ÉCOLE POLYTECHNIQUE

INSTITUT POLYTECHNIQUE DE PARIS

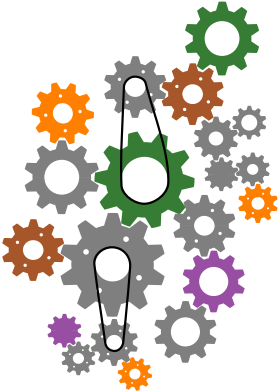
Luiz F. O. Chamon

learning under requirements

Workshop USP-IPP
Mar. 10^{th}, 2024

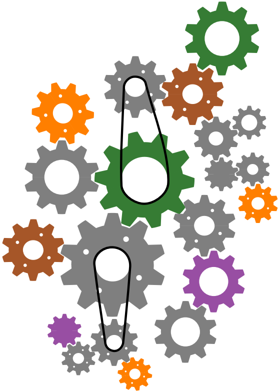
ellis

System engineering cycle

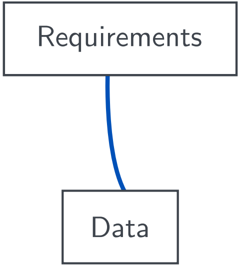


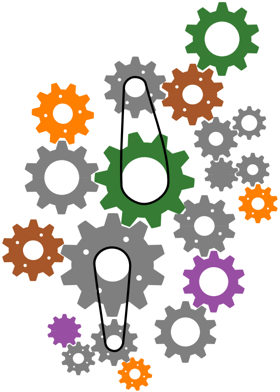
System engineering cycle

Requirements

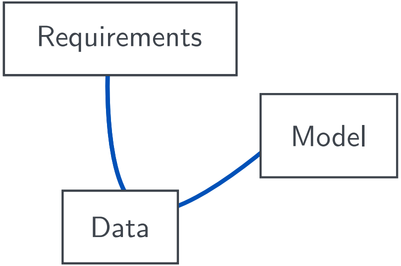


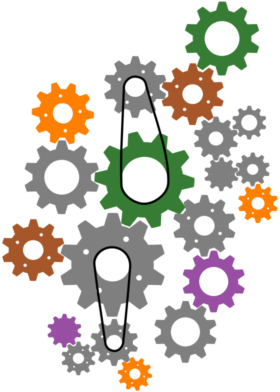
System engineering cycle



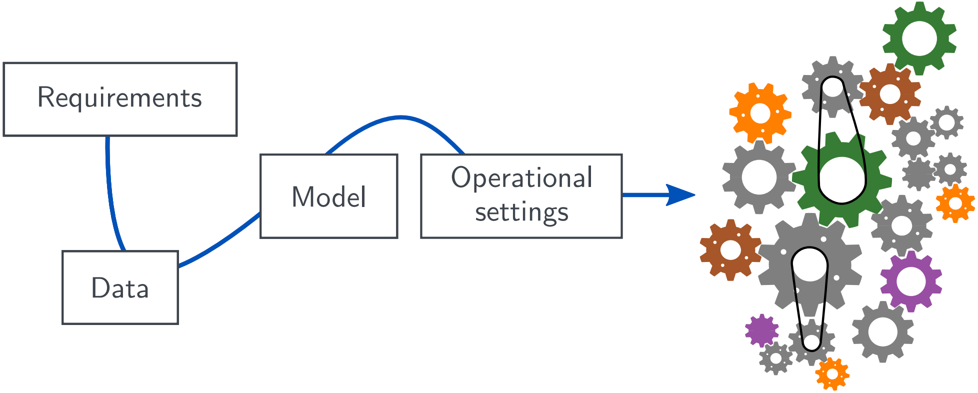


System engineering cycle

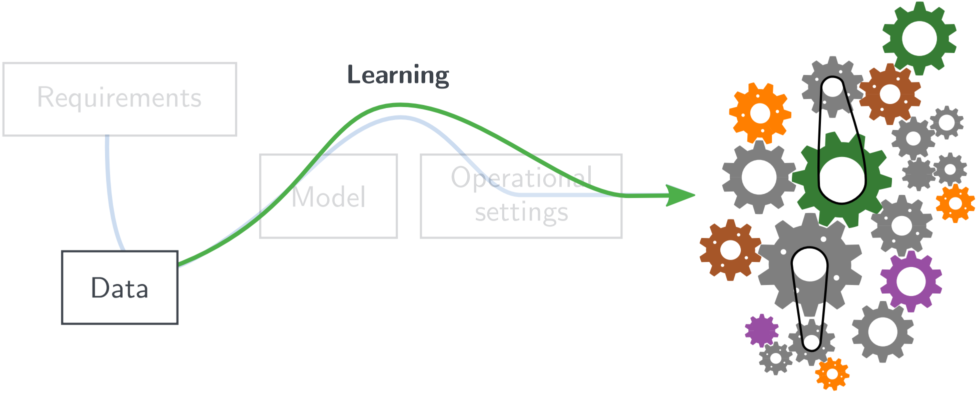




System engineering cycle



The promise of learning



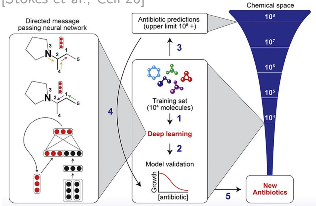
The ~~promise~~ emerging reality of learning

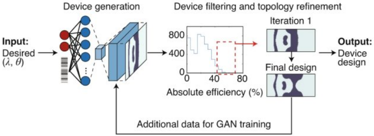
Microsoft Azure will become the preferred cloud platform for Johns Hopkins inHealth precision medicine initiative

June 18, 2020 | Microsoft News Center

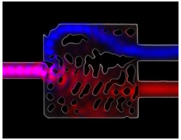
Five-year agreement will support Johns Hopkins Medicine inHealth in driving new medical discoveries to improve disease management and patient care

[Stokes et al., Cell'20]



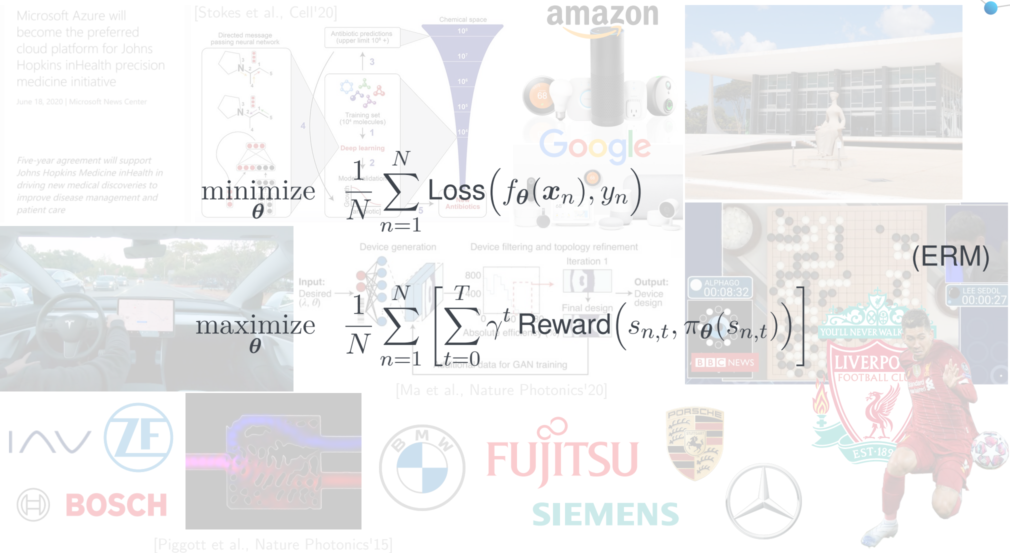


[Ma et al., Nature Photonics'20]



[Piggott et al., Nature Photonics'15]

The ~~promise~~ emerging reality of learning



$$\underset{\boldsymbol{\theta}}{\text{minimize}} \; \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\Big(f_{\boldsymbol{\theta}}(x_n), y_n\Big)$$

(ERM)

$$\underset{\boldsymbol{\theta}}{\text{maximize}} \; \frac{1}{N}\sum_{n=1}^{N} \left[\sum_{t=0}^{T} \gamma^t \, \text{Reward}\Big(s_{n,t}, \pi_{\boldsymbol{\theta}}(s_{n,t})\Big)\right]$$

Learning breakthroughs

Learning breakthroughs

1970s
(theoretical)
Learning theory

Classical learning theory [Vapnik & Chervonenkis, TP'71; Valiant, CACM'84]:

$$\min_{\boldsymbol{\theta}} \ \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big) \xrightarrow{\text{“LLN”}} \min_{\boldsymbol{\theta}} \ \mathbb{E}\Big[\text{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\Big)\Big]$$

- e.g., linear functions, smooth functions (finite RKHS norm, bandlimited), NNs...

