Generative modelling: from theory to state-of-the-art.

Arthur Conmy

Chalk Talk, 13 September 2021

Plan

What is generative modelling?
- Background and motivation.
- Statement of the problem.

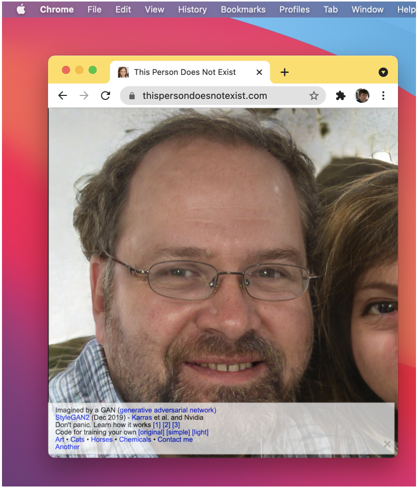
What are some high-level aspects of generative modelling?
- ‘Taxonomy’ of generative models.
- Intuitions and recurring themes.
- Architectural overview of generative models.

What are some examples of generative modelling techniques?
- GANs: a brief history.
- Diffusion models and recent advances.

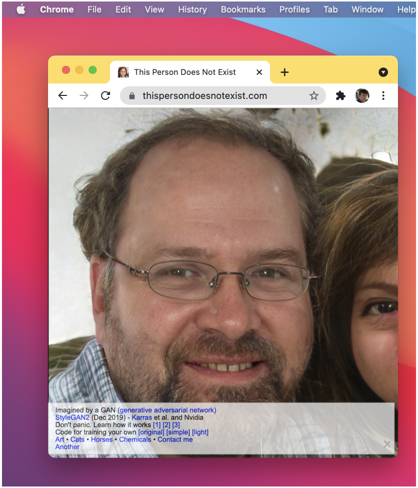
Plan

- What is generative modelling?
  - Background and motivation.
  - Statement of the problem.
- What are some high-level aspects of generative modelling?
  - ‘Taxonomy’ of generative models.
  - Intuitions and recurring themes.
  - Architectural overview of generative models.
- What are some examples of generative modelling techniques?
  - GANs: a brief history.
  - Diffusion models and recent advances.

Background and motivation.



Background and motivation.

- 

- … but how does it all work?

Statement of the problem.

- Consider a *fixed, but unknown* distribution p_d, that we have access to a large number of i.i.d samples from: $x_1, x_2, \dots, x_N \sim p_d$.

Statement of the problem.

- Consider a *fixed, but unknown* distribution p_d, that we have access to a large number of i.i.d samples from: $x_1, x_2, ..., x_N \sim p_d$.
  - Note that the x could be images, text, audio ...

Statement of the problem.

- Consider a *fixed, but unknown* distribution p_d, that we have access to a large number of i.i.d samples from: $x_1, x_2, ..., x_N \sim p_d$.
  - Note that the x could be images, text, audio ...
- Open-ended problem: how can we generate samples $x'_1, x'_2, ...$ that are similar to the samples $x_1, x_2, ..., x_N$?

Statement of the problem.

- Consider a *fixed, but unknown* distribution p_d, that we have access to a large number of i.i.d samples from: $x_1, x_2, ..., x_N \sim p_d$.
  - Note that the x could be images, text, audio ...
- Open-ended problem: how can we generate samples $x'_1, x'_2, ...$ that are similar to the samples $x_1, x_2, ..., x_N$?
  - Big question: what does 'similar' mean?

