Optimizing Early Warning Classifiers to Control False Alarms via a Minimum Precision Constraint

Preetish Rath,

PhD Candidate,

Tufts University

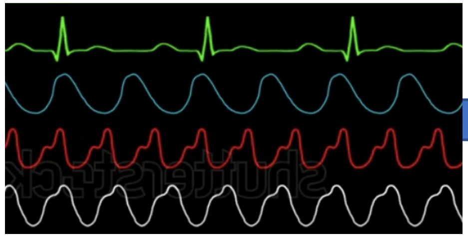
Michael C. Hughes

Assistant Professor,

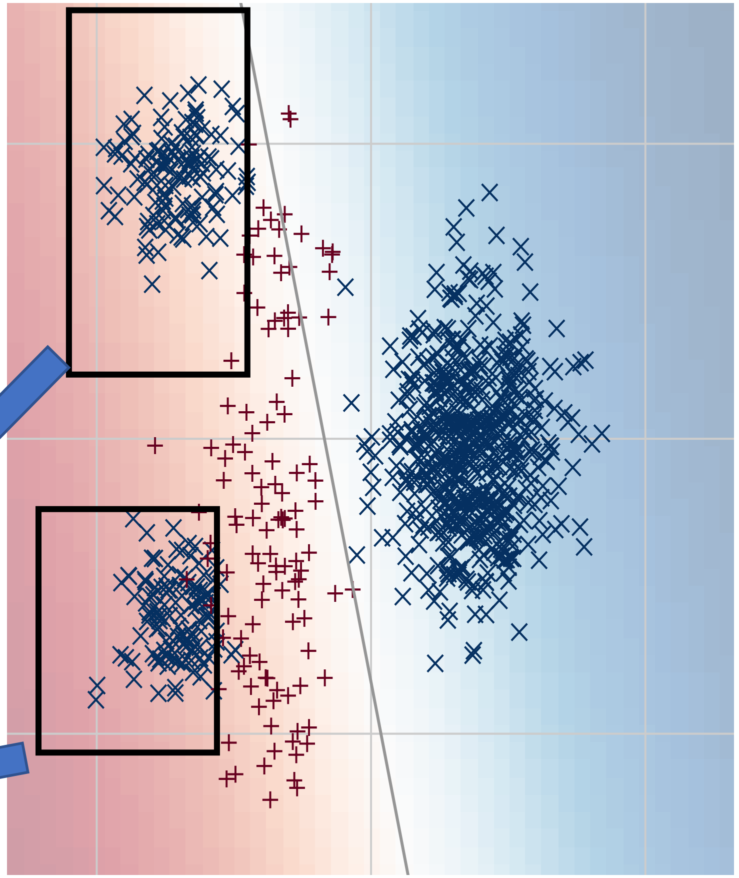
Tufts University

False Alarms in Early Warning Systems Have Severe Consequences

Motivating Application:
Monitoring patient vital signs in critical care



Goal: Binary classifier that continuously predicts risk of deterioration for all patients