Learning breakthroughs

1970s
(theoretical)
Learning theory

2000s
(technological)
Internet

2010s
(computational)
Deep learning

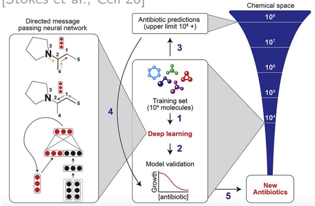
The ~~promise~~ emerging reality of learning

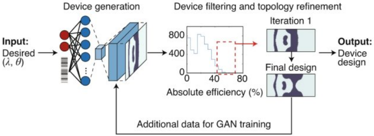
Microsoft Azure will become the preferred cloud platform for Johns Hopkins inHealth precision medicine initiative

June 18, 2020 | Microsoft News Center

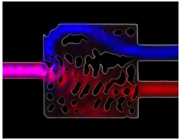
Five-year agreement will support Johns Hopkins Medicine inHealth in driving new medical discoveries to improve disease management and patient care

[Stokes et al., Cell'20]



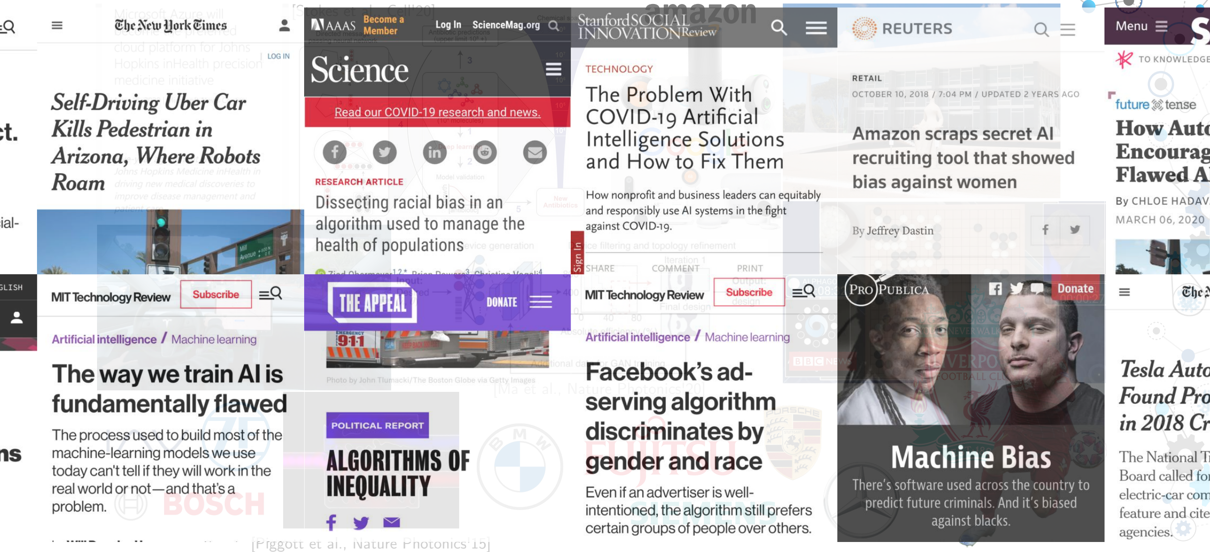


[Ma et al., Nature Photonics'20]



[Piggott et al., Nature Photonics'15]

The ~~promise~~ limitations of learning



Improving ERM

Improving ERM

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \mathsf{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Xie & Yuille, ICLR'20; Guo et al., CVPR'20; Finzi et al., ICML'20; Li et al., ICRL'21; Lu et al., Nature Mach. Intel.'21; Raissi et al., J. Comp. Phys.'19; ...]

Improving ERM

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \mathsf{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Xie & Yuille, ICLR'20; Guo et al., CVPR'20; Finzi et al., ICML'20; Li et al., ICRL'21; Lu et al., Nature Mach. Intel.'21; Raissi et al., J. Comp. Phys.'19; . . .]

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \mathsf{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Kamiran & Calders, KIS'12; Feldman et al., SIGKDD'15; Calmon et al., NeurIPS'17; Chen et al., ICML'20; Müller & Hutter, ICCV'21; Zheng et al., ICRL'22; . . .]

Improving ERM

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \mathsf{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Xie & Yuille, ICLR'20; Guo et al., CVPR'20; Finzi et al., ICML'20; Li et al., ICRL'21; Lu et al., Nature Mach. Intel.'21; Raissi et al., J. Comp. Phys.'19; . . .]

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \mathsf{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Kamiran & Calders, KIS'12; Feldman et al., SIGKDD'15; Calmon et al., NeurIPS'17; Chen et al., ICML'20; Müller & Hutter, ICCV'21; Zheng et al., ICRL'22; . . .]

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \mathsf{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Goodfellow et al., ICLR'15; Arjovsky et al., ICML'17; Madry et al., ICLR'18; Zhang et al., ICML'19; Raissi et al., J. Comp. Phys.'19; Krishnan et al., NeurIPS'20; . . .]

Improving ERM

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Xie & Yuille, ICLR'20; Guo et al., CVPR'20; Finzi et al., ICML'20; Li et al., ICRL'21; Lu et al., Nature Mach. Intel.'21; Raissi et al., J. Comp. Phys.'19; . . .]

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Kamiran & Calders, KIS'12; Feldman et al., SIGKDD'15; Calmon et al., NeurIPS'17; Chen et al., ICML'20; Müller & Hutter, ICCV'21; Zheng et al., ICRL'22; . . .]

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Goodfellow et al., ICLR'15; Arjovsky et al., ICML'17; Madry et al., ICLR'18; Zhang et al., ICML'19; Raissi et al., J. Comp. Phys.'19; Krishnan et al., NeurIPS'20; . . .]

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\Big)$$

[Helmbold & Long, JMLR'15; Mianjy et al., ICML'18; Tashiro et al., NeurIPS'20; Li et al., AISTATS'20; Lin et al., ICML'20; Foret et al., ICRL'21; . . .]

A different paradigm. . .

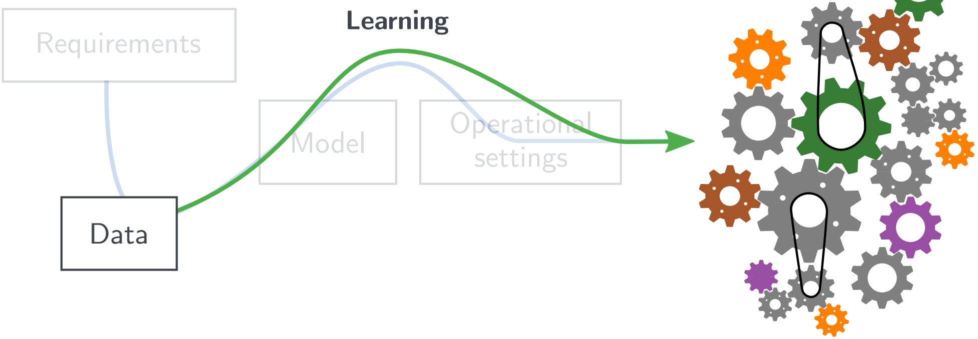
Learning is doing exactly what we asked for.

A different paradigm. . .

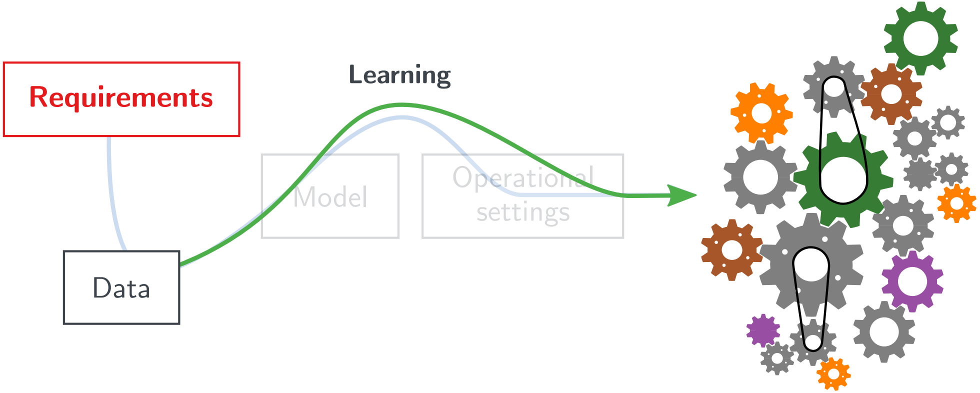
Learning is doing exactly what we asked for.

How can AI learn to do what we want?

The promise of learning



The promise of learning



A different paradigm. . .

Learning is doing exactly what we asked for.

How can AI learn to do what we want?

Constrained learning

Claims

Constrained learning is the right way to learn under requirements

Constrained learning is hard. . .

. . . but possible

Claims

Constrained learning is the right way to learn under requirements

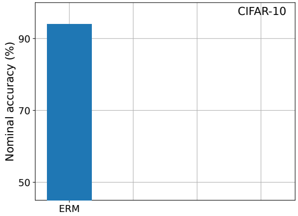
Constrained learning is hard. . .

