Két generatív modell

A Variational Autoencoder és a GAN

Csiszárik Adrián

ELTE TTK Deep Learning Szeminárium, 2018

Felügyelt tanulás vs. felügyelet nélküli tanulás

Felügyelt tanulás (Supervised learning)

Adat: $\{(x_i, y_i)\}_{i=1}^{N}$

Cél: Megtanulni egy $f : X \mapsto Y$ leképezést, amely illeszkedik az adatra (és jól általánosít).

Példák: Klasszifikáció, regresszió, szemantikus szegmentáció, objektum detektálás.

Felügyelt tanulás vs. felügyelet nélküli tanulás

Felügyelt tanulás (Supervised learning)

Adat: $\{(x_i, y_i)\}_{i=1}^N$

Cél: Megtanulni egy $f : X \mapsto Y$ leképezést, amely illeszkedik az adatra (és jól általánosít).

Példák: Klasszifikáció, regresszió, szemantikus szegmentáció, objektum detektálás.

Felügyelet nélküli tanulás (Unsupervised learning)

Adat: $\{(x_i)\}_{i=1}^N$, nincs y.

Cél: Modellezni a $p(x)$ eloszlást, megtanulni az adathalmazban lévő rejtett struktúrát.

Példák: Klaszterezés, dimenzióredukció, generatív modellek.

Felügyelt tanulás vs. felügyelet nélküli tanulás

Felügyelt tanulás (Supervised learning)

Adat: $\{(x_i, y_i)\}_{i=1}^N$

Cél: Megtanulni egy $f : X \mapsto Y$ leképezést, amely illeszkedik az adatra (és jól általánosít).

Példák: Klasszifikáció, regresszió, szemantikus szegmentáció, objektum detektálás.

A felügyelt tanulási módszerek ma már hatékonyan működnek számos modellezési feladatban. (Neuronhálók: Narrow AI.)

Felügyelet nélküli tanulás (Unsupervised learning)

Adat: $\{(x_i)\}_{i=1}^N$, nincs y.

Cél: Modellezni a $p(x)$ eloszlást, megtanulni az adathalmazban lévő rejtett struktúrát.

Példák: Klaszterezés, dimenzióredukció, generatív modellek.

Nagyon jó lenne, erősen kutatott terület.
(Neuronhálók: úton a General vagy strong AI felé.)

Felügyelt tanulás vs. felügyelet nélküli tanulás

Felügyelt tanulás (Supervised learning)

$\$\$\$$Drága$\$\$\$$

Adat: $\{(x_i, y_i)\}_{i=1}^N$

Cél: Megtanulni egy $f : X \mapsto Y$ leképezést, amely illeszkedik az adatra (és jól általánosít).

Példák: Klasszifikáció, regresszió, szemantikus szegmentáció, objektum detektálás.

A felügyelt tanulási módszerek ma már hatékonyan működnek számos modellezési feladatban. (Neuronhálók: Narrow AI.)

Felügyelet nélküli tanulás (Unsupervised learning)

Olcsó, van bőven

Adat: $\{(x_i)\}_{i=1}^N$, nincs y.

Cél: Modellezni a $p(x)$ eloszlást, megtanulni az adathalmazban lévő rejtett struktúrát.

Példák: Klaszterezés, dimenzióredukció, generatív modellek.

Nagyon jó lenne, erősen kutatott terület.
(Neuronhálók: úton a General vagy strong AI felé.)

Felügyelet nélküli tanulás (Unsupervised learning)

$$p(x)$$

Ha $p(x)$ van, minden van.

(Akár $p(x, y)$ is, valamilyen y címkékre.)

(Persze ahhoz, hogy használjuk, ezt ki kell faktorálnunk.)

(Viszont vélhetően kevesebb szupervíziós jellel meg tudjuk ezt tenni.)

