Improving GANs using Game Theory and Statistics

Constantinos Daskalakis
CSAIL and EECS, MIT

Min-Max Optimization

Solve: $\inf_{\theta} \sup_{w} f(\theta, w)$

where θ, w high-dimensional

- **Applications:** Mathematics, Optimization, Game Theory,…

  [von Neumann 1928, Dantzig ’47, Brown’50, Robinson’51, Blackwell’56,..

BEGAN. Bertholet et al. 2017.

- **Best-Case Scenario:** f is convex in θ, concave in w

- **Modern Applications:** GANs, adversarial examples, …
  - exacerbate the importance of first-order methods, non convex-concave objectives

GAN Outputs

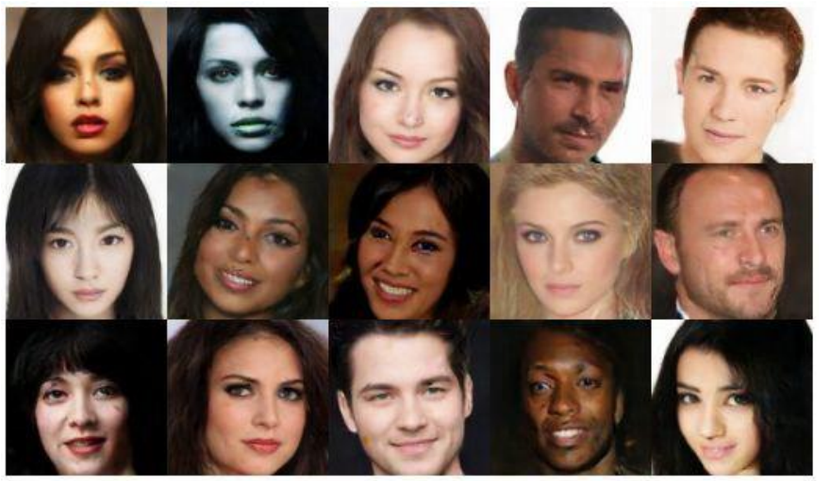
(a) Church outdoor.

(b) Dining room.

(c) Kitchen.

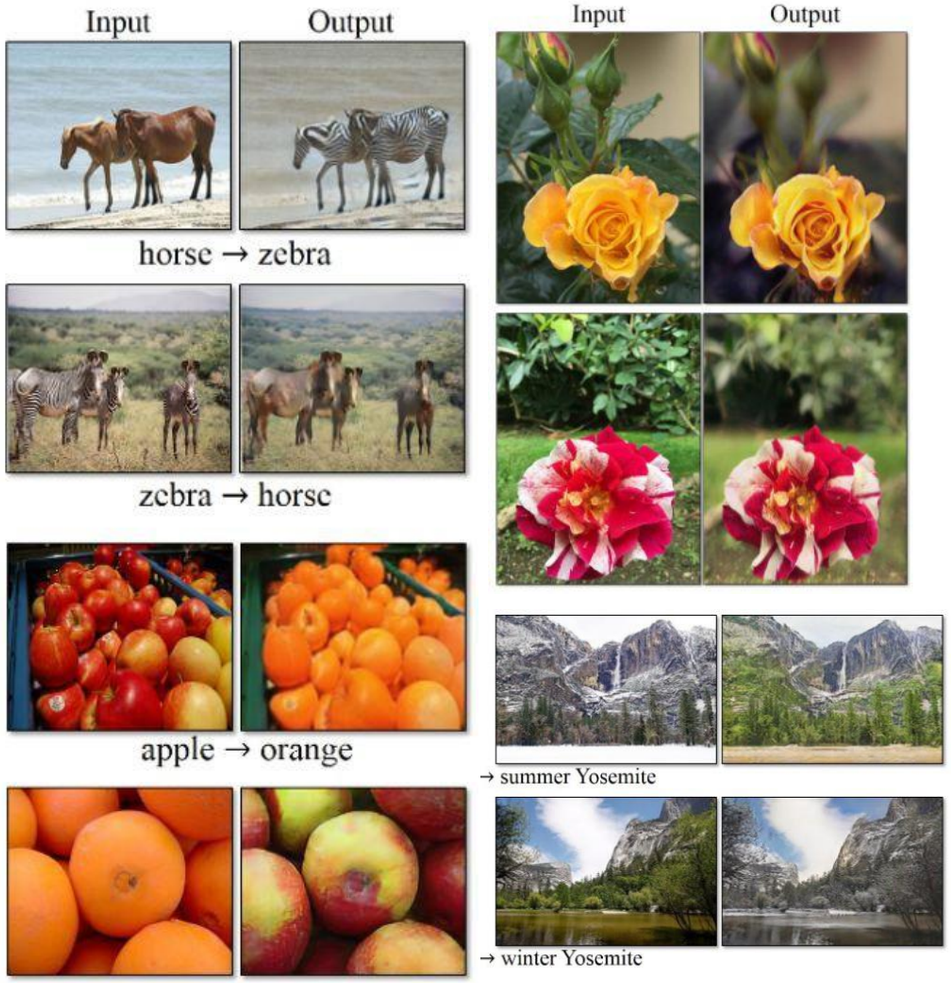
(d) Conference room.

LSGAN. Mao et al. 2017.



BEGAN. Bertholet et al. 2017.

GAN uses



CycleGAN. Zhu et al. 2017.

Text -> Image Synthesis

Reed et al. 2017.

Pix2pix. Isola 2017. Many examples at https://phillipi.github.io/pix2pix/

Many applications:

- Domain adaptation
- Super-resolution
- Image Synthesis
- Image Completion
- Compressed Sensing
- …

Min-Max Optimization

Solve: $\inf_{\theta} \sup_{w} f(\theta, w)$

where θ, w high-dimensional

- **Applications:** Mathematics, Optimization, Game Theory,…

  [von Neumann 1928, Dantzig ’47, Brown’50, Robinson’51, Blackwell’56,..

BEGAN. Bertholet et al. 2017.

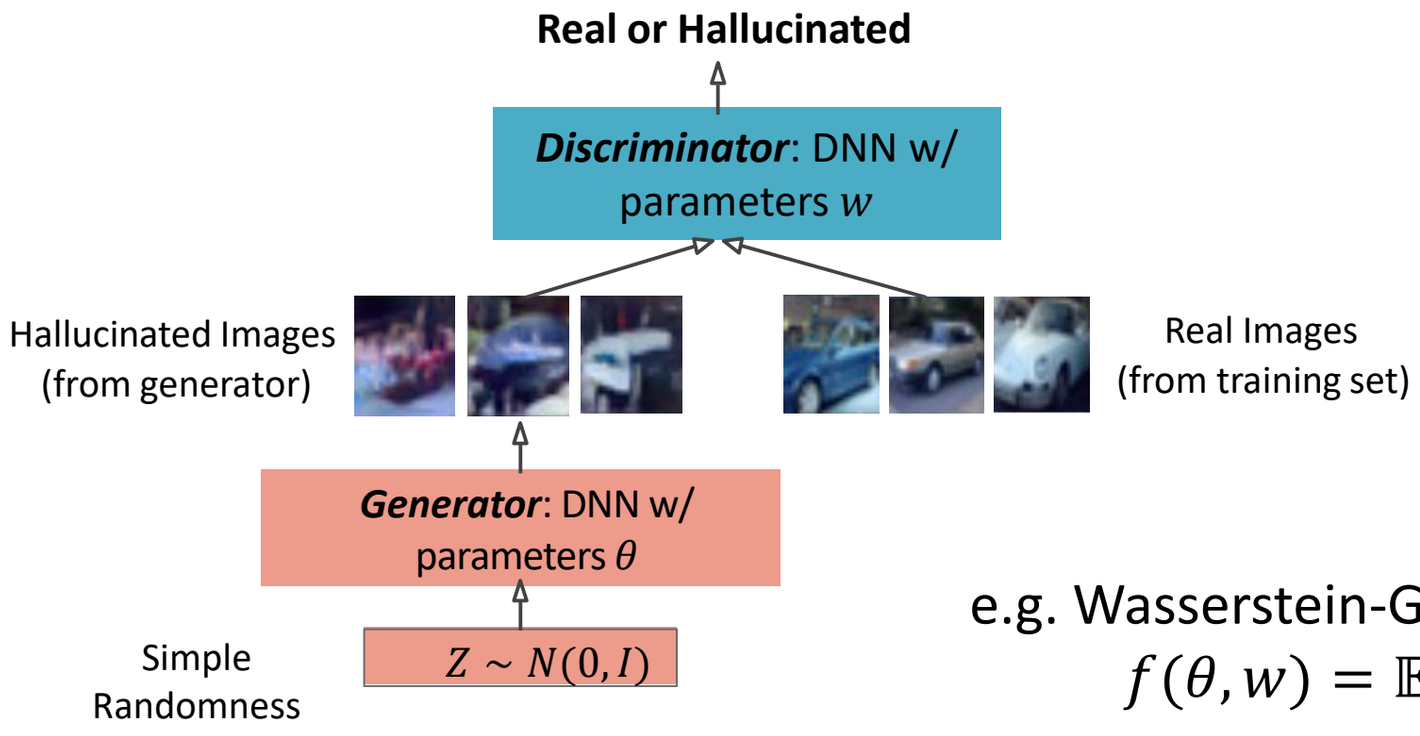
- **Best-Case Scenario:** f is convex in θ, concave in w

