Lecture 9: Generative Adversarial Networks Part 2

Start Recording!

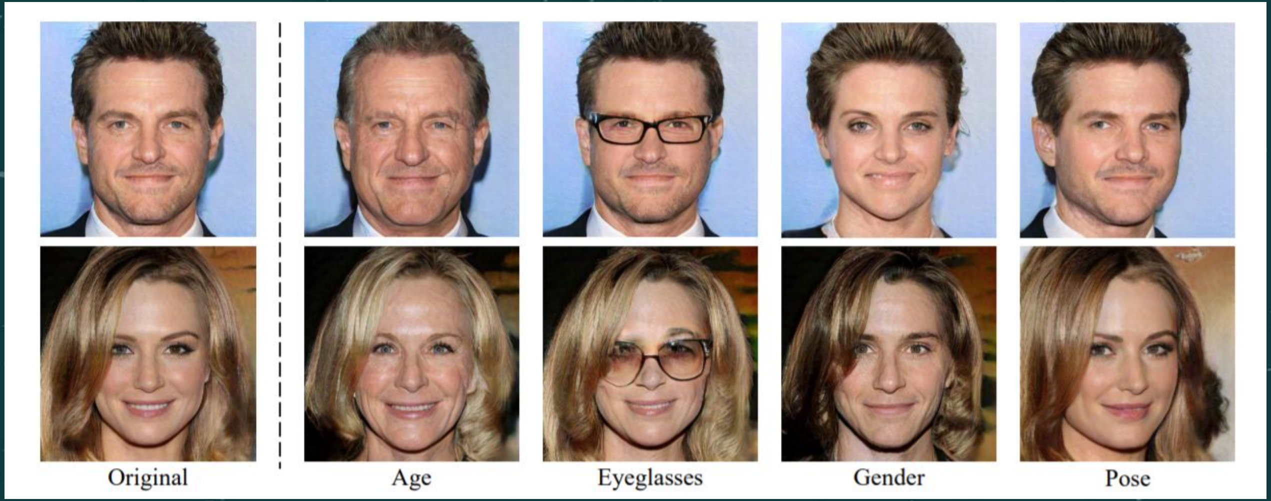
Announcementa

- Office Hours tomorrow (11-12h)
- Talks this Friday. Read the papers } Ask Questions on the papers on TEAMS!
- Scribes notes Available for Lecture 4 and 5!!! (see channel inTEAMS)
- Form to fill for the project [link] (in order for me to know the number of groups)
- Advice for the coding part of the project:
    - Start a Github repository (with frequent commits).
    - It is fine to use some open source code **if you are transparent about it!**
    - If you need advice about the coding workflow/good practices come to the office hours.
    - Getting new and well motivated experimental results can be enough by itself for a project.

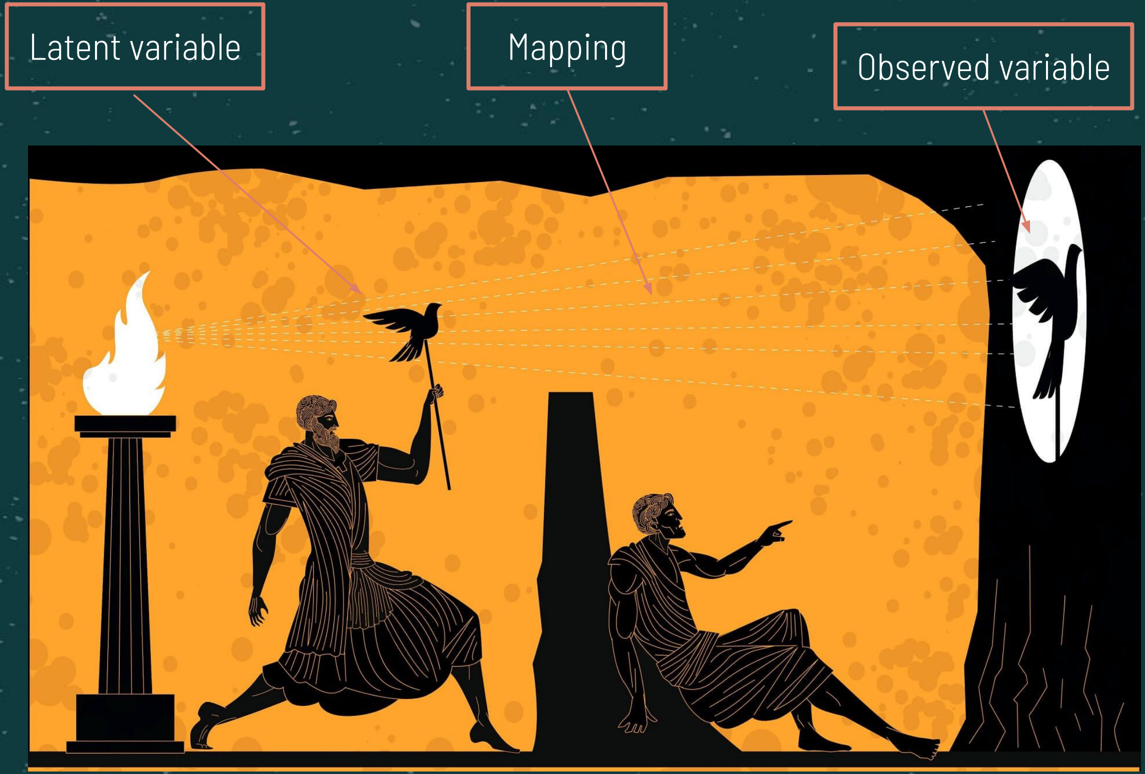
References to read for this course:

1. Salimans, Tim, et al. "Improved techniques for training gans." *arXiv preprint arXiv:1606.03498* (2016).
2. Sajjadi, Mehdi SM, et al. "Assessing generative models via precision and recall." *NeurIPS 2018*
3. Heusel, Martin, et al. "Gans trained by a two time-scale update rule converge to a local nash equilibrium." *NeurIPS 2018.*

Arithmetic in the Latent Space



What is the Latent Space?



