Learning underlaying feature from data.

OUTLINE:

- **Autoencoder: Encode into latent variable(compression) and decode it.**
- **Variational Autoencoder(VAE):**
    - **Encode data into an distribution.**
    - **Use that distribution to sample.**
    - **Loss: reconstruction loss + regularization Loss (enforce encoded distribution close to norm)**
- **Ganerative Aderversarial Networks(GANS):Competing generator and discriminator**
    - **Conditional GAN: paired data**
    - **Cycle GAN: unpaired data:**

Supervised vs unsupervised learning

Supervised Learning

Data: (x, y)
x is data, y is label

Goal: Learn function to map
$x \rightarrow y$

Examples: Classification, regression, object detection, semantic segmentation, etc.

Unsupervised Learning

Data: x
x is data, no labels!

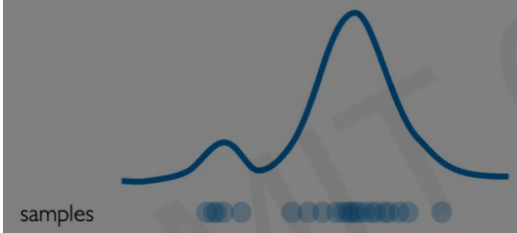
Goal: Learn the *hidden* or *underlying structure* of the data

Examples: Clustering, feature or dimensionality reduction, etc.

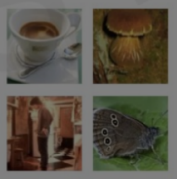
Generative modeling

Goal: Take as input training samples from some distribution and learn a model that represents that distribution

Density Estimation

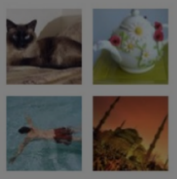


Sample Generation



Input samples

Training data ~ $P_{data}(x)$



Generated samples

Generated ~ $P_{model}(x)$

How can we learn $P_{model}(x)$ similar to $P_{data}(x)$?

Why generative models? Debiasing

Capable of uncovering **underlying features** in a dataset

VS

Homogeneous skin color, pose

Diverse skin color, pose, illumination

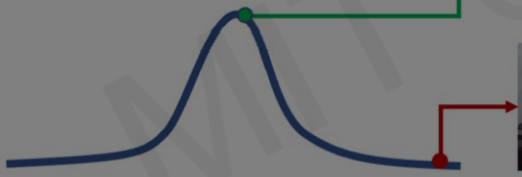
How can we use this information to create fair and representative datasets?

Why generative models? Outlier detection

- **Problem:** How can we detect when we encounter something new or rare?
- **Strategy:** Leverage generative models, detect outliers in the distribution
- Use outliers during training to improve even more!

95% of Driving Data:
(1) sunny, (2) highway, (3) straight road

Detect outliers to avoid unpredictable behavior when training



Edge Cases

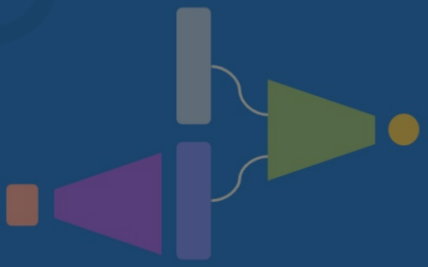
Harsh Weather

Pedestrians

Latent variable models

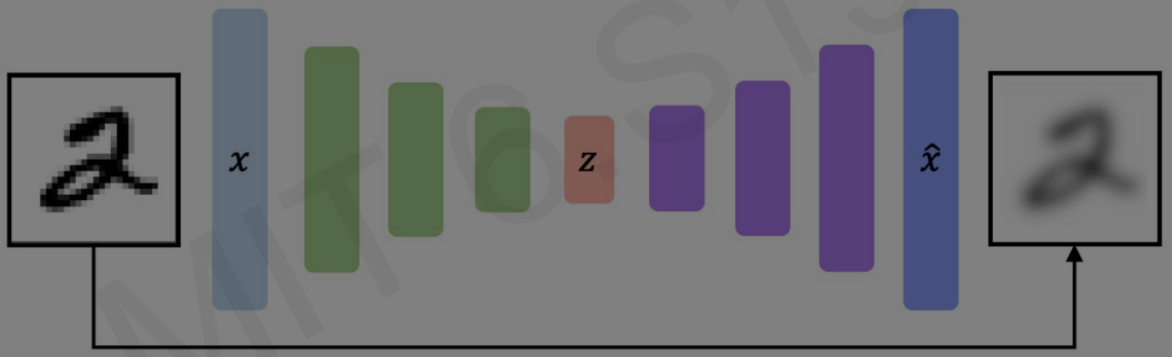
Autoencoders and Variational Autoencoders (VAEs)

Generative Adversarial Networks (GANs)



Autoencoders: background

How can we learn this latent space?
Train the model to use these features to **reconstruct the original data**

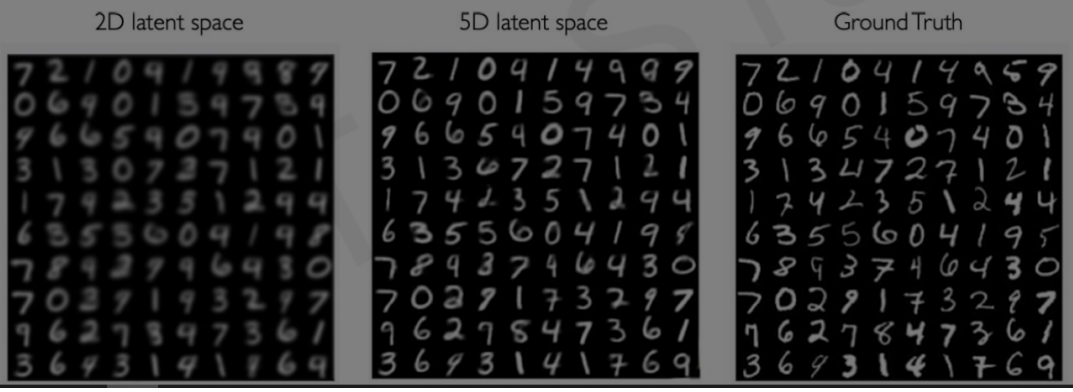


$\mathcal{L}(x, \hat{x}) = \|x - \hat{x}\|^2$

Loss function doesn't use any labels!!

