On Symmetric Losses for Learning from Corrupted Labels

Nontawat Charoenphakdee[1,2] , Jongyeong Lee[1,2]
and Masashi Sugiyama[2,1]

The University of Tokyo[1]

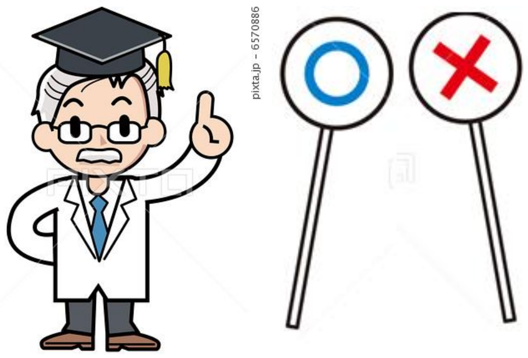
RIKEN Center for Advanced Intelligence Project (AIP)[2]

Supervised learning

Learn from input-output pairs → **Such that** → **Predict** output of **unseen input** accurately

Data collection

Features (Input) **Labels (Output)**



Machine learning →

Prediction function

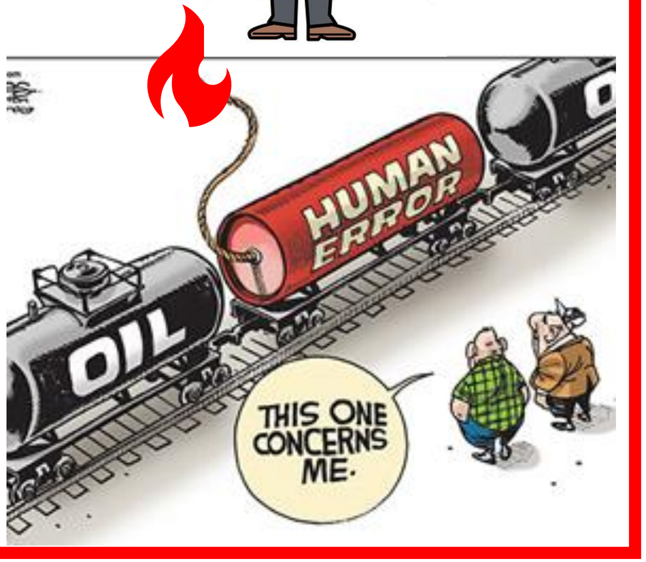
https://t.pimg.jp/006/570/886/1/6570886.jpg
https://www.kullabs.com/uploads/meauring-clip-art-at-clker-com-vector-clip-art-online-royalty-free-H2SJHF-clipart.png
https://d3njjcbhbojbot.cloudfront.net/api/utilities/v1/imageproxy/https://coursera.s3.amazonaws.com/topics/ml/large-icon.png\

Learning from corrupted labels

Data collection

Labeling process

Feature collection



Prediction function

Our goal

Noise-robust ML

Examples:

- Expert labelers (human error)
- Crowdsourcing (non-expert error)

https://thumbs.dreamstime.com/b/power-crowd-d-render-crowdsourcing-concept-30738769.jpg
http://www.process-improvement-institute.com/wp-content/uploads/2015/05/Accounting-for-Human-Error-Probability-in-SIL-Verification.jpg
https://www.kullabs.com/uploads/meauring-clip-art-at-clker-com-vector-clip-art-online-royalty-free-H2SJHF-clipart.png
https://d3njjcbhbojbot.cloudfront.net/api/utilities/v1/imageproxy/https://coursera.s3.amazonaws.com/topics/ml/large-icon.png

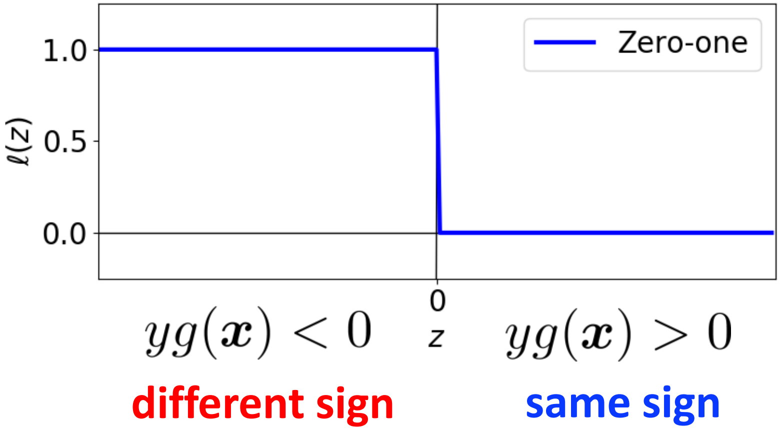
Contents

Warmup: Binary classification

$y \in \{-1, 1\}$: Label
$g: \mathbb{R}^d \to \mathbb{R}$: Prediction function
$\boldsymbol{x} \in \mathbb{R}^d$: Feature vector
$\ell: \mathbb{R} \to \mathbb{R}$: Margin loss function

- **Given**: input-output pairs:

$$\{\boldsymbol{x}_i, y_i\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, y)$$



- **Goal**: minimize **expected error**:

$$R^{\ell_{0\text{-}1}}(g) = \mathbb{E}_{(\boldsymbol{x},y)\sim p(\boldsymbol{x},y)}\left[\ell_{0\text{-}1}(yg(\boldsymbol{x}))\right]$$