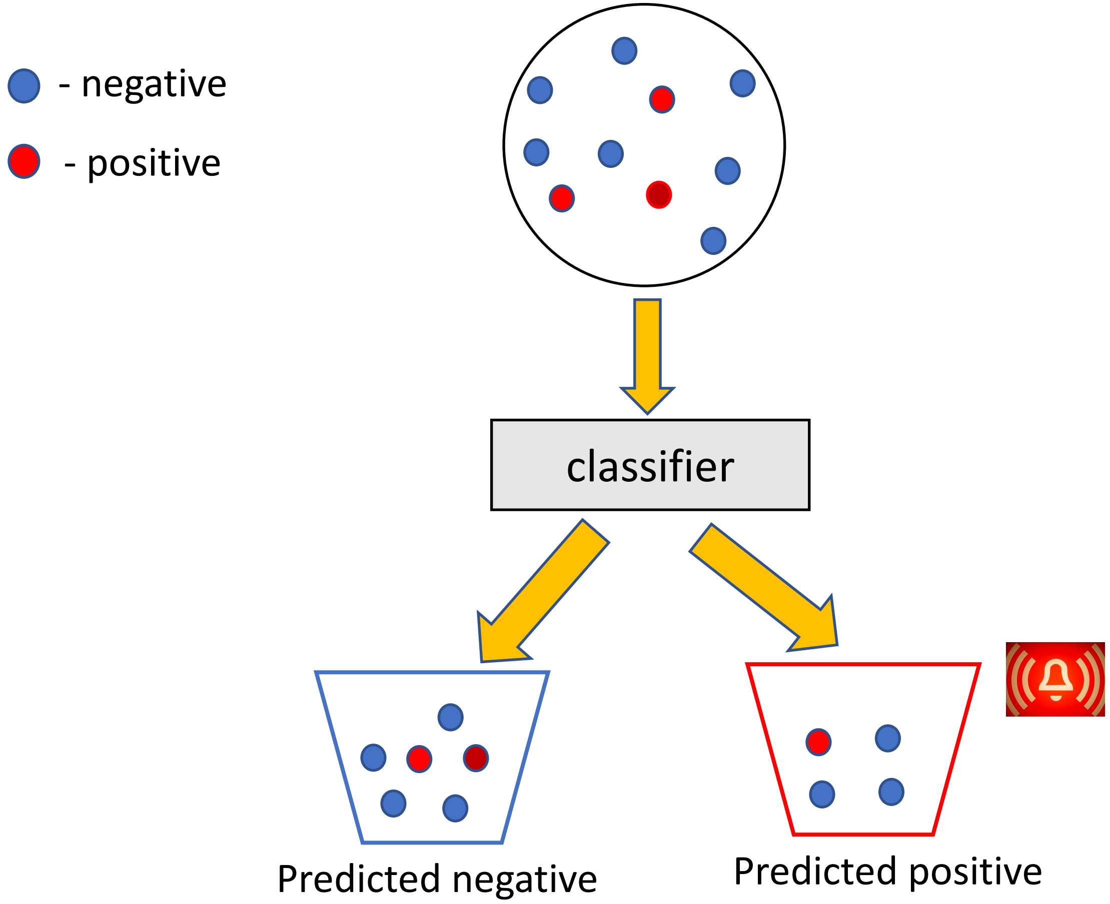
False Alarms

Alarm Fatigue

- False alarm rates of upto 72-99% are prevalent in clinical settings. [1]
- Emergency Care Research Institute (ECRI) listed **alarm fatigue** in the top ten healthcare technology hazards.[2]

[1] Sendelbach, S., & Funk, M. (2013). Alarm fatigue: a patient safety concern. *AACN advanced critical care*.
[2] https://www.ecri.org/Resources/Whitepapers_and_reports/Top_Ten_Technology_Hazards_2015.pdf

Metrics That Matter : Precision and Recall



- $\text{Precision} = \frac{tp}{tp+fp} = \frac{1}{4} = 0.25$
- $\text{Recall} = \frac{tp}{tp+fn} = \frac{1}{3} = 0.33$

- **Precision** measures fraction of alarms that are correct (1 - precision = **false alarm rate)**
- **Recall** measures the ability to identify patients who are **truly at-risk**

Standard Way to Develop Early Warning Classifiers

1. Train classifier via a Binary Cross Entropy (BCE) objective

$$\max_{\theta} \Sigma_{\mathrm{n}}\, y_n \log p_n + (1 - y_n)\log(1 - p_n);\ \text{where}\ p_n = \sigma(f_\theta(x_n))$$

BCE does not treat false alarms differently from other kinds of mistakes

- $x_n : n^{th}\ feature\ vector$
- $y_n : n^{th}\ binary\ label$
- $f_\theta : score\ function\ of\ binary\ classifier\ with\ parameters\ \theta$
- $\sigma : logistic\ sigmoid\ function$

2. After training, try to balance goals for recall and false alarm rate
   - Search across a grid of possible thresholds (operating points)

Proposed Ideal Objective: Maximize Recall subject to a False Alarm Constraint

Train classifier such that :