. . . but possible

Robust image recognition

Problem

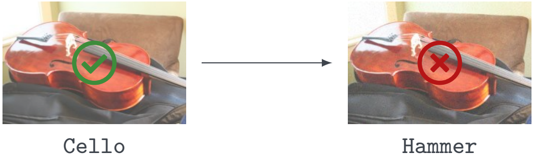
Learn an image classifier

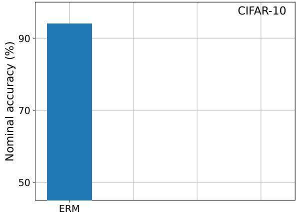


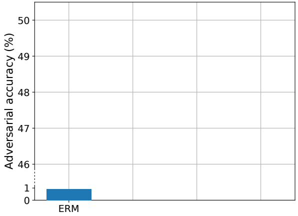
Robust image recognition

Problem

Learn an image classifier







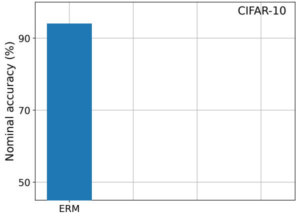
Robust image recognition

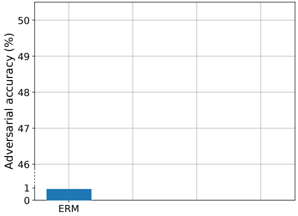
Problem

Learn an image classifier that is robust to input perturbations

Cello

Hammer





Adversarial training

Problem

Learn an image classifier that is robust to input perturbations

- Adversarial training (e.g., [Szegedy et al., ICLR'14; Goodfellow et al., ICLR'15; Madry et al., ICLR'18])

$$\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \longrightarrow \min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \left[\max_{\|\boldsymbol{\delta}\|_\infty \le \epsilon} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n + \boldsymbol{\delta}), y_n\big)\right]$$

Adversarial training

Problem

Learn an image classifier that is robust to input perturbations

- Adversarial training (e.g., [Szegedy et al., ICLR'14; Goodfellow et al., ICLR'15; Madry et al., ICLR'18])

$$\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \longrightarrow \min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \left[\max_{\|\boldsymbol{\delta}\|_\infty \le \epsilon} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n + \boldsymbol{\delta}), y_n\big)\right]$$

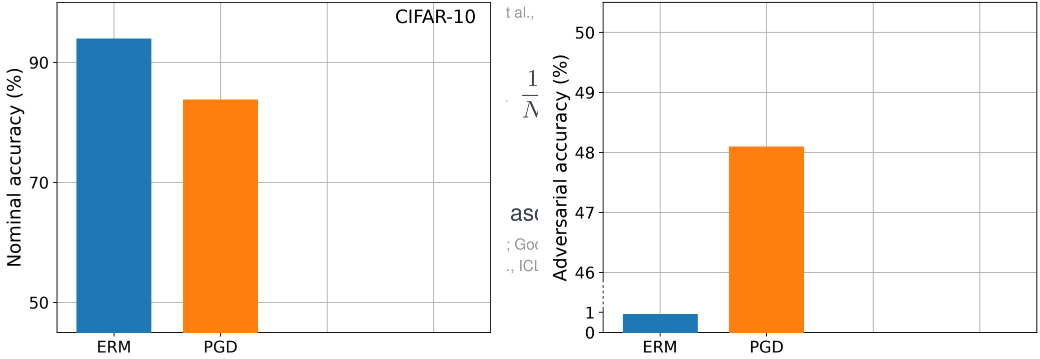
$\approx$ gradient ascent

[Szegedy et al., ICLR'14; Goodfellow et al., ICLR'15; Madry et al., ICLR'18; . . .]

Adversarial training

Problem

Learn an image classifier that is robust to input perturbations



Adversarial training

Problem

Learn an image classifier that is robust to input perturbations

- Adversarial training (e.g., [Zhang et al., ICML'19])

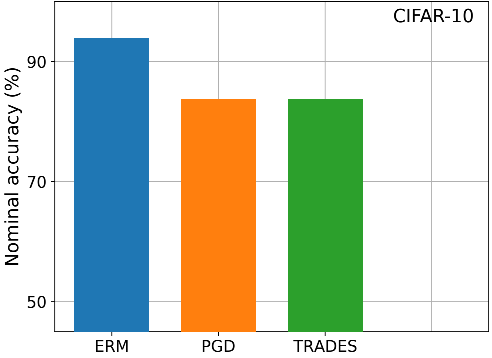
$$\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \mathrm{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \longrightarrow \min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \Big[\max_{\|\boldsymbol{\delta}\|_\infty \le \epsilon} \mathrm{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n + \boldsymbol{\delta}), y_n\big)\Big]$$

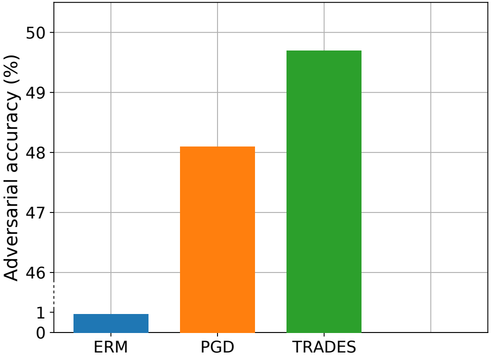
$$\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \mathrm{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) + \lambda\Big[\max_{\|\boldsymbol{\delta}\|_\infty \le \epsilon} \mathrm{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n + \boldsymbol{\delta}), y_n\big)\Big]$$

Adversarial training

Problem

Learn an image classifier that is robust to input perturbations





Penalty-based methods

Problem

Learn an image classifier that is robust to input perturbations

$$\min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) + \lambda \left[\max_{\|\boldsymbol{\delta}\|_\infty \leq \epsilon} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n + \boldsymbol{\delta}), y_n\big) \right]$$

- ✖ No straightforward relation between λ and adversarial loss
- ✖ λ depends on the values of the losses (dataset, model, performance measure)
- ✖ Requirement generalization

Constrained learning for robustness

Problem

Learn an image classifier that is robust to input perturbations

$$\begin{aligned} \min_{\boldsymbol{\theta}} \quad & \text{Nominal loss} \\ \text{subject to} \quad & \text{Adversarial loss} \leq c \end{aligned}$$

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Constrained learning for robustness

Problem

Learn an image classifier that is robust to input perturbations

$$\begin{aligned} \min_{\boldsymbol{\theta}} \quad & \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \\ \text{subject to} \quad & \text{Adversarial loss} \leq c \end{aligned}$$

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Constrained learning for robustness

Problem

Learn an image classifier that is robust to input perturbations

$$\begin{aligned}\min_{\boldsymbol{\theta}} \quad & \frac{1}{N}\sum_{n=1}^{N} \mathsf{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \\ \text{subject to} \quad & \frac{1}{N}\sum_{n=1}^{N} \left[\max_{\|\boldsymbol{\delta}\|_\infty \leq \epsilon} \mathsf{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n + \boldsymbol{\delta}), y_n\big)\right] \leq c\end{aligned}$$

- ✅ More natural: requirement is a constraint, not a cost
- ✅ Decouple performance and requirements

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Constrained learning for robustness

Problem

Learn an image classifier that is robust to input perturbations

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big)$$

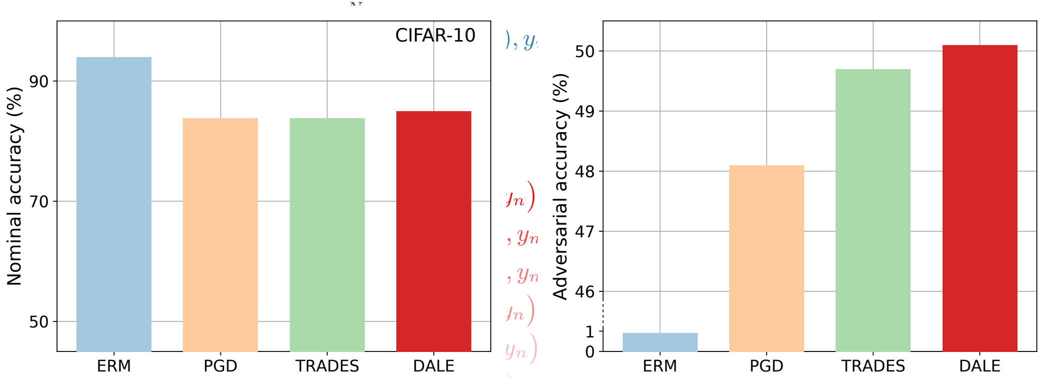
$$\text{subject to} \quad \frac{1}{N}\sum_{n=1}^{N} \left[\max_{\|\boldsymbol{\delta}\|_\infty \leq \epsilon} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n + \boldsymbol{\delta}), y_n\big)\right] \leq c$$

[Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21]

Constrained learning for robustness

Problem

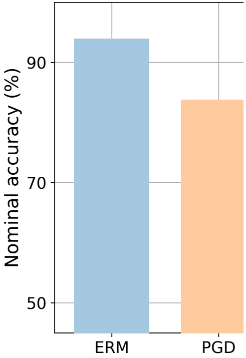
Learn an image classifier that is robust to input perturbations

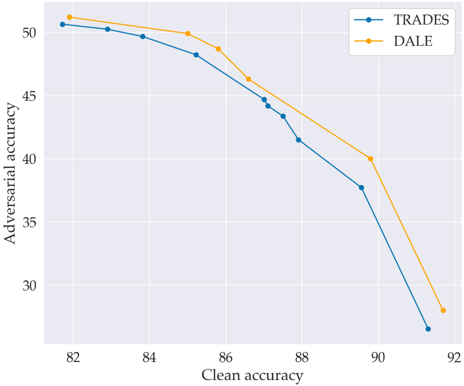


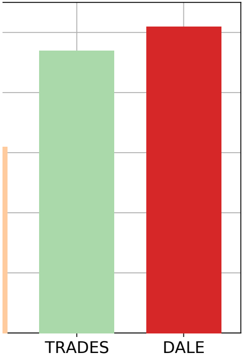
Constrained learning for robustness

Problem

Learn an image classifier that is robust to input perturbations



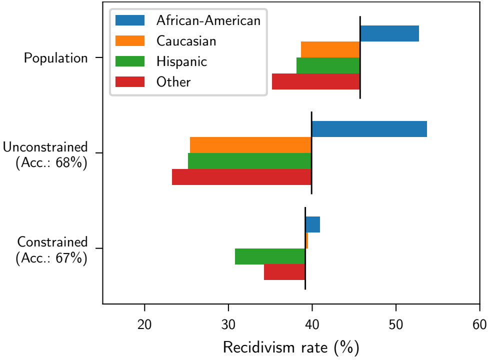




Fair learning

Problem

Predict whether an individual will recidivate at the same rate across races



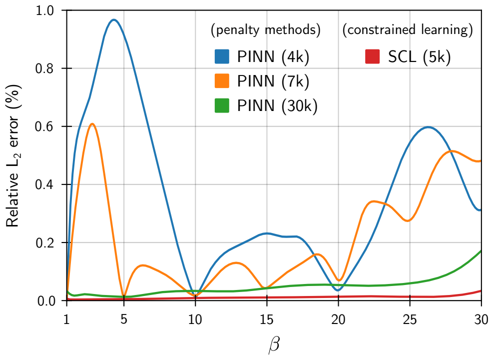
$$\begin{array}{rl} \text{minimize} & \text{Prediction error} \\ \text{subject to} & \text{Prediction rate disparity (Race)} \leq c \end{array}$$