- **Modern Applications:** GANs, adversarial examples, …
  - exacerbate the importance of first-order methods, non convex-concave objectives

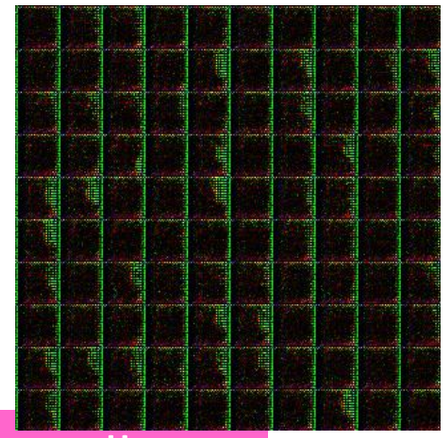
- **Personal Perspective:** applications of min-max optimization will multiply, going forward, as ML develops more complex and harder to interpret algorithms
  - sup players will be introduced to check the behavior of the inf players

Generative Adversarial Networks (GANs)

[Goodfellow et al. NeurIPS'14]



$$\inf_{\theta} \sup_{w} f(\theta, w)$$



expresses how well Discriminator distinguishes true vs generated images

e.g. Wasserstein-GANs:

$$f(\theta, w) = \mathbb{E}_{X \sim p_{real}}[D_w(X)] - \mathbb{E}_{Z \sim N(0,I)}[D_w(G_\theta(Z))]$$

- θ, w: high-dimensional

  ⇝ solve game by having min (resp. max) player run online gradient descent (resp. ascent)

- **major challenges:**
  - training oscillations
  - generated & real distributions high-dimensional ⇝ no rigorous statistical guarantees

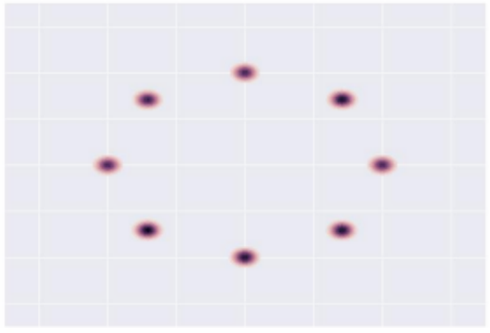
Menu

- **Min-Max Optimization and Adversarial Training**
- Training Challenges:
  - reducing training oscillations
- Statistical Challenges:
  - reducing sample requirements
  - attaining statistical guarantees

Menu

- **Min-Max Optimization and Adversarial Training**
- **Training Challenges:**
  - **reducing training oscillations**
- Statistical Challenges:
  - reducing sample requirements
  - attaining statistical guarantees

Training Oscillations: Gaussian Mixture



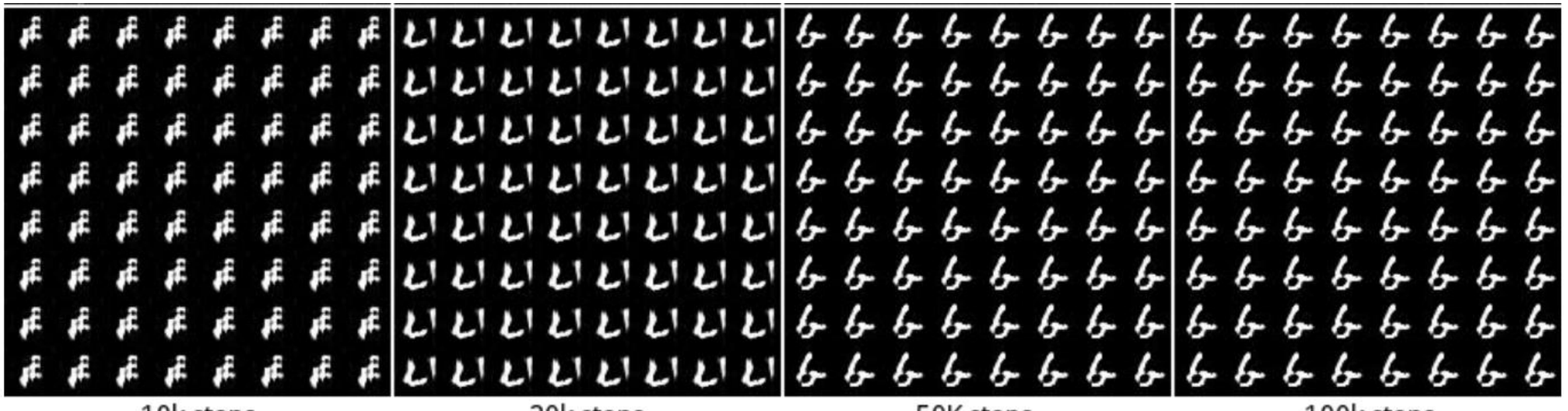
True Distribution: Mixture of
8 Gaussians on a circle

Output Distribution of standard GAN, trained via gradient descent/ascent dynamics:
cycling through modes at different steps of training

from **[Metz et al ICLR'17]**

Training Oscillations: Handwritten Digits

True Distribution: MNIST



Output Distribution of standard GAN, trained via gradient descent/ascent dynamics *cycling through "proto-digits" at different steps of training*

from **[Metz et al ICLR'17]**

Training Oscillations: even for bilinear objectives!

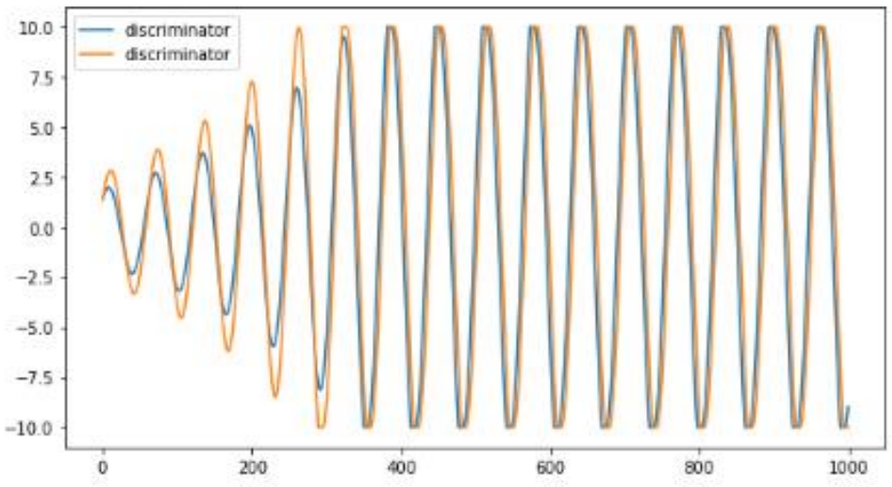
- **True distribution:** isotropic Normal distribution, namely $X \sim \mathcal{N}\left(\begin{bmatrix}3\\4\end{bmatrix}, I_{2\times 2}\right)$
- **Generator architecture**: $G_{\boldsymbol{\theta}}(Z) = \boldsymbol{\theta} + Z$ (adds input Z to internal params)
- **Discriminator architecture**: $D_{\boldsymbol{w}}(\cdot) = \langle \boldsymbol{w}, \cdot \rangle$ (linear projection)

