tpc}(\theta, X, Y) = \sum_{n:y_n=1} \hat{y}_\theta(x_n) = \sum_{n=1}^{N} z(y_n, f_\theta(x_n) - b).$$

Here, $z(\cdot, \cdot)$ is the zero-one function, which is one when both arguments have positive sign and zero otherwise. This count is bounded: $0 \leq \text{tpc} \leq N_+$.

False positive count. A "false positive" is an example whose true label is negative but whose predicted label is positive. False positives lead to false alarms in a clinical setting. We define a dataset's false positive count as:

$$\text{fpc}(\theta, X, Y) = \sum_{n:y_n=0} \hat{y}_\theta(x_n) = \sum_{n=1}^{N} z(1 - y_n, f_\theta(x_n) - b).$$

This count is also bounded: $0 \leq \text{fpc} \leq N_-$.

Recall. Recall is defined as the fraction of all examples that are truly positive which are correctly also called positive by the thresholded decision function $\hat{y}(\cdot) \in \{0, 1\}$.

$$\text{recall}(\theta, X, Y) = \frac{1}{N_+(Y)} \text{tpc}(\theta, X, Y), \qquad (1)$$

where $N_+(Y)$ counts all positive labels in training set Y.

Precision (aka True Alarm Rate). Precision is defined as the fraction of all positive alerts produced by the decision function that are truly positive. This is formalized as:

$$\text{prec}(\theta, X, Y) = \frac{1}{\text{tpc}(\theta, X, Y) + \text{fpc}(\theta, X, Y)} \text{tpc}(\theta, X, Y),$$

where the denominator counts all positive calls, which must either be true positive or false positive. Another name for precision is the "true alarm rate". Precision is equal to one minus the false alarm rate.

2.2. Suggested Optimization Objective

Consider a clinical prediction task trying to identify patients most at risk of a rare adverse outcome, so that additional interventions can be prioritized for these patients. Interventions are a limited resource with a cost (otherwise they could be applied to all patients). At minimum, hospital staff time is limited, and thus an alarm raised for a patient who does not need extra care means that other (perhaps more needy) patients do not get attention. Furthermore, if staff eventually notice that an alarm is rarely related to its intended outcome, they will likely be inclined to ignore it.

It is thus critical to view designing an effective prediction system for this setting as satisfying two goals. First, guaranteeing the false alarm rate is below some critical value established in conversation with care staff to ensure trust. Second, achieving an alert that would help the most possible patients given this constraint.

These goals naturally lead to this optimization problem:

$$\max_{\theta} \quad \text{recall}(\theta, X, Y), \quad \text{subj. to: } \text{prec}(\theta, X, Y) \geq \alpha, \quad (2)$$

where $1 - \alpha$ is set to the maximum allowed false alarm rate. This objective has been previously used (Eban et al., 2017). However, most ML early warning systems are not trained to satisfy some target minimum precision (or equivalently, maximum false alarm rate).

2.3. Baseline: Post-Hoc Threshold Search

After training a binary classifier to minimize cross entropy (or some other suitable objective), we can always perform a post-hoc search procedure to better identify a decision-making threshold b that meets desired performance criteria. In our applications, if we have a maximum allowable false alarm rate in mind, we can fix the parameterized score function $f_\theta(\cdot)$ obtained via training and simply select among a grid of candidate threshold values b the one that best satisfies the original objective in Eq. (2). Graphically, for a linear classifier this is akin to trying all possible decision boundary hyperplanes parallel to the one produced via original training. As a one-dimensional grid search, precision and recall at each threshold can be directly computed from validation data. No gradients or bounds are needed.

This approach is helpful but not optimal, as shown in the toy dataset analysis in Fig. 2. While the post-hoc search certainly improves recall compared to not performing the search, compared to our method it delivers sub-optimal precision (the 0.68 precision from threshold search is much worse than the 0.81 delivered by our method). If the target false alarm rate was 20% (meaning searching for precision above 0.8), there simply is no boundary parallel to the BCE-optimal boundary that can meet that standard. Thus, one-dimensional post-hoc search is inadequate, which we later further demonstrate on real clinical tasks.

3. Related Work

Optimization methods. Many method-development efforts have focused on optimization objectives that directly relate to the performance metric of interest (Rakotomamonjy, 2004; Burges et al., 2006; Yue et al., 2007; Lipton et al., 2014; Eban et al., 2017). However, most of these methods either pursue different objectives (less appropriate for our use case), have limited scalability, or have deficiencies in approximation quality. Metzler & Croft (2005) and Caruana & Niculescu-Mizil (2006) each propose methods of optimizing the area under the precision-recall curve directly. However, these rely on expensive sweeps across large parameter grids, and further cannot guarantee the quality of a specific decision threshold as our later method can. Joachims (2005) use contingency tables to optimize precision at fixed recall, but the cost of computing a single gradient is generally quadratic in the number of training examples. Fathony & Kolter (2020) propose an adversarial prediction framework that allows gradient-based training for a variety of non-decomposable objectives (such as precision at fixed recall). However, their method suffers from scalability issues (requiring quadratic runtime in the number of examples), and further requires a complex model that can sample data features as well as labels-given-features. Our work solves the direct, conceptually simpler problem of predicting labels from features well.

Our work builds upon the work of Eban et al. (2017), which developed a tractable framework for gradient-based learning of non-decomposable losses. Using hinge-loss functions to create bounds on the number of true and false positives, Eban et al. provide a solution for our intended objective in Eq. (2). However, we find in practice these hinge loss bounds are too loose (see Fig. 1). We develop tighter bounds with substantial gains in later experiments.