Dimensionality of latent space → reconstruction quality

Autoencoding is a form of compression!
Smaller latent space will force a larger training bottleneck

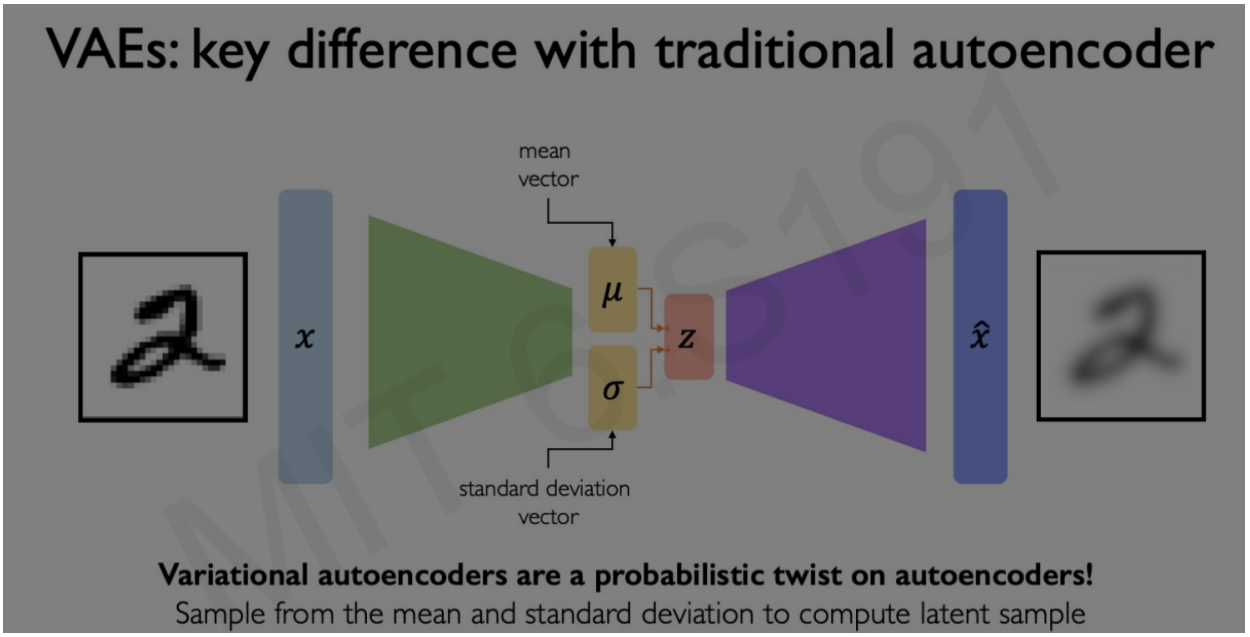


Autoencoders for representation learning

Bottleneck hidden layer forces network to learn a compressed latent representation

Reconstruction loss forces the latent representation to capture (or encode) as much "information" about the data as possible

Autoencoding = **Auto**matically **encoding** data



Shortcoming of AutoEncoder:

回忆一下我们在自动编码器中所做的事，我们需要输入一张图片，然后将一张图片编码之后得到一个隐含向量，这比我们随机取一个随机噪声更好，因为这包含着原图片的信息，然后我们隐含向量解码得到与原图片对应的照片。

但是这样我们其实并不能任意生成图片，因为我们没有办法自己去构造隐藏向量，我们需要通过一张图片输入编码我们才知道得到的隐含向量是什么，这时我们就可以通过变分自动编码器来解决这个问题。

其实原理特别简单，只需要在编码过程给它增加一些限制，迫使其生成的隐含向量能够粗略的遵循一个标准正态分布，这就是其与一般的自动编码器最大的不同。

这样我们生成一张新图片就很简单了，我们只需要给它一个标准正态分布的随机隐含向量，这样通过解码器就能够生成我们想要的图片，而不需要给它一张原始图片先编码。

在实际情况中，我们需要在模型的准确率上与隐含向量服从标准正态分布之间做一个权衡，所谓模型的准确率就是指解码器生成的图片与原图片的相似程度。我们可以让网络自己来做这个决定，非常简单，我们只需要将这两者都做一个loss，然后在将他们求和作为总的loss，这样网络就能够自己选择如何才能够使得这个总的loss下降。另外我们要衡量两种分布的相似程度，如何看过之前一片GAN的数学推导，你就知道会有一个东西叫KL divergence来衡量两种分布的相

