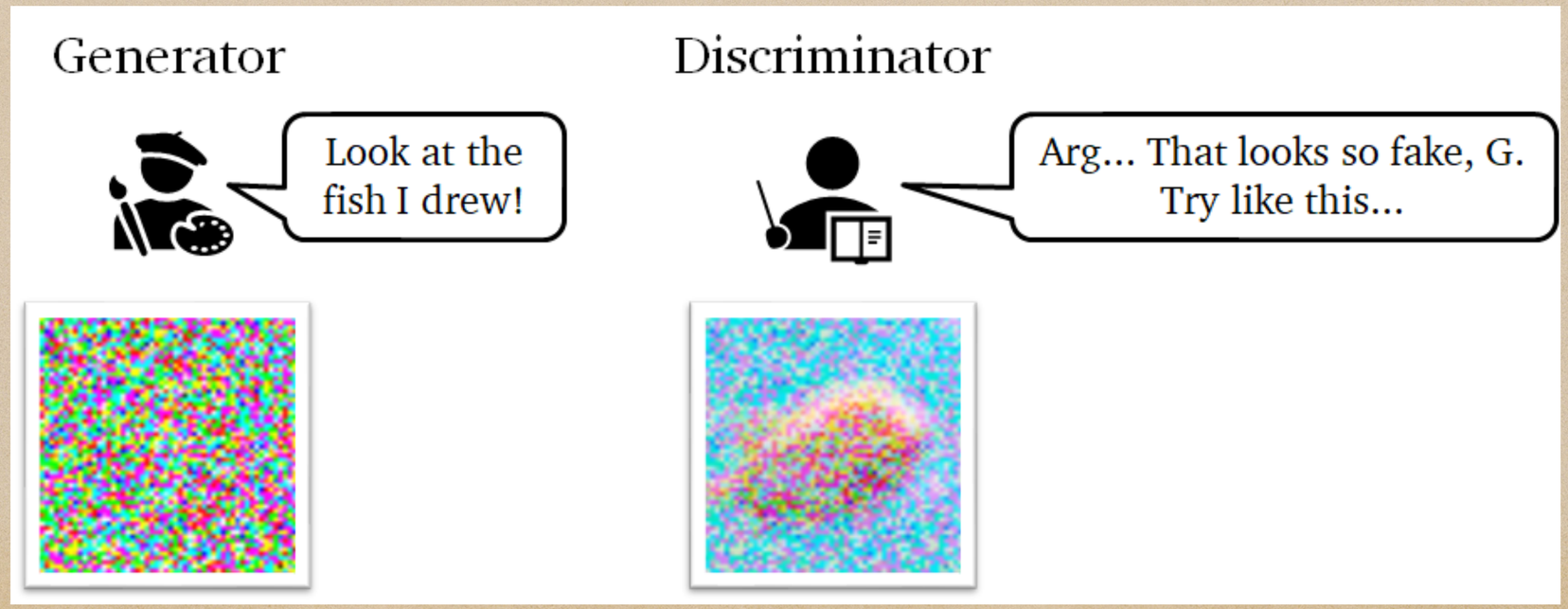
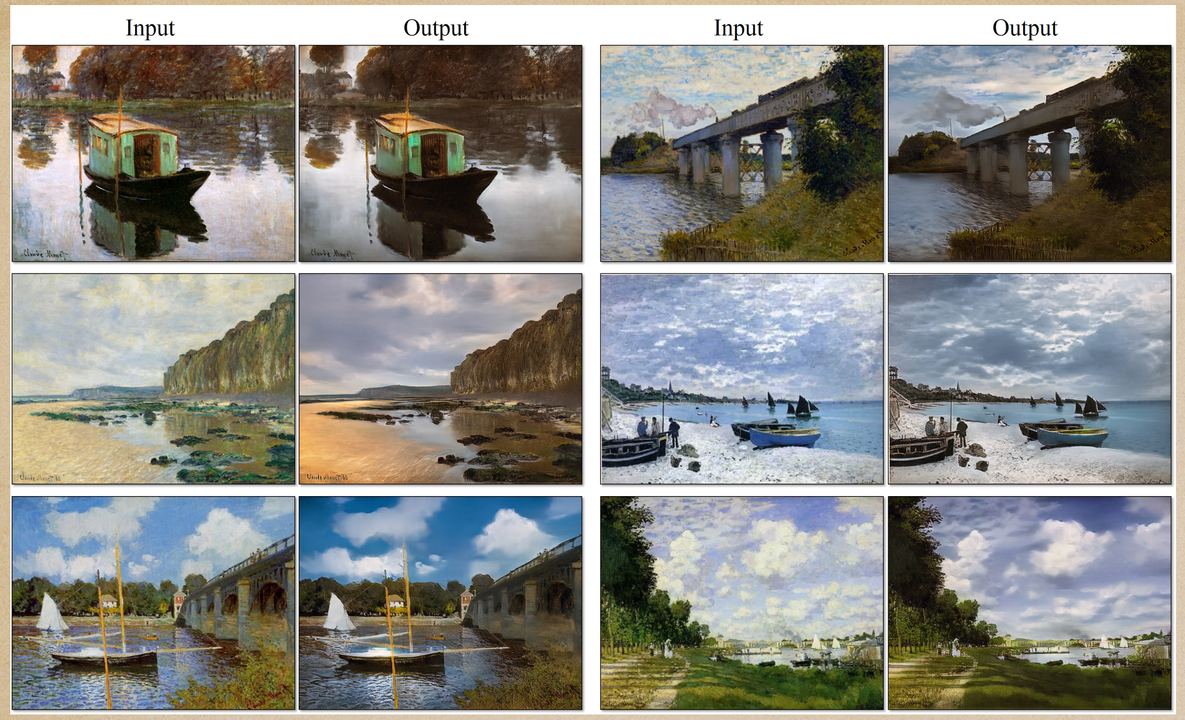
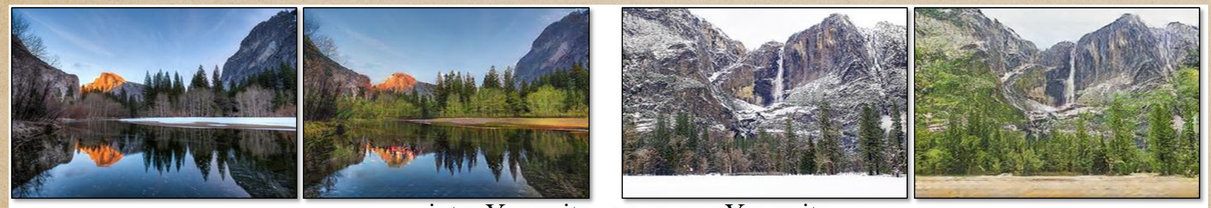
Generative Adversarial Network