Clinical methods. Some false alarm control methods have been directly developed for acute care contexts (Chambrin (2001), Antink et al. (2016), Eerikäinen et al. (2016)). Au-Yeung et al. (2019) suggest a method to reduce false alarms in the ICU via post-hoc feature selection using random forest, but do not encode any limit on false alarms into the *training* objective. Hever et al. (2019) reduce false arrhythmia alarms in the ICU by training a random forests to match expert rules. However, such rules are not easily available for all tasks. Our work allows direct control of the false alarm rate and can be applied to large EHR datasets with multiple data sources.

4. Methods

We begin by restating that our ideal optimization objective given a training dataset $X, Y = \{x_n, y_n\}_{n=1}^N$ is to maximize recall subject to a minimum precision constraint as in

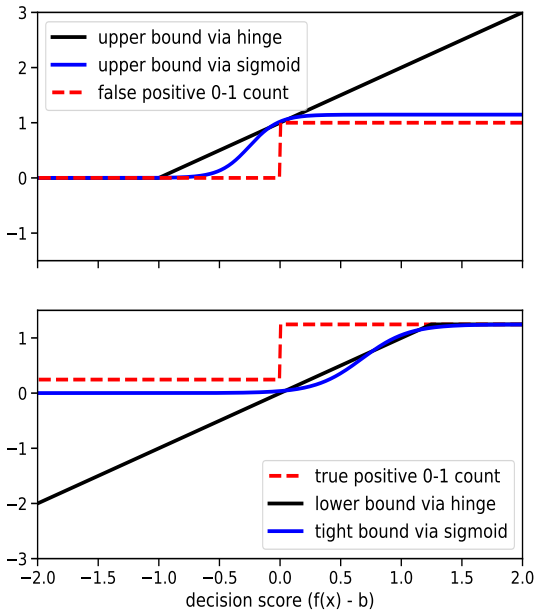

Figure 1. Top: Comparison between the hinge and sigmoid upper bounds for counting false positives. Our proposed sigmoid bound, highlighted in blue, is smooth, differentiable, and noticeably tighter than the hinge bound of Eban et al. (2017). *Bottom:* Comparison between the hinge and sigmoid lower bounds for counting true positives. Our proposed sigmoid bound is tighter and also non-negative, which is crucial to maintain the validity of our constraint on precision (as discussed around Eq. (4)).

Eq. (2).

While we can easily evaluate this objective for any candidate parameters θ, *training* our model – that is, finding good values for parameters θ^* given a dataset – is difficult. Not only is the optimization problem *constrained*, but the key barrier is that the functions involved are based on sums of the flat zero-one function (shown in Fig. 1) and thus have zero gradients at almost all θ values. We could use gradient-free methods, but these are inefficient when θ is high-dimensional as it will be in most realistic clinical settings involving many input variables. We thus need to transform this problem into one where modern efficient gradient-based learning will be effective.

Eban et al. (2017) suggest a promising route that leads to an unconstrained objective with non-zero gradients, which we review here and build upon. First, recognizing that maximizing recall is equivalent to maximizing the true positive count, can rewrite our problem in terms of true and false positive counts:

$$\max_{\theta} \text{tpc}(\theta), \quad \text{subj. to: } \frac{\text{tpc}(\theta)}{\text{tpc}(\theta) + \text{fpc}(\theta)} \geq \alpha. \tag{3}$$

Here for simplicity we write the counts as a function of parameters θ alone. While these counts do depend on the observed training data X and Y, we assume this data is known and fixed throughout.

Next, we can rewrite our problem in an equivalent "standardized" form by framing the constraint as a function that must be less than or equal to zero, and minimizing instead of maximizing:

$$\min_{\theta} \quad -\text{tpc}(\theta), \tag{4}$$

$$\text{subj. to:} \quad \underbrace{-\text{tpc}(\theta) + \frac{\alpha}{1-\alpha}\text{fpc}(\theta)}_{g(\theta)} \leq 0.$$

We emphasize that this transformation is only valid when the sum of tpc + fpc is strictly positive, as we need to multiply both sides of the constraint in Eq. (3) by this sum.

Using well-established optimization theory (Chong & Żak, 2013), we can transform this to an equivalent unconstrained optimization problem via the penalty method with Lagrange multiplier $\lambda > 0$:

$$\min_{\theta} -\text{tpc}(\theta, x, y) + \lambda g^+(\theta) \tag{5}$$

where $g^+(\theta)$ is a penalty function whose value is either zero or the function $g(\theta)$ defined in Eq. (4), whichever is larger. Naturally, if the penalty function is zero, this means the desired constraint is *satisfied*: the precision is at least α (and false alarm rate is less than $1 - \alpha$).

This unconstrained formulation avoids the need to handle a difficult non-linear constraint, but still faces the key problem that the objective as a function of θ is *flat*. That is, infinitesimal changes in θ are unlikely to move any single example's prediction from one side of the decision boundary to the other, and thus both tpc and fpc counts will remain the same for almost all possible small changes to θ. This flatness is a problem because it means any attempt at gradient-based training will not move the parameters θ from their original (poor performing) values.

4.1. Previous Tractable Bounds based on Hinge Loss

Eban et al. (2017) obtain tractability - informative non-zero gradients at almost all possible θ parameter values - for the objective in Eq. (5) by defining non-flat *bounds* on the counts using the hinge loss function $h(y, a) = \max(0, 1 - s(y)a)$, where $s(y)$ is the sign function, which returns $+1$ if $y = 1$ and -1 if $y = 0$, and $a \in \mathbb{R}$ is a real score.