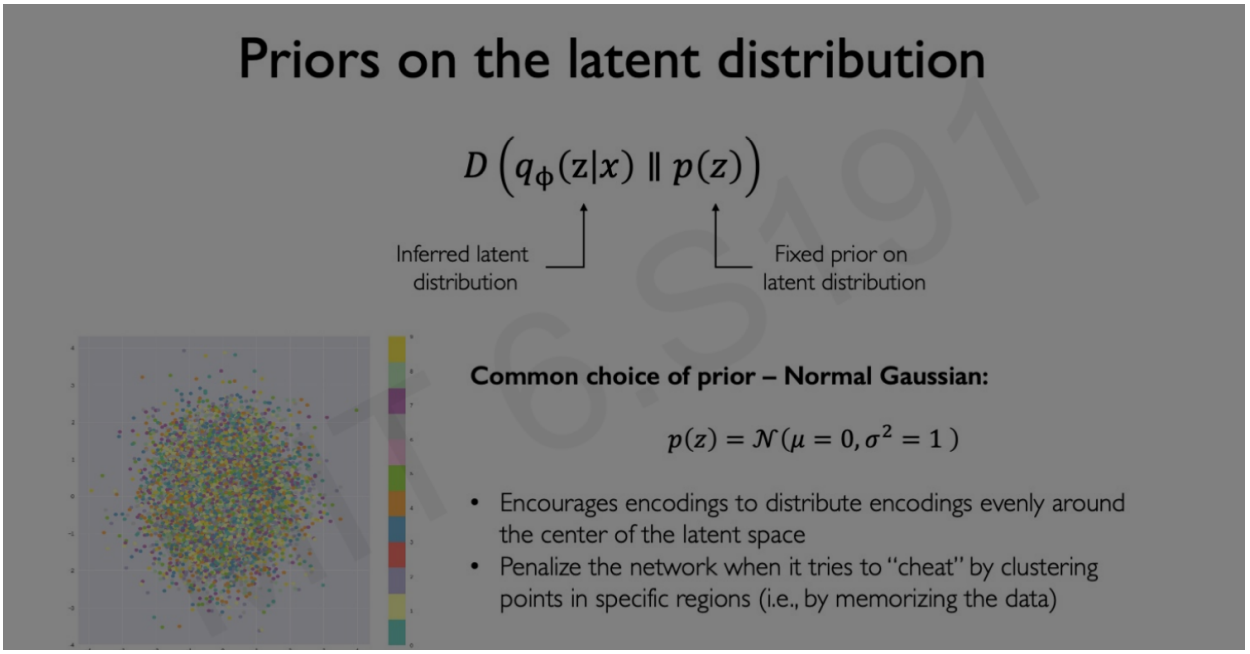
似程度，这里我们就是用KL divergence来表示隐含向量与标准正态分布之间差异的loss，另外一个loss仍然使用生成图片与原图片的均方误差来表示。

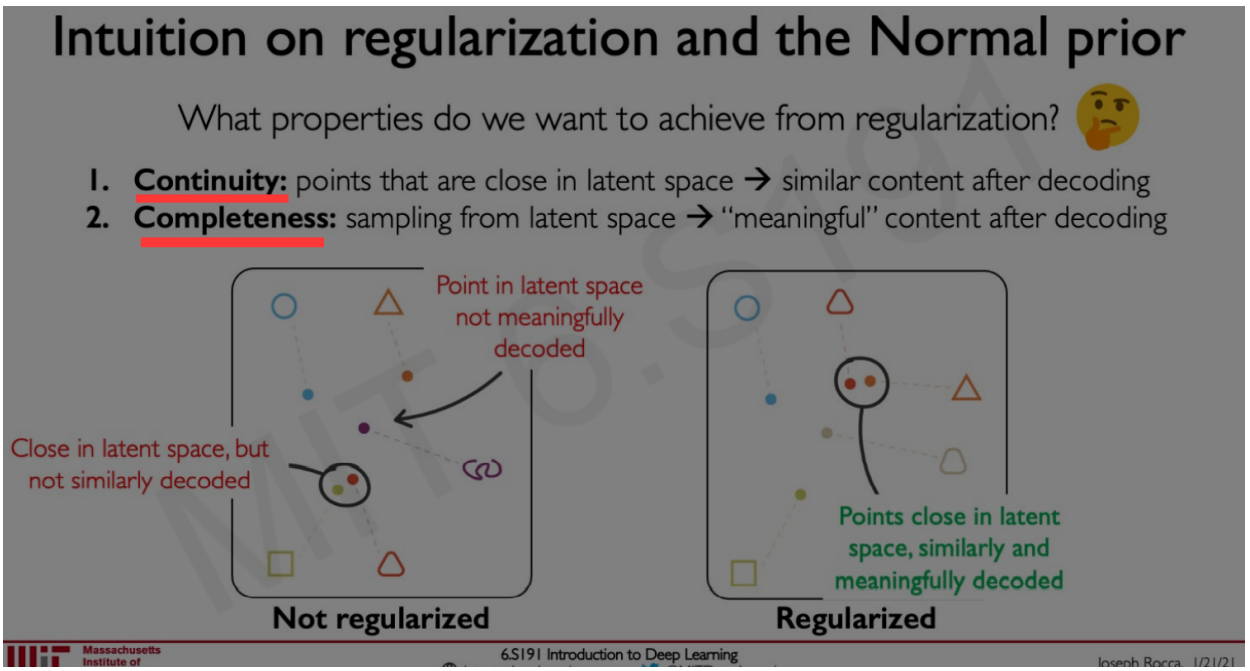
为什么要让latent variable 的分布接近 标准正态分布?

$$\mathcal{L}(\phi, \theta, x) = \text{(reconstruction loss)} + \text{(regularization term)}$$

reconstruction loss 衡量输入数据和生成数据的差距。

regularization term 衡量latent variable 的分布和 标准正态分布的差距。





VAEs: Latent perturbation

Slowly increase or decrease a **single latent variable**
Keep all other variables fixed

