Deep Adversarial Learning

Jihun Hamm

In collaboration with Akshay Mehra and Yungkyun Noh

Tulane University

Content

1/4. Intro and examples

2/4. Privacy preservation

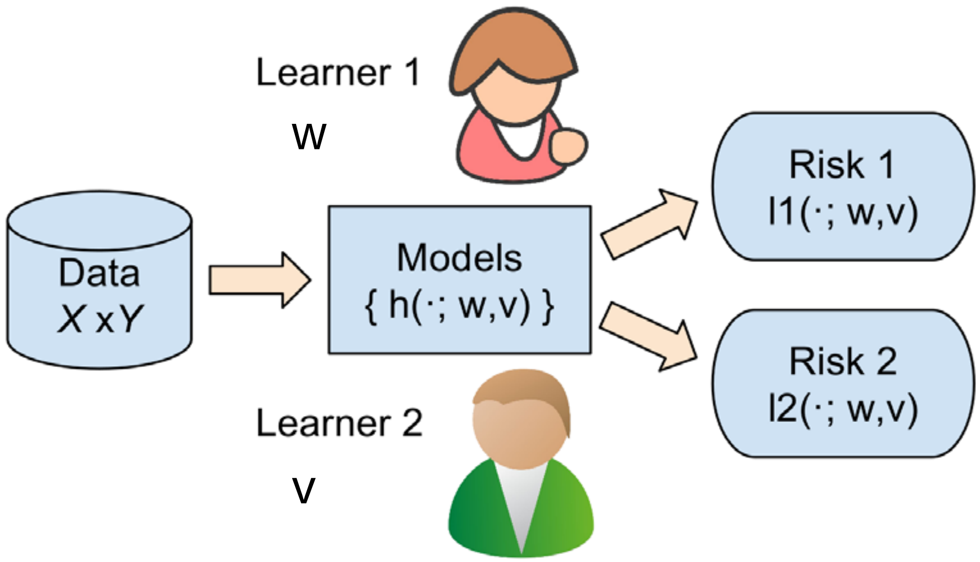
3/4. Optimization

4/4. Ongoing work

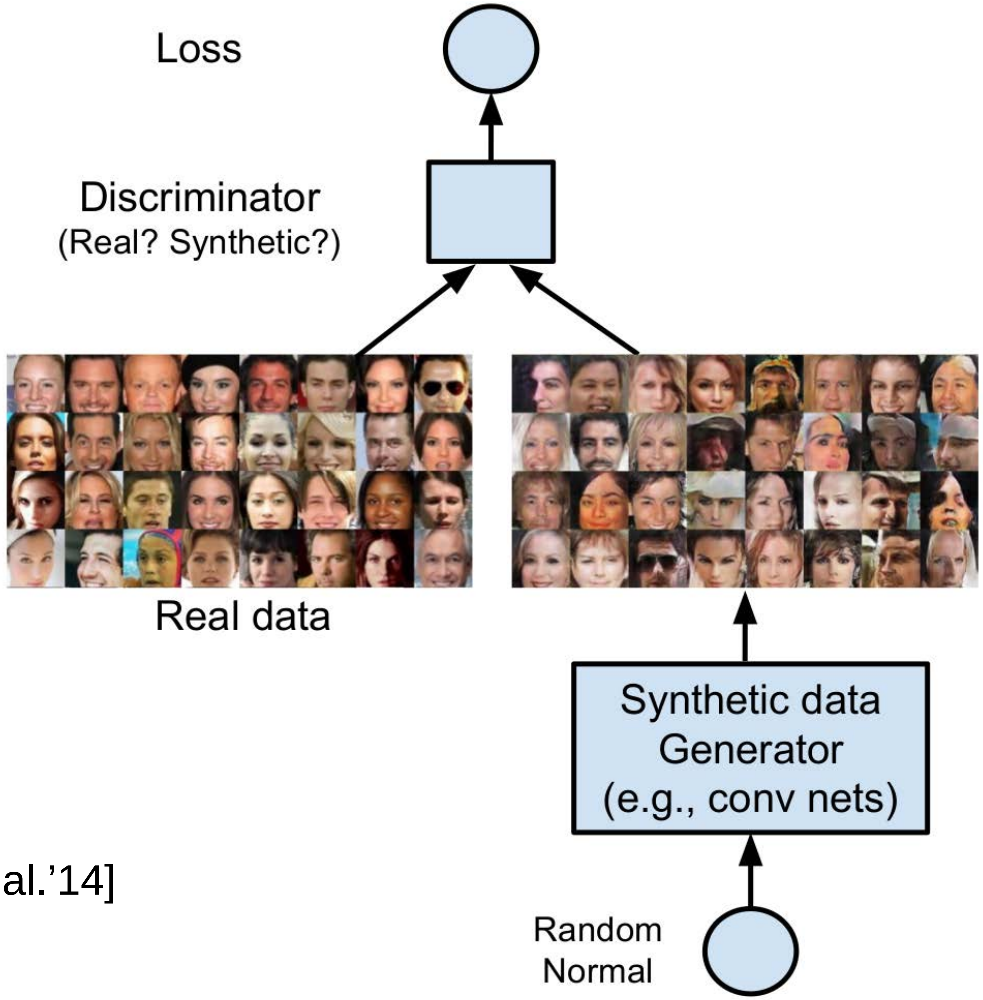
Part 1/4. Introduction

- (Standard) machine learning
  - Single learner & single objective

- Adversarial machine learning in broad sense
  - Multiple learners & multiple objectives
  - Objectives are often conflicting



Generative Adversarial Nets



- [Goodfellow et al.'14]

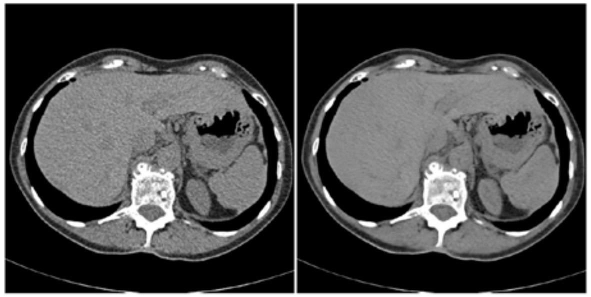
GAN-related work

- Variants
  - Deep convolutional GAN (DCGAN) [Radford et al.'15]
  - Conditional GAN [Mirza et al.'14]
  - Adversarially learned inference (ALI) [Dumoulin et al.'16]
  - Adversarial autoencoder (AAE) [Makhzani et al.'15]
  - Energy-based GAN (EBGAN) [Zhao et al.'16]
  - Wasserstein GAN (WGAN) [Arjovsky et al.'17]
  - Boundary equilibrium GAN (BEGAN) [Berthelot et al.'17]
  - Bayesian GAN [Saatchi et al.'17]
  - …
- Applications

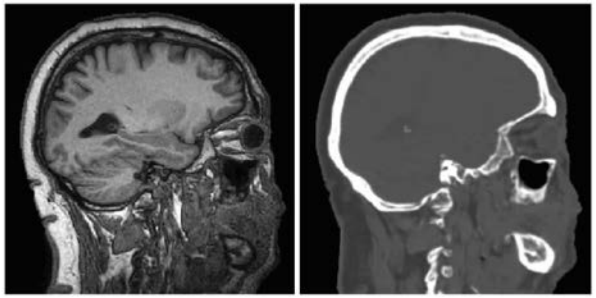
  …

GAN in Medical Imaging

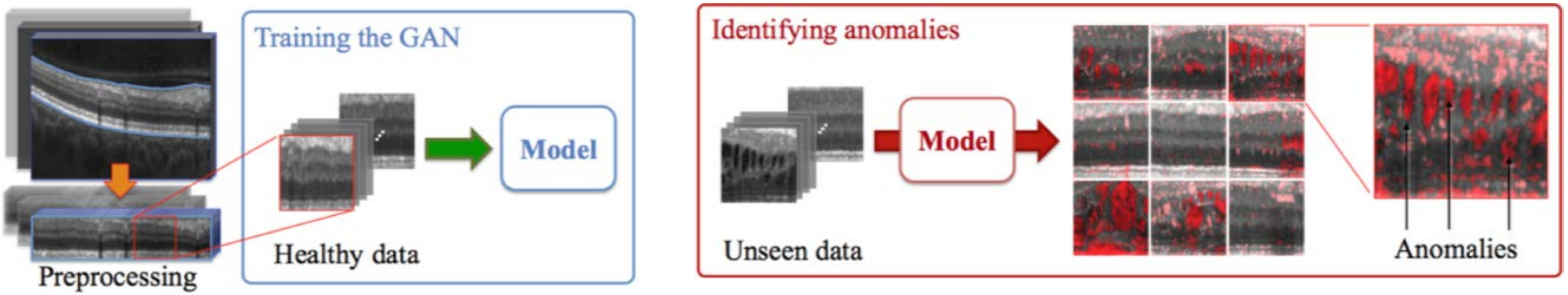
Denoising [Yi et al.'18]



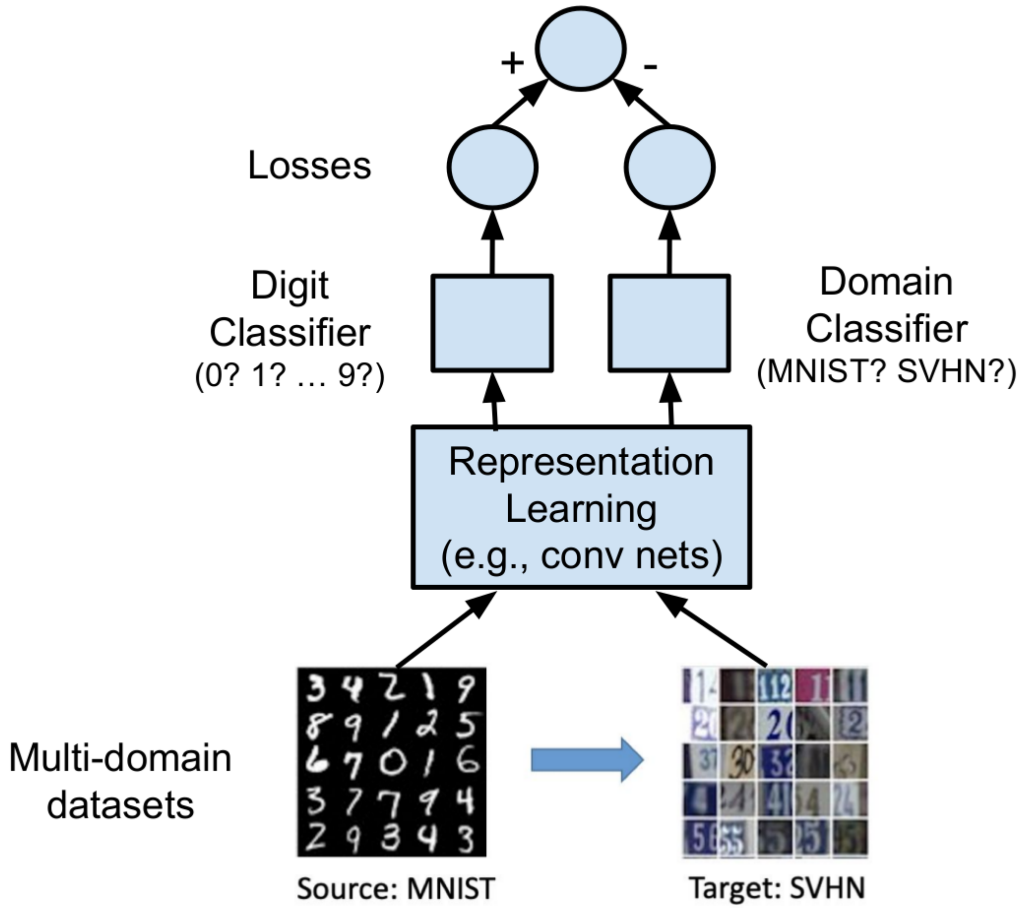
Modality transfer [Wolterink et al.'17]



Anomaly detection [Schlegl et al.'17]



Domain adaptation



- [Ganin et al.'15]

More examples of adversarial ML

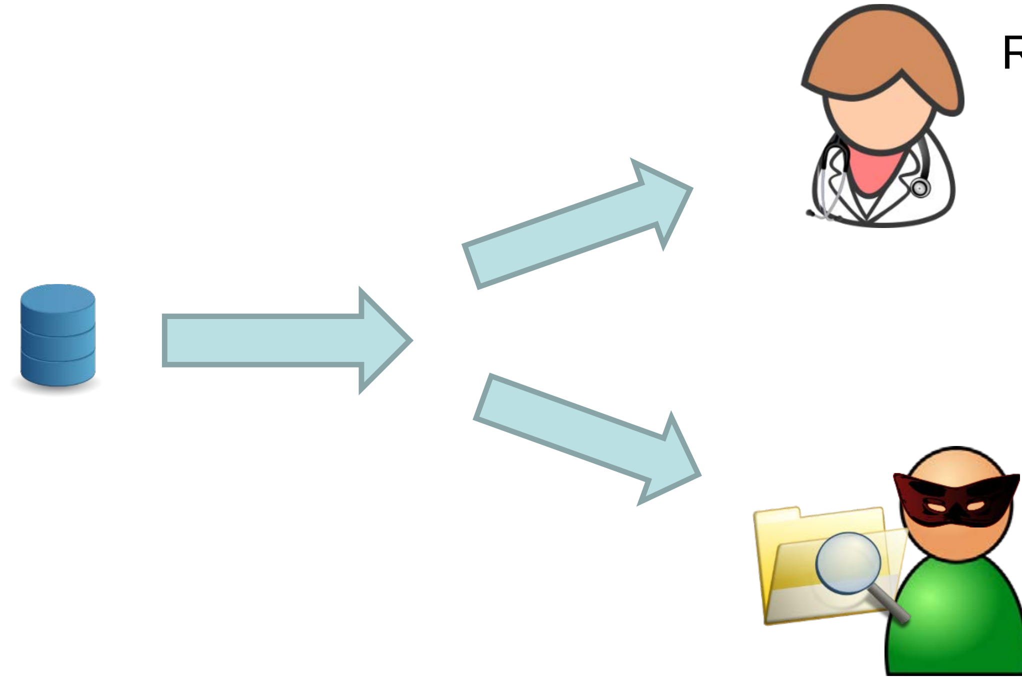
- GAN
- Domain adaptation
- Robust classification: narrow-sense adversarial learning
- Privacy-preservation / fair learning
- Attack on Deep NN
- Training data poisoning
- Hyperparameter learning
- Meta-learning
- …

Part 2/4. Privacy preservation

- Based on
  - J. Hamm, "*Minimax Filter: Learning to Preserve Privacy from Inference Attacks*," JMLR, 2017
  - J. Hamm, "*Preserving Privacy of Continuous High-dimensional Data with Minimax Filters*," AISTATS, 2015
- Also related to
  - J. Hamm, P. Cao, and M. Belkin, "*Learning Privately from Multiparty Data,*" ICML, 2016