First, the false positive count is upper bounded:

$$\text{fpc}^h(\theta) = \sum_{n:y_n=0} h(y_n, f_\theta(x_n)). \tag{6}$$

Similarly, the true positive count is lower bounded by:

$$\text{tpc}^h(\theta) = \sum_{n:y_n=1} 1 - h(y_n, f_\theta(x_n)). \tag{7}$$

Both bounds are visualized in Fig. 1. In these visuals, we can observe two key problems with the hinge loss bounds suggested by Eban et al. (2017). First, they are *loose* bounds: for just one term in the sum each bound can differ by 1 or more from the ideal value, this makes the total error (across

all N_- or N_+ terms in each sum) large. Second, the sum $\text{fpc}^h + \text{tpc}^h$ could be less than zero, as tpc^h may be negative. This condition would spoil the penalty function's interpretation as a valid bound on precision, since we derived it assuming the denominator in Eq. (3) was strictly positive.

4.2. New Tighter Bounds via Sigmoid Functions

We now derive two *families* of functions that bound the zero-one function, one family of upper bounds (for the false positive count) and one family of lower bounds (for the true postive count). These are illustrated in Fig. 1. Both bounds are found via a shifted and scaled version of the sigmoid function $\sigma(a) = \frac{1}{1+e^{-a}}$. Concrete bounds can be obtained by fixing tolerance hyperparameters to specific values. By varying these values, a user can make the bounds tighter (more accurate but with flatter gradients) or looser (more tractable). Even modest hyperparameter values deliver far tighter bounds than the loose hinge loss bounds of Eban et al. (2017). Furthermore, we'll show that under all conditions our bounds meet the denominator positivity condition required for our unconstrained objective to be properly limiting the false alarm rate.

Upper bound on fpc. We first seek an improved *upper bound* on the zero-one function, denoted $u(a)$. We want this smooth function to meet the following conditions necessary for a tight upper bound:

$$u(-\infty) \to 0 \qquad u(-\epsilon) \approx \delta$$
$$u(+\infty) \to 1 + \gamma\delta \qquad u(0) \approx 1 + \delta$$

where tolerance factors $\epsilon > 0, \delta > 0$ and scaling factor $\gamma > 1$ are specified by the user. Choosing large values makes the function smoother, choosing small values makes the bound tighter. In the limit as $\gamma \to 1, \epsilon \to 0, \delta \to 0$, our function will converge to the zero-one function.

A natural choice for this function to make it a sigmoid whose amplitude is $1+\gamma\delta$ with learnable horizontal shift and slope:

$$u_{m,b}(a) = (1 + \gamma\delta)\sigma(ma + b), \tag{8}$$

where the two parameters are slope $m \in \mathbb{R}$ and intercept $b \in \mathbb{R}$. By definition, this class of function satisfies the two limit conditions in Eq. (8). To meet the two approximation constraints, we can numerically solve for the optimal parameters that minimize squared error:

$$\hat{m}, \hat{b} = \underset{m\in\mathbb{R}, b\in\mathbb{R}}{\arg\min} (\delta - u_{m,b}(-\epsilon))^2 + (1 + \delta - u_{m,b}(0))^2.$$

Thus, for user-specified tolerance parameters γ, δ, ϵ, we can obtain a tight non-flat differentiable upper bound on the false positive count:

$$\text{fpc}^\sigma(\theta) = \textstyle\sum_{n=1}^{N}(1 + \gamma\delta)\sigma(\hat{m} f_\theta(x_n) + \hat{b}). \tag{9}$$

If we concretely select a modest setting of our tolerance parameters – $\gamma = 7.00$, $\delta = 0.021$, $\epsilon = 0.735$ – we use BFGS to solve the minimization problem and yield $\hat{m} = 8.26$ and $\hat{b} = 2.09$.

Lower bound on tpc. A challenge of defining a lower bound is that we need our lower bound to be strictly non-negative for all inputs, so that the denominator positivity condition in Eq. (3) is satisfied. To achieve this, let us first redefine the true positive count so that to each zero-one function we add a factor of $\tilde{\gamma}\tilde{\delta}$. This means each example's contribution is either $\{\tilde{\gamma}\tilde{\delta}, 1+\tilde{\gamma}\tilde{\delta}\}$ instead of $\{0, 1\}$. Shifting the utility of all positive examples *upward* by a positive constant should not impact any learned boundary, as the net gain for classifying each example correctly is still the same.

We thus seek a lower bound ℓ of a vertically-shifted zero-one function, meeting the conditions:

$$\ell(-\infty) \to 0 \qquad \ell(0) \approx \tilde{\delta}$$
$$\ell(+\infty) \to 1 + \tilde{\gamma}\tilde{\delta} \qquad \ell(+\tilde{\epsilon}) \approx 1 + \tilde{\delta}$$

Again tolerance factors $\tilde{\epsilon} > 0, \tilde{\delta} > 0, \tilde{\gamma} > 1$ are specified by the user, with similar interpretation as before.

