Massachusetts Institute of Technology

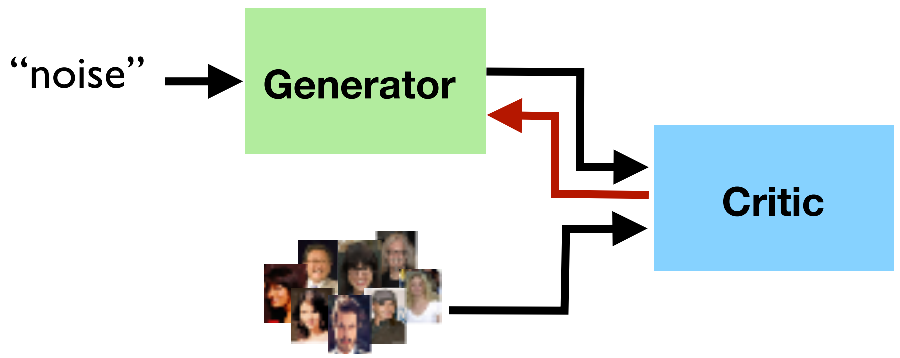
CSAIL

Robustness in GANs and in Black-box Optimization

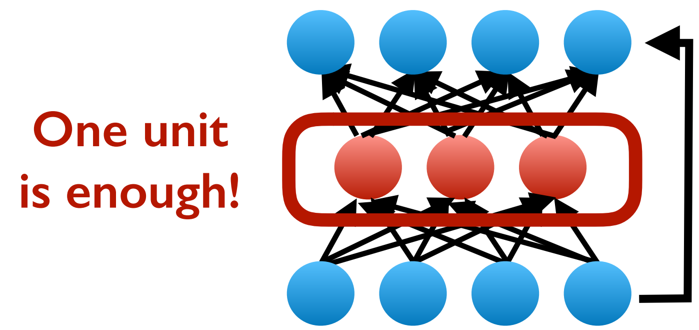
Stefanie Jegelka
MIT CSAIL

joint work with Zhi Xu, Chengtao Li,
Ilija Bogunovic, Jonathan Scarlett and Volkan Cevher