Head pose

Different dimensions of z encodes **different interpretable latent features**

VAEs: Latent perturbation

Smile

Head pose

Ideally, we want latent variables that are uncorrelated with each other

Enforce diagonal prior on the latent variables to encourage independence

Disentanglement

Latent space disentanglement with β-VAEs

β-VAE loss:

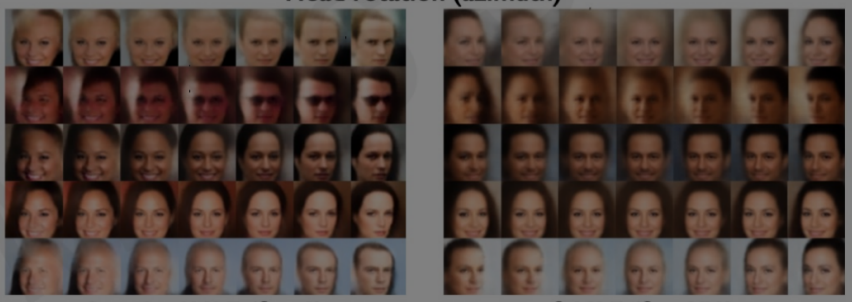
$$\mathcal{L}(\theta, \phi; \mathbf{x}, \mathbf{z}, \beta) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \,\|\, p(\mathbf{z}))$$

Reconstruction term **Regularization term**

β > 1: constrain latent bottleneck, encourage efficient latent encoding → disentanglement

Head rotation (azimuth)

Smile also changing!

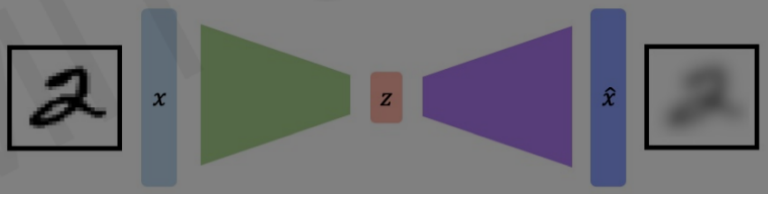


Smile relatively constant!

Standard VAE (β = 1) **β-VAE (β = 250)**

VAE summary

1. Compress representation of world to something we can use to learn
2. Reconstruction allows for unsupervised learning (no labels!)
3. Reparameterization trick to train end-to-end
4. Interpret hidden latent variables using perturbation
5. Generating new examples



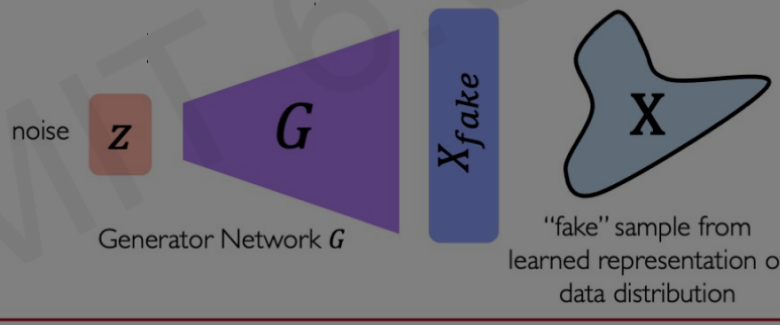
Generative Adversarial Networks (GANs)

What if we just want to sample?

Idea: don't explicitly model density, and instead just sample to generate new instances.

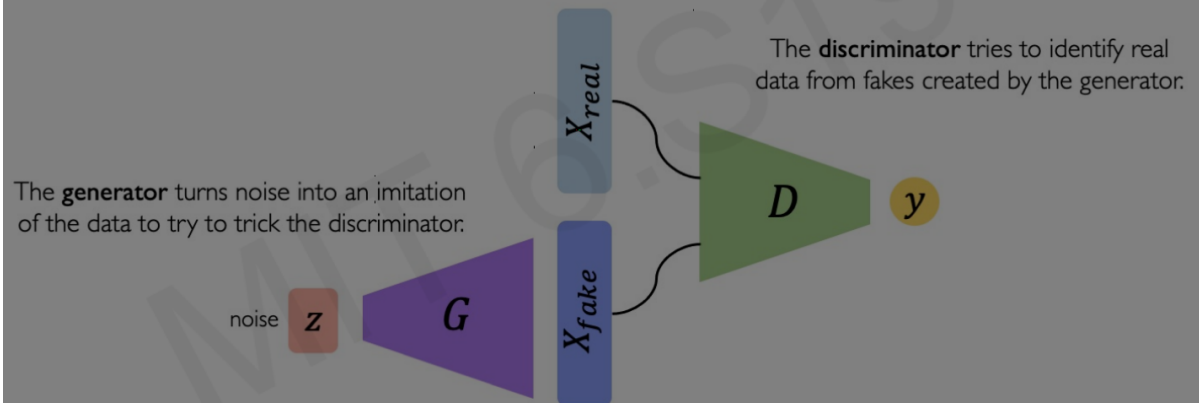
Problem: want to sample from complex distribution – can't do this directly!

Solution: sample from something simple (e.g., noise), learn a transformation to the data distribution.