Identify most possible patients truly at-risk

$$\max_{\theta} \text{recall}(\theta)$$

$$s.t. \; precision(\theta) \geq \alpha$$

Ensure false alarm rate is less than $1 - \alpha$

Allows stakeholders to specify tolerable false alarm rate via α

Challenge: Make Ideal Objective Tractable for Gradient Descent

Ideal objective

$$\max_{\theta} \text{recall}(\theta)$$
$$s.t\ precision(\theta) \geq \alpha$$

$$true\ positive\ count\ (tpc) = \sum_{n:y_n=1} z(f_\theta(x_n) - b)$$

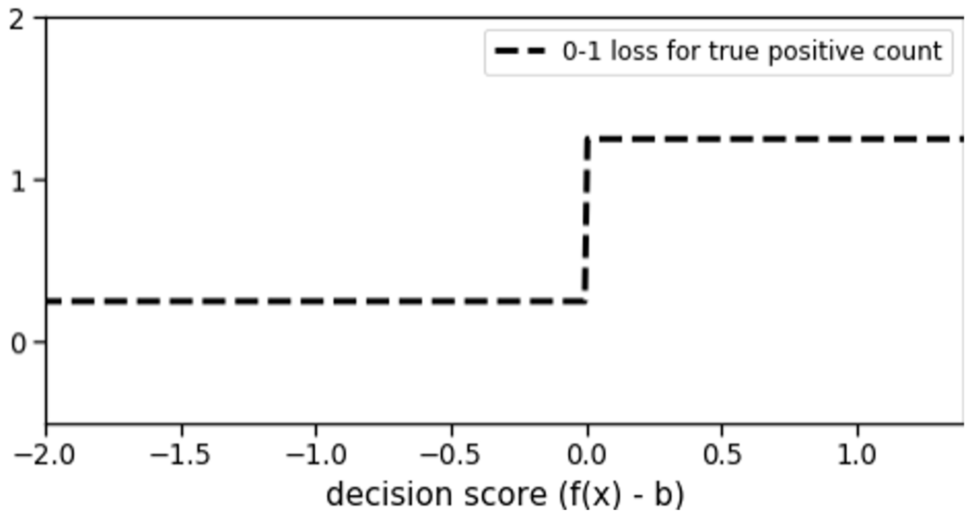
$$false\ positive\ count\ (fpc\ ) = \sum_{n:y_n=0} z(f_\theta(x_n) - b)$$

Equivalent unconstrained objective (via penalty method)

$$\min_{\theta} -tpc(\theta) + \lambda\, max(0, (-tpc(\theta) + \left(\frac{\alpha}{1+\alpha}\right) fpc(\theta)))$$

If constraint is satisfied

Constraint not satisfied



Problem: zero-one function is *flat,*
so gradients uninformative for learning

Previous work: Hinge bounds on zero-one function

$$\min_{\theta} -tpc(\theta) + \lambda\, max(0, (-tpc(\theta) + \left(\frac{\alpha}{1+\alpha}\right) fpc(\theta)))$$