We define again a horizontally-shifted-and-scaled sigmoid function whose amplitude is $1 + \tilde{\gamma}\tilde{\delta}$:

$$\ell_{\tilde{m},\tilde{b}}(a) = (1 + \tilde{\gamma}\tilde{\delta})\sigma(\tilde{m}a + \tilde{b}) \tag{10}$$

where the two parameters are slope $\tilde{m} \in \mathbb{R}$ and intercept $\tilde{b} \in \mathbb{R}$. Again, this function by definition solves the limit conditions, and we can numerically ensure the approximate conditions by solving for $\tilde{m}, \tilde{b}$ that minimize squared error:

$$\underset{\tilde{m}\in\mathbb{R}, \tilde{b}\in\mathbb{R}}{\arg\min} (\tilde{\delta} - \ell_{\tilde{m},\tilde{b}}(0))^2 + (1 + \tilde{\delta} - \ell_{\tilde{m},\tilde{b}}(\tilde{\epsilon}))^2$$

Thus, given tolerance parameters $\tilde{\gamma}, \tilde{\delta}, \tilde{\epsilon}$, we obtain a lower bound on the vertically-shifted true positive count:

$$\text{tpc}^\sigma(\theta) = \textstyle\sum_{n=1}^{N}(1 + \tilde{\gamma}\tilde{\delta})\sigma(\tilde{m} f_\theta(x_n) + \tilde{b}) \tag{11}$$

Concretely fixing tolerances to $\tilde{\gamma} = 7.00$, $\tilde{\delta} = 0.035$, $\tilde{\epsilon} = 0.99$, we obtain $\tilde{m} = 5.19$ and $\tilde{b} = -3.54$.

4.3. Practical Implementation

Our proposed bounds allow tractable gradient-based optimization of our intended maximum-recall-while-guaranteeing-precision objective in Eq. (2). Here we discuss key practical engineering efforts required to make our bounds useable on large clinical datasets.

Overcome local optima via many diverse initializations. Linear-boundary classifiers like logistic regression (minimizing BCE) or support vector machines (minimizing hinge loss), have *convex* training problems. However, due to our sigmoid bounds even when our classifier f_θ is a generalized linear model, our objective is *non-convex* and thus sensitive to initialization. To avoid poor local optima, we take the best of many random initializations (in terms of validation-set performance). While performing many runs and then keeping the best makes the runtime of training visibly higher than alternatives, we argue this is *worth it* in an overall cost-benefit analysis: our method yields substantial gains in

improved recall (patients for whom alerts would be beneficial). In most clinical settings training happens infrequently (once every few weeks or months).

In our experiments, we try two kinds of initialization. First, in "cold start" runs the weights are initialized via Glorot procedures (Glorot & Bengio, 2010). Second, "warm start" runs start at small perturbations of the optimal weights found by pretraining on a BCE objective. We typically try 25 initializations of each type.

Minibatch gradient descent. If datasets are small enough, we can directly optimize our unconstrained objective in Eq. (5) using gradients computed from all examples. However, to scale to large datasets we use stochastic estimates of the gradient from a *minibatch*. Estimating unbiased gradients of the g^+ function in Eq. (5) from a minibatch is difficult, because the maximum is performed outside the sum over all training examples. To address this, we simply select large minibatches (500 or more examples). We find that while not formally unbiased, in practice this choice is effective. We reach high quality solutions that meet the intended precision when assessed on the entire training set.

In practice, for large datasets (MIMIC and e-ICU) we use the ADAM algorithm with minibatches of size 512, 1024 and the full training set. For all methods, we try a range of learning rates (from 0.0005 to 0.005) and select the best performing one on the validation set.

Penalty strength hyperparameter selection. We must choose a value for penalty strength $\lambda > 0$. We found that trying two λ values, 1000 and 10000, was sufficient for the desired false alarm constraint to be satisfied (penalty function g evaluates to less than zero after training). We use these values for all experiments.

5. Experimental Evaluation

5.1. Results on Synthetic Data

To gain insight, we designed a toy classification task with 2 features where the differences between objectives can be easily visualized. Our goal is to train a logistic regression (LR) classifier whose decisions maximize recall subject to a maximum false alarm rate of 20% (minimum precision of $\alpha = 0.8$). Fig. 2 shows all training examples ($N_+ = 120, N_- = 450$) together with the learned decision boundaries for four ways of training the LR model. We try BCE with default threshold and BCE with threshold selected to satisfy our α-valued constraint (or get as close as possible). We further show Eban et al. (2017)'s hinge-bound approach as well as our proposed tighter sigmoid bounds.

Only our method can meet the desired precision constraint, all others do not achieve 0.8 precision even on the training data, instead producing noticeably worse precision values

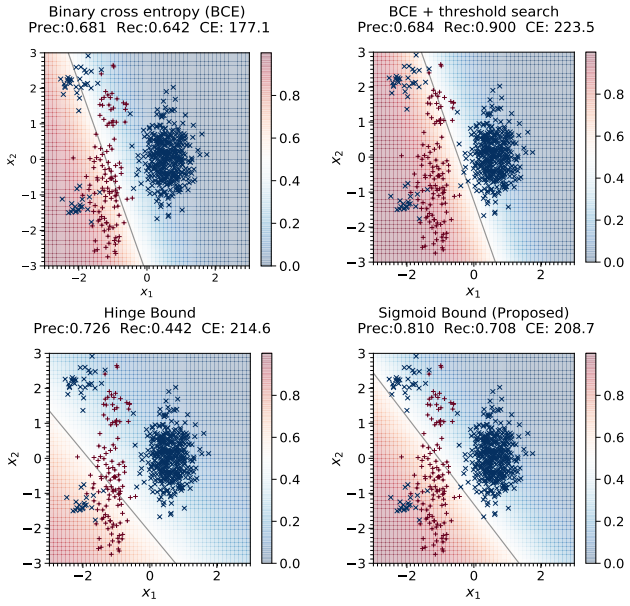

Figure 2. Linear decision boundaries found under various objectives on synthetic task in Sec. 5.1. Red and blue points represent positive and negative examples respectively. The black line represents the decision boundary. Our goal is to find a decision boundary that maximizes recall at a minimum precision of 0.8. We report the precision, recall and cross entropy (CE) for 4 different solutions. *Top*: BCE with default threshold and post-hoc selected threshold. Both are unable to exceed a precision of 0.684. *Bottom Left*: Optimizing the unconstrained objective with Eban et al. (2017)'s hinge bound also fails to meet the desired precision of 0.8 . *Bottom Right*: Our proposed sigmoid bound is the only one that satisfies the desired minimum precision requirement of 0.8, while simultaneously achieving better recall than the hinge bound.