Scenario

Subjects



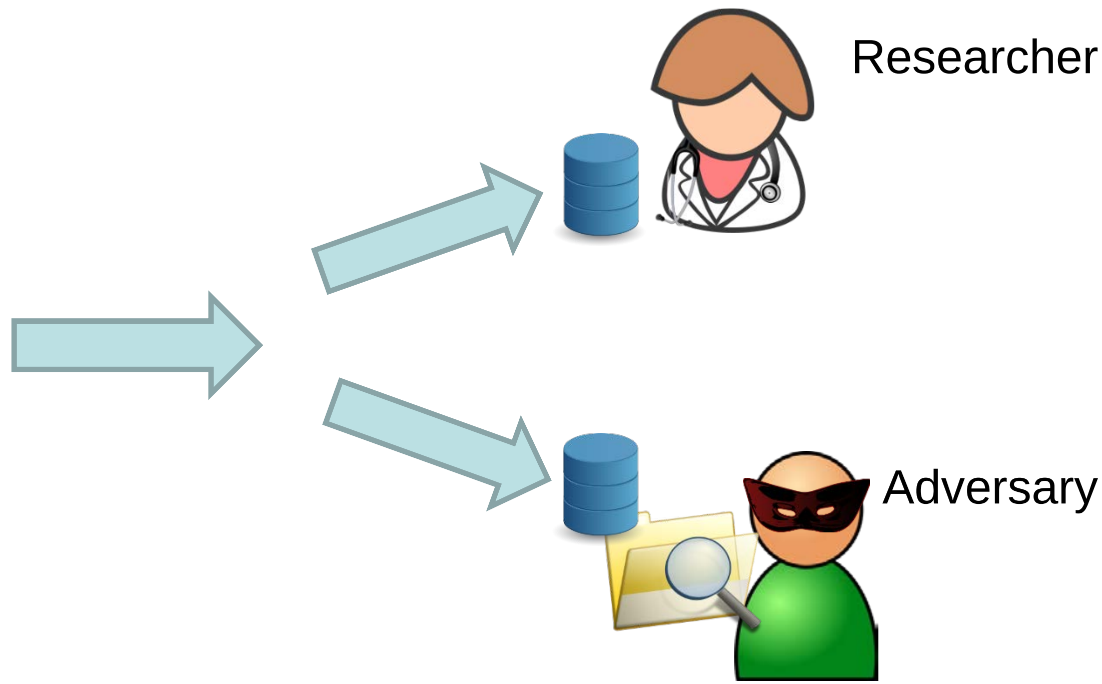
Researcher

Adversary

x : MRI data
y : identity
z : disease

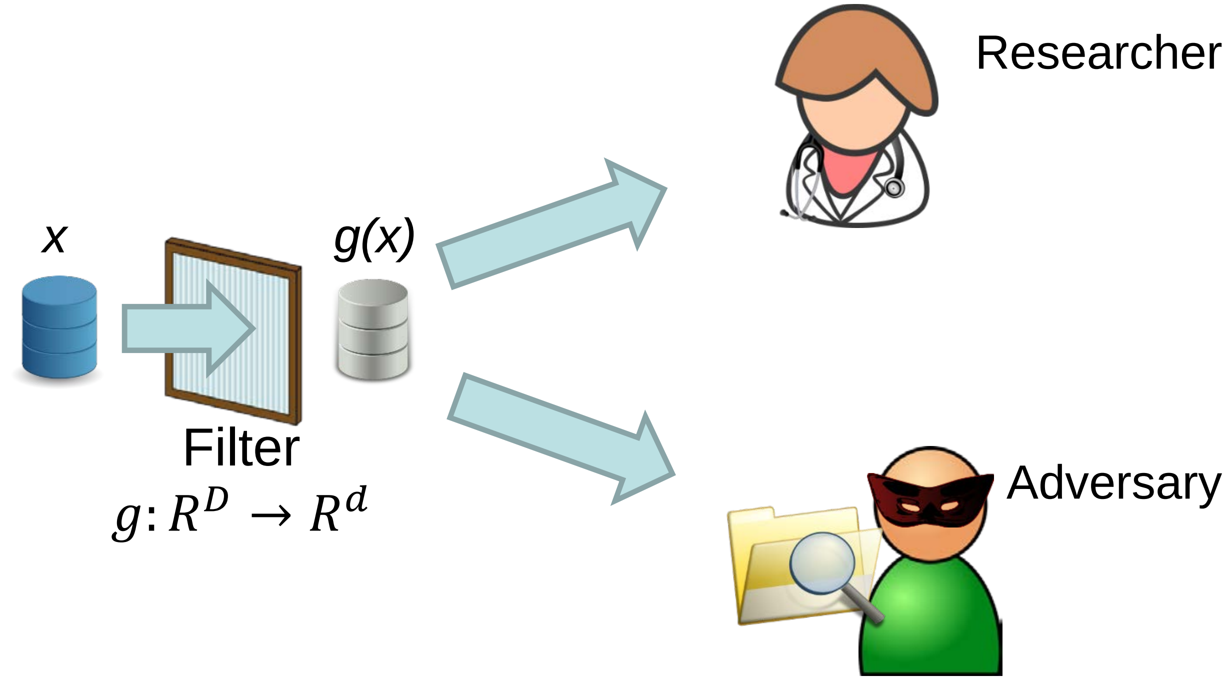
Scenario

Subjects



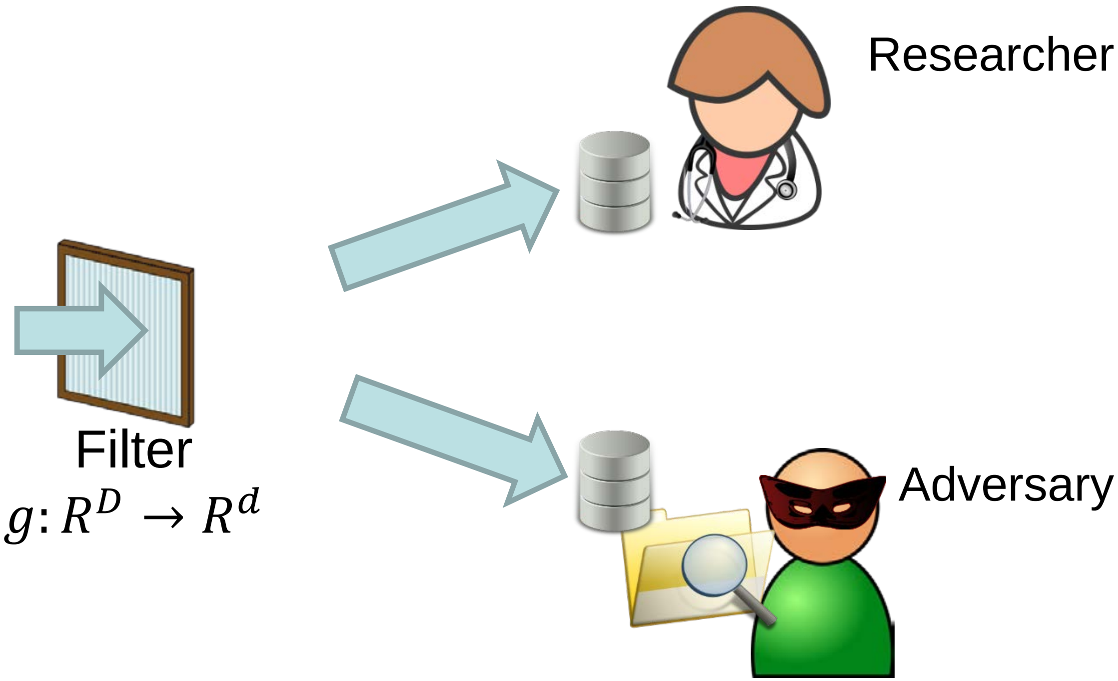
x : MRI data
y : identity
z : disease

Filter



x : MRI data
y : identity
z : disease

Filter

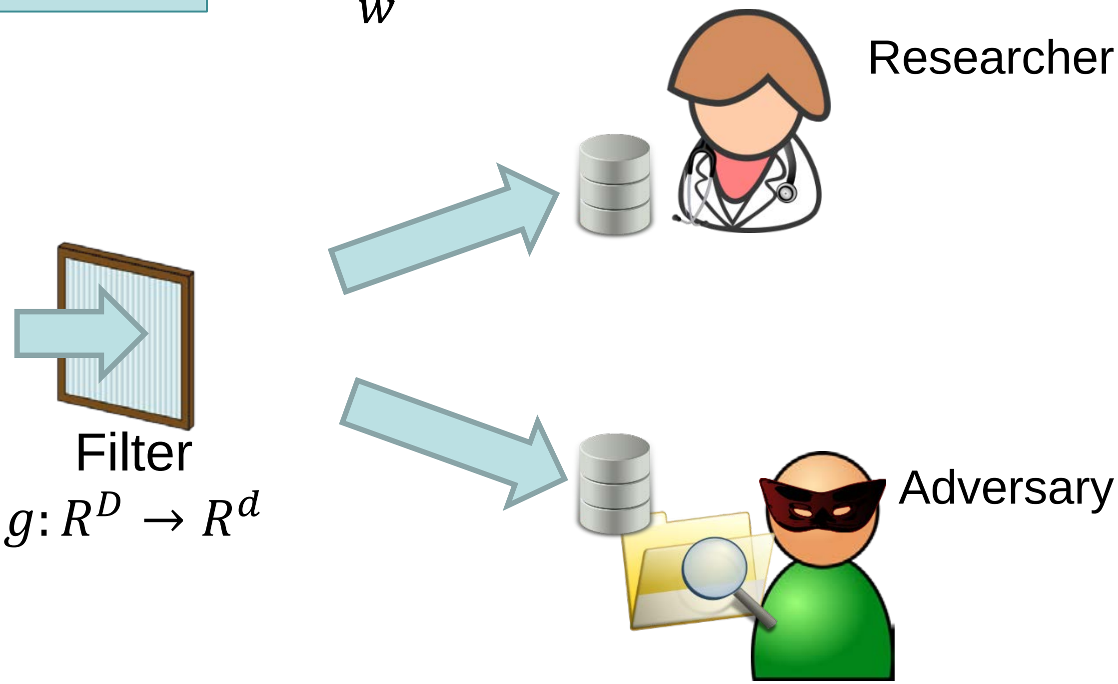


x : MRI data
y : identity
z : disease

Game formulation

Researcher: min utility risk

$$\min_{w} f_{\text{util}}(w) = E[l(g(x; w), z)]$$

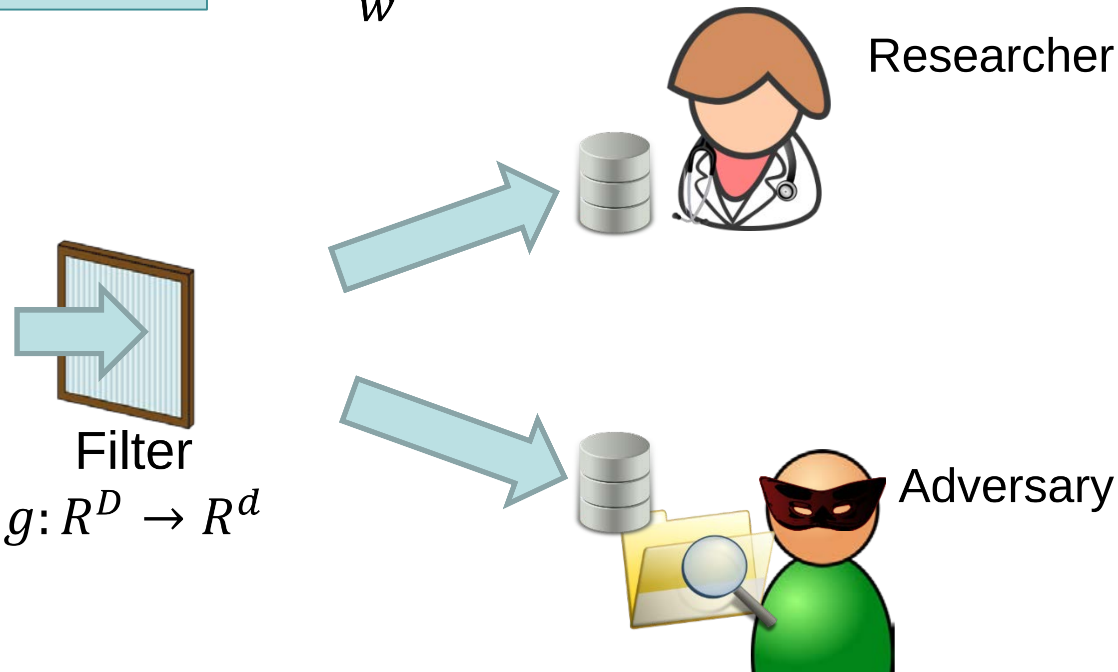


x : MRI data
y : identity
z : disease

Game formulation

Researcher: min utility risk

$$\min_{w} f_{\text{util}}(w) = E[l(g(x;w),z)]$$



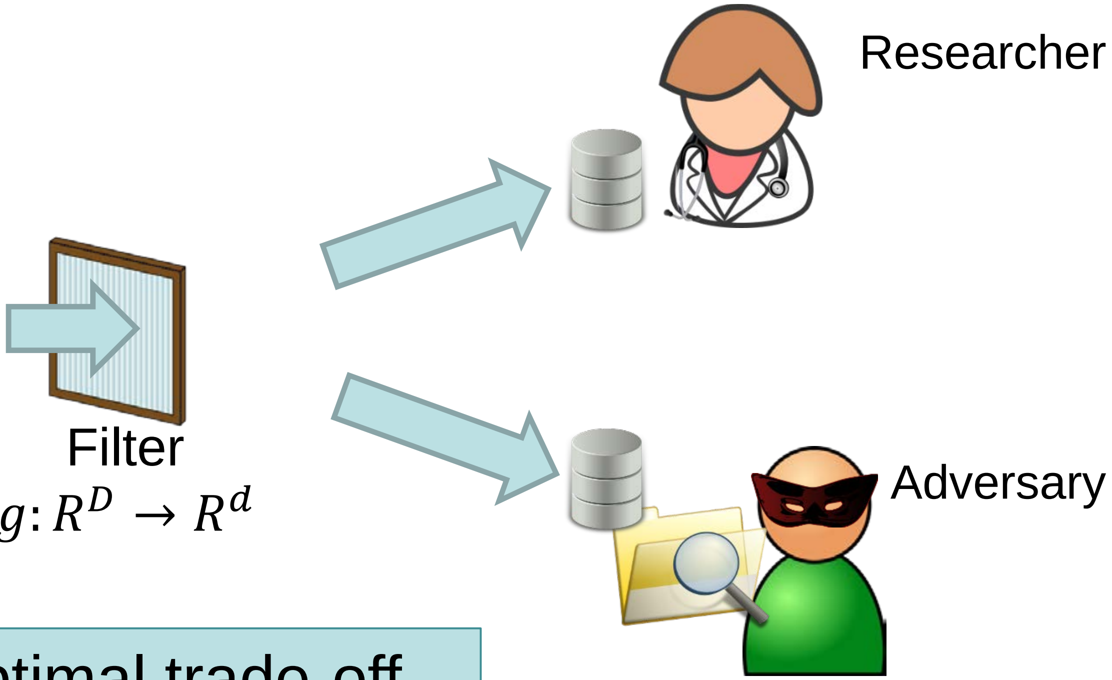
$g: R^D \rightarrow R^d$

x : MRI data
y : identity
z : disease

Adversary: min privacy risk

$$\min_{v} f_{\text{priv}}(v) = E[l(g(x;v),y)]$$

Game formulation



x : MRI data
y : identity
z : disease

Filter: find optimal trade-off

$\min_{g}$ [min privacy risk + min utility risk]

Optimal trade-off

- Solve

$$\min_g \text{[min privacy risk + min utility risk]}$$

Optimal trade-off

- Solve

$$\min_g \text{[min privacy risk + min utility risk]}$$

$$= \min_g \left[-\min_v f_{priv}(g, v) + \rho \min_w f_{util}(g, w) \right]$$

Optimal trade-off

- Solve

$$\min_{g} [\text{ min privacy risk} + \text{ min utility risk }]$$

$$= \min_{g} \left[-\min_{v} f_{priv}(g, v) + \rho \min_{w} f_{util}(g, w) \right]$$

$$= \quad \ldots$$

$$= \min_{g,w} [\max_{v} f_{comb}(g, v, w)]$$

Optimal trade-off

- Solve

$$\min_g \text{[min privacy risk + min utility risk]}$$

$$= \min_g \left[-\min_v f_{priv}(g,v) + \rho \min_w f_{util}(g,w) \right]$$

$$= \quad \ldots$$

$$= \min_{g,w} [\max_v f_{comb}(g,v,w)]$$

- Solution is called **Minimax filter**, which is optimal by design

Optimization

- Continuous minimax optimization isn't too easy
- Classic first-order method [Kiwiel'87]

Repeat:

Linearize f: $f^l(q,v) = f(u,v) + \langle \nabla_u f(u,v),\ q - u \rangle$

Solve

$$\min_q \ [\max_v f^l(q,v) + \frac{1}{2}\|q\|^2]$$

Update

$$u \leftarrow u + \eta\, q$$

- An analog of steepest descent for minimax
- Converges under mild assumptions
- Uses line-search

Experiments

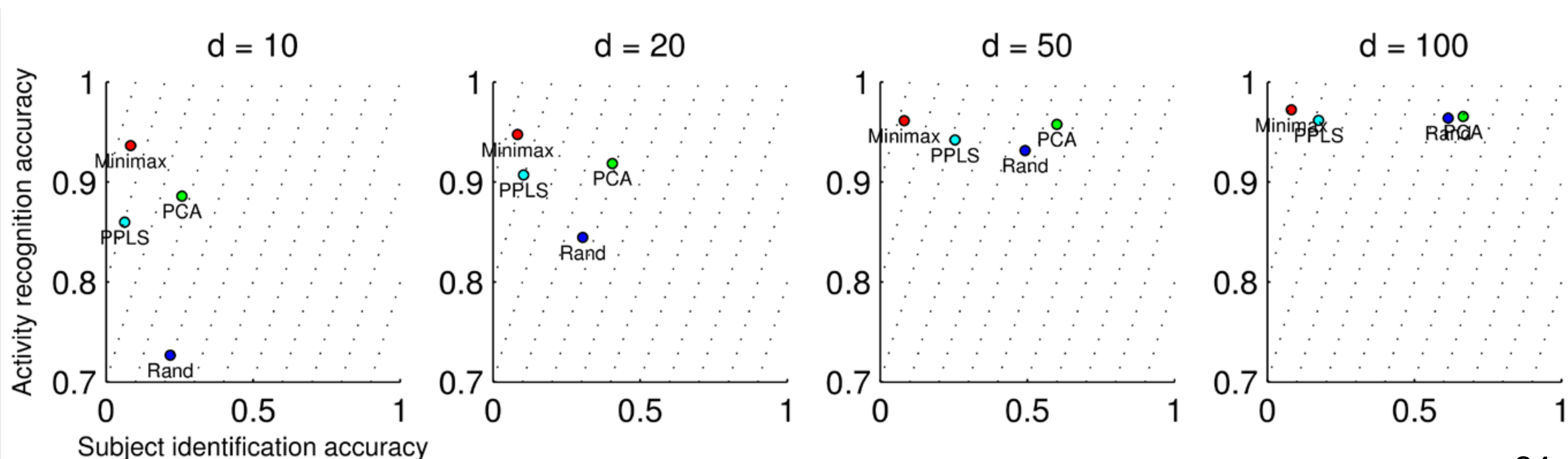
- Filters compared
  - Minimax
    - Minimax 1 (linear),   Minimax 2 (2-layer NN)
  - Non-private
    - Random proj, PCA
  - Private
    - Private Partial Least Squares (Enev, 2012)
    - Discriminately Decreasing Discriminability (Whitehill, 2012)

- Loss/classifier: multiclass logistic regression

Experiment 1

- Activity recognition from motion data (UCI HAR dataset)
  (x=time-freq features, N=10299, S=30, 6 activities)

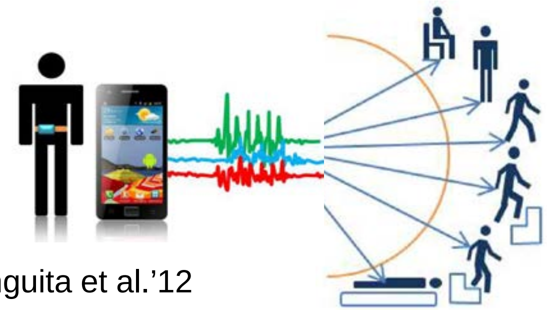
Anguita et al.'12



Experiment 1

- Activity recognition from motion data (UCI HAR dataset)

  (x=time-freq features, N=10299, S=30, 6 activities)

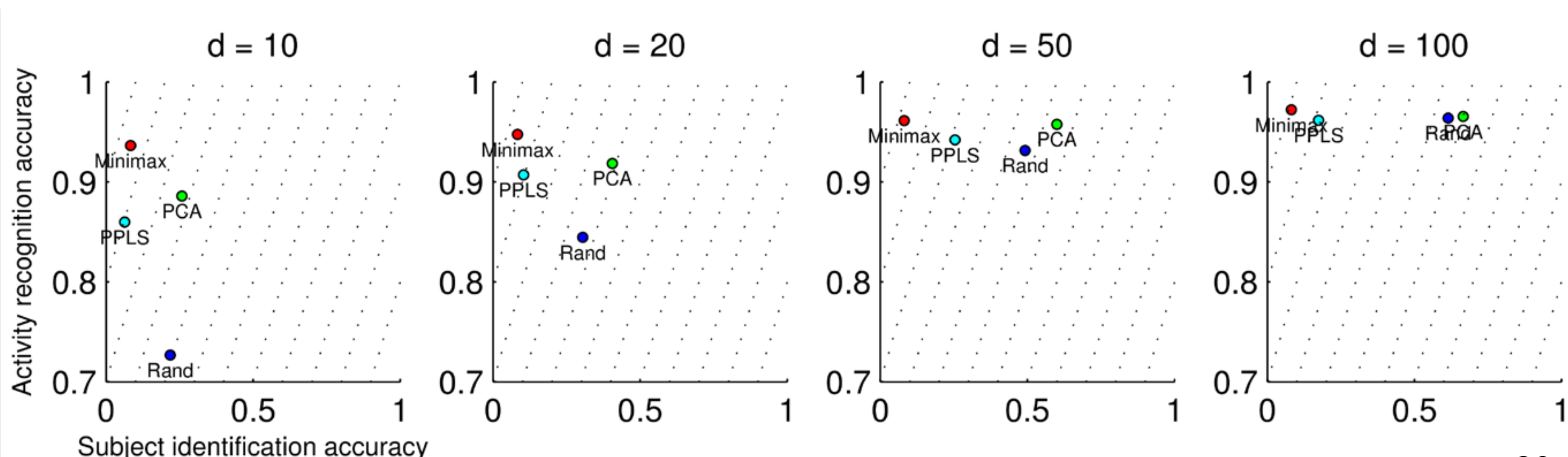


Anguita et al.'12

Experiment 1

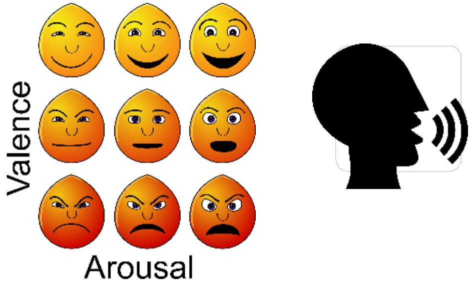
- Activity recognition from motion data (UCI HAR dataset)
  (x=time-freq features, N=10299, S=30, 6 activities)