Learns one-layer neural net via SGD

- Vasilis Kontonis, Sihan Liu, Christos Tzamos







winter Yosemite → summer Yosemite

summer Yosemite → winter Yosemite

Theory is when you know everything but nothing works.

Practice is when everything works but no one knows why.

In our lab, theory and practice are combined: nothing works and nobody knows why.

Simpler Setting: One layer Network

- Some notations

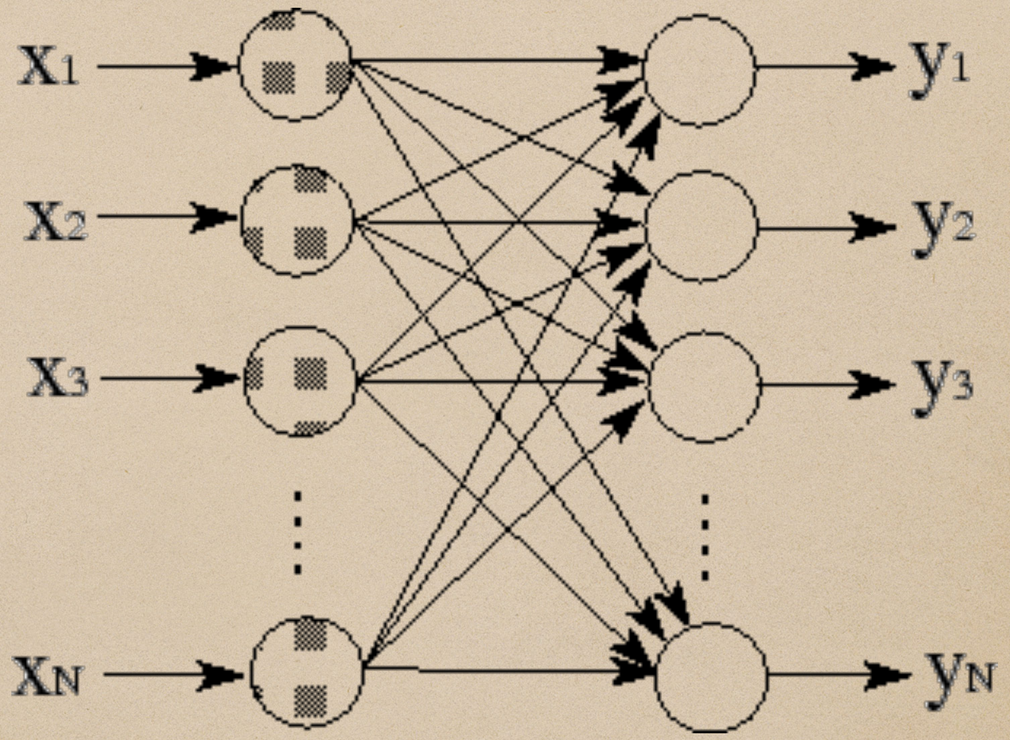
$\mathbf{W} : d \times d$ (Unknown)