of 0.68-0.72. Even though the hinge bound is trained to meet the target α value, the bound's looseness precludes an adequate solution. To further verify the quality of our bound, we perform an exhaustive search of possible LR parameters (two weights, one bias) to optimize our objective in Eq. (2). (This search is only feasible in low-dimensional problems.) Our sigmoid bound reached precision 0.81 and recall 0.70, which differs only slightly from the grid search optimum of precision 0.80 and recall 0.74, whereas the hinge bound only reaches a precision of 0.72 and recall 0.44.

5.2. Results on Semi-Synthetic Data

Clinical tasks often includes many features, most of which are only weakly relevant to the outcome. To assess our method's robustness to such features, we designed a *semi-synthetic* task. Building on the toy generation process that produced features x_n and labels y_n shown in Fig. 2, we append to each x_n vector a 98-dimensional feature vector representing vitals, labs, and demographics for one randomly-chosen patient-stay in the MIMIC-III dataset (Johnson et al. (2016)). To assess generalization, we split data into 684

Table 1. Comparison of precision and recall for semi-synthetic experiment. Among the 3 methods, only our proposed sigmoid bound finds a linear boundary that satisfies the 0.8 minimum precision requirement. Additionally, our method achieves better precision and recall on the heldout test set, implying better generalization.

Method	Precision			Recall		
	Train	Valid	Test	Train	Valid	Test
BCE + threshold search	0.70	0.69	0.69	0.61	0.72	0.61
Hinge Bound	0.63	0.61	0.63	**0.72**	0.75	0.71
Sigmoid Bound (Ours)	0.80	0.72	0.74	0.67	**0.77**	**0.76**

train, 513 valid, and 512 test examples. We stress that the labels remain the same, and our goal remains to reach minimum precision of $\alpha = 0.8$. To avoid local optima, we run our method many times (25 random seeds, 2 initialization scale factors, and 2 λ values). We select the best run that maximizes validation set recall while meeting the α constraint.

Table 1 shows that even in this high-dimensional problem (100 total features), only our method can meet the desired 0.8 precision (others range from 0.63-0.7). Our method also exceeds others on test-set recall by at least 0.05.

5.3. Results: Mortality Prediction on MIMIC

We assessed our method in an acute care setting using the MIMIC-III dataset. Our goal is to predict in-hospital mortality after observing the entire sequence of a patient's stay up to the last measurement discharge or death. We extract 2 demographics, 10 vitals, and 94 lab measurements discretized to hourly bins for 34472 patient stays using MIMIC Extract (Wang et al., 2020). Our train/valid./test splits yield 20682/6895/6895 patient-stays, with $\sim 9.5\%$ patient stays resulting in death. Each patient-stay's multivariate time-series is transformed into a feature vector as follows. For each raw feature, we apply 7 summarization functions (missing indicator; time since last non-missing; plus min, max, median, slope, and variance of non-missing values) to 4 possible time windows (0-100%, 50%-100%, last 16 hr, and last 24 hr). The resulting feature vector has size 106x7x4=2968.

Our training goal is to predict in-hospital mortality while limiting with false alarm rate less than 10% (precision $\alpha \geq 0.9$). We found this α challenging to meet on heldout data, so to select the best runs we seek validation set precision ≥ 0.8. We assessed both logistic regression (LR) models and multi-layer perceptrons (MLP) with 1 hidden layer (32 hidden units; RELU activation). To avoid local optima, we search 25 random seeds, 2 initialization scale factors (1.0 and 3.0), 2 λ values, and 5 weight decays.

Table 2 shows our tight sigmoid bounds satisfy the false alarm constraints while improving recall from 0.54 to 0.69 for LR models and from 0.69 to 0.72 for MLPs compared to BCE training. Moreover, our model consistently beats the hinge bound baseline.

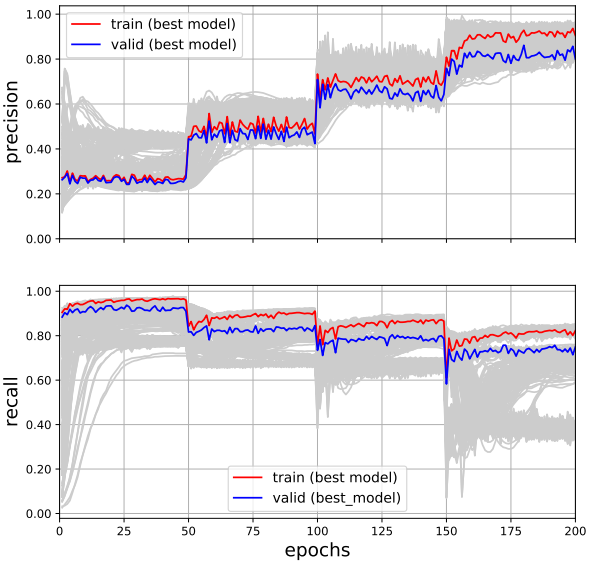

Figure 3. Evidence of successful false-alarm-averse training of a model for MIMIC mortality risk prediction. Every 50 epochs, the desired minimum precision α is stair-stepped up towards 0.9. After each step, the model's empirical precision is quickly improved to approximately satisfy the desired α via gradient-based learning. Each gray line is one run from a random initialization, the best performing run (maximizing recall while satisfying our precision goals on training and validation) is highlighted in color.