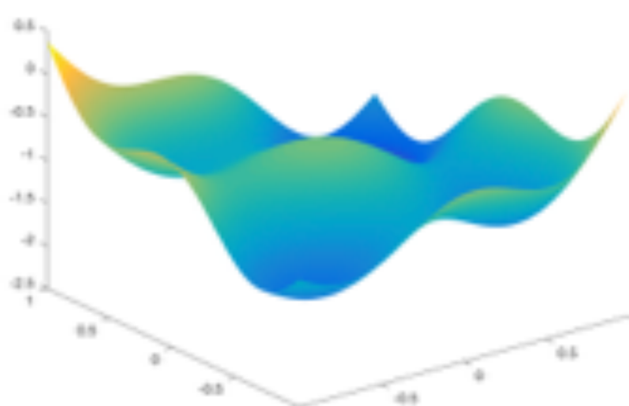
Robustness in ML



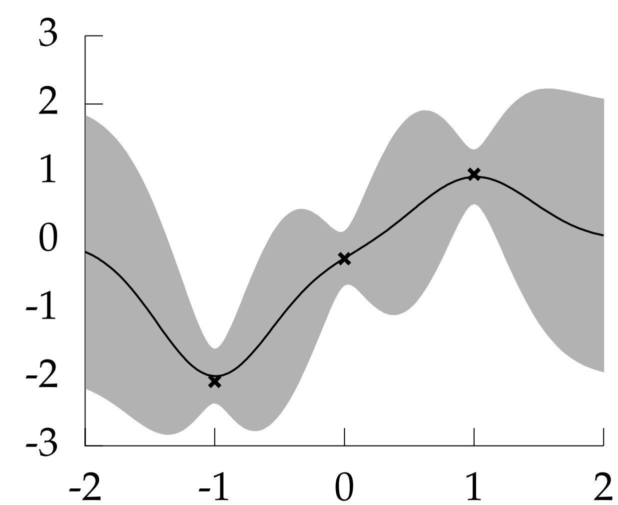
Robustness
in GANs



Representational Power
in Deep Learning

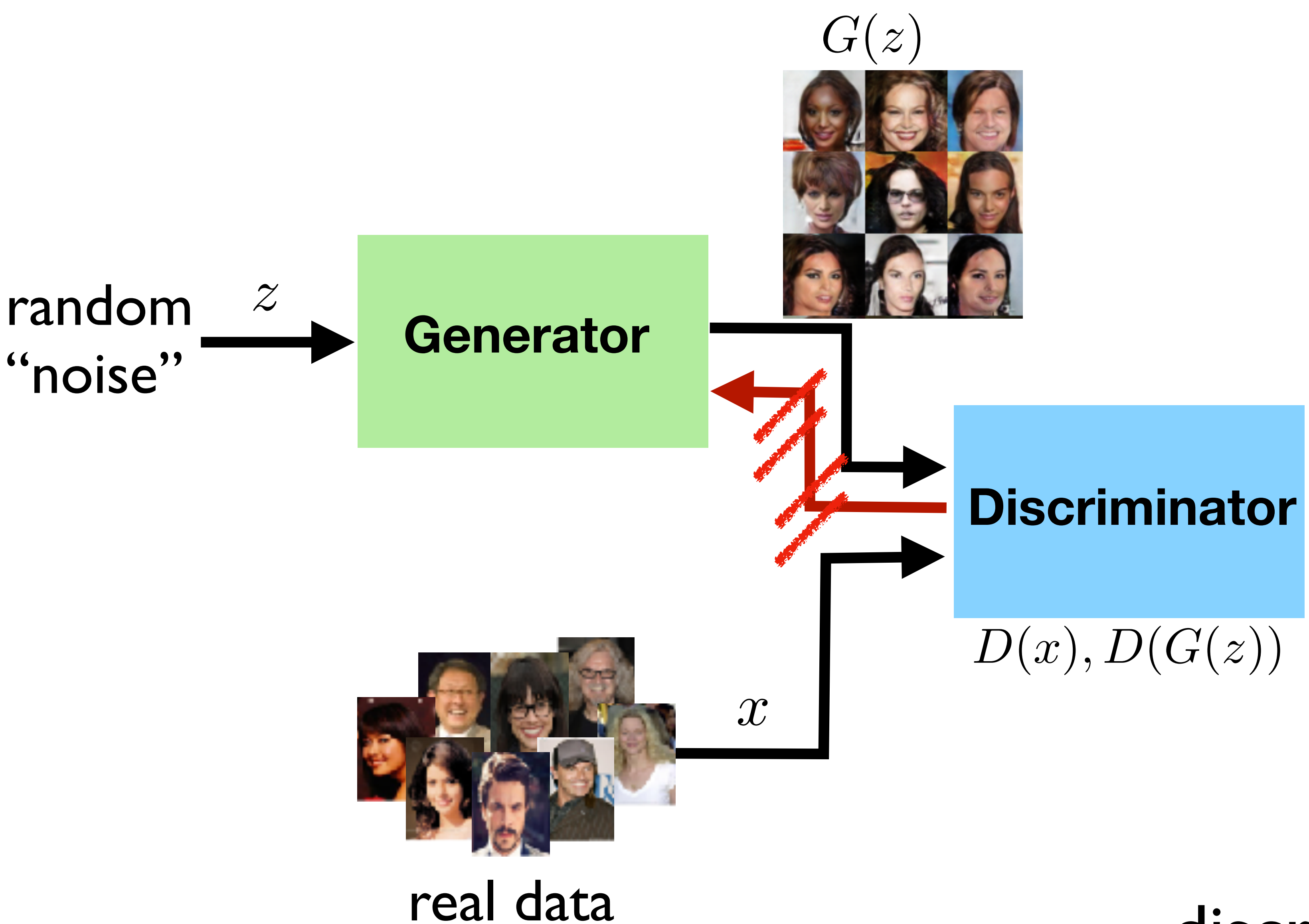


Robust Optimization,
Generalization,
Discrete & Nonconvex
Optimization



Robust Black-Box
Optimization

Generative Adversarial Networks



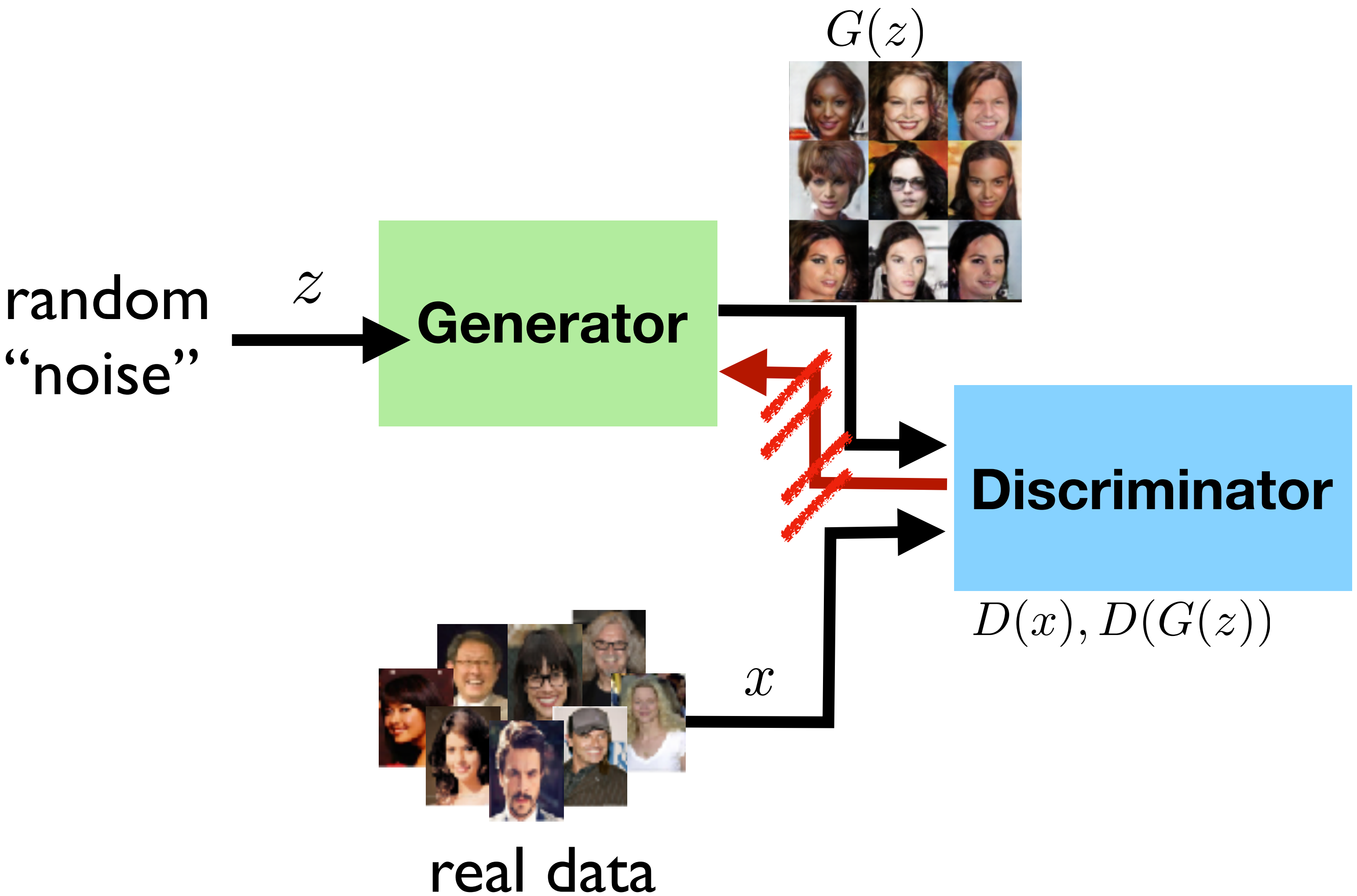
- attack:
  with probability p<0.5, discriminator’s output is manipulated
- generator doesn’t know which feedback is honest

discriminator: $\max_D V(G, D)$

generator: $\min_G V(G, A(D))$

$$\min_G \max_D V(G, D)$$

Generative Adversarial Networks

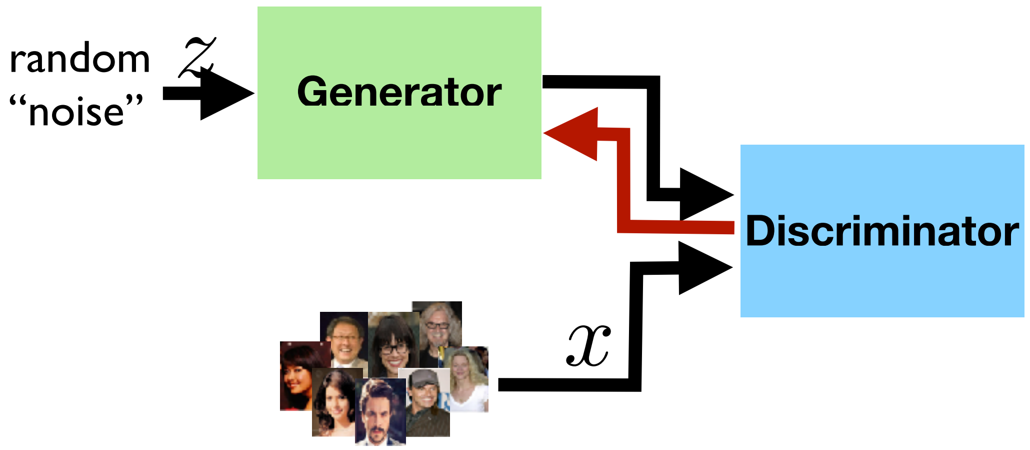


- attack:
  with probability $p<0.5$, discriminator's output is manipulated
- generator doesn't know which feedback is honest

$$A(D(x)) = \begin{cases} 1 - D(x) & \text{with probability } p \\ D(x) & \text{otherwise.} \end{cases}$$

Theorem: If adversary does a simple sign flip, then standard GAN no longer learns the right distribution.