$\phi : \mathbb{R}^d \to \mathbb{R}^d$ (Known)

$\mathbf{x} \sim \mathcal{N}(\mathbf{I^d})$ (Gaussian Noise)

$\mathbf{y} = \phi(\mathbf{W}\mathbf{x})$ (Output Sample)

$\mathbf{y} \sim p(\mathbf{W}, \phi)$ (Target Distribution)



Require ϕ to be partially invertible Included: tanh, sigmoid, ReLU …

Main result: Convergence via SGD

- Nested Stochastic Gradient Ascent-Descent uses $\tilde{O}(d^2/\epsilon^2)$ samples from the target network $(\mathbf{W}^*, \phi)$, performs $\tilde{O}(d^4/\epsilon^6)$ gradient updates, and converges to a matrix $\widetilde{\mathbf{W}}$ such that $d_{\mathrm{TV}}\left(p(\mathbf{W}_*, \phi), p(\widetilde{\mathbf{W}}, \phi)\right) \leq \epsilon$ with probability at least 99%

Discriminator Design

- What is the optimal Discriminator?
- $D(x) = \dfrac{p_d(x)}{p_d(x) + p_g(x)}$
- However, can be arbitrarily complex …

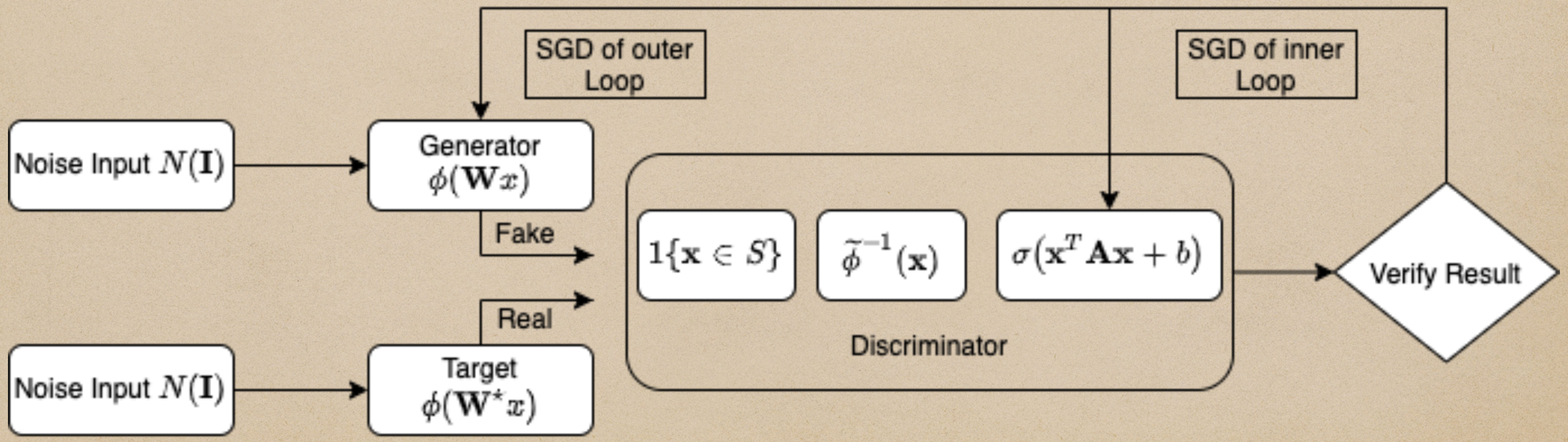
  i.e $\phi := ReLU(x)$

$$\frac{\int_{\mathbf{z} \in \mathcal{V}_{\mathbf{x}}} \mathcal{N}(\mathbf{z}; \mathbf{W}_*) d\mathbf{z}}{\int_{\mathbf{z} \in \mathcal{V}_{\mathbf{x}}} \mathcal{N}(\mathbf{z}; \mathbf{W}) d\mathbf{z} + \int_{\mathbf{z} \in \mathcal{V}_{\mathbf{x}}} \mathcal{N}(\mathbf{z}; \mathbf{W}_*) d\mathbf{z}},$$

where $\mathcal{V}_{\mathbf{x}} := \{\mathbf{z}_i = \mathbf{x}_i \text{ if } \mathbf{x}_i > 0 \text{ else } \mathbf{z}_i \leq 0\}$