5.4. Results: Mortality Prediction on eICU

The eICU Collaborative Research Database (Pollard et al. (2018)) contains data from many critical care units throughout the U.S. We extract 3 demographics, 8 vitals, and 6 lab measurements discretized to hourly bins using eICU Extract (Wang et al. (2020)) for 72670 patient stays. We featurize each patient-stay the same way as MIMIC (see Sec. 5.3), yielding patient-stay vectors of size 17x7x4=417. Our train/valid/test split yields 43642/14509/14518 patient-stays, with $\sim 8.2\%$ resulting in death.

Our training goal is again to predict in-hospital mortality using the entire stay. Again, we enforce training set precision above $\alpha = 0.9$ on the training set (limiting false alarm rates to less than 10%), and enforce validation set precision above 0.8. We train the same LR and MLP models as before with the same hyperparameter grids (see Sec. 5.3).

In Table 3, we again find our tight sigmoid bounds satisfy the false alarm constraints while improving recall from 0.10 to 0.20 for LR and from 0.28 to 0.30 for MLP compared to standard BCE objectives. Moreover, our model again beats the hinge bound baseline soundly and consistently.

To better understand the gains of our method in terms of common evaluation criteria, in Appendix A, we compare the receiver operating curves and precision-recall curves of all logistic regression methods on the eICU dataset. Our

Table 2. Predicting in-hospital mortality on MIMIC-III, when the goal is to maximize recall at a minimum precision of $\alpha = 0.9$ (false alarm rate ≤ 0.1). While all methods meet the precision constraint, our sigmoid bound achieves better recall on heldout test data.

	Method	Precision			Recall			% of 1500 runs
		Train	Valid	Test	Train	Valid	Test	w/ prec. $\geq \alpha$
Logistic Regression	BCE + threshold search	0.90	0.84	0.86	0.57	0.59	0.54	-
	Hinge Bound (Eban et al.)	0.96	0.84	0.85	0.69	0.65	0.62	-
	Sigmoid Bound (warm init.)	0.90	0.86	0.86	0.73	0.71	0.67	14.5%
	Sigmoid Bound (cold init.)	0.90	0.84	0.85	**0.76**	**0.73**	**0.69**	3.1%
1-layer MLP	BCE + threshold search	0.90	0.83	0.84	0.76	0.72	0.69	-
	Hinge Bound (Eban et al.)	0.98	0.89	0.91	0.74	0.67	0.63	-
	Sigmoid Bound (warm init.)	0.91	0.86	0.86	0.76	0.72	0.68	15.0%
	Sigmoid Bound (cold init.)	0.92	0.83	0.82	**0.82**	**0.75**	**0.72**	55.3%

Table 3. Predicting in-hospital mortality on e-ICU, when the goal is to maximize recall at a minimum precision of $\alpha = 0.9$ (false alarm rate ≤ 0.1). While all methods meet the precision constraint, our sigmoid bound achieves better recall on heldout test data. For logistic regression gains are especially strong: recall of 0.20 for ours vs. 0.10 for post-hoc threshold search and 0.02 for the hinge bound.

	Method	Precision			Recall			% of 1500 runs
		Train	Valid	Test	Train	Valid	Test	w/ prec. $\geq \alpha$
Logistic Regression	BCE + threshold search	0.90	0.94	0.86	0.12	0.12	0.10	-
	Hinge Bound (Eban et al.)	0.86	0.75	0.80	0.02	0.02	0.02	-
	Sigmoid Bound (warm init.)	0.91	0.83	0.79	0.12	0.14	0.10	13%
	Sigmoid Bound (cold init.)	0.91	0.82	0.79	**0.21**	**0.20**	**0.20**	1%
1-layer MLP	BCE + threshold search	0.90	0.85	0.80	0.32	0.30	0.28	-
	Hinge Bound (Eban et al.)	0.99	0.83	0.77	0.36	0.25	0.23	-
	Sigmoid Bound (warm init.)	0.91	0.81	0.80	0.07	0.07	0.06	5%
	Sigmoid Bound (cold init.)	0.92	0.81	0.78	**0.38**	**0.33**	**0.30**	30%

method shows consistent gains in both curves. We do stress that measuring area under the ROC curve (as is commonly done) has little to do with the quality of the alerts produced by a specific operating threshold (Romero-Brufau et al., 2015).

6. Conclusion

Limitations. A key drawback of our method is the vulnerability to local optima due to non-convexity of the sigmoid bound. Given reasonable compute power at training time, we find we can overcome local optima by taking the best of many runs. For MLPs, cold starts using Glorot initialization (Glorot & Bengio, 2010) satisfy the false alarm contraint more often than warm starts from a BCE solution (see Tables 2-3).

Selection of the constraint value to enforce α is critical to success. While we selected α values in our experiments that seemed reasonable, in practice, α needs to be chosen in collaboration with experienced clinical staff. Our method's performance is sensitive to several other hyperparameters, including the penalty strength λ and the tolerance parameters γ, δ, ϵ that govern the tractability-tightness tradeoff for our bound. While we found reasonable values for our tasks, future applications may need to tune values for better performance.

Finally, although our model's goal is to achieve false alarms below a user-specified rate on the training set, it does not guarantee to satisfy this constraint on heldout datases (see Table 3). Improving generalization guarantees is an open research question.