Latent variable

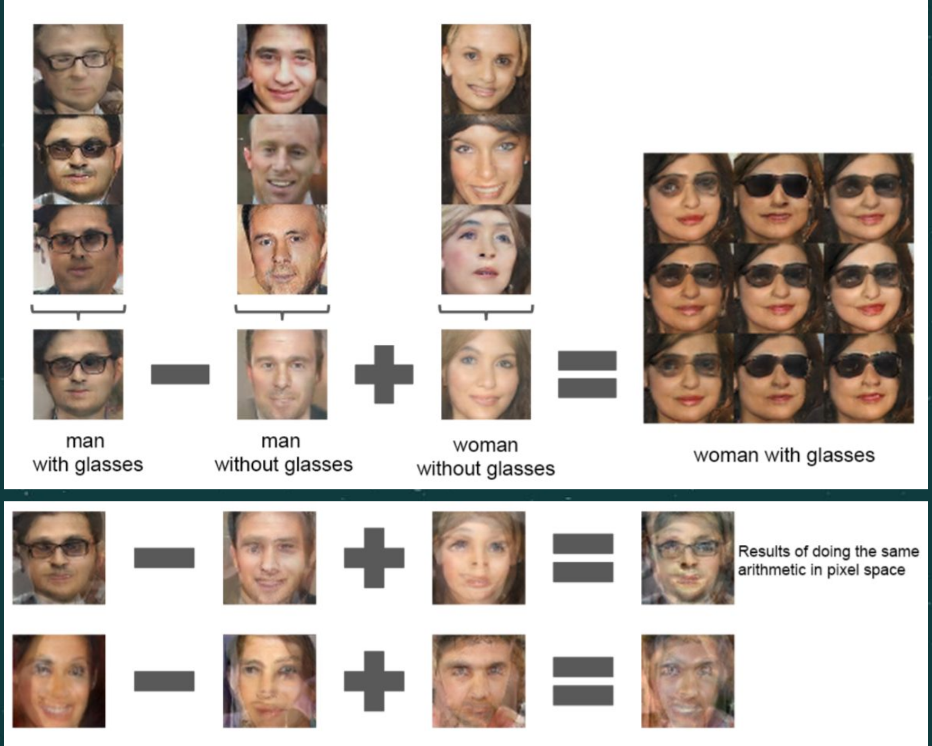
Mapping

Observed variable

Source: https://ourpolitics.net/the-allegory-of-the-cave-textual-analysis/

Arithmetic in the Latent Space

- Initial idea from Radford et al [2016]

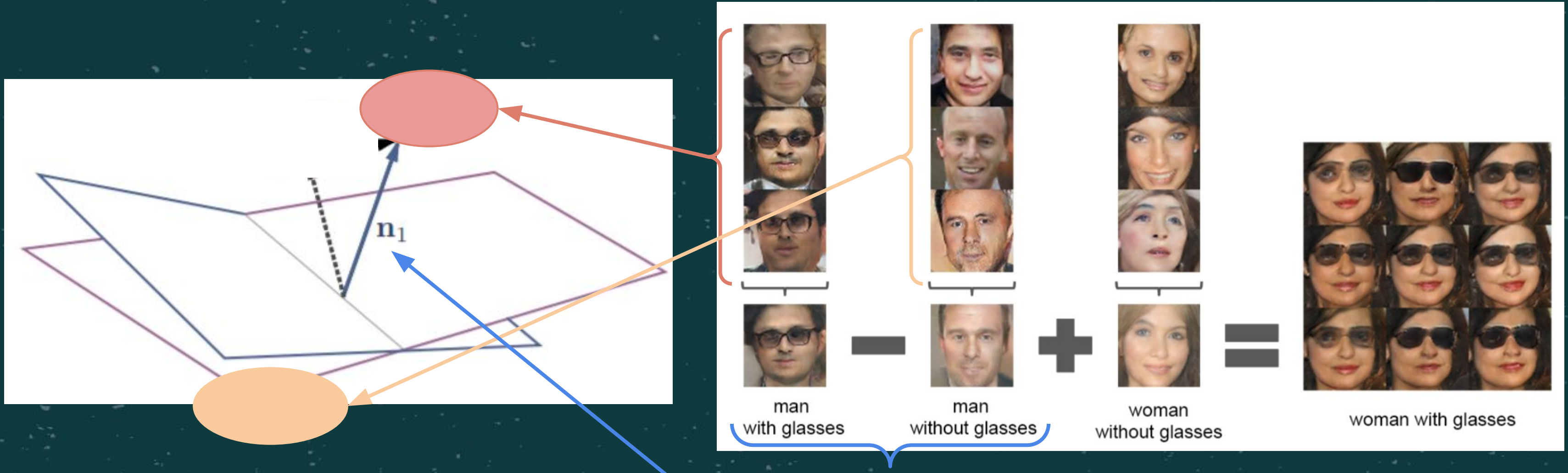


Arithmetic in latent space

Arithmetic in pixel space

Arithmetic in the Latent Space

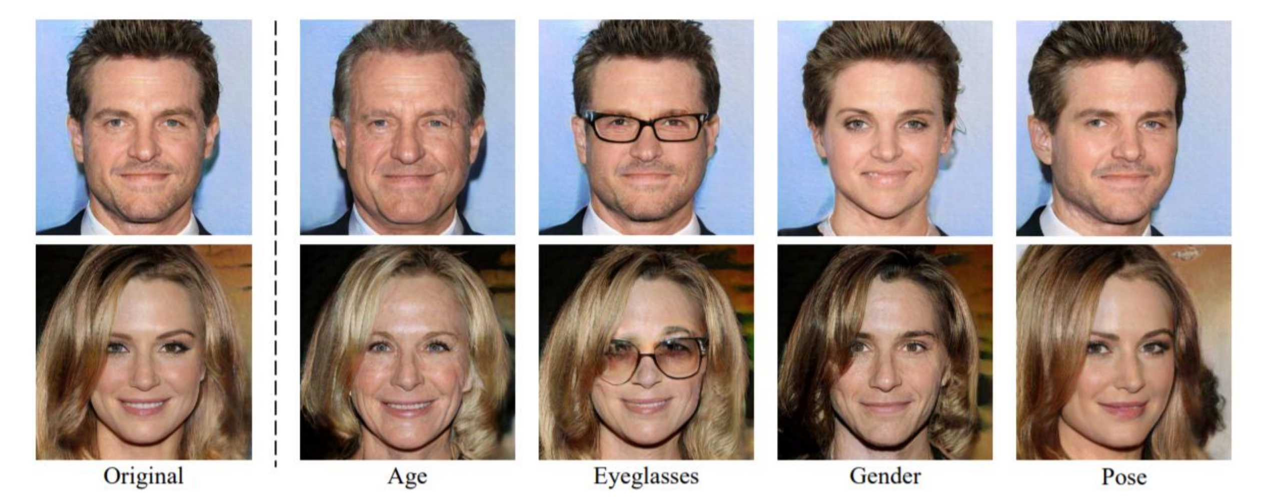
- Initial idea from Radford et al [2016]