Plan

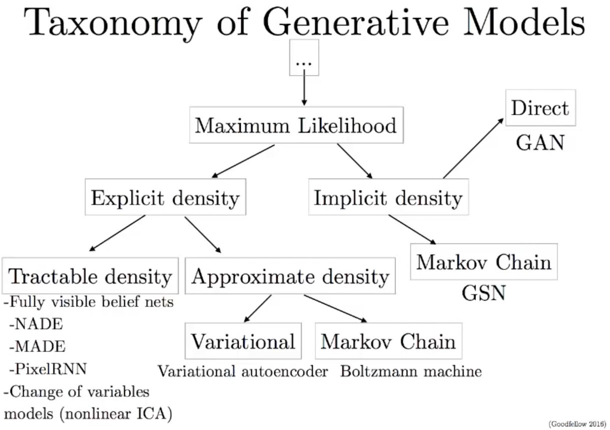
What is generative modelling?
- Background and motivation.
- Statement of the problem.

What are some high-level aspects of generative modelling?
- ‘Taxonomy’ of generative models.
- Intuitions and recurring themes.
- Architectural overview of generative models.

What are some examples of generative modelling techniques?
- GANs: a brief history.
- Diffusion models and recent advances.

‘Taxonomy’ of generative models.

From Ian Goodfellow’s 2016 NIPS GANs tutorial (perhaps slightly outdated, but great talk).



Intuitions and recurring themes.

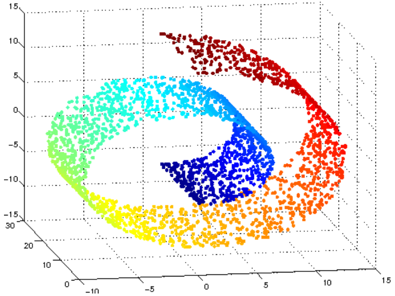
1. Real-world datasets are high-dimensional but they admit a low dimensional support (and this leads to counterintuitive behaviour).

Intuitions and recurring themes.

1. Real-world datasets are high-dimensional but they admit a low dimensional support (and this leads to counterintuitive behaviour).
   - E.g for a HQ image dataset, the number of dimensions may be $C \times H \times W = 3 \times 1024 \times 1024 = \boxed{3,145,728}$.

Intuitions and recurring themes.

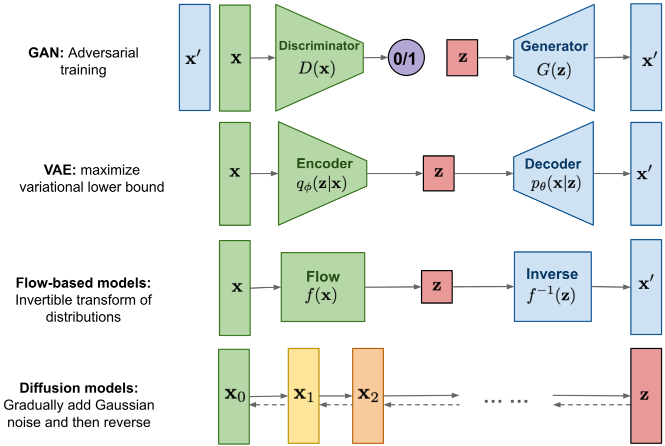
1. Real-world datasets are high-dimensional but they admit a low dimensional support (and this leads to counterintuitive behaviour).
   - E.g for a HQ image dataset, the number of dimensions may be $C \times H \times W = 3 \times 1024 \times 1024 = \boxed{3,145,728}$.
   - Lower dimensional support: connection to manifold learning:



2. When we want to represent a large class of functions $\mathcal{F}$, neural networks are a great choice.

Architectural overview of generative models.

Great resource: `https://lilianweng.github.io/lil-log/2021/07/11/diffusion-models.html`



Plan

What is generative modelling?
- Background and motivation.
- Statement of the problem.

What are some high-level aspects of generative modelling?
- 'Taxonomy' of generative models.
- Intuitions and recurring themes.
- Architectural overview of generative models.

What are some examples of generative modelling techniques?
- GANs: a brief history.
- Diffusion models and recent advances.

Where GANs came from.