What about invertible functions?

$$D_*(\mathbf{x};\mathbf{W}) = \frac{\mathcal{N}(\phi^{-1}(\mathbf{x});\mathbf{W}^*)}{\mathcal{N}(\phi^{-1}(\mathbf{x});\mathbf{W}^*) + \mathcal{N}(\phi^{-1}(\mathbf{x});\mathbf{W})} = \sigma\left(\phi^{-1}(\mathbf{x})^T\mathbf{A}\phi^{-1}(\mathbf{x}) + b\right)$$



Algorithm Sketch

Algorithm 1 Nested Stochastic Gradient Descent Ascent on Standard GAN

Set $k = \widetilde{O}(d^2/\epsilon^2)$
Initialize $\boldsymbol{W} = \boldsymbol{I}$.
Sample $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(k)}$ from $p(\boldsymbol{W}_*, \phi)$
for $i = 1$ to $\widetilde{O}(d^2/\epsilon^4)$ **do**

  ▷ Discriminator Training
  Sample $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(k)}$ from $p(\boldsymbol{W}, \phi)$
  for $j = 1$ to k **do**
    Update $\boldsymbol{A}$ and b with stochastic gradient

$$\nabla_{\boldsymbol{A},b}\big[\log(D(\boldsymbol{x}^{(j)}; \boldsymbol{A}, b)) + \log(1 - D(\boldsymbol{y}^{(j)}; \boldsymbol{A}, b))\big]$$

  end for

  ▷ Generator Training
  Sample $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{I})$
  Update $\boldsymbol{W}$ with stochastic gradient

$$\nabla_{\boldsymbol{W}} \log(D(\phi(\boldsymbol{W}\boldsymbol{z}); \boldsymbol{A}, b) \tag{3}$$

end for

Discriminator Training

- Can train Discriminator to be almost optimal with $O(d^2/\epsilon^2)$ iterations
  - Strong concavity
  - Sub-exponential gradient oracle
  - Convergence Guarantee from standard convex-optimization argument

Generator Training

- Training Gradients from $L_G(\mathbf{W}; D) = E_{\mathbf{x}\sim\mathcal{N}(\mathbf{I})} \log(1 - D(\phi(\mathbf{W}\mathbf{x})))$
- Envelop Theorem: given $V(t) = \max_{x\in\mathcal{X}} f(x,t)$, $V'(t) = \frac{\partial f}{\partial t}(x_*, t)$
- Virtual Training Criteria: training optimal discriminator parameter

$$V(\mathbf{W}) = \max_{\theta} E_{\mathbf{x}\sim\mathcal{N}(\mathbf{I})} \log\left(1 - D_\theta\left(\phi(\mathbf{W}\mathbf{x})\right)\right) + E_{\mathbf{x}\sim\mathcal{N}(\mathbf{I})} \log\left(D_\theta(\phi(\mathbf{W}^*\mathbf{x})\right)$$

$$\implies \nabla_{\mathbf{W}} V(\mathbf{W}) = \nabla_{\mathbf{W}} L_G(\mathbf{W}, D_{\theta_*})$$

Markov Inequality

Given a random variable X, it holds $Pr\left[|X| \geq a\right] \leq \dfrac{E\left[|X|\right]}{a}$