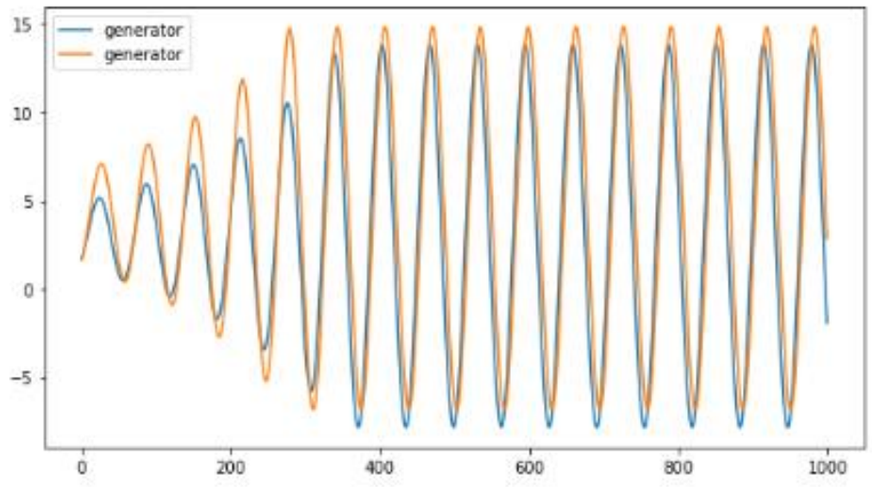
Z, θ, w: 2-dimensional

- **W-GAN objective:** $\min_{\boldsymbol{\theta}} \max_{\boldsymbol{w}} \mathbb{E}_X[D_{\boldsymbol{w}}(X)] - \mathbb{E}_Z[D_{\boldsymbol{w}}(G_{\boldsymbol{\theta}}(Z))]$

$$= \min_{\boldsymbol{\theta}} \max_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}} \cdot \left(\begin{bmatrix}3\\4\end{bmatrix} - \boldsymbol{\theta}\right)$$

convex-concave function

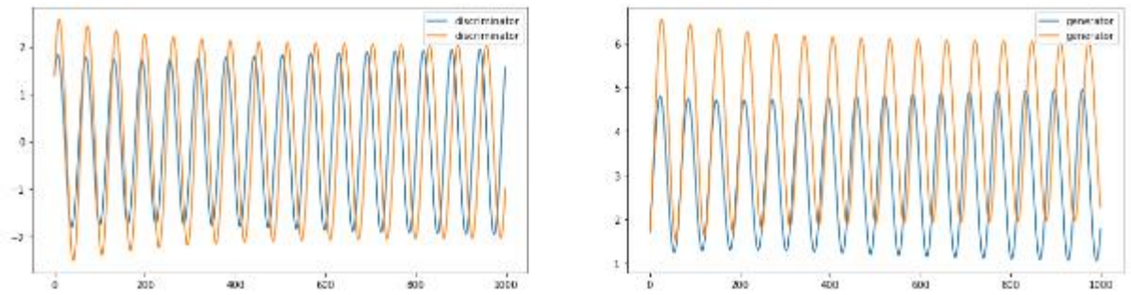




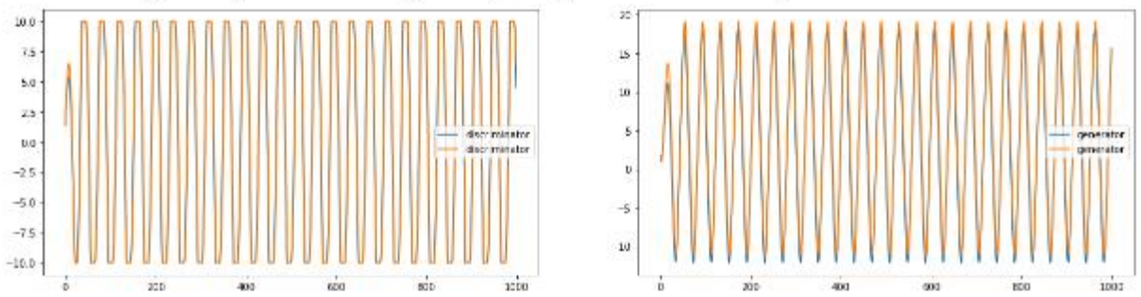
Gradient Descent Dynamics

from **[Daskalakis, Ilyas, Syrgkanis, Zeng ICLR'18]**

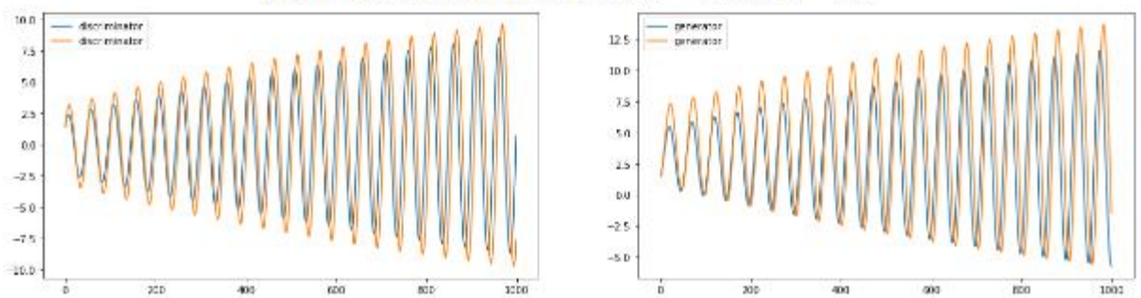
Training Oscillations:
persistence under many variants of Gradient Descent



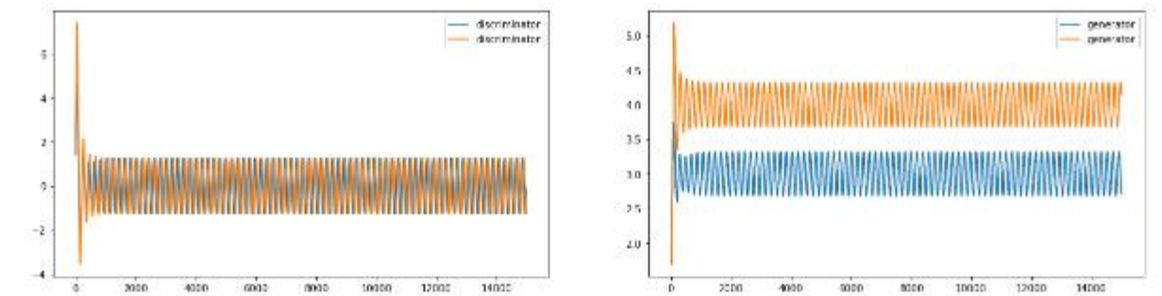
(a) GD dynamics with a gradient penalty added to the loss. $\eta = 0.1$ and $\lambda = 0.1$.



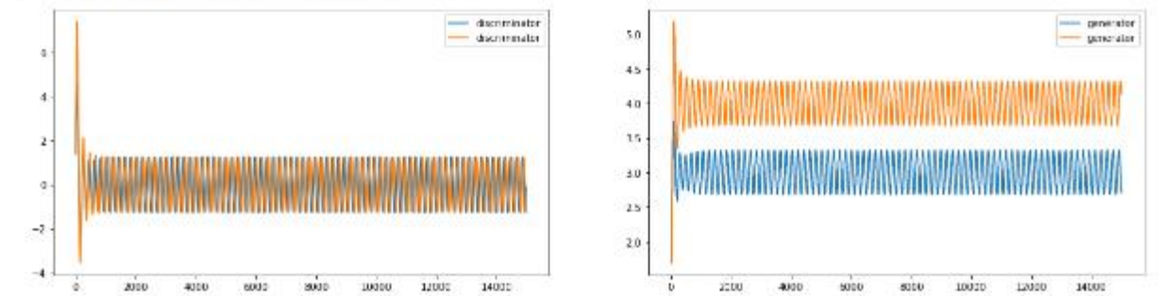
(b) GD dynamics with momentum. $\eta = 0.1$ and $\gamma = 0.5$.



(c) GD dynamics with momentum and gradient penalty. $\eta = .1$, $\gamma = 0.2$ and $\lambda = 0.1$.



(d) GD dynamics with momentum and gradient penalty, training generator every 15 training iterations of the discriminator. $\eta = .1$, $\gamma = 0.2$ and $\lambda = 0.1$.



(e) GD dynamics with Nesterov momentum and gradient penalty, training generator every 15 training iterations of the discriminator. $\eta = .1$, $\gamma = 0.2$ and $\lambda = 0.1$.

from **[Daskalakis, Ilyas, Syrgkanis, Zeng ICLR'18]**

Training Oscillations:
Online Learning Perspective

- **Best-Case Scenario:** Given ***convex-concave*** $f(x, y)$, solve: $\min_{x \in X} \max_{y \in Y} f(x, y)$
- **[von Neumann'28]:** min-max=max-min; solvable via convex-programming
- **Online Learning**: if min and max players run any no-regret learning procedure they converge to minimax equilibrium
  - E.g. follow-the-regularized-leader (FTRL), follow-the-perturbed-leader, MWU
  - Follow-the-regularized-leader with ℓ_2^2-regularization $\equiv$ gradient descent
- **"Convergence:"** Sequence $(x_t, y_t)_t$ converges to minimax equilibrium in the **average sense**, i.e. $f\left(\frac{1}{t}\sum_{\tau \le t} x_\tau, \frac{1}{t}\sum_{\tau \le t} y_\tau\right) \to \min_{x \in X} \max_{y \in Y} f(x, y)$
- **Can we show point-wise convergence of no-regret learning methods?**
  - **[Mertikopoulos-Papadimitriou-Piliouras SODA'18]:** No for any FTRL

Negative Momentum

- Variant of gradient descent:

$$\forall t\colon\ x_{t+1} = x_t - \eta \cdot \nabla f(x_t) + \boldsymbol{\eta/2 \cdot \nabla f(x_{t-1})}$$