Advantages. The main advantage of our method is the ability to enforce a maximum desired false alarm rate in acute care settings. This is visually demonstrated in Figure 3, where the desired α is stepped up every 50 epochs, and the empirical precision on heldout data responds accordingly. Given an acceptable false alarm rate, our objective clearly outperforms alternatives like post-hoc threshold search.

Our bounds can work with any classifier trained via SGD. Future work could use our bounds to train more flexible differentiable classifiers such as recurrent NNs, convolutional NNs, or graph NNs.

For improved interpretability, future work can explore the integration of our loss with penalities for weight sparsity using L1-norm (Tibshirani, 1996) or L0-norm (Ustun & Rudin, 2016) penalties.

References

Antink, C. H., Leonhardt, S., and Walter, M. Reducing false alarms in the icu by quantifying self-similarity of multimodal biosignals. *Physiological measurement*, 37 (8):1233, 2016.

Au-Yeung, W.-T. M., Sahani, A. K., Isselbacher, E. M., and Armoundas, A. A. Reduction of false alarms in the intensive care unit using an optimized machine learning based approach. *NPJ digital medicine*, 2(1):1–5, 2019.

Burges, C., Ragno, R., and Le, Q. Learning to rank with non-smooth cost functions. *Advances in neural information processing systems*, 19:193–200, 2006.

Caruana, R. and Niculescu-Mizil, A. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd International Conference on Machine Learning*, pp. 161–168, 2006.

Chambrin, M.-C. Alarms in the intensive care unit: how can the number of false alarms be reduced? *Critical Care*, 5 (4):1–5, 2001.

Chong, E. K. P. and Żak, S. H. Chapter 23: Algorithms for Constrained Optimization. In *An Introduction to Optimization*, Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, New York, fourth edition edition, 2013.

Cvach, M. Monitor alarm fatigue: an integrative review. *Biomedical instrumentation & technology*, 46(4):268–277, 2012.

Deb, S. and Claudio, D. Alarm fatigue and its influence on staff performance. *IIE Transactions on Healthcare Systems Engineering*, 5(3):183–196, 2015.

Eban, E., Schain, M., Mackey, A., Gordon, A., Rifkin, R., and Elidan, G. Scalable Learning of Non-Decomposable Objectives. In *Artificial Intelligence and Statistics*, pp. 832–840, 2017. URL http://proceedings.mlr.press/v54/eban17a.html.

Eerikäinen, L. M., Vanschoren, J., Rooijakkers, M. J., Vullings, R., and Aarts, R. M. Reduction of false arrhythmia alarms using signal selection and machine learning. *Physiological measurement*, 37(8):1204, 2016.

Fathony, R. and Kolter, Z. AP-perf: Incorporating generic performance metrics in differentiable learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 4130–4140. PMLR, 2020.

Futoma, J., Hariharan, S., and Heller, K. Learning to Detect Sepsis with a Multitask Gaussian Process RNN Classifier. In *International Conference on Machine Learning*, 2017. URL http://arxiv.org/abs/1706.04152.

Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256. JMLR Workshop and Conference Proceedings, 2010.

Hever, G., Cohen, L., O'Connor, M. F., Matot, I., Lerner, B., and Bitan, Y. Machine learning applied to multi-sensor information to reduce false alarm rate in the icu. *Journal of clinical monitoring and computing*, pp. 1–14, 2019.

Hyland, S. L., Faltys, M., Hüser, M., Lyu, X., Gumbsch, T., Esteban, C., Bock, C., Horn, M., Moor, M., et al. Early prediction of circulatory failure in the intensive care unit using machine learning. *Nature Medicine*, 26(3):364–373, 2020.

Joachims, T. A support vector method for multivariate performance measures. In *Proceedings of the 22nd International Conference on Machine Learning*, pp. 377–384, 2005.

Johnson, A. E., Pollard, T. J., Shen, L., Li-Wei, H. L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

Lipton, Z. C., Elkan, C., and Naryanaswamy, B. Optimal thresholding of classifiers to maximize F1 measure. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 225–239. Springer, 2014.

Metzler, D. and Croft, W. B. A markov random field model for term dependencies. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 472–479, 2005.

Pollard, T. J., Johnson, A. E., Raffa, J. D., Celi, L. A., Mark, R. G., and Badawi, O. The eicu collaborative research database, a freely available multi-center database for critical care research. *Scientific data*, 5(1):1–13, 2018.

Rakotomamonjy, A. Optimizing area under roc curve with SVMs. In *ROCAI*, pp. 71–80, 2004.

Romero-Brufau, S., Huddleston, J. M., Escobar, G. J., and Liebow, M. Why the C-statistic is not informative to evaluate early warning scores and what metrics to use. *Critical Care*, 19(1), 2015.

Sendak, M., Elish, M. C., Gao, M., Futoma, J., Ratliff, W., Nichols, M., Bedoya, A., Balu, S., O'Brien, C., et al. "The human body is a black box": Supporting clinical decision-making with deep learning. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 99–109, Barcelona Spain, 2020a. ACM.

Sendak, M. P., Ratliff, W., Sarro, D., Alderton, E., Futoma, J., Gao, M., Nichols, M., Revoir, M., Yashar, F., et al. Real-World Integration of a Sepsis Deep Learning Technology Into Routine Clinical Care: Implementation Study. *JMIR Medical Informatics*, 8(7), 2020b.

Sendelbach, S. and Funk, M. Alarm fatigue: a patient safety concern. *AACN advanced critical care*, 24(4):378–386, 2013.

Tibshirani, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Ustun, B. and Rudin, C. Supersparse linear integer models for optimized medical scoring systems. *Machine Learning*, 102(3):349–391, 2016.