Anguita et al.'12

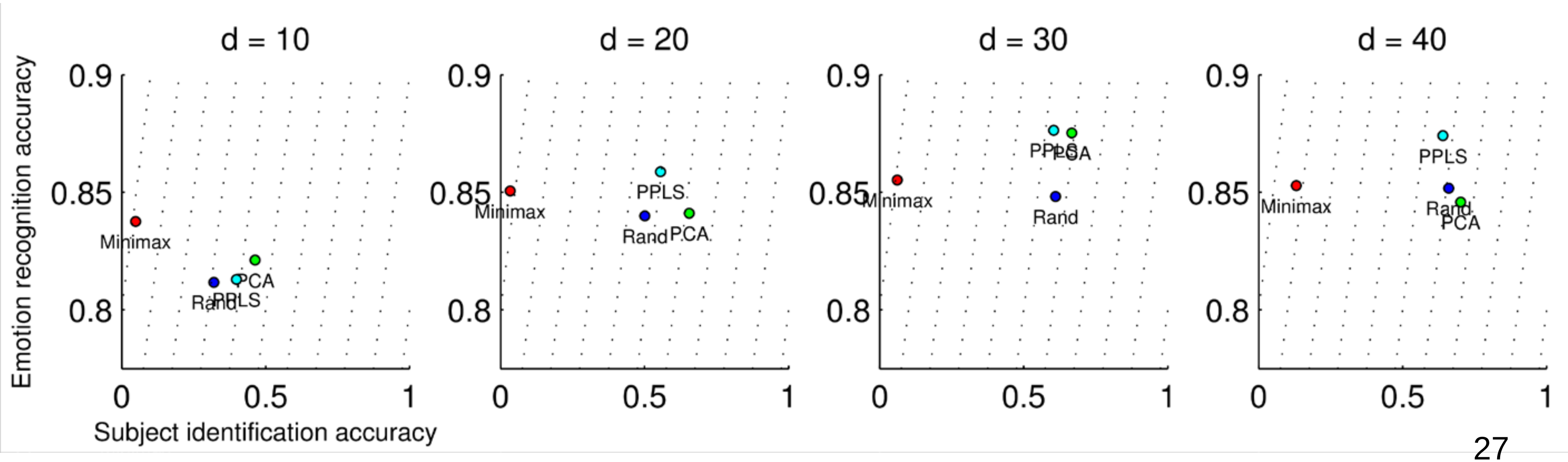


Experiment 2

- Emotion recognition from speech (Enterface dataset)
  (x=MFCC, N=427, S=43, {happy,non-happy}



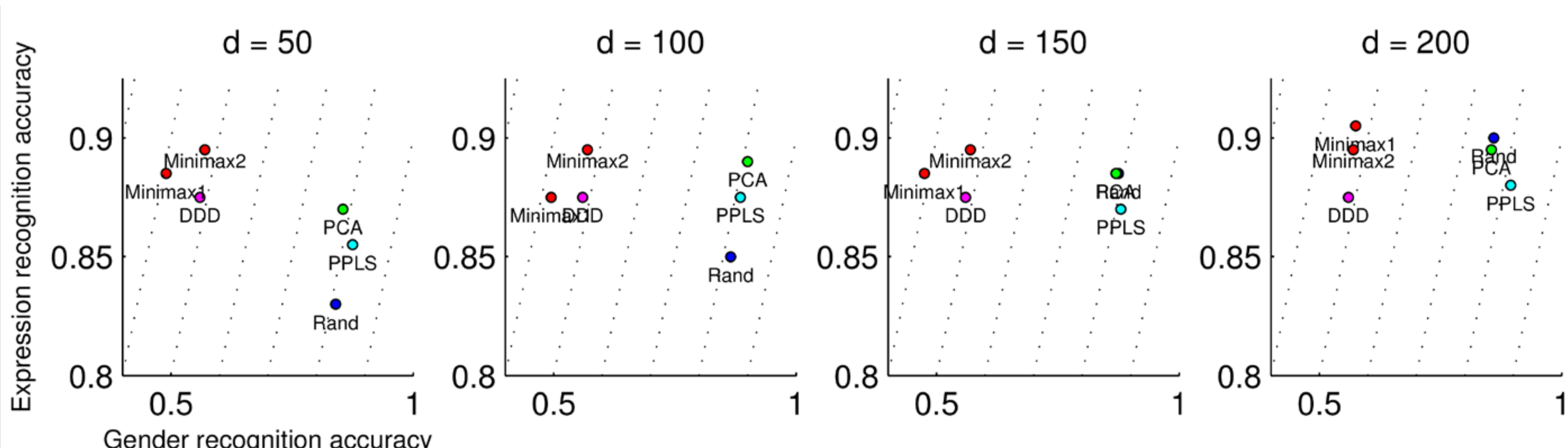
Martin et al.'06



Experiment 3

- Gender/expression recognition from face (Genki dataset)
  (x=raw image, N=1740, {male,female}, {smile, non-smile}}

Whitehill et al.'12



Generalization bound

- Q. Does it generalize well?

Generalization bound

- Q. Does it generalize well?

- A. Theorem.

$$E_{S\sim D^m}\left[\left|E_D[l_J(u^*, v^*, w^*)] - E_D[l_J(\hat{u}, \hat{v}, \hat{w})]\right|\right]$$
$$\leq 4E_{S\sim D^m}\left[\mathfrak{R}(l_p \circ H_p \circ G \circ S) + \rho\, \mathfrak{R}(l_u \circ H_u \circ G \circ S)\right]$$

where

- $l_J(g, v, w) = -l_p(g, v) + \rho\, l_u(g, w)$: joint privacy-utility loss
- $(\hat{g},\ \hat{v},\ \hat{w}\ )$: empirical minimax solution
- (g^*, v^*, w^*) : expected minimax solution

Part 3/4. Optimization

- Based on
  - J. Hamm and Y.K. Noh, "*K-Beam Minimax: Efficient Optimization for Deep Adversarial Learning,*" ICML, 2018

Minimax optimization

- Minimax filter [Hamm'15]:

$$\min_g \left[-\min_v f_{priv}(g,v) + \rho \min_w f_{util}(g,w)\right]$$

- GAN [Goodfellow et al.'14]:

$$\min_u \max_v \left[E[\log D(x;v)] + E[\log(1 - D(G(z;u);v))]\right]$$

- Domain adaptation [Ganin et al.'15]:

$$\min_{u=(u',w)} \max_v \left[-E[D_1(G(x;u');v)] + \lambda E[D_2(G(x;u');w)]\right]$$

Minimax optimization

- General form: $\min_{u \in U} \max_{v \in V} f(u, v)$
  - Zero-sum leader-follower games
  - u: leader/min player, v: follower/max player
  - $f(u, v)$: payoff of follower/max player

Minimax optimization

inner maximization

- General form: $\min_{u \in U} \boxed{\max_{v \in V} f(u, v)}$

  - Zero-sum leader-follower games
  - u: leader/min player, v: follower/max player
  - $f(u, v)$: payoff of follower/max player

Minimax optimization

inner maximization

- General form: $$\min_{u \in U} \max_{v \in V} f(u, v)$$

outer minimization

  - Zero-sum leader-follower games
  - u: leader/min player, v: follower/max player
  - $f(u, v)$: payoff of follower/max player

GAN as minimax problem

- GAN: $\min_u \max_v f(u, v)$, where

$$f(u,v) = E_{P(x)}[\log D(x; v)] + E_{P_Z(z)}[\log(1 - D(G(z; u); v))]$$

- Analytical solution to inner problem $\arg\max_v V(u, v)$ is [Goodfellow'14]

$$D^*(x) = \frac{P(x)}{P(x)+P_Z(G(z))}$$, and consequently

$$f^*(u) := \max_v f(u,v) = 2\ \mathrm{JSD}(P(x)||P_Z(G(z))) + const$$

- GAN: $\min_u f^*(u) = \min_u 2\ \mathrm{JSD} + const$

⇒ GAN is a Jensen-Shannon Divergence minimization problem

- Assumptions: inner maximization is feasible and numerically solvable

f-GAN

- GAN: JS-divergence minimization
- *f*-divergence $F(P||Q) = E_Q[f(P/Q)]$
  - Generalization of JS-divergence
- $F(P||Q) \geq \max_v \left[E_{x\sim P}\, T(x;v) - E_{x\sim Q(u)}\, f^*(T(x;v))\right]$, where
  T is variational function, f is generator function
  $f^*(t) = \sup_u (ut - f(u))$ is Fenchel dual of f [Nowozin'16]

  ⇒ *f*-GAN minimizes (lower bound of) *f*-divergence $F(P||Q)$
- GAN: $\max_v \left[E_{P(x)}[\log D(x;v)] + E_{P_z(z)}[\log(1 - D(G(z;u);v))]\right]$

- Assumptions: inner maximization is feasible and numerically solvable

In reality

- Assumptions: inner maximization is feasible and numerically solvable
- Training procedure is quite different from theory
- Ideally:

  1) Solve $v = \underset{v}{\operatorname{argmax}} f(u, v)$ accurately

  2) Solve $u = \operatorname{argmin}_u f(u, v)$ approximately, and repeat 1) - 2)
- What people really do:

  1) Solve $v = \operatorname{argmax}_v f(u, v)$ approximately

  2) Solve $u = \operatorname{argmin}_u f(u, v)$ approximately, and repeat 1) - 2)

Gradient Descent

- Simplest way to solve min (or max) approximately
- Alternating

$$u \leftarrow u - \rho \nabla_u f, \quad v \leftarrow v + \eta \nabla_v f$$

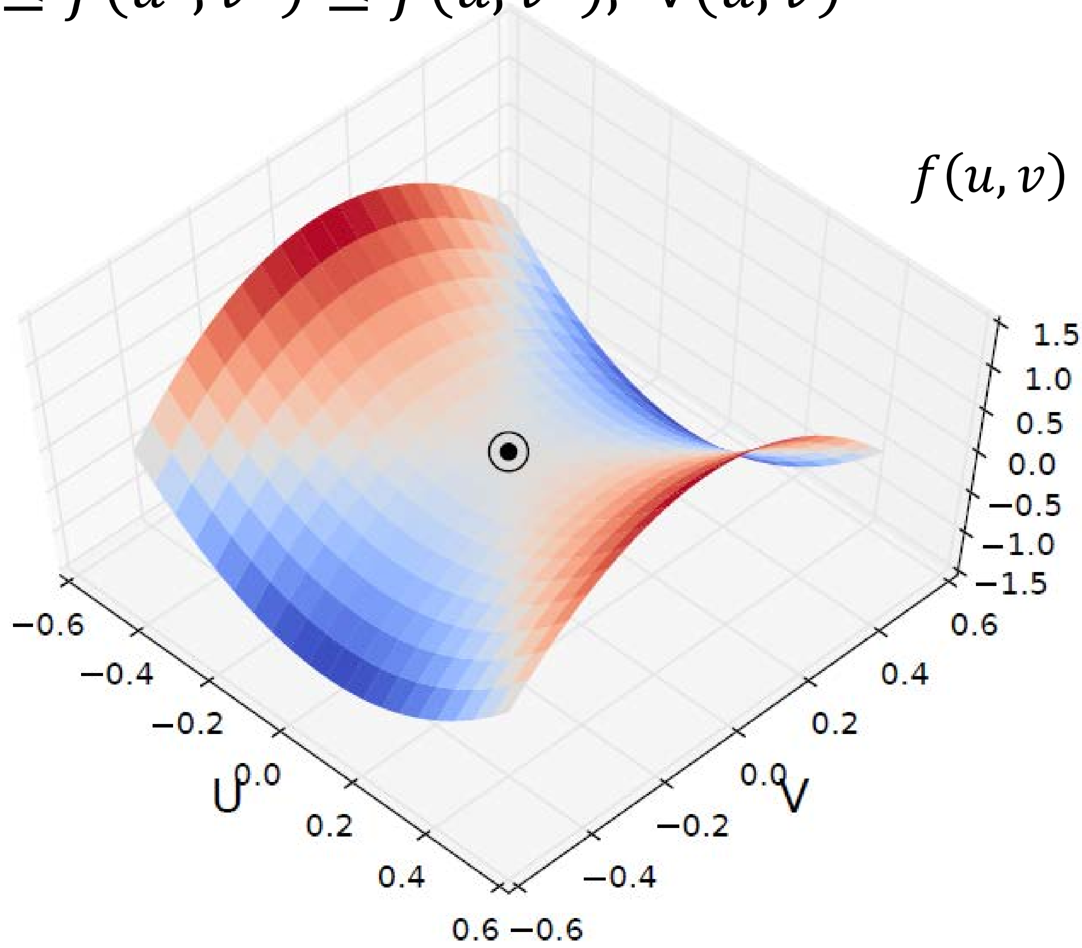
- Simultaneous

$$\begin{pmatrix} u \\ v \end{pmatrix} \leftarrow \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} -\rho I & 0 \\ 0 & \eta I \end{pmatrix} \nabla f$$

- Q. What solutions can alternating/simultaneous gradient descent find?

Saddle point

- Def: (u^*, v^*) is a saddle point if
  $$f(u^*, v) \le f(u^*, v^*) \le f(u, v^*),\ \forall(u, v)$$



$$f(u, v) = u^2 - v^2$$

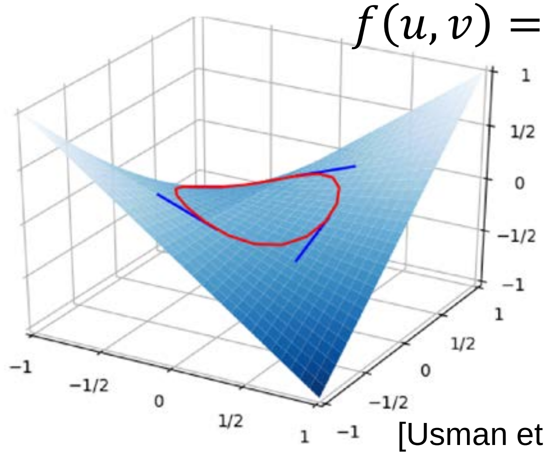
Saddle point theory & algorithm

- Early researchers (50’s – 80’s)
  - Arrow, Hurwicz, Uzawa, Dem’yanov, Evtushenko, …
- Related topics
  - Extragradient, Proximal Gradient, Mirror Descent, Chambolle-Pock, Variational Inequality, …
- Recent work
  - [Hazan et al.,’17], [Daskalakis et al.’18], [Adolphs et al.’18], [Metrikopoulous et al.’18], [Rafique et al.’18], [Lin et al.’18], …

Failure of gradient descent

- Gradient descent does not always converge

$f(u, v) = uv$



[Usman et al.'18]

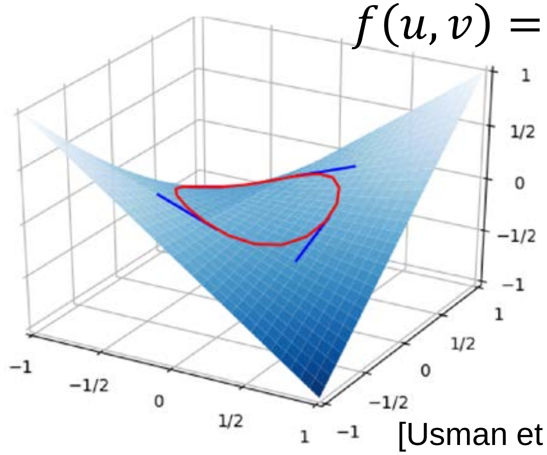
Failure of gradient descent

- Gradient descent does not always converge

→ Modify gradient descent

[Mescheder et al.’17; Nagarajan et al.’17; Roth et al.’17]

$f(u,v) = uv$



[Usman et al.’18]

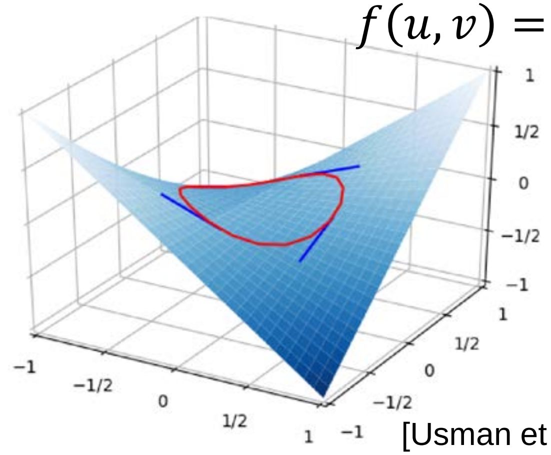
Failure of gradient descent

- Gradient descent does not always converge

→ Modify gradient descent

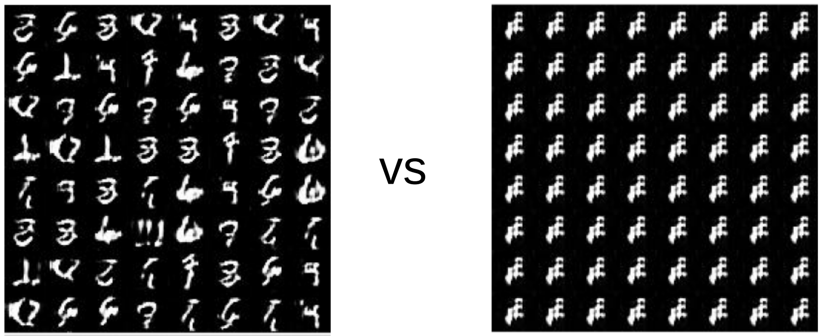
[Mescheder et al.’17; Nagarajan et al.’17; Roth et al.’17]

$f(u, v) = uv$



[Usman et al.’18]

- GAN training fails frequently



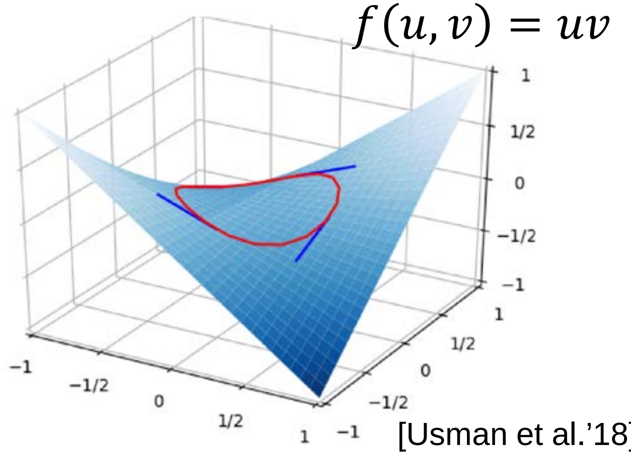
[Mets et al.’17]

Failure of gradient descent

- Gradient descent does not always converge

→ Modify gradient descent

[Mescheder et al.'17; Nagarajan et al.'17; Roth et al.'17]



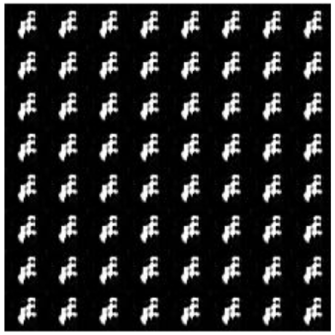
[Usman et al.'18]

- GAN training fails frequently

→ Change the GAN objective

[Uehara et al.'16; Nowozin et al.'16; Arjovsky et al.'17]

vs



[Mets et al.'17]