Felügyelt tanulás vs. felügyelet nélküli tanulás

Még több motiváció:

- Nincs minden becímkézve, és rengeteg címkézetlen adat van.
- Gazdagabb reprezentációs struktúra jön (jöhet!) létre, ha nem csak a feladat szempontjából kapja a szupervíziós jelet a tanulás során a modell.

Konkrét feladatok:

- Density estimation.
- Outlier detection.
- Semi-supervised learning.
- Unsupervised pre-training.
- Generatív modellek.

Generatív modellek

Feladat: Adott $\{x_1, ..., x_N\}$ minta valamilyen $p(x)$ eloszlásból.
Generáljunk elemeket $p(x)$ eloszlás szerint.

Generatív modellek

Feladat: Adott $\{x_1, ..., x_N\}$ minta valamilyen $p(x)$ eloszlásból.
Generáljunk elemeket $p(x)$ eloszlás szerint.

Egy természetes megközelítés:
Legnagyobb valószínűség módszer: a parametrikus modellünk θ paramétereit hangoljuk úgy, hogy a mintánknak (tanítópontoknak) minél nagyobb legyen a valószínűsége.

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \log p_\theta(x_1, x_2, ..., x_n) = \sum_{i=1}^{N} \log p_\theta(x_i)$$

Generatív modellek

Feladat: Adott $\{x_1, ..., x_N\}$ minta valamilyen $p(x)$ eloszlásból.
Generáljunk elemeket $p(x)$ eloszlás szerint.

Egy természetes megközelítés:
Legnagyobb valószínűség módszer: a parametrikus modellünk θ paramétereit hangoljuk úgy, hogy a mintánknak (tanítópontoknak) minél nagyobb legyen a valószínűsége.

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \log p_\theta(x_1, x_2, ..., x_n) = \sum_{i=1}^{N} \log p_\theta(x_i)$$

Elvárás: persze itt is szeretnénk, ha általánosítana a modellünk, azaz nem csak a bemagolt tanítópontokat adná vissza.
Mire jó?

- Szimulációk a megtanult $p_\theta(x)$ segítségével.
- Példák generálása tanulási feladatokhoz.
- Density estimation, ha meg tudjuk kérdezni a modelltől, hogy mennyi $p(x)$.

Rejtett változós valószínűségi modell

Rejtett változós valószínűségi modell

Feltételezzük, hogy az $\{x_i\}_{i=1}^N$ tanítópontok egy rejtett (látens) z reprezentáció alapján vannak generálva.

A $p(x, z)$ együttes eloszlást tekintjük
a látens változók z vektorával.

Rejtett változós valószínűségi modell

Rejtett változós valószínűségi modell

Feltételezzük, hogy az $\{x_i\}_{i=1}^N$ tanítópontok egy rejtett (látens) z reprezentáció alapján vannak generálva.

A $p(x, z)$ együttes eloszlást tekintjük
a látens változók z vektorával.

A Variational Autoencoder és a GAN esetében is

a z explicit megjelenik.

Rejtett változós valószínűségi modell

Rejtett változós valószínűségi modell

Feltételezzük, hogy az $\{x_i\}_{i=1}^N$ tanítópontok egy rejtett (látens) z reprezentáció alapján vannak generálva.

A $p(x, z)$ együttes eloszlást tekintjük
a látens változók z vektorával.

A Variational Autoencoder és a GAN esetében is

a z explicit megjelenik.

$p(x)$-ből nehéz mintavételezni, ez az a bonyolult eloszlás amit modellezni akarunk.
$p(z)$: feltételezünk valamilyen számunkra elérhető eloszlást, amiből tudunk mintavételezni. Például: $p(z) \sim \mathcal{N}(0, I)$.

Csináljunk z-kből x-eket egy neuronhálóval.

Rejtett változós valószínűségi modell

Rejtett változós valószínűségi modell

Feltételezzük, hogy az $\{x_i\}_{i=1}^{N}$ tanítópontok egy rejtett (látens) z reprezentáció alapján vannak generálva.

A $p(x, z)$ együttes eloszlást tekintjük
a látens változók z vektorával.