What makes GANs more robust?



$$\min_G \max_D V(G, D)$$

1. Model properties: transformation function

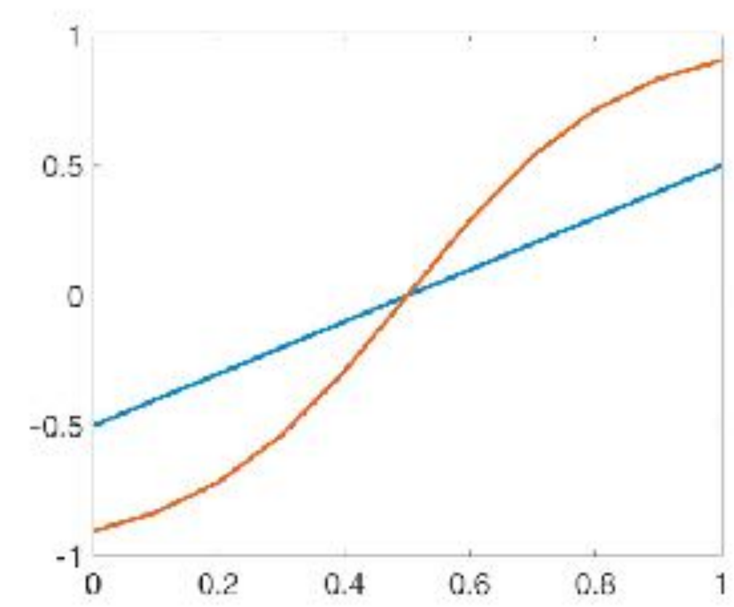
$$\min_G \max_D \mathbb{E}_{x\sim\mathbb{P}_{\text{data}}}[f(D(x))] + \mathbb{E}_{z\sim\mathbb{P}_z}[f(1 - D(G(z)))]$$

score for real data

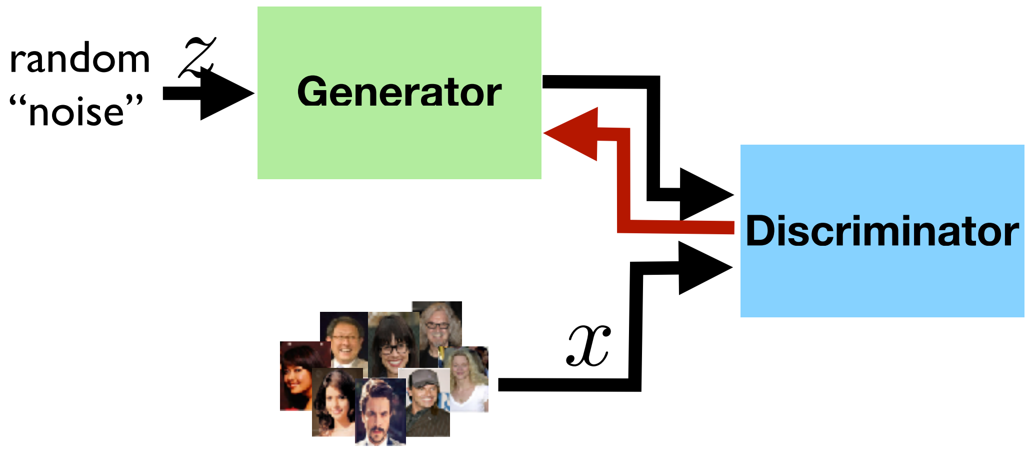
“inverse” score for fake data

If:
- f is strictly increasing and differentiable
- symmetry: $f(a) = -f(1-a)$ for $a \in [0, 1]$

then model is robust



What makes GANs more robust?



$$\min_G \max_D V(G, D)$$

1. Model properties:
   symmetric transformation function

2. Training Algorithm:
   weight clipping (regularization) helps robustness

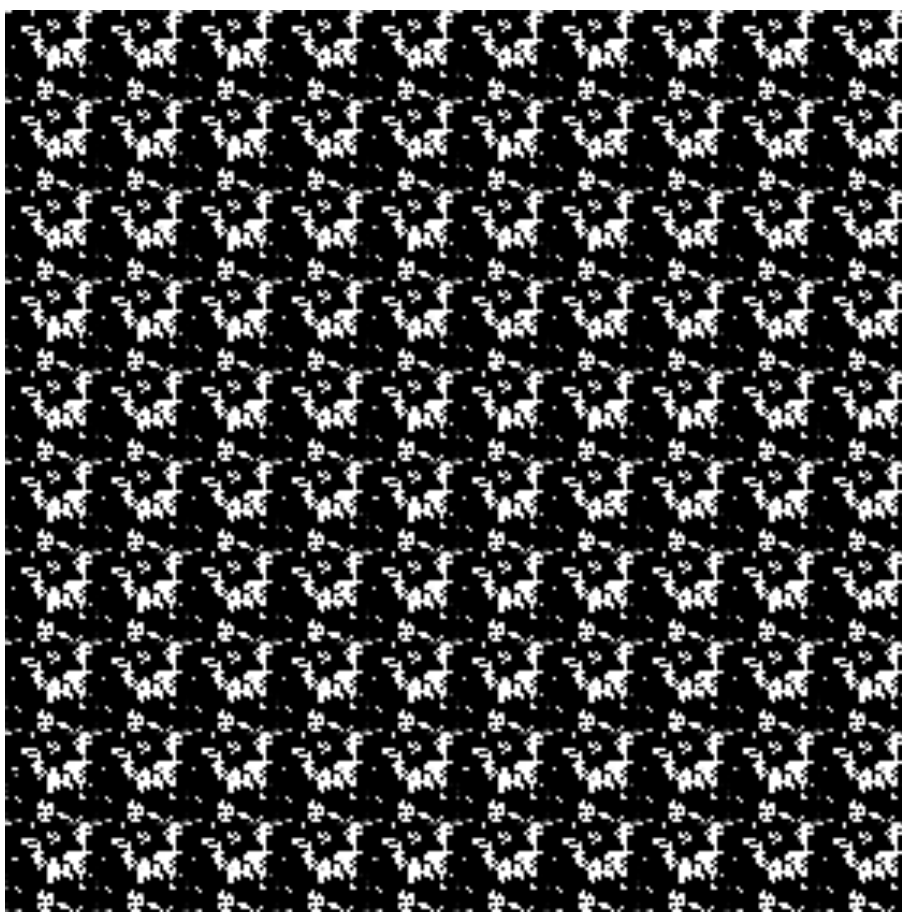
Generalization includes Wasserstein GANs!