Wang, S., McDermott, M. B., Chauhan, G., Ghassemi, M., Hughes, M. C., and Naumann, T. Mimic-extract: A data extraction, preprocessing, and representation pipeline for mimic-iii. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, pp. 222–235, 2020.

Wellner, B., Grand, J., Canzone, E., Coarr, M., Brady, P. W., Simmons, J., Kirkendall, E., Dean, N., Kleinman, M., et al. Predicting Unplanned Transfers to the Intensive Care Unit: A Machine Learning Approach Leveraging Diverse Clinical Elements. *JMIR Medical Informatics*, 5 (4), 2017.

Yue, Y., Finley, T., Radlinski, F., and Joachims, T. A support vector method for optimizing average precision. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 271–278, 2007.

A. Further Results

At the request of an anonymous reviewer, in Fig. A.1 we provide the full receiver operating curve and precision-recall curve for all possible objectives for training logistic regression on the eICU dataset. We see that our proposed method delivers noticeable gains especially in the precision-recall curve, which we argue is more suitable to early warning systems.

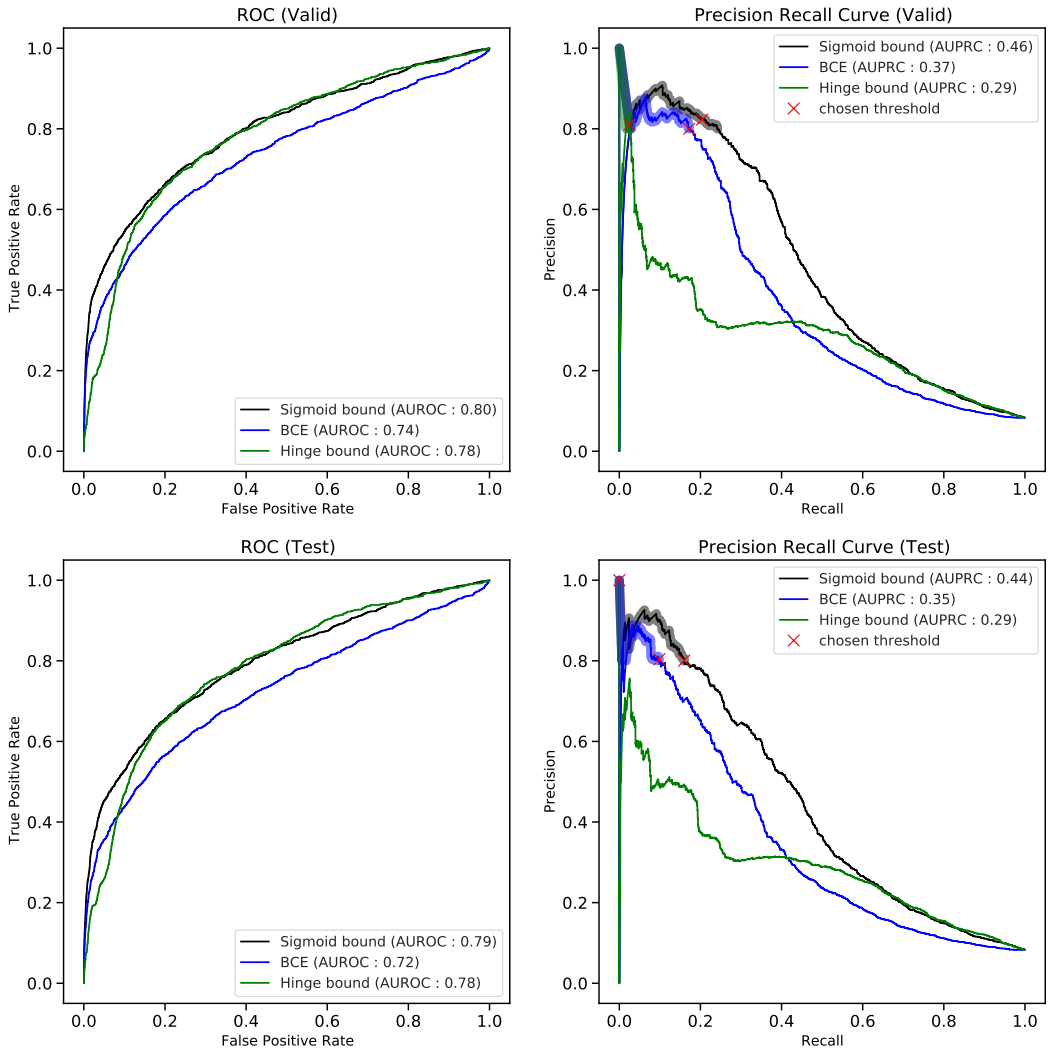

Figure A.1. Tradeoffs between the decisions produced by logistic regression models trained using different optimization objectives on the eICU dataset (top row: validation set; bottom row: test set). Each line traces out the performance of a single method while varying its decision threshold. *Left column:* Receiver operating curve (ROC), comparing true positive rate (TPR, y-axis) to false positive rate (FPR, x-axis). *Right column:* Precision-recall curve, comparing precision (also known as positive predictive value (PPV), y-axis) to recall (TPR, x-axis). Our sigmoid bound and Eban et al. (2017)'s hinge bound were trained to maximize recall subject to a constraint on precision: above 0.9 on training set; above 0.8 on validation set. The shaded regions for each method on the precision-recall curves denote thresholds satisfying precision above 0.8 on validation set. The selected operating point for each method (chosen to maximize our optimization objective on the validation set) is shown as a red cross on the precision-recall curves.