Title: PSI 2018/2019 - Machine Learning - Lecture 15

Speakers: Michael Albergo

Collection: PSI 2018/2019 - Machine Learning (Hayward Sierens)

Date: April 12, 2019 - 9:00 AM

URL: http://pirsa.org/19040010

Yesterday

Explicit Likelihood

Today

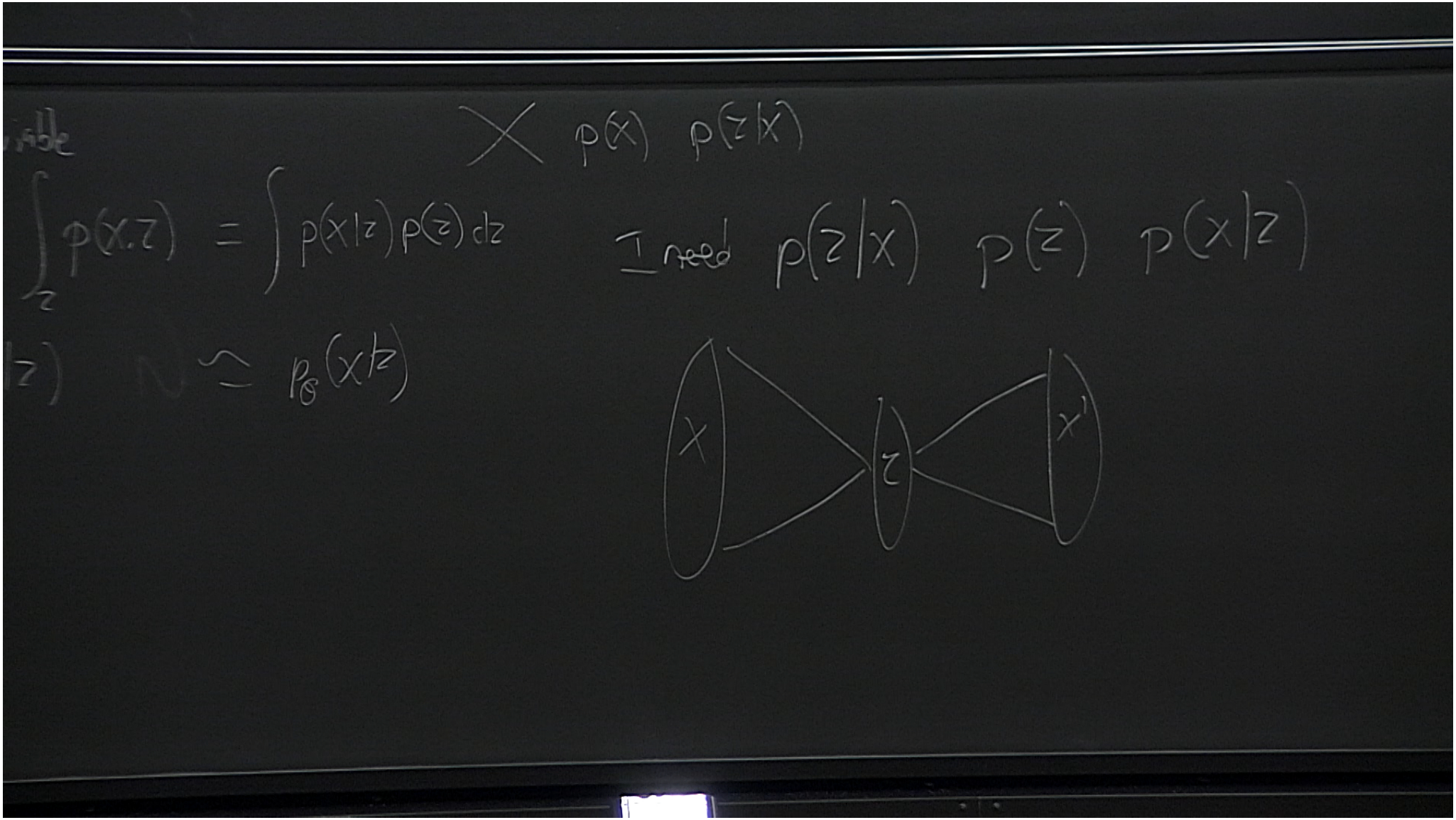
2. Explicit but Approx.

3. Implicit Likelihood

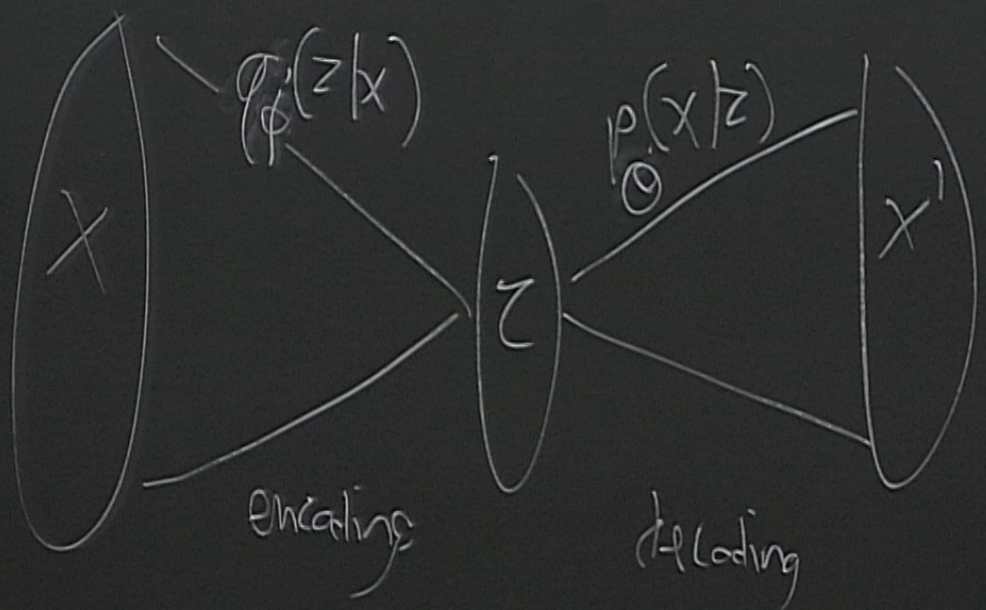
2. Latent Variable

$$p(x) = \int_z p(x,z) = \int p(x|z)\,p(z)\,dz$$

Approx. $p(x|z) \simeq p_\theta(x|z)$


p(x) p(z|x)
p(x,z) = ∫ p(x|z) p(z) dz
I need p(z|x) p(z) p(x|z)
≃ p_θ(x|z)
x
z
x'