Latent direction for glasses

Arithmetic in the Latent Space

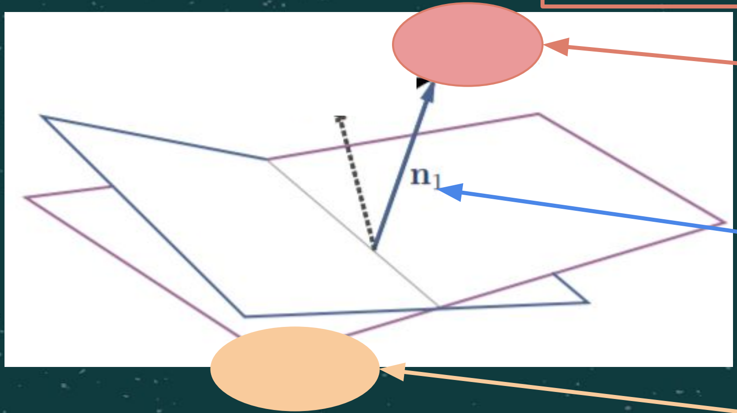
Idea: **learn** the "latent directions" of these features(Age, Eyeglasses, Gender, pose).



https://genforce.github.io/interfacegan/ (2020)

Arithmetic in the Latent Space

Idea: **learn** the “latent directions” of these features(Age, Eyeglasses, Gender, pose).



Mapping from image to latent space

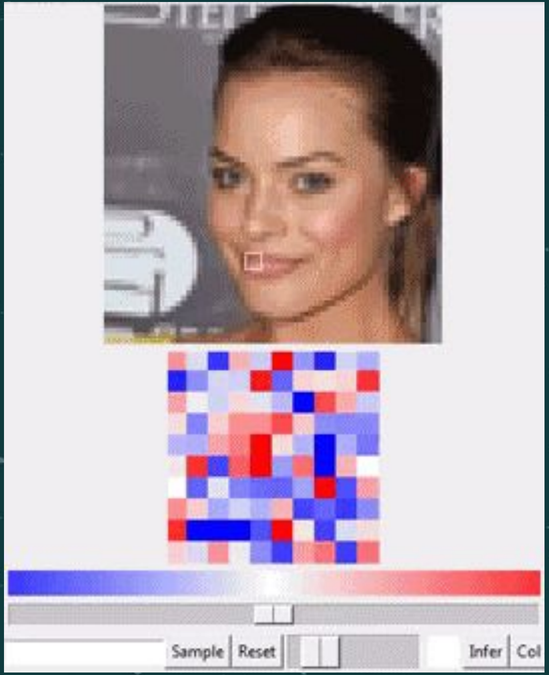
Male pictures

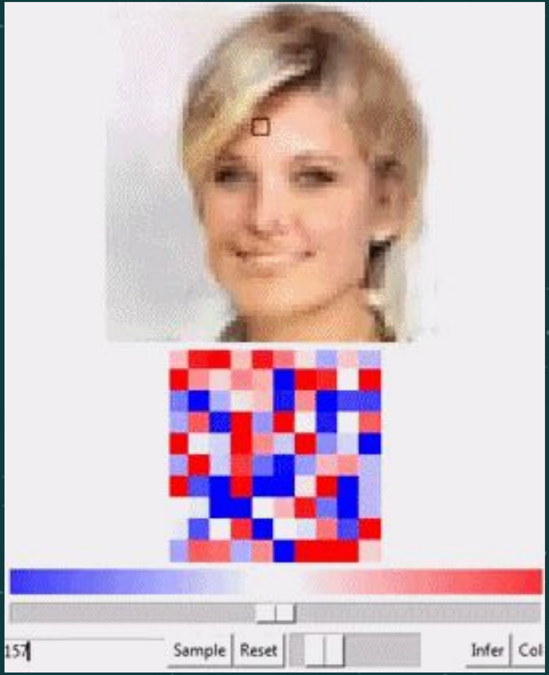
Direction of a linear binary classifier

Female pictures

Playing with the Latent Space

https://github.com/ajbrock/Neural-Photo-Editor (ICLR 2017)




157

Evaluation of Generative Models

- Previous lecture Theis et al. [2016] mentioned that we could have
  - Poor LL but samples.
  - Great LL but poor samples.
- Question: How do we evaluate sample fidelity and diversity?
  - By looking at the samples.

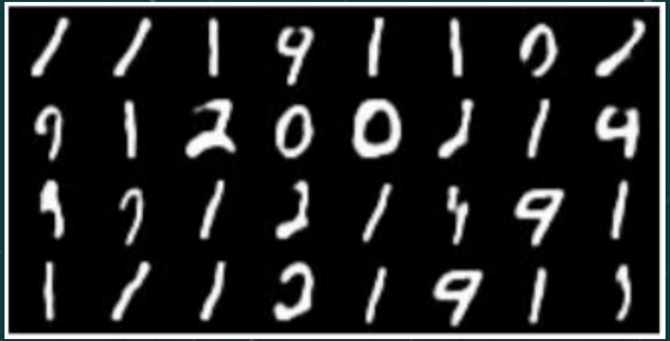
    Pro: it is what we care at the end.