- **Interpretation:** undo today, some of yesterday's gradient; ie negative momentum

- Gradient Descent w/ negative momentum

  = **Optimistic** FTRL w/ ℓ_2^2-regularization

  [Rakhlin-Sridharan COLT'13, Syrgkanis et al. NeurIPS'15]

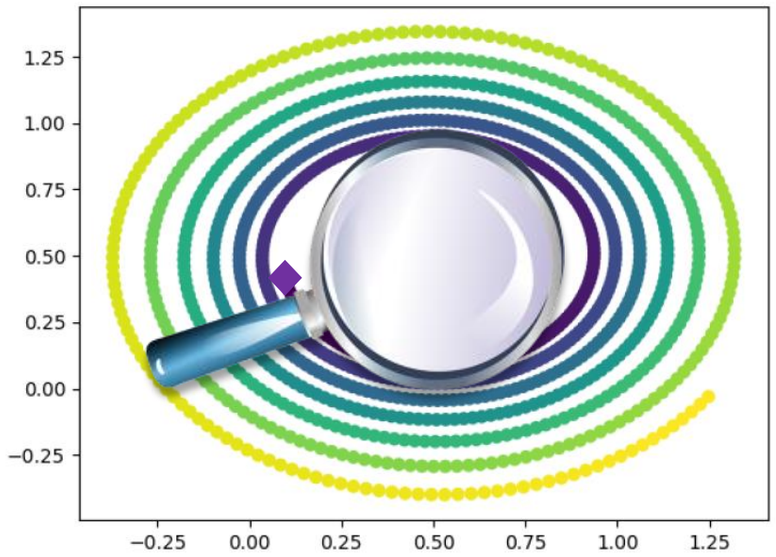
  $\approx$ **extra-gradient** method

  [Korpelevich'76, Chiang et al COLT'12, Gidel et al'18, Mertikopoulos et al'18]

- **Does it help in min-max optimization?**

Negative Momentum: why it could help

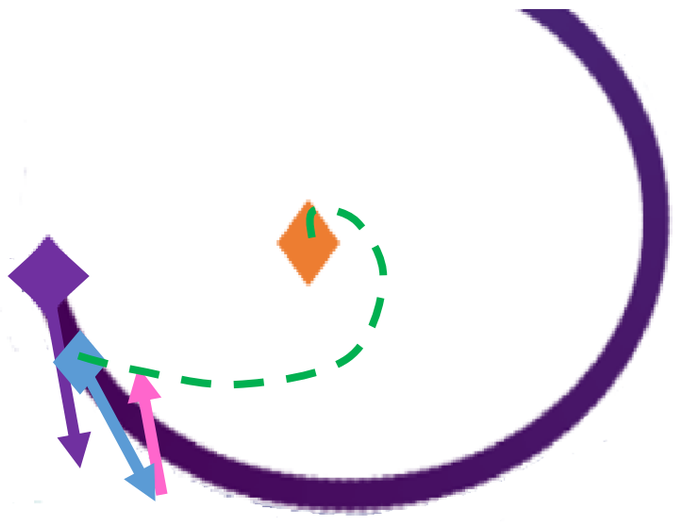
- E.g. $f(x, y) = (x - 0.5) \cdot (y - 0.5)$



$$x_{t+1} = x_t - \eta \cdot \nabla_x f(x_t, y_t)$$
$$y_{t+1} = y_t + \eta \cdot \nabla_y f(x_t, y_t)$$

◆ : start

◆ : min-max equilibrium



$$x_{t+1} = x_t - \eta \cdot \nabla_x f(x_t, y_t) + \eta/2 \cdot \nabla_x f(x_{t-1}, y_{t-1})$$

$$y_{t+1} = y_t + \eta \cdot \nabla_y f(x_t, y_t) - \eta/2 \cdot \nabla_y f(x_{t-1}, y_{t-1})$$

Negative Momentum: convergence

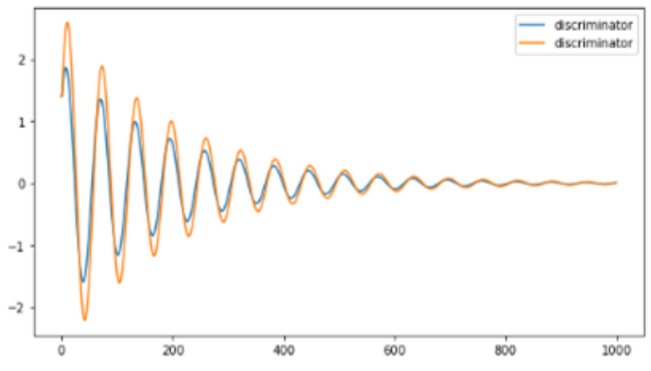
- **Optimistic gradient descent-ascent (OGDA)** dynamics:

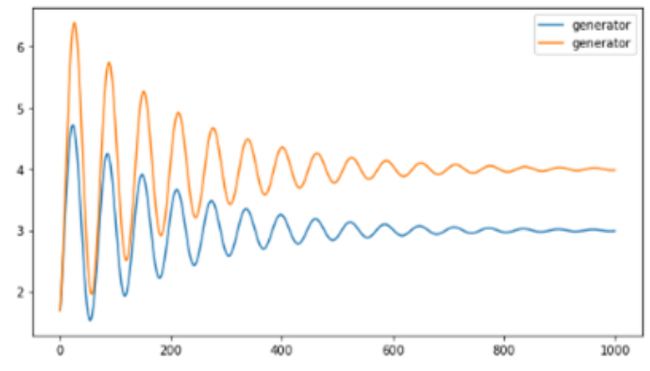
$$\forall t\colon\ x_{t+1} = x_t - \eta \cdot \nabla_x f(x_t, y_t) + \frac{\eta}{2} \cdot \boldsymbol{\nabla_x f(x_{t-1}, y_{t-1})}$$
$$y_{t+1} = y_t + \eta \cdot \nabla_y f(x_t, y_t) - \frac{\eta}{2} \cdot \boldsymbol{\nabla_y f(x_{t-1}, y_{t-1})}$$

- **[Daskalakis-Ilyas-Syrgkanis-Zeng ICLR'18]:** OGDA exhibits last iterate convergence for *unconstrained* bilinear games: $\min_{x\in\mathbb{R}^n} \max_{y\in\mathbb{R}^m} f(x,y) = x^{\mathrm{T}} A y + b^T x + c^T y$

- **[Liang-Stokes'18]:** …convergence rate is geometric if A is well-conditioned, extends to strongly convex-concave functions $f(x,y)$

- E.g. in previous isotropic Gaussian case: $X \sim \mathcal{N}\big((3,4), I_{2\times 2}\big)$, $G_\theta(Z) = \theta + Z$, $D_w(\cdot) = \langle w, \cdot \rangle$





Negative Momentum: convergence

- **Optimistic gradient descent-ascent (OGDA)** dynamics:

$$\forall t\colon\ x_{t+1} = x_t - \eta \cdot \nabla_x f(x_t, y_t) + \frac{\eta}{2} \cdot \nabla_x f(x_{t-1}, y_{t-1})$$
$$y_{t+1} = y_t + \eta \cdot \nabla_y f(x_t, y_t) - \frac{\eta}{2} \cdot \nabla_y f(x_{t-1}, y_{t-1})$$

- **[Daskalakis-Ilyas-Syrgkanis-Zeng ICLR'18]:** OGDA exhibits last iterate convergence for *unconstrained* bilinear games: $\min_{x\in\mathbb{R}^n} \max_{y\in\mathbb{R}^m} f(x,y) = x^{\mathrm{T}} A y + b^T x + c^T y$