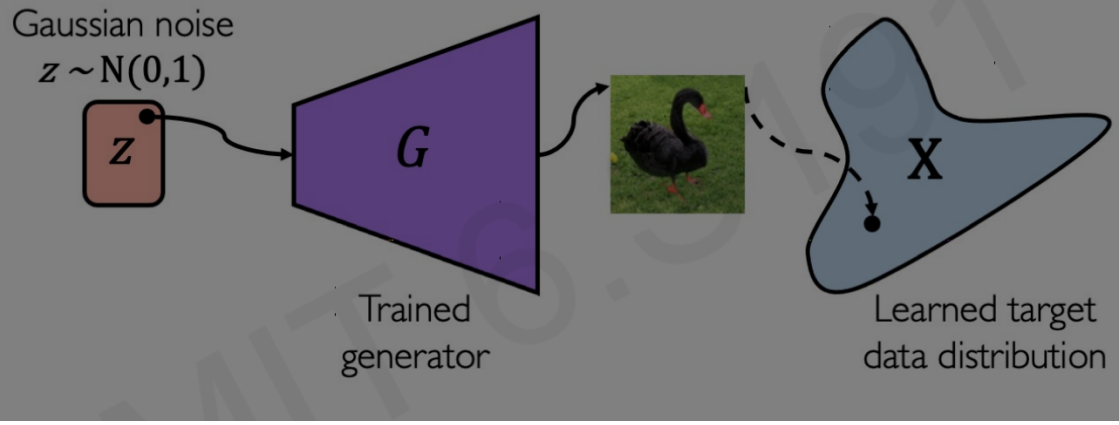


Generative Adversarial Networks (GANs)

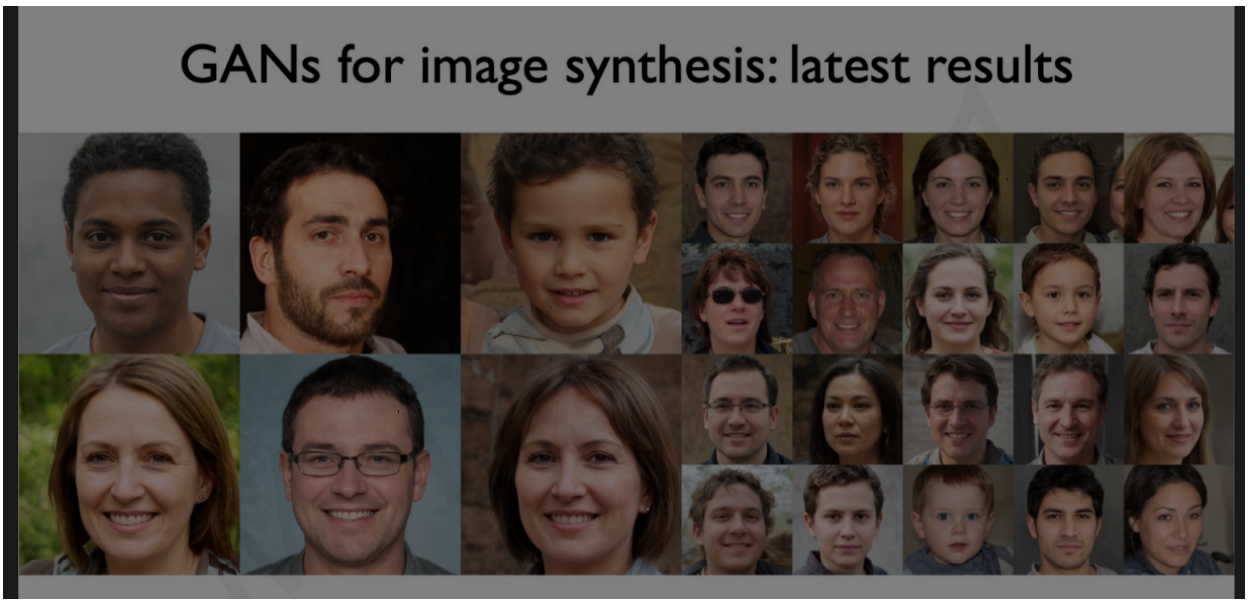
Generative Adversarial Networks (GANs) are a way to make a generative model by having two neural networks compete with each other.