    Cons: hard to actually evaluate diversity well, we are biased, we do not provide rigorous metric

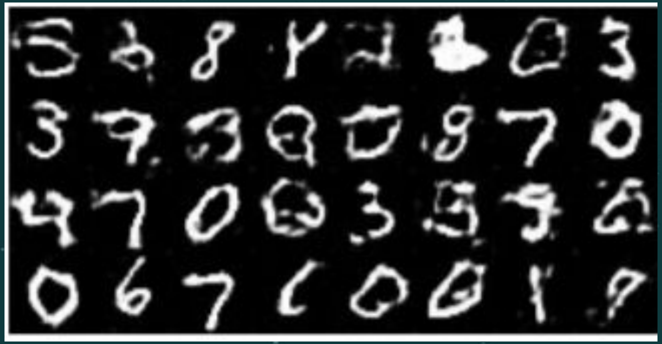
Eyeballing Evaluation

Question: which model is the best?

FID: 32 (lower is better)

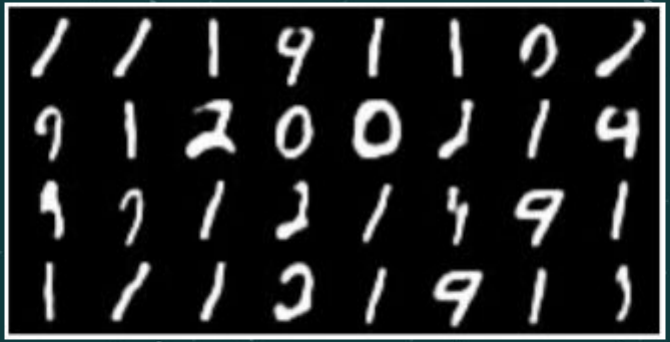


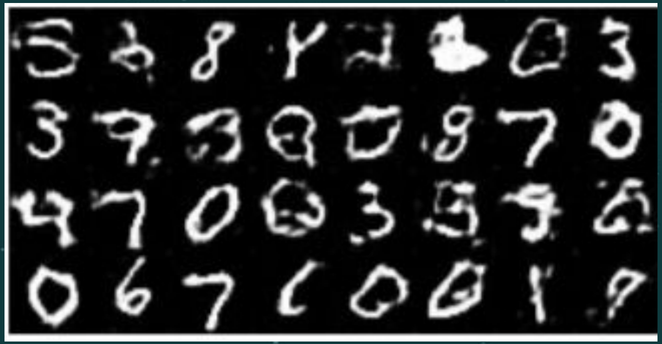
FID: 29



Eyeballing Evaluation

Question: which model is the best?

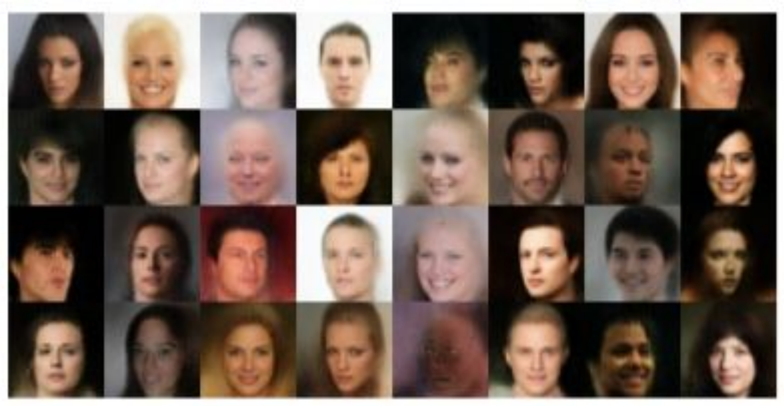




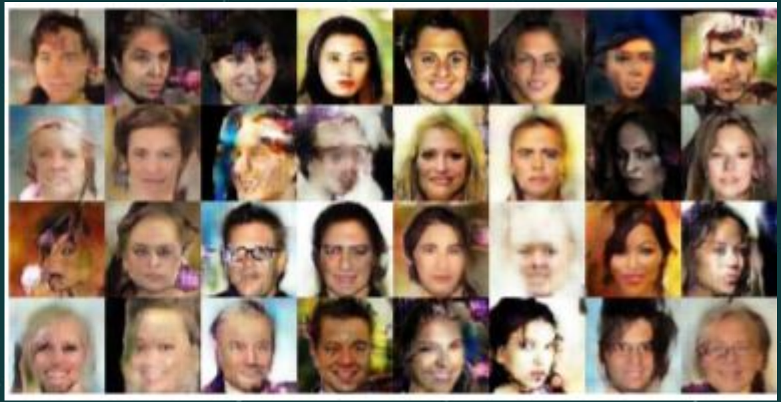
Eyeballing Evaluation

Question: which model is the best?

FID: 65 (lower is better)



FID: 62



Evaluation of Generative Models

- Previous lecture Theis et al. [2016] mentioned that we could have
    - Poor LL but samples.
    - Great LL but poor samples.
- Question: How do we evaluate sample fidelity and diversity?
    - By looking at the samples.

      Pro: it is what we care at the end.

      Cons: hard to actually evaluate diversity well, we are biased, we do not provide rigorous metric
    - **By using a pretrained classifier on these images!**

Inception Score

Inception score:

- Proposed by Salimans et al [2016]
- Use a Standard pretrained Classifier. (Inception Model)
- We can thus estimate label distribution with this model:

$$p_g(y|x) \approx f_\theta(x)$$

Inception Score

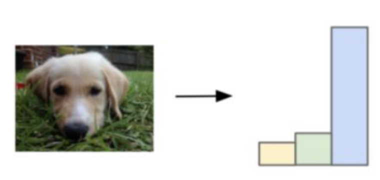
Idea: Generated dataset should be well classified by a pretrained classifier :