A Variational Autoencoder és a GAN esetében is

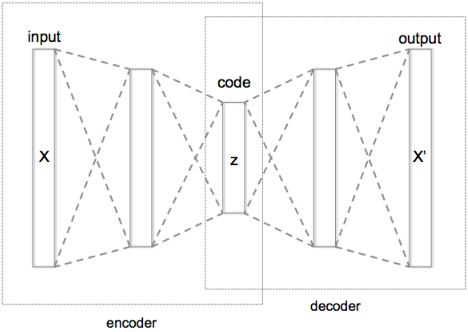
a z explicit megjelenik.

$p(x)$-ből nehéz mintavételezni, ez az a bonyolult eloszlás amit modellezni akarunk.
$p(z)$: feltételezünk valamilyen számunkra elérhető eloszlást, amiből tudunk mintavételezni. Például: $p(z) \sim \mathcal{N}(0, I)$.

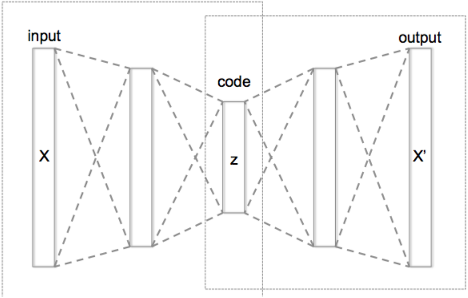
Csináljunk z-kből x-eket egy neuronhálóval.Wow.

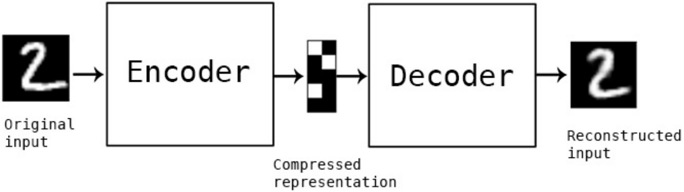
$$f_\theta(z) = x$$

Az autoencoder struktúra

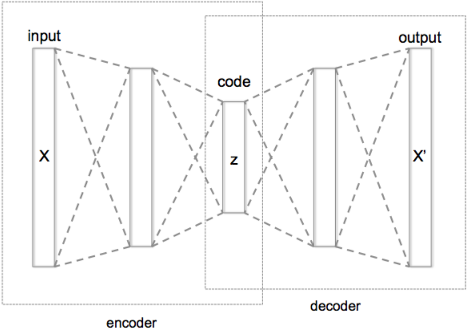


Az autoencoder struktúra





Az autoencoder struktúra

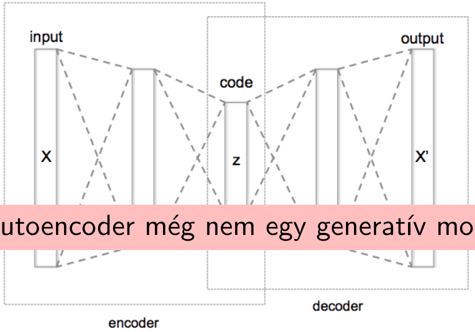


Nemlinearitások nélkül: kapcsolat PCA-val

Variációk, különböző megszorításokkal:

- Sparse autoencoder (szűkületben l_0 vagy l_1 sparsitás)
- Denoising autoencoder
- Variational Autoencoder

Az autoencoder struktúra



Az autoencoder még nem egy generatív modell.