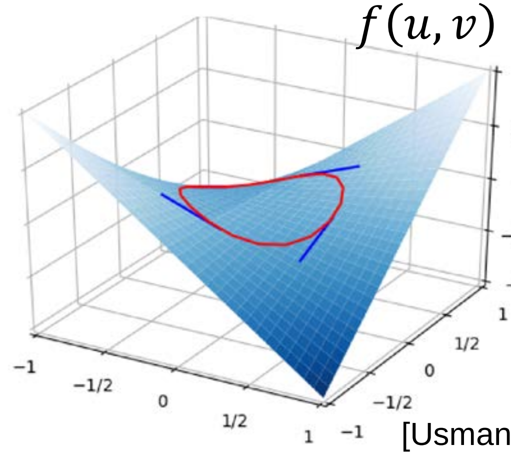
Failure of gradient descent

- Gradient descent does not always converge

→ Modify gradient descent

[Mescheder et al.'17; Nagarajan et al.'17; Roth et al.'17]

$f(u, v) = uv$

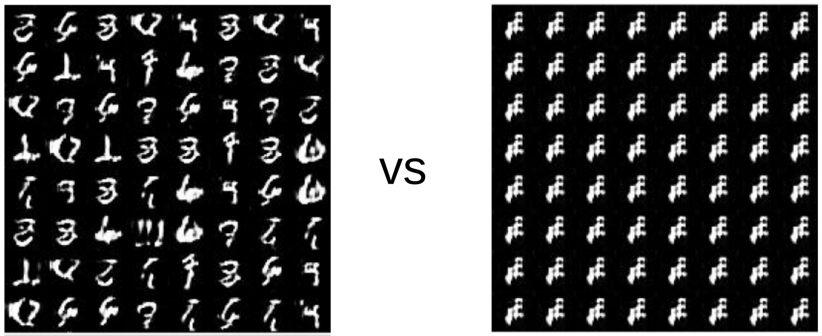


[Usman et al.'18]

- GAN training fails frequently

→ Change the GAN objective

[Uehara et al.'16; Nowozin et al.'16; Arjovsky et al.'17]

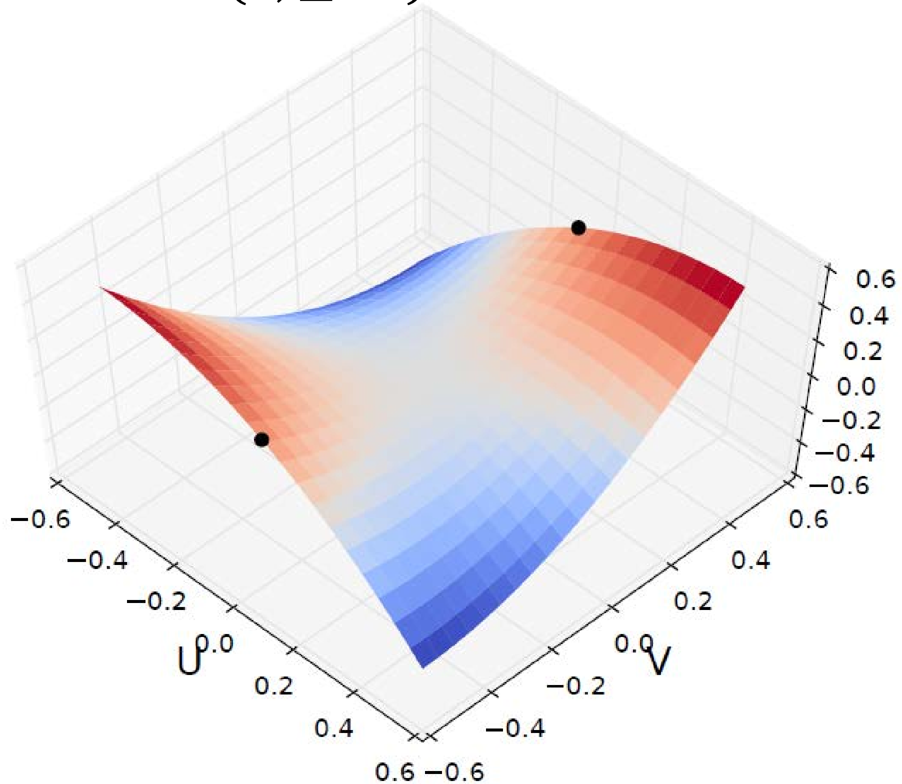


[Mets et al.'17]

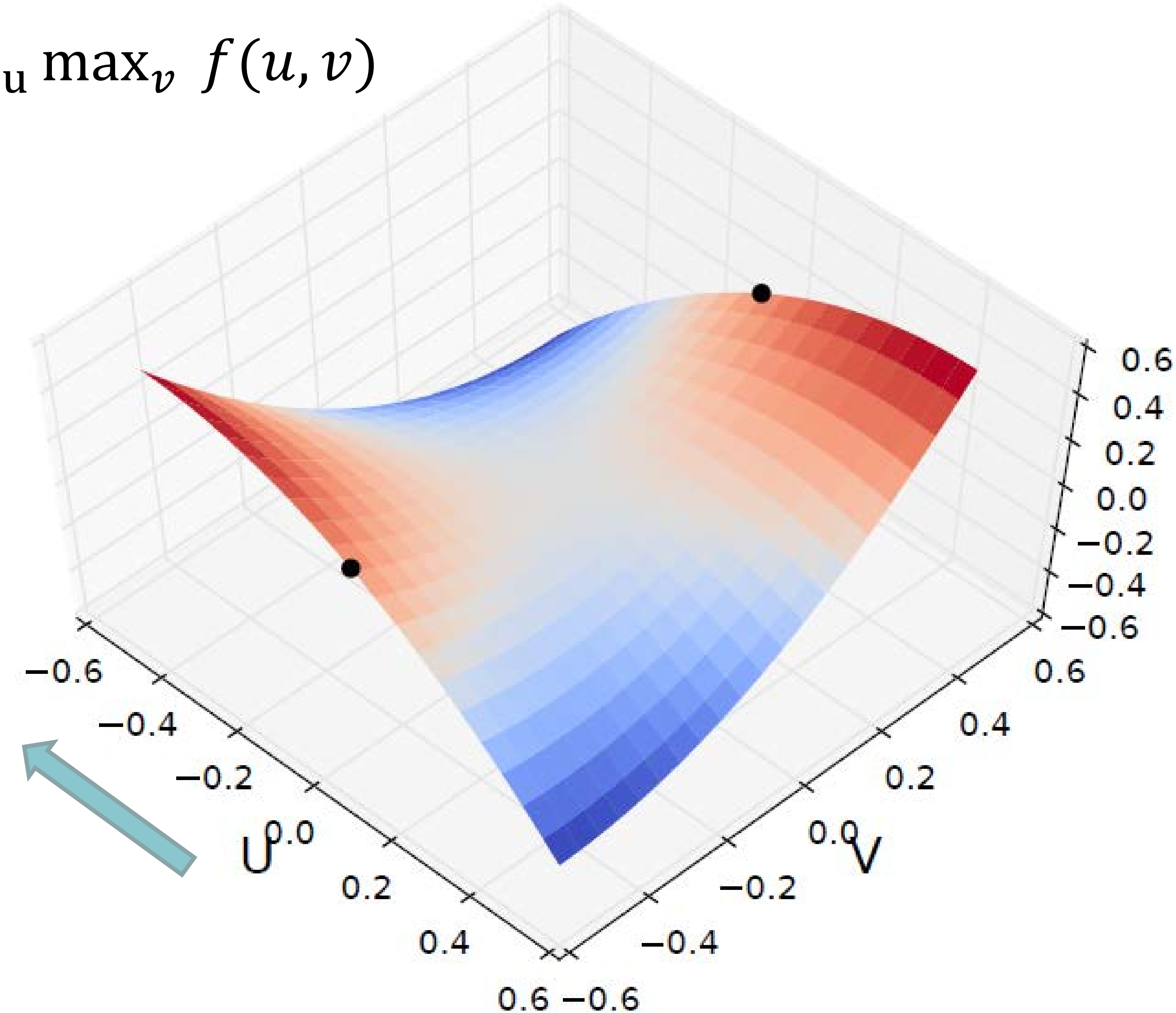
- Q: What if GAN solution is not a saddle point?

Example

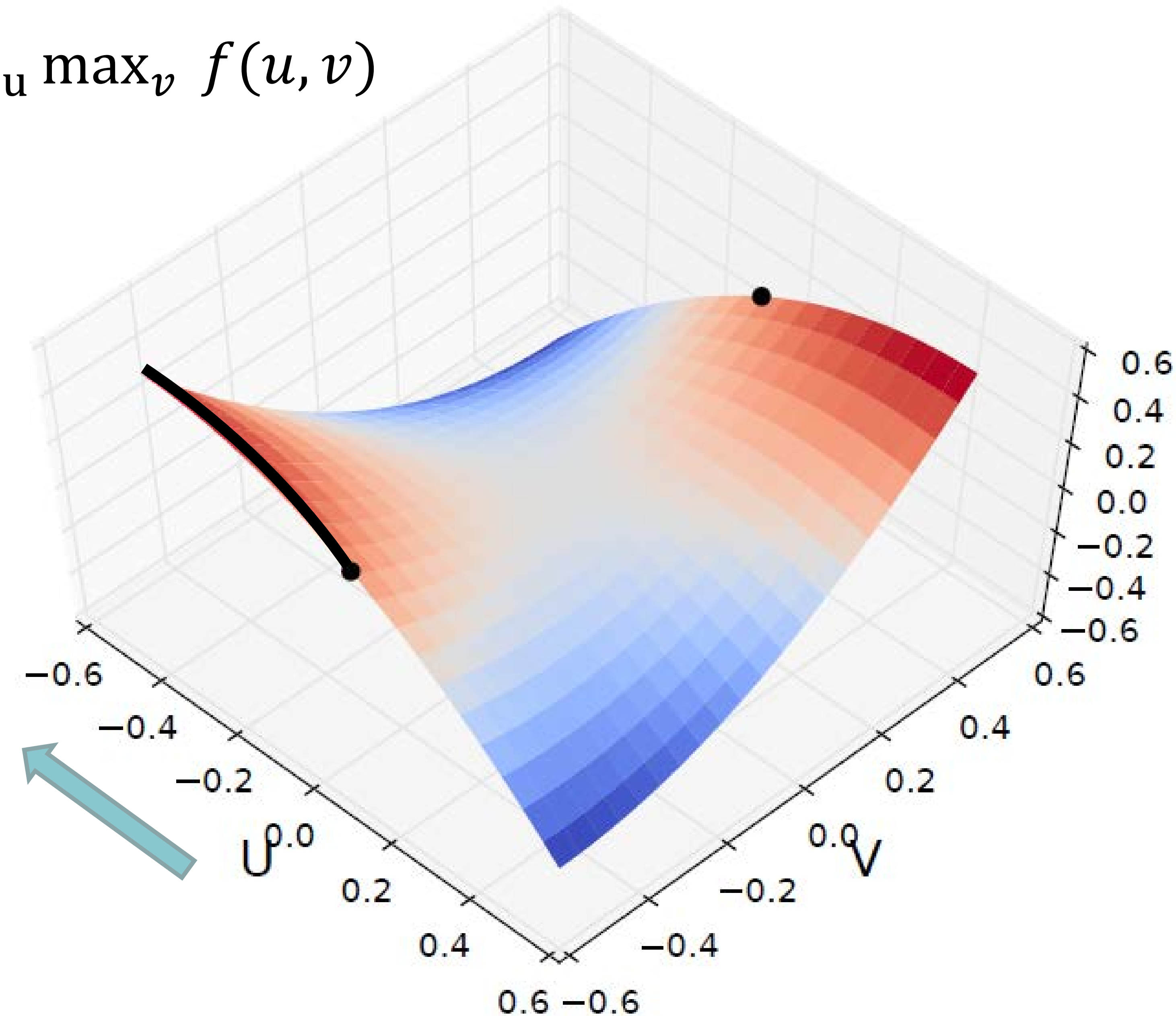
- Anti-saddle $f(u,v) = -u^2 + v^2 + 2uv, \qquad (|u| \leq .5, |v| \leq .5)$
  - Concave-convex
  - No saddle point
  - Minimax solution at $(0, \pm 0.5)$