[**C.**, Paternain, Calvo-Fullana, Ribeiro, ICASSP'20 (best student paper); **C.** and Ribeiro, NeurIPS'20; **C.**, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23]

Learning to solve PDEs

Problem

Obtain (weak) solutions for a parametric family of boundary value problems



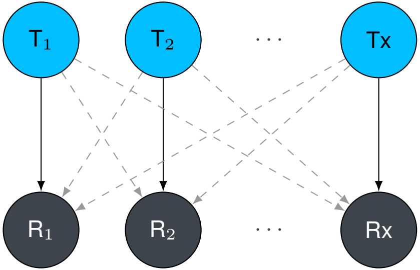
$$\begin{array}{rl} \text{minimize} & \text{Boundary condition error} \\ \text{subject to} & \text{Weak formulation error} \leq \epsilon, \ \ \forall(x, t) \end{array}$$

[Moro and **C.**, ICLR'25]

Wireless resource allocation

Problem

Allocate the least transmit power to m device pairs to achieve a communication rate



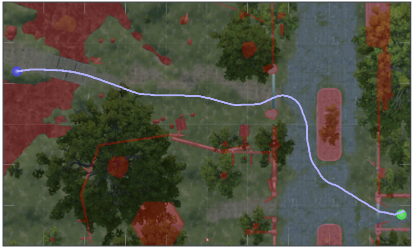
$$\begin{array}{rl} \text{minimize} & \text{Total power} \\ \text{subject to} & \text{Communicate rate } (\mathrm{T}_i) \geq c_i \end{array}$$

[Eisen, Zhang, **C.**, Lee, Ribeiro, IEEE TSP'19]

Safe reinforcement learning

Problem

Learn a control policy that navigates the environment effectively and safely



$$
\begin{aligned}
\operatorname*{maximize} \quad & \text{Task reward} \\
\operatorname{subject\ to} \quad & \mathbb{P}\left(\text{Colliding with obstacles}\right) \leq \delta
\end{aligned}
$$

[Paternain, Calvo-Fullana, **Chamon**, Ribeiro, IEEE TAC'23]

Claims

Constrained learning is the right way to learn under requirements

Constrained learning is hard. . .

. . . but possible

(Un)constrained learning

$$\hat{P}_{\mathsf{U}}^{\star} = \min_{\boldsymbol{\theta}} \quad \frac{1}{N} \sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big)$$

- ℓ, g are bounded, Lipschitz continuous (possibly non-convex) functions
- $f_{\boldsymbol{\theta}}$ is a (possibly nonlinear) parametrization [e.g., logistic classifier, (G)(C)NN]
- $(\boldsymbol{x}_n, y_n) \sim \mathfrak{D}$, $(\boldsymbol{x}_m, y_m) \sim \mathfrak{A}$ (i.i.d.)

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

(Un)constrained learning

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big)$$

$$\text{subject to} \quad \frac{1}{N}\sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c$$

- ℓ, g are bounded, Lipschitz continuous (possibly non-convex) functions
- $f_{\boldsymbol{\theta}}$ is a (possibly nonlinear) parametrization [e.g., logistic classifier, (G)(C)NN]
- $(\boldsymbol{x}_n, y_n) \sim \mathfrak{D}$, $(\boldsymbol{x}_m, y_m) \sim \mathfrak{A}$ (i.i.d.)

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

(Un)constrained learning

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad \frac{1}{N} \sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big)$$

$$\text{subject to} \quad \frac{1}{N} \sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c$$

- ℓ, g are bounded, Lipschitz continuous (possibly non-convex) functions
- $f_{\boldsymbol{\theta}}$ is a (possibly nonlinear) parametrization [e.g., logistic classifier, (G)(C)NN]
- $(\boldsymbol{x}_n, y_n) \sim \mathfrak{D}$, $(\boldsymbol{x}_m, y_m) \sim \mathfrak{A}$ (i.i.d.)

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Constrained learning

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad \frac{1}{N} \sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \qquad \xrightarrow{?} \qquad P^\star = \min_{\boldsymbol{\theta}} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big]$$

$$\text{subject to} \quad \frac{1}{N} \sum_{m=1}^{M} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c \qquad\qquad \text{subject to} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c$$

Challenges

- ✖ *Statistical*: does the solution of the constrained empirical problem generalize?

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Constrained learning

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \quad \xrightarrow{\text{"LLN"}} \quad P^\star = \min_{\boldsymbol{\theta}} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big]$$

$$\text{subject to} \quad \frac{1}{N}\sum_{m=1}^{M} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c \qquad \text{subject to} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c$$

Challenges

- *Statistical*: does the solution of the constrained empirical problem generalize?

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Constrained learning

$$
\begin{aligned}
\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad & \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \\
\text{subject to} \quad & \frac{1}{N}\sum_{m=1}^{M} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c
\end{aligned}
\quad \xrightarrow{\;?\;} \quad
\begin{aligned}
P^\star = \min_{\boldsymbol{\theta}} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \\
\text{subject to} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c
\end{aligned}
$$

Challenges

- *Statistical*: does the solution of the constrained empirical problem generalize?

Constrained learning

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big)$$
$$\text{subject to} \quad \frac{1}{N}\sum_{m=1}^{M} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c$$

$$\xrightarrow{?}$$

$$P^\star = \min_{\boldsymbol{\theta}} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big]$$
$$\text{subject to} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c$$

Challenges

- ✖ *Statistical*: does the solution of the constrained empirical problem generalize?
- ✖ *Computational*: can we solve the constrained empirical problem?

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Constrained learning

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \qquad \xrightarrow{?} \qquad P^\star = \min_{\boldsymbol{\theta}} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big]$$

NON-CONVEX

$$\text{subject to} \quad \frac{1}{N}\sum_{m=1}^{M} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c \qquad\qquad \text{subject to} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c$$

$$\not\Updownarrow$$

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) + \lambda \frac{1}{N}\sum_{m=1}^{M} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big)$$

Challenges

- ✖ *Statistical*: does the solution of the constrained empirical problem generalize?
- ✖ *Computational*: can we solve the constrained empirical problem?

[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Constrained learning

$$
\begin{aligned}
\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad & \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \\
\text{subject to} \quad & \frac{1}{N}\sum_{m=1}^{M} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c
\end{aligned}
\quad \xrightarrow{\;?\;} \quad
\begin{aligned}
P^\star = \min_{\boldsymbol{\theta}} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \\
\text{subject to} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c
\end{aligned}
$$

Challenges

- ❌ *Statistical*: does the solution of the constrained empirical problem generalize?
- ❌ *Computational*: can we solve the constrained empirical problem?

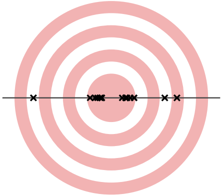
[**C.** and Ribeiro, NeurIPS'20; Robey*, **C.***, Pappas, Hassani, and Ribeiro, NeurIPS'21; **C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

What classical learning theory says?

$$\min_{\boldsymbol{\theta}} \ \frac{1}{N} \sum_{n=1}^{N} \text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \xrightarrow{\text{“LLN”}} \min_{\boldsymbol{\theta}} \ \mathbb{E}\Big[\text{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big]$$

- ✅ $f_{\boldsymbol{\theta}}$ is *probably approximately correct (PAC)* learnable

  e.g., linear functions, smooth functions (finite RKHS norm, bandlimited), NNs. . .
  $(N \approx 1/\epsilon^2)$



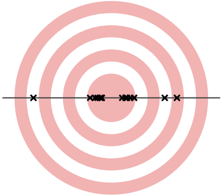
What classical learning theory says?

$$\min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \mathsf{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \xrightarrow{\text{“LLN”}} \min_{\boldsymbol{\theta}} \mathbb{E}\left[\mathsf{Loss}\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\right]$$

- ✅ $f_{\boldsymbol{\theta}}$ is *probably approximately correct (PAC)* learnable

  e.g., linear functions, smooth functions (finite RKHS norm, bandlimited), NNs…
  $(N \approx 1/\epsilon^2)$

- ❌ Constraints?



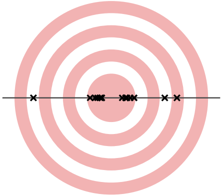
What’s in a solution?

Definition (PAC learnability)

$f_{\boldsymbol{\theta}}$ is a *probably approximately correct (PAC)* learnable if for every ϵ, δ and every distributions $\mathfrak{D}, \mathfrak{A}$, we can obtain $f_{\boldsymbol{\theta}^\dagger}$ from $N_f(\epsilon, \delta)$ samples such that, with prob. $1 - \delta$,

- near-optimal

$$P^\star - \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\left[\ell\big(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\big)\right] \leq \epsilon$$



What's in a solution?