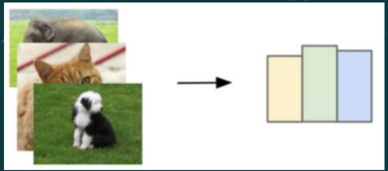
Inception Model

Estimation of the distribution of labels

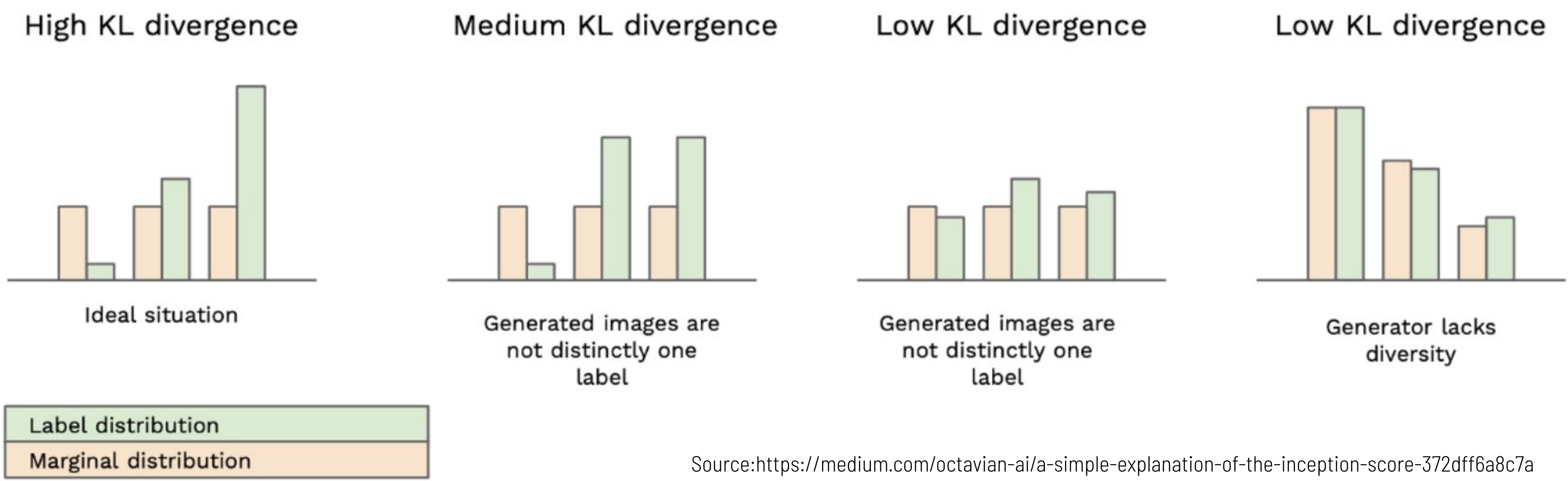
$$\begin{cases} p_g(y|x) \approx f_\theta(x) \\ p_g(y) \approx \mathbb{E}_{x\sim p_g}[f_\theta(x)] \end{cases}$$

$$\begin{cases} p_g(y|x) \longrightarrow \boxed{\text{Should be very picky (fidelity)}} \\ p_g(y) \longrightarrow \boxed{\text{Should be uniform (diversity)}} \end{cases}$$





Inception Score



High KL divergence

Ideal situation

Medium KL divergence

Generated images are not distinctly one label

Low KL divergence

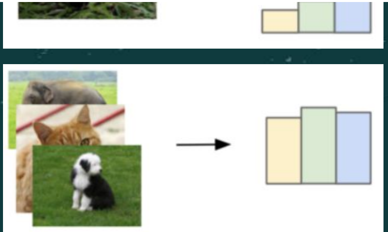
Generated images are not distinctly one label

Low KL divergence

Generator lacks diversity

Source:https://medium.com/octavian-ai/a-simple-explanation-of-the-inception-score-372dff6a8c7a

$p_g(y)$ → Should be uniform (diversity)



Inception Score

Idea: Generated dataset should be well classified by a pretrained classifier :

Estimation of the distribution of labels

$$\left\{\begin{aligned} p_g(y|x) &= f_\theta(x) \\ p_g(y) &\approx \mathbb{E}_{x\sim p_g}[f_\theta(x)] \end{aligned}\right.$$

Inception Model

$$IS(G) := \exp(\mathbb{E}_{x\sim p_g}[KL(p(y|x)||p(y))])$$

Generated images

Inception Score

Observation: IS correlates well with performances.

Problems with IS:

1. Depends on the weights θ (different results with pytorch and TF)

$$p_g(y|x) = f_\theta(x)$$

2. Not reporting overfitting (repeating the train set would give great IS)
3. Only care about labels diversity (not about diversity within labels)

$$IS(G) := \exp(\mathbb{E}_{x\sim p_g}[KL(p(y|x)||p(y))])$$

Fréchet Inception Distance

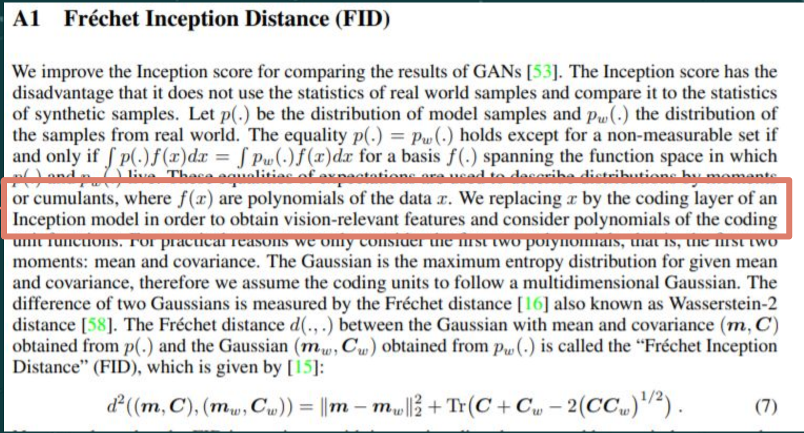
Based on a different idea. **If** we assume:

$$x_{data} \sim \mathcal{N}(\mu_1, \Sigma_1) \quad \text{and} \quad x_{fake} \sim \mathcal{N}(\mu_2, \Sigma_2)$$

Then we have a distance defined as:

$$d(p_{data}, p_g) = \underbrace{\|\mu_1 - \mu_2\|_2^2}_{\text{Distance btw the means}} + \underbrace{Tr(\Sigma_1 + \Sigma_2 - 2(\Sigma_1 \Sigma_2)^{1/2})}_{\substack{\text{Distance btw the Covariances} \\ \text{(Can you see why?)}}}$$

Fréchet Inception Distance



A1 Fréchet Inception Distance (FID)