I need $p(z|x)$ $p(z)$ $p(x|z)$



$$\ln p(x) = E_{z \sim q_\phi(z|x)}\left[\ln p(x)\right]$$

$$= E_{z \sim q_\phi(z|x)}\left[\ln \frac{p_\theta(x|z)p(z)}{p_\theta(z|x)} \frac{q_\phi(z|x)}{q_\phi(z|x)}\right]$$

$$= E_{z \sim q}\left[\ln p(x|z)\right] - E_{z \sim q}\left[\ln \frac{q_\phi(z|x)}{p_\theta(z)}\right] + E_{z \sim q}\left[\ln \frac{q_\phi(z|x)}{p_\theta(z|x)}\right]$$

$$\ln p(x) = E_{z\sim q_\phi(z|x)}\left[\ln p(x)\right]$$

$$= E_{z\sim q_\phi(z|x)}\left[\ln \frac{p_\theta(x|z)p(z)}{p_\theta(z|x)} \frac{q_\phi(z|x)}{q_\phi(z|x)}\right]$$

$$= E_{z\sim q}\left[\ln p(x|z)\right] - E_{z\sim q}\left[\ln \frac{q_\phi(z|x)}{p_\theta(z)}\right] + E_{z\sim q}[\ln$$

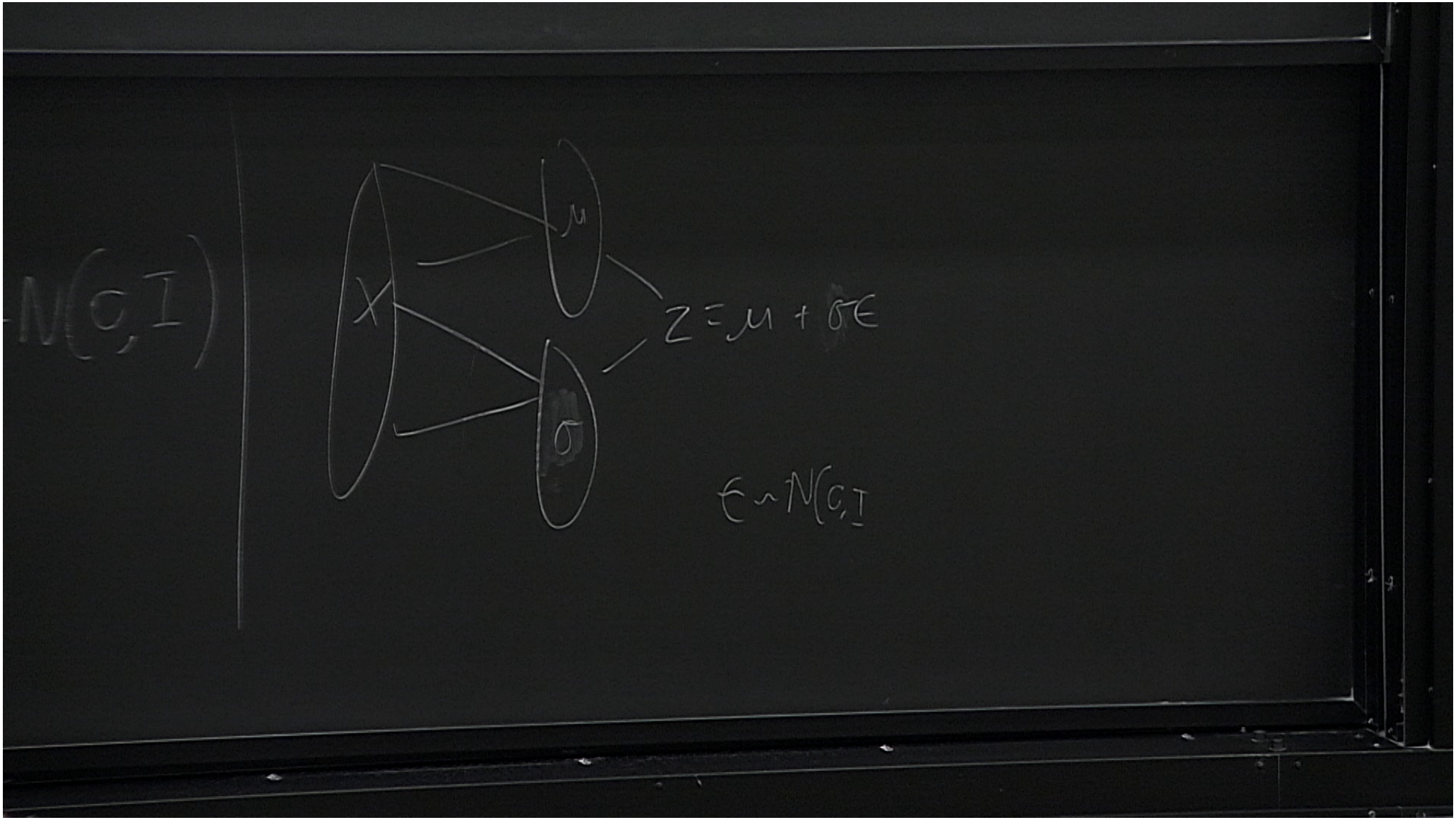
$$\ln p(x) \geq \; \sim\; - D_{KL}\left(q_\phi(z|x) \,\|\, p_\theta(z)\right) + \cancel{D_{KL}}($$

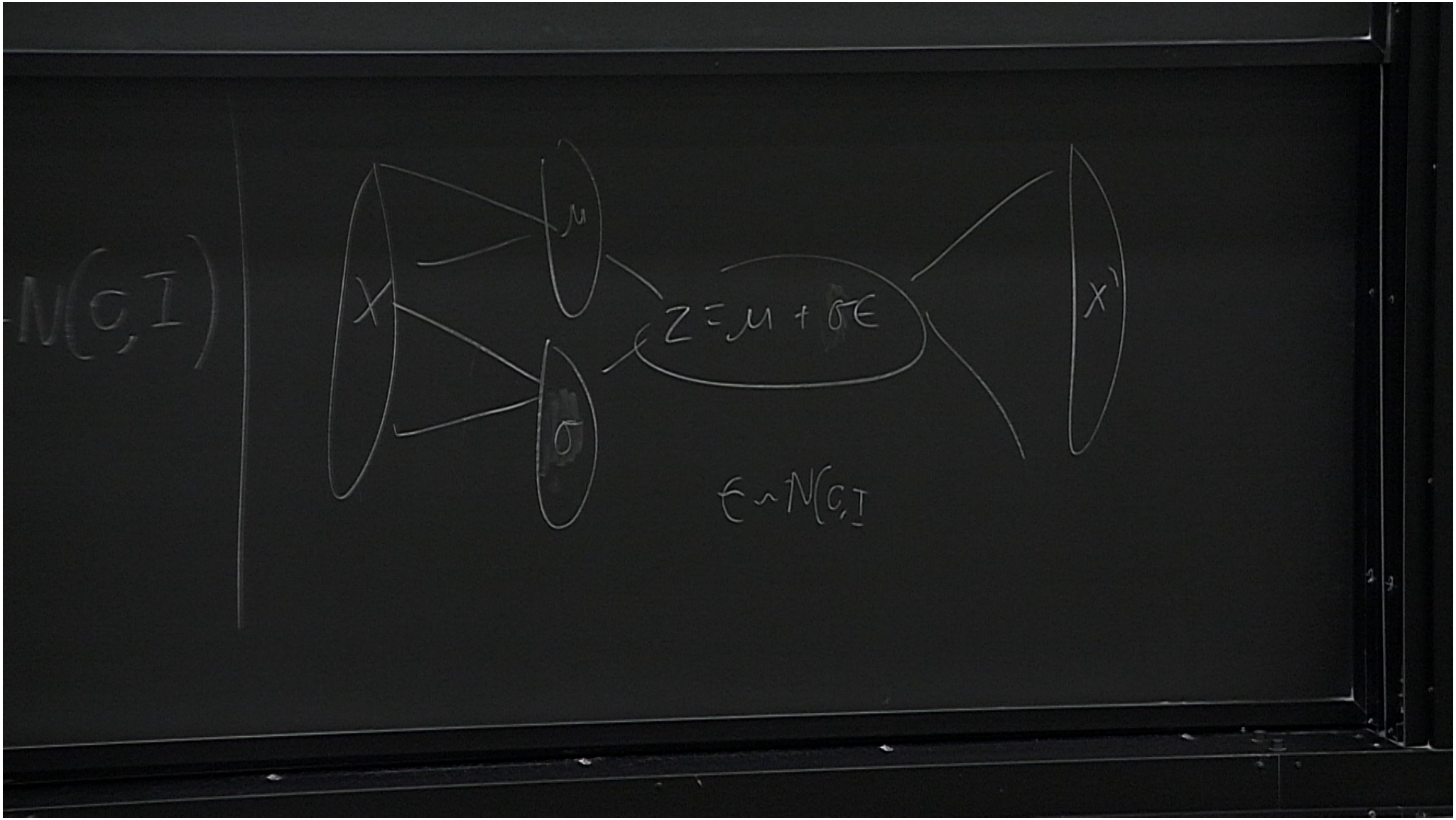
$$\frac{q_\phi(z|x)}{q_\phi(z|x)}\Big]$$