Empirical Results

- MNIST

GAN

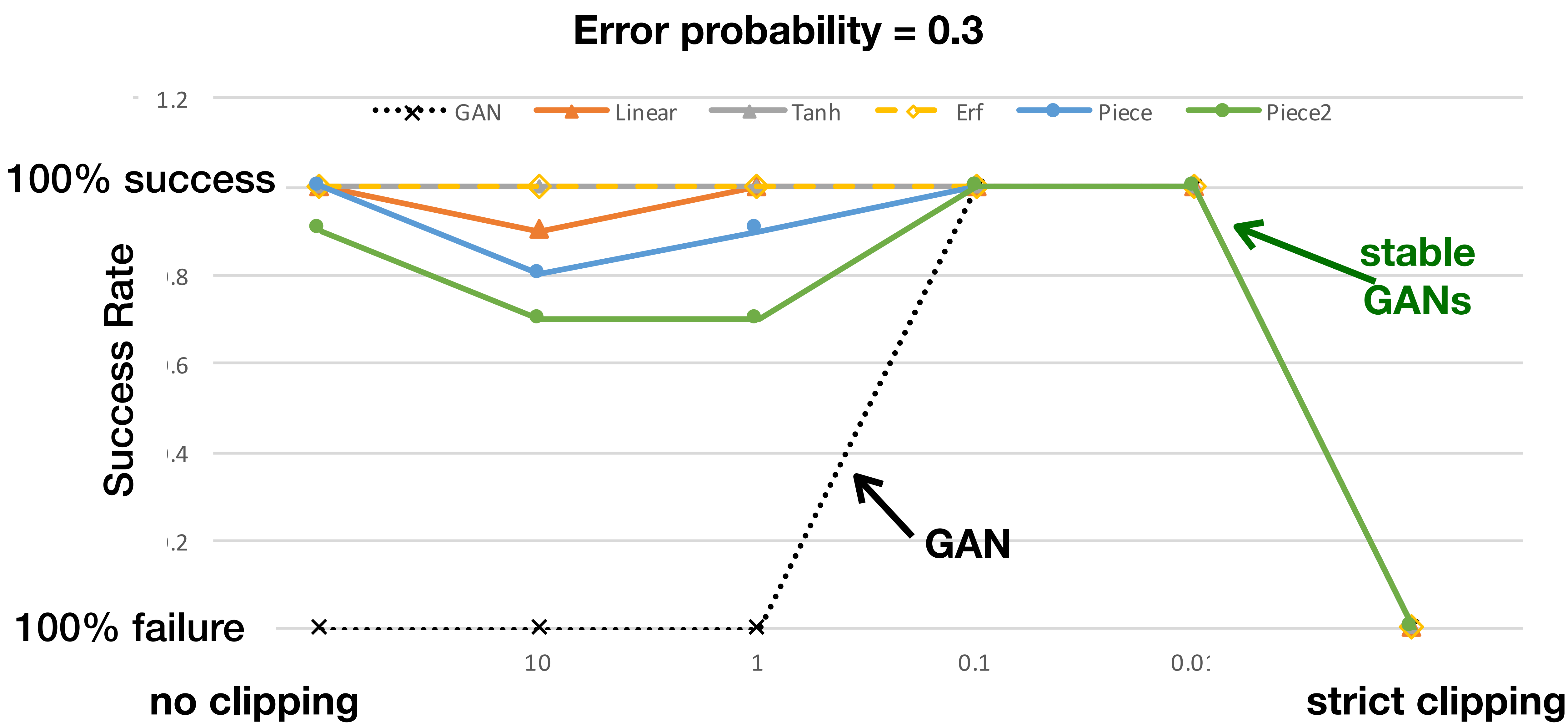
Error Probability = 0.3



stable GAN

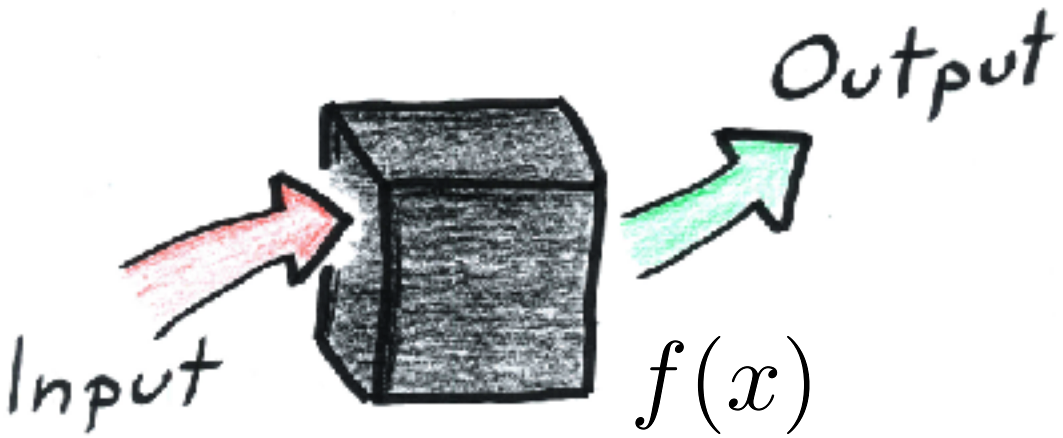
Error Probability = 0.3

Empirical Results

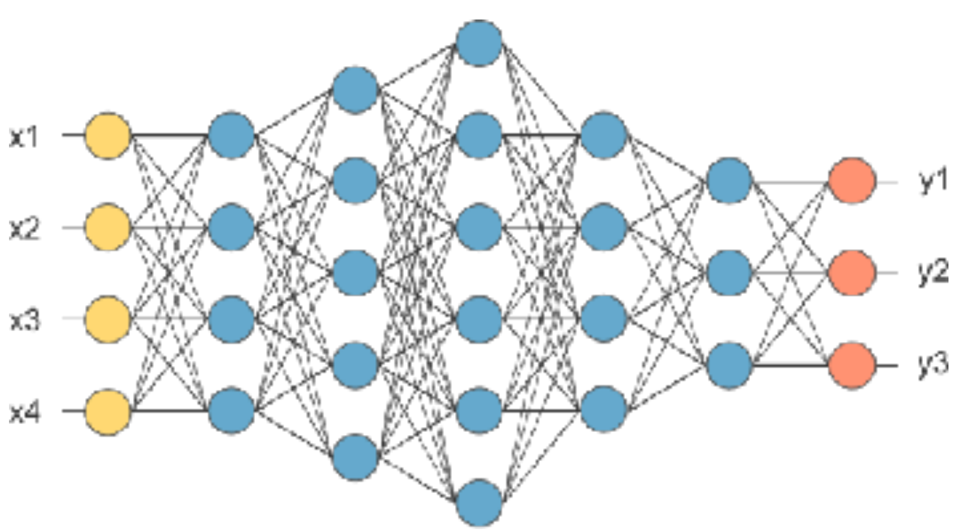


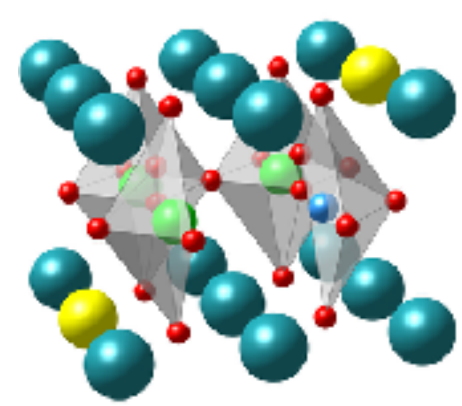
- Weight Clipping (Regularization) helps robustness