We improve the Inception score for comparing the results of GANs [53]. The Inception score has the disadvantage that it does not use the statistics of real world samples and compare it to the statistics of synthetic samples. Let $p(.)$ be the distribution of model samples and $p_w(.)$ the distribution of the samples from real world. The equality $p(.) = p_w(.)$ holds except for a non-measurable set if and only if $\int p(.)f(x)dx = \int p_w(.)f(x)dx$ for a basis $f(.)$ spanning the function space in which $p(.)$ and $p_w(.)$ live. These equalities of expectations are used to describe distributions by moments or cumulants, where $f(x)$ are polynomials of the data x. We replacing x by the coding layer of an Inception model in order to obtain vision-relevant features and consider polynomials of the coding unit functions. For practical reasons we only consider the first two polynomials, that is, the first two moments: mean and covariance. The Gaussian is the maximum entropy distribution for given mean and covariance, therefore we assume the coding units to follow a multidimensional Gaussian. The difference of two Gaussians is measured by the Fréchet distance [16] also known as Wasserstein-2 distance [58]. The Fréchet distance $d(.,.)$ between the Gaussian with mean and covariance $(\boldsymbol{m}, \boldsymbol{C})$ obtained from $p(.)$ and the Gaussian $(\boldsymbol{m}_w, \boldsymbol{C}_w)$ obtained from $p_w(.)$ is called the "Fréchet Inception Distance" (FID), which is given by [15]:

$$d^2((\boldsymbol{m}, \boldsymbol{C}), (\boldsymbol{m}_w, \boldsymbol{C}_w)) = \|\boldsymbol{m} - \boldsymbol{m}_w\|_2^2 + \mathrm{Tr}\big(\boldsymbol{C} + \boldsymbol{C}_w - 2\big(\boldsymbol{C}\boldsymbol{C}_w\big)^{1/2}\big) . \quad (7)$$

Very important point

Mean and covariance in the feature space!!

What does it do:

- Unlike IS, FID can detect intra-class mode dropping, i.e. a model that generates only one image per class can score a perfect IS, but will have a bad FID

Problems with FID

- Still impossible to detect overfitting with it.
- Not really a distance. (only a distance for Gaussians distributions)

Summary of evaluation of Generative Models

1. Human evaluation (there is always pictures in GANs papers)
   a. Pros: We like pretty picture and it is in some sense “close” to the final task
   b. Cons: not a explicit value. (Hard to compare models that are close). Hard to get a sense of the diversity. Not robust against **cherry picking!**
2. Evaluation with a classifier:
   a. Pros: Reproducible metric
   b. Depends on the classifier
   c. Does not take into account generalization
   d. A single number for fidelity vs. diversity [Sajjadi et al. 2018]

Summary of evaluation of Generative Models

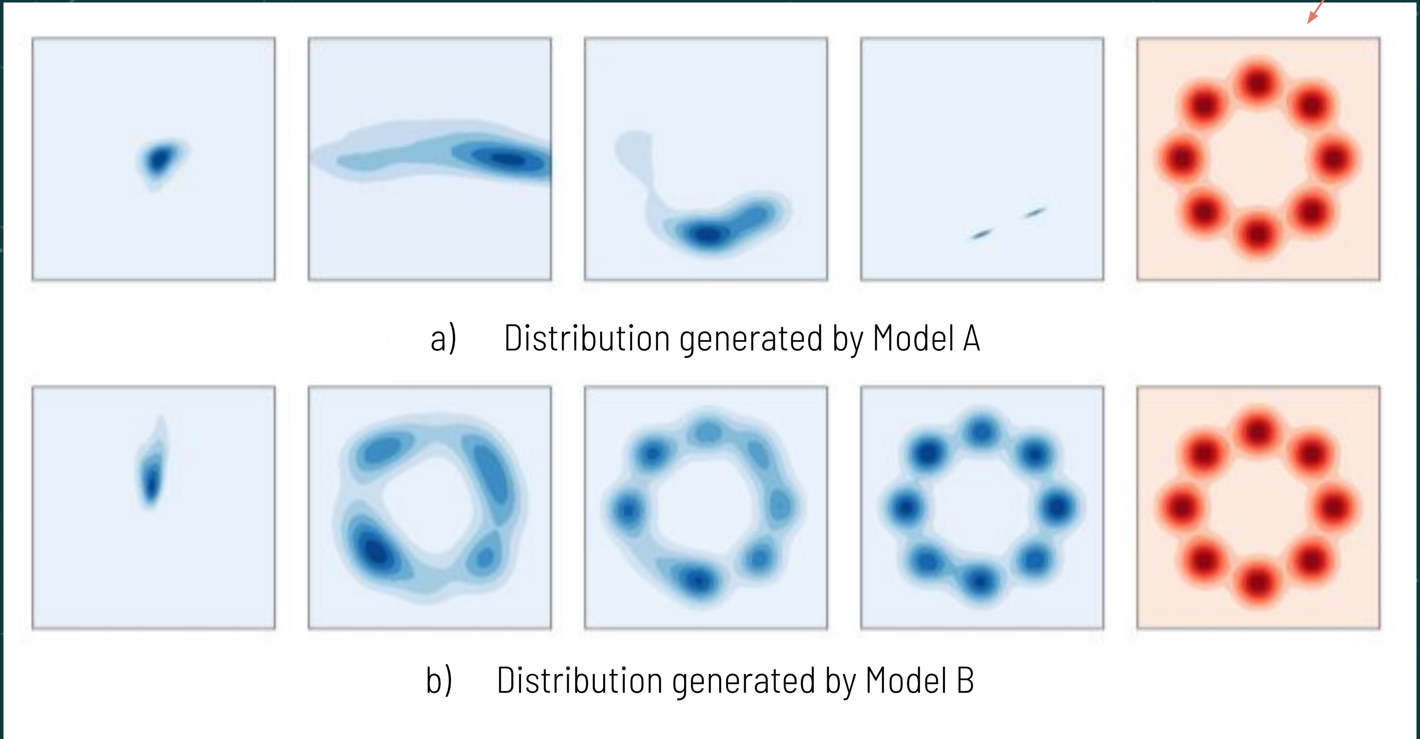
1. Human evaluation (there is always pictures in GANs papers)
   a. Pros: We like pretty picture and it is in some sense “close” to the final task
   b. Cons: not a explicit value. (Hard to compare models that are close). Hard to get a sense of the diversity. Not robust against **cherry picking!**

Important: Do not optimize these metrics directly !!!