$$p_\theta(z) \sim N(0, I)$$

$$-E_{z\sim q}\left[\ln \frac{q_\phi(z|x)}{p_\theta(z)}\right] + E_{z\sim q}\left[\ln \frac{q_\phi(z|x)}{p_\theta(z|x)}\right]$$

$$-D_{KL}(q_\phi(z|x) \| p_\theta(z)) + D_{KL}(q_\phi(z|x) \| p_\theta(z|x)) \geq 0$$


N(0,I)
X
μ
σ
Z = μ + σε
ε ~ N(0,I)


N(0,I)
x
μ
σ
z = μ + σε
x'
ε ~ N(0,I)

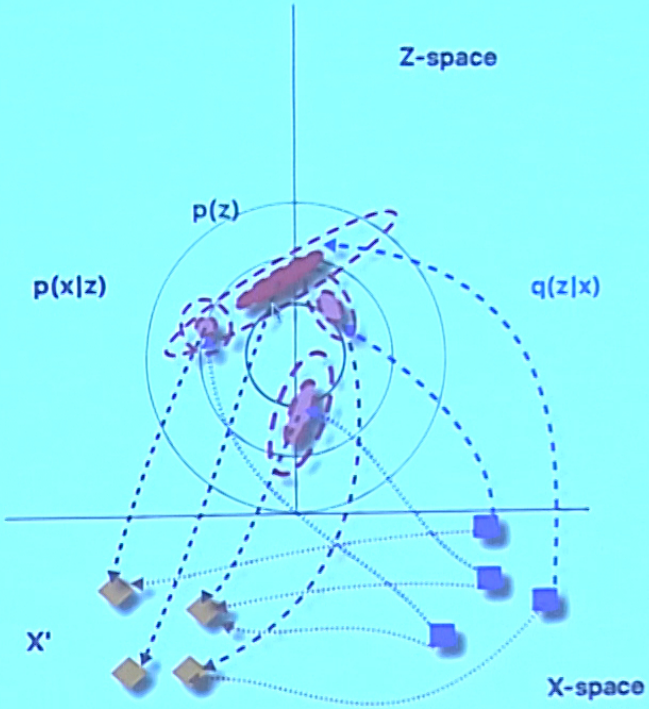


Figure 5: Visualizing how datapoints $\mathbf{x}$ are mapped into latent space, and potential redundancy in how they are decoded.

As the KL term is more strongly enforced, data will be better spread around the Gaussian prior, but the generated image quality will be worse. If the KL term is not enforced as strongly (as might be seen above), posterior Gaussians can overlap, and multiple inputs might map to the

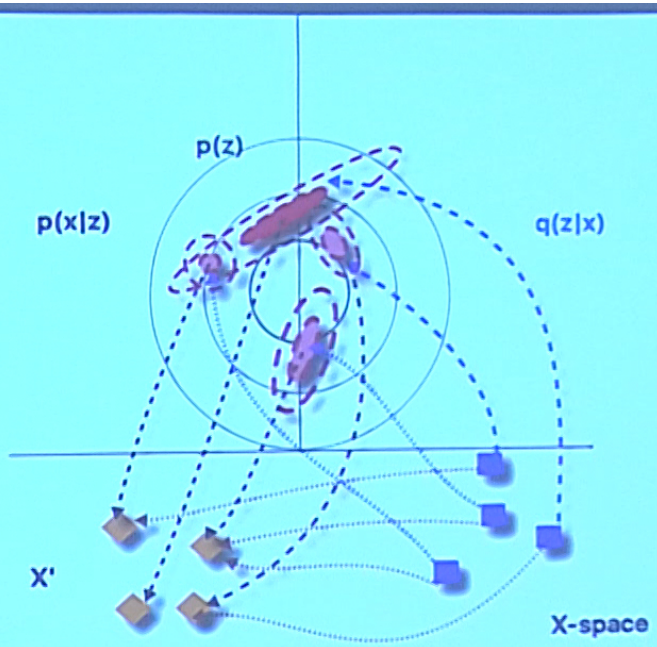


Figure 5: Visualizing how datapoints $\mathbf{x}$ are mapped into latent space, and potential redundancy in how they are decoded.