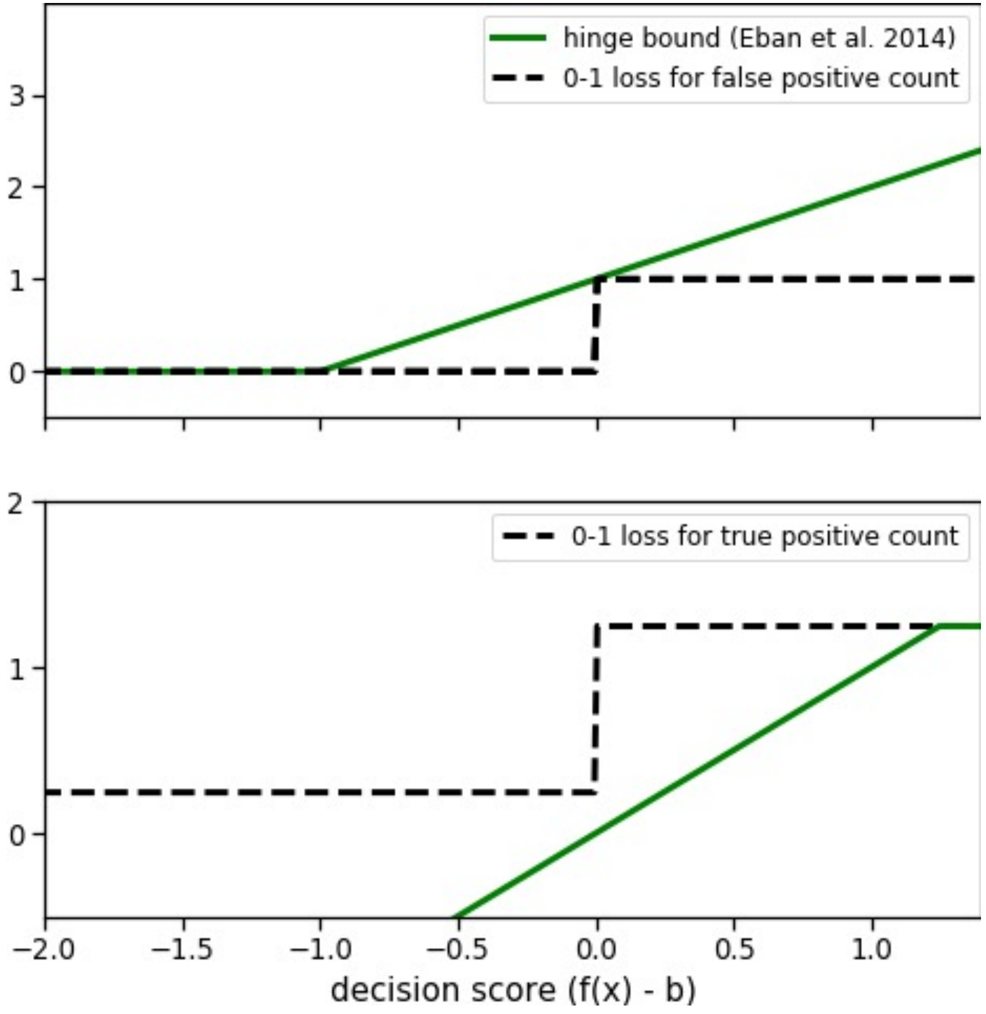
Value when constraint is satisfied   Value when constraint is yet to be satisfied

$$true\ positive\ count\ (tpc) = \sum_n 1 - max(0, 1 - y_n.(f_\theta(x_n) - b))$$

$$false\ positive\ count\ (fpc\ ) = \sum_n \max(0, 1 - y_n.(f_\theta(x_n) - b))$$

$$y_n \in \{+1, -1\}$$

- Hinge Bound (Eban et al, 2017) [1]
  - Piecewise linear, but **bounds too loose**
  - Bound on true positives count goes **negative**



[1] Eban, E., Schain, M., Mackey, A., Gordon, A., Rifkin,R., and Elidan, G. Scalable Learning of Non-Decomposable Objectives. In AISTATS 2017.

Proposed Method: Sigmoid Bounds That Ensure Positivity

$$\min_{\theta} -tpc(\theta) + \lambda\, max(0, (-tpc(\theta) + \left(\frac{\alpha}{1+\alpha}\right) fpc(\theta)))$$