Definition (PACC learnability)

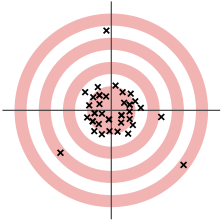
$f_{\boldsymbol{\theta}}$ is a *probably approximately correct constrained (PACC)* learnable if for every ϵ, δ and every distributions $\mathfrak{D}, \mathfrak{A}$, we can obtain $f_{\boldsymbol{\theta}^\dagger}$ from $N_f(\epsilon, \delta)$ samples such that, with prob. $1 - \delta$,

- near-optimal

$$\left| P^\star - \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\left[\ell\left(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\right)\right]\right| \leq \epsilon$$

- approximately feasible

$$\mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\left[g\left(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\right)\right] \leq c + \epsilon$$



[Chamon and Ribeiro, NeurIPS'20; Chamon et al., IEEE TIT'23]

When is constrained learning possible?

$$
\begin{aligned}
\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad & \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \\
\text{subject to} \quad & \frac{1}{N}\sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c
\end{aligned}
\quad \overset{\text{PACC}}{\not\longrightarrow} \quad
\begin{aligned}
P^\star = \min_{\boldsymbol{\theta}} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \\
\text{subject to} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c
\end{aligned}
$$

Proposition

$f_{\boldsymbol{\theta}}$ is PAC learnable $\not\Rightarrow$ $f_{\boldsymbol{\theta}}$ is PACC learnable

[Chamon and Ribeiro, NeurIPS'20; Chamon et al., IEEE TIT'23]

Claims

Constrained learning is the right way to learn under requirements

Constrained learning is hard. . .

. . . but possible

Constrained learning

$$
\begin{aligned}
\hat{P}^\star = \min_{\boldsymbol{\theta}} \quad & \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \\
\text{subject to} \quad & \frac{1}{N}\sum_{m=1}^{M} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c
\end{aligned}
\quad \not\xrightarrow{\cancel{\text{PACC}}} \quad
\begin{aligned}
P^\star = \min_{\boldsymbol{\theta}} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \\
\text{subject to} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c
\end{aligned}
$$

Challenges

- ✖ *Statistical*: does the solution of the constrained empirical problem generalize?
- ✖ *Computational*: can we solve the constrained empirical problem?

Duality

PRIMAL

↕

DUAL

Duality

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \text{ subject to } \frac{1}{N}\sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c$$

PRIMAL

$\updownarrow$

DUAL

Duality

PRIMAL

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \text{ subject to } \frac{1}{N} \sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c$$

$$\updownarrow$$

DUAL

$$\hat{D}^\star = \max_{\lambda \geq 0} \min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda \left[\frac{1}{N} \sum_{m=1}^{N} g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\right) - c\right]$$

Duality

PRIMAL

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \text{ subject to } \frac{1}{N} \sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c$$

$$\updownarrow$$

DUAL

$$\hat{D}^\star = \max_{\lambda \geq 0} \min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda \left[\frac{1}{N} \sum_{m=1}^{N} g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\right) - c\right]$$

- In general, $\hat{D}^\star \leq \hat{P}^\star$
- But in some cases, $\hat{D}^\star = \hat{P}^\star$ (strong duality) [e.g., convex optimization]

Duality

PRIMAL

$$\hat{P}^\star = \min_{\boldsymbol{\theta}} \ \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \ \text{ subject to } \ \frac{1}{N}\sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c$$

$$\not\Updownarrow$$

DUAL

$$\hat{D}^\star = \max_{\lambda \geq 0} \ \min_{\boldsymbol{\theta}} \ \frac{1}{N}\sum_{n=1}^{N} \ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda \left[\frac{1}{N}\sum_{m=1}^{N} g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\right) - c\right]$$

- In general, $\hat{D}^\star \leq \hat{P}^\star$
- ~~But in some cases, $\hat{D}^\star = \hat{P}^\star$ (strong duality) [e.g., convex optimization]~~

Non-convex variational duality

Convex optimization: Primal $\longleftrightarrow$ Dual

Non-convex, finite dimensional optimization: Primal $\not\longleftrightarrow$ Dual

Non-convex variational duality

Convex optimization:	Primal	$\longleftrightarrow$	Dual
Non-convex, finite dimensional optimization:	Primal	$\not\longleftrightarrow$	Dual
Non-convex, **infinite** dimensional optimization:	Primal	$\longleftrightarrow$	Dual

[Chamon et al., IEEE TSP'20]

Sparse logistic regression

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} - \sum_{n=1}^{N} \log \left[1 + \exp \left(y_n \cdot \boldsymbol{\theta}^T \boldsymbol{x}_n \right) \right]$$

$$\text{s. to } \|\boldsymbol{\theta}\|_0 = \sum_{t=1}^{p} \mathbb{I}\left[\boldsymbol{\theta}_t \neq 0\right] \leq k$$

Discrete, non-convex

[Chen et al., JMLR'19]: NP-hard

Sparse logistic regression

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} -\sum_{n=1}^{N} \log \left[1 + \exp\left(y_n \cdot \boldsymbol{\theta}^T \boldsymbol{x}_n\right)\right]$$

$$\text{s. to } \|\boldsymbol{\theta}\|_0 = \sum_{t=1}^{p} \mathbb{I}\left[\boldsymbol{\theta}_t \neq 0\right] \leq k$$

Discrete, non-convex

[Chen et al., JMLR’19]: NP-hard

$$\min_{\theta \in L_2} -\sum_{n=1}^{N} \log \left[1 + \exp\left(y_n \cdot \int \theta(t) x_n(t) dt\right)\right]$$

$$\text{s. to } \|\theta\|_{L_0} = \int \mathbb{I}\left[\theta(t) \neq 0\right] dt \leq \frac{k}{p}$$

Continuous, non-convex

[Chamon et al., IEEE TSP’20]: tractable

Sparse logistic regression

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} -\sum_{n=1}^{N} \log\left[1 + \exp\left(y_n \cdot \boldsymbol{\theta}^T \boldsymbol{x}_n\right)\right]$$

$$\text{s. to } \|\boldsymbol{\theta}\|_0 = \sum_{t=1}^{p} \mathbb{I}\left[\boldsymbol{\theta}_t \neq 0\right] \leq k$$

Discrete, non-convex

[Chen et al., JMLR'19]: NP-hard

$$\min_{\theta \in L_2} -\sum_{n=1}^{N} \log\left[1 + \exp\left(y_n \cdot \int \theta(t) x_n(t) dt\right)\right]$$

$$\text{s. to } \|\theta\|_{L_0} = \int \mathbb{I}\left[\theta(t) \neq 0\right] dt \leq \frac{k}{p}$$

Continuous, non-convex

[Chamon et al., IEEE TSP'20]: tractable

How to learn under constraints?

$$
\begin{aligned}
\min_{\boldsymbol{\theta}} \quad & \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \\
\text{subject to} \quad & \frac{1}{N}\sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c
\end{aligned}
$$

PACC

$$
\begin{aligned}
\min_{\boldsymbol{\theta}} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \\
\text{subject to} \quad & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c
\end{aligned}
$$

$$
\max_{\lambda \geq 0} \; \min_{\boldsymbol{\theta}} \; \frac{1}{N}\sum_{n=1}^{N} \ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda \left[\frac{1}{N}\sum_{m=1}^{N} g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\right) - c\right]
$$

How to learn under constraints?

$$\min_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N} \ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big)$$

$$\text{subject to} \quad \frac{1}{N}\sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\big) \leq c$$

PACC

$$\min_{\boldsymbol{\theta}} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big]$$

$$\text{subject to} \quad \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{A}}\Big[g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\big)\Big] \leq c$$

$$\max_{\lambda \geq 0} \; \min_{\boldsymbol{\theta}} \; \frac{1}{N}\sum_{n=1}^{N} \ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda\left[\frac{1}{N}\sum_{m=1}^{N} g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\right) - c\right]$$

[**C.**, Paternain, Calvo-Fullana, and Ribeiro, ICASSP'20 *(best student paper)*; **C.** and Ribeiro, NeurIPS'20; **C**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Dual (near-)PACC learning

Theorem

Let f be ν-universal, i.e., for each $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$, and $\gamma \in [0,1]$ there exists $\boldsymbol{\theta}$ such that

$$\mathbb{E}\left[\left|\gamma f_{\boldsymbol{\theta}_1}(\boldsymbol{x}) + (1-\gamma) f_{\boldsymbol{\theta}_2}(\boldsymbol{x}) - f_{\boldsymbol{\theta}}(\boldsymbol{x})\right|\right] \leq \nu$$

$\big[\{f_{\boldsymbol{\theta}}\}$ is a good covering of $\overline{\operatorname{conv}}(\{f_{\boldsymbol{\theta}}\})\big]$

Dual (near-)PACC learning

Theorem

Let f be ν-universal, i.e., for each $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$, and $\gamma \in [0, 1]$ there exists $\boldsymbol{\theta}$ such that

$$\mathbb{E}\Big[\big|\gamma f_{\boldsymbol{\theta}_1}(\boldsymbol{x}) + (1-\gamma) f_{\boldsymbol{\theta}_2}(\boldsymbol{x}) - f_{\boldsymbol{\theta}}(\boldsymbol{x})\big|\Big] \leq \nu$$

Then $\hat{D}^\star$ is a (near-)PACC learner, i.e., for all $(\boldsymbol{\theta}^\dagger, \boldsymbol{\lambda}^\dagger)$ that achieve $\hat{D}^\star$, with probability $1-\delta$,