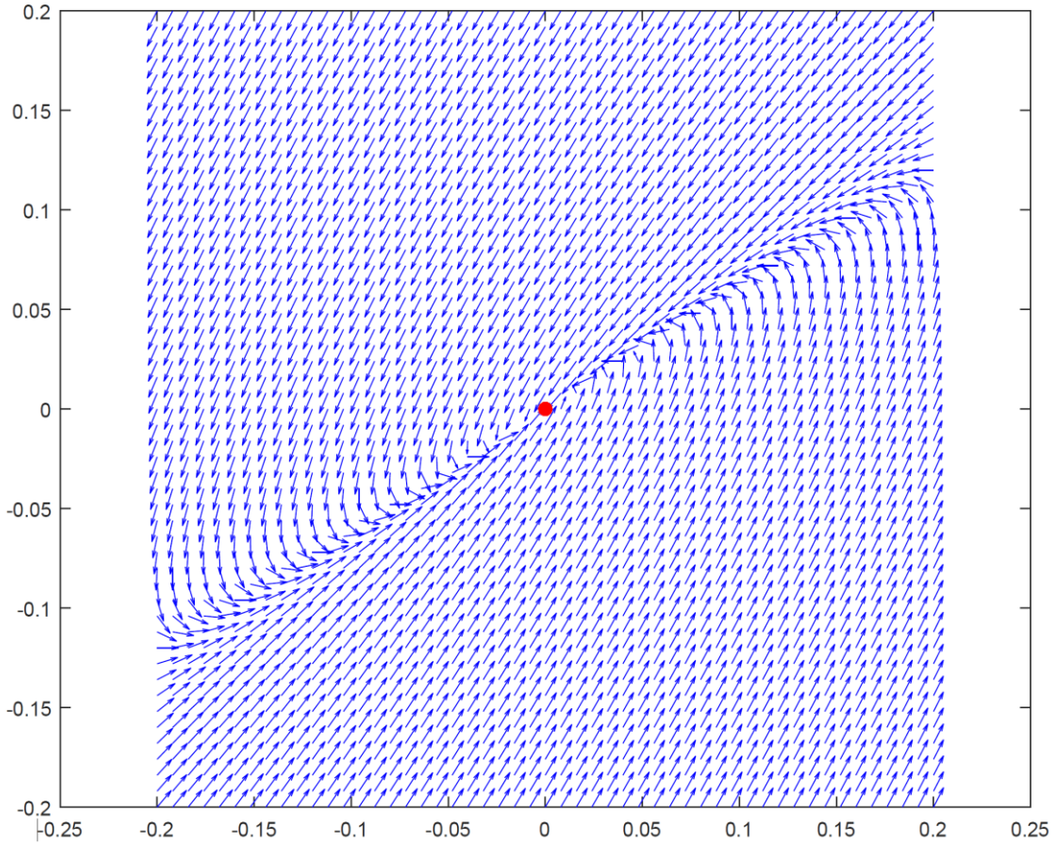
- **[Liang-Stokes'18]:** …convergence rate is geometric if A is well-conditioned, extends to strongly convex-concave functions $f(x,y)$

- **[Daskalakis-Panageas ITCS'18]:** ***Projected*** OGDA exhibits last iterate convergence even for *constrained* bilinear games: $\min_{x\in\Delta_n} \max_{y\in\Delta_m} x^{\mathrm{T}} A y$

  = all linear programming

Negative Momentum: in the Wild

- Can try optimism for non convex-concave min-max objectives $f(x, y)$
- **Issue [Daskalakis, Panageas NeurIPS'18]:** No hope that stable points of **OGDA** or GDA are only local min-max points
- e.g. $f(x, y) = -\frac{1}{8} \cdot x^2 - \frac{1}{2} \cdot y^2 + \frac{6}{10} \cdot x \cdot y$



Gradient Descent-Ascent field

- Nested-ness: Local Min-Max $\subseteq$ Stable Points of GDA $\subseteq$ Stable Points of **OGDA**

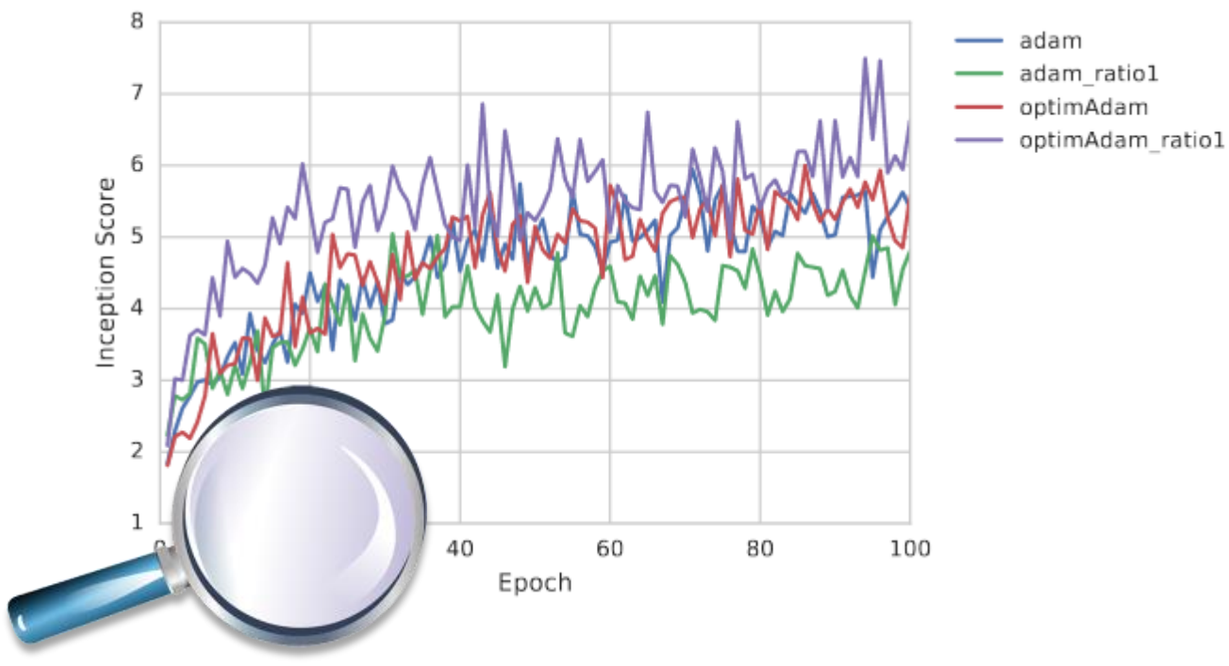
Negative Momentum: in the Wild

- Can try optimism for non convex-concave min-max objectives $f(x, y)$
- **Issue [Daskalakis, Panageas NeurIPS'18]:** No hope that stable points of **OGDA** or GDA are only local min-max points
  - Local Min-Max $\subseteq$ Stable Points of GDA $\subseteq$ Stable Points of **OGDA**
- also **[Adolphs et al. 18]: left inclusion**
- **Question:** identify first-order method converging to local min-max w/ probability 1

- While this is pending, evaluate optimism in practice…
- **[Daskalakis-Ilyas-Syrgkanis-Zeng ICLR'18]:** propose ***optimistic Adam***
  - ***Adam***, a variant of gradient descent proposed by **[Kingma-Ba ICLR'15]**, has found wide adoption in deep learning, although it doesn't always converge **[Reddi-Kale-Kumar ICLR'18]**
  - ***Optimistic Adam*** is the right adaptation of Adam to "undo some of the past gradients"

Optimistic Adam on CIFAR10

- Compare Adam, **Optimistic Adam**, trained on CIFAR10, in terms of Inception Score
- No fine-tuning for **Optimistic Adam**, used same hyper-parameters for both algorithms as suggested in Gulrajani et al. (2017)



Optimistic Adam on CIFAR10

- Compare Adam, **Optimistic Adam**, trained on CIFAR10, in terms of Inception Score
- No fine-tuning for **Optimistic Adam**, used same hyper-parameters for both algorithms as suggested in Gulrajani et al. (2017)

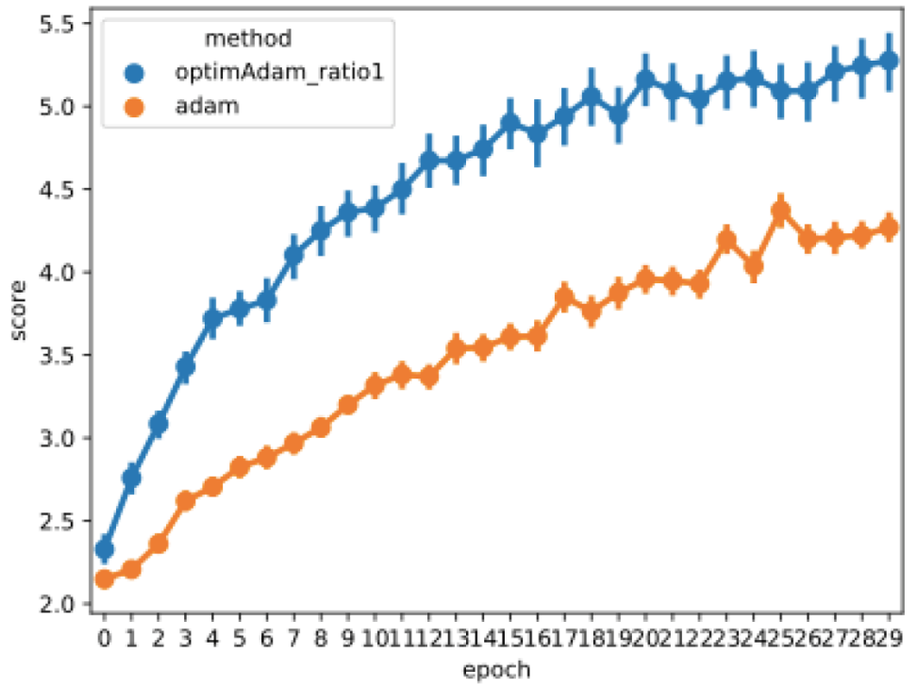


Figure 14: The inception scores across epochs for GANs trained with Optimistic Adam (ratio 1) and Adam (ratio 5) on CIFAR10 (the two top-performing optimizers found in Section 6, with 10%-90% confidence intervals. The GANs were trained for 30 epochs and results gathered across 35 runs.

(b) Sample of images from Generator of Epoch 94, which had the highest inception score.

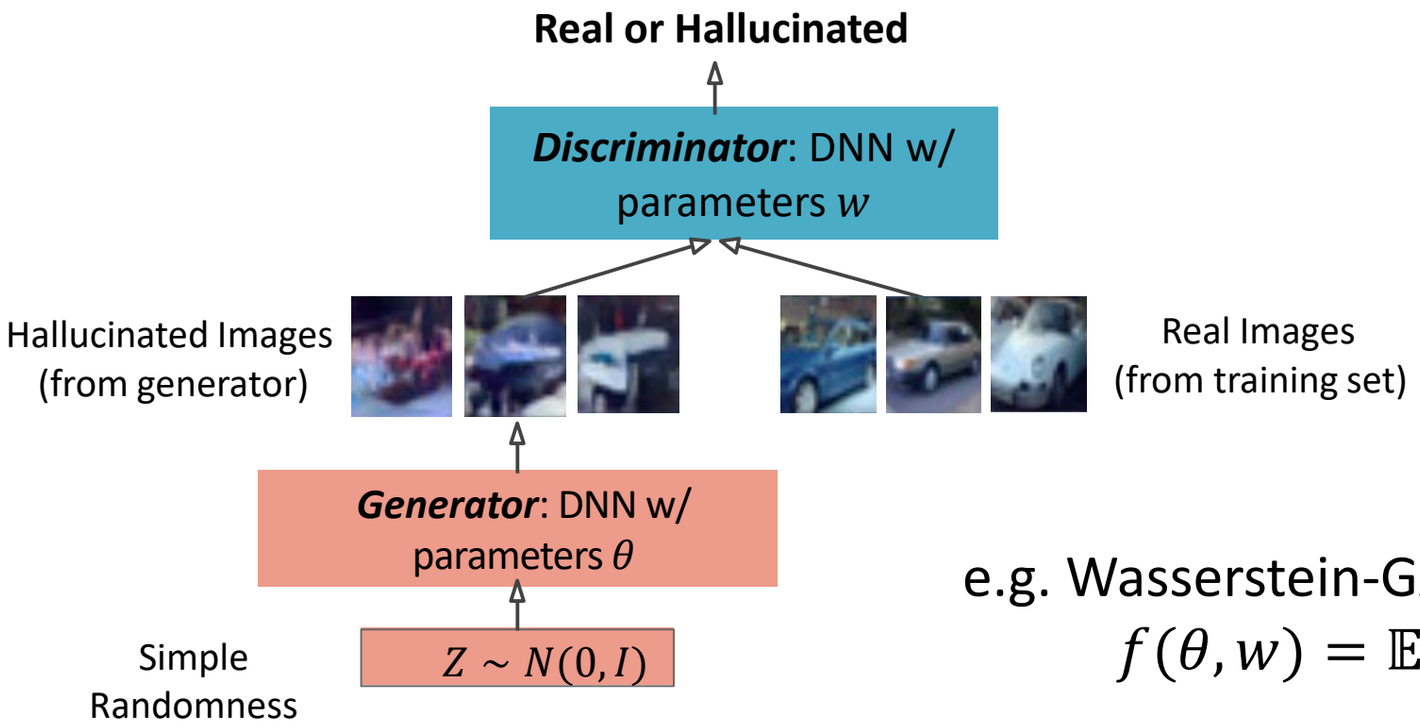
Menu

- **Min-Max Optimization and Adversarial Training**
- **Training Challenges:**
  - **reducing training oscillations**
- Statistical Challenges:
  - reducing sample requirements
  - attaining statistical guarantees

Menu

- **Min-Max Optimization and Adversarial Training**
- **Training Challenges:**
  - **reducing training oscillations**
- **Statistical Challenges:**
  - **reducing sample requirements**
  - **attaining statistical guarantees**

Generative Adversarial Networks (GANs)



$$\inf_{\theta} \sup_{w} \boldsymbol{f(\theta, w)}$$

expresses how well Discriminator distinguishes true from generated images

e.g. Wasserstein-GANs:

$$f(\theta, w) = \mathbb{E}_{X \sim p_{real}}[D_w(X)] - \mathbb{E}_{Z \sim N(0,I)}[D_w(G_\theta(Z))]$$

- **Inner sup (*Discrimination*) problem**: a statistical estimation problem
  - how close is p_{real} and $p_{generated}$ in distance defined by test functions expressible in the architecture of the discriminator?
  - because training will fail to solve min-max problem to optimality, this distance won't be truly minimized
- **major statistical challenges:**
  - Certifying a trained GAN: how close is p_{real} and $p_{generated}$ in some distance of interest?
  - Alleviating computational & statistical burden of discrimination
  - Scaling up the dimensionality of generated distributions

GANs: Statistical Challenges

- **Certifying a trained GAN:** how close is p_{real} and $p_{generated}$ in some distance of interest?
- Fundamental Challenge: curse of dimensionality
  - **claim (birthday paradox):** given sample access to dist'n P over $\{0,1\}^n$, and Q=Unif $(\{0,1\}^n)$, estimating Wasserstein(P, Q) to within $\pm 1/4$ requires $\Omega\left(2^{n/2}\right)$ samples
  - for n=1000's (e.g. CIFAR)

    ⇝ infeasible, unless *lower-dimensional structure* in p_{real} and $p_{generated}$ is exploited

- **Alleviating Computational & Statistical Burden of Discriminator:**

  ⇝ infeasible, unless *lower-dimensional structure* in p_{real} and $p_{generated}$ is exploited

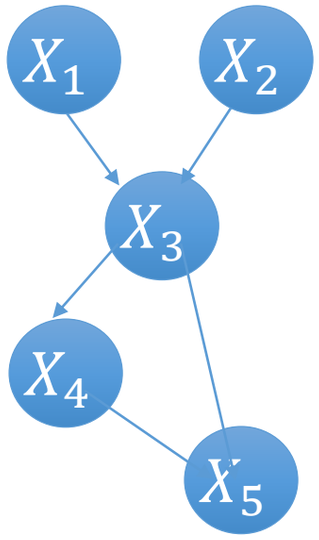
- **Scaling-up Dimensionality of Generated Distribution (e.g. video generation):**

  ⇝ infeasible, unless *lower-dimensional structure* in p_{real} is exploited

Lower-Dimensional Structure: Bayesian Networks

- Probability distribution defined in terms of a DAG $G = (V, E)$
- Node v associated w/ random variable $X_v \in \Sigma$
- Distribution factorizable in terms of parenthood relationships

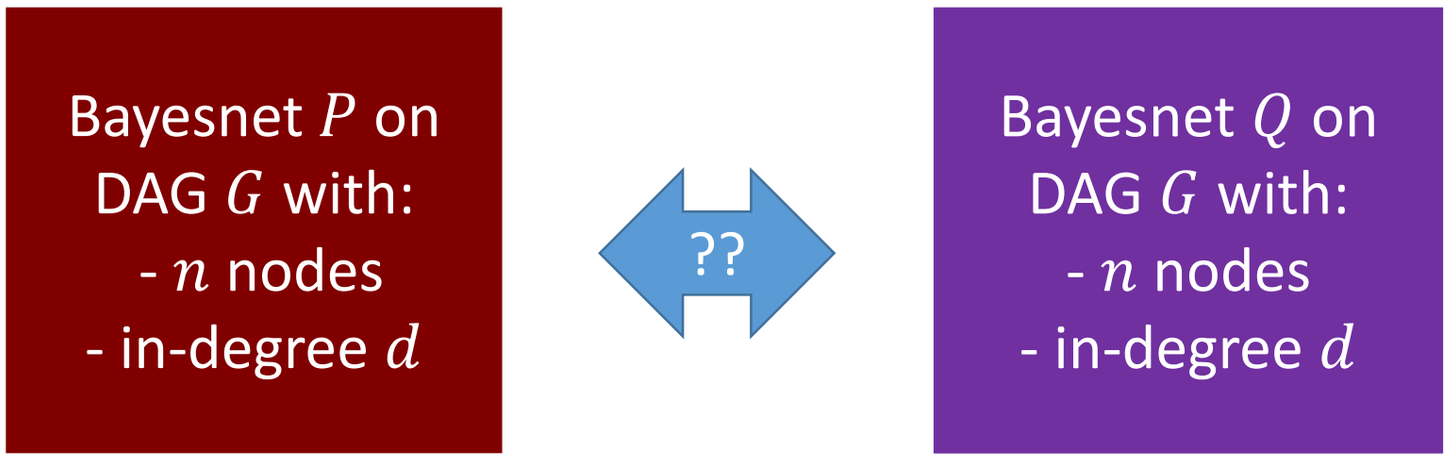
$$\Pr(x) = \prod_v \Pr_{X_v | X_{\Pi_v}}\left(x_v | x_{\Pi_v}\right), \forall x \in \Sigma^V$$



$$\Pr[\vec{x}] = \Pr[\mathrm{x}_1] \cdot \Pr[x_2] \cdot \Pr[x_3 | x_1, x_2] \cdot \Pr[x_4 | x_3] \cdot \Pr[x_5 | x_3, x_4]$$

Is it easier to discriminate between Bayes-nets whose structure is known?

BayesNet Discrimination



Goal: Given samples from P, Q and ε, distinguish: $P = Q$ vs $dist(P, Q) > \varepsilon$

[Daskalakis-Pan COLT'17]: If $dist$ is Total Variation distance, there exist computationally efficient testers using $\tilde{O}\left(\frac{|\Sigma|^{0.75\,(d+1)}n}{\varepsilon^2}\right)$ samples.

Moreover, the dependence on n, ε of both bounds is tight up to a $O(\log n)$ factor, and the exponential in d dependence is necessary and essentially tight.

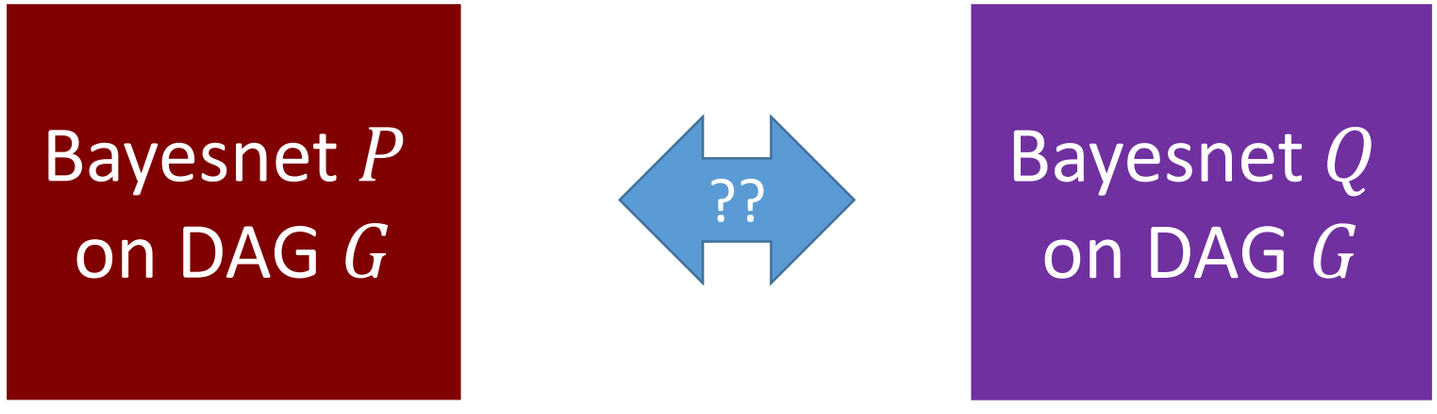
[Canonne et al. COLT'17]: Identify conditions under which dependence on n can be made $\sqrt{n}$ when one of the two Bayesnets is known

Effective dimensionality is: ~~n~~ d

BayesNet Discrimination in TV

- **Goal:** distinguish $P = Q$ vs $d_{TV}(P, Q) > \varepsilon$
- **Idea:** distance localization
- prove statement of the form: "If BayesNets P and Q are far in TV, there exists a small size witness set S of variables such that P_S and Q_S, the ***marginals*** of P and Q on variables S, are also somewhat far away"
  - reduces the original problem to identity testing on small size sets whose distributions ***can be sampled***
- **Question:** which distances are localizable?
  - $KL(P||Q) \leq \sum_v KL\left(P_{v\cup\Pi_v}||Q_{v\cup\Pi_v}\right)$ (chain rule of KL)
  - $d_{\mathrm{TV}}(P, Q) \leq \sum_v d_{\mathrm{TV}}\left(P_{v\cup\Pi_v}, Q_{v\cup\Pi_v}\right) + \sum_v d_{\mathrm{TV}}\left(P_{\Pi_v}, Q_{\Pi_v}\right)$ (hybrid argument)
  - $H^2(P, Q) \leq \sum_v H^2\left(P_{v\cup\Pi_v}, Q_{v\cup\Pi_v}\right)$

Wasserstein Subadditivity



Q: Does Wasserstein satisfy subadditivity

$$\text{Wass}(P,Q) \leq \sum_{v} \text{Wass}\left(P_{v\cup\Pi_v} || Q_{v\cup\Pi_v}\right) \quad ?$$

A: Not always; exist pair of Markov Chains: $X \to Y \to Z$ and $X' \to Y' \to Z'$ such that

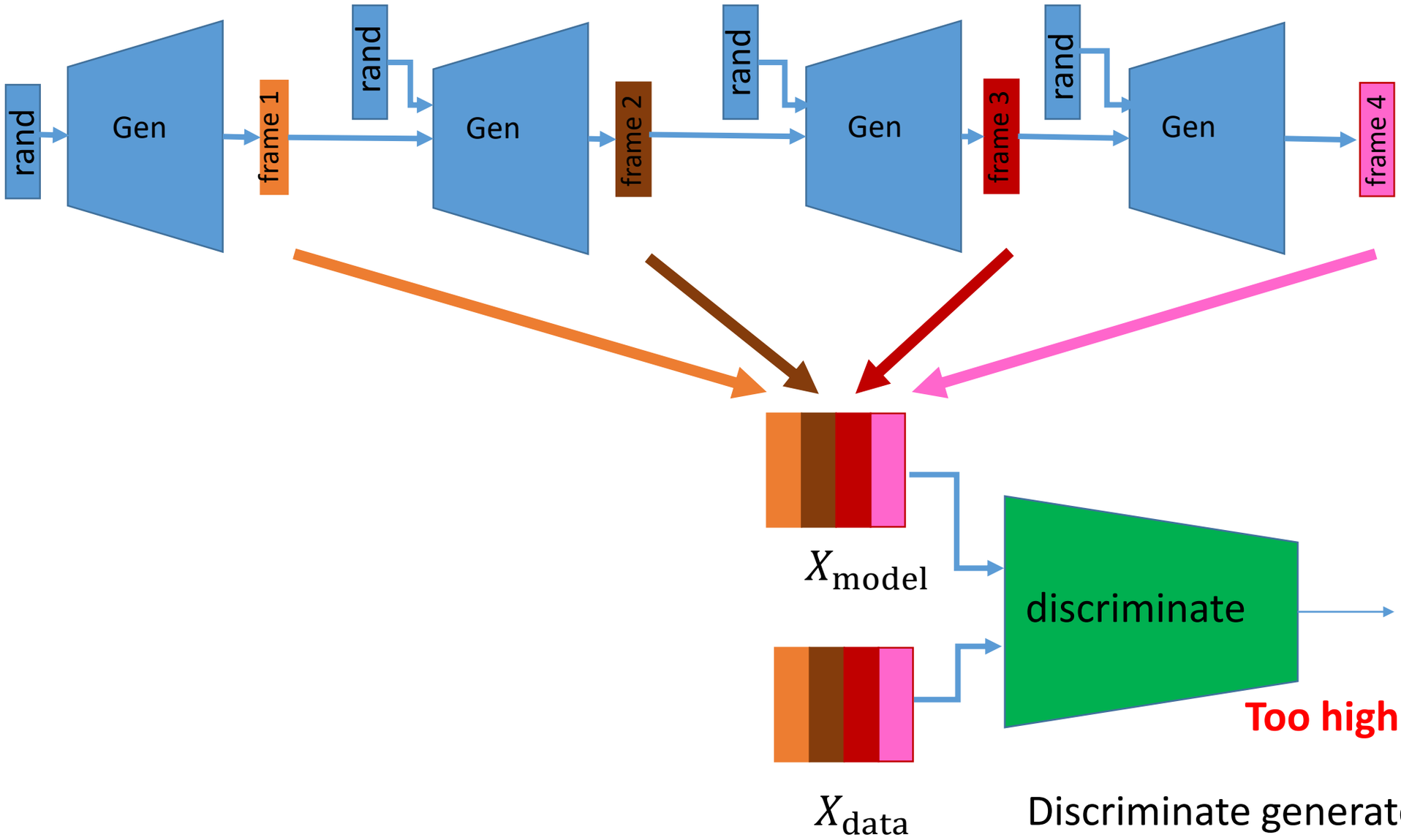
$$\frac{\text{Wass}\big((X,Y),(X',Y')\big) + \text{Wass}\big((Y,Z),(Y',Z')\big)}{\text{Wass}\big((X,Y,Z),(X',Y',Z')\big)}$$

can be made arbitrarily small.

[Preliminary Result]: Wasserstein distance between two Markov Chains $X_1, \dots, X_T$ and $Y_1, \dots, Y_T$ satisfies subbadditivity if the conditional densities $f_{\boldsymbol{X}}(x_t|x_{t-1})$ and $f_{\boldsymbol{Y}}(y_t|y_{t-1})$ are Lipschitz wrt x_{t-1} and y_{t-1} respectively, for all t.

(extends to Bayesian Networks)

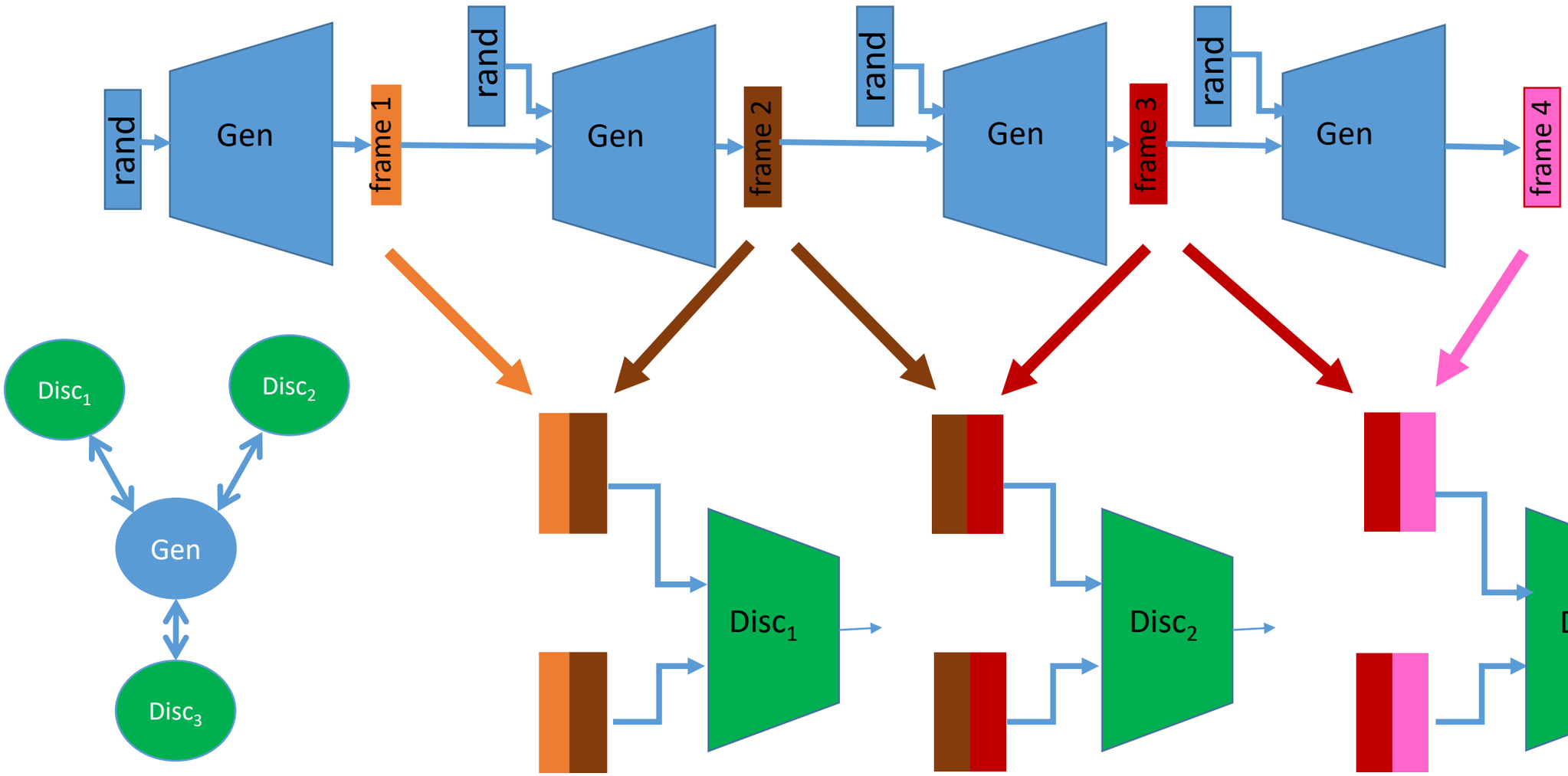
Video Generation



Too high dimensional

Discriminate generated video against target distribution over videos

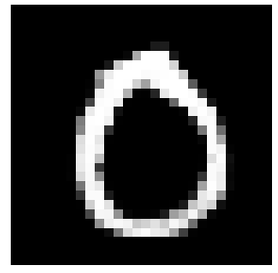
Video Generation



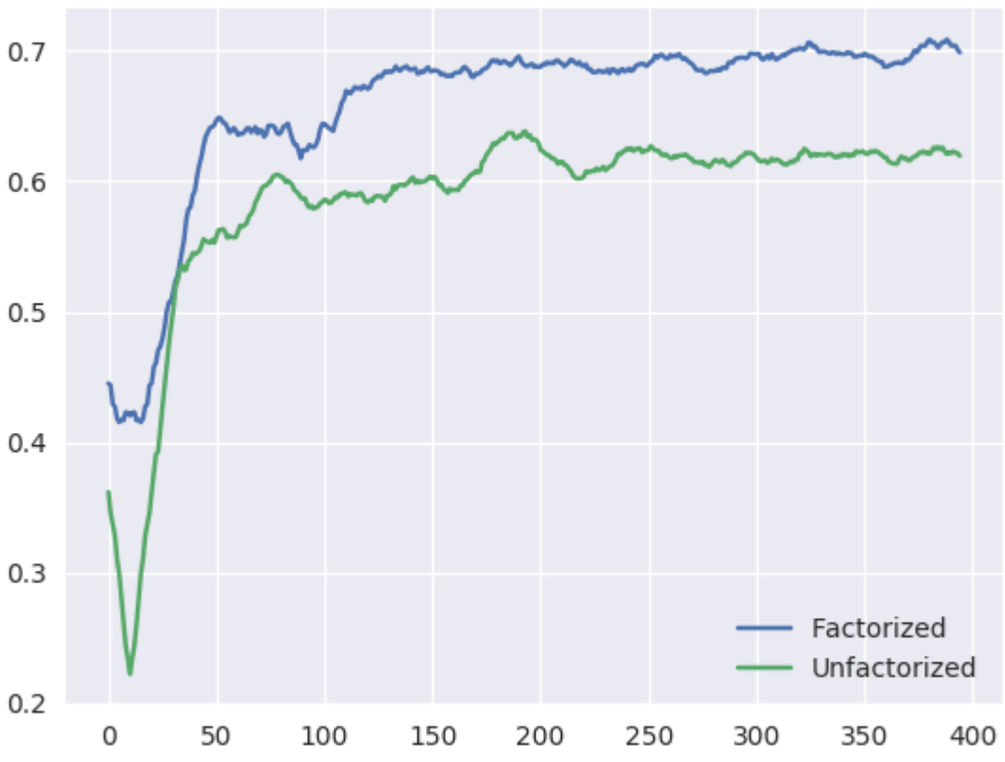
can **exploit subadditivity** and **discriminate only pairs of consecutive frames** of generated distribution against pairs of consecutive frames of target distribution

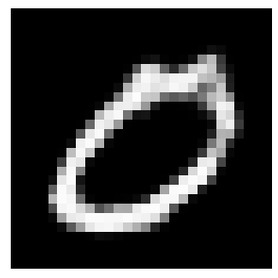
N.B. resulting multi-player zero-sum game falls in realm of **[D-Papadimitriou ICALP'09], [Even-Dar et al STOC'09], [Cai-D SODA'11], [Cai et al MATHOR'15];** efficient dynamics known

Video Generation: experiment [Ilyas'18]



- Created random 4-frame videos of MNIST digits
  - in every training video, digits are weakly increasing in time
- Trained two video GANs:
  - a GAN w/ an un-factorized discriminator
  - and a GAN w/ a factorized discriminator
- GANs must learn both how to hallucinate handwritten digits, and that they need to put them in increasing order
- Compare factorized vs unfactorized models in terms of accuracy





Conclusions

- Min-Max Optimization has found numerous applications in Optimization, Game Theory, Adversarial Training
- Applications to Generative Adversarial Networks pose serious challenges, of optimization (oscillations) and statistical (curse of dimensionality) nature
- We propose gradient descent with *negative momentum* as an approach to ease training oscillations
- We prove Wasserstein subadditivity in Bayesnets and propose modeling dependencies in the data as an approach to ease the curse of dimensionality
- Lots of interesting theoretical and practical challenges going forward

Thanks!

The First Auction by Christie’s