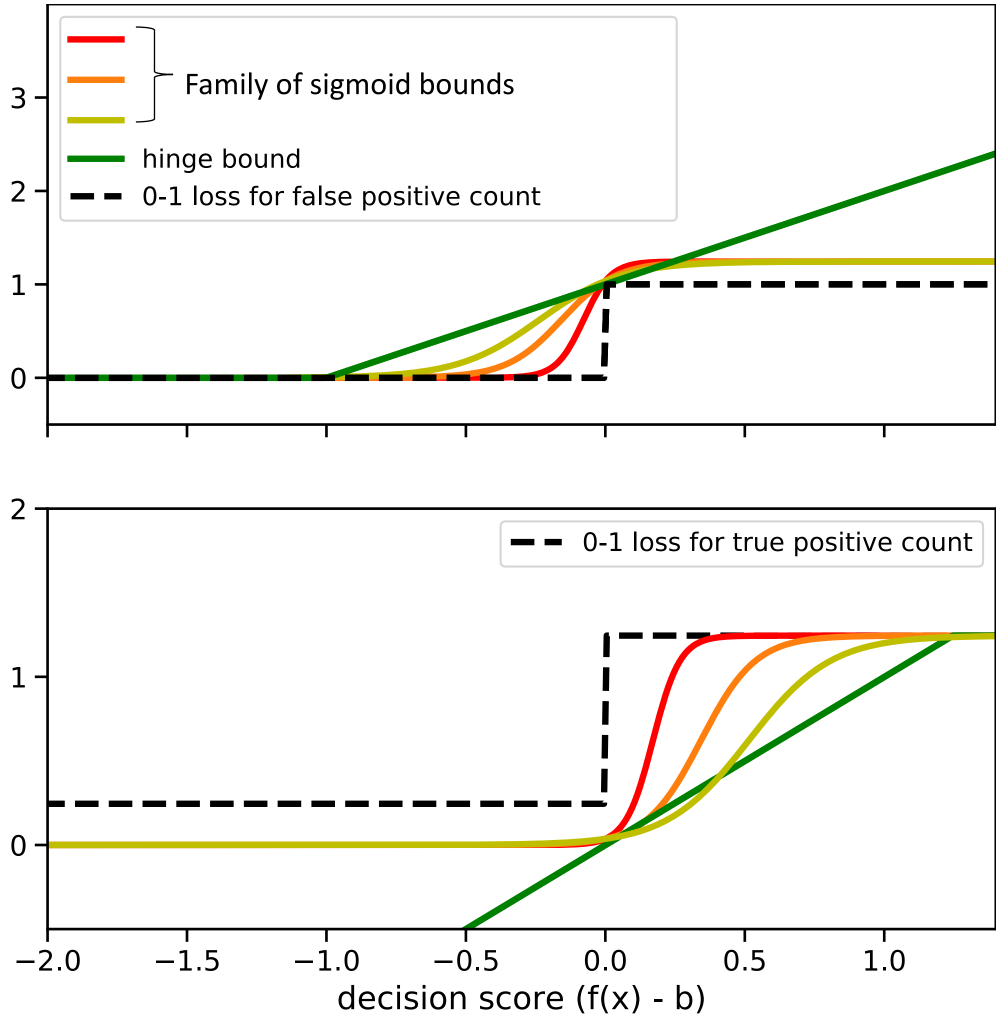
Value when constraint is satisfied Value when constraint is yet to be satisfied

$$true\ positive\ count\ (tpc) = \sum_n (1+\gamma\delta)\sigma(m\, f_\theta(x_n) + b)$$

$$false\ positive\ count\ (fpc) = \sum_n (1+\hat{\gamma}\,\hat{\delta})\sigma(\hat{m}\, f_\theta(x_n) + \hat{b})$$