The original motivation for GANs came from a game theoretic standpoint; pit two neural networks G and D against each other and define the natural analogue of cross entropy loss in this case:

$$V(D, G) = \mathbb{E}_{x \sim p_d}[\log D(x)] + \mathbb{E}_{z \sim \mathcal{N}_n(0,I)}[\log(1 - D(G(z)))] \quad (1)$$

(the distribution of generated images, $G(z)$ will be denoted p_g).

- We're already in the setting to apply back-prop, so what's wrong?

Where GANs came from.

The original motivation for GANs came from a game theoretic standpoint; pit two neural networks G and D against each other and define the natural analogue of cross entropy loss in this case:

$$V(D, G) = \mathbb{E}_{x \sim p_d}[\log D(x)] + \mathbb{E}_{z \sim \mathcal{N}_n(0,I)}[\log(1 - D(G(z)))] \quad (1)$$

(the distribution of generated images, $G(z)$ will be denoted p_g).

- We're already in the setting to apply back-prop, so what's wrong?
- This miserably fails.

Where GANs came from.

The original motivation for GANs came from a game theoretic standpoint; pit two neural networks G and D against each other and define the natural analogue of cross entropy loss in this case:

$$V(D,G) = \mathbb{E}_{x \sim p_d}[\log D(x)] + \mathbb{E}_{z \sim \mathcal{N}_n(0,I)}[\log(1 - D(G(z)))] \quad (1)$$

(the distribution of generated images, $G(z)$ will be denoted p_g).

- We're already in the setting to apply back-prop, so what's wrong?
- This miserably fails.
- More theoretical analysis leads to modifying the V above to fix the vanishing gradients problem.

Where GANs came from.

The original motivation for GANs came from a game theoretic standpoint; pit two neural networks G and D against each other and define the natural analogue of cross entropy loss in this case:

$$V(D, G) = \mathbb{E}_{x \sim p_d}[\log D(x)] + \mathbb{E}_{z \sim \mathcal{N}_n(0,I)}[\log(1 - D(G(z)))] \quad (1)$$

(the distribution of generated images, $G(z)$ will be denoted p_g).

- We're already in the setting to apply back-prop, so what's wrong?
- This miserably fails.
- More theoretical analysis leads to modifying the V above to fix the vanishing gradients problem.
- However, the training remains unstable, and highly dependent on heuristics and parameter tuning.

Picture interlude: what we're approaching.

Wasserstein and theoretically principled GANs.

Reference: Arjovsky, Chintala and Bottou (2017) and Gulrajani et al (2017).

- Given the true distribution P_r and a generated distribution P_g, optimize

$$\mathcal{L}(p_r, p_g) \tag{2}$$

  where $\mathcal{L}$ is some loss function between probability distributions.

Wasserstein and theoretically principled GANs.

Reference: Arjovsky, Chintala and Bottou (2017) and Gulrajani et al (2017).

- Given the true distribution P_r and a generated distribution P_g, optimize

$$\mathcal{L}(p_r, p_g) \tag{2}$$

  where $\mathcal{L}$ is some loss function between probability distributions.

  - $\mathcal{L}$ needs to be estimable from iid samples.

Wasserstein and theoretically principled GANs.

Reference: Arjovsky, Chintala and Bottou (2017) and Gulrajani et al (2017).

- Given the true distribution P_r and a generated distribution P_g, optimize
$$\mathcal{L}(p_r, p_g) \tag{2}$$
where $\mathcal{L}$ is some loss function between probability distributions.
    - $\mathcal{L}$ needs to be estimable from iid samples.
    - $\mathcal{L}$ needs to be differentiable.

Wasserstein and theoretically principled GANs.

Reference: Arjovsky, Chintala and Bottou (2017) and Gulrajani et al (2017).

- Given the true distribution P_r and a generated distribution P_g, optimize

$$\mathcal{L}(p_r, p_g) \tag{2}$$

  where $\mathcal{L}$ is some loss function between probability distributions.
  - $\mathcal{L}$ needs to be estimable from iid samples.
  - $\mathcal{L}$ needs to be differentiable.
  - This leaves a lot of possiblities!