2. Evaluation with a classifier:
   a. Pros: Reproducible metric
   b. Depends on the classifier
   c. Does not take into account generalization
   d. A single number for fidelity vs. diversity [Sajjadi et al. 2018]

Problems for evaluation

Data ~ Mixture of Gaussians



a) Distribution generated by Model A

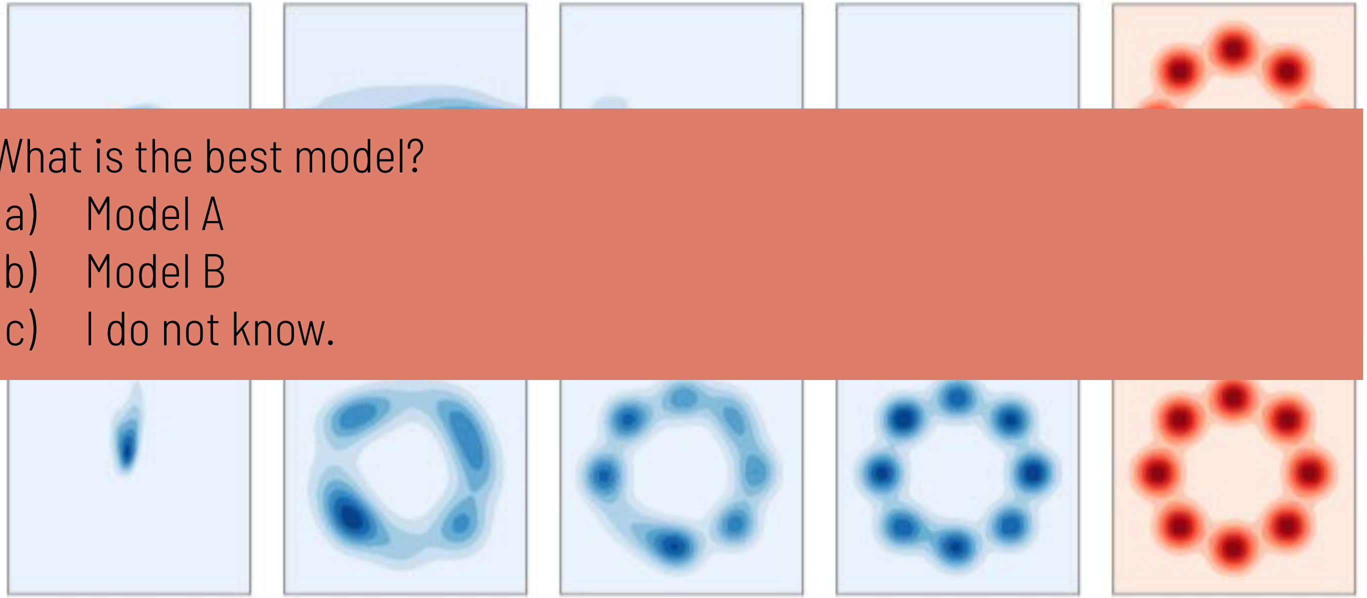
b) Distribution generated by Model B

Problems for evaluation

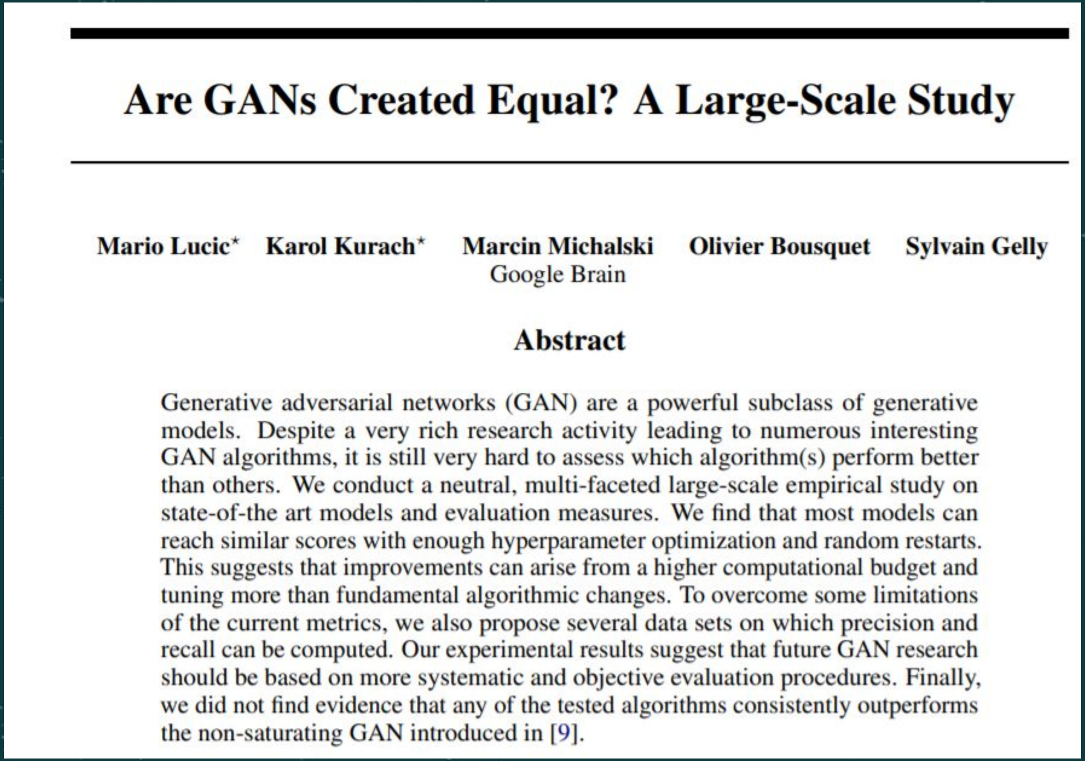
Data ~ Mixture of Gaussians

What is the best model?

a) Model A
b) Model B
c) I do not know.



b) Distribution generated by Model B



Are GANs Created Equal? A Large-Scale Study

Mario Lucic[⋆] **Karol Kurach**[⋆] **Marcin Michalski** **Olivier Bousquet** **Sylvain Gelly**
Google Brain

Abstract

Generative adversarial networks (GAN) are a powerful subclass of generative models. Despite a very rich research activity leading to numerous interesting GAN algorithms, it is still very hard to assess which algorithm(s) perform better than others. We conduct a neutral, multi-faceted large-scale empirical study on state-of-the art models and evaluation measures. We find that most models can reach similar scores with enough hyperparameter optimization and random restarts. This suggests that improvements can arise from a higher computational budget and tuning more than fundamental algorithmic changes. To overcome some limitations of the current metrics, we also propose several data sets on which precision and recall can be computed. Our experimental results suggest that future GAN research should be based on more systematic and objective evaluation procedures. Finally, we did not find evidence that any of the tested algorithms consistently outperforms the non-saturating GAN introduced in [9].