- Stable GANs are more robust to clipping threshold

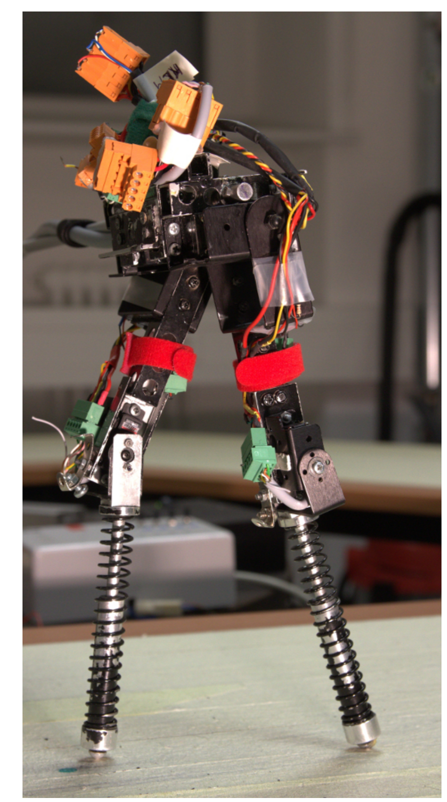
Black-Box Optimization



Goal:
optimize a complex function that is only accessible via expensive queries







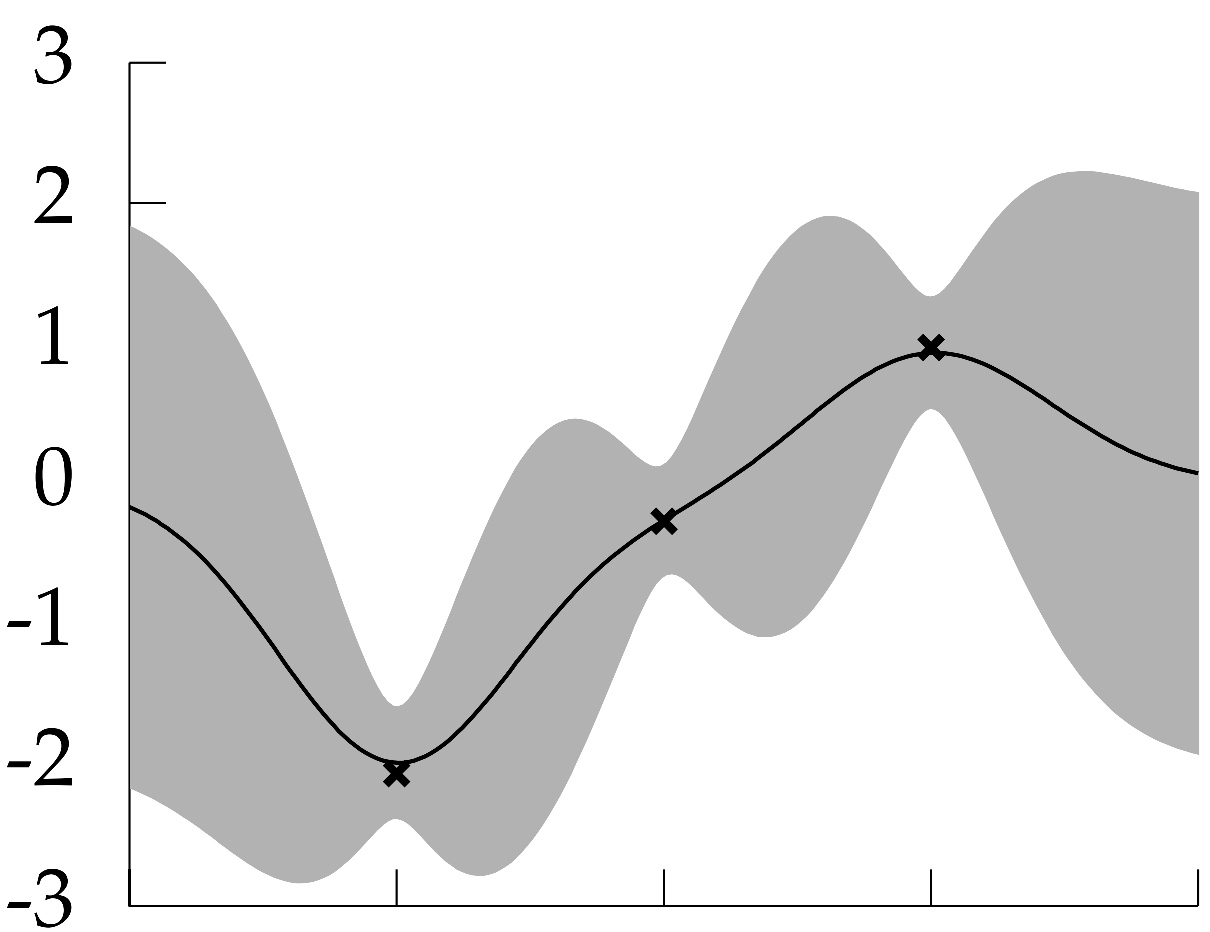
Black-Box Optimization

Sequential Optimization: build internal model of *f(x)*

In each time step:

- Select a query point *x*
- Observe (noisy) *f(x)*
- update model

often:
Gaussian Process



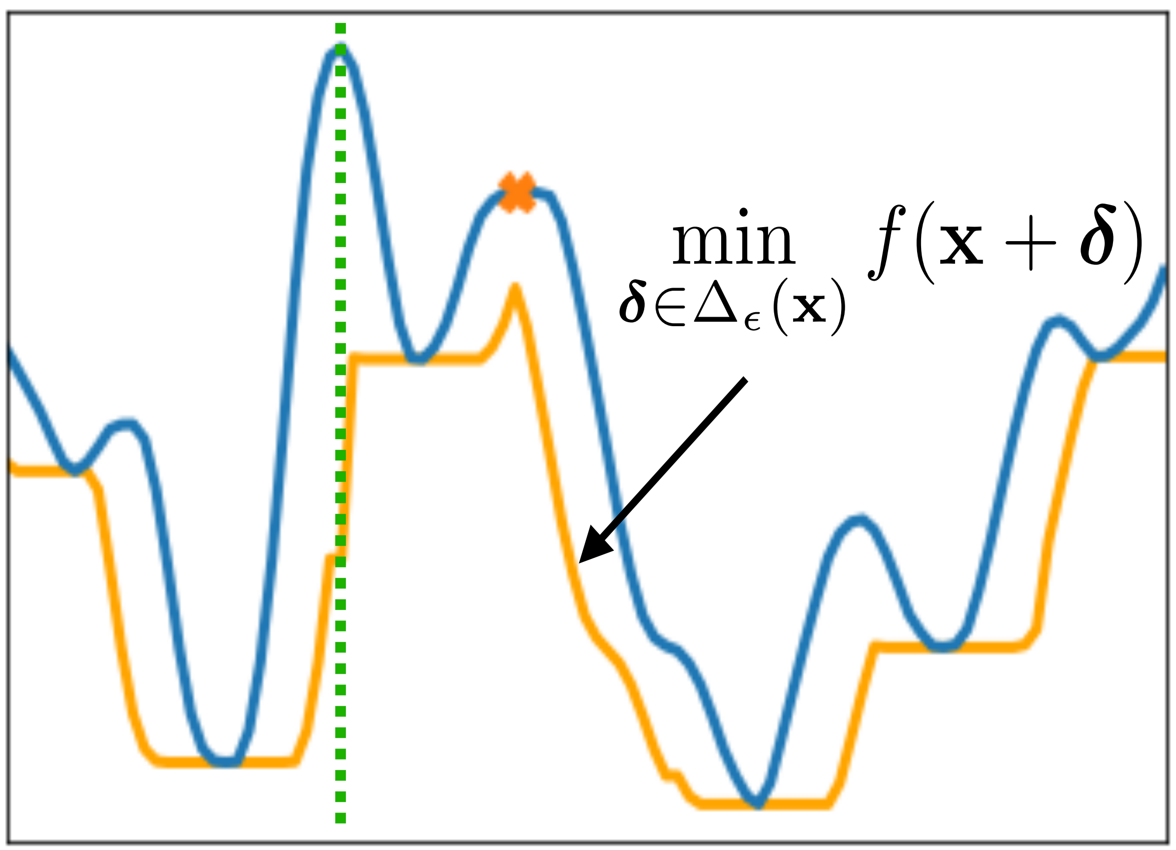
select queries:
uncertainty and expected function value,
e.g. maximizing

$$\mathrm{ucb}(x) = \hat{f}(x) + \beta^{1/2}\sigma(x)$$

Robust Optimization with GPs

Observed f(x) may be (adversarially) perturbed:

- Robotics: simulations vs. real data
- Parameter tuning: estimation with limited data
- Time-varying functions



suboptimal decision!

Problem: $\max_{\boldsymbol{x} \in D} \min_{\boldsymbol{\delta} \in \Delta_\epsilon(\boldsymbol{x})} f(\boldsymbol{x} + \boldsymbol{\delta})$

$$\Delta_\epsilon(\boldsymbol{x}) = \left\{ \boldsymbol{x}' - \boldsymbol{x} \ : \ \boldsymbol{x}' \in D \ \text{ and } \ d(\boldsymbol{x}, \boldsymbol{x}') \le \epsilon \right\}$$

standard methods can fail!

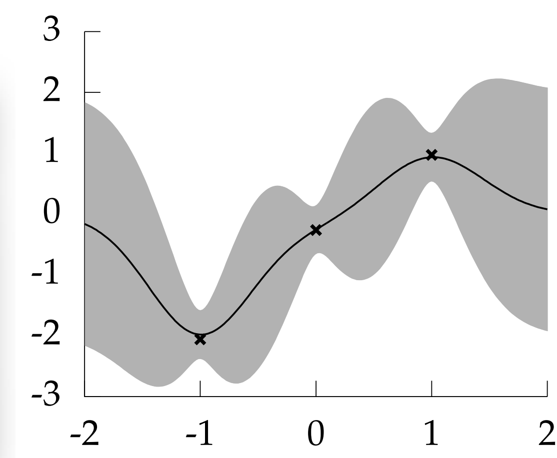
StableOpt Algorithm

In every round:

- select $\tilde{\boldsymbol{x}}_t = \underset{\boldsymbol{x} \in D}{\operatorname{argmax}} \min_{\boldsymbol{\delta} \in \Delta_\epsilon(\boldsymbol{x})} \operatorname{ucb}_{t-1}(\boldsymbol{x} + \boldsymbol{\delta})$ *upper confidence bound*
- select $\boldsymbol{\delta}_t = \underset{\boldsymbol{\delta} \in \Delta_\epsilon(\tilde{\boldsymbol{x}}_t)}{\operatorname{argmin}} \operatorname{lcb}_{t-1}(\tilde{\boldsymbol{x}}_t + \boldsymbol{\delta})$ *lower confidence bound*
- observe $y_t = f(\tilde{\boldsymbol{x}}_t + \boldsymbol{\delta}_t) + z_t$ and update with $\{(\tilde{\boldsymbol{x}}_t + \boldsymbol{\delta}_t, y_t)\}$

Theorem (RBF kernel): sample complexity:

$O^*\left(\frac{1}{\eta^2}\left(\log \frac{1}{\eta}\right)^{2p}\right)$ steps for η -suboptimality whp.



- closely matching lower bound
- algorithm generalizes to many other settings!