Wasserstein and theoretically principled GANs.

Reference: Arjovsky, Chintala and Bottou (2017) and Gulrajani et al (2017).

- Given the true distribution P_r and a generated distribution P_g, optimize
$$\mathcal{L}(p_r, p_g) \tag{2}$$
where $\mathcal{L}$ is some loss function between probability distributions.
  - $\mathcal{L}$ needs to be estimable from iid samples.
  - $\mathcal{L}$ needs to be differentiable.
  - This leaves a lot of possiblities!
- The cross-entropy loss on the previous slide leads to GANs minimising the **Jensen-Shannon** divergence $\mathcal{L}_{JS}$ between the distributions. D_{KL} fixes vanishing gradients.

So, which loss function?

- Define the KL divergence as

$$D_{\mathrm{KL}}(P \parallel Q) = \int_{\mathbb{R}^n} p(x) \log \left(\frac{p(x)}{q(x)} \right) \, dx \tag{3}$$

then the Jensen-Shannon divergence is

$$\mathrm{JSD}(P \parallel Q) = \frac{1}{2} D(P \parallel M) + \frac{1}{2} D(Q \parallel M) \tag{4}$$

where $M = \frac{1}{2}(P + Q)$.

So, which loss function?

- Define the KL divergence as

$$D_{\mathrm{KL}}(P \parallel Q) = \int_{\mathbb{R}^n} p(x) \log \left(\frac{p(x)}{q(x)} \right) \, dx \tag{3}$$

then the Jensen-Shannon divergence is

$$\mathrm{JSD}(P \parallel Q) = \frac{1}{2} D(P \parallel M) + \frac{1}{2} D(Q \parallel M) \tag{4}$$

where $M = \frac{1}{2}(P + Q)$.

- A much better choice is the Wasserstein, or so-called Earth-Mover distribution between distributions.

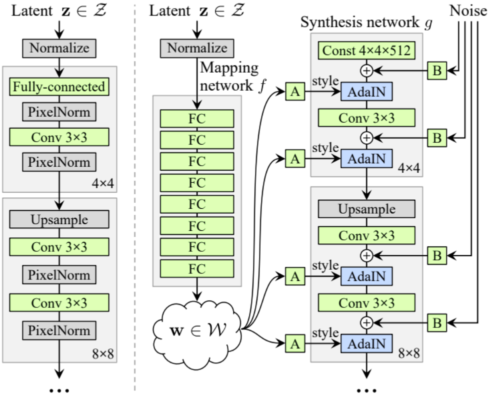
$$\mathrm{EMD}(P_r, P_\theta) = \sup_{\|f\|_{L \leq 1}} \mathbb{E}_{x \sim P_r} f(x) - \mathbb{E}_{x \sim P_\theta} f(x). \tag{5}$$

The StyleGAN Architecture.

- This is what's behind `ThisPersonDoesNotExist.com`!

The StyleGAN Architecture.

- This is what's behind `ThisPersonDoesNotExist.com`!



(a) Traditional (b) Style-based generator

-

Diffusion models

This is a different approach to dealing with the low-dimensional problem.

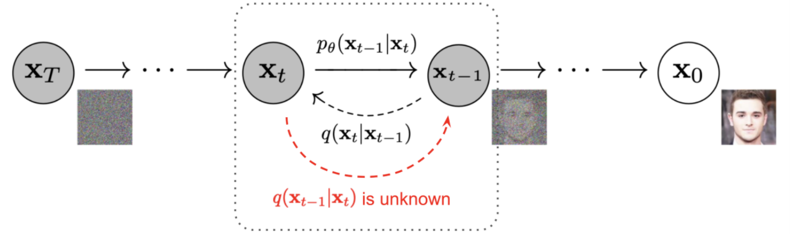


Fig. 2. The Markov chain of forward (reverse) diffusion process of generating a sample by slowly adding (removing) noise. (Image source: Ho et al. 2020 with a few additional annotations)

Diffusion models