Near-optimal: $$\left|P^\star - \hat{D}^\star\right| \leq \widetilde{O}\left(\nu + \frac{1}{\sqrt{N}}\right)$$

Approximately feasible: $$\mathbb{E}\Big[g\big(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\big)\Big] \leq c + \widetilde{O}\left(\nu + \frac{1}{\sqrt{N}}\right)$$

(ℓ strongly convex and g convex)

(mild additional conditions apply)

[Chamon and Ribeiro, NeurIPS'20; Chamon, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23; Elenter, Chamon, Ribeiro, ICLR'24]

Dual (near-)PACC learning

Theorem

Let f be ν-universal with VC dimension $d_{\text{VC}} < \infty$, ℓ strongly convex, and g convex. Then, $f_{\boldsymbol{\theta}^\dagger}$ is a (near-)PACC solution of (P-CSL) for all $(\boldsymbol{\theta}^\dagger, \boldsymbol{\lambda}^\dagger)$ that achieve $\hat{D}^\star$, i.e., with probability at least $1-\delta$,

$$\left| P^\star - \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\left[\ell\big(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\big)\right]\right| \leq (1+\Delta)\big(\epsilon_0 + \epsilon\big)$$

$$\mathbb{E}\left[g\big(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\big)\right] \leq c + (1+\Delta)^{3/2}\big(M\sqrt{\epsilon_0} + \epsilon\big)$$

$$\epsilon_0 = M\nu \qquad \epsilon = B\sqrt{\frac{1}{N}\left[1+\log\left(\frac{4m(2N)^{d_{\text{VC}}}}{\delta}\right)\right]} \qquad \Delta = \max\left(\left\|\boldsymbol{\lambda}^\star\right\|_1, \left\|\hat{\boldsymbol{\lambda}}^\star\right\|_1, \left\|\tilde{\boldsymbol{\lambda}}^\star\right\|_1\right)$$

Sources of error

parametrization richness (ν) sample size (N) requirements difficulty ($\boldsymbol{\lambda}^\star$)

[Chamon and Ribeiro, NeurIPS'20; Chamon, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23; Elenter, Chamon, Ribeiro, ICLR'24]

Dual (near-)PACC learning

Theorem

Let f be ν-universal with VC dimension $d_{\text{VC}} < \infty$, ℓ strongly convex, and g convex. Then, $f_{\boldsymbol{\theta}^\dagger}$ is a (near-)PACC solution of (P-CSL) for all $(\boldsymbol{\theta}^\dagger, \boldsymbol{\lambda}^\dagger)$ that achieve $\hat{D}^\star$, i.e., with probability at least $1 - \delta$,

$$\left| P^\star - \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\left[\ell\big(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\big)\right] \right| \leq (1+\Delta)\big(\epsilon_0 + \epsilon\big)$$

$$\mathbb{E}\left[g\big(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\big)\right] \leq c + (1+\Delta)^{3/2}\big(M\sqrt{\epsilon_0} + \epsilon\big)$$

$$\epsilon_0 = M\nu \qquad \epsilon = B\sqrt{\frac{1}{N}\left[1 + \log\left(\frac{4m(2N)^{d_{\text{VC}}}}{\delta}\right)\right]} \qquad \Delta = \max\left(\left\|\boldsymbol{\lambda}^\star\right\|_1, \left\|\hat{\boldsymbol{\lambda}}^\star\right\|_1, \left\|\tilde{\boldsymbol{\lambda}}^\star\right\|_1\right)$$

Sources of error

parametrization richness (ν) sample size (N) requirements difficulty ($\boldsymbol{\lambda}^\star$)

[Chamon and Ribeiro, NeurIPS'20; Chamon, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23; Elenter, Chamon, Ribeiro, ICLR'24]

Dual (near-)PACC learning

Theorem

Let f be ν-universal with VC dimension $d_{\text{VC}} < \infty$, ℓ strongly convex, and g convex. Then, $f_{\boldsymbol{\theta}^\dagger}$ is a (near-)PACC solution of (P-CSL) for all $(\boldsymbol{\theta}^\dagger, \boldsymbol{\lambda}^\dagger)$ that achieve $\hat{D}^\star$, i.e., with probability at least $1-\delta$,

$$\left| P^\star - \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\left[\ell\left(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\right)\right]\right| \leq (1+\Delta)\left(\epsilon_0 + \epsilon\right)$$

$$\mathbb{E}\left[g\left(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\right)\right] \leq c + (1+\Delta)^{3/2}\left(M\sqrt{\epsilon_0} + \epsilon\right)$$

$$\epsilon_0 = M\nu \qquad \epsilon = B\sqrt{\frac{1}{N}\left[1 + \log\left(\frac{4m(2N)^{d_{\text{VC}}}}{\delta}\right)\right]} \qquad \Delta = \max\left(\left\|\boldsymbol{\lambda}^\star\right\|_1, \left\|\hat{\boldsymbol{\lambda}}^\star\right\|_1, \left\|\tilde{\boldsymbol{\lambda}}^\star\right\|_1\right)$$

Sources of error

parametrization richness (ν) sample size (N) requirements difficulty ($\boldsymbol{\lambda}^\star$)

[Chamon and Ribeiro, NeurIPS'20; Chamon, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23; Elenter, Chamon, Ribeiro, ICLR'24]

Dual (near-)PACC learning

Theorem

Let f be ν-universal with VC dimension $d_{\text{VC}} < \infty$, ℓ strongly convex, and g convex. Then, $f_{\boldsymbol{\theta}^\dagger}$ is a (near-)PACC solution of (P-CSL) for all $(\boldsymbol{\theta}^\dagger, \boldsymbol{\lambda}^\dagger)$ that achieve $\hat{D}^\star$, i.e., with probability at least $1 - \delta$,

$$\left| P^\star - \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\left[\ell\big(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\big)\right]\right| \leq (1+\Delta)\big(\epsilon_0 + \epsilon\big)$$

$$\mathbb{E}\left[g\big(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\big)\right] \leq c + (1+\Delta)^{3/2}\big(M\sqrt{\epsilon_0} + \epsilon\big)$$

$$\epsilon_0 = M\nu \qquad \epsilon = B\sqrt{\frac{1}{N}\left[1 + \log\left(\frac{4m(2N)^{d_{\text{VC}}}}{\delta}\right)\right]} \qquad \Delta = \max\left(\left\|\boldsymbol{\lambda}^\star\right\|_1, \left\|\hat{\boldsymbol{\lambda}}^\star\right\|_1, \left\|\tilde{\boldsymbol{\lambda}}^\star\right\|_1\right)$$

Sources of error

parametrization richness (ν) sample size (N) requirements difficulty ($\boldsymbol{\lambda}^\star$)

[Chamon and Ribeiro, NeurIPS'20; Chamon, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23; Elenter, Chamon, Ribeiro, ICLR'24]

Dual (near-)PACC learning

Theorem

Let f be ν-universal with VC dimension $d_{\text{VC}} < \infty$, ℓ strongly convex, and g convex. Then, $f_{\boldsymbol{\theta}^\dagger}$ is a (near-)PACC solution of (P-CSL) for all $(\boldsymbol{\theta}^\dagger, \boldsymbol{\lambda}^\dagger)$ that achieve $\hat{D}^\star$, i.e., with probability at least $1-\delta$,

$$\left| P^\star - \mathbb{E}_{(\boldsymbol{x},y)\sim\mathfrak{D}}\left[\ell\left(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\right)\right]\right| \leq (1+\Delta)\left(\epsilon_0+\epsilon\right)$$

$$\mathbb{E}\left[g\left(f_{\boldsymbol{\theta}^\dagger}(\boldsymbol{x}), y\right)\right] \leq c + (1+\Delta)^{3/2}\left(M\sqrt{\epsilon_0}+\epsilon\right)$$

$$\epsilon_0 = M\nu \qquad \epsilon = B\sqrt{\frac{1}{N}\left[1+\log\left(\frac{4m(2N)^{d_{\text{VC}}}}{\delta}\right)\right]} \qquad \Delta = \max\left(\left\|\boldsymbol{\lambda}^\star\right\|_1, \left\|\hat{\boldsymbol{\lambda}}^\star\right\|_1, \left\|\tilde{\boldsymbol{\lambda}}^\star\right\|_1\right)$$

Sources of error

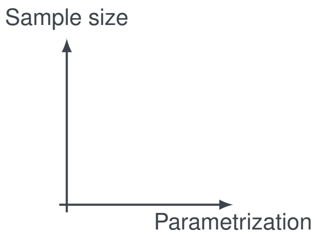
parametrization richness (ν) sample size (N) requirements difficulty ($\boldsymbol{\lambda}^\star$)

[Chamon and Ribeiro, NeurIPS'20; Chamon, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23; Elenter, Chamon, Ribeiro, ICLR'24]

Dual learning trade-offs

- Unconstrained learning

  parametrization $\times$ sample size



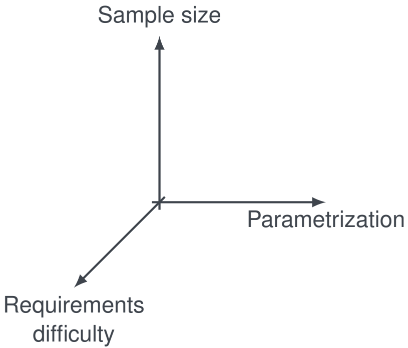
Dual learning trade-offs

- Unconstrained learning

  parametrization $\times$ sample size

- Constrained learning

  parametrization $\times$ sample size $\times$ requirements



When is constrained learning possible?