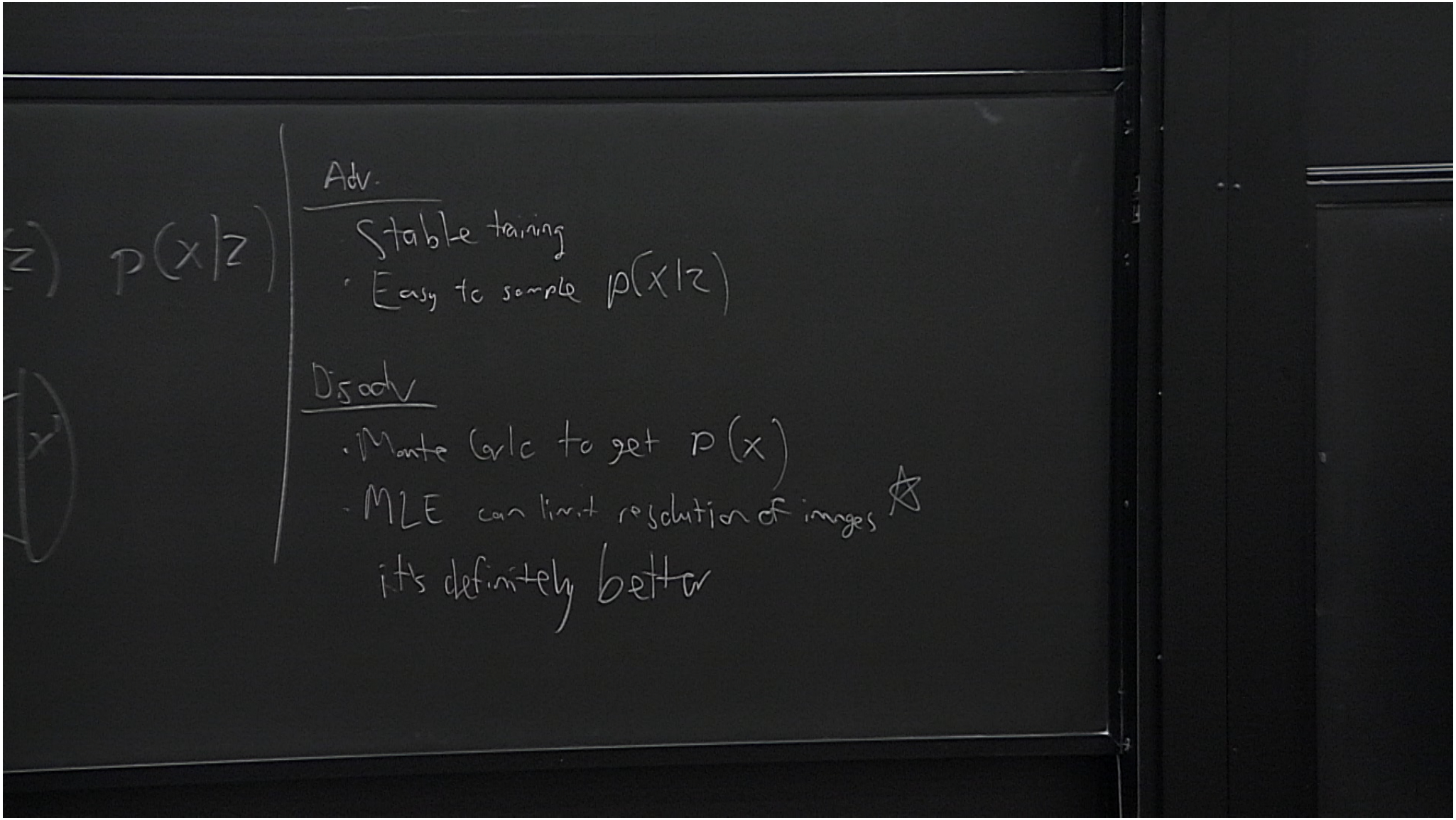
As the KL term is more strongly enforced, data will be better spread around the Gaussian prior, but the generated image quality will be worse. If the KL term is not enforced as strongly (as might be seen above), posterior Gaussians can overlap, and multiple inputs might map to the same generated $\mathbf{x}'$. This can be improved with the Wasserstein Autoencoder by Tolstikhin et al in which $q(\mathbf{z}) = \int q(\mathbf{z}|\mathbf{x})dp(\mathbf{x})$ is forced to match $p(\mathbf{z})$ rather than each $q_\phi(\mathbf{z}|\mathbf{x})$ on their own.

Figure 5: Visualizing how datapoints $\mathbf{x}$ are mapped into latent space, and potential redundancy in how they are decoded.

As the KL term is more strongly enforced, data will be better spread around the Gaussian prior, but the generated image quality will be worse. If the KL term is not enforced as strongly (as might be seen above), posterior Gaussians can overlap, and multiple inputs might map to the same generated $\mathbf{x}'$. This can be improved with the Wasserstein Autoencoder by Tolstikhin et al in which $q(\mathbf{z}) = \int q(\mathbf{z}|\mathbf{x})dp(\mathbf{x})$ is forced to match $p(\mathbf{z})$ rather than each $q_\phi(\mathbf{z}|\mathbf{x})$ on their own. **Note**: there are many many advances and perspectives on these models, and more keep coming. I just reference the Wasserstein Autoencoder case because it elucidates some of the behavior of the latent space visually well.

For example of some other advances in the understanding, one can take a look at ways of disentangling the latent space of these models so that each latent dimension has some more interpretable understanding of its representation. This can be done by enforcing independence between the latent dimensions, which becomes apparent if you break down the KL term into its newly discovered 3 subparts, one of which is the Total Correlation. This can be done adversarially, with d-Hilbert-Schmidt Independence Criteria, or with MMD and variations of adversarial training. One can also tailor their latent space to be a sampling on a manifold that corresponds to the symmetries/transformations that govern the true data.

Advantages:


p(x|z)
Adv.
Stable training
Easy to sample p(x|z)
Disadv
Monte Carlo to get p(x)
MLE can limit resolution of images
it's definitely better