σ : sigmoid function

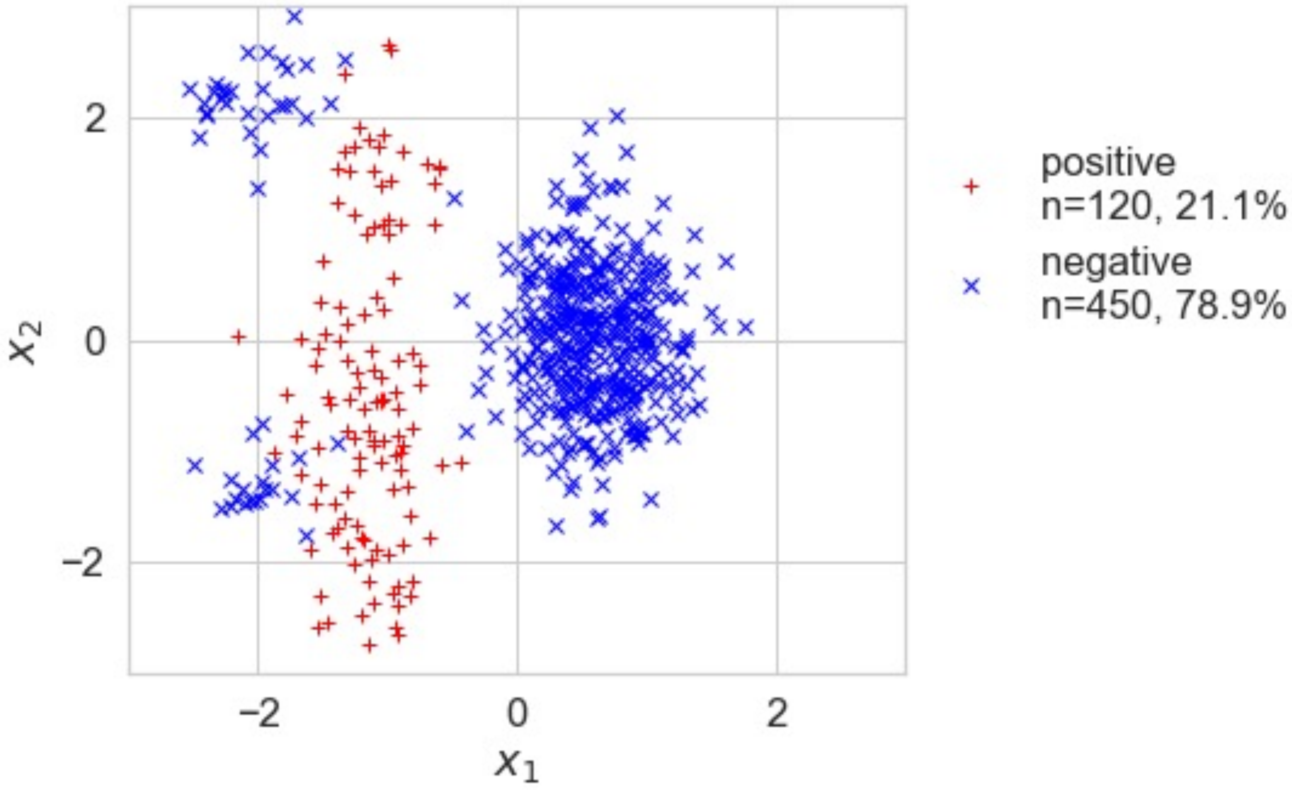
γ, δ, m, b : parameters to control tightness of sigmoid



Our proposed sigmoid bounds

- ✓ **Tight, differentiable**, and **easily adapts to classifiers trainable via SGD**
- ✓ **Bounds** on true positive and false positive are **always positive** (not true for Eban et.al's hinge bound)
- ✓ **Linear runtime O(n)**

Setting Up A Toy Classification Task



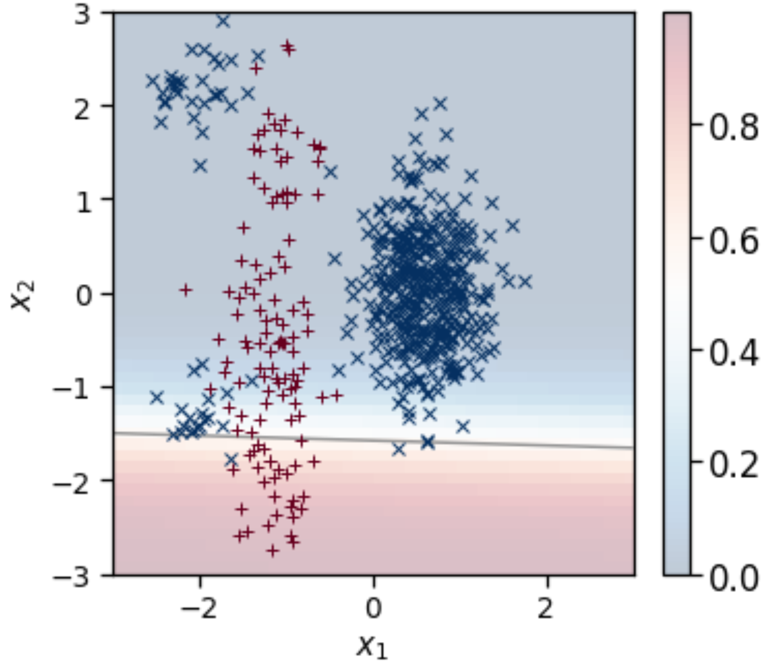
Goal : Train linear model to maximize recall at precision $\geq \mathbf{0.9}$