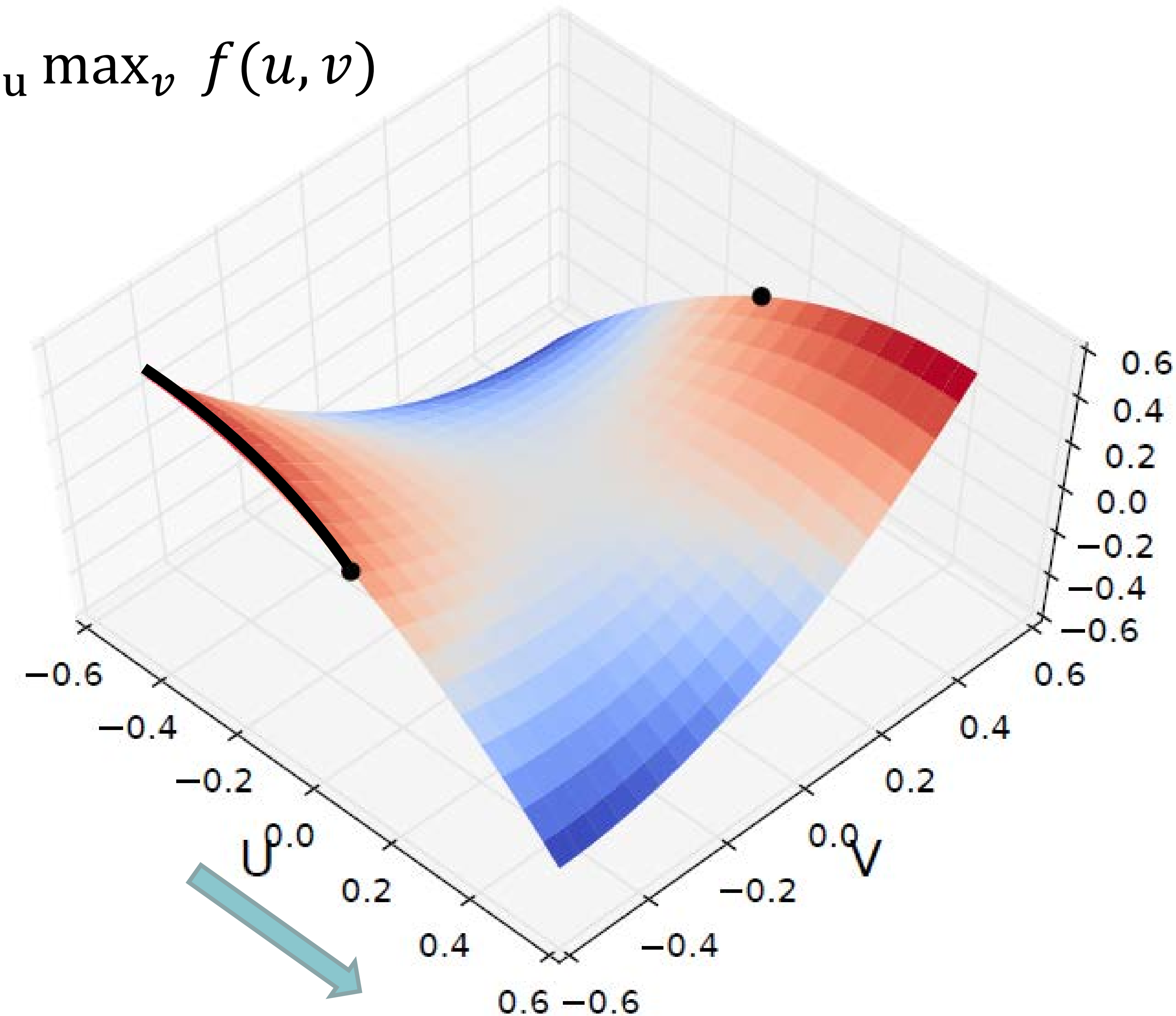
$\min_{\mathrm{u}} \max_{v} \; f(u, v)$



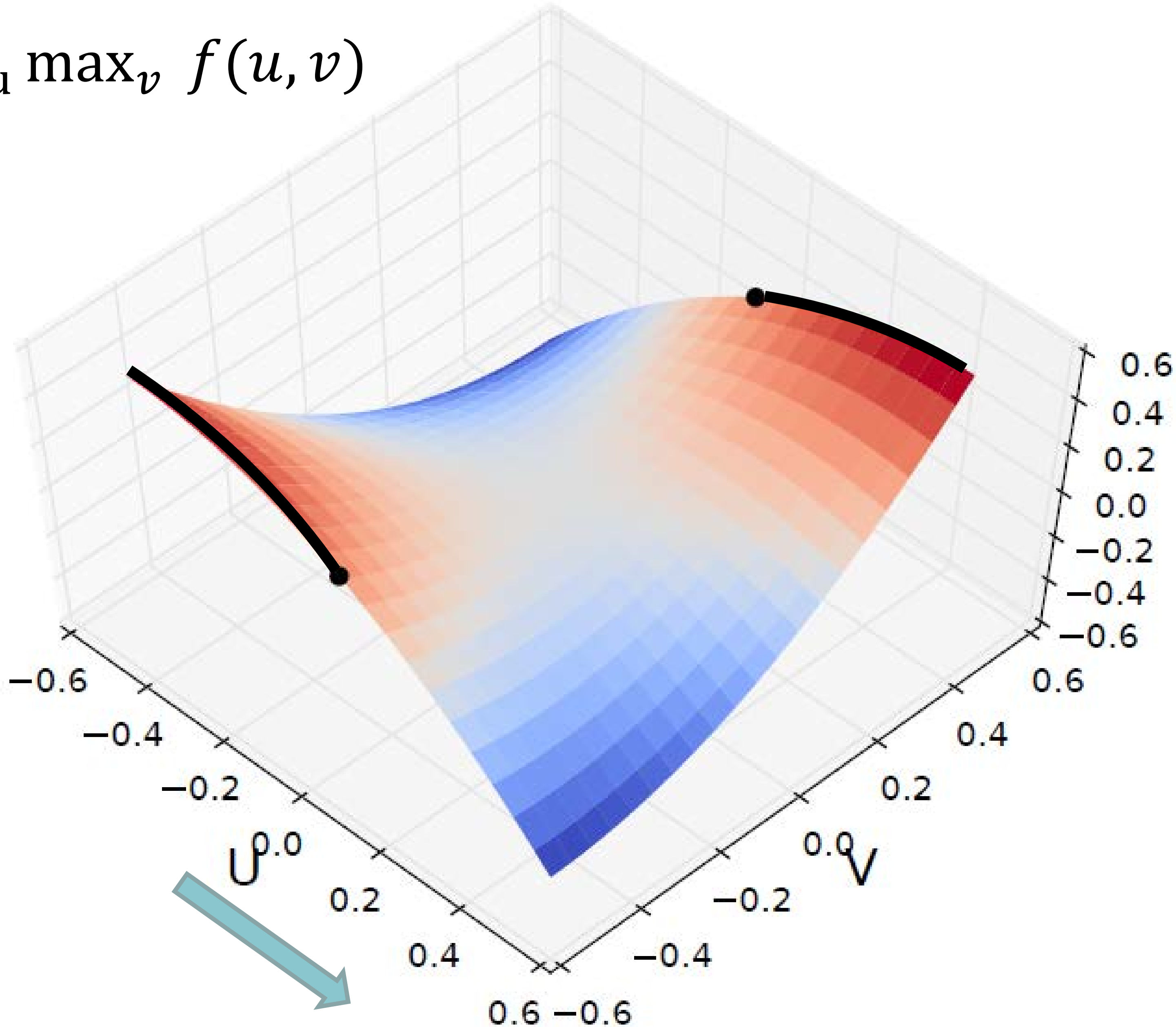
$\min_{u} \max_{v} \ f(u,v)$



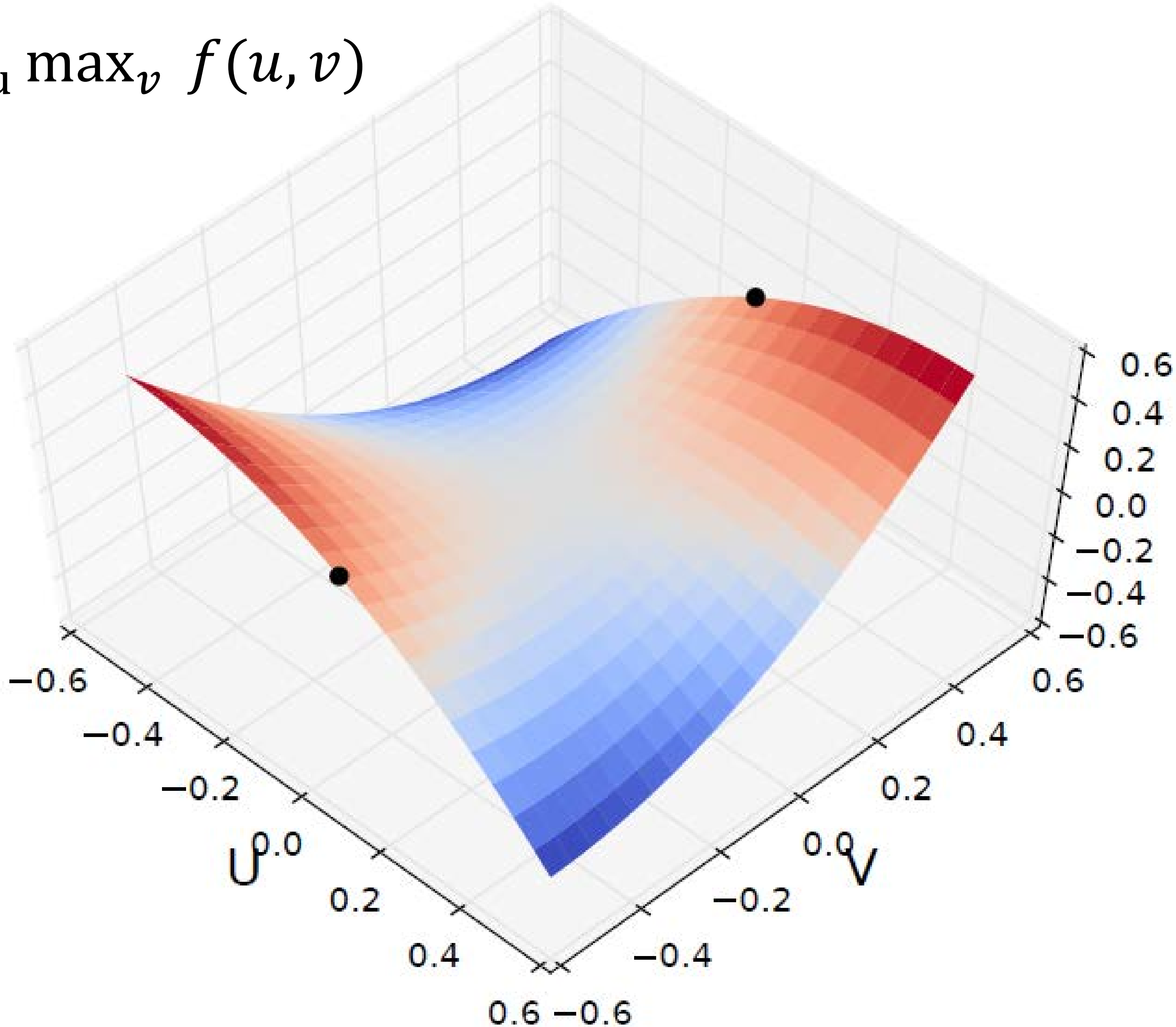
$\min_{\mathrm{u}} \max_{v} \ f(u,v)$



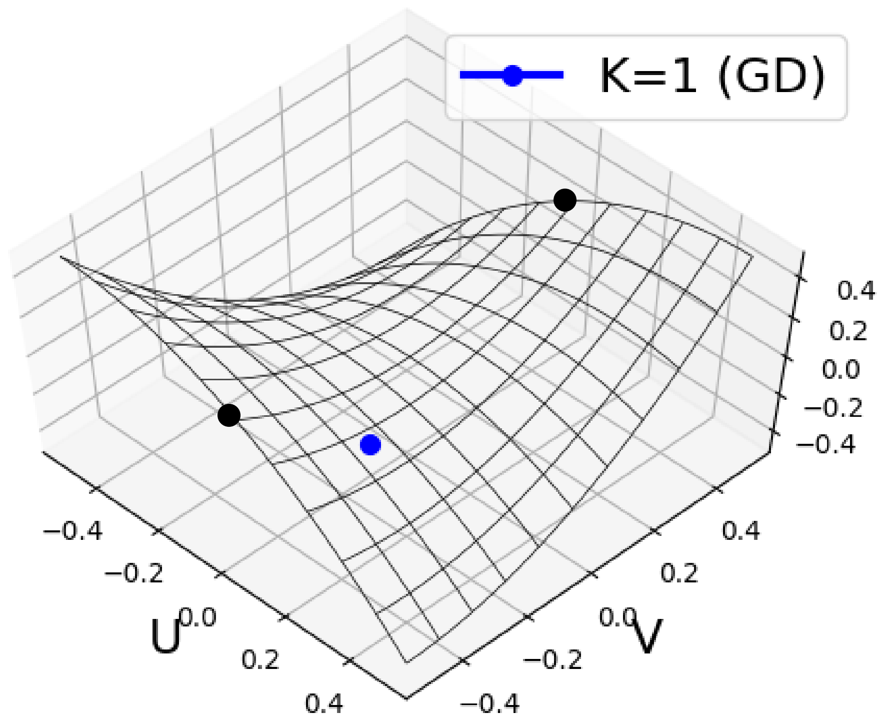
$\min_{u} \max_{v} \ f(u, v)$



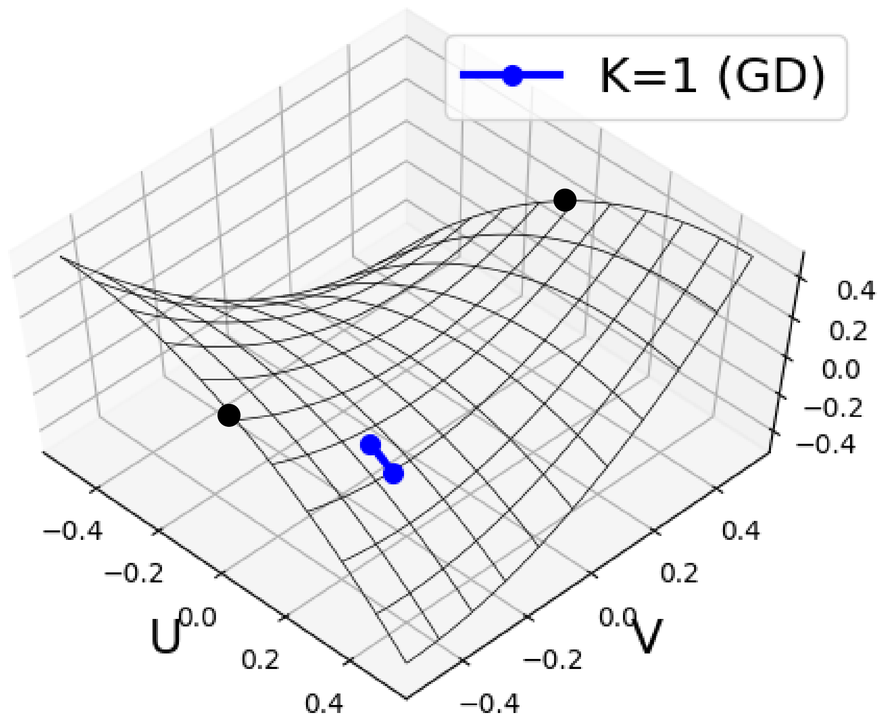
$\min_{u} \max_{v} \ f(u,v)$



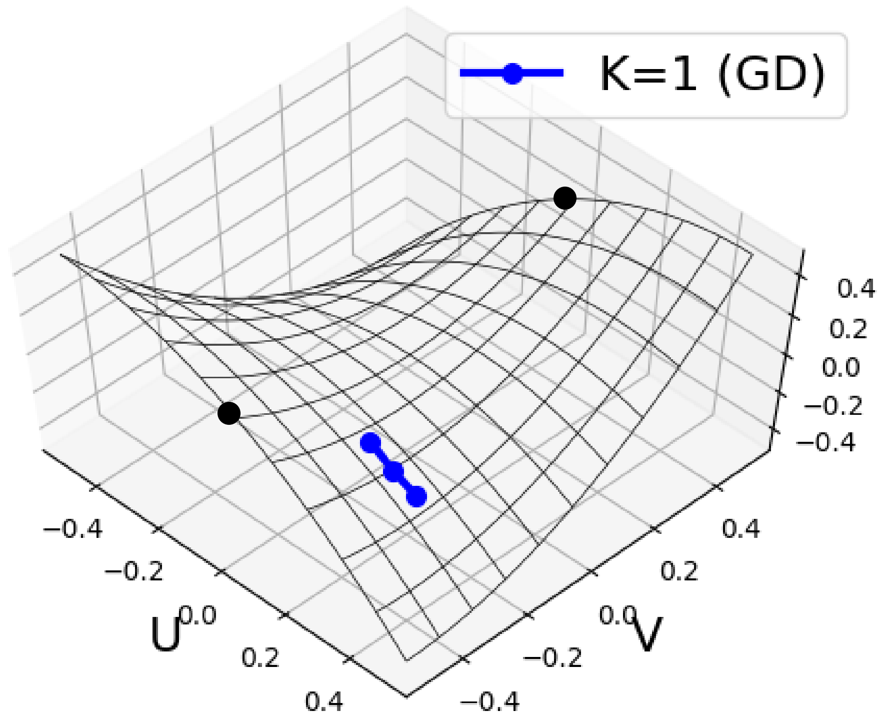
Gradient descent/ascent



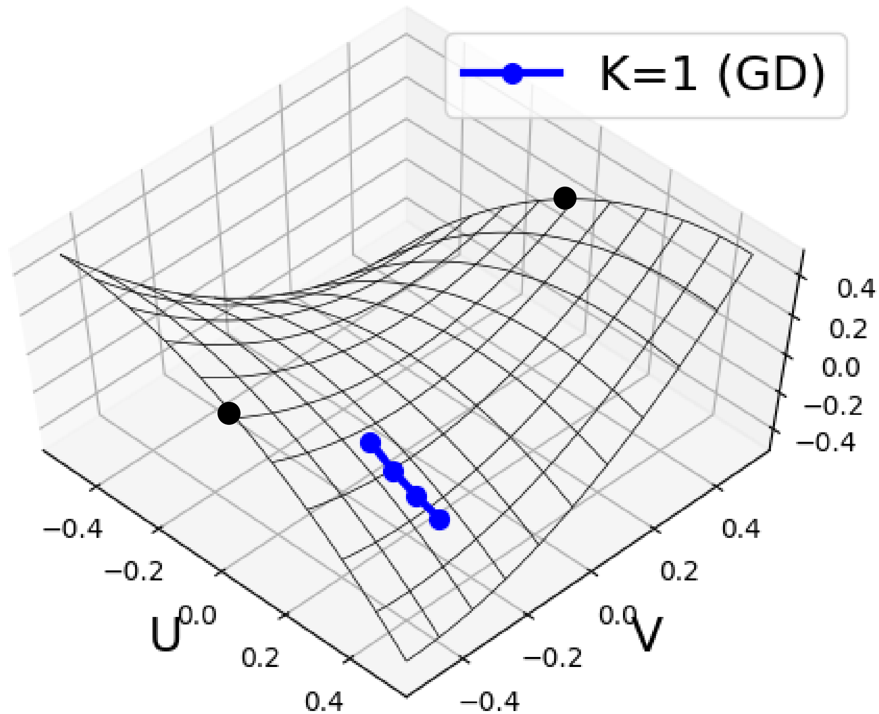
Gradient descent/ascent



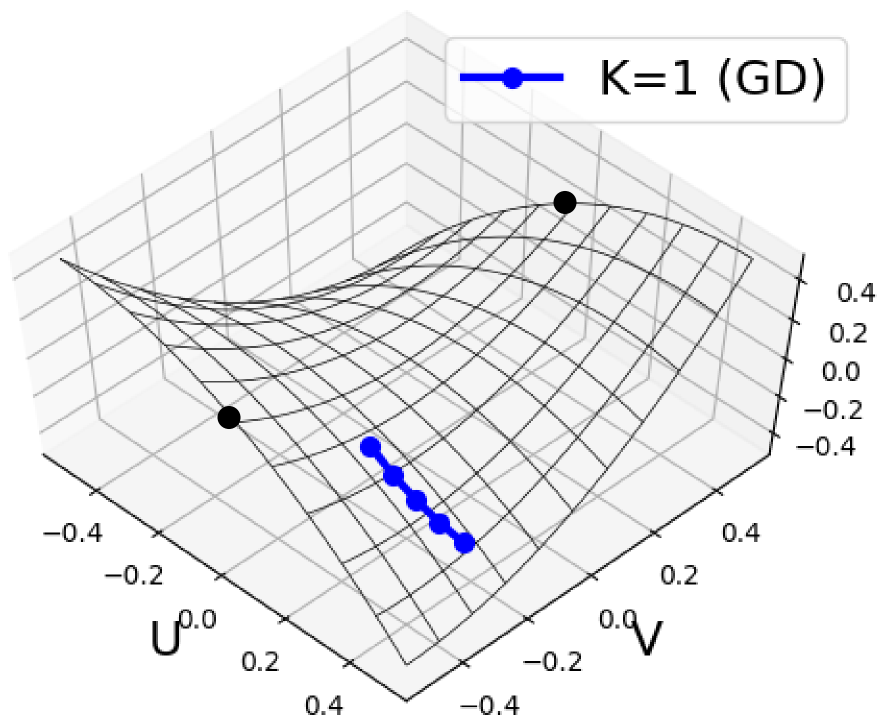
Gradient descent/ascent



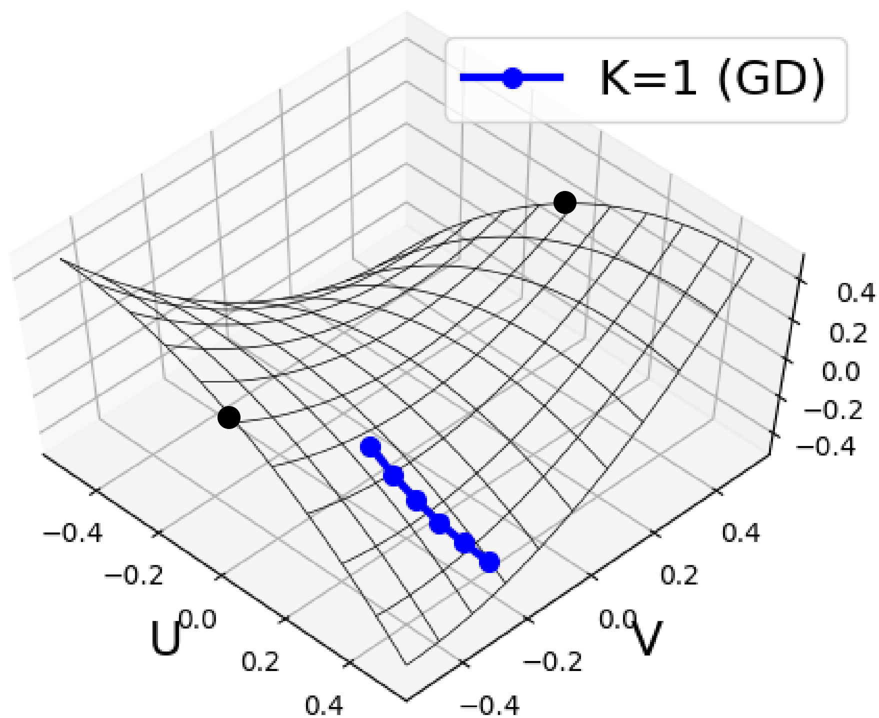
Gradient descent/ascent



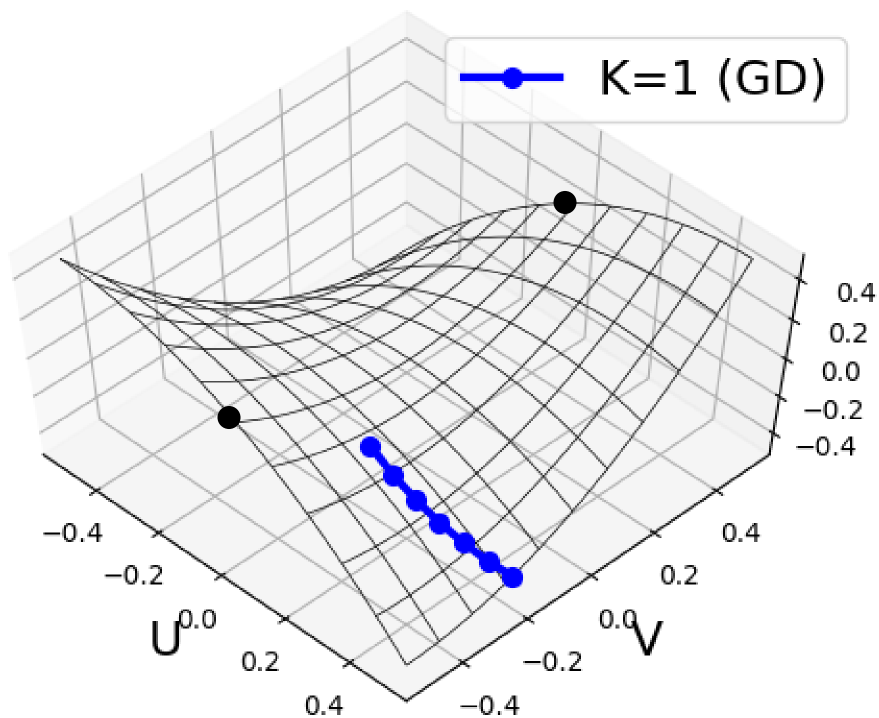
Gradient descent/ascent



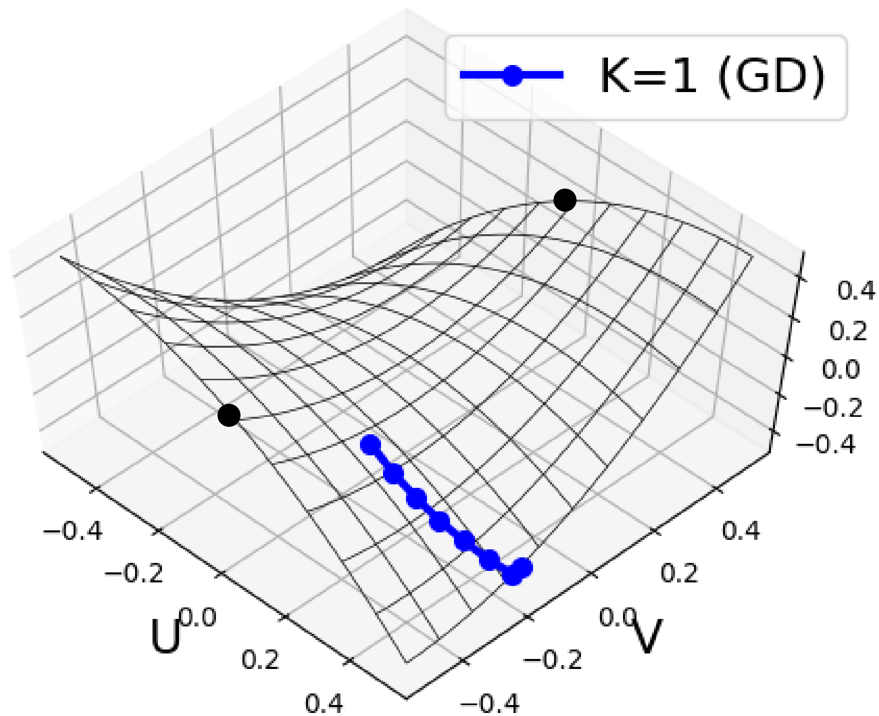
Gradient descent/ascent



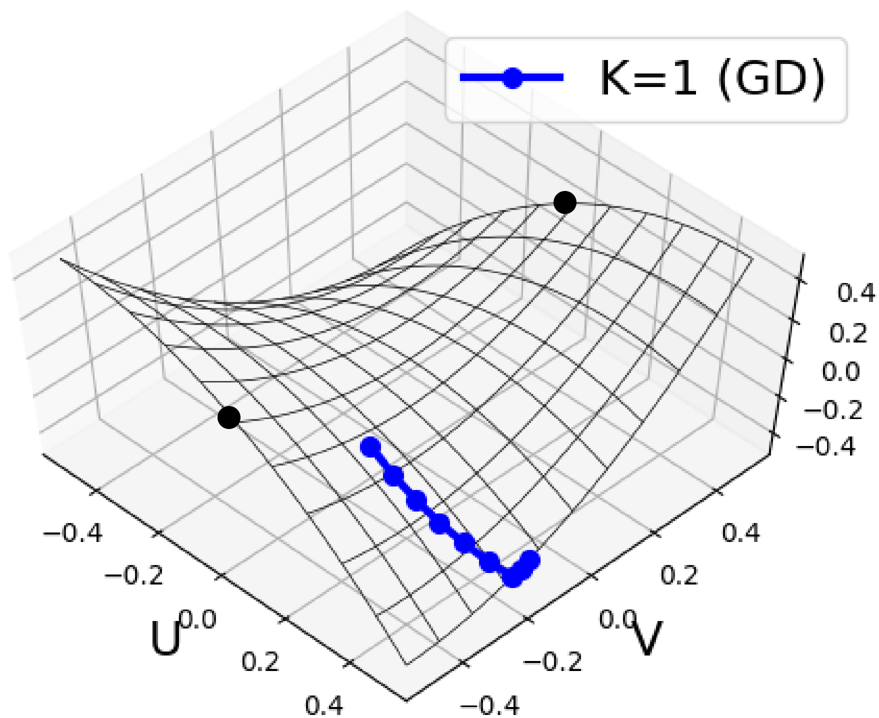
Gradient descent/ascent



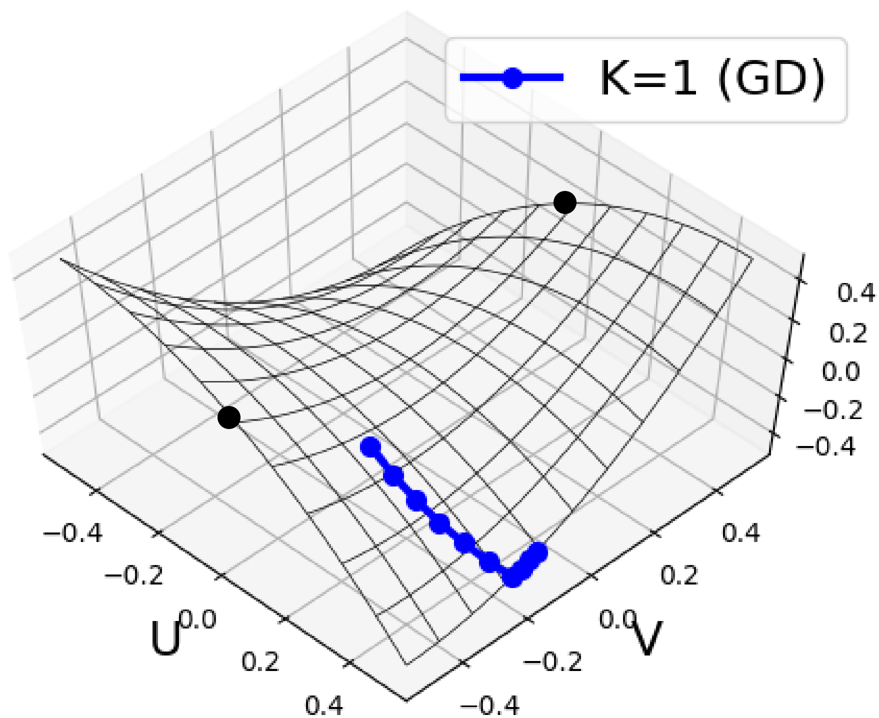
Gradient descent/ascent



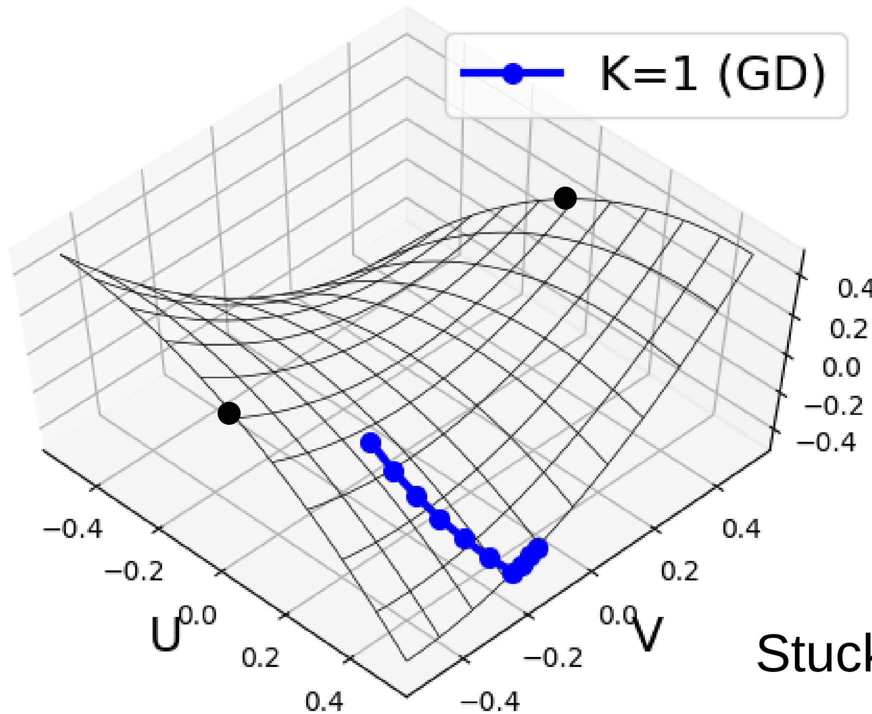
Gradient descent/ascent



Gradient descent/ascent



Gradient descent/ascent



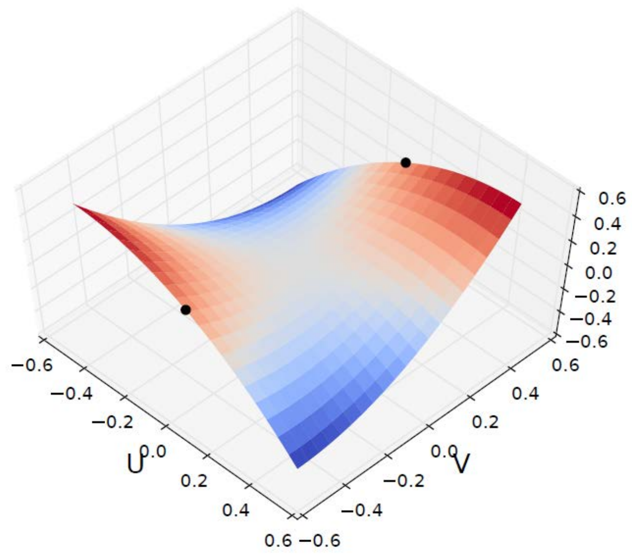
Stuck !!!

Minimax points ≠ saddle points

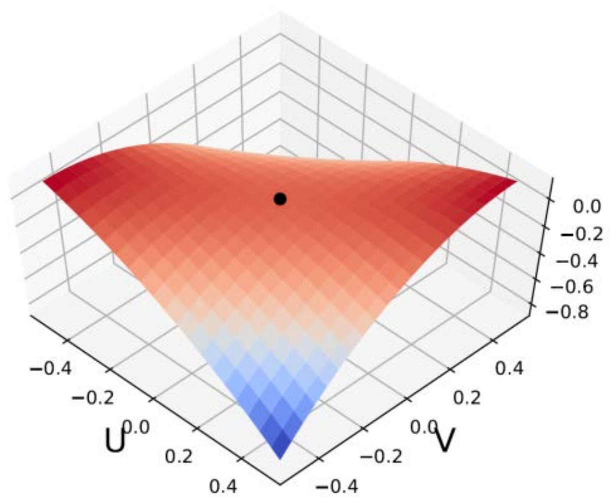
- Minimax point: $(u^*, v^*) = \text{argmin}_{\text{u}} \max_v f(u, v)$
- Saddle point : $f(u^*, v) \leq f(u^*, v^*) \leq f(u, v^*), \quad \forall (u, v)$

- Examples:

$f(u, v) = -u^2 + v^2 + 2uv,$



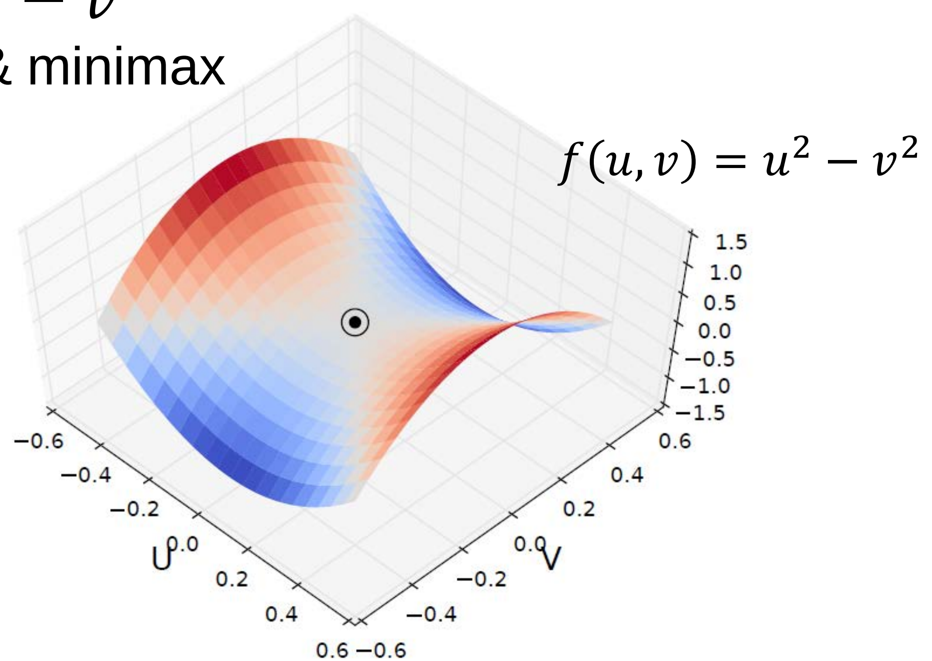
$f(u, v) = -0.5u^2 + 2uv - v^2$



Minimax points = saddle points

- Minimax point: $(u^*, v^*) = \operatorname{argmin}_{\mathrm{u}} \max_v \ f(u, v)$
- Saddle point : $f(u^*, v) \leq f(u^*, v^*) \leq f(u, v^*), \quad \forall (u, v)$

- Exception: von Neumann (1928)
  - Same if $f(u, v)$ is convex-concave
  - $f(u, v) = u^2 - v^2$
  - (0,0): saddle & minimax



Proposal

- Challenges
  - The solution we are seeking for GAN-like problems may not be a saddle point
  - Multiple local optima $R(u) = \underset{v}{\operatorname{argmax}} f(u, v)$
  - $R(u)$ is discontinuous even if $f(u, v)$ is continuous

Proposal

- Challenges
  - The solution we are seeking for GAN-like problems may not be a saddle point