Variations

Robustness to unknown parameters

- $f : D \times \Theta \to \mathbb{R}$ is smooth wrt input x and parameters θ

$$\boxed{\max_{\boldsymbol{x} \in D} \min_{\boldsymbol{\theta} \in \Theta} f(\boldsymbol{x}, \boldsymbol{\theta})}$$

Robust estimation

- $\bar{\theta}$ estimate of true parameters $\theta^* \in \Theta$

$$\boxed{\max_{\boldsymbol{x} \in D} \min_{\boldsymbol{\delta_\theta} \in \Delta_\epsilon(\bar{\boldsymbol{\theta}})} f(\boldsymbol{x}, \bar{\boldsymbol{\theta}} + \boldsymbol{\delta_\theta})}$$

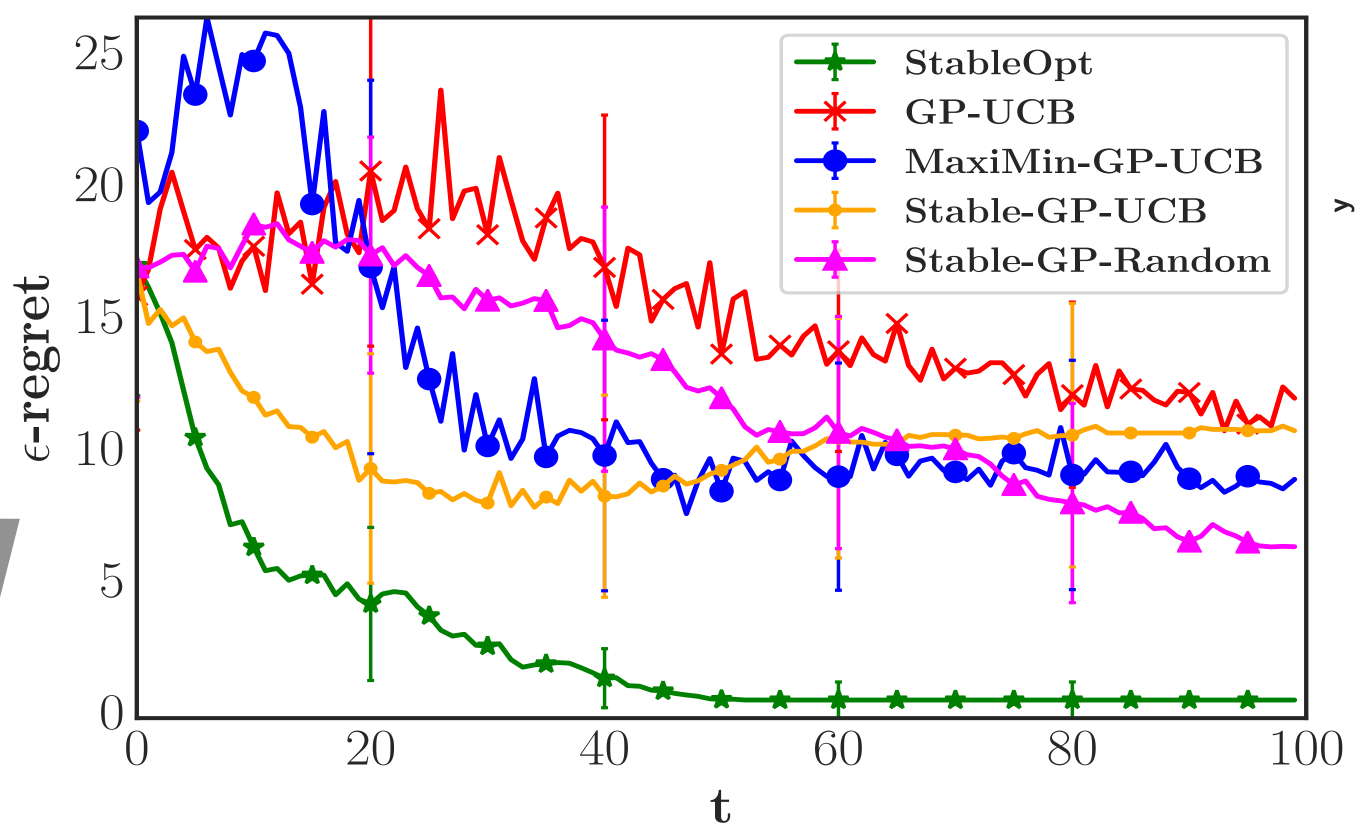
Robust group identification

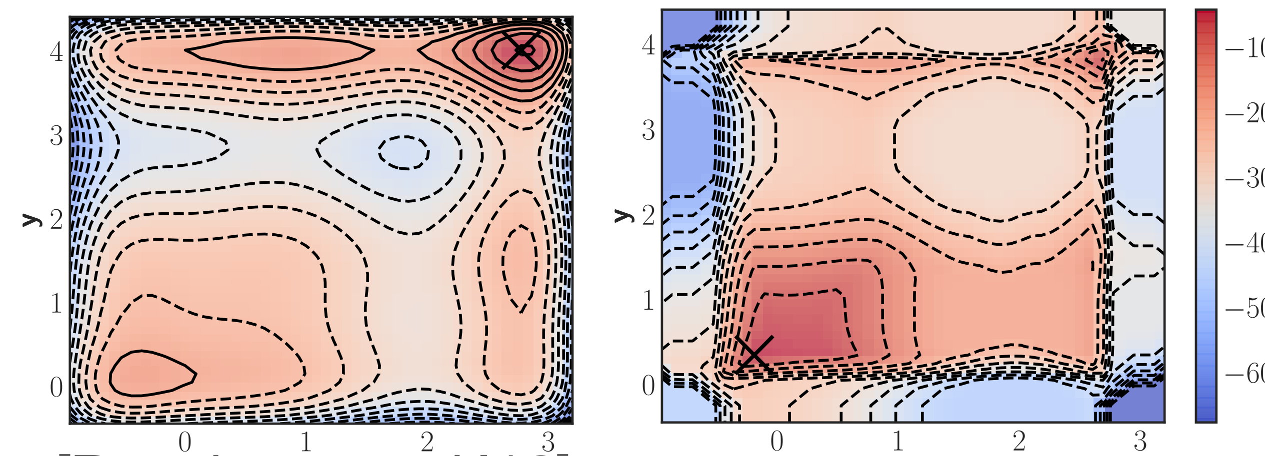
- input space partitioned into groups $\mathscr{G} = \{G_1, \ldots, G_k\}$
  group with highest worst-case value $\boxed{\max_{G \in \mathscr{G}} \min_{\boldsymbol{x} \in G} f(\boldsymbol{x})}$