Sigmoid Bound Delivers Best Recall & Satisfies Precision Constraint

Goal : Train linear model to maximize recall at precision $\geq \mathbf{0.9}$

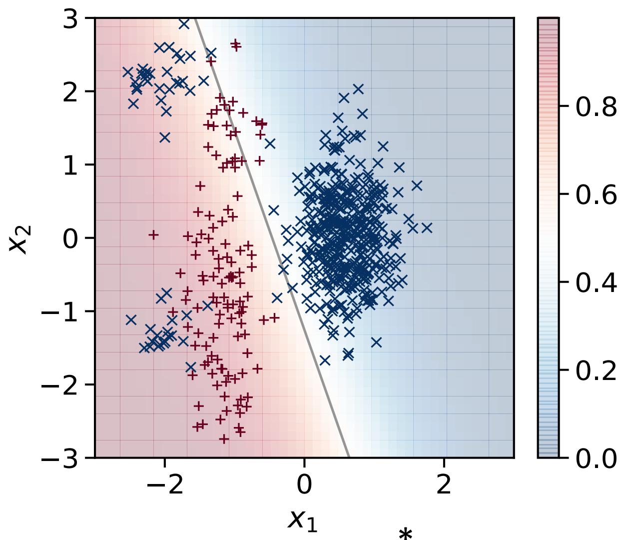
Best possible solution

Prec:0.906 Rec:0.242



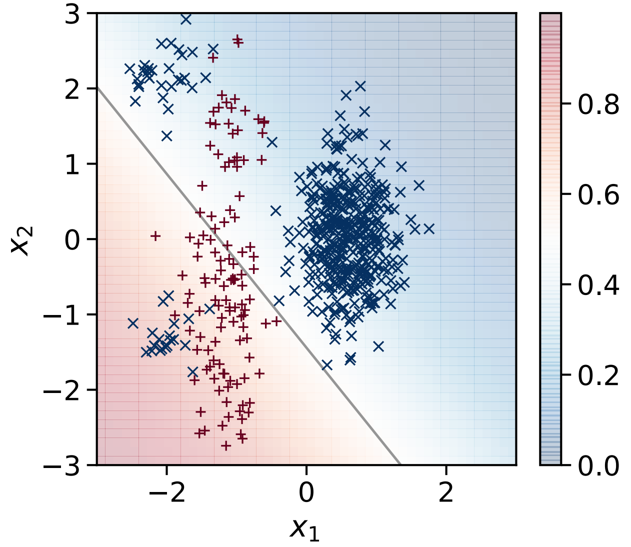
BCE + threshold search

Prec:0.684 Rec:0.900 Runtime:10 sec



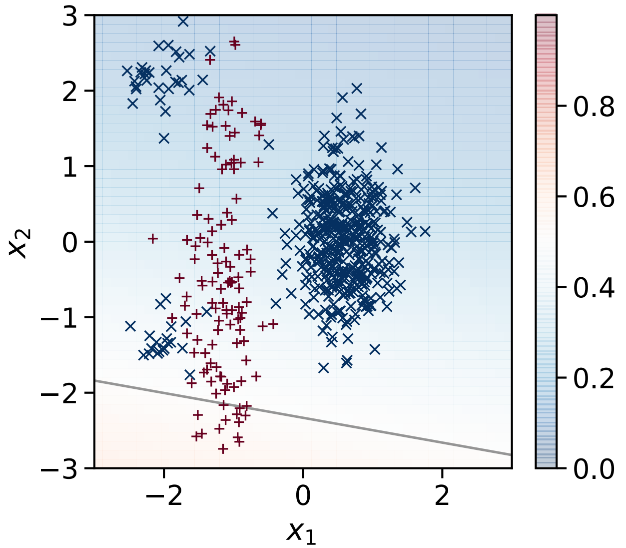
Hinge Bound + thresh search (α=0.8)