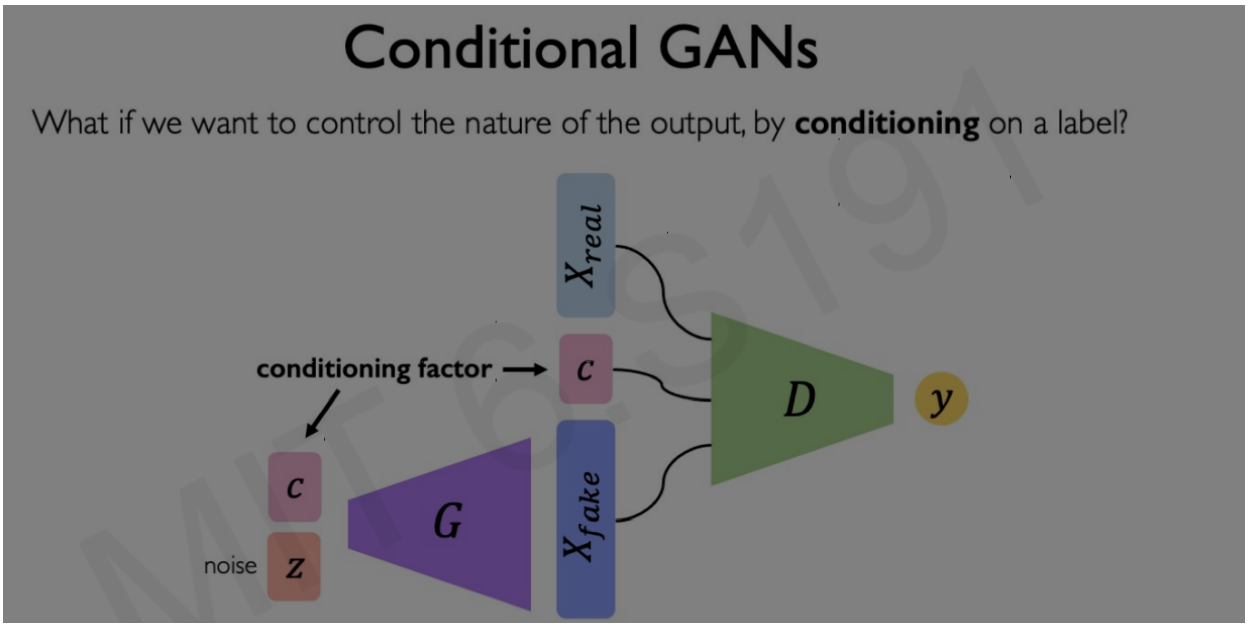


GANs are distribution transformers

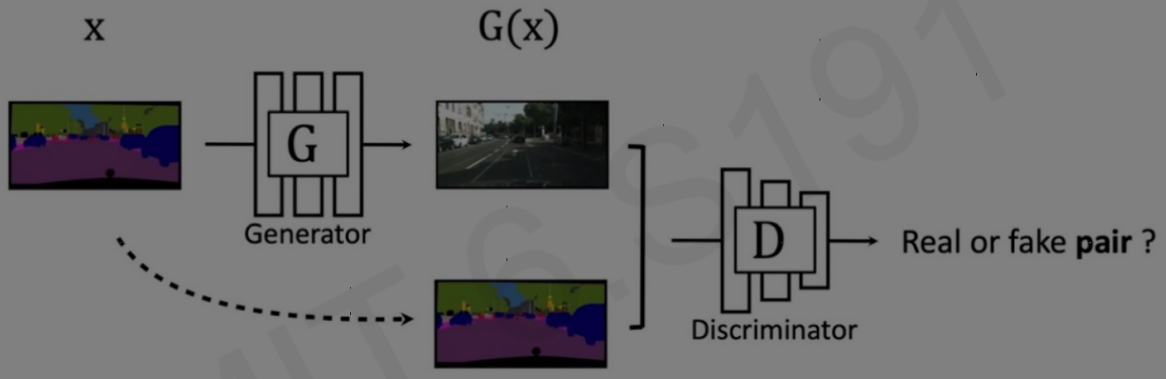


transform norm to target data distribution.





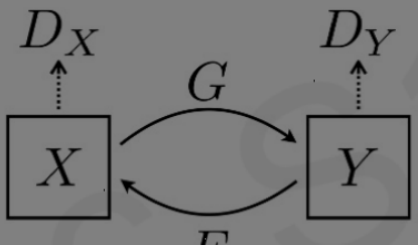
Conditional GANs and pix2pix: paired translation



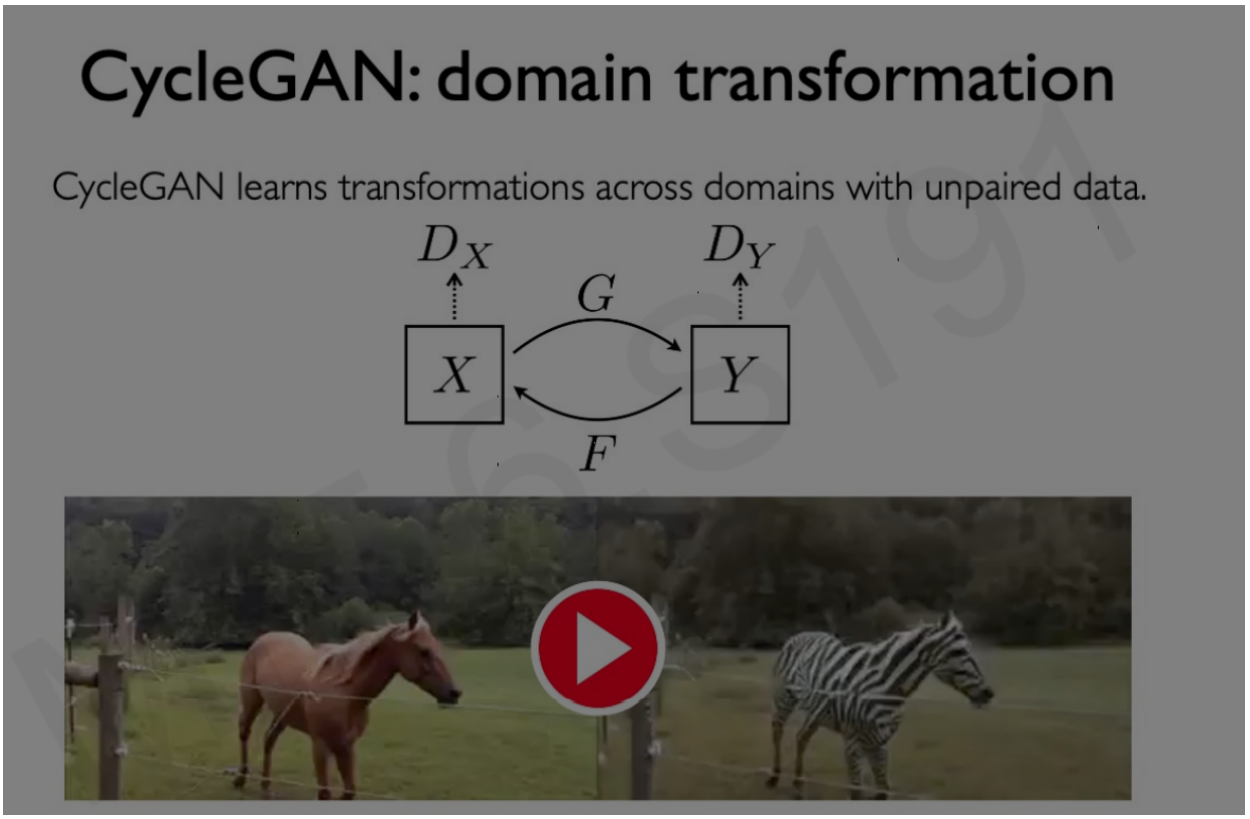
The discriminator, D, classifies between fake and real **pairs**.
The generator, G, learns to fool the discriminator.