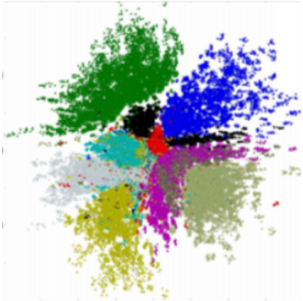
Nemlinearitások nélkül: kapcsolat PCA-val
Variációk, különböző megszorításokkal:

- Sparse autoencoder (szűkületben l_0 vagy l_1 sparsitás)
- Denoising autoencoder
- Variational Autoencoder

Látens tér struktúra mintavételezéshez

Jó

Nem jó



A Variational Autoencoder

A tanítás a $\log p_\theta(x)$ maximalizálásával történik a tanítóhalmazon.
A z látens teret formáljuk normálissá: $p(z) \sim \mathcal{N}(0, I)$ prior.
A látens változós modellben:

$$\log p_\theta(x) = \log \int p_\theta(x, z) dz$$

A Variational Autoencoder

A tanítás a $\log p_\theta(x)$ maximalizálásával történik a tanítóhalmazon.
A z látens teret formáljuk normálissá: $p(z) \sim \mathcal{N}(0, I)$ prior.
A látens változós modellben:

$$\log p_\theta(x) = \log \int p_\theta(x, z) dz$$

Rossz hír: ez intraktábilis.
A legtöbb z-re viszont a $p_\theta(x|z)$ tipikusan nulla vagy $\ll \epsilon$.

A Variational Autoencoder

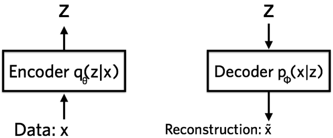
A tanítás a $\log p_\theta(x)$ maximalizálásával történik a tanítóhalmazon.
A z látens teret formáljuk normálissá: $p(z) \sim \mathcal{N}(0, I)$ prior.
A látens változós modellben:

$$\log p_\theta(x) = \log \int p_\theta(x, z) dz$$

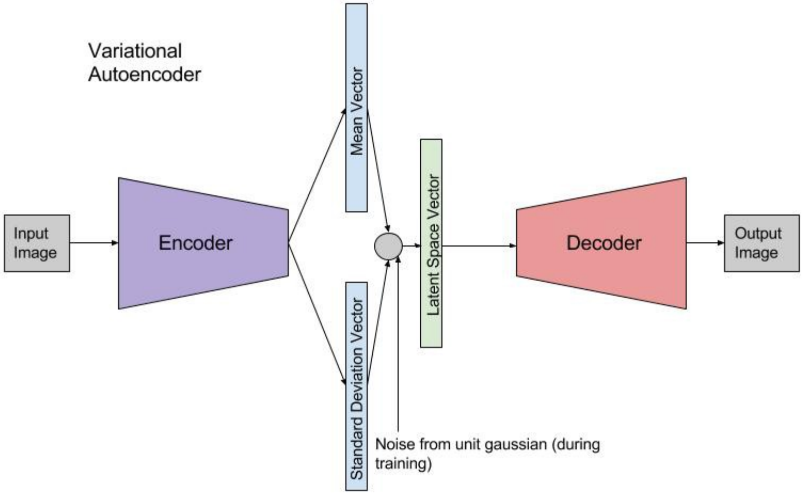
Rossz hír: ez intraktábilis.
A legtöbb z-re viszont a $p_\theta(x|z)$ tipikusan nulla vagy $\ll \epsilon$.

A Variational Autoencoder két komponensből áll:

- Decoder: $p_\theta(x|z)$, ennek segítségével modellezzzük
  a $p(x|z)$ eloszlást, a likelihoodot.
- Encoder: $q_\phi(z|x)$, ennek segítségével modellezzük
  a $p_\theta(z|x)$ eloszlást, a posteriort.



VAE



Variational
Autoencoder
Input
Image
Encoder
Mean Vector
Standard Deviation Vector
Latent Space Vector
Decoder
Output
Image
Noise from unit gaussian (during
training)