Implicit GM

GAN

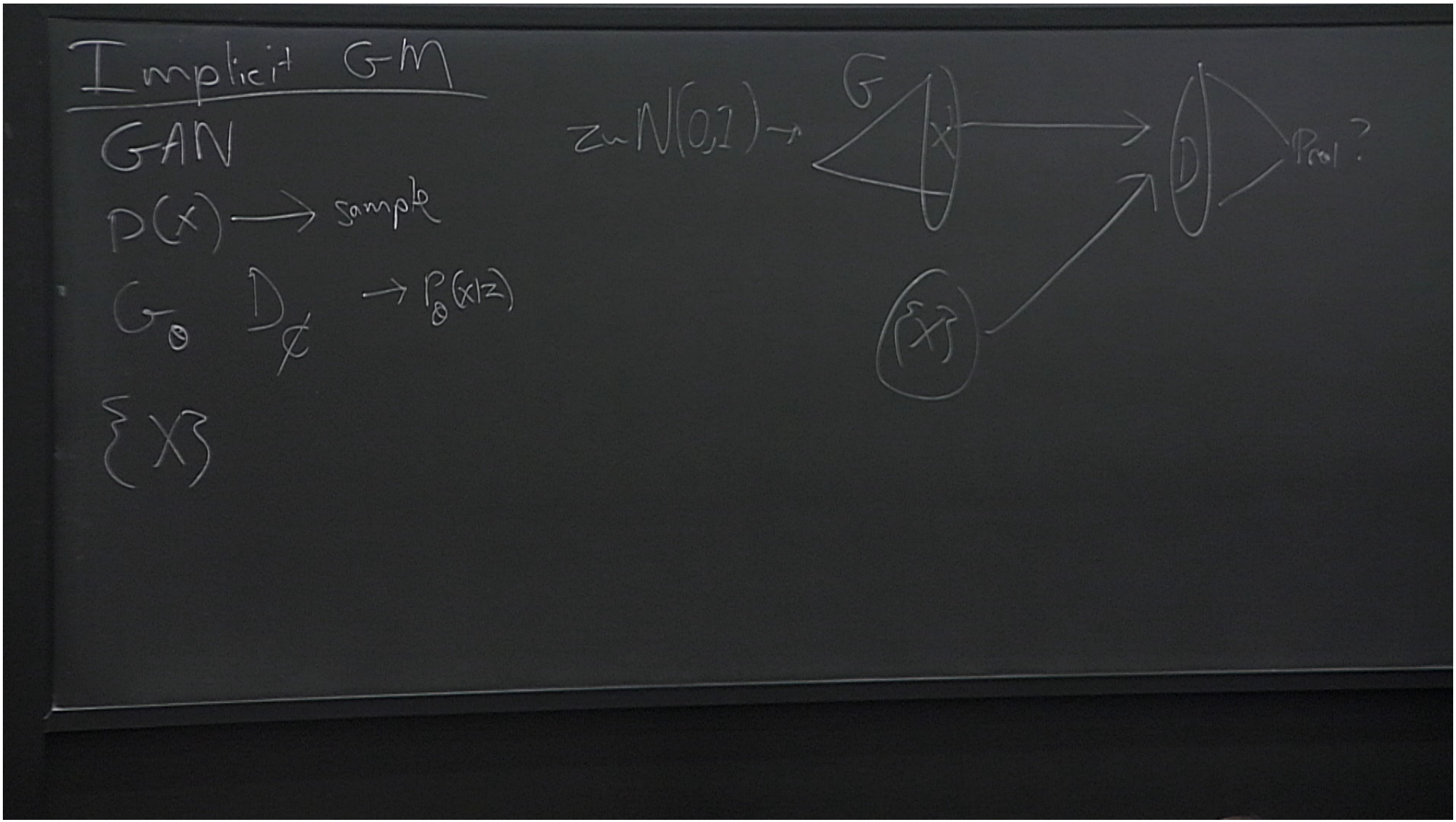
$P(x) \longrightarrow$ sample

G_θ D_ϕ $\rightarrow P_\theta(x|z)$

$\{X\}$

$z \sim N(0,1) \rightarrow$ G $\rightarrow$ x' $\rightarrow$ D $\rightarrow$ Real?