  - Multiple local optima $R(u) = \underset{v}{\operatorname{argmax}} f(u, v)$
  - $R(u)$ is discontinuous even if $f(u, v)$ is continuous

- Our suggestion
  - Approximate the global solution $R(u)$
  - By keeping *K* candidate solutions

$$A_t = (v_t^1, v_t^2, \dots, v_t^K)$$

  - Can handle non-unique, discontinuous $R(u)$

K-beam minimax

- *Randomly initialize K candidates* $A_t = (v_t^1, v_t^2, \ldots, v_t^K)$
- *For t = 1,..., T*

  max step:

  - *For i = 1,...,K in parallel*

$$v_{t+1}^i \leftarrow v_t^i + \eta_t \nabla_v f(u_t, v_t^i)$$

  min step:

  - $\hat{v} = \operatorname*{argmax}_{v \in A_t} f(u, v)$
  - $u_{t+1} \leftarrow u_t - \rho_t \nabla_u f(u_t, \hat{v})$: gradient evaluated at the best v

- Related ideas [Durugkar'16, Saatci'17,Dem'yanov'72, Salmon'68]

ϵ-subgradient version

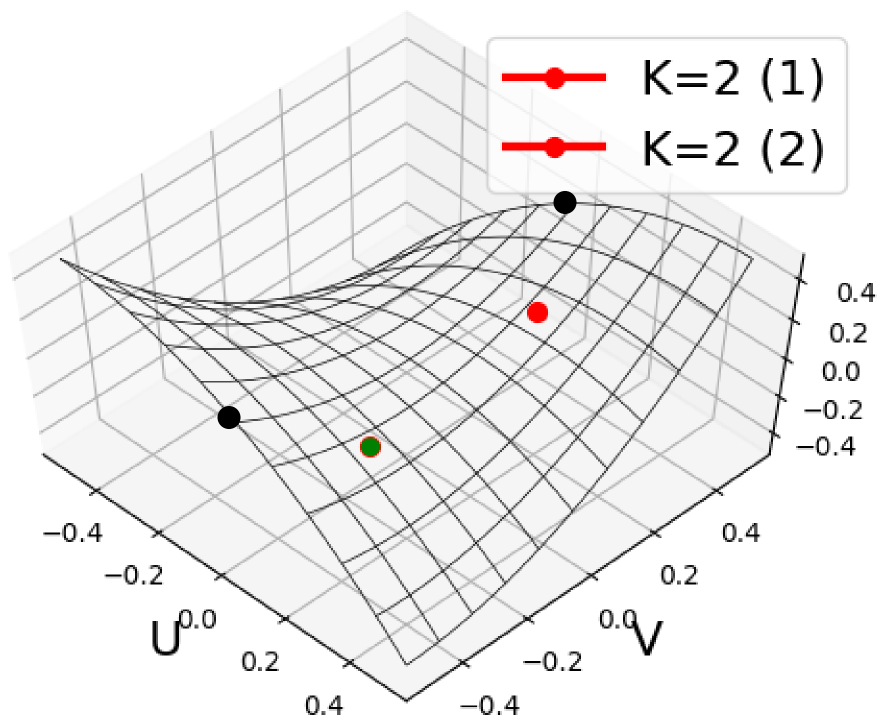
Algorithm 1 K-beam ϵ-subgradient descent

Input: $f, K, N, (\rho_i), (\eta_i), (\epsilon_i)$
Output: u_N, A_N
Initialize $u_0, A_0 = (v_0^1, ..., v_0^K)$
Begin
  for $i = 1, \ldots, N$ **do**
    Min step:
      Update $u_i = u_{i-1} + \rho_i \; g(u_i, A_i, \epsilon_i)$ where g is a descent direction from Alg. 2.
    Max step:
      for $k = 1, \ldots, K$ in parallel **do**
        Update $v_i^k \leftarrow v_{i-1}^k + \eta_i \; \nabla_v f(u_i, v_{i-1}^k)$.
      end for
      Set $A_i = (v_i^1, \ldots, v_i^K)$.
  end for

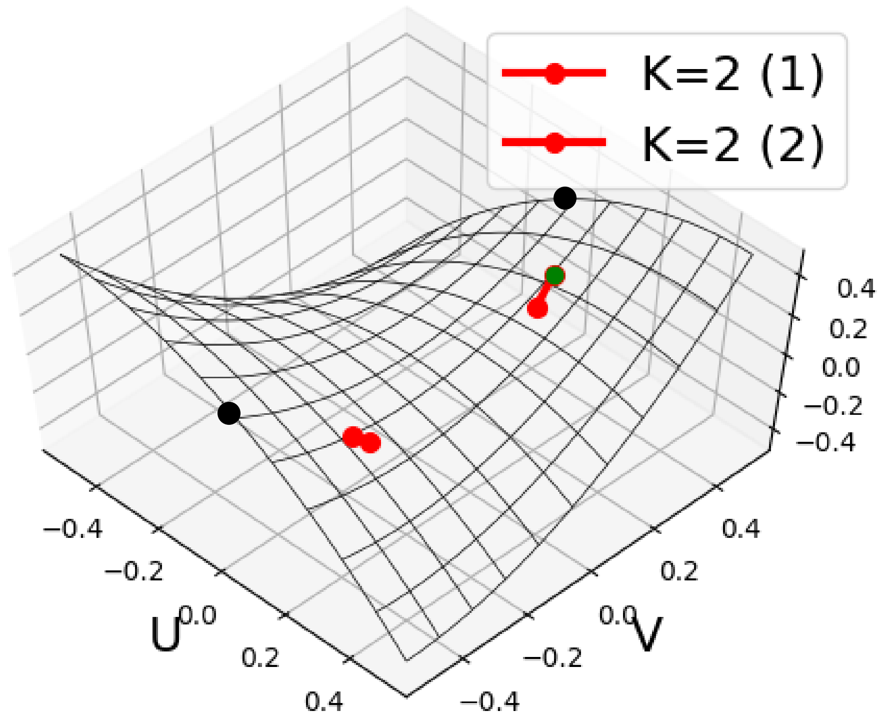
Algorithm 2 Descent direction

Input: $f, u, A = (v^1, ..., v^K), \epsilon$
Output: g
Begin
  Find $k_{\max} = \arg\max_{1 \le k \le K} f(u, v^k)$.
  Find $\{v^{k_1}, ..., v^{k_n}\} = R_A^\epsilon(u) = \{v \in A \mid f(u, v^{k_{\max}}) - f(u, v^k) \le \epsilon\}$.
  Compute $z_j = \nabla_u f(u, v^{k_j})$ for $j = 1, .., n$.
  Optional stopping criterion:
  if $0 \in \text{co}\{z_1 \cup ... \cup z_n\}$ **then**
    Found a stationary point. Quit optimization.
  end if
  Decent direction:
  if $n = 1$ **then**
    Return $g = -z_1$.
  else
    Randomly choose $z \in \text{co}\{z_1 \cup ... \cup z_n\}$ and return $g = -z$.
  end if