Prec:0.798 Rec:0.658 Runtime:10 sec



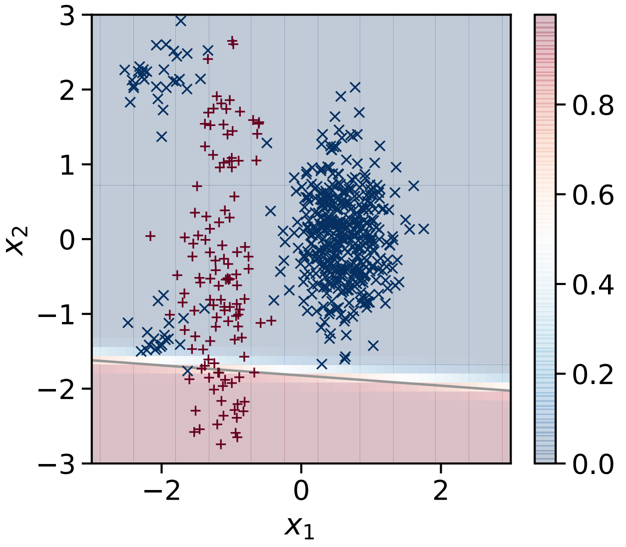
Adversarial Prediction* (α=0.9)

Prec:1.000 Rec:0.108 Runtime:3000 sec



Proposed Sigmoid Bound (α=0.9)

Prec:0.962 Rec:0.208 Runtime:10 sec



* Fathony, R. and Kolter, Z. AP-perf: Incorporating generic performance metrics in differentiable learning. In AISTATS 2020

Evaluation: Mortality risk prediction on 2 hospital datasets

- MIMIC-3 (Johnson et al, 2016)
  - EHR Data from Boston hospital
  - 34472 patient stays
  - 2 demographics, 10 vitals, 94 laboratory measurements
  - ~9.5% patient stays resulted in death
- eICU (Pollard et al 2018)
  - EHR data from 59 critical care units across USA
  - 72670 patient stays
  - 3 demographics, 8 vitals, 10 laboratory measurements
  - ~8.2% patient stays resulted in death

Our Sigmoid Bounds Achieve Best Recall on Test Set on both EHR Datasets

- Evaluation : Predictions per sequence
- Goal : Maximize recall s.t. precision ≥ 0.9

	Method	Recall on MIMIC Train	Recall on MIMIC Test	Recall on eICU Train	Recall on eICU Test
Logistic Regression	BCE + threshold search	.572 (.566, .580)	.540 (.532, .551)	.100 (.096, .104)	.087 (.082, .092)
	Hinge Bound (Eban et al.)	.696 (.691, .704)	.625 (.612, .635)	.018 (.015, .019)	.016 (.012, .018)
	Sigmoid Bound (ours)	.763 (.755, .766)	.684 (.675, .696)	.211 (.206, .218)	.195 (.188, .206)
1-layer MLP	BCE + threshold search	.766 (.757, .772)	.689 (.676, .708)	.325 (.321, .330)	.280 (.271, .290)
	Hinge Bound (Eban et al.)	.741 (.734, .750)	.640 (.620, .654)	.359 (.355, .363)	.235 (.229, .246)
	Sigmoid Bound (ours)	.821 (.817, .827)	.717 (.705, .729)	.385 (.379, .392)	.298 (.288, .305)

Summary

Novel framework for limiting false alarms with tight sigmoid bounds

Applies to any classifier trainable via SGD

Robust empirical evaluation shows reduced false alarms in critical care settings

Recall improves by 3 to 15% while ensuring false alarm rate below 10%

Python Code URL :

github.com/tufts-ml/false-alarm-control