$\{X\} \rightarrow$ D

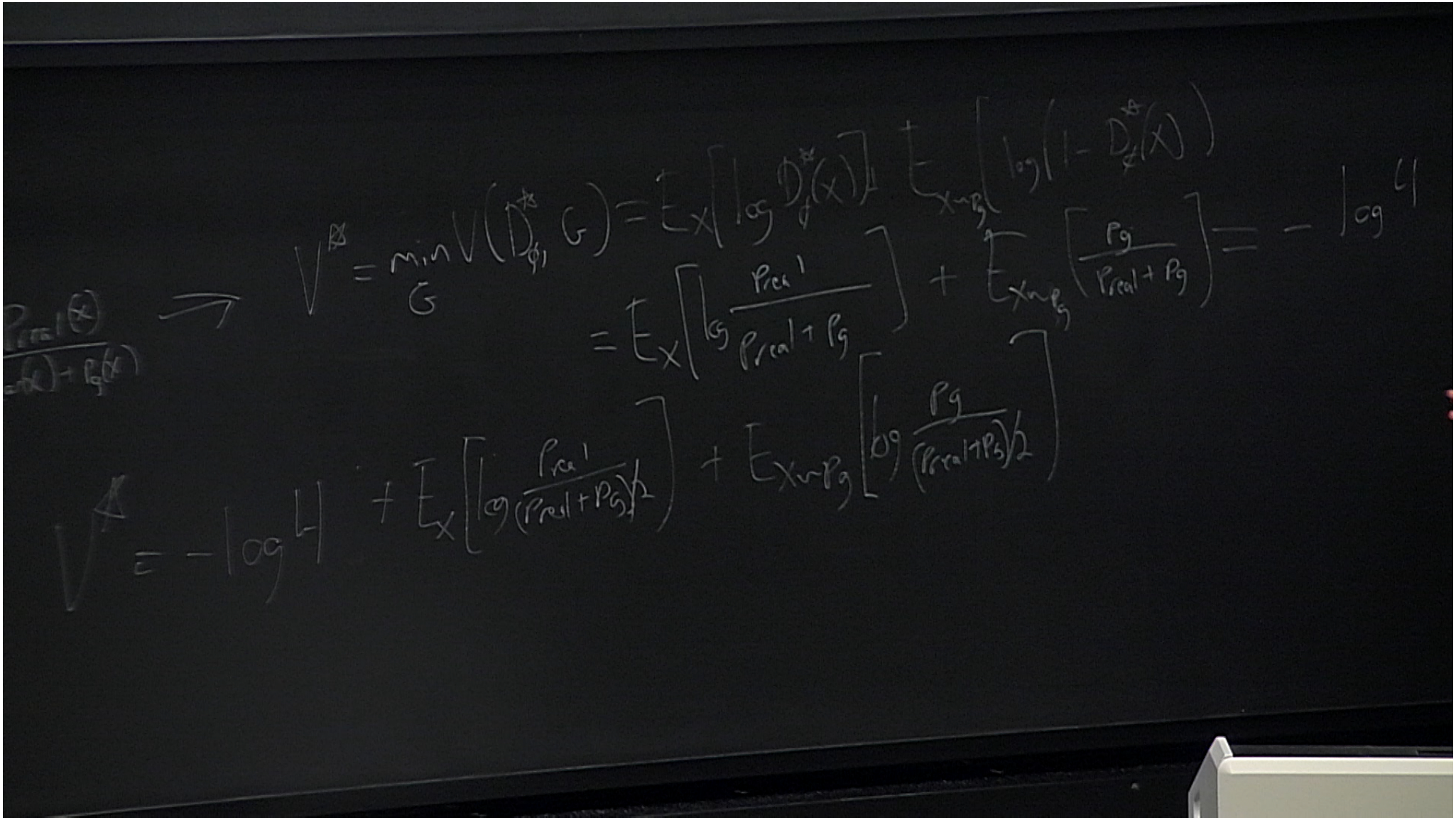


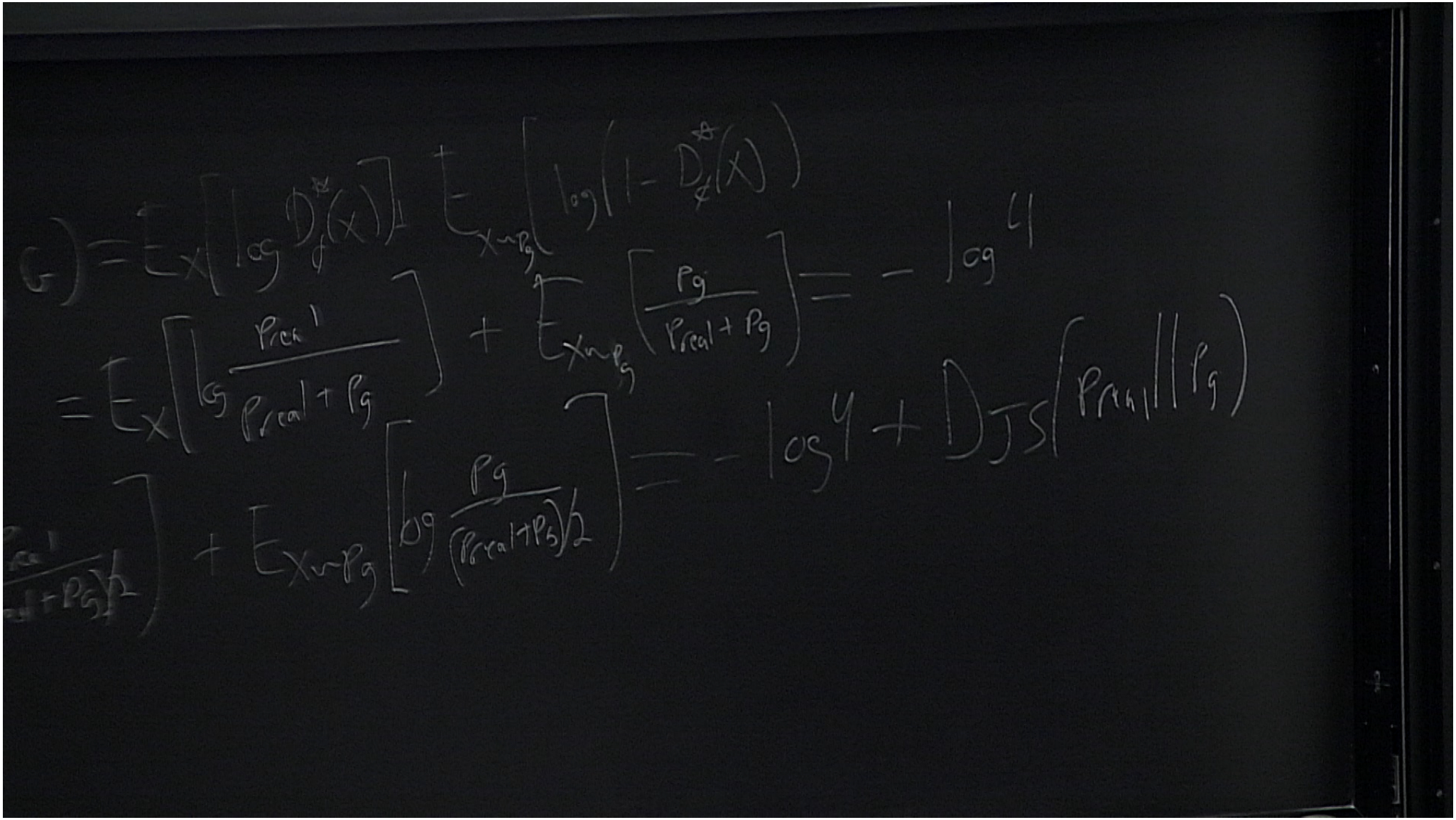
$$\min_G \max_D V(D,G) = \mathbb{E}_{x \sim p_{real}}[\log D(x)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z)))]$$

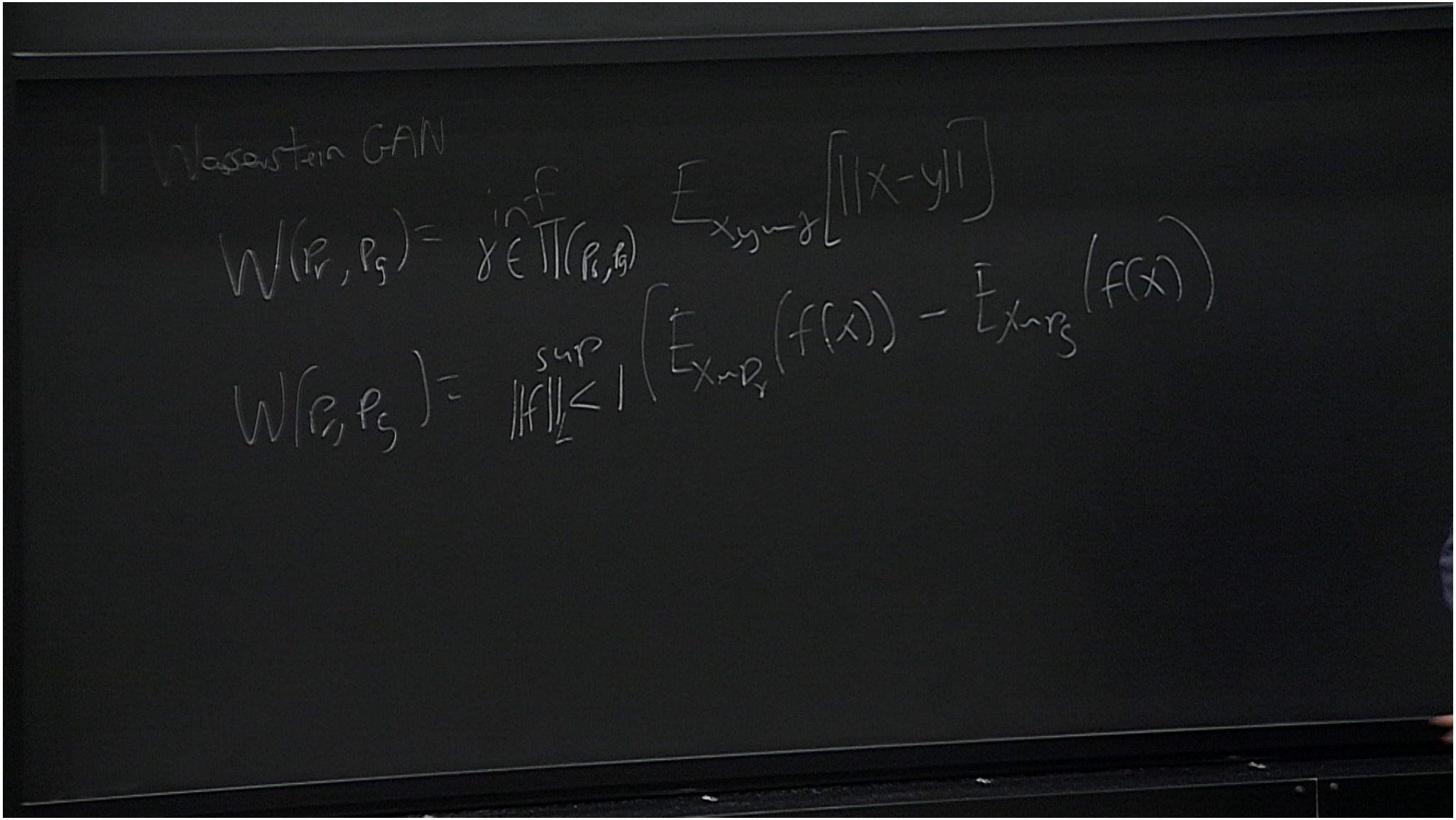
$$V(D,G) = \int_x p_{real} \log D \, dx + \int_z p_z(z) \log(1 - D(G(z))) \, dz$$