Corollary

$$f_\theta \text{ is PAC learnable } \approx^* f_\theta \text{ is PACC learnable}$$

Constrained learning is **essentially as hard as** unconstrained learning

(mild conditions apply)

[Chamon and Ribeiro, NeurIPS'20; Chamon, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23; Elenter, Chamon, Ribeiro, ICLR'24]

When is constrained learning possible?

Corollary

$$
\begin{tikzcd}
& \text{Uniform convergence} \arrow[dl, Rightarrow] \arrow[dr, Rightarrow] & \\
\text{PAC} \arrow[ur, dashed, bend left, "\text{0-1 loss}"] & & \text{PACC} \arrow[ll, Rightarrow]
\end{tikzcd}
$$

(mild conditions apply)

[Chamon and Ribeiro, NeurIPS'20; Chamon, Paternain, Calvo-Fullana, Ribeiro, IEEE TIT'23; Elenter, Chamon, Ribeiro, ICLR'24]

Claims

Constrained learning is the right way to learn under requirements

Constrained learning is hard. . .

. . . but possible. How?

Dual learning algorithm

$$\hat{D}^{\star} = \max_{\lambda \geq 0} \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \quad \frac{1}{N} \sum_{n=1}^{N} \ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda \left[\frac{1}{N} \sum_{m=1}^{N} g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\right) - c \right]$$

Dual learning algorithm

- Minimize the primal ($\equiv$ **ERM**)

$$\boldsymbol{\theta}^\dagger \in \operatorname*{argmin}_{\boldsymbol{\theta}\in\mathbb{R}^p} \ \frac{1}{N}\sum_{n=1}^{N}\left[\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right)\right]$$

$$\hat{D}^\star = \max_{\lambda\geq 0} \ \min_{\boldsymbol{\theta}\in\mathbb{R}^p} \quad \frac{1}{N}\sum_{n=1}^{N}\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda\left[\frac{1}{N}\sum_{m=1}^{N} g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\right) - c\right]$$

Dual learning algorithm

- Minimize the primal ($\equiv$ **ERM**)

$$\boldsymbol{\theta}^{+} \approx \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \Big[\ell\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) + \lambda g\big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\big) \Big], \quad n = 1, 2, \dots$$

[Ge et al., ICLR'18; Soltanolkotabi et al., IEEE TIT'18; Mei et al., PNAS'18; Kawaguchi et al., AISTATS'20...]

$$\hat{D}^{\star} = \max_{\lambda \geq 0} \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \quad \frac{1}{N} \sum_{n=1}^{N} \ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda \left[\frac{1}{N} \sum_{m=1}^{N} g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_m), y_m\right) - c \right]$$

Dual learning algorithm

- Minimize the primal ($\equiv$ **ERM**)

$$\theta^+ \approx \theta - \eta \nabla_\theta \Big[\ell\big(f_\theta(\boldsymbol{x}_n), y_n\big) + \lambda g\big(f_\theta(\boldsymbol{x}_n), y_n\big) \Big], \quad n = 1, 2, \ldots$$

- Update the dual

$$\lambda^+ = \left[\lambda + \eta \left(\frac{1}{N} \sum_{m=1}^{N} g\big(f_{\theta^+}(\boldsymbol{x}_m), y_m\big) - c \right) \right]_+$$

$$\hat{D}^\star = \max_{\lambda \geq 0} \min_{\theta \in \mathbb{R}^p} \quad \frac{1}{N} \sum_{n=1}^{N} \ell\left(f_\theta(\boldsymbol{x}_n), y_n\right) + \lambda \left[\frac{1}{N} \sum_{m=1}^{N} g\left(f_\theta(\boldsymbol{x}_m), y_m\right) - c \right]$$

A (near-)PACC learner

Theorem

Suppose $\boldsymbol{\theta}^\dagger$ is a ρ-approximate solution of the regularized ERM:

$$\boldsymbol{\theta}^\dagger \approx \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\operatorname{argmin}} \ \frac{1}{N}\sum_{n=1}^{N}\Big(\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right) + \lambda g\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n\right)\Big).$$

Then, after $T = \left\lceil \frac{\|\boldsymbol{\lambda}^\star\|^2}{2\eta M\nu} \right\rceil + 1$ dual iterations with step size $\eta \leq \frac{2\epsilon}{mB^2}$,

the iterates $\left(\boldsymbol{\theta}^{(T)}, \boldsymbol{\lambda}^{(T)}\right)$ are such that

$$\left| P^\star - L\left(\boldsymbol{\theta}^{(T)}, \boldsymbol{\lambda}^{(T)}\right)\right| \leq (2+\Delta)(\epsilon_0+\epsilon) + \rho$$

with probability $1-\delta$ over sample sets.

[Chamon et al., IEEE TIT'23]

In practice…

Initialize: $\boldsymbol{\theta}_0$, λ_0

for $t = 1, \dots, T$

  $\boldsymbol{\beta}_1 \leftarrow \boldsymbol{\theta}_{t-1}$

  for $n = 1, \dots, N$

    $\boldsymbol{\beta}_{n+1} \leftarrow \boldsymbol{\beta}_n - \eta_\theta \nabla_{\boldsymbol{\beta}} \left[\ell\big(f_{\boldsymbol{\beta}_n}(\boldsymbol{x}_n), y_n\big) + \lambda_{t-1} g\big(f_{\boldsymbol{\beta}_n}(\boldsymbol{x}_n), y_n\big) \right]$

  end

  $\boldsymbol{\theta}_t \leftarrow \boldsymbol{\beta}_{N+1}$

SGD

  $\lambda_t = \left[\lambda_{t-1} + \eta_\lambda \left(\frac{1}{N} \sum_{m=1}^{N} g\big(f_{\boldsymbol{\theta}_t}(\boldsymbol{x}_m), y_n\big) - c \right) \right]_+$

Dual update

end

Output: $\boldsymbol{\theta}_T, \lambda_T$

PyTorch

https://github.com/lfochamon/csl

Penalty-based vs. dual learning

Penalty-based learning

$$\theta^\dagger \in \underset{\theta}{\operatorname{argmin}}\ \text{Loss}(\theta) + \lambda \cdot \text{Penalty}(\theta)$$

- Parameter: λ (data-dependent)
- Generalizes with respect to Loss $+\ \lambda$Penalty

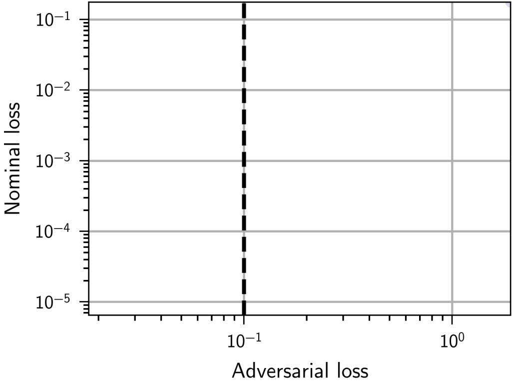
Dual learning

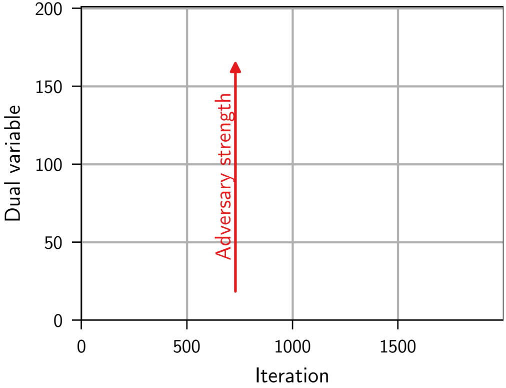
$$\theta^\dagger \in \underset{\theta}{\operatorname{argmin}}\ \text{Loss}(\theta) + \lambda \cdot \text{Penalty}(\theta)$$

$$\lambda^+ = \left[\lambda + \eta\left(\text{Penalty}(\theta^\dagger) - c\right)\right]_+$$

- Parameter: c (requirement-dependent)
- Generalizes with respect to Loss and Penalty $\leq c$