CycleGAN: domain transformation

CycleGAN learns transformations across domains with unpaired data.

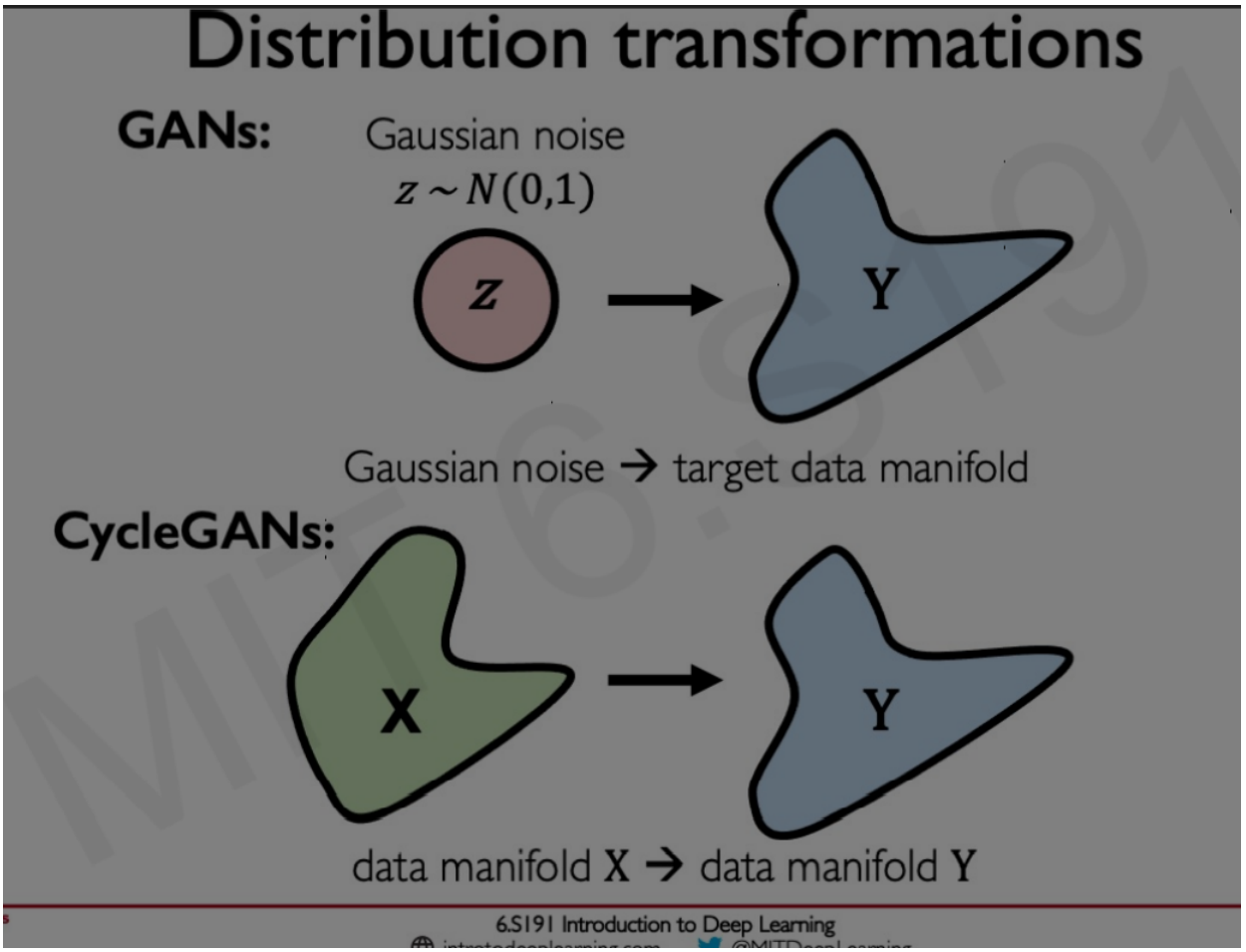


Paired translation: results



cycleGAN的研究目的是：拥有一个域X（比如斑马）和另一个域Y（比如普通马），能不能学习到一个模型，使得输入的普通马可以保持姿势和体型等内容不变的情况下，变为斑马风格呢？同样的，斑马能不能变为普通马的风格呢？域X和域Y即被看做是域转换，或者说是图图转换问题：