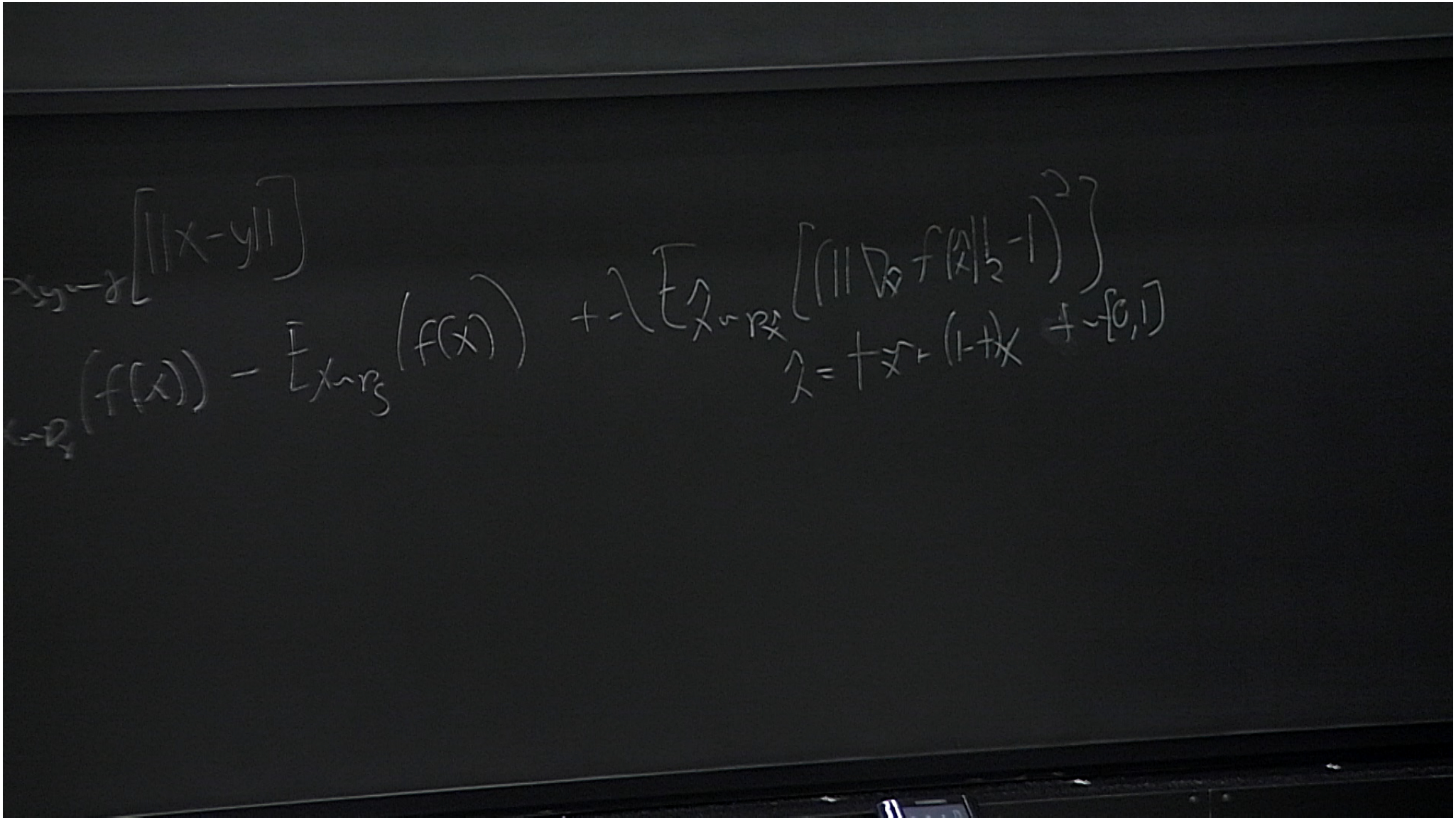
$$= \int_x p_{real} \log D(x) + p_g \log(1 - D(x)) \, dx$$

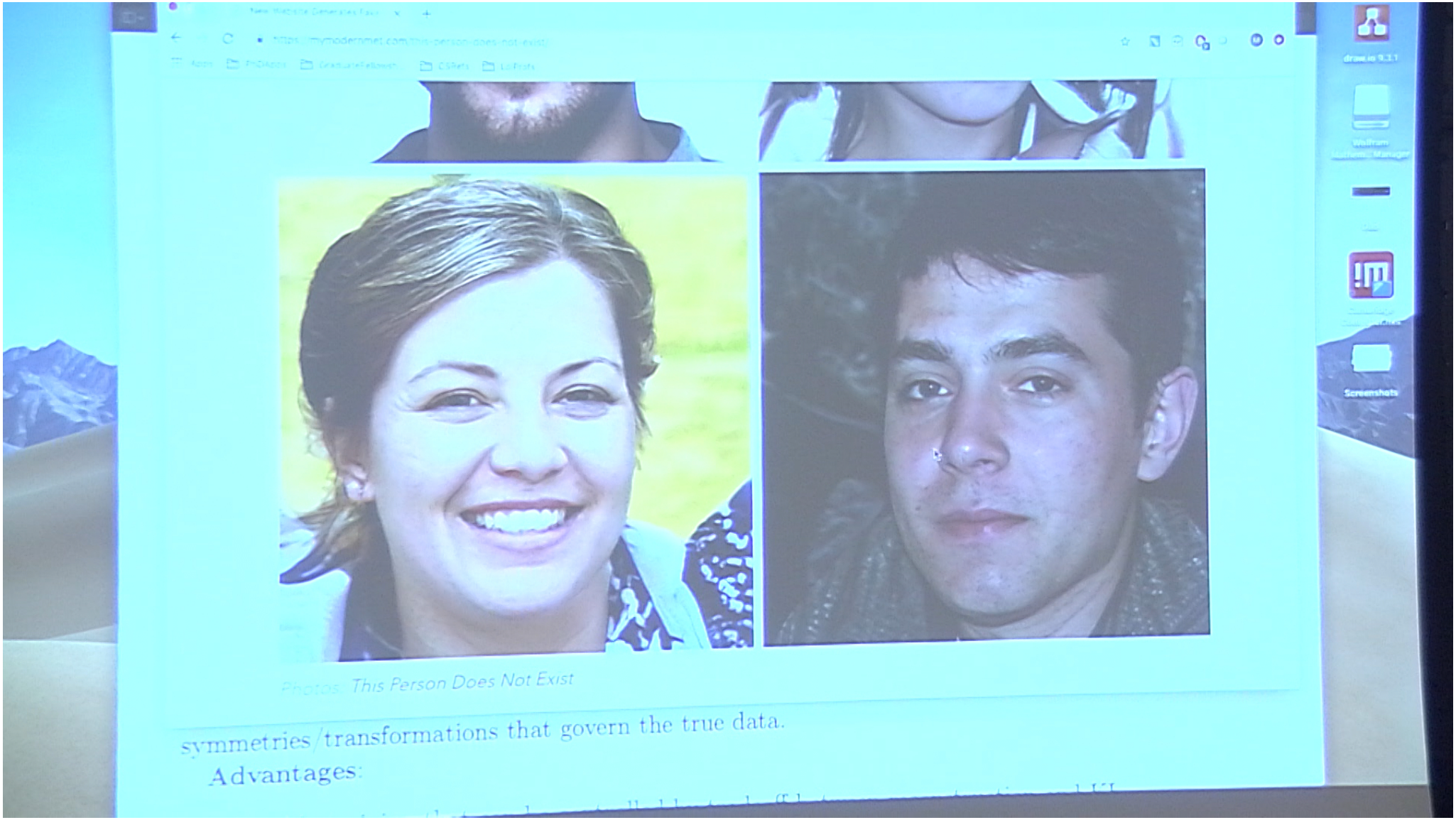
$$D_G^* = \frac{p_{real}(x)}{p_{real}(x) + p_g(x)}$$