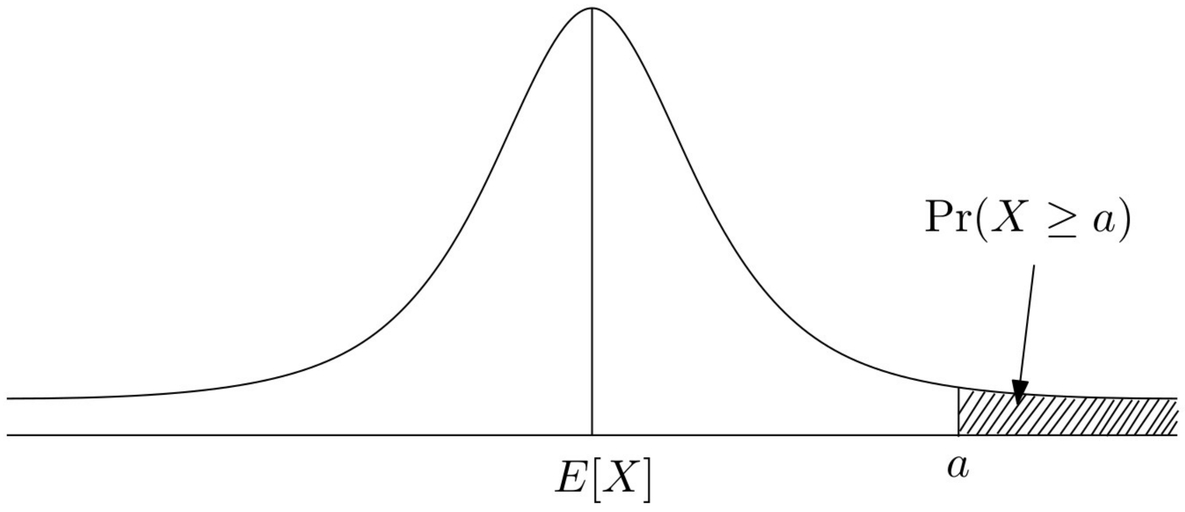


Figure : Markov's Inequality bounds the probability of the shaded region.

Anti-concentration of Polynomials

Let $k, \gamma \in \mathbb{R}+$, $\mathbf{m} \in \mathbb{R}^d$, $\mathbf{\Sigma} \in \mathbb{R}^{d \times d}$ such that $\mathbf{\Sigma}$ is positive semidefinite and $p : \mathbb{R}^d \to \mathbb{R}$ be a multivariate polynomial of degree at most ℓ,

we define $\bar{Q} = \left\{ \mathbf{x} \in \mathbb{R}^d \middle| \, |p(\mathbf{x})| \leq \gamma \right\}$,

then there exists an absolute constant C such that

$$\mathcal{N}(\bar{Q}; \mathbf{\Sigma}^{1/2}) \leq \frac{Ck\gamma^{1/\ell}}{\left(E_{\mathbf{z} \sim \mathcal{N}(\mathbf{\Sigma}^{1/2})} \left[p(\mathbf{z})^{k/\ell} \right] \right)^{1/k}}$$

Virtual Training Criteria

- Non-convex optimization
- Yet, first order stationary points are good

$$\left\| \mathbf{W}\mathbf{W}^T - \mathbf{W}^\star\mathbf{W}^{\star T} \right\|_F \cdot \left(\min_{\mathbf{z} s.t \|\mathbf{z}\|_2 = 1} E_{\mathbf{x}\sim\mathcal{N}(\mathbf{I})} \left[f(p(\mathbf{x})) \cdot \mathbf{z}^T \mathbf{q}(\mathbf{x}) \mathbf{z} \right] \right)$$

$f : \mathbb{R} \mapsto \mathbb{R}$, monotonicity

$p : \mathbb{R}^d \mapsto \mathbb{R}$, concentration around expectation

$q : \mathbb{R}^d \mapsto \mathbb{R}^{d\times d}$ (P.S.D), anti-concentration under Gaussian measure

Except …

- Discriminator is not fully optimized
- Bias in training gradients used by Generator. i.e

$$\left\| E\left[\mathbf{g}^{(t)} \,|\, \mathbf{g}^{(t-1)} \ldots \mathbf{g}^{(0)}\right] - \nabla f(\mathbf{x}) \right\|_2 \leq \alpha$$

- Nevertheless, converge to $(\epsilon + \sqrt{\alpha})$-stationary points, where α is the bound on "bias". (Biased Stochastic Gradient Descent)

Bias Control

- Training Gradient is Lipschitz-continuous with respect to Discriminator parameter θ

$$\| \nabla_{\mathbf{W}} L_G(\mathbf{W}; D_{\tilde{\theta}}) - \nabla_{\mathbf{W}} V(\mathbf{W}) \| = \| \nabla_{\mathbf{W}} L_G(\mathbf{W}; D_{\tilde{\theta}}) - \nabla_{\mathbf{W}} L_G(\mathbf{W}; D_{\theta_*}) \|$$

$$\leq c \cdot \| \tilde{\theta} - \theta_* \|$$

- Strong concavity: $\| \tilde{\theta} - \theta_* \|$ is small in each iteration with high probability

"Practicing (learning) without analyzing (reflection) is a waste;
Analyzing without practicing is dangerous"

–Confucius ("No, I didn't say that …")