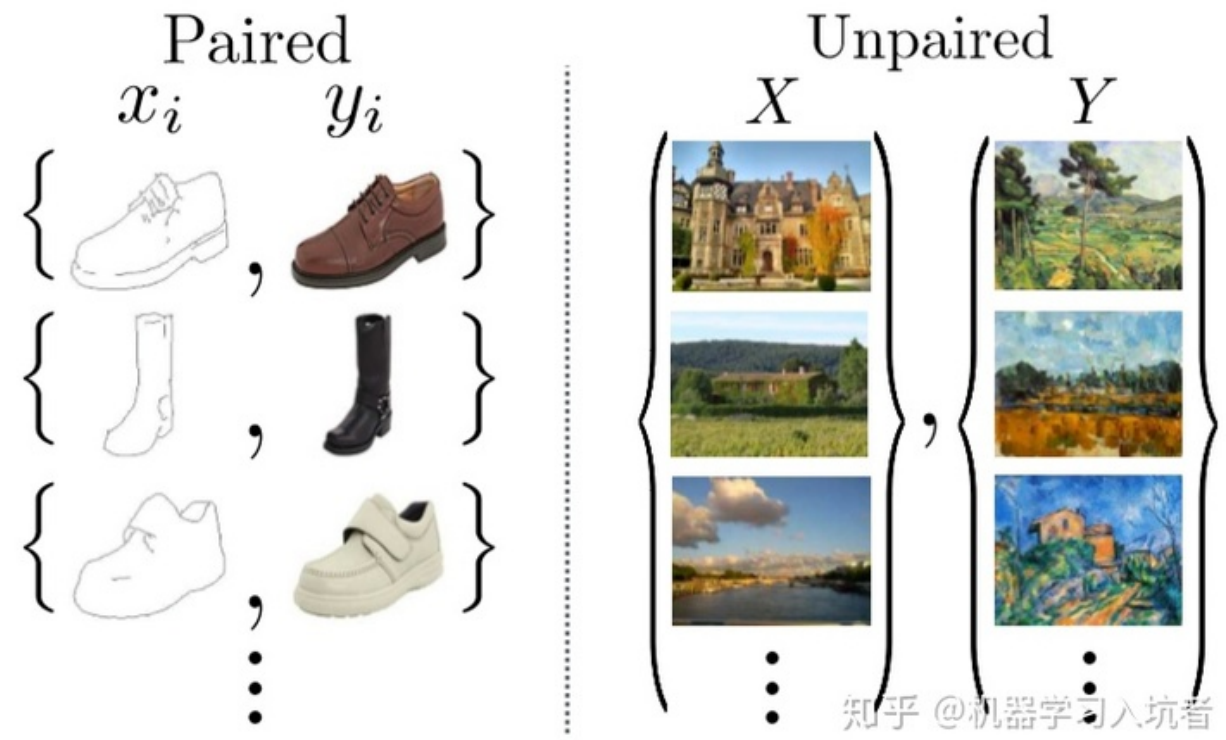
cycleGAN损失函数包含两个部分，第一部分是经典GAN网络包含的对抗损失adversarial loss，第二部分是论文提出的“cycle consistency loss”，中文译为循环一致性损失，对应上图中x映射到y^再映射回x^的过程，也就是重构损失。

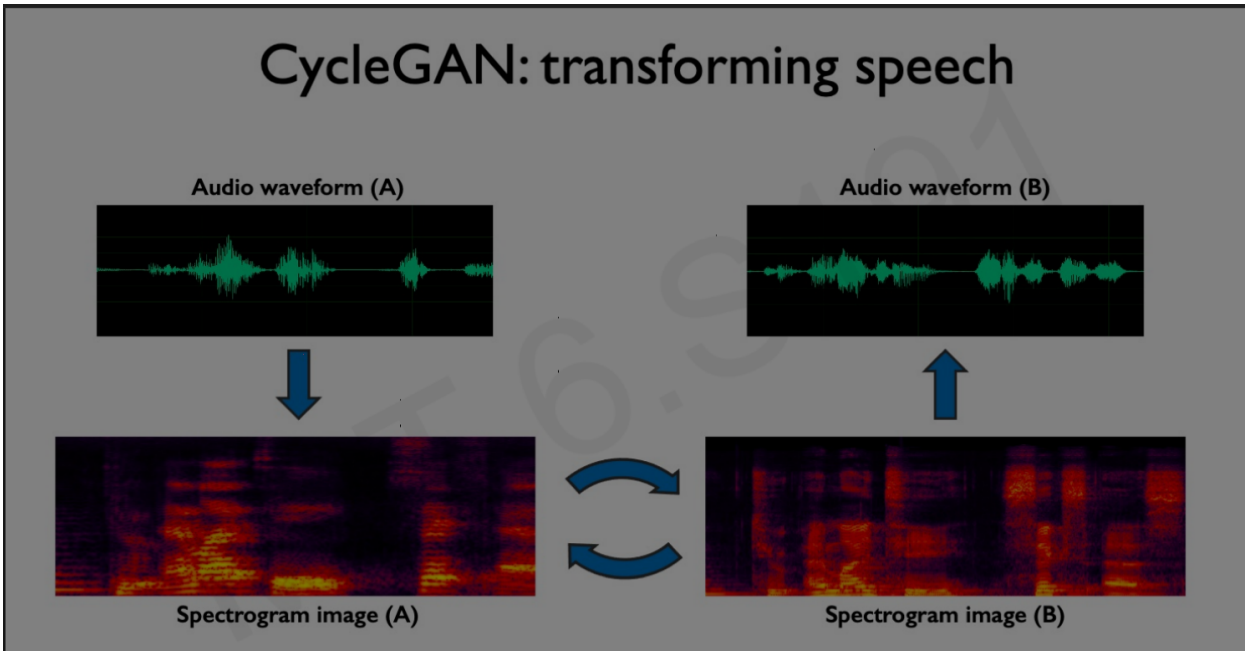


Conditional GAN V.S. Cycle GAN:

下图中左边表示成对的数据，每一张素描鞋子都有对应的真实鞋子；右边表示非成对的数据，一张现实拍摄的照片不存在和它内容相同的油画。获取严格意义上的成对数据是非常困难的，所以不依赖成对数据的算法具有非常重要的实际意义。

在cycleGAN以前就已经有使用生成对抗网络进行图图转换的论文，比如pix2pix就已经实现了基本的图图转换功能，但是pix2pix是需要成对数据的：





作者：机器学习入坑者

链接：https://zhuanlan.zhihu.com/p/92810385

来源：知乎