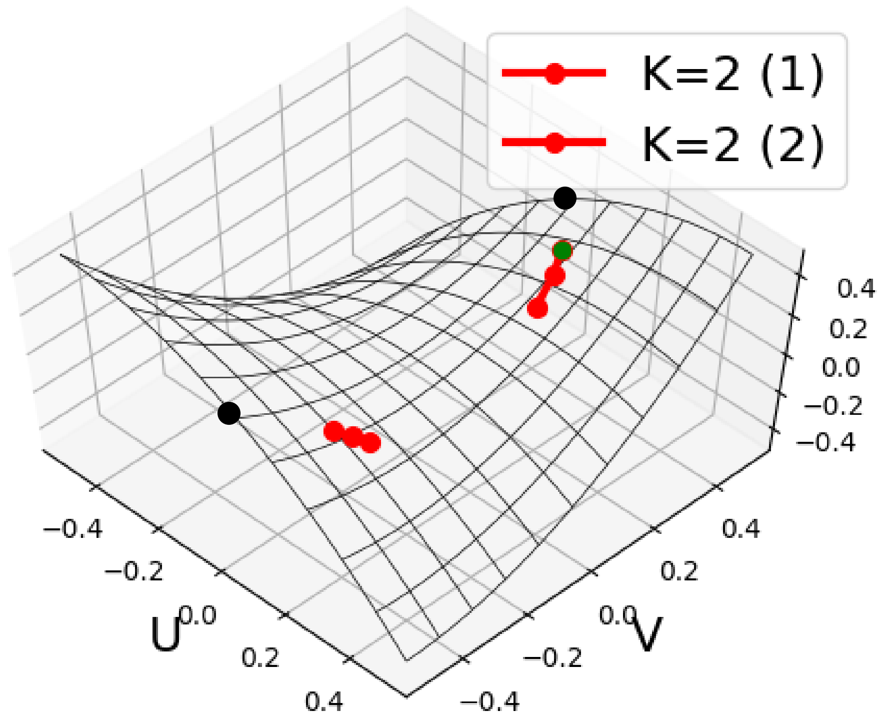
K-beam minimax



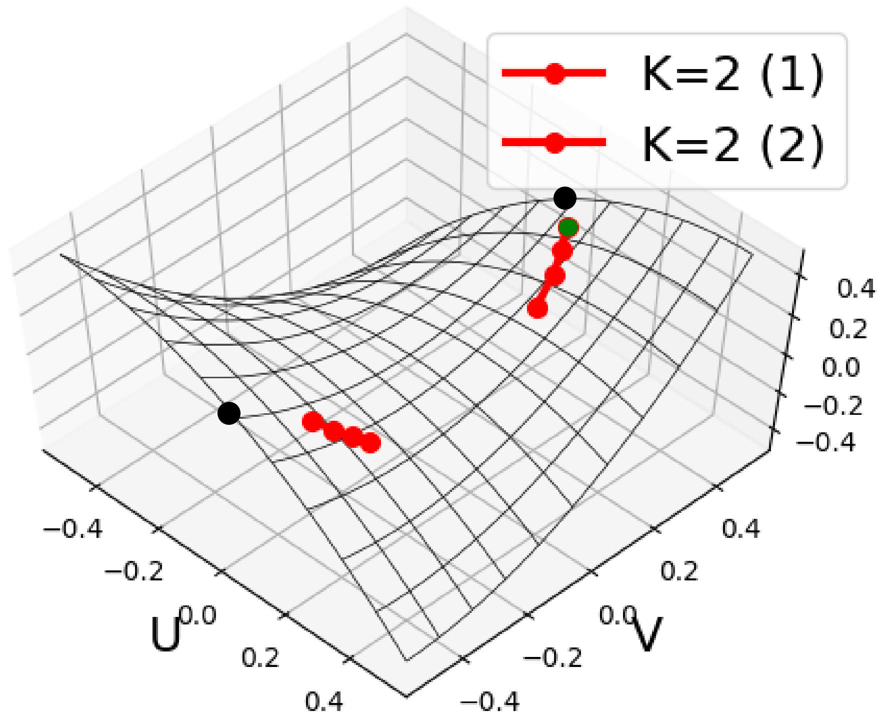
K-beam minimax



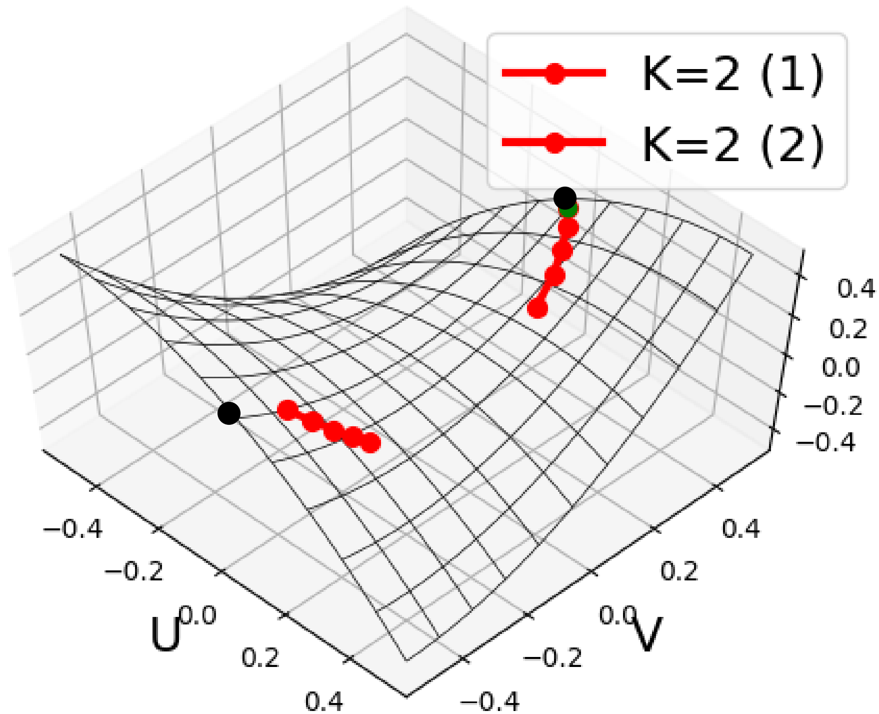
K-beam minimax



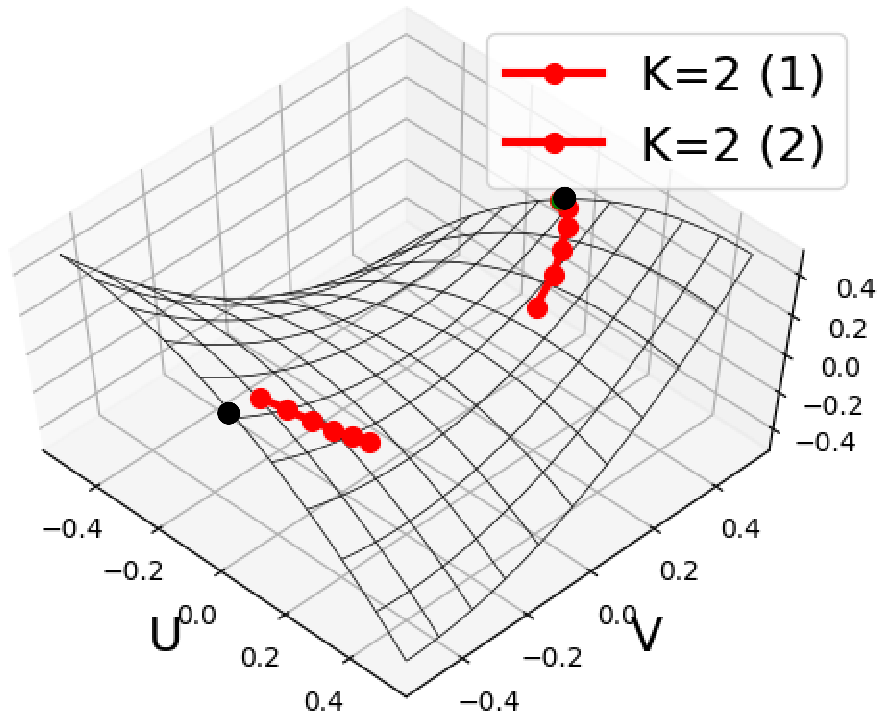
K-beam minimax



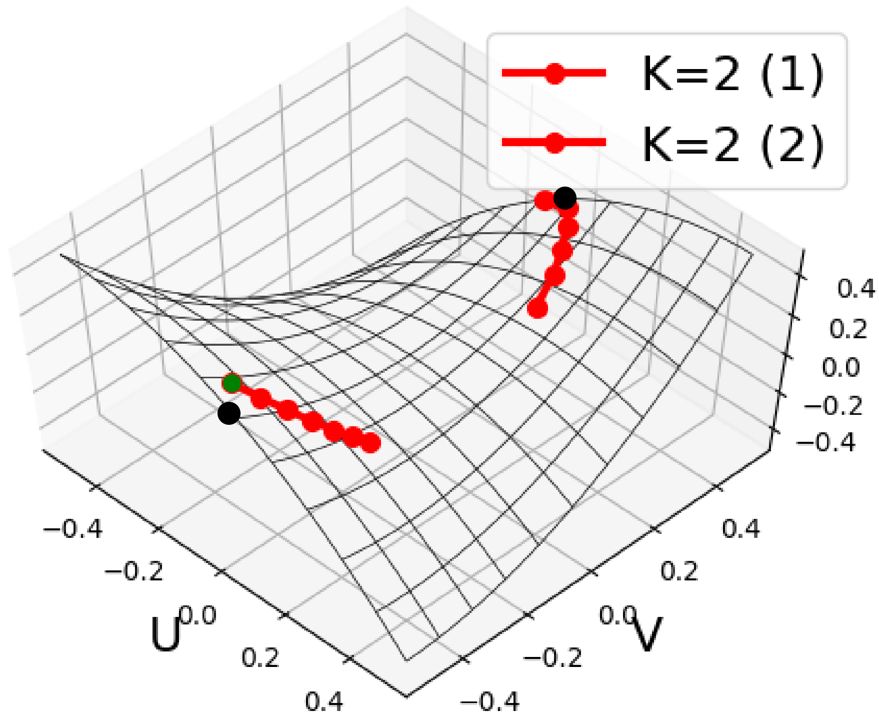
K-beam minimax



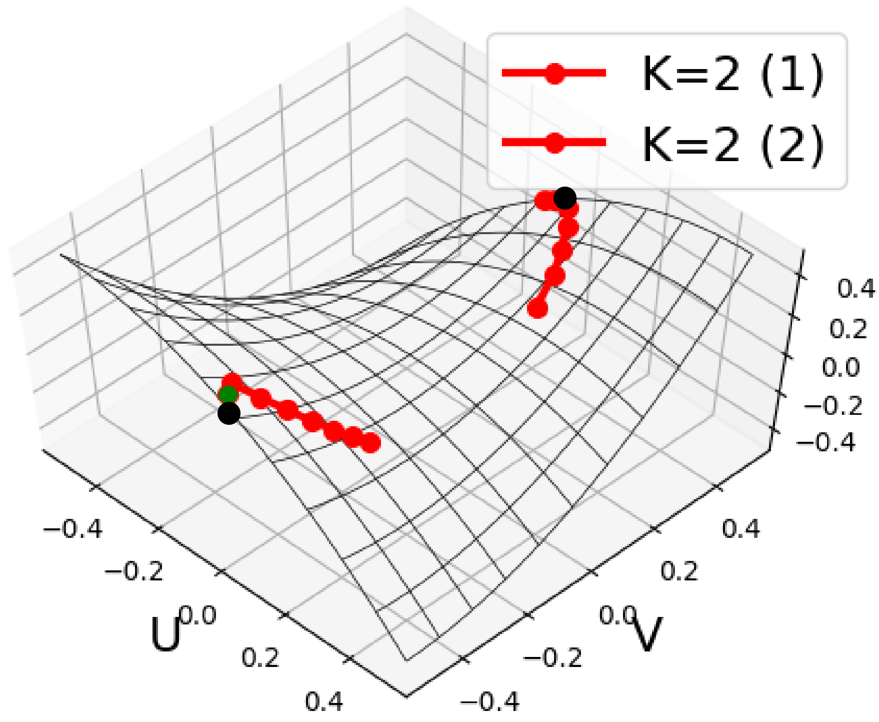
K-beam minimax



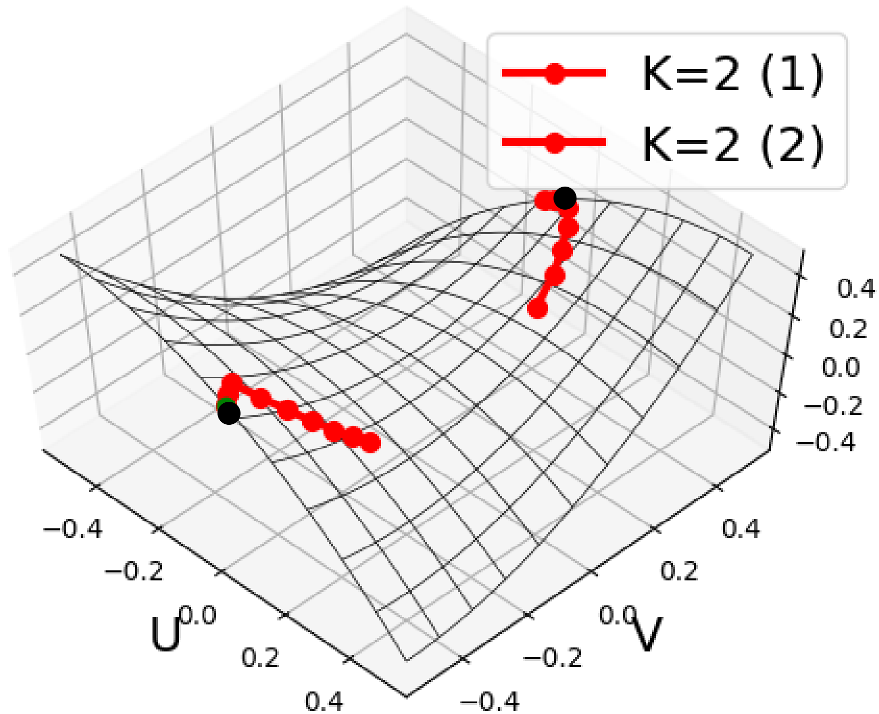
K-beam minimax



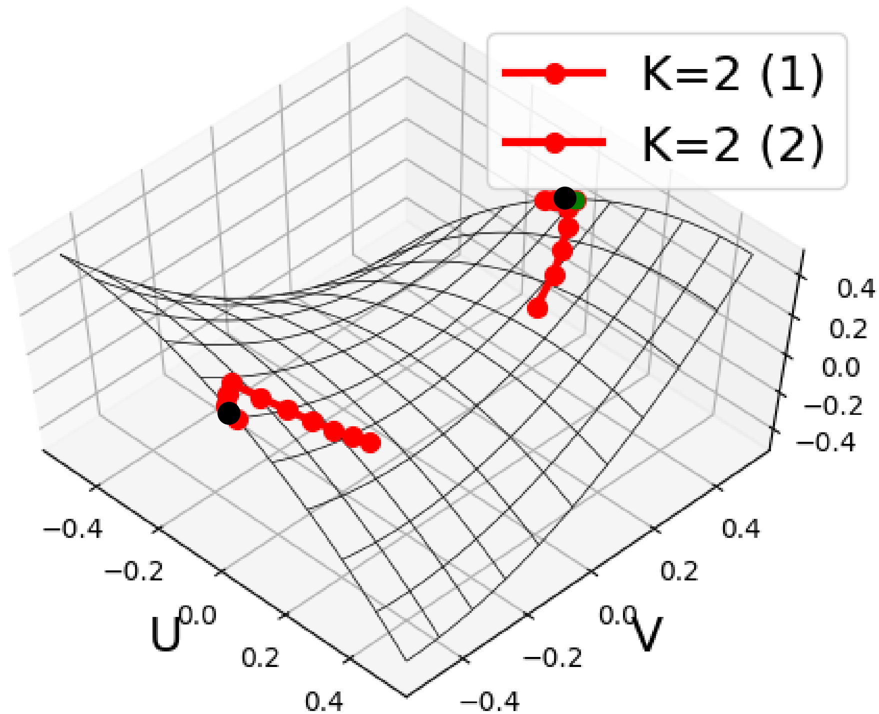
K-beam minimax



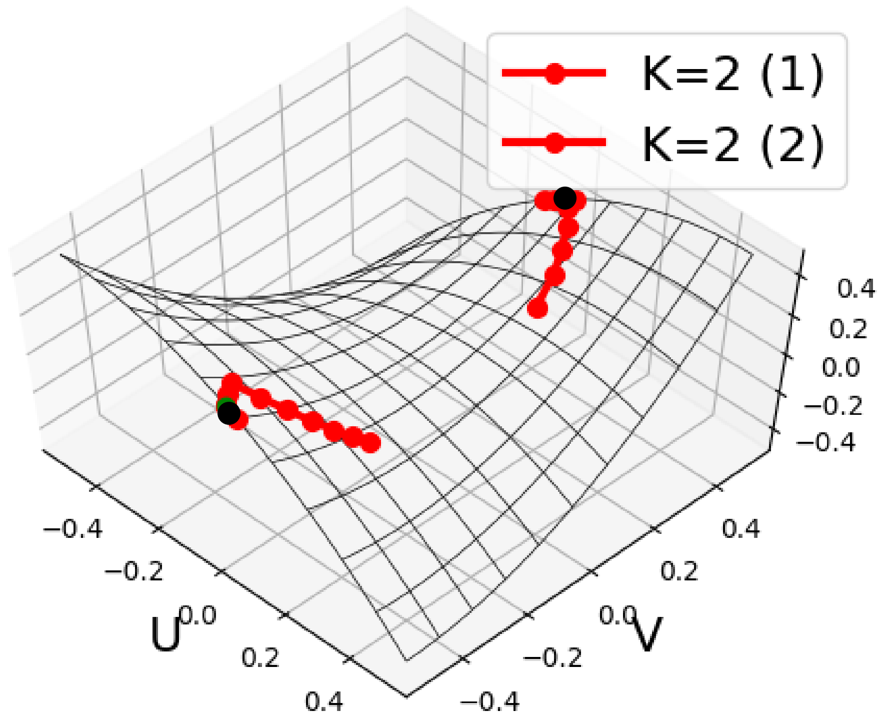
K-beam minimax



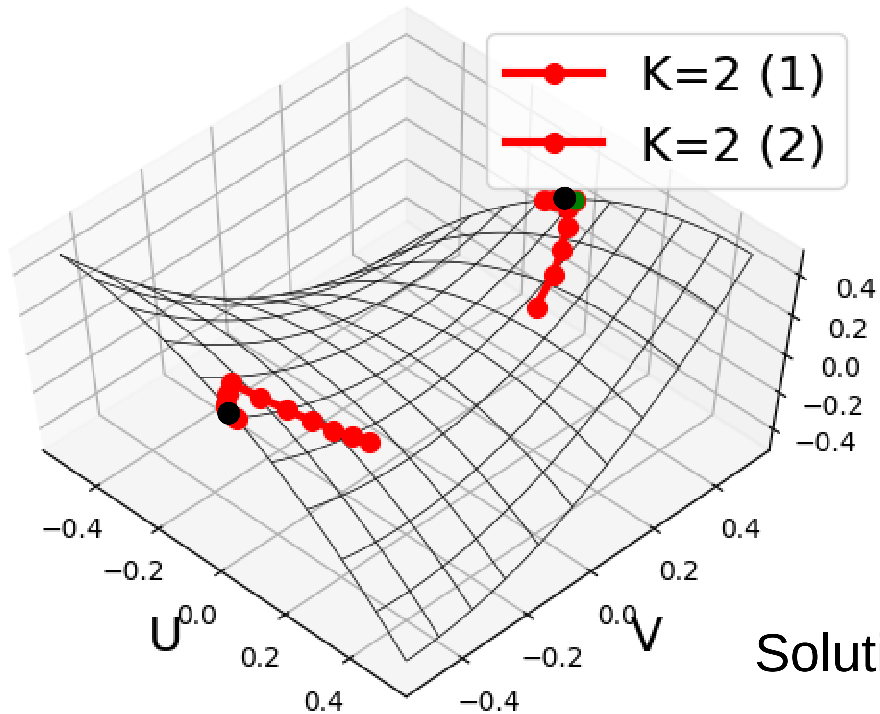
K-beam minimax



K-beam minimax



K-beam minimax



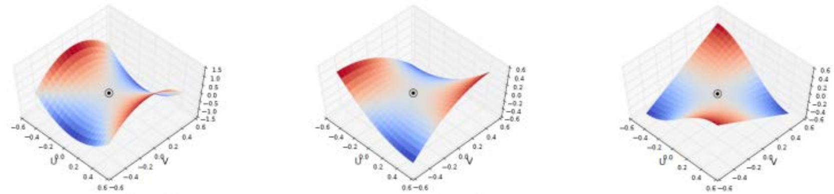
Solution reached !!!

Analysis

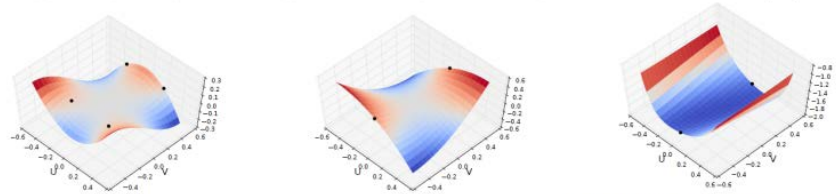
- Summary of theorems
  - K-beam converges to minimax points if A_t are sufficiently close to true $R(u_t)$ as $t \to \infty$
  - Closeness measured by two Hausdorff distances
    - $d_H(R(u_t), A_t) \coloneqq \max_{v \in R(u_t)} \min_{v' \in A_t} \|v - v'\|$ (how close is each true max to one of the K candidates)
    - $d_H\big(A_t, S(u_t)\big) \coloneqq \max_{v' \in A_t} \min_{v \in S(u_t)} \|v - v'\|$ (how close is each candidate to local maxima)
- Previously: requires $R(u)$ is unique, or $f = f_0(u) + g_0(v) + u^T B v$ (bilinear)
- Our proof: $f(u, v)$ need not be concave w.r.t. v

Experiments

- Simple 2D surface

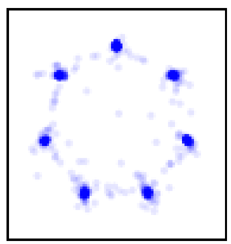


(a) Saddle $(u^2 - v^2)$ (b) Rotated saddle $(u^2 - v^2 + 2uv)$ (c) Seesaw $(-v \sin(\pi u))$



(d) Monkey saddle $(v^3 - 3vu^2)$ (e) Anti-saddle $(-u^2 + v^2 + 2uv)$ (f) Weapons (Danskin, 1967) $(e^{-10(u+.5)e^{-(v+.5)}} + e^{-10(.5-u)e^{v-.5}})$

- GAN with MoG

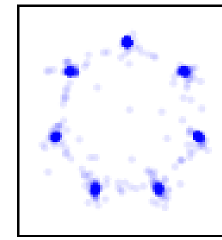


- Domain adaptation with MNIST and MNIST-M

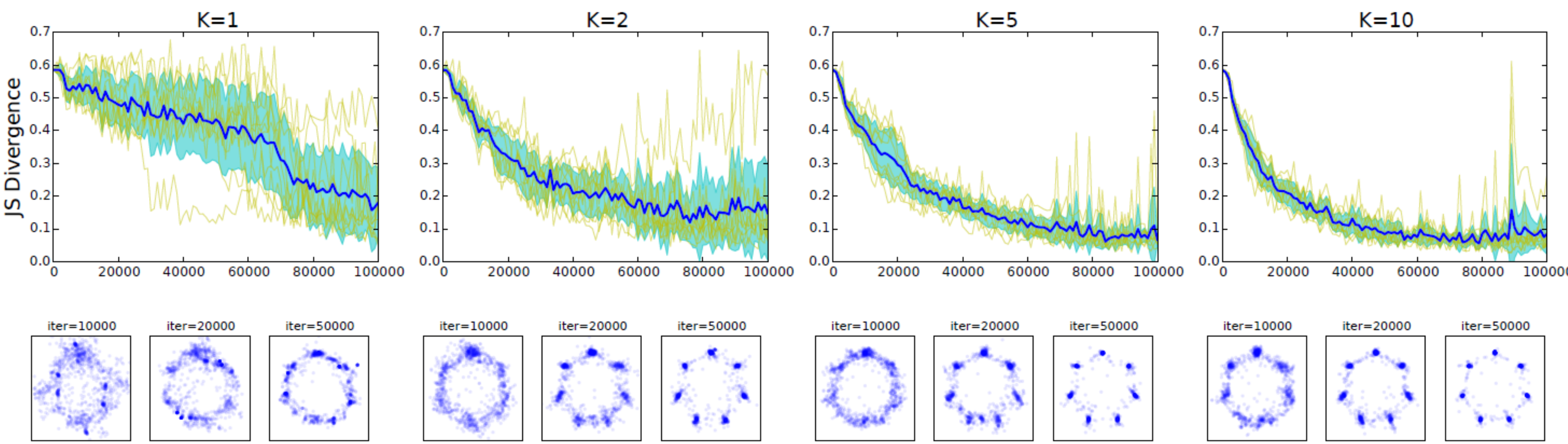
Experiments - GAN

- Jensen-Shannon divergence

$$\mathrm{JSD}(P,Q) := \frac{1}{2}KL\left(P,\frac{P+Q}{2}\right)+\frac{1}{2}KL\left(Q,\frac{P+Q}{2}\right)$$



- P: mixture of Gaussians, Q: generative (two-layer neural net)

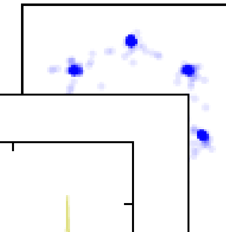


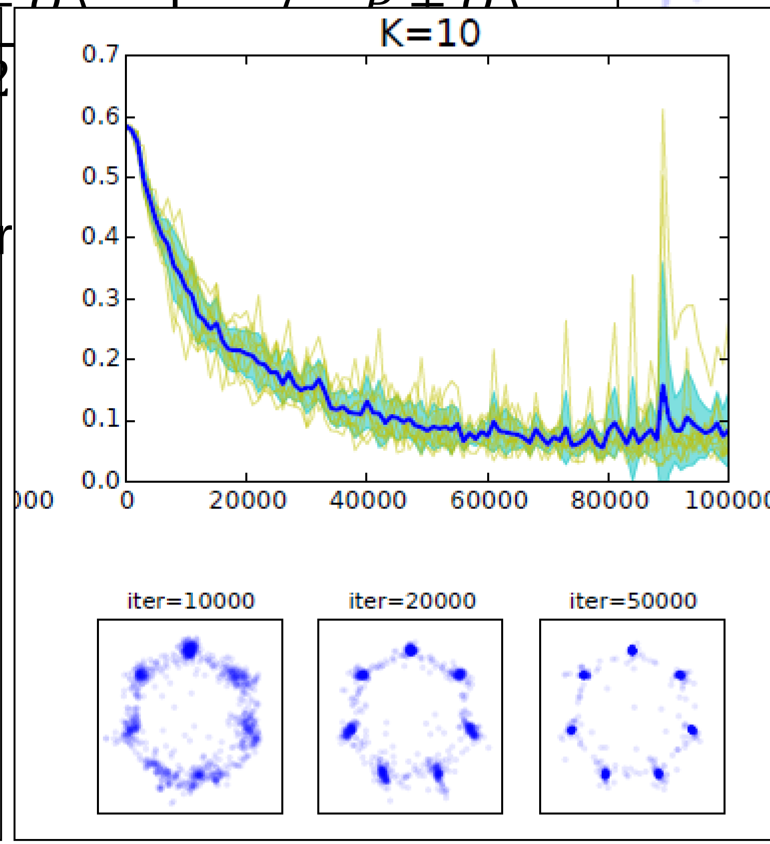
Experiments - GAN

- Jensen-Shannon divergence

$$\mathrm{JSD}(P,Q) := \frac{1}{2} KL\left(P, \frac{P+Q}{2}\right) \ldots$$

- P: mixture of Gaussians, Q: gener





Part 4/4. On-going research

- Efficient bilevel optimization
  - Based on "*Penalty Method for Inversion-Free Bilevel Optimization," arXiv, 2019*
- Adversarial attack
  - Based on J. Hamm and A. Mehra, "*Machine vs Machine: Minimax-Optimal Defense Against Adversarial Examples,*" NIPS Workshop on Machine Deception, 2017

Nash vs Stackelberg game

- From Saddle point to Nash equilibrium
  - Two-player zero-sum (Saddle point) :

$$\forall(u, v), \qquad f(u^*, v) \le f(u^*, v^*) \le f(u, v^*),$$

  - N-players, general-sum:

$$\forall u, \forall i, \qquad f_i(u_i^*, u_{-i}^*) \le f_i(u_i, u_{-i}^*)$$

- From Minimax to Stackelberg equilibrium
  - Two-player zero-sum (Minimax):

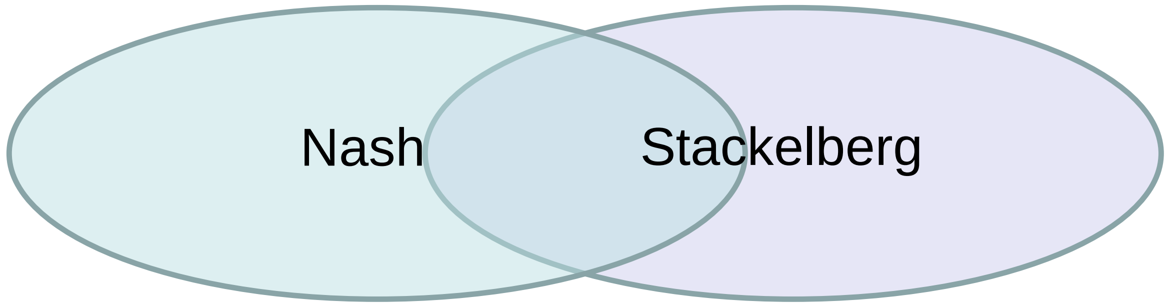
$$(u^*, v^*) = \mathrm{argmin_u} \max_v \; f(u, v)$$

  - N-player, general-sum:

$$u_1^* = \mathrm{argmin}_{u_1} f_1(u_1, u_{-1})$$
$$s.t.\; u_2 = \mathrm{argmin}_{u_2} f_2(u_2, u_{-2})$$
$$s.t. \cdots \quad s.t.\; u_N = \mathrm{argmin}_{u_N} f_N(u_N, u_{-N})$$

	Topic 1 - Nash Equilibrium	Topic 2 - Stackelberg Equilibrium
zero-sum, two-player	"Saddle-point problem" $\forall(u,v), \quad f(u^*,v) \leq f(u^*,v^*)$ $\forall(u,v), \quad f(u^*,v^*) \leq f(u,v^*)$	"Minimax problem" $(u^*,v^*) = \arg\min_{\mathrm{u}} \max_{v} \ f(u,v)$
general-sum, two-player	$\forall u, \quad f(u^*,v^*) \leq f(u,v^*)$ $\forall v, \quad g(u^*,v) \leq g(u^*,v^*)$	"Bilevel programming" $u^* = \min_{u} f(u,v)$ $s.t. \ \ v = \min_{v} g(u,v)$
general-sum, N-player	$\forall u, \forall i,$ $f_i(u_i^*, u_{-i}^*) \leq f_i(u_i, u_{-i}^*)$	"Multilevel programming" $u_1^* = \mathrm{argmin}_{u_1} \ f_1(u_1, u_{-1})$ $s.t. \ u_2 = \mathrm{argmin}_{u_2} \ f_2(u_2, u_{-2})$ $s.t. \ \cdots$ $s.t. \ u_N = \mathrm{argmin}_{u_N} \ f_N(u_N, u_{-N})$

Is GAN Nash or Stackelberg game?



- Some GAN-like problems seems to fit either NE or SE better
- Some problems, GAN in particular, are unclear
  - GAN was analyzed as SE problem, but solved as NE problem

- Properties and algorithms for NE is relatively well-known
- Less so for SE, much room for improving SE

Finding Stackelberg Equilibrium

- Bilevel problem: two-player, general-sum game
  - Upper/outer problem: $\min_{u,v} \ f(u,v)$
  - Lower/inner problem: $s.t. \quad v = \text{argmin}_v \ g(u,v)$
  - Minimax $\min_u \max_v f(u,v)$ is a special (=zero-sum) case:
    $$g(u,v) = -f(u,v)$$
  - More complicated than minimax

Bilevel optimization is numerically hard

- $\min_{u,v} \; f(u,v) \quad s.t. \quad v = \text{argmin}_v \; g(u,v)$
- Single-level equivalent: $\min_{u,v} f(u, v^*(u))$, where
$$v^*(u) = \underset{v}{\text{argmin}} \, g(u,v)$$
- Total derivative $\frac{d}{du} f\big(u, v^*(u)\big)$ is, from implicit function theorem,
$$\frac{d}{du} f = \nabla_u f - \nabla^2_{uv} g \boxed{(\nabla^2_{vv} g)^{-1} \nabla_v f}$$
- Requires product of inverse-Hessian and gradient !!
  - Can we avoid it?

- Total derivate requires computing $(\nabla^2_{vv} g)^{-1} \nabla_v f$
- Several methods proposed
  - Approximate Hessian inverse [Domke'12,Pedregosa'16l]
  - Forward/Reverse-mode differentiation [Maclaurin et al.'15,Francheschi et al.'17]
  - High space or time complexity

Penalty-based method

- Original: $\min\limits_{u,v} f(u,v) \quad s.t. \quad v = \text{argmin}\, g(u,v)$ (1)
- For convex, differentiable g, same as

$$\min_{u,v} f(u,v) \quad s.t. \quad \nabla_v g(u,v) = 0 \qquad (2)$$

- Use Penalty-method:

$$\min_{u,v} \tilde{f}\,(u,v;\gamma_k) = f(u,v) + \frac{\gamma_k}{2}\|\nabla_v g(u,v)\|^2 \qquad (3)$$

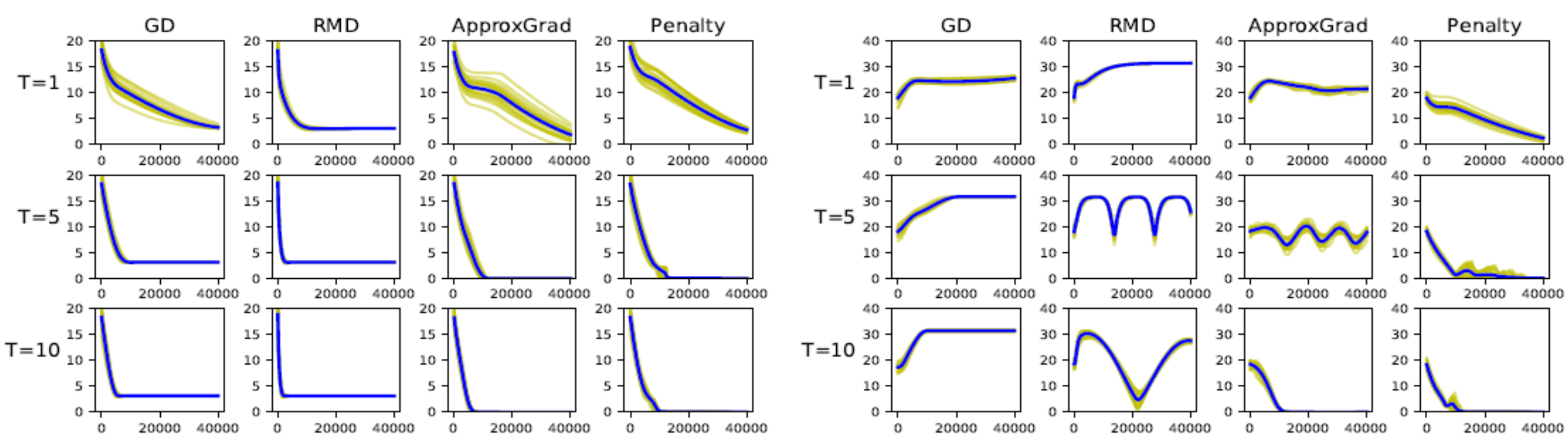
- Theorem: As $\gamma_k \to \infty$, limit point of the solution of (3) is a KKT point of (2)

Preliminary results

- Complexity

Method	v-update	Intermediate update	Time	Space
FMD	$v \leftarrow v - \rho\nabla_v g$	$P \leftarrow P(I - \rho\nabla^2_{vv}g) - \rho\nabla^2_{uv}g$	$O(UV^2T)$	$O(UV)$
RMD	$v \leftarrow v - \rho\nabla_v g$	$p \leftarrow p - \rho\nabla^2_{uv}g \cdot q$ $q \leftarrow q - \rho\nabla^2_{vv}g \cdot q$	$O(VT)$	$O(U + VT)$
ApproxGrad	$v \leftarrow v - \rho\nabla_v g$	$q \leftarrow q - \rho\nabla^2_{vv}g[\nabla^2_{vv}g \cdot q - \nabla_v f]$	$O(VT)$	$O(U+V)$
Penalty	$v \leftarrow v - \rho[\nabla_v f + \gamma\nabla^2_{vv}g\nabla_v g]$	Not required	$O(VT)$	$O(U+V)$

- Synthetic examples



Applications of SE

- Data denoising by importance learning

$$\min_{u} E_{val}(u,w) \qquad s.t.\ \ w = \text{argmin} \frac{1}{\sum_i u_i} \sum_{(x_i,y_i)\in D_{tr}} u_i l(h(x_i;w),y_i)$$

- Few-shot learning

Let $E_{val}(u,w_i) := \frac{1}{N_{val}} \sum_{(x_i,y_i)\in D_{val}} l(h_i(T(x_i;u);w_i),y_i)$

$$\min_{\text{u}} \sum_i E_{val}(u,w_i) \qquad s.t.\quad w_i = \arg\min_{w_i} E_{tr}(u,w_i)\,,$$

- Training data poisoning

Let $E_{poison}(u,w) := \frac{1}{N} \sum_{(x_i,y_i)\in X'\times Y} l(h(x_i;u,w),y_i)$

$$\min_{\text{u}} -E_{val}(u,w) \qquad s.t.\quad w = \text{argmin}_{\text{w}}\, E_{poison}(u,w)$$

Thank you !

- Adversarial learning (game + ML) is a new approach in ML with potentially ground-breaking advances
- Many applications in biomedical science
- Lots of interesting open questions to explore