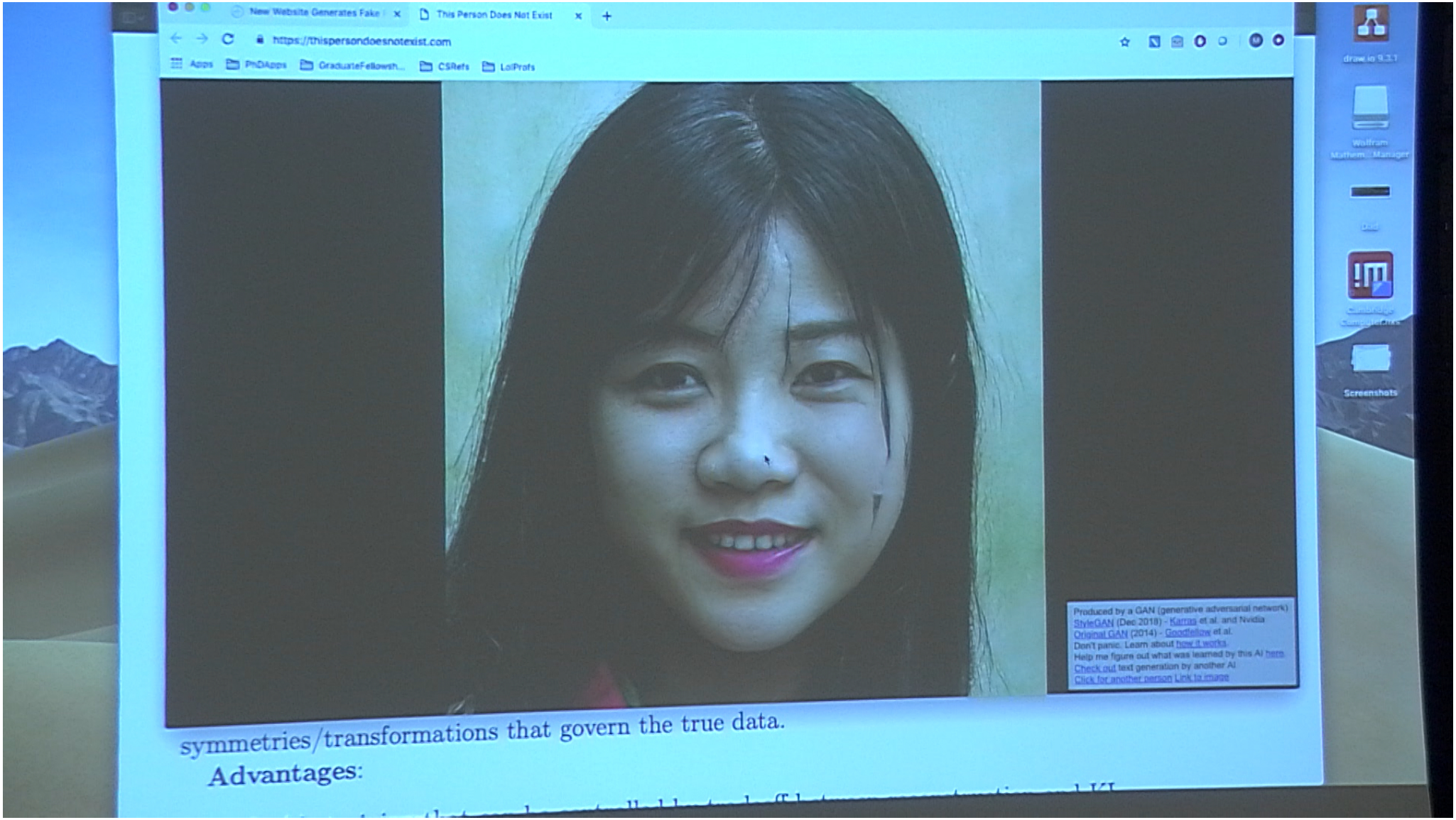

$\frac{P_{real}(x)}{P_{real}(x)+P_g(x)} \rightarrow$
$V^* = \min_G V(D^*_\phi, G) = E_x[\log D^*_\phi(x)] + E_{x\sim P_g}[\log(1 - D^*_\phi(x))]$
$= E_x\left[\log \frac{P_{real}}{P_{real}+P_g}\right] + E_{x\sim P_g}\left[\frac{P_g}{P_{real}+P_g}\right] = -\log 4$
$V^* = -\log 4 + E_x\left[\log \frac{P_{real}}{(P_{real}+P_g)/2}\right] + E_{x\sim P_g}\left[\log \frac{P_g}{(P_{real}+P_g)/2}\right]$


G) = E_x[log D*_G(x)] + E_{x~P_g}[log(1 - D*_G(x))
= E_x[log P_real / (P_real + P_g)] + E_{x~P_g}[P_g / (P_real + P_g)] = -log 4
P_real / (P_real + P_g)/2] + E_{x~P_g}[log P_g / (P_real + P_g)/2] = -log 4 + D_JS(P_real || P_g)


Wasserstein GAN
$W(P_r, P_g) = \inf_{\gamma \in \Pi(P_r, P_g)} E_{x,y \sim \gamma}[\|x-y\|]$
$W(P_r, P_g) = \sup_{\|f\|_L \leq 1} \left(E_{x \sim P_r}(f(x)) - E_{x \sim P_g}(f(x)) \right)$


$\mathbb{E}_{x,y\sim\gamma}[\|x-y\|]$
$\mathbb{E}_{x\sim P_r}[f(x)] - \mathbb{E}_{x\sim P_g}[f(x)] + \lambda \mathbb{E}_{\hat{x}\sim P_{\hat{x}}}[(\|\nabla_{\hat{x}} f(\hat{x})\|_2 - 1)^2]$
$\hat{x} = t x + (1-t)\tilde{x}$
$t \sim U[0,1]$


Photos: This Person Does Not Exist
symmetries/transformations that govern the true data.
Advantages:


New Website Generates Fake
This Person Does Not Exist
https://thispersondoesnotexist.com
Apps
PhDApps
GraduateFellowsh...
CSRefs
LolProfs
Produced by a GAN (generative adversarial network)
StyleGAN (Dec 2018) - Karras et al. and Nvidia
Original GAN (2014) - Goodfellow et al.
Don't panic. Learn about how it works.
Help me figure out what was learned by this AI here
Check out text generation by another AI
Click for another person Link to image
symmetries/transformations that govern the true data.
Advantages: