Learning Theory Bridges Loss Functions

July 13rd, 2020

Han Bao (The University of Tokyo / RIKEN AIP)

つつみ　ふくむ
Han Bao (包 含)

https://hermite.jp/

- 2nd-year Ph.D. student @ Sugiyama-Honda-Yokoya Lab
- Research Interests:
  robustness and knowledge transfer via loss function

robustness

- **Calibrated Surrogate Maximization of Linear-fractional Utility in Binary Classification. (AISTATS2020)**
- **Calibrated Surrogate Losses for Adversarially Robust Classification. (COLT2020)**
- Calibrated surrogate maximization of Dice. (MICCAI2020)

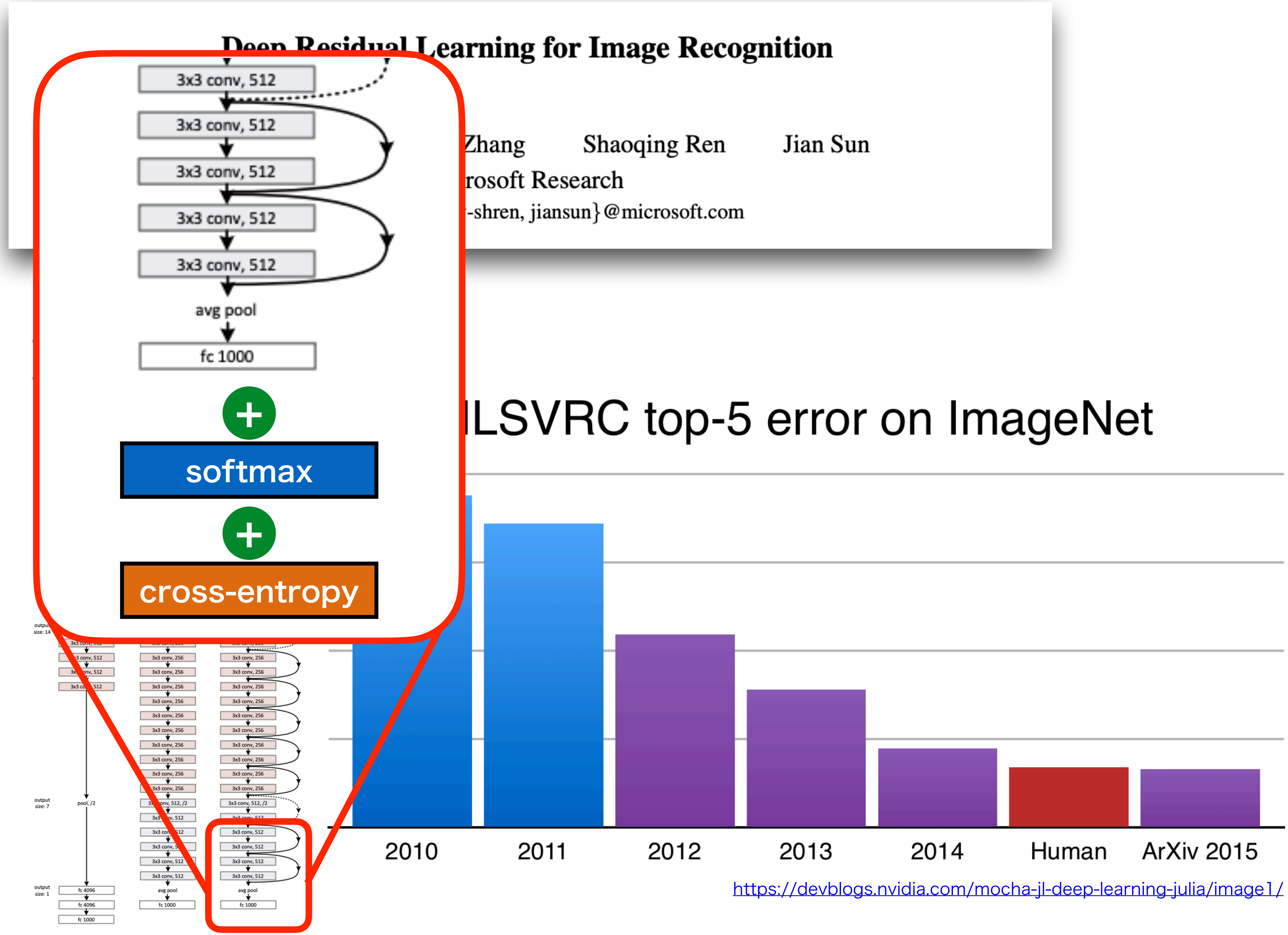
knowledge transfer

transfer learning

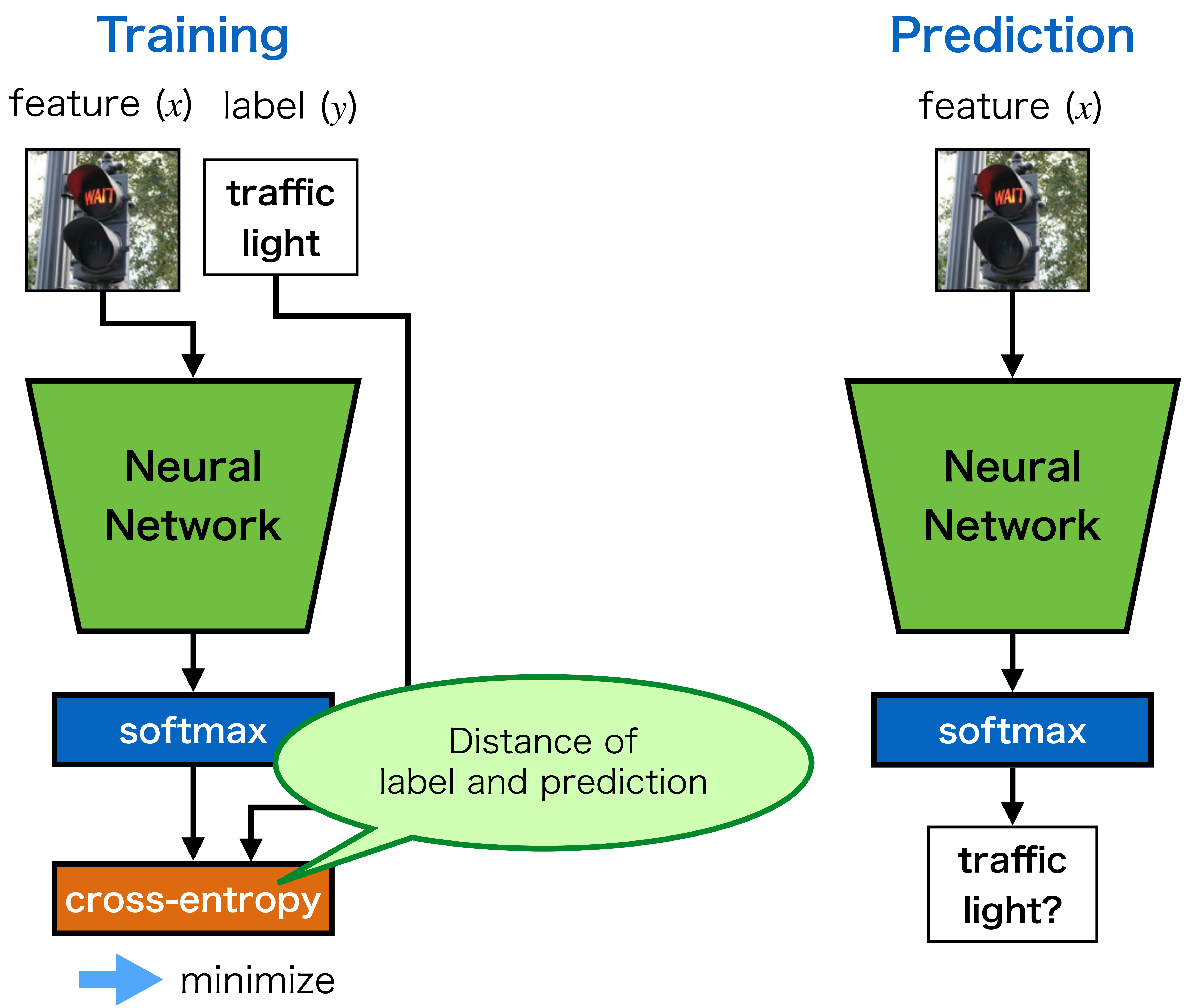
- Unsupervised Domain Adaptation Based on Source-guided Discrepancy. (AAAI2019)

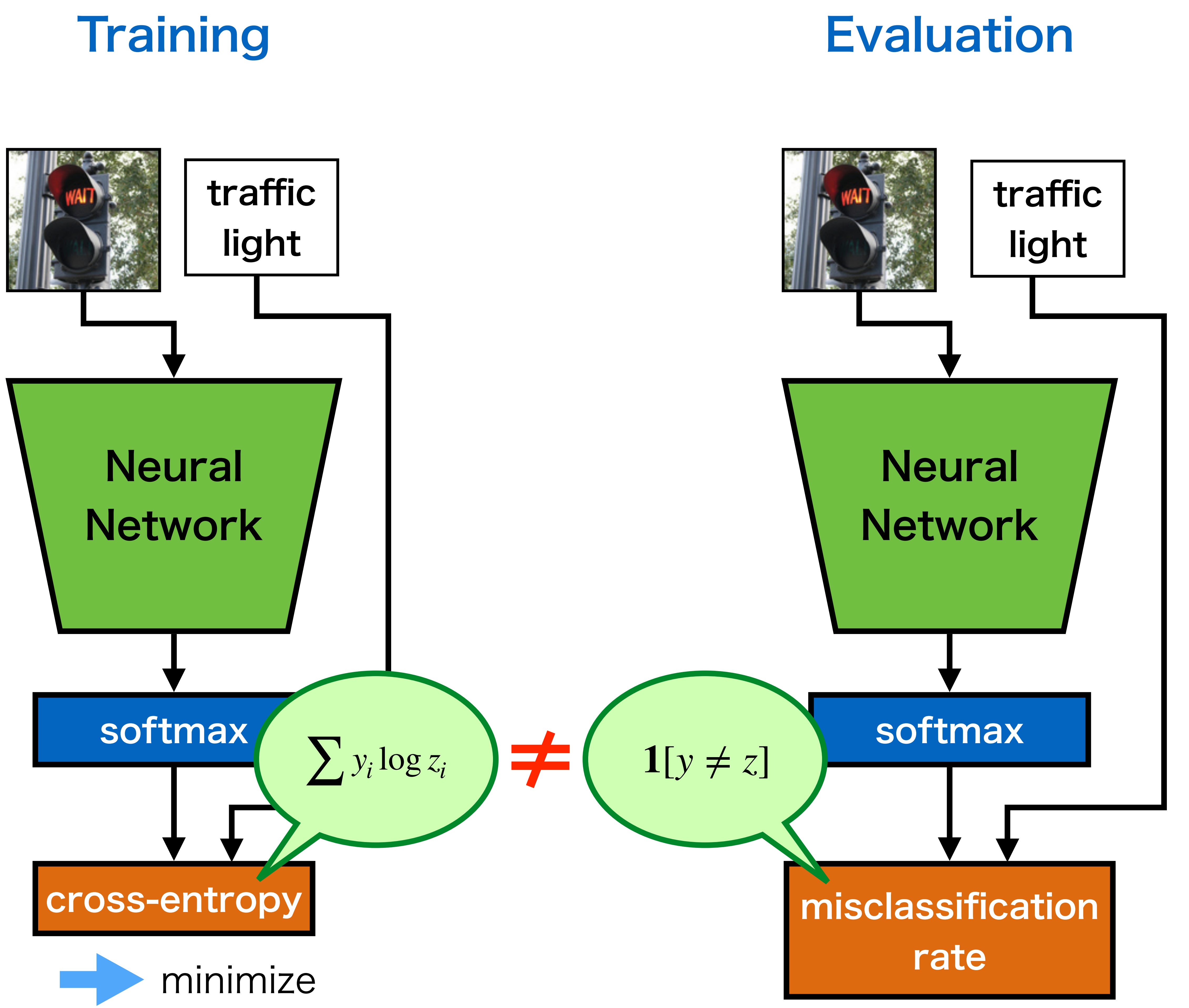
similarity learning

- Similarity-based Classification: Connecting Similarity Learning to Binary Classification. (preprint)
- Classification from Pairwise Similarity and Unlabeled Data. (ICML2018)



https://devblogs.nvidia.com/mocha-jl-deep-learning-julia/image1/

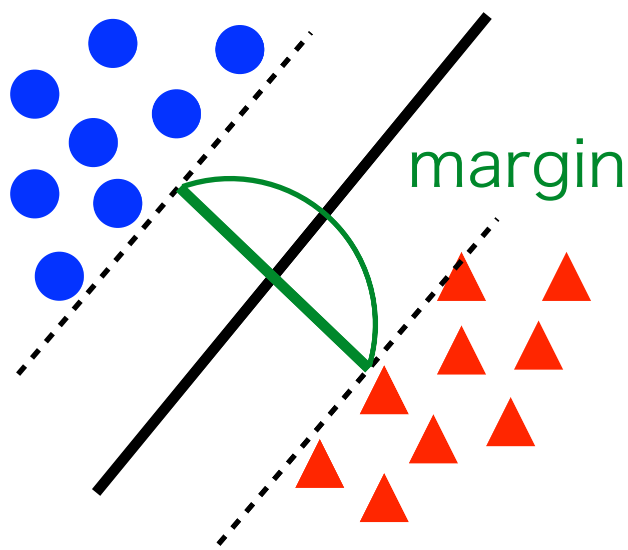

Training
feature (x)
label (y)
traffic light
Neural Network
softmax
Distance of label and prediction
cross-entropy
minimize
Prediction
feature (x)
Neural Network
softmax
traffic light?


Training
Evaluation
traffic light
traffic light
Neural Network
Neural Network
softmax
softmax
$\sum y_i \log z_i$
$\neq$
$\mathbf{1}[y \neq z]$
cross-entropy
misclassification rate
minimize

Machine Learning, 20, 273–297 (1995)

Support-Vector Networks

CORINNA CORTES corinna@neural.att.com
VLADIMIR VAPNIK vlad@neural.att.com
AT&T Bell Labs., Holmdel, NJ 07733, USA



margin maximization

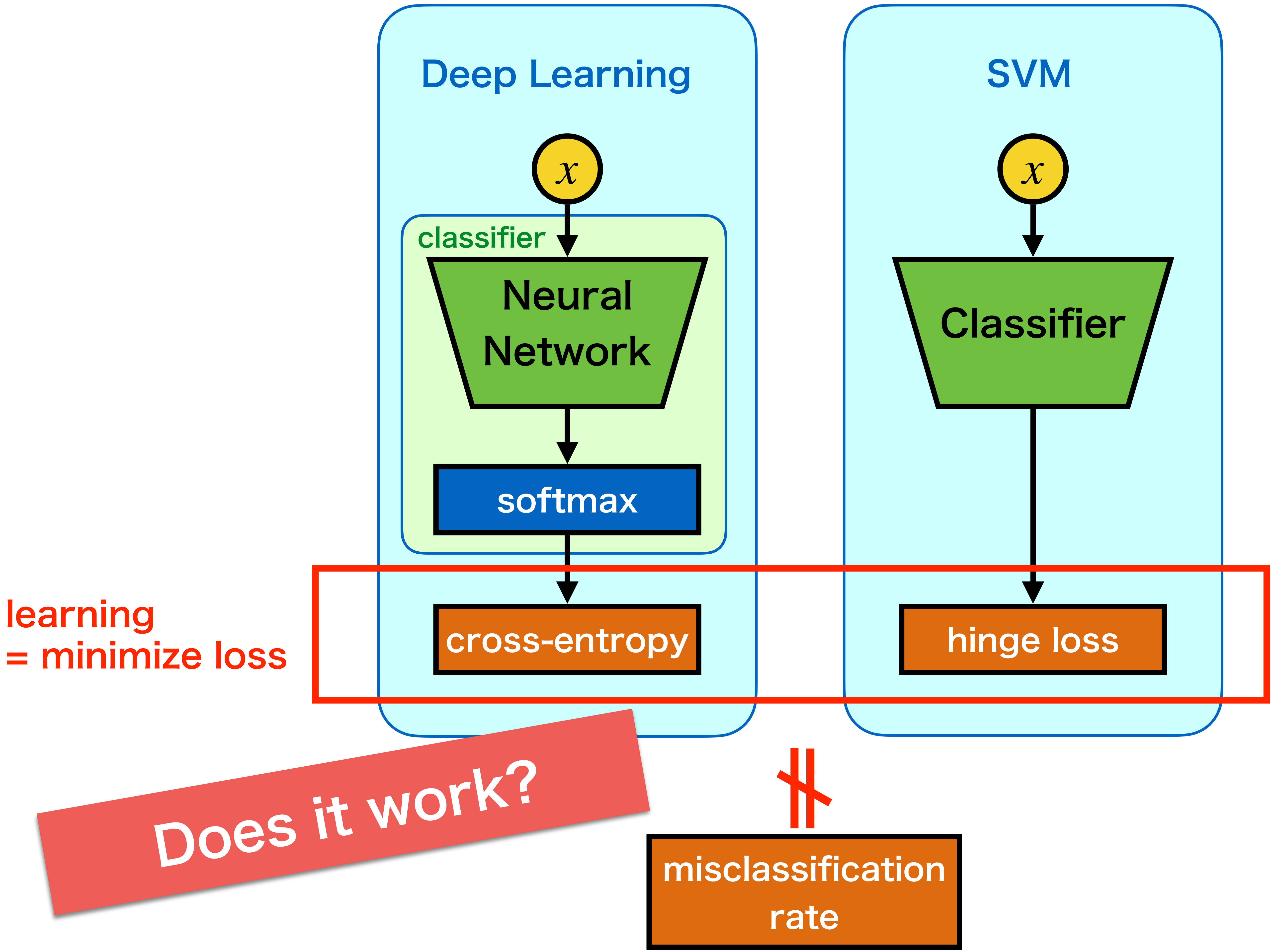
=

$$\min_{w,b} \sum_i \max\left\{0,\ 1 - y_i(w^\top x_i + b)\ \right\}$$

hinge loss minimization

$\neq$

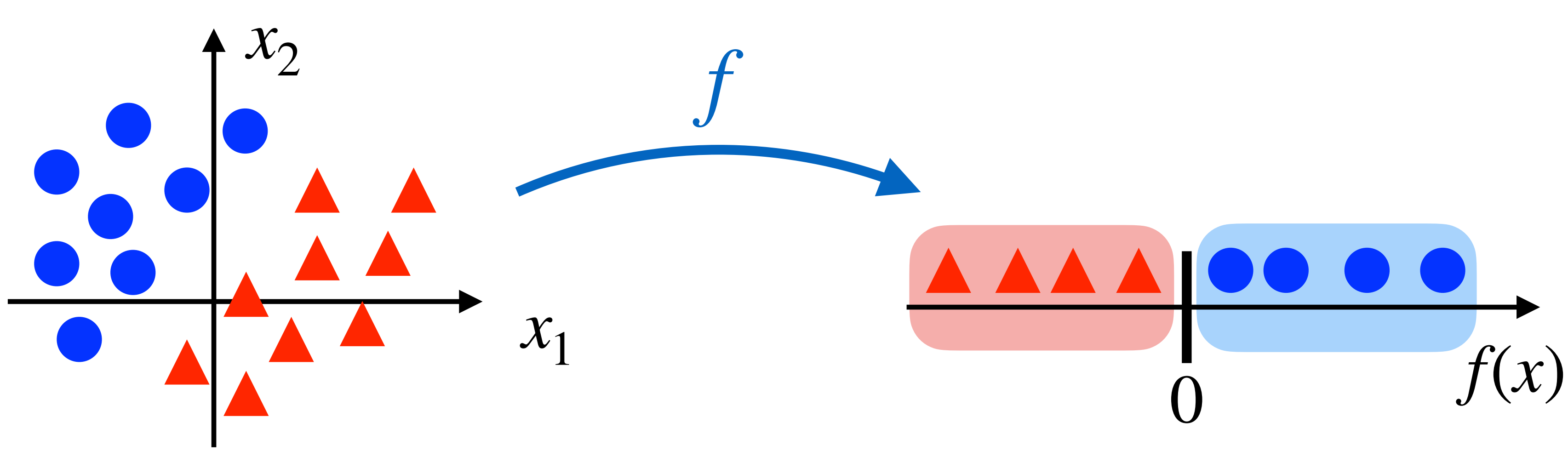
misclassification rate


Deep Learning
SVM
x
x
classifier
Neural Network
Classifier
softmax
learning
= minimize loss
cross-entropy
hinge loss
Does it work?
$\not\Leftrightarrow$
misclassification rate

Background: Binary Classification

- Input
  - sample $\{(x_i, y_i)\}_{i=1}^{n}$: pair of feature $x_i \in \mathcal{X}$ and label $y_i \in \{\pm 1\}$
- Output
  - classifier $f : \mathcal{X} \to \mathbb{R}$
  - predict class by $\mathrm{sign}(f(\cdot))$
  - criteria: misclassification rate $R_{01}(f) = \mathbb{E}\left[\mathbf{1}[Y \neq \mathrm{sign}(f(X))]\right]$

1 if $Y \neq \mathrm{sign}(f(X))$,
0 if $Y = \mathrm{sign}(f(X))$



Loss function and Risk

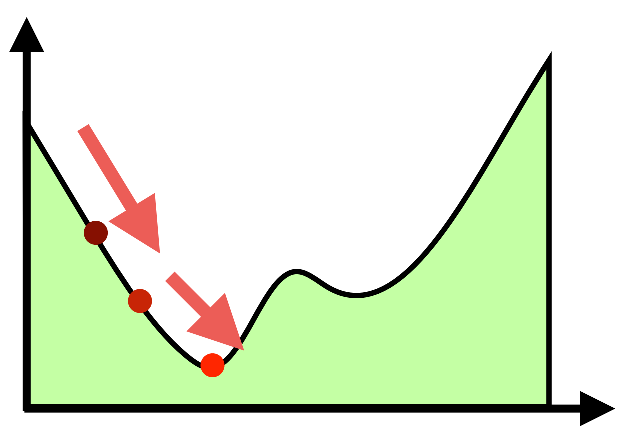
- Goal of classification: minimize misclassification rate

$$R_{01}(f) = \mathbb{E}\left[\mathbf{1}[Y \neq \mathrm{sign}(f(X))]\right]$$

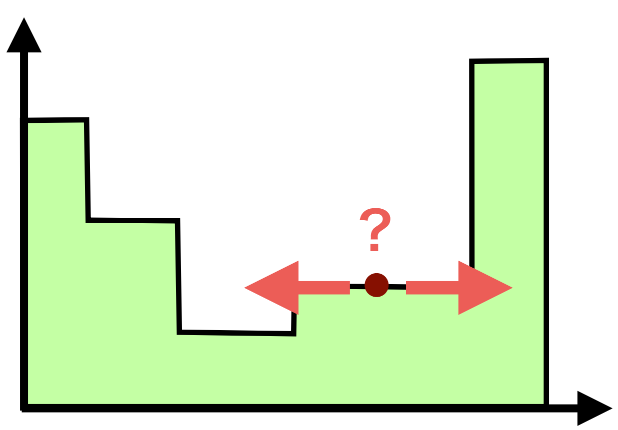
- Misclassification rate $=$ $\underbrace{\text{expectation of 0-1 loss}}$ (0-1 risk)

$$\mathbf{1}[Y \neq \mathrm{sign}(f(X))] = \phi_{01}(Yf(X))$$

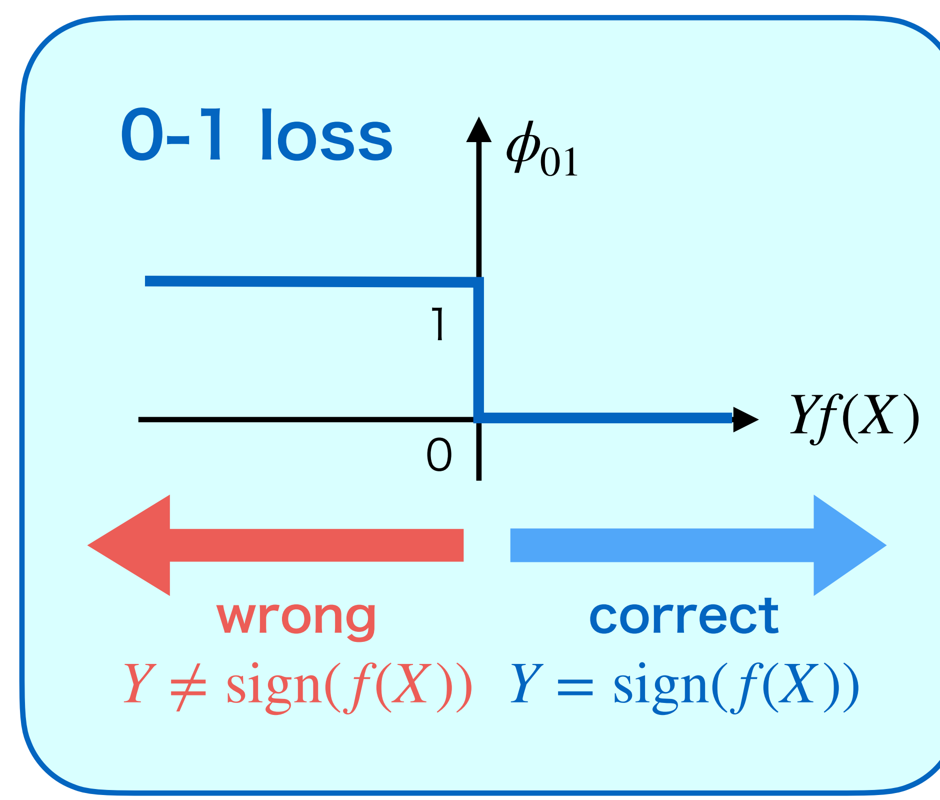
- Minimizing R_{01} is NP-hard [Feldman+ 2012]



minimization by
gradient descent



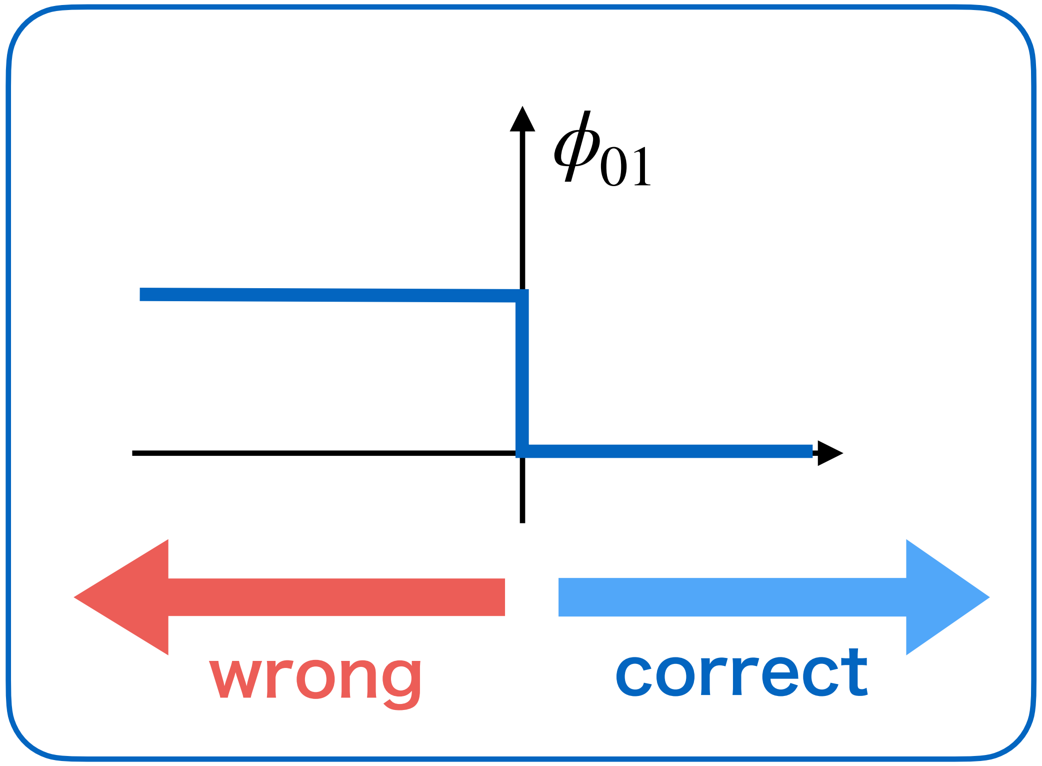
no gradient for
discrete function



Feldman, V., Guruswami, V., Raghavendra, P., & Wu, Y. (2012). Agnostic learning of monomials by halfspaces is hard. *SIAM Journal on Computing*, *41*(6), 1558-1590.

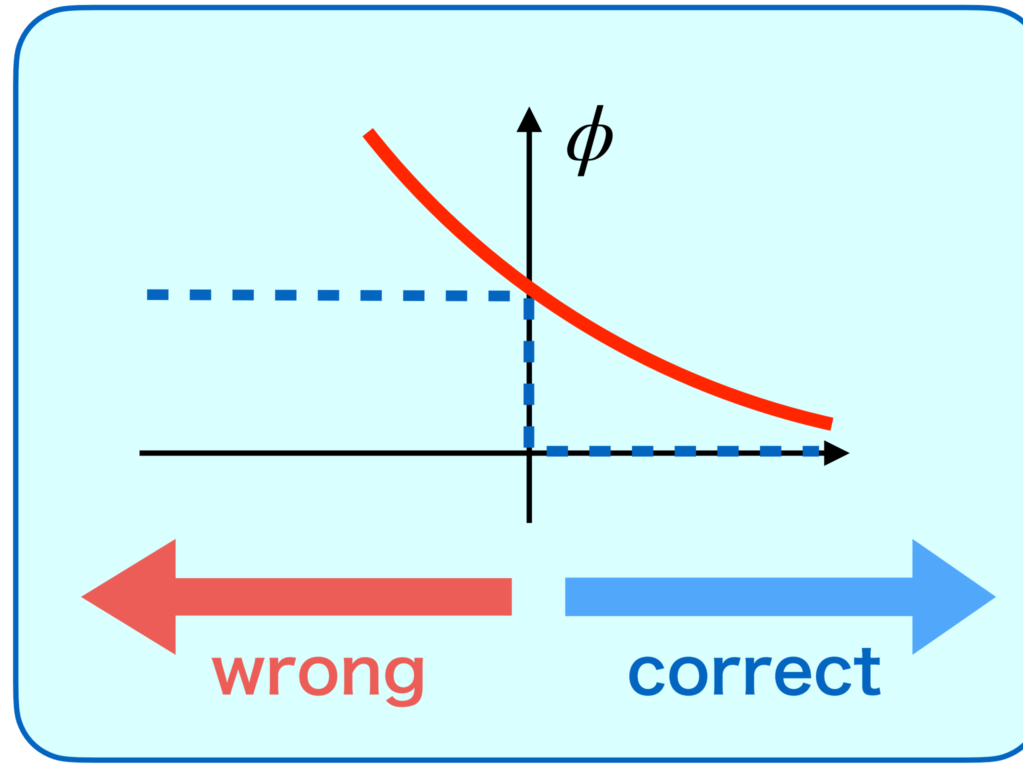
Target Loss vs. Surrogate Loss

Target loss (0-1 loss)



- Final learning criterion
- Hard to optimize
  - nonconvex, no gradient

Surrogate loss



- Different from target loss
- Easily-optimizable criterion
  - usually convex, smooth

Elements of Learning Theory

(empirical)
surrogate risk

$$\hat{R}_\phi(f) = \frac{1}{n}\sum_{i=1}^{n}\phi(y_i f(x_i))$$

Generalization theory:
If model is not too complicated,
then converges (roughly speaking)

(population)
surrogate risk

$$R_\phi(f) = \mathbb{E}[\phi(Yf(X))]$$

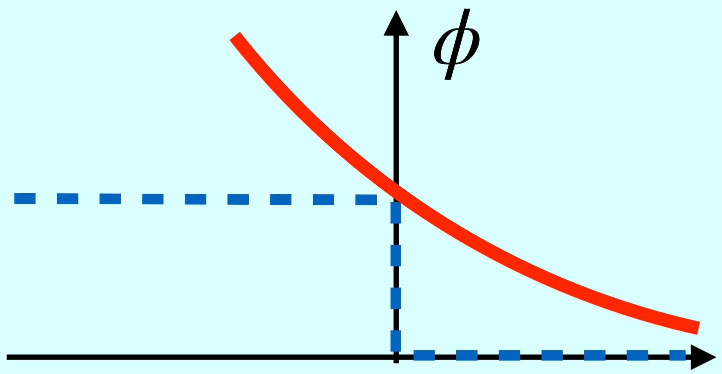
Key ingredient:
Calibration theory for
loss functions

target risk

$$R_{01}(f) = \mathbb{E}[\ell_{01}(Yf(X))]$$

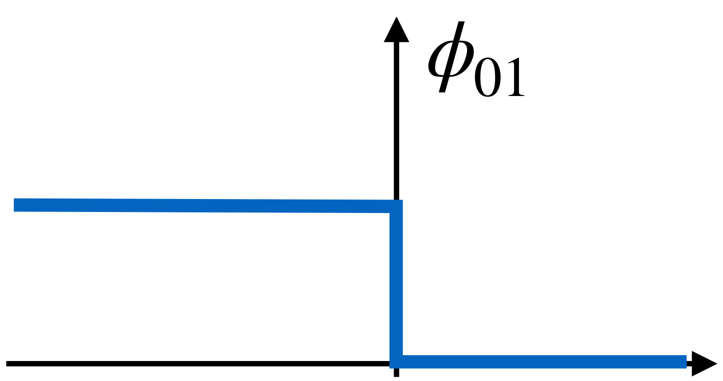
What surrogate is desirable?

Surrogate loss



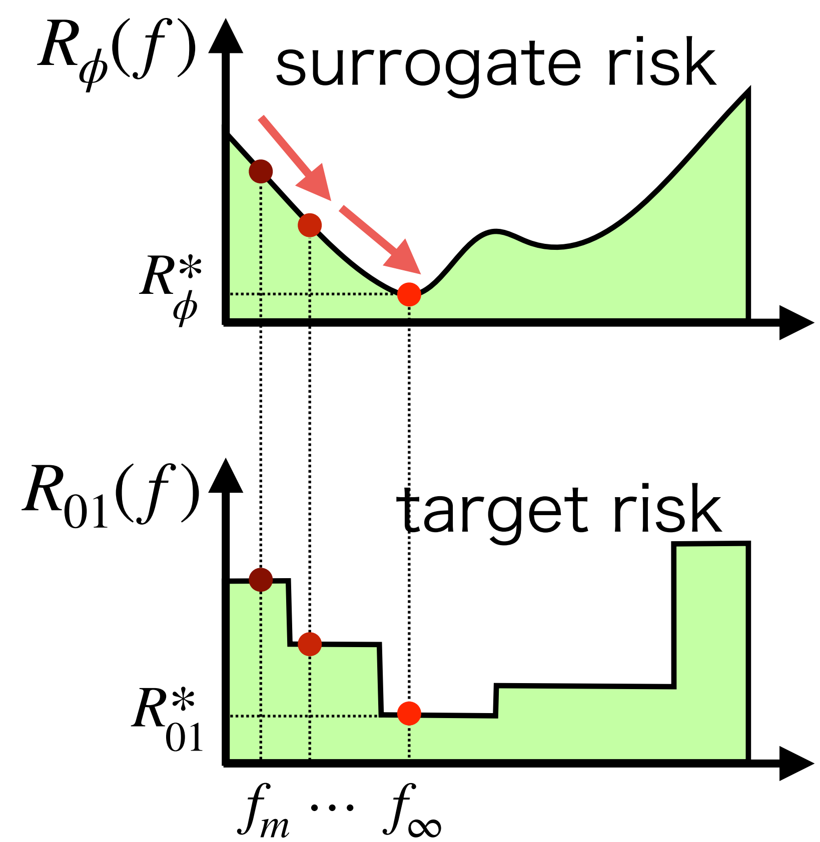
easily optimizable

Target loss (0-1 loss)



final learning criterion

Calibrated surrogate



$$R_\phi(f_m) \xrightarrow{m\to\infty} R^*_\phi \implies R_{01}(f_m) \xrightarrow{m\to\infty} R^*_{01}$$

How to check risk convergence?

[Steinwart 2007]

Definition. ϕ is ψ-**calibrated** for a target loss ψ if for any $\varepsilon > 0$, there exists $\delta > 0$ such that for all f,

$$R_\phi(f) - R_\phi^* < \delta \implies R_\psi(f) - R_\psi^* < \varepsilon.$$

surrogate (excess) risk target (excess) risk

Idea: write δ as function of ε (by using contraposition)

Definition. (calibration function)

$$\delta(\varepsilon) = \inf_f R_\phi(f) - R_\phi^* \quad \text{s.t.} \quad R_\psi(f) - R_\psi^* \geq \varepsilon$$

surrogate (excess) risk target (excess) risk

If $\delta(\varepsilon) > 0$ for all $\varepsilon > 0$, surrogate is calibrated!

Steinwart, I. (2007). How to compare different loss functions and their risks. *Constructive Approximation*, *26*(2), 225-287.

Main Tool: Calibration Function

Definition. (calibration function)

$$\delta(\varepsilon) = \inf_f \; R_\phi(f) - R_\phi^* \quad \text{s.t.} \quad R_\psi(f) - R_\psi^* \geq \varepsilon$$

surrogate (excess) risk — target (excess) risk

- Provides **iff condition**
  - ψ-calibrated $\Longleftrightarrow \delta(\varepsilon) > 0$ for all $\varepsilon > 0$
- Provides **excess risk bound**
  - ψ-calibrated $\Longrightarrow R_\psi(f) - R_\psi^* \leq (\delta^{**})^{-1}\left(R_\phi(f) - R_\phi^*\right)$

monotonically increasing

target excess risk — surrogate excess risk

δ^{**}: biconjugate of δ

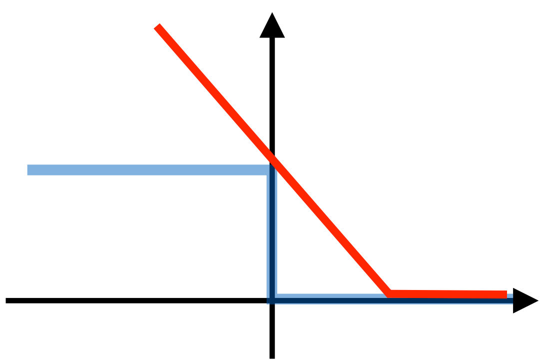
Example: Binary Classification (ϕ_{01})

[Bartlett+ 2006]

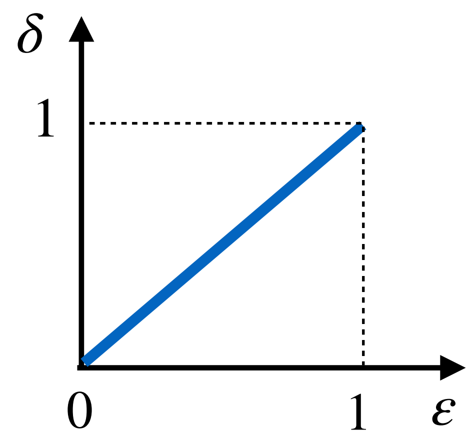
Theorem. If surrogate ϕ is convex, it is ϕ_{01}-calibrated iff

- differentiable at 0
- $\phi'(0) < 0$

hinge loss

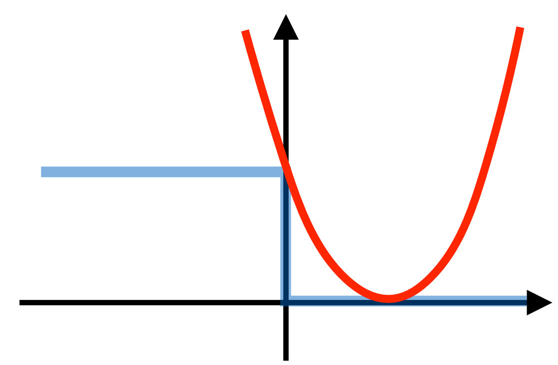


$\phi(\alpha) = [1 - \alpha]_+$

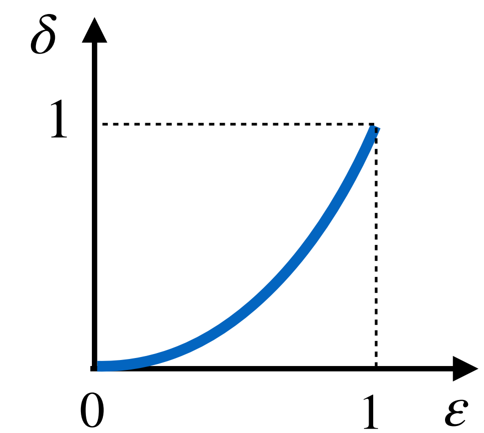


$\delta(\varepsilon) = \varepsilon$

squared loss



$\phi(\alpha) = (1 - \alpha)^2$

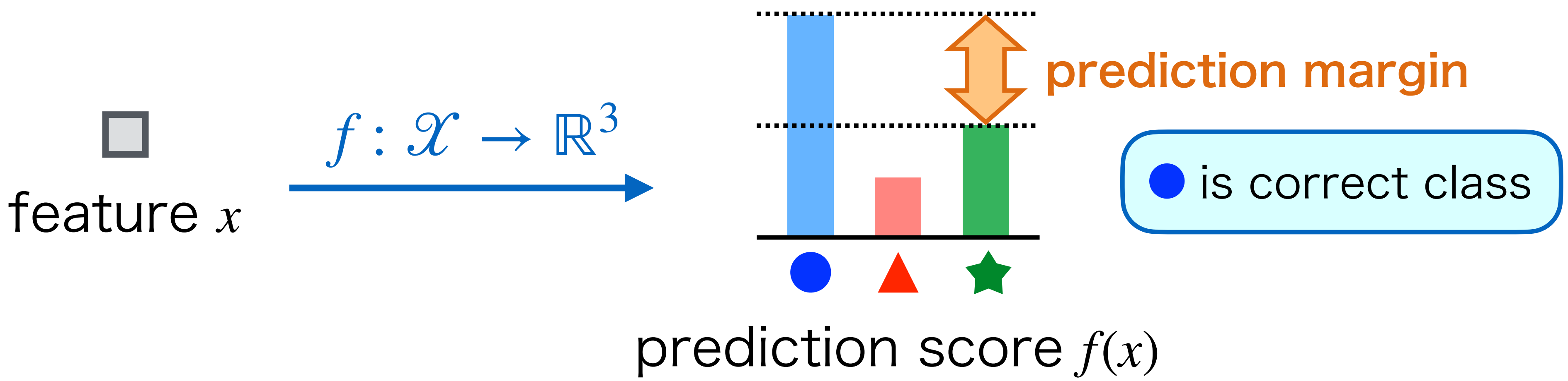


$\delta(\varepsilon) = \varepsilon^2$

P. L. Bartlett, M. I. Jordan, & J. D. McAuliffe. (2006).
Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473), 138-156.

Counterintuitive Result

- e.g. multi-class classification $\Rightarrow$ maximize prediction margin



Crammer-Singer loss [Crammer & Singer 2001]

$\max\{0, 1 -$ pred. margin$\}$

one of multi-class extensions of hinge loss

Crammer-Singer loss is not calibrated to 0-1 loss !

(similar extension of logistic loss is calibrated)

[Zhang 2004]

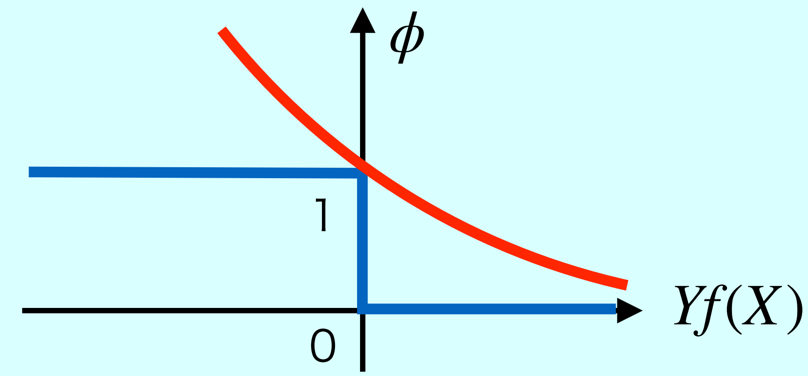
Crammer, K., & Singer, Y. (2001). On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of machine learning research*, 2(Dec), 265-292

Zhang, T. (2004). Statistical analysis of some multi-category large margin classification methods. *Journal of Machine Learning Research*, 5(Oct), 1225-1251.

Summary: Calibration Theory

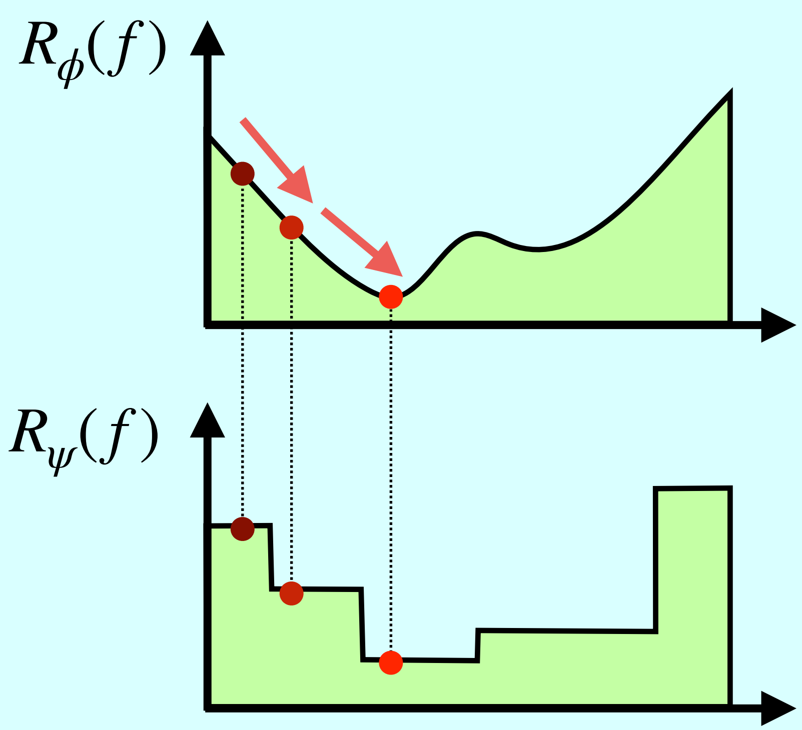
Surrogate vs. Target loss

Target loss is often hard to optimize
⇒ replace with **surrogate** loss



Calibrated Surrogate

leading to minimization of target



Binary Classification

Hinge, logistic is calibrated
Calibrated iff $\phi'(0) < 0$

Multi-class Classification

CS-loss (MC-hinge loss) is not calibrated!

cross-entropy is calibrated (omitted)

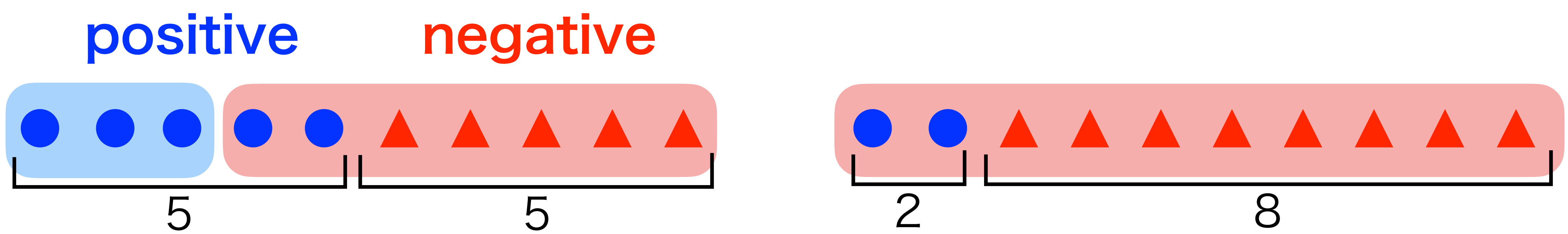
Stringent justification of surrogate loss!

When target is not 0-1 loss

H. Bao and M. Sugiyama.
Calibrated Surrogate Maximization of Linear-fractional Utility in Binary Classification. In *AISTATS*, 2020.

Is accuracy appropriate?

- Our focus: **binary classification**



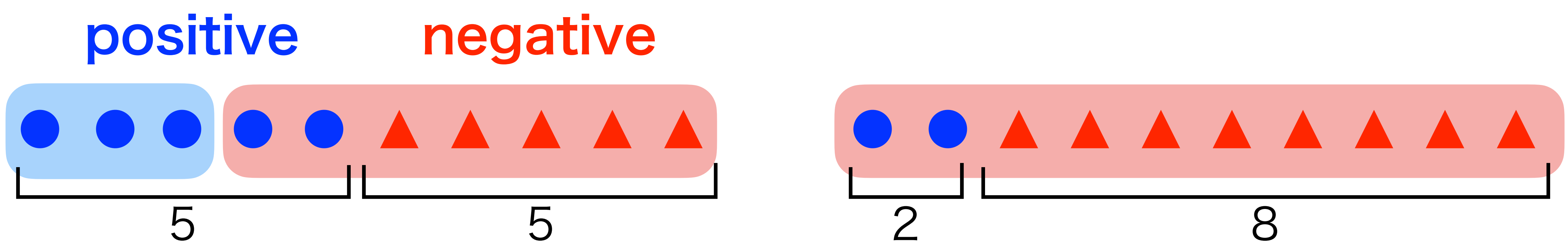
seemingly sensible classifier | unreasonable classifier

accuracy: **0.8** | accuracy: **0.8**

Accuracy can’t detect unreasonable classifiers under **class imbalance**!

Is accuracy appropriate?

- F-measure is more appropriate under **class imbalance**



accuracy: **0.8**

F-measure: **0.75**

accuracy: **0.8**

F-measure: **0**

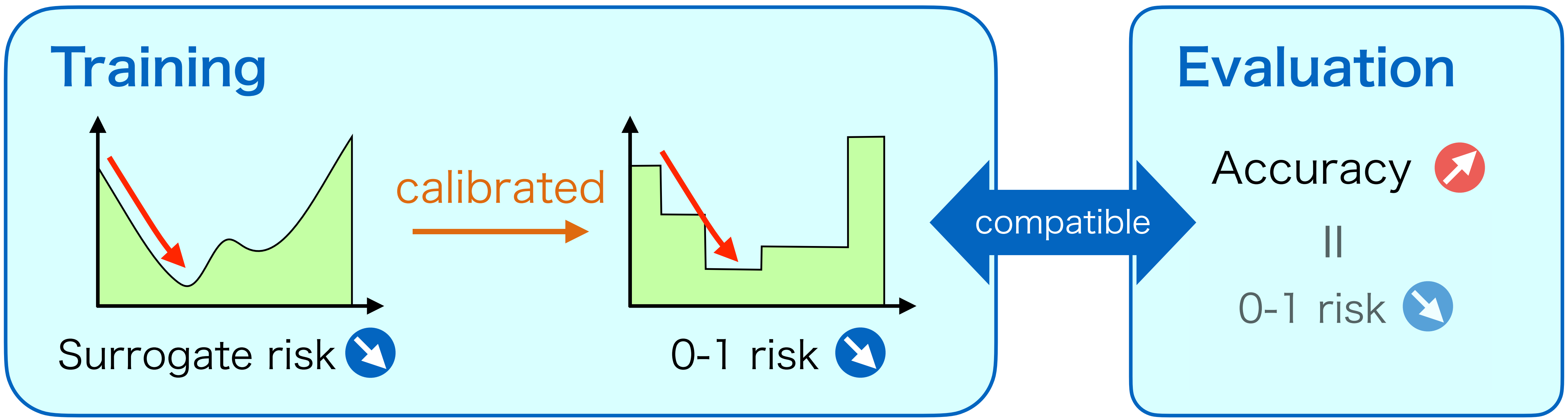
F-measure $$\mathrm{F}_1 = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

$$\mathrm{TP} = \mathbb{E}_{X,Y=+1}[1_{\{f(X)>0\}}] \qquad \mathrm{TN} = \mathbb{E}_{X,Y=-1}[1_{\{f(X)<0\}}]$$

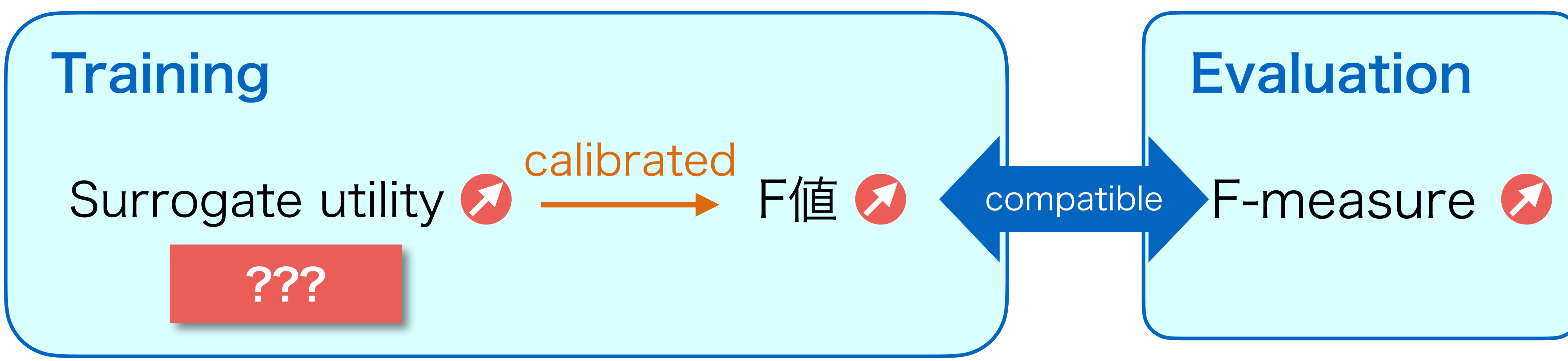
$$\mathrm{FP} = \mathbb{E}_{X,Y=-1}[1_{\{f(X)>0\}}] \qquad \mathrm{FN} = \mathbb{E}_{X,Y=+1}[1_{\{f(X)<0\}}]$$

Training and Evaluation

- Usual training with accuracy



- Training with accuracy but evaluating with F-measure



Not only F_1, but many others

Q. Can we handle in the same way?

Accuracy

$$\mathrm{Acc} = \mathrm{TP} + \mathrm{TN}$$

Weighted Accuracy

$$\mathrm{WAcc} = \frac{w_1\mathrm{TP} + w_2\mathrm{TN}}{w_1\mathrm{TP} + w_2\mathrm{TN} + w_3\mathrm{FP} + w_4\mathrm{FN}}$$

F-measure

$$\mathrm{F}_1 = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

Balanced Error Rate

$$\mathrm{BER} = \frac{1}{\pi}\mathrm{FN} + \frac{1}{1-\pi}\mathrm{FP}$$

Gower-Legendre index

$$\mathrm{GLI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \alpha(\mathrm{FP} + \mathrm{FN}) + \mathrm{TN}}$$

Jaccard index

$$\mathrm{Jac} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

Unification of Metrics

Note:

$\mathrm{TN} = \mathbb{P}(Y = -1) - \mathrm{FP}$

$\mathrm{FN} = \mathbb{P}(Y = +1) - \mathrm{TP}$

Actual Metrics

$$\mathrm{F}_1 = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

$$\mathrm{Jac} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

linear-fraction

$$U(f) = \frac{a_0\mathrm{TP} + b_0\mathrm{FP} + c_0}{a_1\mathrm{TP} + b_1\mathrm{FP} + c_1}$$

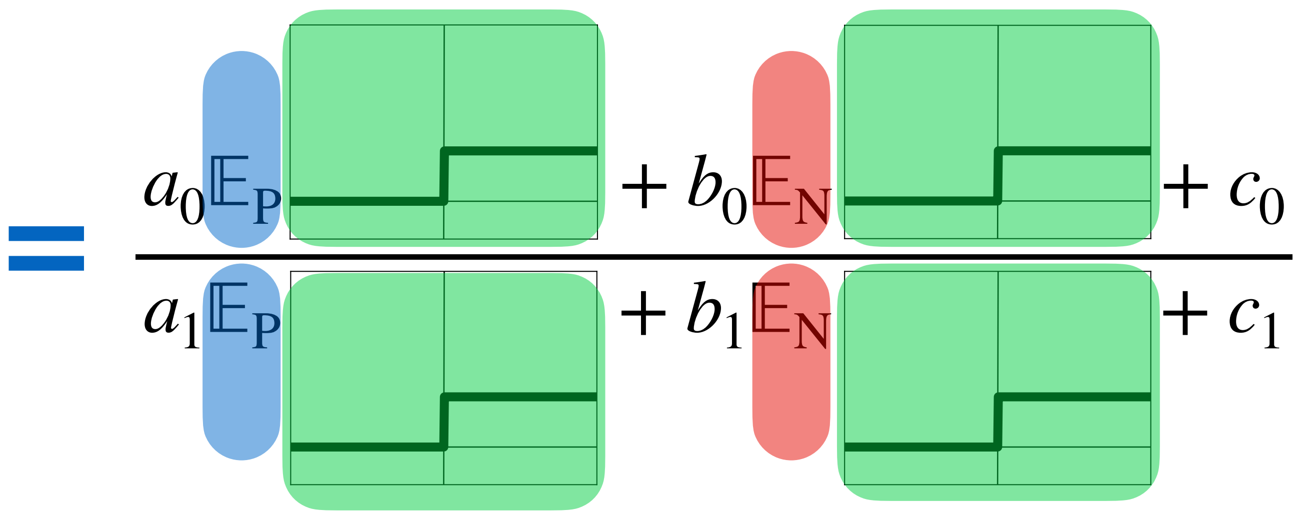
a_k, b_k, c_k : constants

Unification of Metrics

linear-fraction

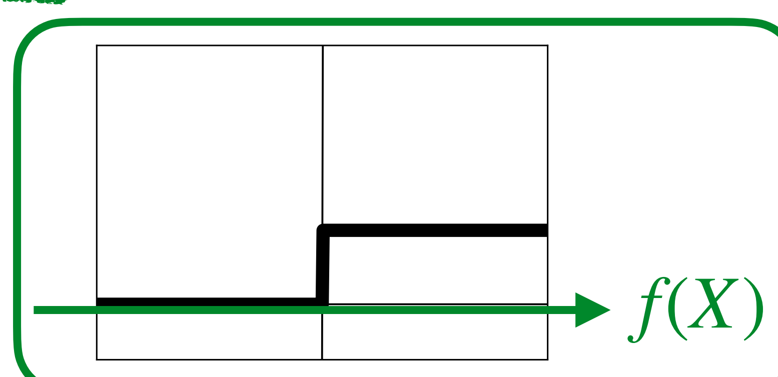
$$U(f) = \frac{a_0\mathrm{TP} + b_0\mathrm{FP} + c_0}{a_1\mathrm{TP} + b_1\mathrm{FP} + c_1}$$

expectation divided by expecation



$$:= \frac{\mathbb{E}_X[W_0(f(X))]}{\mathbb{E}_X[W_1(f(X))]}$$

- TP, FP = expectation of 0/1-loss
  - TP $= \mathbb{E}_{X,Y=+1}\big[\mathbf{1}[f(X)>0]\big]$
    positive data && positive prediction
  - FP $= \mathbb{E}_{X,Y=-1}\big[\mathbf{1}[f(X)>0]\big]$
    negative data && positive prediction



Goal of This Talk

Given a metric (utility) $U(f) = \dfrac{a_0\mathrm{TP} + b_0\mathrm{FP} + c_0}{a_1\mathrm{TP} + b_1\mathrm{FP} + c_1}$

Q. How to optimize $U(f)$ directly?

- without estimating class-posterior probability

labeled sample $\{(x_i, y_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} \mathbb{P}$

metric U

→

classifier $f : \mathcal{X} \to \mathbb{R}$

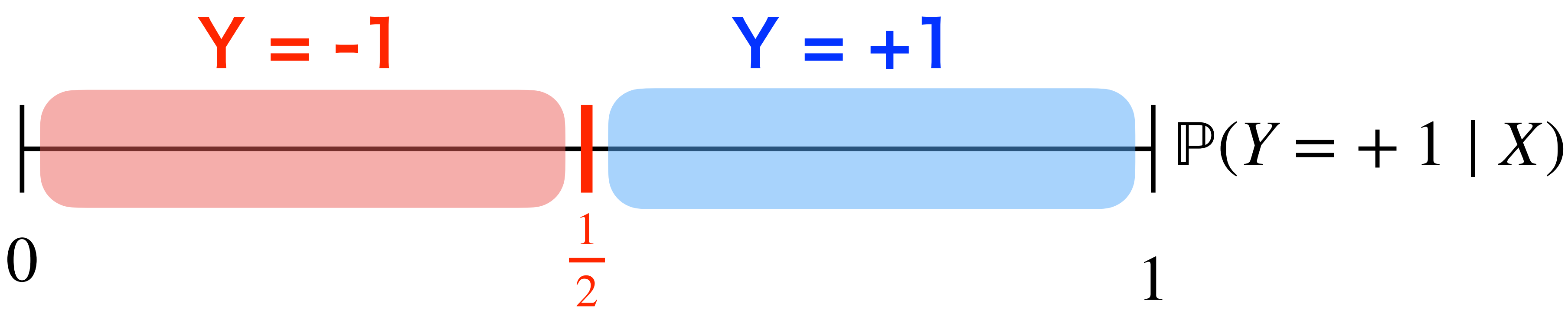
s.t. $U(f) = \sup_{f'} U(f')$

Related: Plug-in Classifier

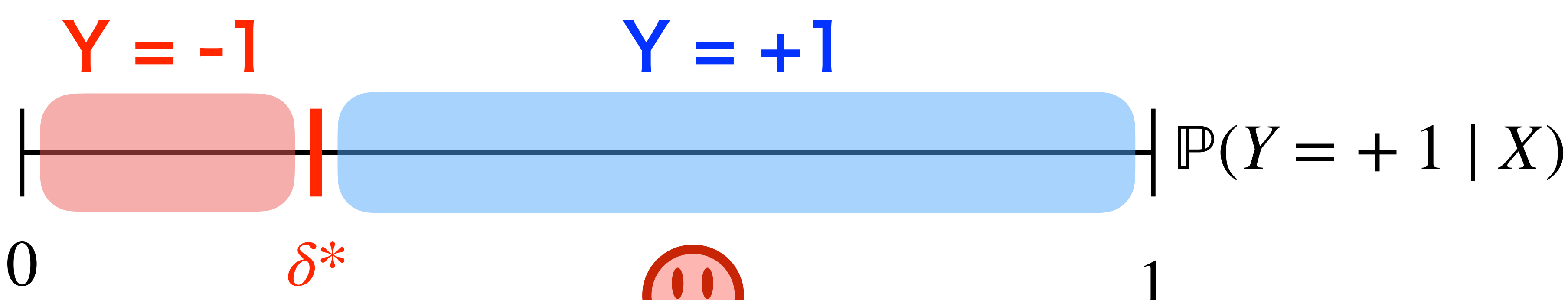
[Koyejo+ NIPS2014; Yan+ ICML2018]

- Estimating class-posterior probability is costly!

Bayes-optimal classifier (accuracy): $\mathbb{P}(Y=+1\,|\,x)-\frac{1}{2}$



Bayes-optimal classifier (general case): $\mathbb{P}(Y=+1\,|\,x)-\delta^*$



$\Rightarrow$ estimate $\boxed{\mathbb{P}(Y=+1\,|\,x)}$ and δ^* independently

O. O. Koyejo, N. Natarajan, P. K. Ravikumar, & I. S. Dhillon.
Consistent binary classification with generalized performance metrics. In *NIPS*, 2014.

B. Yan, O. Koyejo, K. Zhong, & P. Ravikumar.
Binary classification with Karmic, threshold-quasi-concave metrics. In *ICML*, 2018.

Convexity & Statistical Property

Q. How to make tractable surrogate?

Accuracy

tractable (convex)

$$R_\phi(f) = \mathbb{E}[\phi(Yf(X))]$$

$$\Big\downarrow \text{ calibrated}$$

intractable

$$R_{01}(f) = \mathbb{E}[\phi_{01}(Yf(X))]$$

Linear-fractional Metrics

① tractable?

? ? ?

$$\Big\downarrow \text{ ② calibrated?}$$

intractable

$$U(f) = \frac{\mathbb{E}_X[W_0(f(X))]}{\mathbb{E}_X[W_1(f(X))]}$$

Non-concave, but quasi-concave

Idea: $\dfrac{\text{concave}}{\text{convex}}$ = quasi-concave

$\dfrac{f(x)}{g(x)}$ is quasi-concave

if f : concave, g : convex,

$f(x) \geq 0$ and $g(x) > 0$ for $\forall x$

(proof) Show $\{x \mid f/g \geq \alpha\}$ is convex.

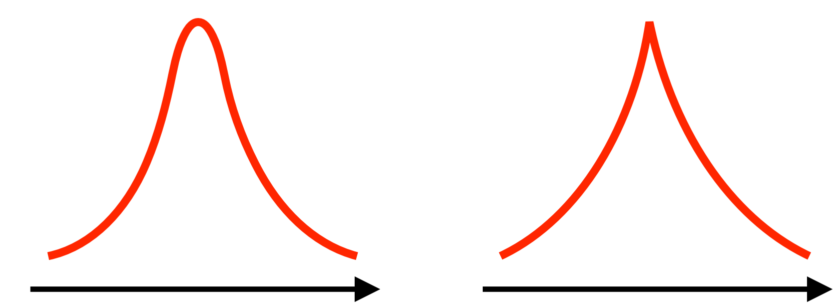
$$\frac{f(x)}{g(x)} \geq \alpha \iff \underbrace{f(x) - \alpha g(x)}_{\text{concave}} \geq 0$$

NB: super-level set of concave func. is convex

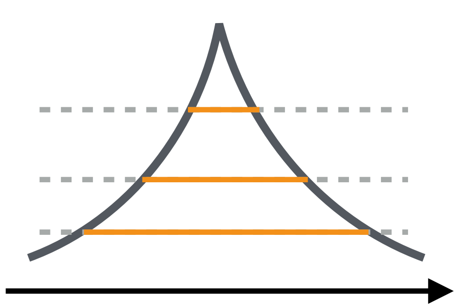
$\therefore \{x \mid f/g \geq \alpha\}$ is convex for $\forall \alpha \geq 0$

non-concave, but unimodal

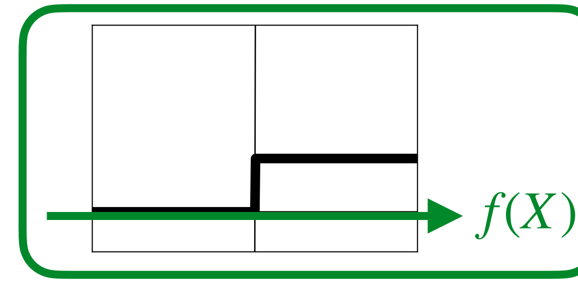
⇒ efficiently optimized



- quasi-concave ⊇ concave
- super-levels are convex



Surrogate Utility

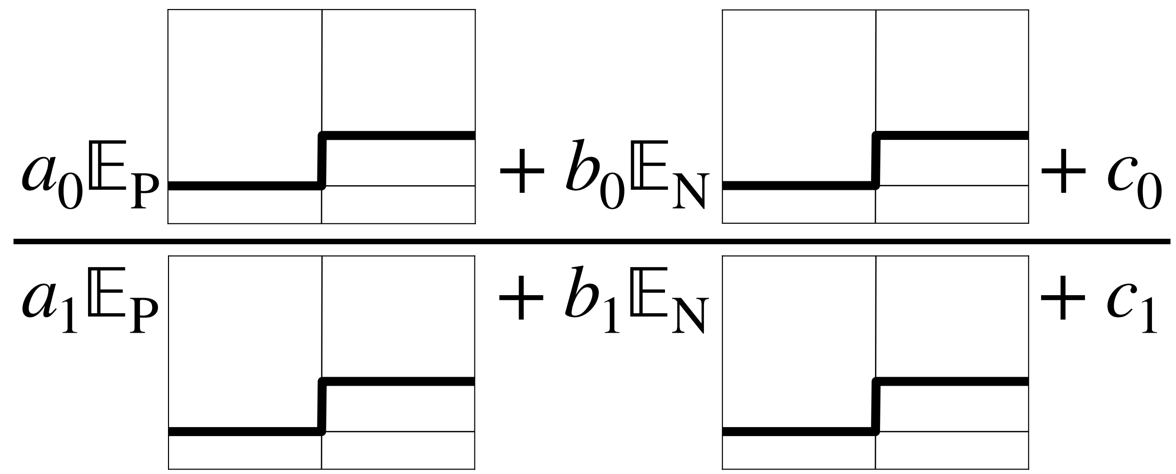


- Idea: bound true utility from below

linear-fraction

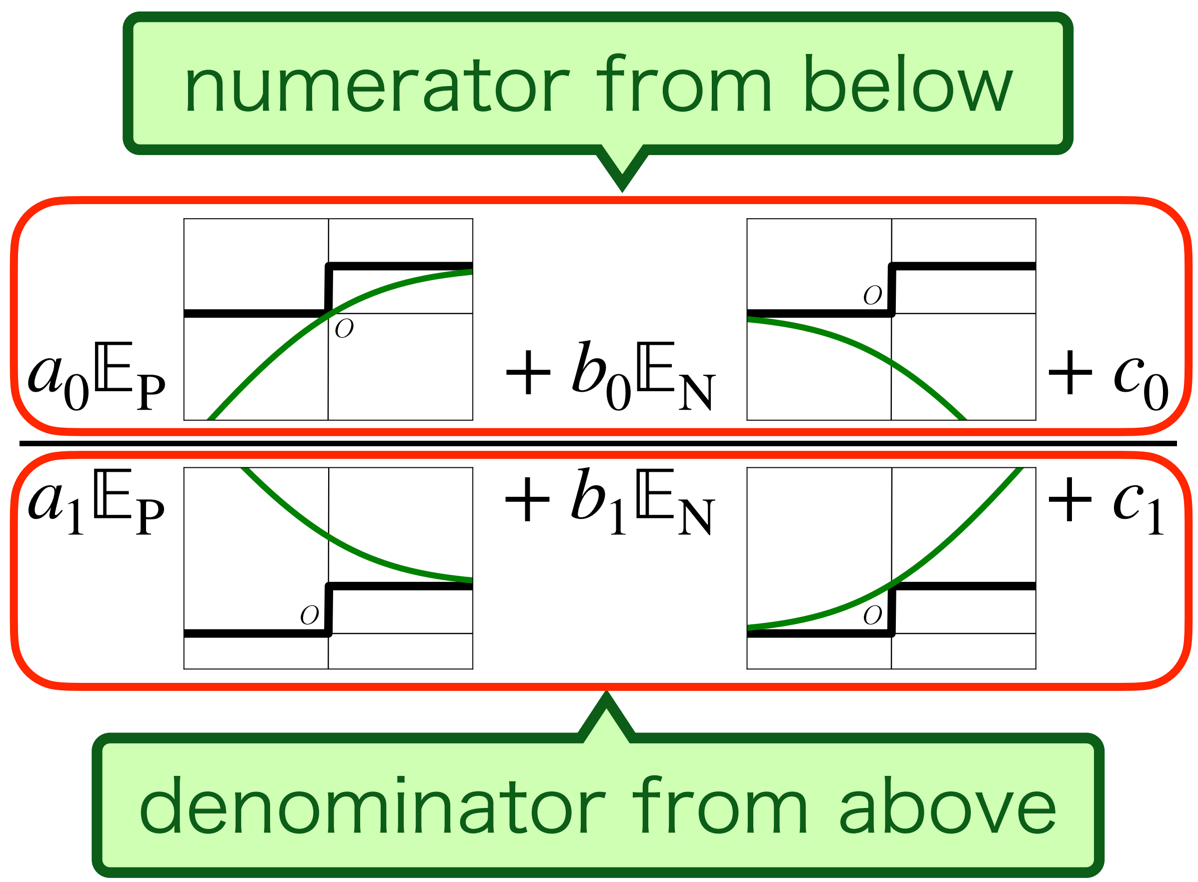
$$U(f) = \frac{a_0\mathrm{TP} + b_0\mathrm{FP} + c_0}{a_1\mathrm{TP} + b_1\mathrm{FP} + c_1}$$

=



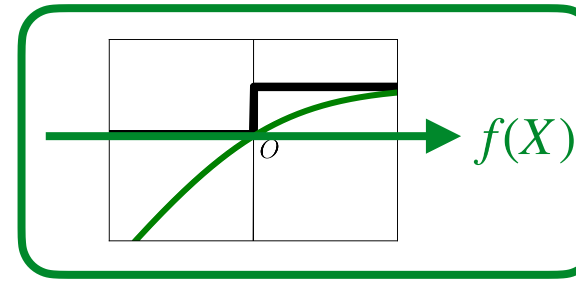
non-negative sum of concave
⇒ concave

≥



non-negative sum of convex
⇒ convex

Surrogate Utility

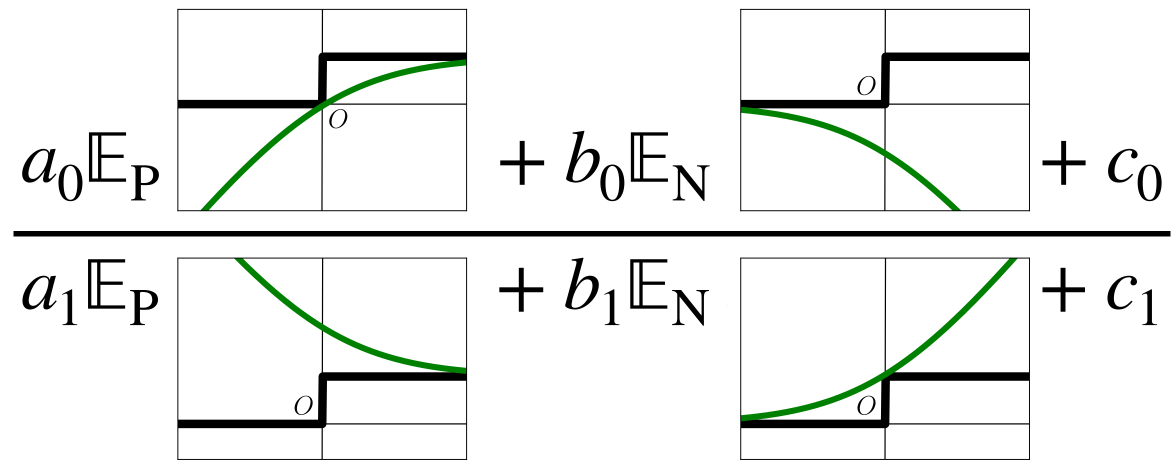


- Idea: bound true utility from below

linear-fraction

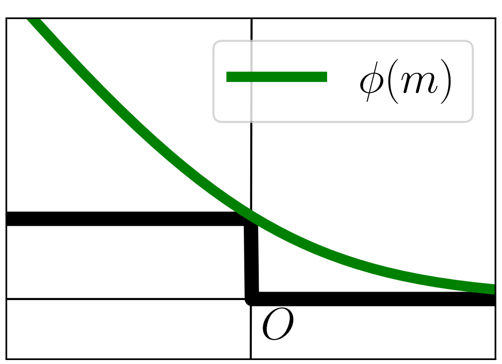
$$U(f) = \frac{a_0\mathrm{TP} + b_0\mathrm{FP} + c_0}{a_1\mathrm{TP} + b_1\mathrm{FP} + c_1}$$

$\geq$



$\|$

surrogate loss



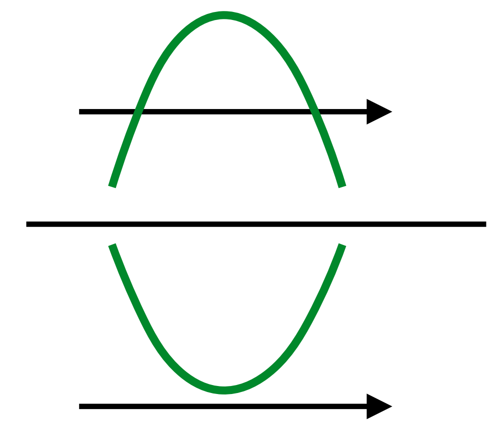
$$U_\phi(f) = \frac{a_0\mathbb{E}_\mathrm{P}[1 - \phi(f(X))] + b_0\mathbb{E}_\mathrm{N}[-\phi(-f(X))] + c_0}{a_1\mathbb{E}_\mathrm{P}[1 + \phi(f(X))] + b_1\mathbb{E}_\mathrm{N}[\ \phi(-f(X))\ ] + c_1}$$

$$:= \frac{\mathbb{E}[W_{0,\phi}]}{\mathbb{E}[W_{1,\phi}]}$$

: Surrogate Utility

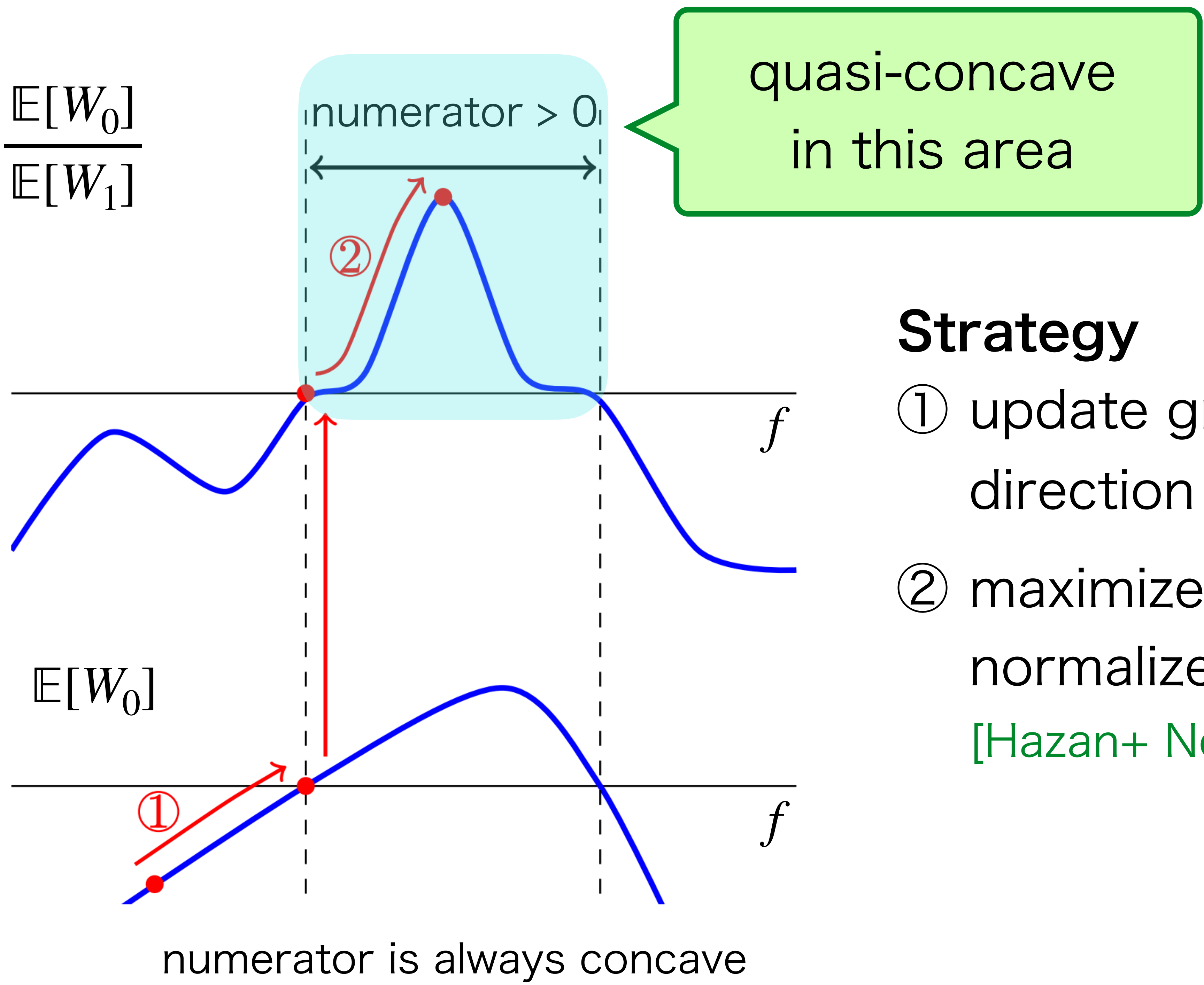
Hybrid Optimization Strategy

$$U_\phi(f) = \frac{a_0 \mathbb{E}_{\mathrm{P}} + b_0 \mathbb{E}_{\mathrm{N}} + c_0}{a_1 \mathbb{E}_{\mathrm{P}} + b_1 \mathbb{E}_{\mathrm{N}} + c_1} =$$



- Note: numerator can be negative
  - U_ϕ isn't quasi-concave only if numerator < 0
  - make numerator positive first (concave), then maximize fractional form (quasi-concave)

Hybrid Optimization Strategy



Strategy

① update gradient-ascent direction while $\mathbb{E}[W_0] < 0$

② maximize fraction by normalized-gradient ascent [Hazan+ NeurIPS2015]

Hazan, E., Levy, K., & Shalev-Shwartz, S. (2015). Beyond convexity: Stochastic quasi-convex optimization. In *Advances in Neural Information Processing Systems* (pp. 1594-1602).

Convexity & Statistical Property

Q. How to make surrogate calibrated?

Accuracy

tractable (convex)

$$R_\phi(f) = \mathbb{E}[\phi(Yf(X))]$$

calibrated

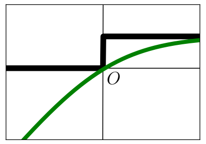
intractable

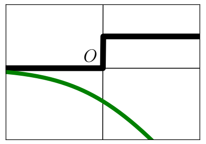
$$R_{01}(f) = \mathbb{E}[\phi_{01}(Yf(X))]$$

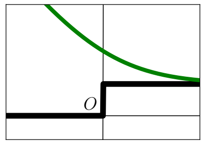
Linear-fractional Metrics

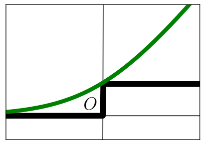
① tractable

$$\frac{a_0\mathbb{E}_{\mathrm{P}}\cdot \quad + b_0\mathbb{E}_{\mathrm{N}} \quad + c_0}{a_1\mathbb{E}_{\mathrm{P}}\cdot \quad + b_1\mathbb{E}_{\mathrm{N}} \quad + c_1}$$









② calibrated?

intractable

$$U(f) = \frac{\mathbb{E}_X[W_0(f(X))]}{\mathbb{E}_X[W_1(f(X))]}$$

Special Case: F1-measure

Theorem

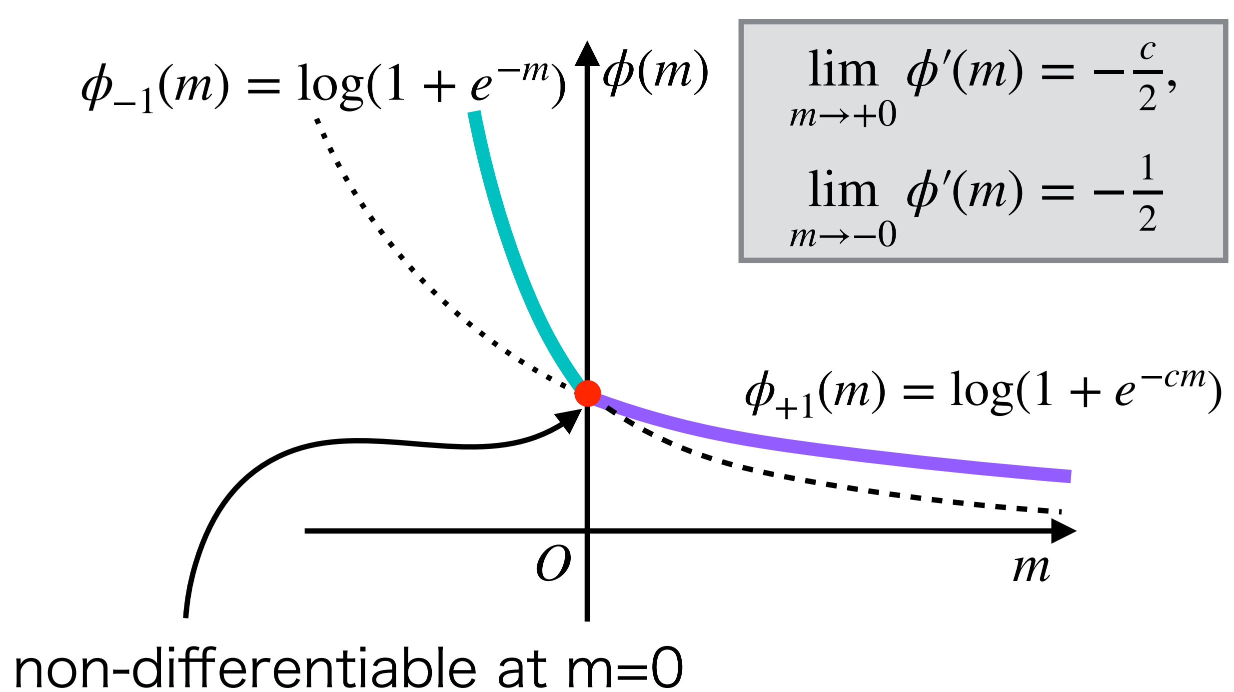
$U_\phi(f_n) \overset{n\to\infty}{\to} 1 \implies U(f_n) \overset{n\to\infty}{\to} 1 \quad \forall\{f_n\}$

if ϕ satisfies

- $\exists c \in (0,1)$ s.t. $\sup_f U_\phi(f) \geq \frac{2c}{1-c}$, $\lim_{m\to+0} \phi'(m) \geq c \lim_{m\to-0} \phi'(m)$
- ϕ is non-increasing
- ϕ is convex

Note: informal

■ Example



Intuition:
trade off TP and FP
by gradient steepness

Experiment: F_1-measure

(F_1-measure)	Proposed		Baselines		
Dataset	U-GD	U-BFGS	ERM	W-ERM	Plug-in
adult	0.617 (101)	0.660 (11)	0.639 (51)	0.676 (18)	**0.681 (9)**
australian	**0.843 (41)**	**0.844 (45)**	0.820 (123)	0.814 (116)	0.827 (51)
breast-cancer	**0.963 (31)**	**0.960 (32)**	0.950 (37)	0.948 (44)	0.953 (40)
cod-rna	0.802 (231)	0.594 (4)	0.927 (7)	0.927 (6)	**0.930 (2)**
diabetes	**0.834 (32)**	**0.828 (31)**	0.817 (50)	0.821 (40)	0.820 (42)
fourclass	**0.638 (70)**	**0.638 (64)**	0.601 (124)	0.591 (212)	0.618 (64)
german.numer	0.561 (102)	**0.580 (74)**	0.492 (188)	0.560 (107)	**0.589 (73)**
heart	**0.796 (101)**	**0.802 (99)**	**0.792 (80)**	0.764 (151)	0.764 (137)
ionosphere	**0.908 (49)**	**0.901 (43)**	0.883 (104)	0.842 (217)	**0.897 (54)**
madelon	**0.666 (19)**	0.632 (67)	0.491 (293)	0.639 (110)	**0.663 (24)**
mushrooms	1.000 (1)	0.997 (7)	**1.000 (1)**	1.000 (2)	0.999 (4)
phishing	0.937 (29)	**0.943 (7)**	**0.944 (8)**	0.940 (12)	**0.944 (8)**
phoneme	**0.648 (27)**	0.559 (22)	0.530 (201)	0.616 (135)	0.633 (35)
skin_nonskin	0.870 (3)	0.856 (4)	0.854 (7)	**0.877 (8)**	0.838 (5)
sonar	**0.735 (95)**	**0.740 (91)**	0.706 (121)	0.655 (189)	**0.721 (113)**
spambase	0.876 (27)	0.756 (61)	0.887 (42)	0.881 (58)	**0.903 (18)**
splice	0.785 (49)	**0.799 (46)**	0.785 (55)	0.771 (67)	**0.801 (45)**
w8a	0.297 (80)	0.284 (96)	0.735 (35)	**0.742 (29)**	**0.745 (26)**

(F_1-measure is shown)

model: $f_\theta(x) = \theta^\top x$

surrogate loss: $\phi(m) = \max\{\log(1+e^{-m}), \log(1+e^{-\frac{m}{3}})\}$

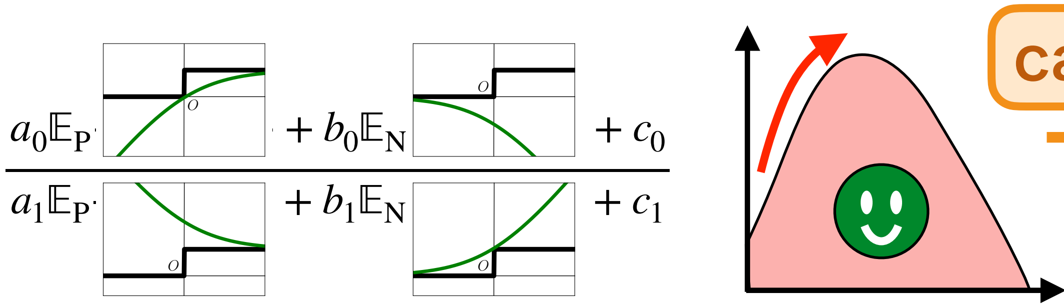
Loss for Complicated Metrics

Linear-fractional metrics

contains F-measure, Jaccard
often used with imbalanced data

$$U(f) = \frac{a_0\mathrm{TP} + b_0\mathrm{FP} + c_0}{a_1\mathrm{TP} + b_1\mathrm{FP} + c_1}$$

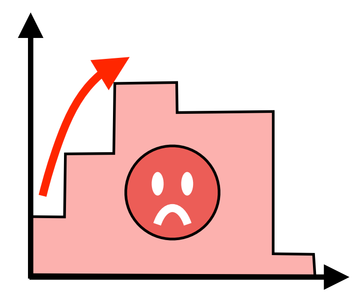
surrogate utility



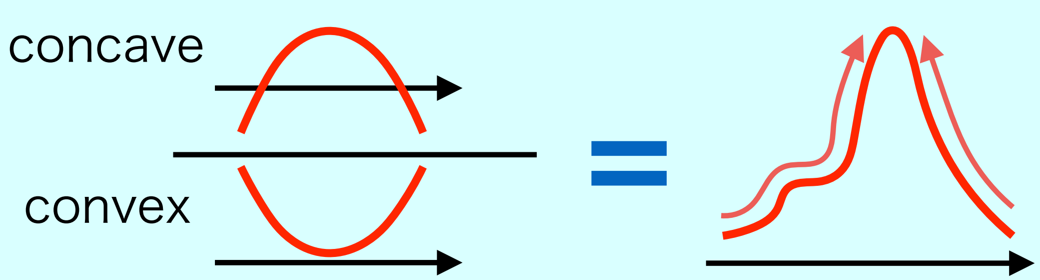
calibrated

target utility

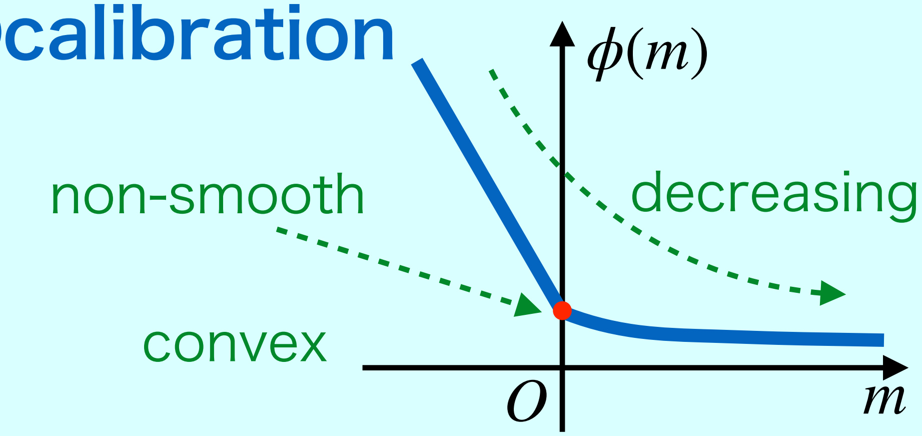
$$U(f) = \frac{a_0\mathrm{TP} + b_0\mathrm{FP} + c_0}{a_1\mathrm{TP} + b_1\mathrm{FP} + c_1}$$



①tractability (quasi-concave)



②calibration



Provides guideline of designing loss for complicated metrics!