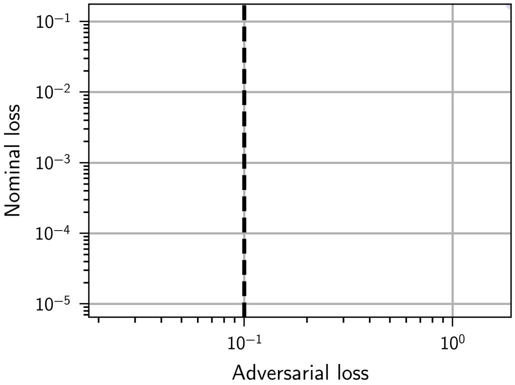
Robust image recognition

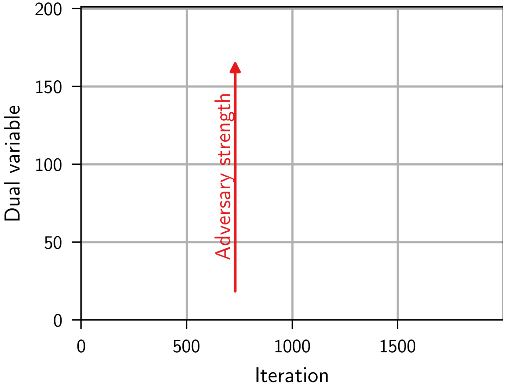




[**C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

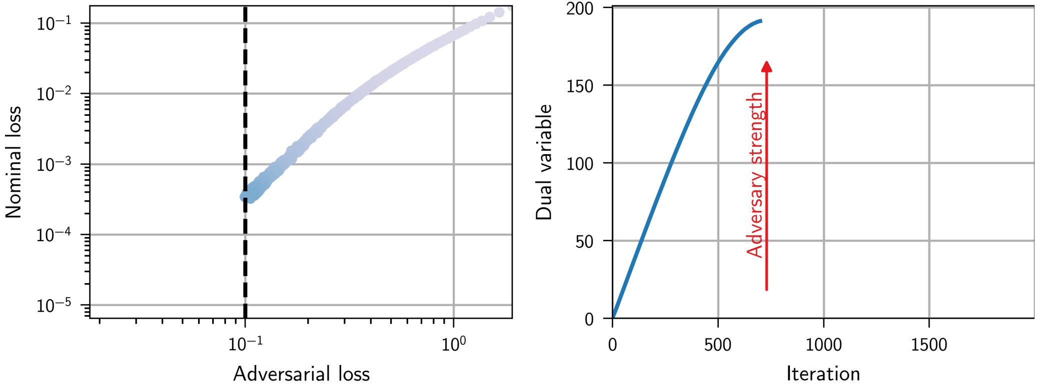
Robust image recognition





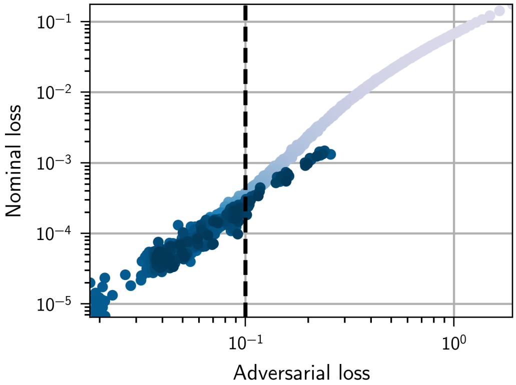
[C., Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

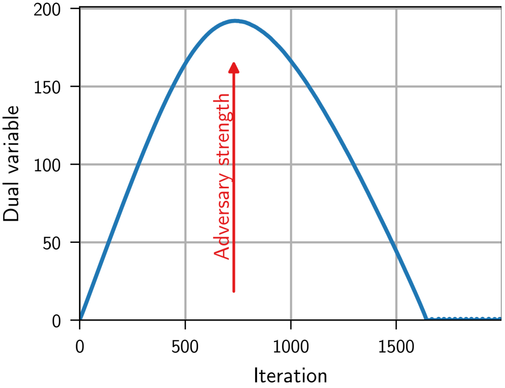
Robust image recognition



[**C.**, Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

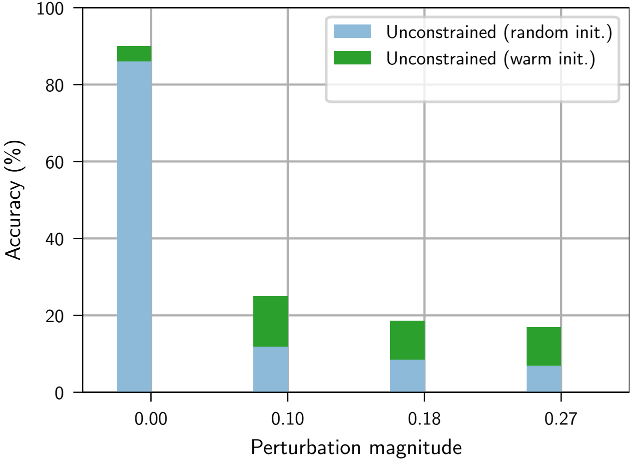
Robust image recognition





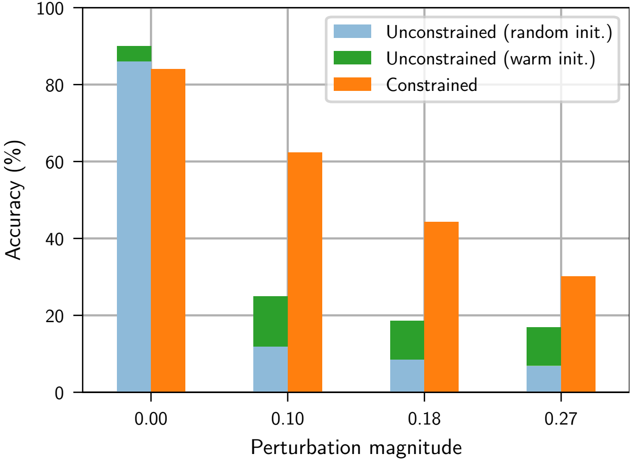
Empirical observations: [Zhang et al., ICML'20; Sitawarin, ArXiv'20]

Robust image recognition



[C., Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Robust image recognition



[C., Paternain, Calvo-Fullana, and Ribeiro, IEEE TIT'23]

Claims

- Constrained learning is the right tool to learn under requirements

- Constrained learning is hard. . .

- . . . but possible

Claims

- **Constrained learning is ~~the right~~ a good tool to learn under requirements**

  Constrained learning imposes requirements during training that generalize at test time, e.g.,

  - robustness [**C**R, NeurIPS'20; R*C*PH, NeurIPS'21; R**C**PH, ICML'22 (spotlight); **C**PCR, IEEE TIT'23]
  - fairness [**C**PCR, ICASSP'20 (best student paper); **C**R, NeurIPS'20; **C**PCR, IEEE TIT'23]
  - invariance and data augmentation [H**C**R, ICML'23]
  - (manifold) smoothness [C**C**HVR, ICML'23]
  - resilience [HR**C**, NeurIPS'23]
  - safe RL [P**C**CR, NeurIPS'19; PC**C**R, IEEE TAC'23; CP**C**R, IEEE TAC'24]
  - learning to solve PDEs [M**C**, ICLR'25]
  - . . .

- **Constrained learning is hard. . .**

- **. . . but possible**

Claims

- **Constrained learning is ~~the right~~ a good tool to learn under requirements**

  Constrained learning imposes requirements during training that generalize at test time, e.g.,
  - robustness [**C**R, NeurIPS'20; R*C*PH, NeurIPS'21; R**C**PH, ICML'22 (spotlight); **C**PCR, IEEE TIT'23]
  - fairness [**C**PCR, ICASSP'20 (best student paper); **C**R, NeurIPS'20; **C**PCR, IEEE TIT'23]
  - invariance and data augmentation [H**C**R, ICML'23]
  - (manifold) smoothness [C**C**HVR, ICML'23]
  - resilience [HR**C**, NeurIPS'23]
  - safe RL [P**C**CR, NeurIPS'19; PC**C**R, IEEE TAC'23; CP**C**R, IEEE TAC'24]
  - learning to solve PDEs [M**C**, ICLR'25]
  - . . .

- **Constrained learning is hard. . .**

  Constrained, non-convex, statistical optimization problem

- **. . . but possible**

Claims

- **Constrained learning is ~~the right~~ a good tool to learn under requirements**

  Constrained learning imposes requirements during training that generalize at test time, e.g.,

  - robustness [**C**R, NeurIPS'20; R*__C__*PH, NeurIPS'21; R**C**PH, ICML'22 (spotlight); **C**PCR, IEEE TIT'23]
  - fairness [**C**PCR, ICASSP'20 (best student paper); **C**R, NeurIPS'20; **C**PCR, IEEE TIT'23]
  - invariance and data augmentation [H**C**R, ICML'23]
  - (manifold) smoothness [C**C**HVR, ICML'23]
  - resilience [HR**C**, NeurIPS'23]
  - safe RL [P**C**CR, NeurIPS'19; PC**C**R, IEEE TAC'23; CP**C**R, IEEE TAC'24]
  - learning to solve PDEs [M**C**, ICLR'25]
  - . . .

- **Constrained learning is hard. . .**

  Constrained, non-convex, statistical optimization problem

- **. . . but possible**

  We can learn under requirements (essentially) whenever we can learn at all

Claims

- **Constrained learning is ~~the right~~ a good tool to learn under requirements**

  Constrained learning imposes requirements during training that generalize at test time, e.g.,

  - robustness [**C**R, NeurIPS'20; R***C***PH, NeurIPS'21; R**C**PH, ICML'22 (spotlight); **C**PCR, IEEE TIT'23]
  - fairness [**C**PCR, ICASSP'20 (best student paper); **C**R, NeurIPS'20; **C**PCR, IEEE TIT'23]
  - invariance and data augmentation [H**C**R, ICML'23]
  - (manifold) smoothness [C**C**HVR, ICML'23]
  - resilience [HR**C**, NeurIPS'23]
  - safe RL [P**C**CR, NeurIPS'19; PC**C**R, IEEE TAC'23; CP**C**R, IEEE TAC'24]
  - learning to solve PDEs [M**C**, ICLR'25]
  - . . .

- **Constrained learning is hard. . .**

  Constrained, non-convex, statistical optimization problem

- **. . . but possible. How?**

  We can learn under requirements (essentially) whenever we can learn at all by solving *(penalized) ERM problems*

learning under requirements

Luiz F. O. Chamon

École Polytechnique

Institut Polytechnique de Paris

Collaborators:

Aneesh Barthakur, Miguel Calvo-Fullana, Juan Cerviño, Mark Eisen, Yonina C. Eldar, Hamed Hassani, Ignacio Hounie, Dyonisios Kalogerias, Dan D. Lee, Viggo Moro, George J. Pappas, Santiago Paternain, Alejandro Ribeiro, Alexander Robey, Luana Ruiz, Anastasios Tsiamis, Rene Vidal, Clark Zhang

www.luizchamon.com

luiz.chamon@polytechnique.edu

ellis