Empirical Results

Benchmark

better

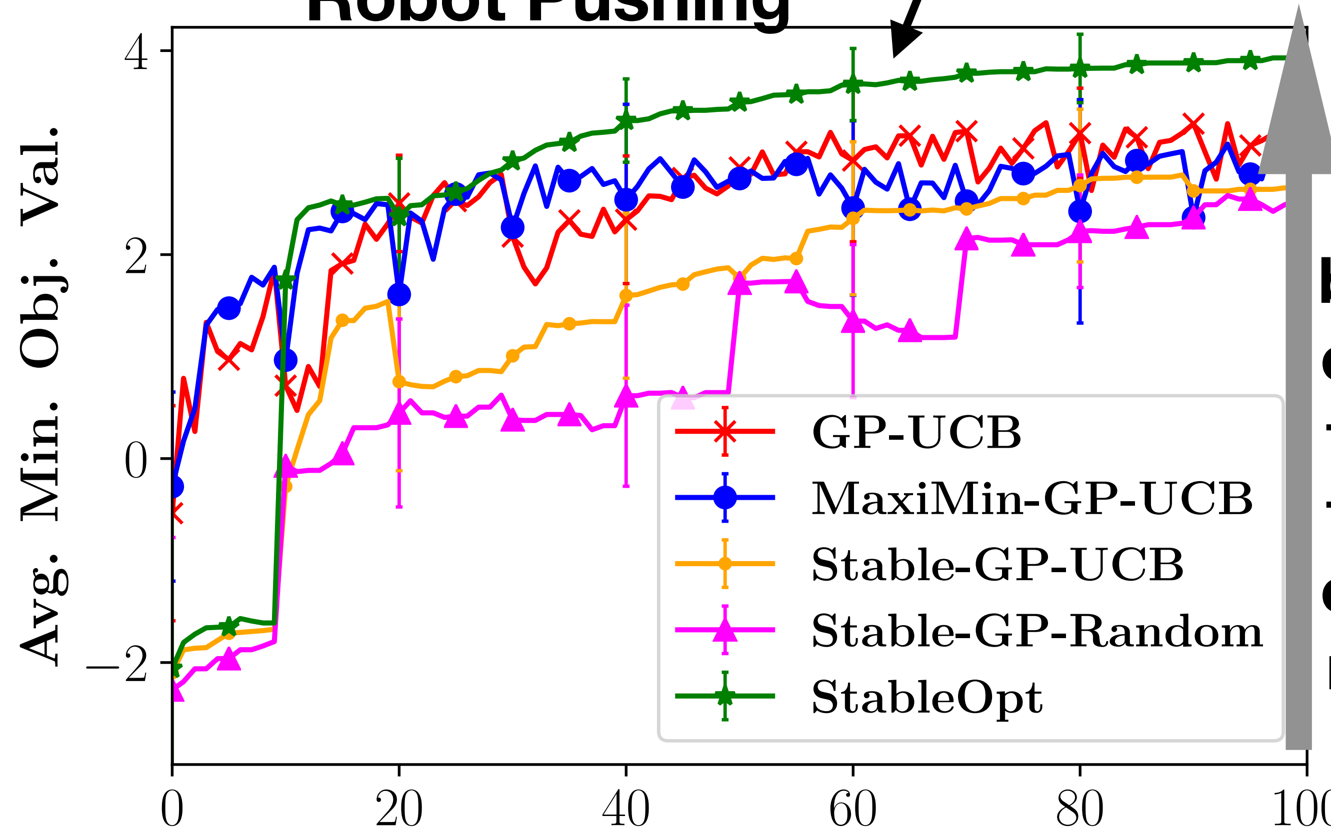




[Bertsimas *et al.*'10]

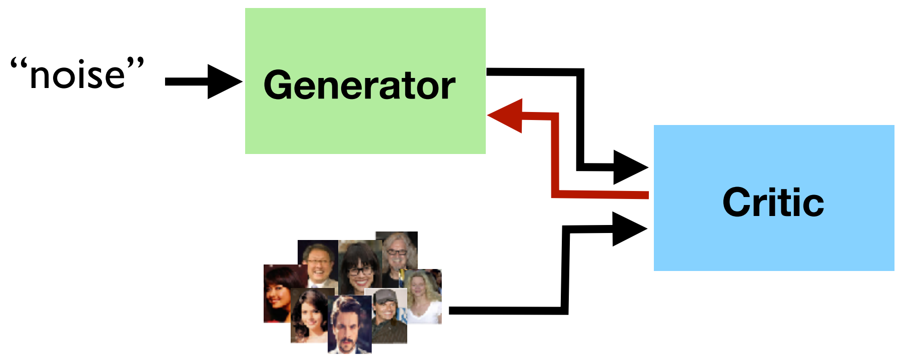
StableOpt

Robot Pushing

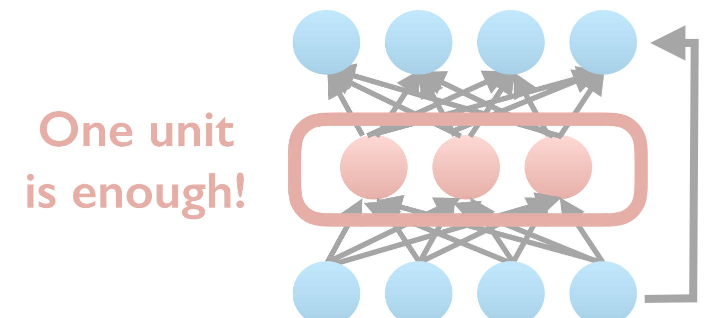


better

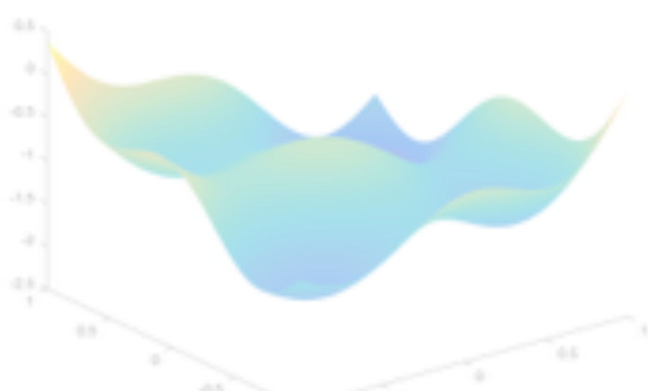
Robustness in ML



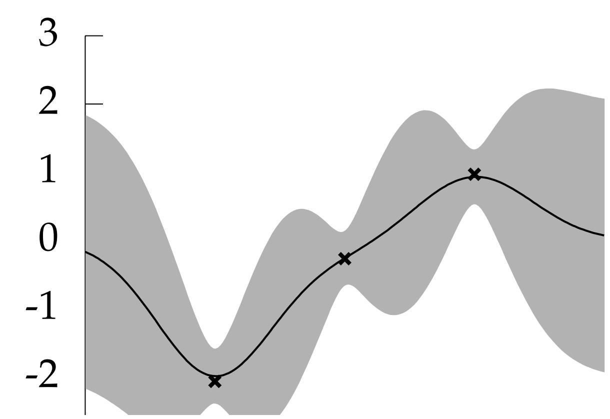
Robustness in GANs
(Z. Xu, C. Li, S. Jegelka, arXiv)



Representational Power in Deep Learning



Robust Optimization, Generalization, Discrete & Nonconvex Optimization



Robust Black-Box Optimization
(I. Bogunovic, J. Scarlett, S. Jegelka, V. Cevher NIPS 2018)