When adversary presents

H. Bao, C. Scott, and M. Sugiyama.
Calibrated Surrogate Losses for Adversarially Robust Classification.
In *COLT*, 2020.

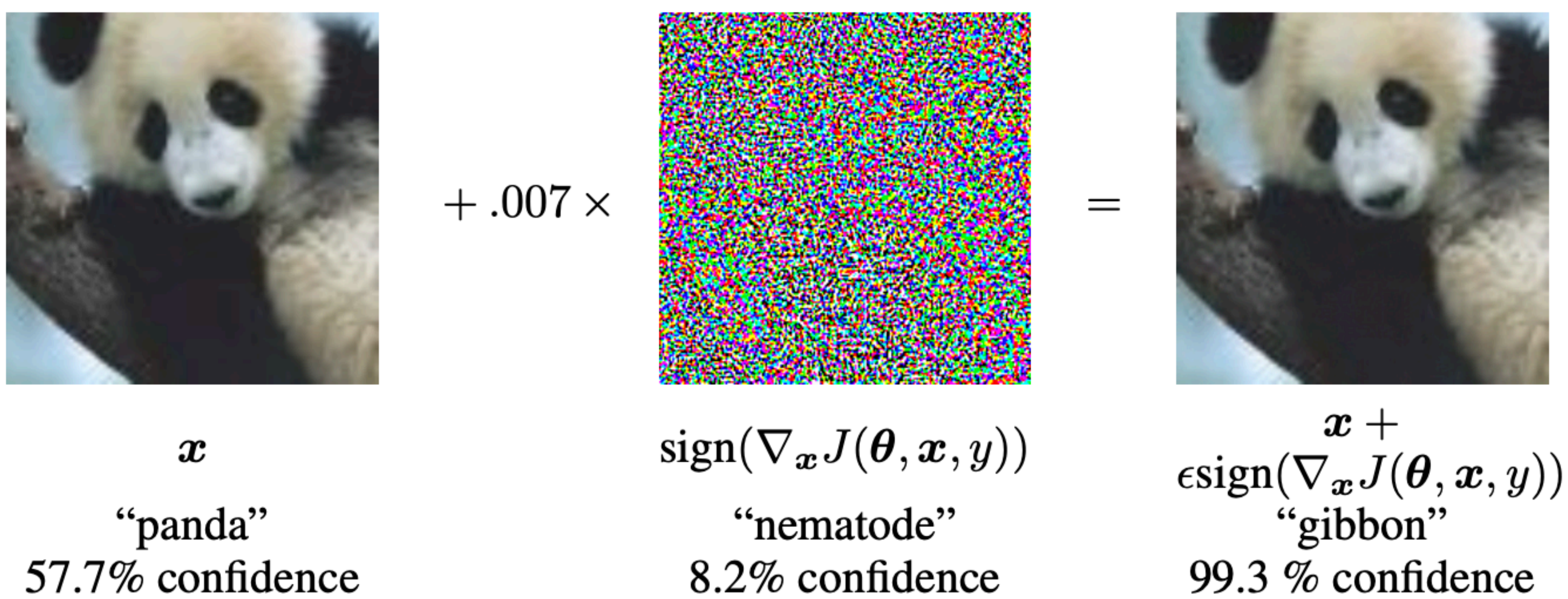
Adversarial Attacks

[Goodfellow+ 2015]

Adding inperceptible small noise can fool classifiers!

original data

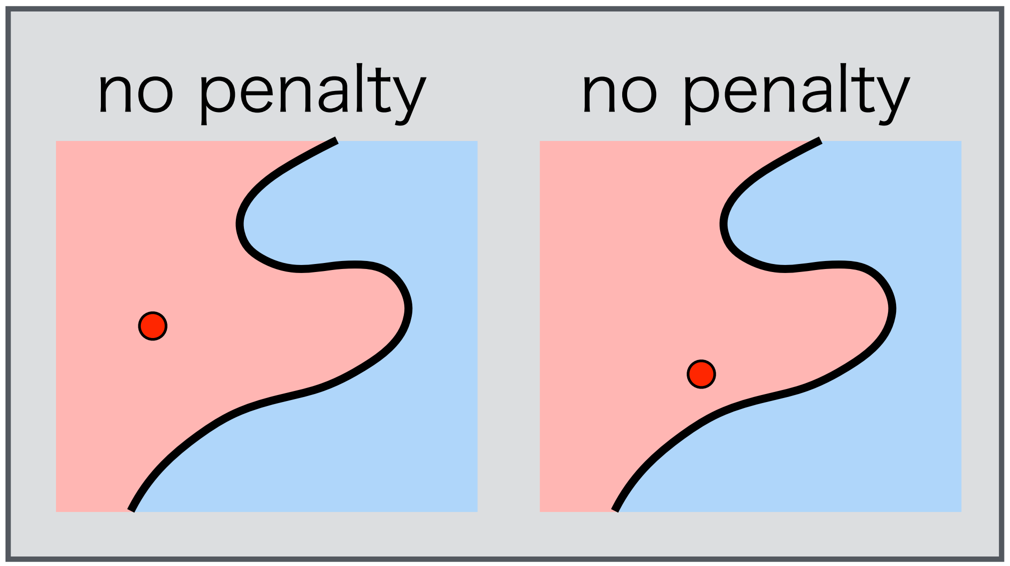
perturbed data



Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Penalize Vulnerable Prediction

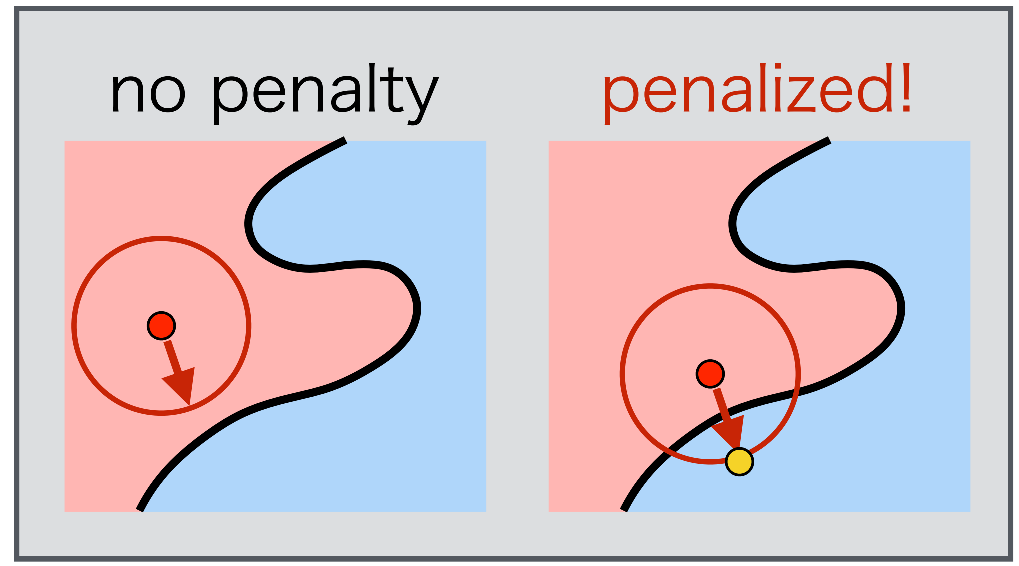
Usual Classification



usual 0-1 loss

$$\ell_{01}(x, y, f) = \begin{cases} 1 \text{ if } yf(x) \leq 0 \\ 0 \text{ otherwise} \end{cases}$$

Robust Classification



robust 0-1 loss

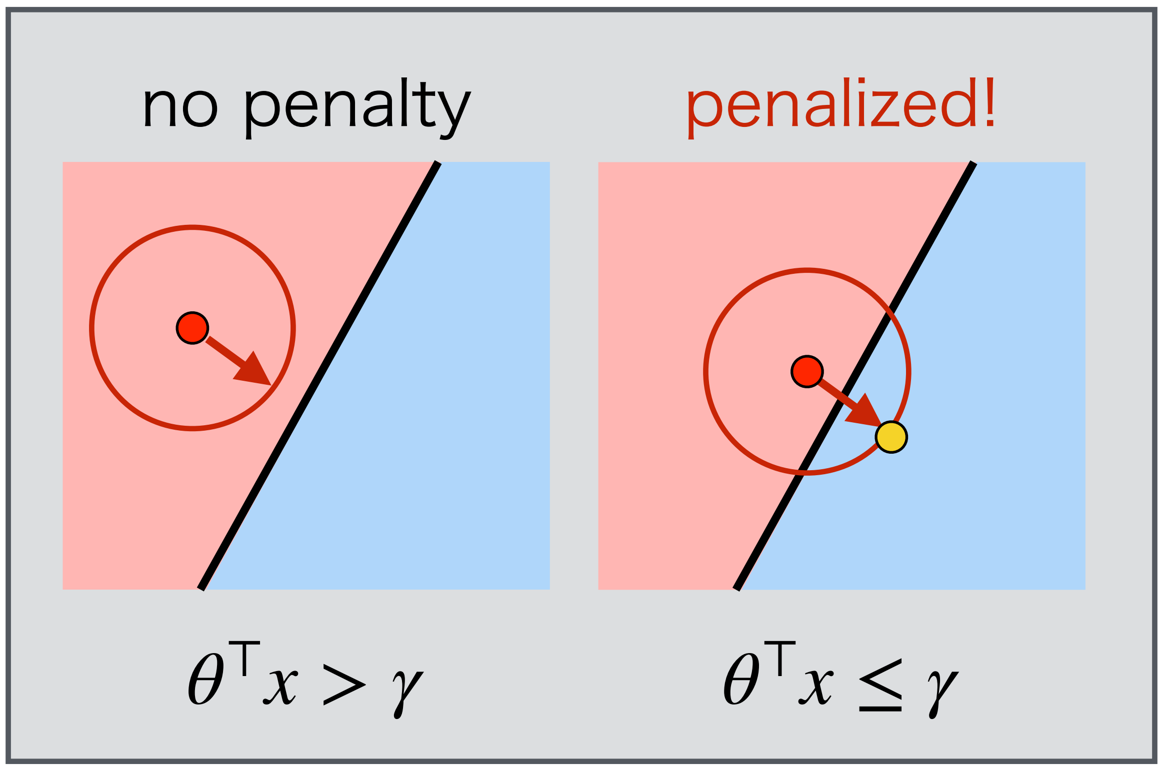
$$\ell_{\gamma}(x, y, f) = \begin{cases} 1 & \text{if } \exists \Delta \in \mathbb{B}_2(\gamma) \,.\, yf(x + \Delta) \leq 0 \\ 0 & \text{otherwise} \end{cases}$$

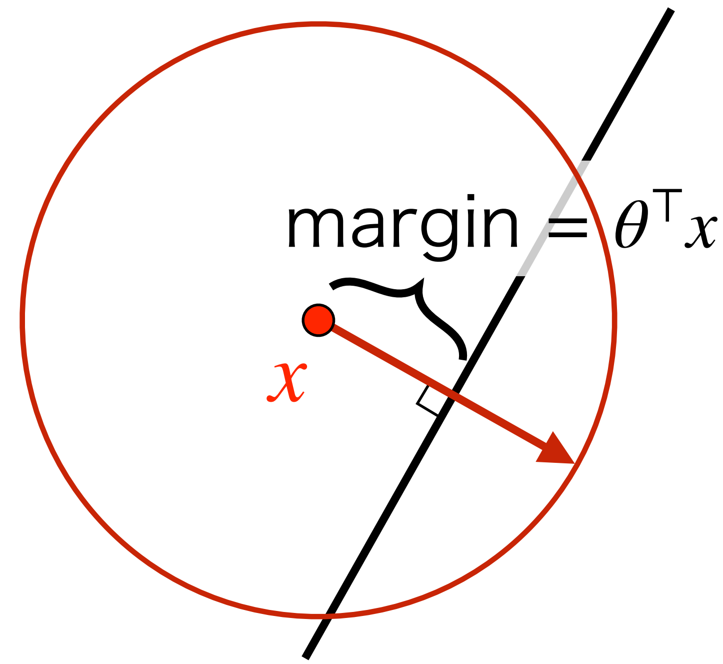
prediction too close to boundary should be penalized

$\mathbb{B}_2(\gamma) = \{x \in \mathbb{R}^d \mid \|x\|_2 \leq \gamma\}$: γ-ball

In Case of Linear Predictors

linear predictors $\mathscr{F}_{\text{lin}} = \{x \mapsto \theta^\top x \mid \|\theta\|_2 = 1\}$





robust 0-1 loss

$$\ell_\gamma(x, y, f) = \begin{cases} 1 & \text{if } \exists \Delta \in \mathbb{B}_2(\gamma) \,.\, yf(x+\Delta) \leq 0 \\ 0 & \text{otherwise} \end{cases} \quad = \quad \mathbf{1}\{yf(x) \leq \gamma\} := \phi_\gamma(yf(x))$$

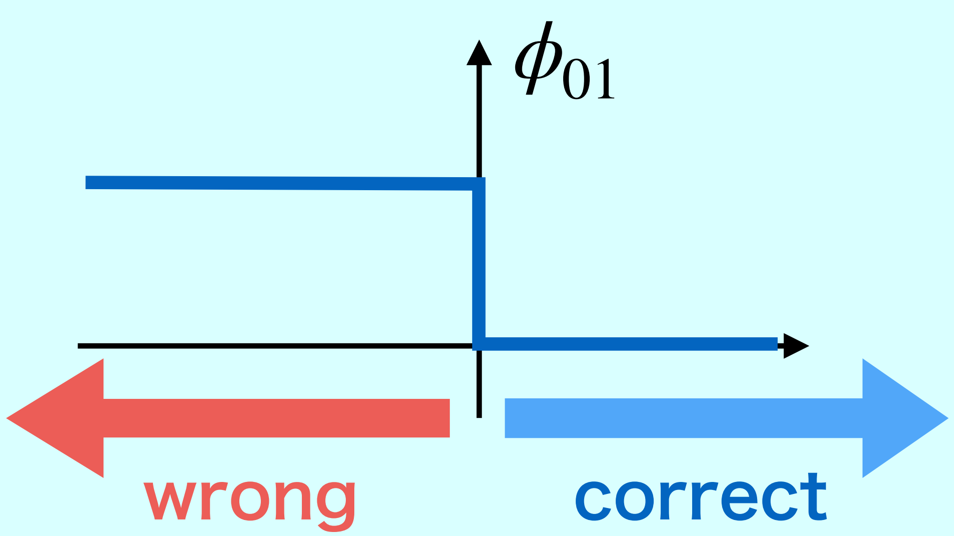
Formulation of Classification

Usual Classification

minimize 0-1 risk

$$R_{\phi_{01}}(f) = \mathbb{E}\left[\phi_{01}(Yf(X))\right]$$

0-1 loss $\phi_{01}(\alpha) = \mathbf{1}\{\alpha \leq 0\}$



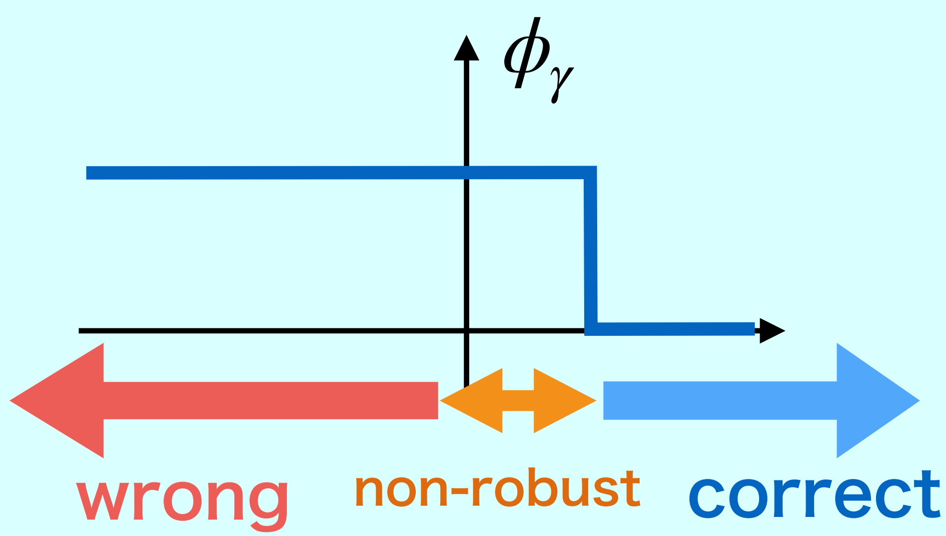
Robust Classification

minimize γ-robust 0-1 risk

$$R_{\phi_\gamma}(f) = \mathbb{E}\left[\phi_\gamma(Yf(X))\right]$$

(restricted to linear predictors)

robust 0-1 loss $\phi_\gamma(\alpha) = \mathbf{1}\{\alpha \leq \gamma\}$

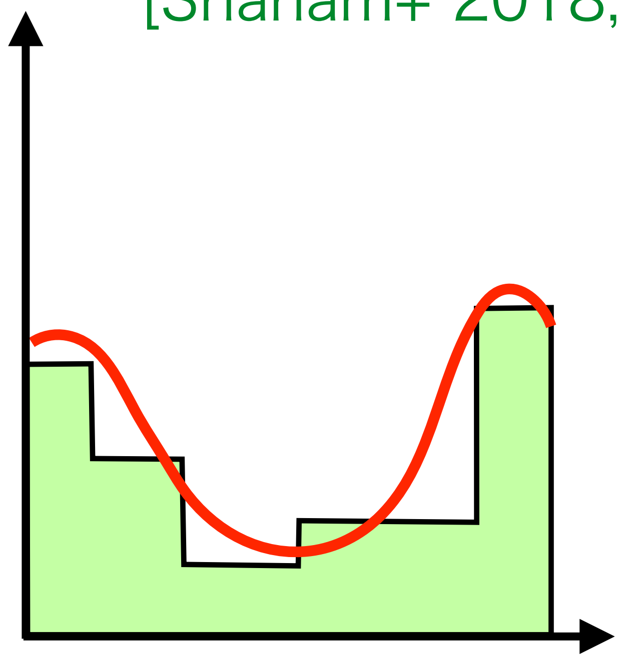


Existing Approaches

Direct optimization of robust risk $R_{\phi_\gamma}(f)$ is intractable

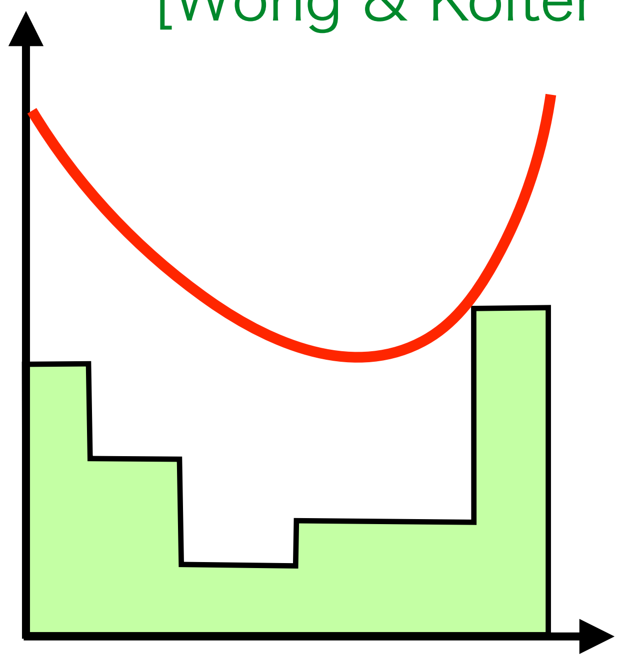
Taylor approximation

[Shaham+ 2018; etc.]



Convex upper bound

[Wong & Kolter 2018; etc.]



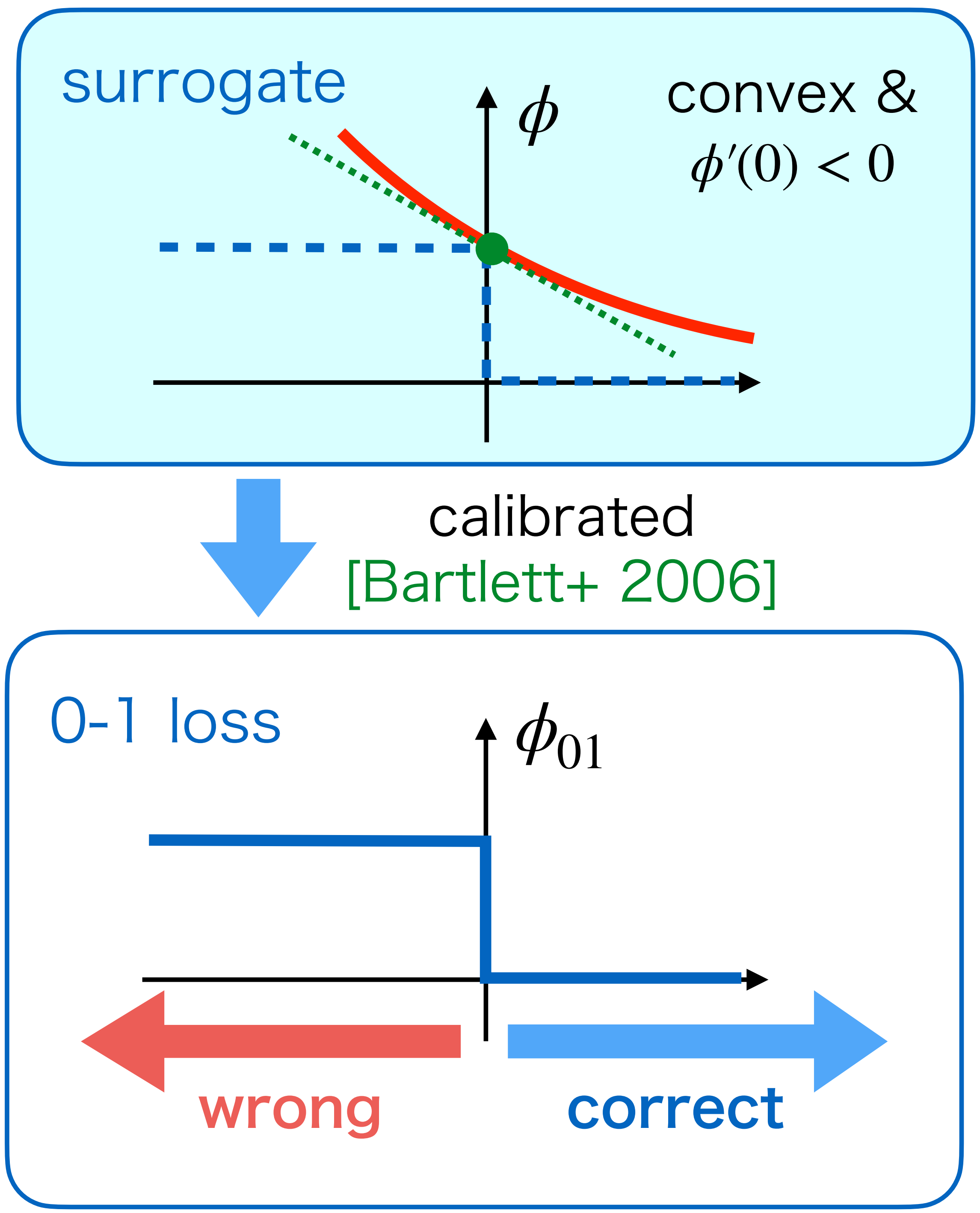
Both do not necessarily lead to true minimizer!

Shaham, U., Yamada, Y., & Negahban, S. (2018).
Understanding adversarial training: Increasing local stability of supervised models through robust optimization. *Neurocomputing*, 195-204.

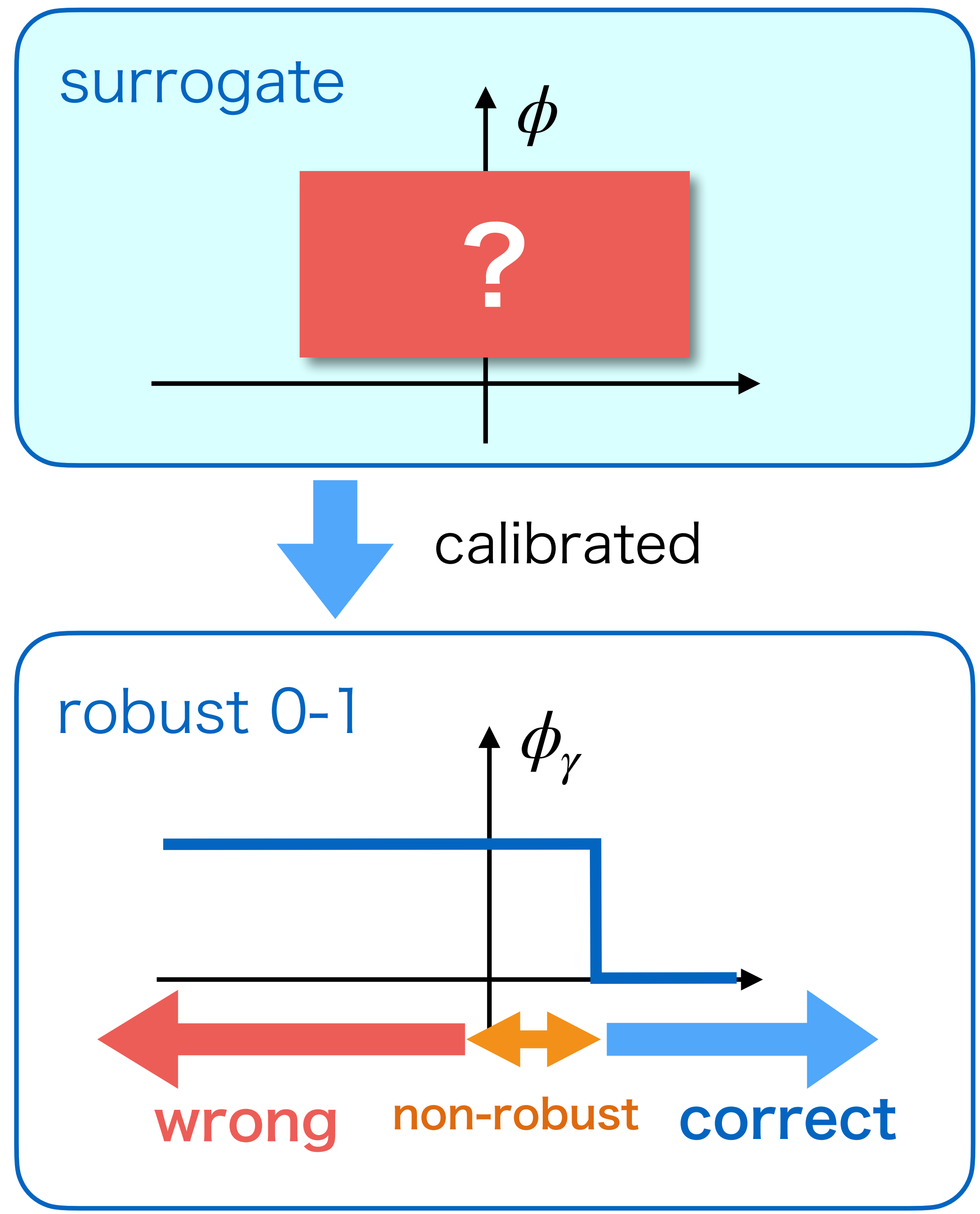
Wong, E., & Kolter, Z. (2018,). Provable Defenses against Adversarial Examples via the Convex Outer Adversarial Polytope.
In *International Conference on Machine Learning* (pp. 5286-5295).

What surrogate is calibrated?

Usual Classification



Robust Classification

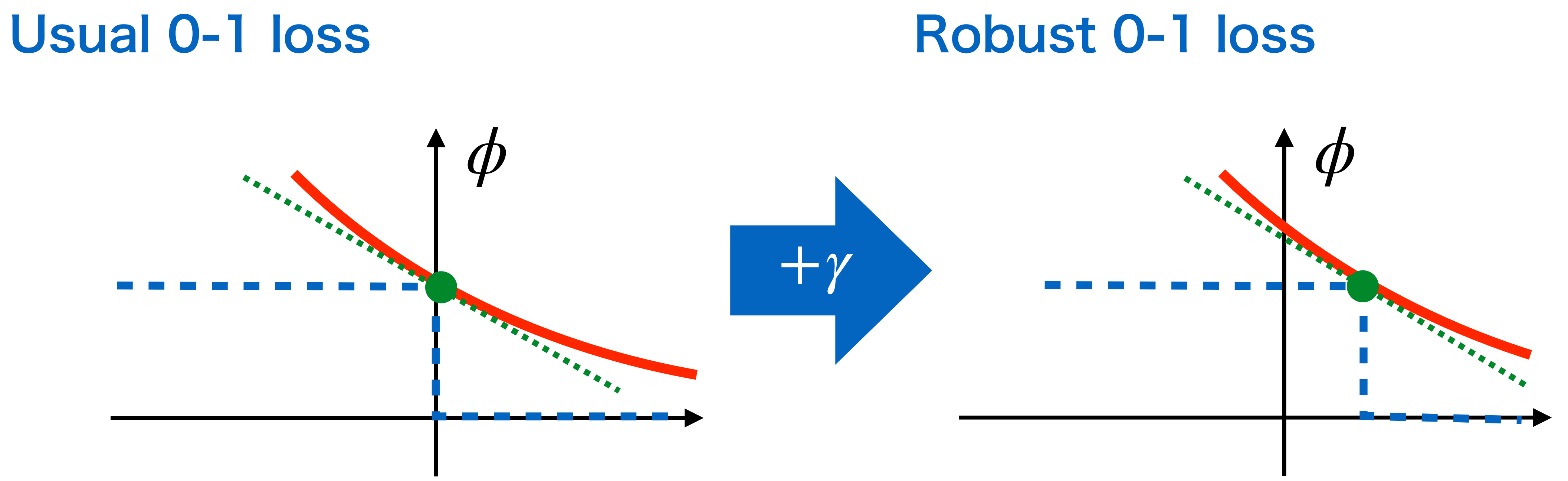


P. L. Bartlett, M. I. Jordan, & J. D. McAuliffe. (2006).
Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473), 138-156.

Isn't it a piece of cake?

Theorem. If surrogate ϕ is convex, it is ϕ_{01}-calibrated iff

- differentiable at 0
- $\phi'(0) < 0$



If $\phi'(\gamma) < 0$, then calibrated to robust 0-1 loss?

No convex calibrated surrogate

Theorem. Any convex surrogate is not ϕ_γ-calibrated.

(under linear predictors)

Proof Sketch

Idea: to show $\delta(\varepsilon) = 0$ for some $\varepsilon > 0$

calibration function

convex in f

f is non-robust ($|f(x)| < \gamma$)

$$\delta(\varepsilon) = \inf_f R_\phi(f) - R_\phi^* \quad \text{s.t.} \quad R_{\phi_\gamma}(f) - R_{\phi_\gamma}^* \geq \varepsilon$$

$$\delta(\varepsilon) = 0 \iff \inf_{f: R_{\phi_\gamma}(f) - R_{\phi_\gamma}^* \geq \varepsilon} R_\phi(f) = \inf_f R_\phi(f)$$

"non-robust" minimizer ($|f(x)| < \gamma$)

optimal minimizer

No convex calibrated surrogate

Theorem. Any convex surrogate is not ϕ_γ-calibrated.

(under linear predictors)

Proof Sketch

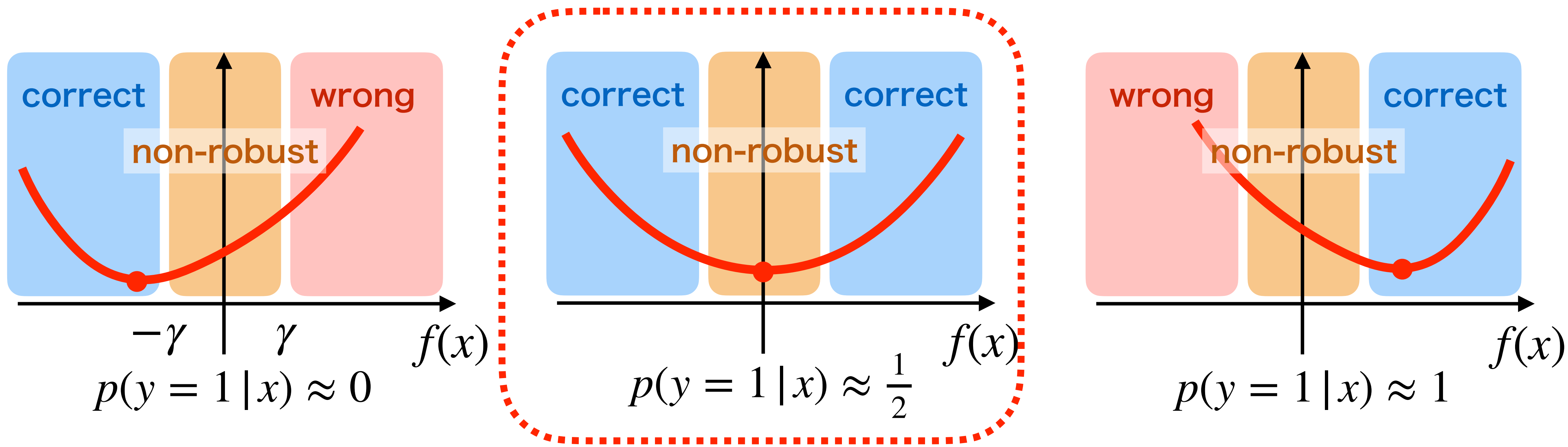
calibration function

convex in f

f is non-robust ($|f(x)| < \gamma$)

$$\delta(\varepsilon) = \inf_f R_\phi(f) - R_\phi^* \quad \text{s.t.} \quad R_{\phi_\gamma}(f) - R_{\phi_\gamma}^* \geq \varepsilon$$

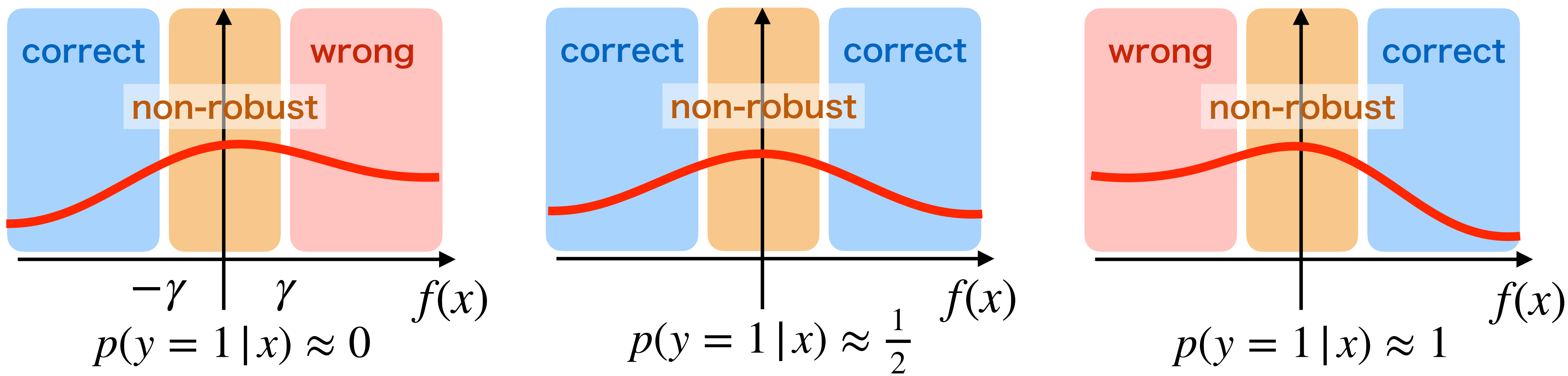
non-robust minimizer!



surrogate **conditional** risk is plotted

How to find calibrated surrogate?

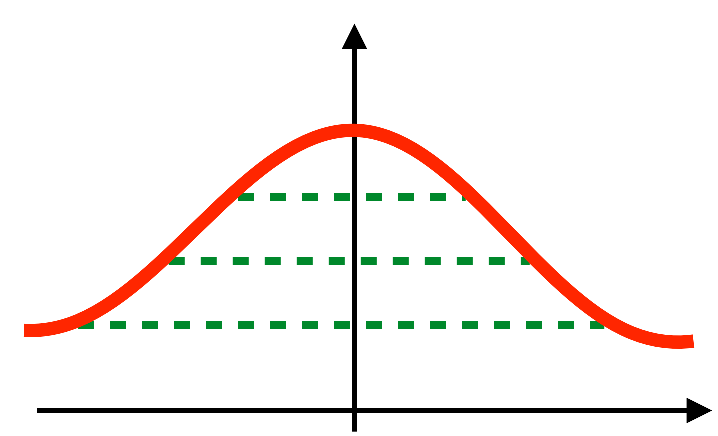
Idea. To make conditional risk not minimized in **non-robust area**



surrogate conditional risk is plotted

consider a surrogate ϕ such that

conditional risk is **quasiconcave**

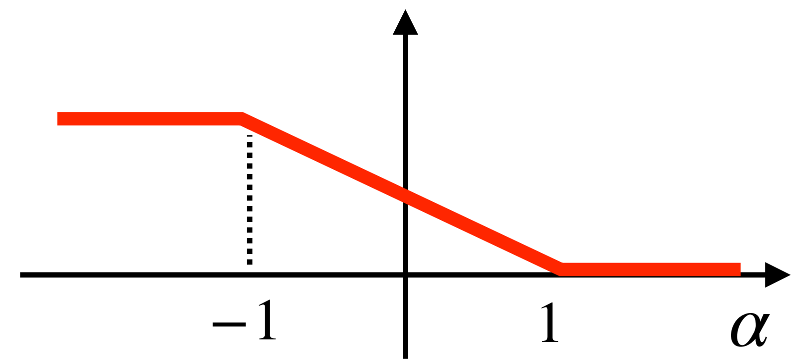


all superlevels are convex

Example: Shifted Ramp Loss

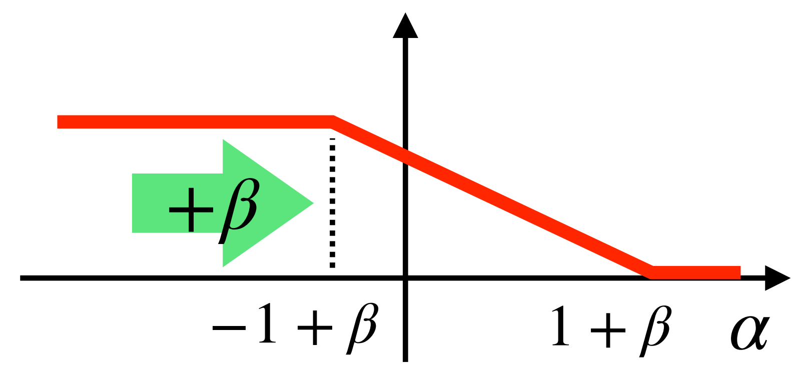
Ramp loss

$$\phi(\alpha) = \mathrm{clip}_{[0,1]}\left(\frac{1-\alpha}{2}\right)$$

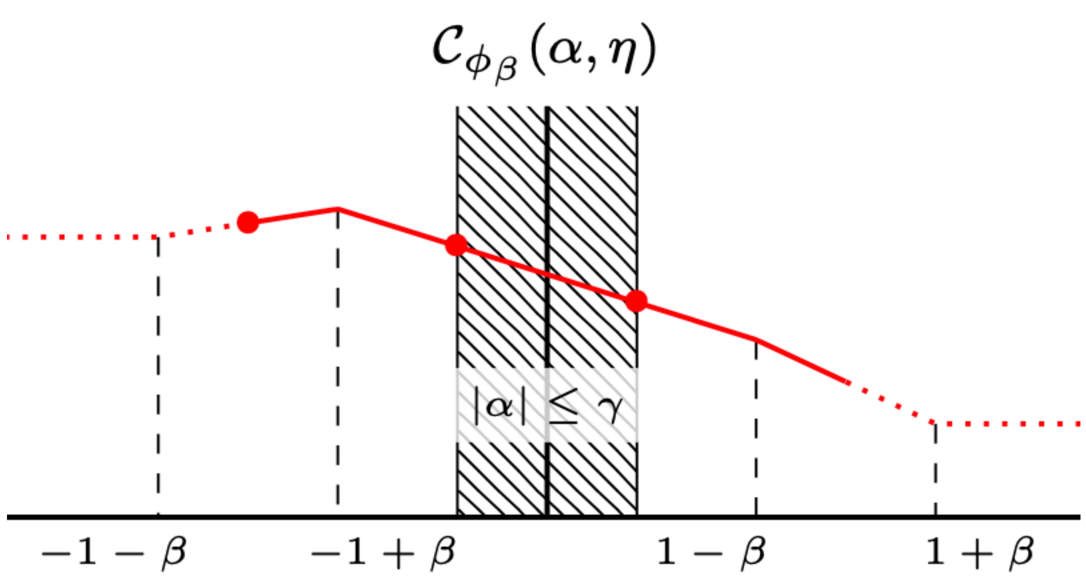


Shifted ramp loss

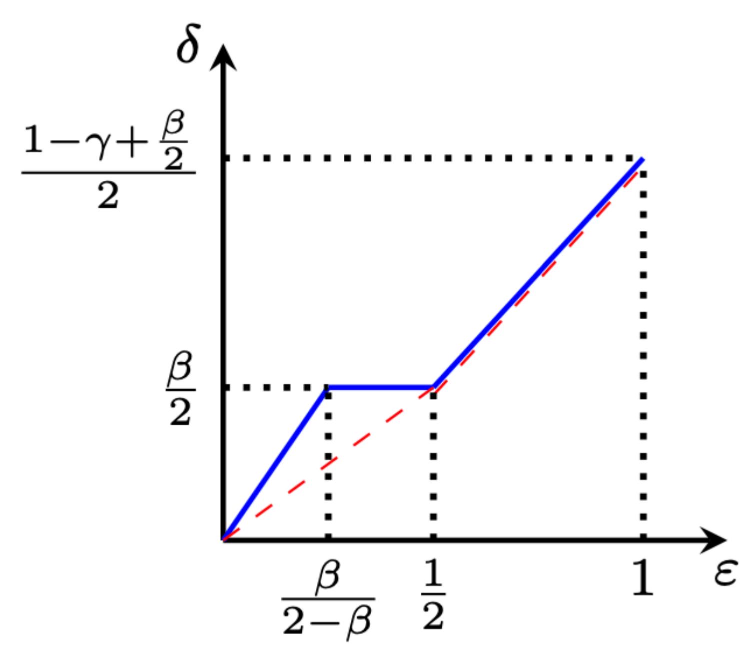
$$\phi_\beta(\alpha) = \mathrm{clip}_{[0,1]}\left(\frac{1-\alpha+\beta}{2}\right)$$



conditional risk $(p(y=1\,|\,x) > \frac{1}{2})$



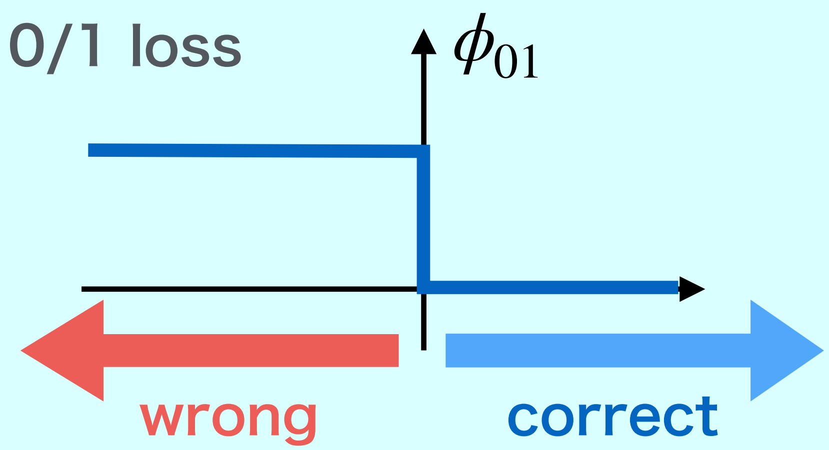
calibration function

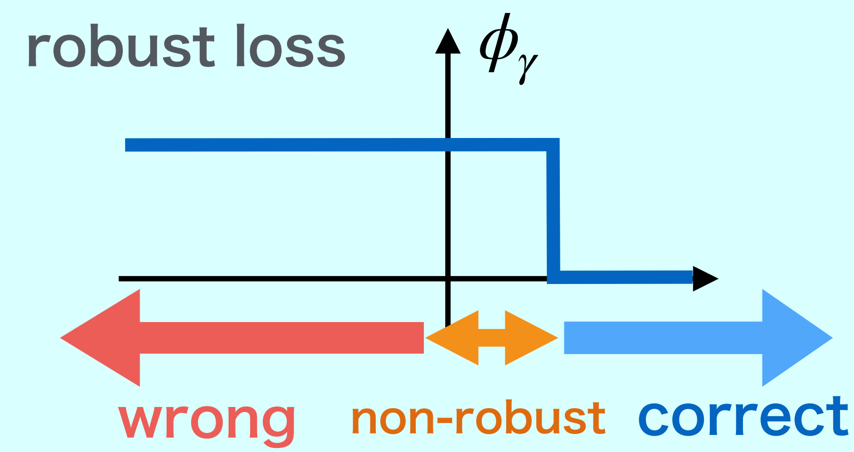


assume $0 < \beta < 1-\gamma$

Loss for Robust Learning

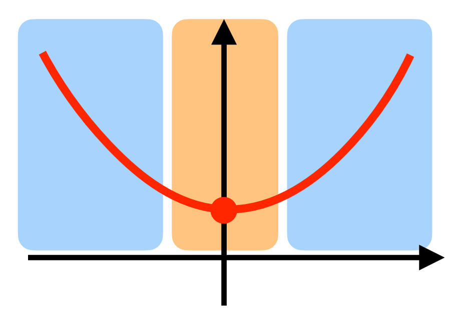
“Embed” robustness into loss function



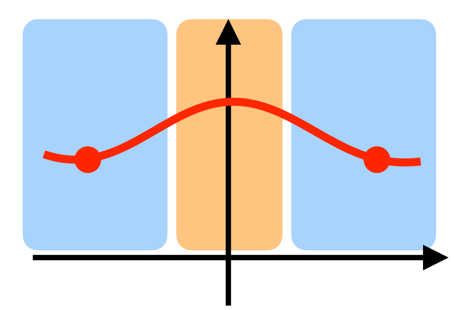


loss function can not only accommodate classification performance but also robustness

Inability of convex loss



convex loss is minimized in non-robust area

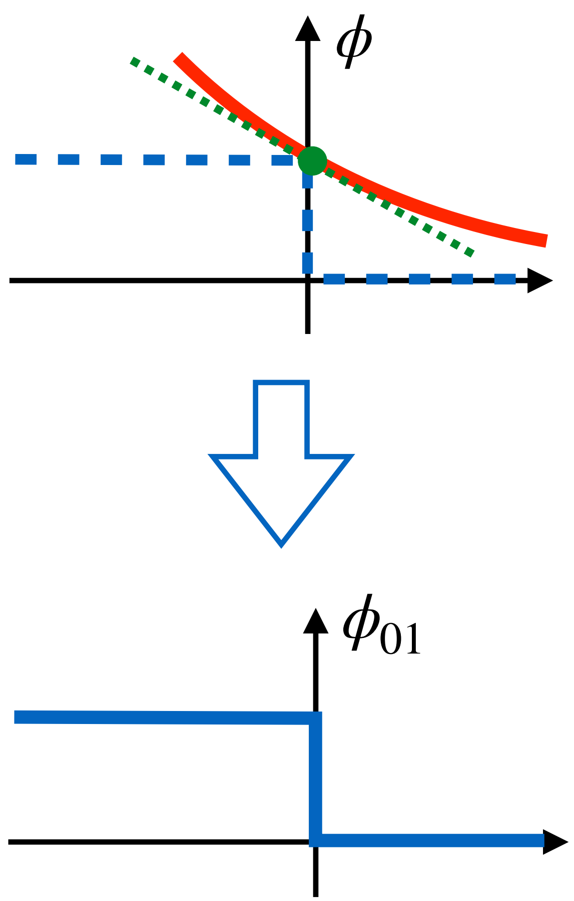


robust objective

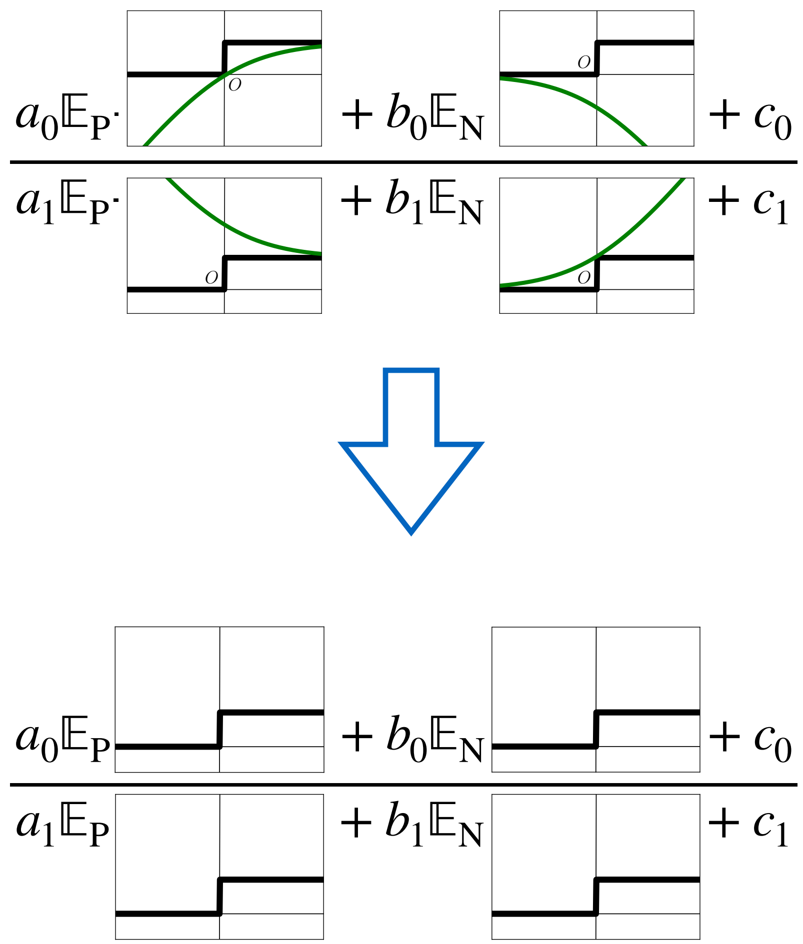
Calibration theory helps to reveal classifiers’ property!

Summary

Binary Classification

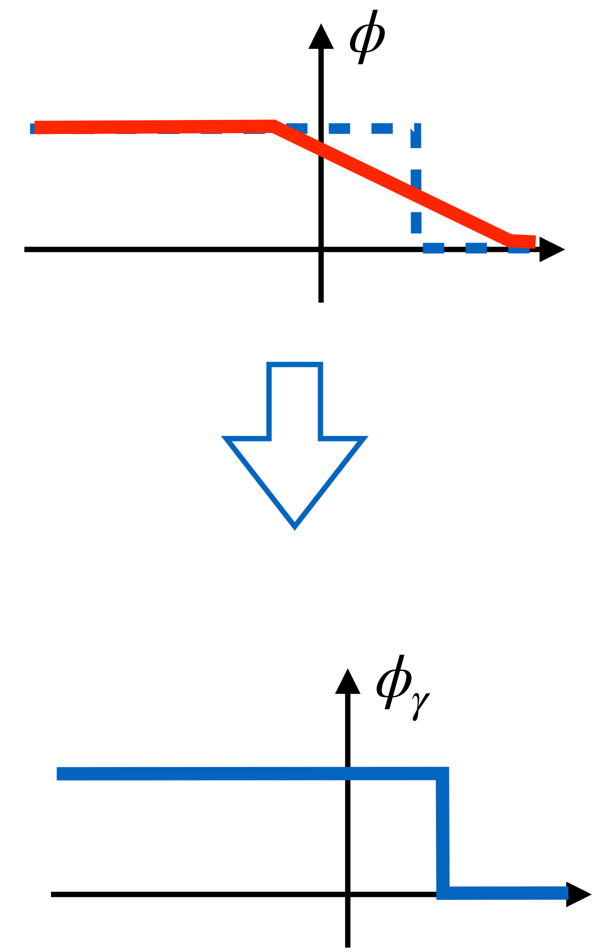


Linear-fractional Metrics

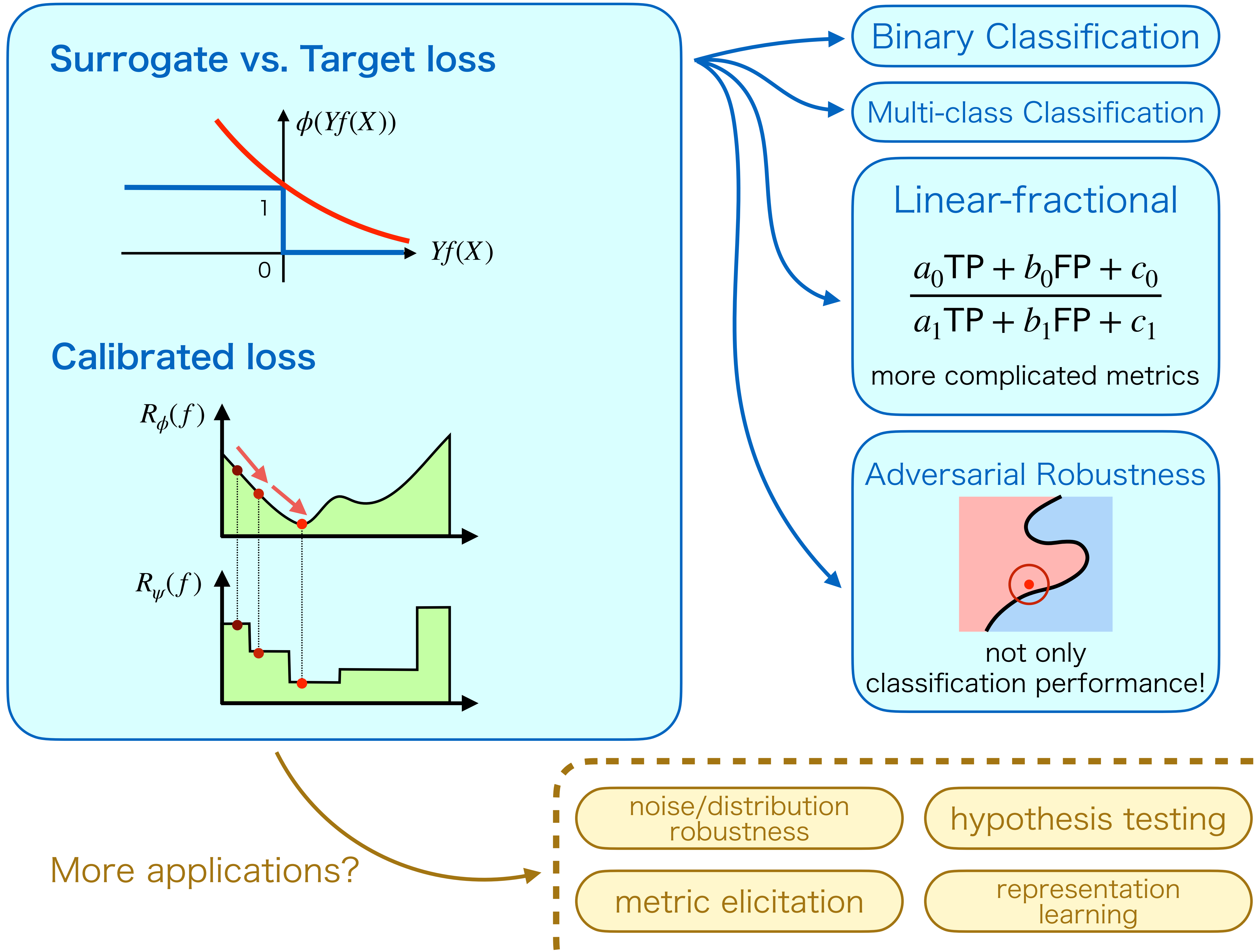


quasi-concave surrogate utility

Adversarial Robustness



no convex & calibrated surrogate


Surrogate vs. Target loss
$\phi(Yf(X))$
1
0
$Yf(X)$
Calibrated loss
$R_\phi(f)$
$R_\psi(f)$
Binary Classification
Multi-class Classification
Linear-fractional
$\frac{a_0\mathrm{TP}+b_0\mathrm{FP}+c_0}{a_1\mathrm{TP}+b_1\mathrm{FP}+c_1}$
more complicated metrics
Adversarial Robustness
not only
classification performance!
More applications?
noise/distribution robustness
hypothesis testing
metric elicitation
representation learning