Challenges of a Fair Comparison

- Which metric to use? **Take:** Use different ones.
- Which hyperparameters? **Take:** cross validation?
- Which random seed? **Take:** DO NOT optimize. Make several runs
- Which dataset? **Take:** Use several ones.
- Which budget? **Take:** Same budget for each method (not easy in practice)
- Which Optimizer? **Take:** Fix it. But does not give the whole picture.
- Which NN Architecture? **Take:** Fix it. But does not give the whole picture.

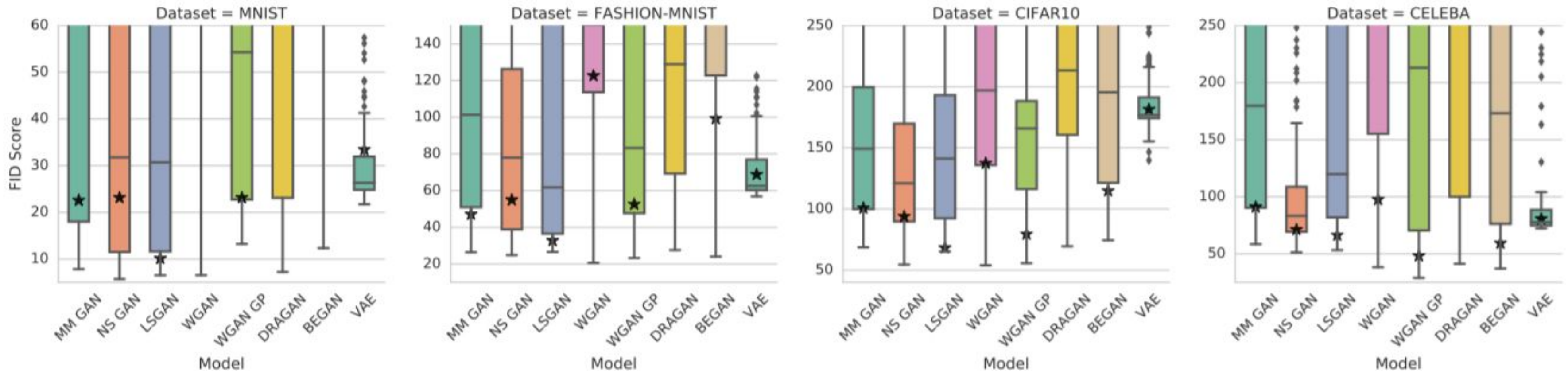
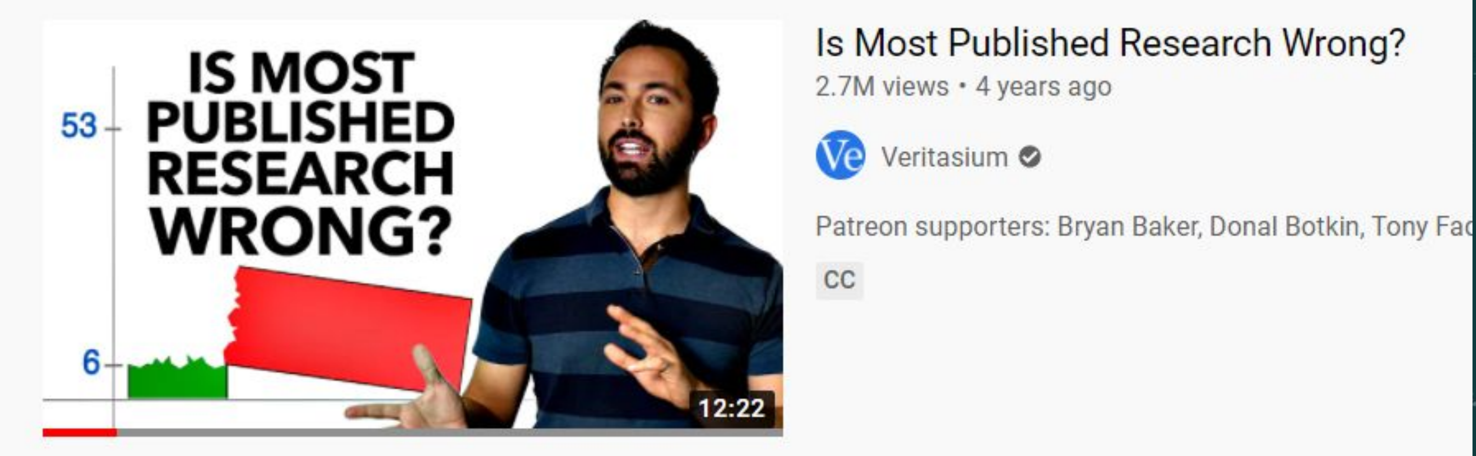


Figure 4: A *wide range* hyperparameter search (100 hyperparameter samples per model). Black stars indicate the performance of suggested hyperparameter settings. We observe that GAN training is extremely sensitive to hyperparameter settings and there is no model which is significantly more stable than others.

We need to be very careful with GANs!!!

Take Away

- Be careful with pretty pictures.
- Several runs with several random seeds are important!
- Ablation Study!!!!! (Harder, but It is the way to do good science)



Is Most Published Research Wrong?

2.7M views • 4 years ago

Veritasium

Patreon supporters: Bryan Baker, Donal Botkin, Tony Fac

CC

Useful Links:

- Salimans, Tim, et al. "Improved techniques for training gans." *arXiv* (2016).
- Sajjadi, Mehdi SM, et al. "Assessing generative models via precision and recall." *NeurIPS 2018*
- Heusel, Martin, et al. "Gans trained by a two time-scale update rule converge to a local nash equilibrium." *NeurIPS 2018.*
- Barratt, Shane, and Rishi Sharma. "A note on the inception score." *arXiv* (2018).