No access to distribution: minimize **empirical error** (Vapnik, 1998):

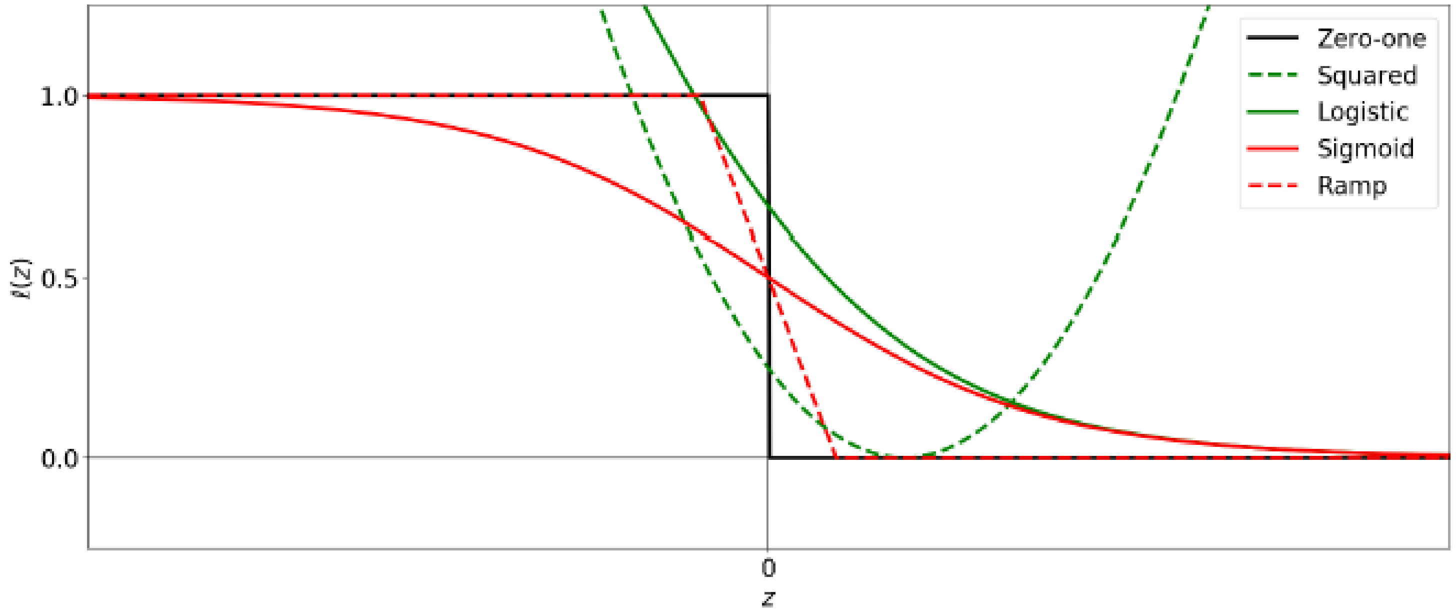
$$\hat{R}^{\ell_{0\text{-}1}}(g) = \frac{1}{n}\sum_{i=1}^n \ell_{0\text{-}1}(y_i g(\boldsymbol{x}_i))$$

Surrogate losses

Minimizing 0-1 loss directly is **difficult.**

- Discontinuous and not differentiable (Ben-david+, 2003, Feldman+, 2012)

In practice, we minimize a **surrogate loss** (Zhang, 2004, Bartlett+, 2006).



$z = yg(\boldsymbol{x})$: Margin

$y \in \{-1, 1\}$: Label

$g \colon \mathbb{R}^d \to \mathbb{R}$: Prediction function

$\boldsymbol{x} \in \mathbb{R}^d$: Feature vector

Learning from corrupted labels

(Scott+, 2013, Menon+, 2015, Lu+, 2019)

Given: Two sets of corrupted data:

Positive: $X_{\mathrm{CP}} := \{\boldsymbol{x}_i^{\mathrm{CP}}\}_{i=1}^{n_{\mathrm{CP}}} \overset{\text{i.i.d.}}{\sim} \pi\,\mathrm{pos}(\boldsymbol{x}) + (1-\pi)\,\mathrm{neg}(\boldsymbol{x})$

Negative: $X_{\mathrm{CN}} := \{\boldsymbol{x}_i^{\mathrm{CN}}\}_{i=1}^{n_{\mathrm{CN}}} \overset{\text{i.i.d.}}{\sim} \pi'\,\mathrm{pos}(\boldsymbol{x}) + (1-\pi')\,\mathrm{neg}(\boldsymbol{x})$

Class priors

$\pi, \pi' \in [0,1]$
$\mathrm{pos}(\boldsymbol{x}) : p(\boldsymbol{x}|y=1)$
$\mathrm{neg}(\boldsymbol{x}) : p(\boldsymbol{x}|y=-1)$

Clean: $\pi = 1, \pi' = 0$

Positive-unlabeled: $\pi = 1, \pi' < 1$ (du Plessis+, 2014)

This setting covers many weakly-supervised settings (Lu+, 2019).

Issue on class priors

Given: Two sets of corrupted data:

Positive: $X_{\mathrm{CP}} := \{\boldsymbol{x}_i^{\mathrm{CP}}\}_{i=1}^{n_{\mathrm{CP}}} \overset{\text{i.i.d.}}{\sim} \pi\, \mathrm{pos}(\boldsymbol{x}) + (1-\pi)\, \mathrm{neg}(\boldsymbol{x})$

Negative: $X_{\mathrm{CN}} := \{\boldsymbol{x}_i^{\mathrm{CN}}\}_{i=1}^{n_{\mathrm{CN}}} \overset{\text{i.i.d.}}{\sim} \pi'\mathrm{pos}\,(\boldsymbol{x}) + (1-\pi')\, \mathrm{neg}(\boldsymbol{x})$

Assumption: $\pi > \pi'$

Problem: π, π' are **unidentifiable** from samples (Scott+, 2013).

How to learn **without estimating** π, π' ?

Related work:

Class priors are needed! (Lu+, 2019)

Classification error:

$$R^{\ell_{0\text{-}1}}(g) = \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim p(\boldsymbol{x},y)}[\ell_{0\text{-}1}(yg(\boldsymbol{x}))]$$

$\mathbb{E}_{\mathrm{P}}[\cdot] : \mathop{\mathbb{E}}_{\boldsymbol{x}\sim \mathrm{pos}(\boldsymbol{x})}[\cdot]$

$\mathbb{E}_{\mathrm{N}}[\cdot] : \mathop{\mathbb{E}}_{\boldsymbol{x}\sim \mathrm{neg}(\boldsymbol{x})}[\cdot]$

Class priors are not needed! (Menon+, 2015)

Balanced error rate (BER):

$$R^{\ell_{0\text{-}1}}_{\mathrm{Bal}}(g) = \frac{1}{2}\mathbb{E}_{\mathrm{P}}\left[\ell_{0\text{-}1}(g(\boldsymbol{x}^{\mathrm{P}}))\right] + \frac{1}{2}\mathbb{E}_{\mathrm{N}}\left[\ell_{0\text{-}1}(-g(\boldsymbol{x}^{\mathrm{N}}))\right]$$

Area under the receiver operating characteristic curve (AUC) risk:

$$R^{\ell_{0\text{-}1}}_{\mathrm{AUC}}(g) = \mathbb{E}_{\mathrm{P}}[\mathbb{E}_{\mathrm{N}}[\ell_{0\text{-}1}(g(\boldsymbol{x}^{\mathrm{P}}) - g(\boldsymbol{x}^{\mathrm{N}}))]]$$

Related work: BER and AUC optimization

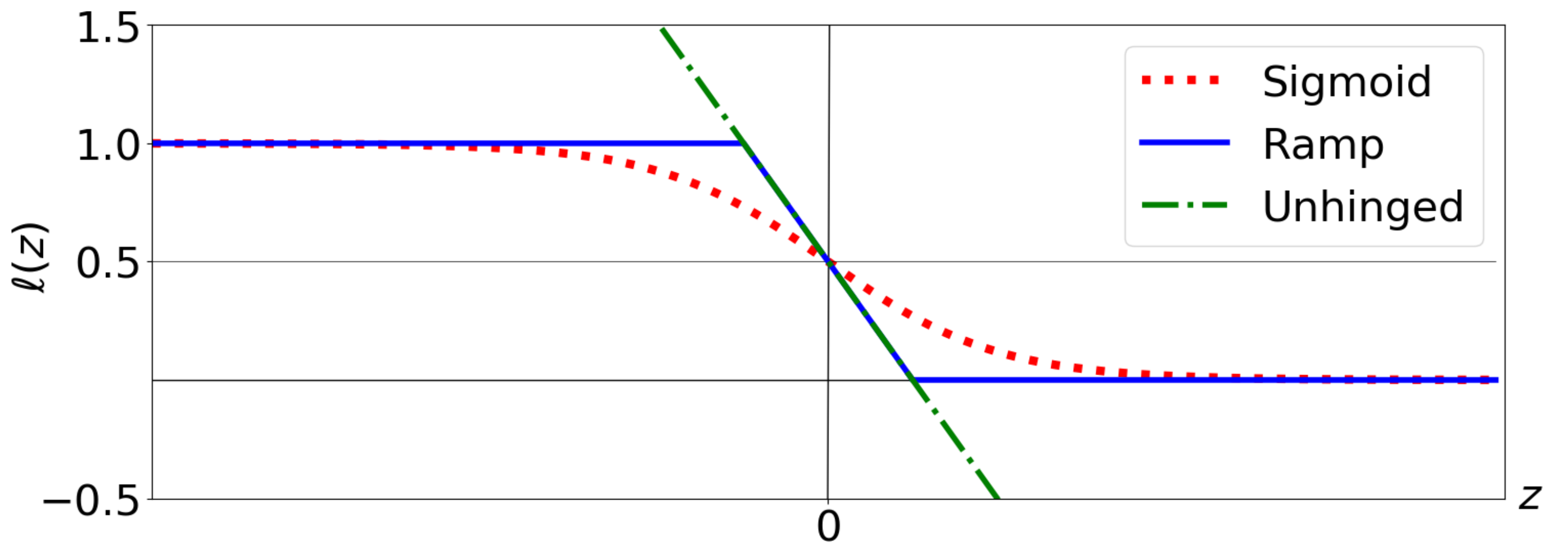
Menon+, 2015: we can treat ***corrupted data as if they were clean***.
The proof relies on a **property of 0-1 loss.**
Squared loss was used in experiments.

van Rooyen+, 2015: **symmetric losses** are also useful for **BER** minimization (no experiments).

Ours: using symmetric loss is preferable for both BER and AUC theoretically and experimentally!

Contents

Symmetric losses $\ell(z) + \ell(-z) = \text{Constant}$



Applications:

Risk estimator simplification in weakly-supervised learning

(du Plessis+, 2014, Kiryo+, 2017, Lu+, 2018)

Robustness under symmetric noise (label flip with a fixed probability)

(Ghosh+, 2015, van Rooyen+, 2015)

AUC maximization

$$f(\boldsymbol{x}, \boldsymbol{x}') = g(\boldsymbol{x}) - g(\boldsymbol{x}')$$

Theorem 1. *Let* $\gamma^\ell(\boldsymbol{x}, \boldsymbol{x}') = \ell(f(\boldsymbol{x}', \boldsymbol{x})) + \ell(f(\boldsymbol{x}, \boldsymbol{x}'))$. *Then* $R^\ell_{\text{AUC-Corr}}(g)$ *can be expressed as*

$$R^\ell_{\text{AUC-Corr}}(g) = (\pi - \pi')R^\ell_{\text{AUC}}(g) + \underbrace{(\pi' - \pi\pi')\mathbb{E}_+[\mathbb{E}_-[\gamma^\ell(\boldsymbol{x}_+, \boldsymbol{x}_-)]]}_{\text{Excessive term}}$$

$$+ \underbrace{\frac{\pi\pi'}{2}\mathbb{E}_{+'}[\mathbb{E}_+[\gamma^\ell(\boldsymbol{x}_{+'}, \boldsymbol{x}_+)]] + \frac{(1-\pi)(1-\pi')}{2}\mathbb{E}_{-'}[\mathbb{E}_-[\gamma^\ell(\boldsymbol{x}_{-'}, \boldsymbol{x}_-)]]}_{\text{Excessive term}}.$$

Corrupted risk

Clean risk

Symmetric losses: $\ell(z) + \ell(-z) = K$

When $\gamma^\ell(\boldsymbol{x}, \boldsymbol{x}') = K$ which holds for symmetric losses, we have

$$R^\ell_{\text{AUC-Corr}}(g) = (\pi - \pi')R^\ell_{\text{AUC}}(g) + K\left(\frac{1 - \pi + \pi'}{2}\right).$$

Excessive terms become constant!

Excessive terms can be safely ignored with symmetric losses ☺

BER minimization

Theorem 3. *Let* $\gamma^\ell(\boldsymbol{x}) = \ell(g(\boldsymbol{x})) + \ell(-g(\boldsymbol{x}))$, $R^\ell_{\text{Bal-Corr}}(g)$ *can be expressed as*

$$R^\ell_{\text{Bal-Corr}}(g) = (\pi - \pi')R^\ell_{\text{Bal}}(g) + \underbrace{\frac{\pi'\mathbb{E}_+[\gamma^\ell(\boldsymbol{x})] + (1-\pi)\mathbb{E}_-[\gamma^\ell(\boldsymbol{x})]}{2}}_{\text{Excessive term}}.$$

Corrupted risk — Clean risk

Symmetric losses: $\ell(z) + \ell(-z) = K$

Excessive term becomes constant!

When $\gamma^\ell(\boldsymbol{x}) = K$ which holds for symmetric losses, we have

$$R^\ell_{\text{Bal-Corr}}(g) = (\pi - \pi')R^\ell_{\text{Bal}}(g) + K\left(\frac{1-\pi+\pi'}{2}\right).$$

Coincides with van Rooyen 2015+

Excessive terms can be safely ignored with symmetric losses☺

Contents

Theoretical properties of symmetric losses

Nonnegative symmetric losses are **non-convex**.

- Theory of convex losses cannot be applied. ☹ (du Plessis+, 2014, Ghosh+, 2015)

We provide a better understanding of symmetric losses: ☺

- **Necessary and sufficient condition** for **classification-calibration**
- **Excess risk bound** in binary classification
- **Inability** to estimate **class posterior probability**
- A **sufficient condition** for **AUC-consistency**
  - Covers many symmetric losses, e.g., sigmoid, ramp.

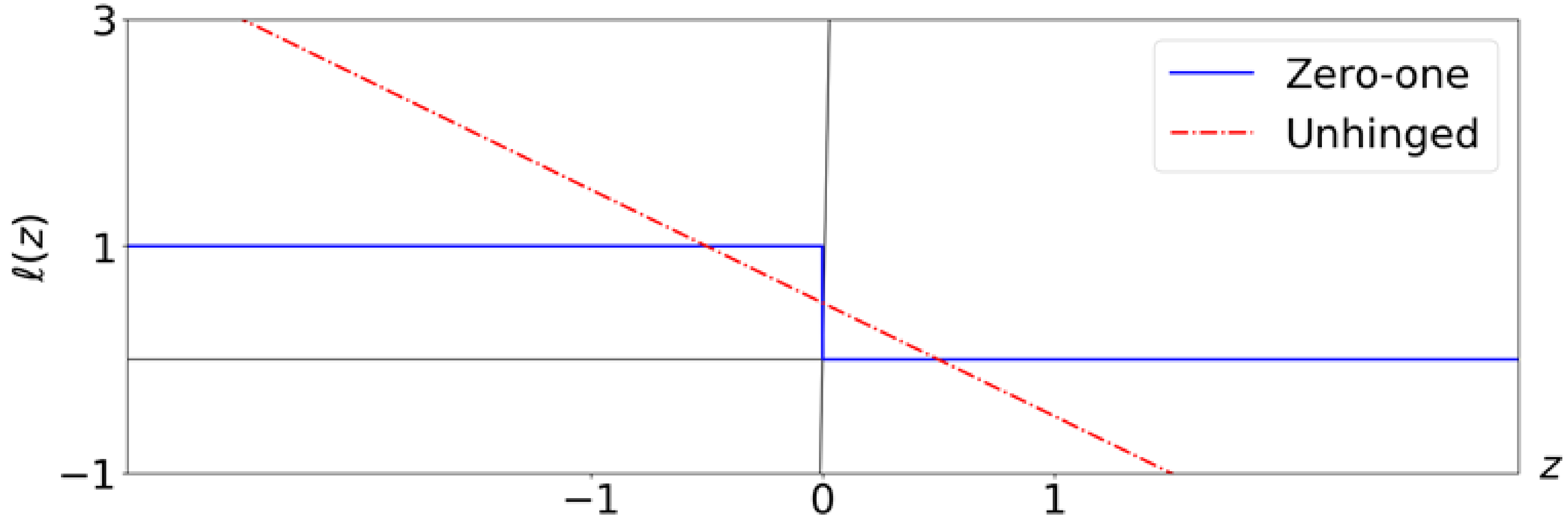
Well-known symmetric losses, e.g., sigmoid, ramp are classification-calibrated and AUC-consistent!

Contents

Convex symmetric losses?

By sacrificing nonnegativity:

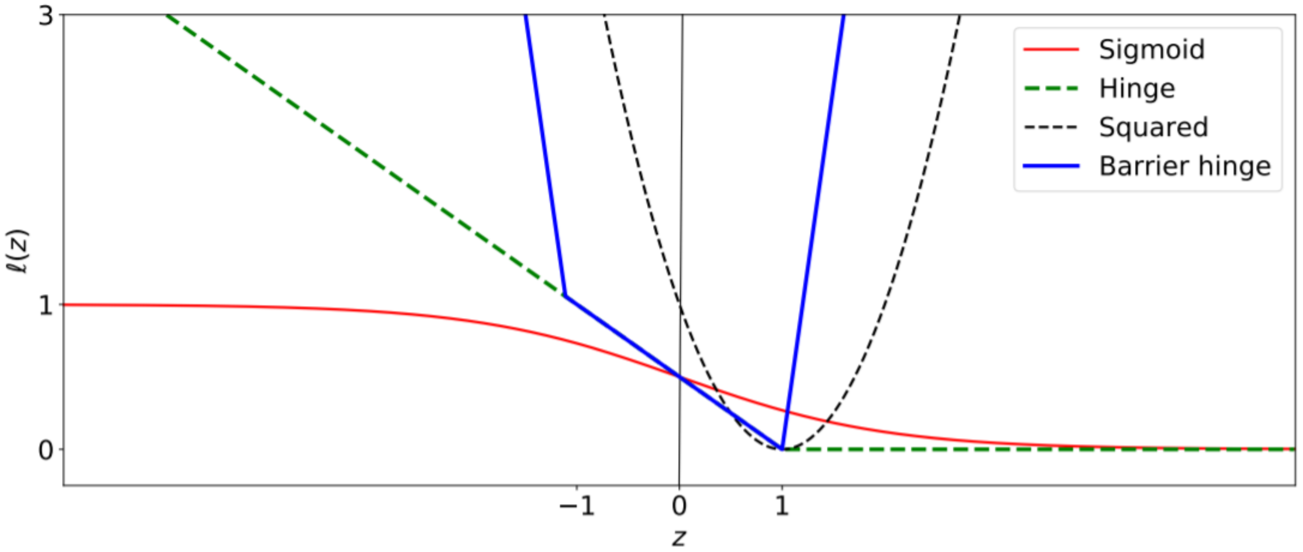
only unhinged loss is convex and symmetric (van Rooyen+, 2015).



This loss has been considered (although robustness was not discussed).

(Devroye+, 1996, Schoelkopf+, 2002, Shawe-Taylor+, 2004, Sriperumbudur+, 2009, Reid+, 2011)

Barrier hinge loss



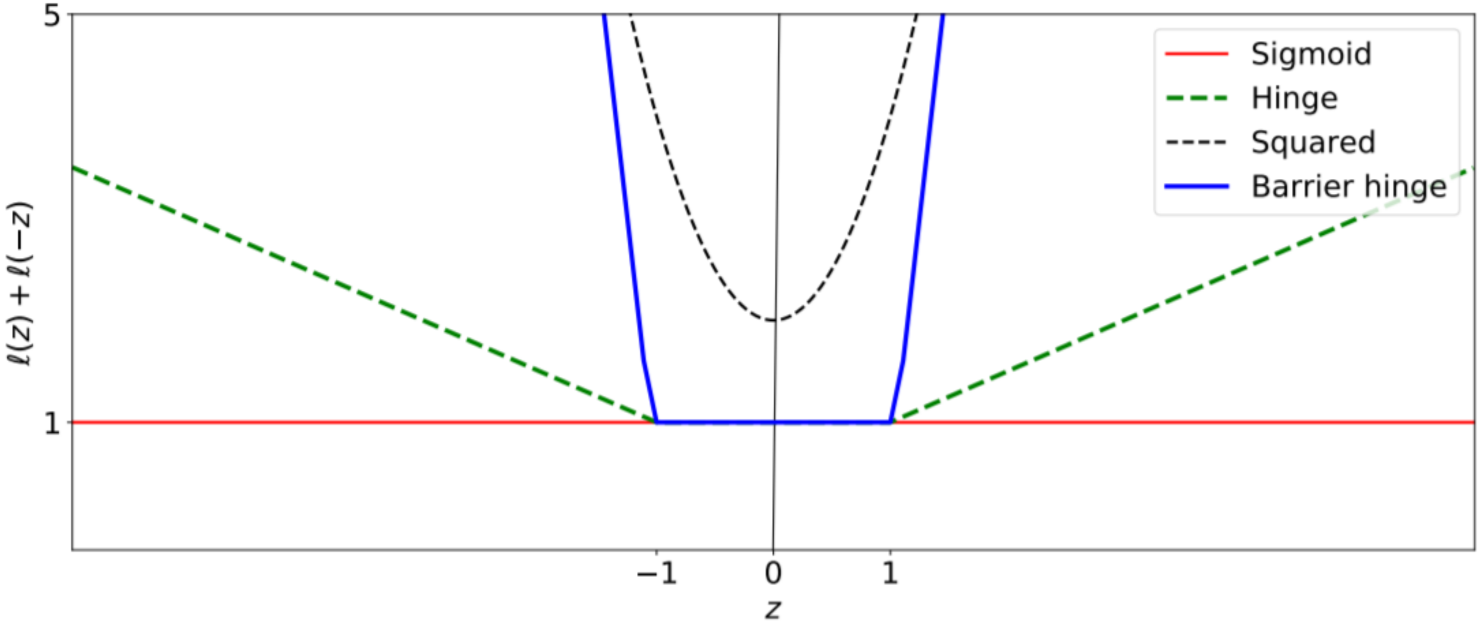
$$\ell(z) = \max(-s(w+z)+w, \max(s(z-w), w-z))$$

$s > 1$ **slope** of the **non-symmetric** region.

$w > 0$ **width** of **symmetric** region.

High penalty if **misclassify or** output is **outside symmetric region**.

Symmetricity of barrier hinge loss



Satisfies symmetric property in an interval.

If output range is restricted in a symmetric region:

unhinged, **hinge** , **barrier** are equivalent.

Contents

Experiments: BER/AUC optimization from corrupted labels

To empirically answer the following questions:

1. Does the symmetric condition significantly help?
2. Do we need a loss to be symmetric everywhere?
3. Does the negative unboundedness degrade the practical performance?

We conducted the following experiments: Fix the models, vary the loss functions

Losses: **Barrier [b=200, r=50]**, **Unhinged, Sigmoid**, Logistic, Hinge, Squared, Savage

Experiment 1:

MLPs on UCI/LIBSVM datasets.

Experiment 2:

CNNs on more difficult datasets (MNIST, CIFAR-10).

Experiments: BER/AUC optimization from corrupted labels

For UCI datasets:

Multilayered perceptrons (MLPs) with one hidden layer: [d-500-1]

Activation function: Rectifier Linear Units (ReLU) (Nair+, 2010)

MNIST and CIFAR-10:

Convolutional neural networks (CNNs):

[d-Conv[18,5,1,0]-Max[2,2]-Conv[48,5,1,0]-Max[2,2]-800-400-1]

ReLU after fully connected layer follows by dropout layer (Srivastava+, 2010)

MNIST: Odd numbers vs Even numbers

CIFAR: One class vs Airplane (follows Ishida+, 2017)

Conv[18, 5, 1 , 0]: 18 channels, 5 x 5 convolutions, stride 1, padding 0
Max[2,2]: max pooling with kernel size 2 and stride 2

Experiment 1: MLPs on UCI/LIBSVM datasets

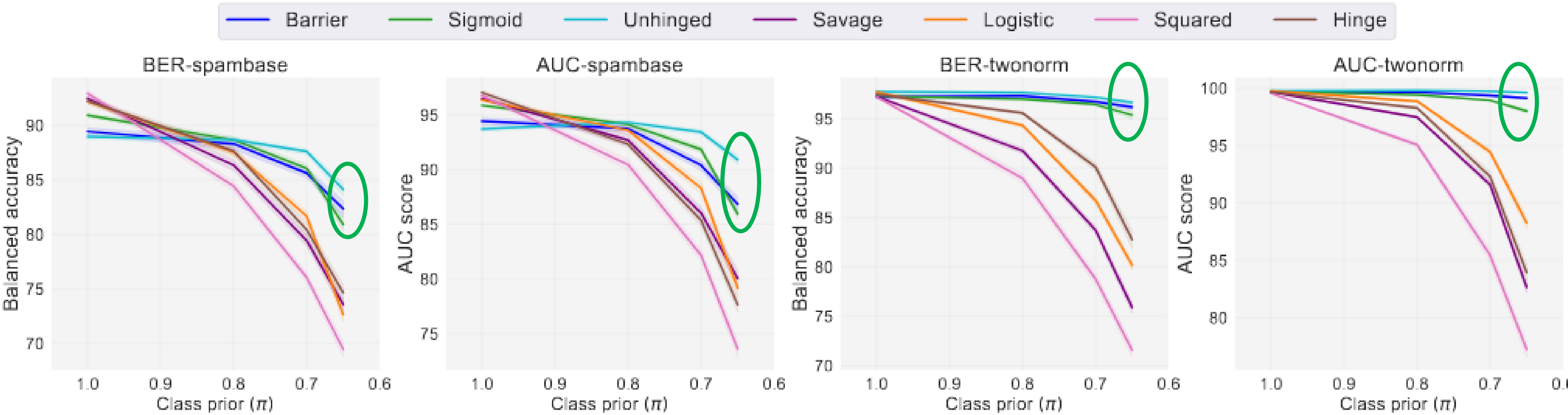


Figure 4: Mean balanced accuracy (1-BER) and AUC score using multilayer perceptrons (rescaled to 0-100) with varying noise rates $(\pi = 1.0, \pi' = 0.0), (\pi = 0.8, \pi' = 0.3), (\pi = 0.7, \pi' = 0.4), (\pi = 0.65, \pi' = 0.45)$. The experiments were conducted 20 times.

The higher the better.

Dataset information and more experiments and can be found in our paper.

Experiment 1: MLPs on UCI/LIBSVM datasets

Symmetric losses and barrier hinge loss are preferable!

Table 2. Mean balanced accuracy (BAC=1-BER) and AUC score using multilayer perceptrons (rescaled to 0-100), where $\pi = 0.65$ and $\pi' = 0.45$. Outperforming methods are highlighted in boldface using one-sided t-test with the significance level 5%. The experiments were conducted 20 times.

Dataset	Task	Barrier	Unhinged	Sigmoid	Logistic	Hinge	Squared	Savage
spambase	BAC	82.3(0.8)	**84.1 (0.6)**	80.9(0.6)	72.6(0.7)	74.7(0.7)	69.5(0.7)	73.6(0.6)
	AUC	86.8(0.7)	**90.9 (0.4)**	86.0(0.4)	79.2(0.8)	77.7(0.7)	73.6(0.8)	80.1(0.8)
waveform	BAC	**86.1 (0.4)**	**87.1 (0.6)**	85.4(0.6)	75.8(0.7)	78.3(0.7)	69.2(0.6)	73.2(0.6)
	AUC	**92.2 (0.4)**	**91.7 (0.6)**	**90.9 (0.6)**	82.3(0.7)	79.8(0.9)	75.1(0.7)	80.1(0.6)
twonorm	BAC	**96.2 (0.3)**	**96.7 (0.2)**	95.4(0.4)	80.2(0.5)	82.8(0.9)	71.6(0.7)	75.9(0.6)
	AUC	99.1(0.1)	**99.6 (0.0)**	98.0(0.2)	88.3(0.5)	83.9(0.7)	77.3(0.7)	82.7(0.5)
mushroom	BAC	**93.4 (0.8)**	91.1(0.9)	**94.4 (0.7)**	81.3(0.5)	84.5(1.0)	72.2(0.6)	79.5(0.8)
	AUC	**98.4 (0.2)**	97.2(0.4)	**97.8 (0.3)**	89.0(0.5)	82.2(0.6)	77.8(0.6)	88.1(0.7)

The higher the better.

Experiment 2: CNNs on MNIST/CIFAR-10

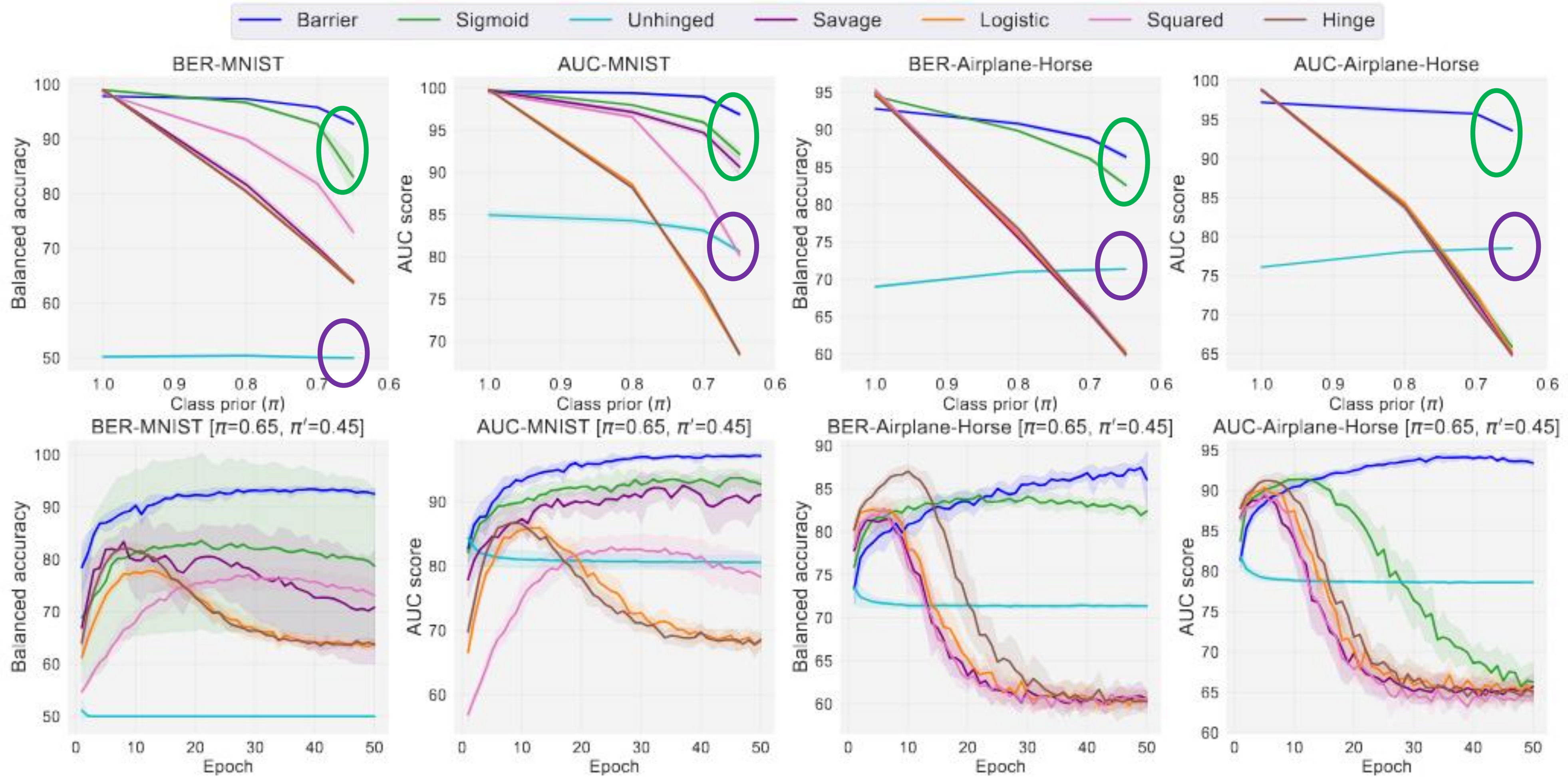
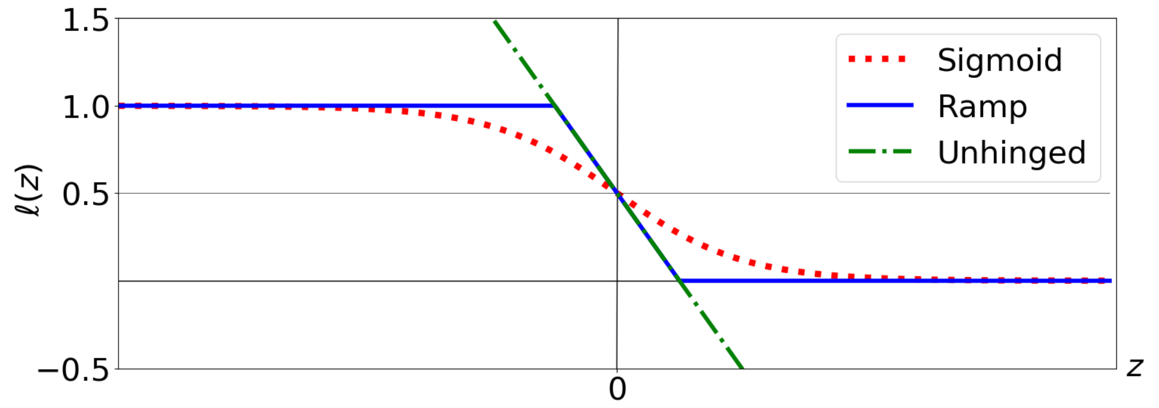


Figure 5: Mean balanced accuracy (1-BER) and AUC score using convolutional neural networks (rescaled to 0-100). (Top) the varying noise rates ranged from $(\pi = 1.0, \pi' = 0.0), (\pi = 0.8, \pi' = 0.3), (\pi = 0.7, \pi' = 0.4), (\pi = 0.65, \pi' = 0.45)$. (Bottom) the noise rate is $\pi = 0.65$ and $\pi' = 0.45$. The experiments were conducted 10 times.

Poster#135: today 6:30-9:00PM

Conclusion



We showed that **symmetric loss is preferable under corrupted labels** for:

- Area under the receiver operating characteristic curve (**AUC**) maximization
- Balanced error rate (**BER**) minimization

We provided **general theoretical properties** for symmetric losses:

- Classification-calibration, excess risk bound, AUC-consistency
- Inability of estimating the class posterior probability

We proposed a **barrier hinge loss**:

- As a proof of concept of the importance of symmetric condition
- **Symmetric only in an interval** but benefits greatly from symmetric condition
- Significantly outperformed all losses in BER/AUC optimization using CNNs