Az Evidence Lower BOund (ELBO)

$$\log p_\theta(x_i) = \mathop{\mathbb{E}}_{z\sim q_\phi(z|x_i)}[\log p_\theta(x_i)]$$

$$= \mathop{\mathbb{E}}_{z}[\log \frac{p_\theta(x_i|z)p_\theta(z)}{p_\theta(z|x_i)}] \text{ (Bayes)}$$

$$= \mathop{\mathbb{E}}_{z}[\log \frac{p_\theta(x_i|z)p_\theta(z)}{p_\theta(z|x_i)} \frac{q_\phi(z|x_i)}{q_\phi(z|x_i)}] \text{ (Szorozva 1-el)}$$

$$= \mathop{\mathbb{E}}_{z}[\log p_\theta(x_i|z)] - \mathop{\mathbb{E}}_{z}[\log \frac{q_\phi(z|x_i)}{p_\theta(z)}] + \mathop{\mathbb{E}}_{z}[\log \frac{q_\phi(z|x_i)}{p_\theta(z|x_i)}]$$

$$= \underbrace{\mathop{\mathbb{E}}_{z}[\log p_\theta(x_i|z)]}_{\substack{\text{rekonstrukciós hiba,}\\ \text{mintavételezéssel}\\ \text{tudjuk közelíteni}\\ \text{(reparam. trick)}}} - \underbrace{\mathrm{KL}(q_\phi(z|x_i) \parallel p_\theta(z))}_{\substack{\text{A posteriort}\\ \text{a priorba húzó erő.}\\ \text{Ha p(z) normális,}\\ \text{van zárt alakja.}}} + \underbrace{\mathrm{KL}(q_\phi(z|x_i) \parallel p_\theta(z|x_i))}_{\substack{\text{Még mindig nehéz,}\\ \text{viszont } \geq 0\text{, dobjuk el,}\\ \text{és akkor van egy alsó becslésünk,}\\ \text{így kapjuk az ELBO-t}}}$$

[1] Diederik P Kingma, Max Welling: "Auto-Encoding Variational Bayes", ICLR 2014 https://arxiv.org/abs/1312.6114

Az Evidence Lower BOund (ELBO)

$$\log p_\theta(x_i) = \mathop{\mathbb{E}}_{z \sim q_\phi(z|x_i)} [\log p_\theta(x_i)]$$

$$= \mathbb{E}[\log \frac{p_\theta(x_i|z)p_\theta(z)}{\dots}] \text{ (Bayes)}$$

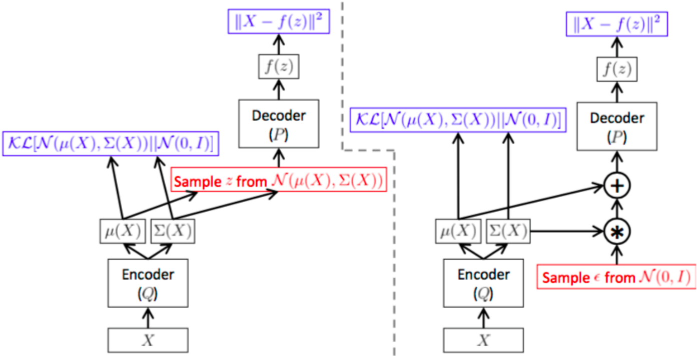
```python
# instantiate VAE model
vae = Model(x, x_decoded_mean)

# Compute VAE loss
xent_loss = original_dim * metrics.binary_crossentropy(x, x_decoded_mean)
kl_loss = - 0.5 * K.sum(1 + z_log_var - K.square(z_mean) - K.exp(z_log_var), axis=-1)
vae_loss = K.mean(xent_loss + kl_loss)
```

$$= \underbrace{\mathop{\mathbb{E}}_{z}[\log p_\theta(x_i|z)]}_{\substack{\text{rekonstrukciós hiba,} \\ \text{mintavételezésel} \\ \text{tudjuk közelíteni} \\ \text{(reparam. trick)}}} - \underbrace{\mathrm{KL}(q_\phi(z|x_i) \parallel p_\theta(z))}_{\substack{\text{A posteriort} \\ \text{a priorba húzó erő.} \\ \text{Ha p(z) normális,} \\ \text{van zárt alakja.}}} + \underbrace{\mathrm{KL}(q_\phi(z|x_i) \parallel p_\theta(z|x_i))}_{\substack{\text{Még mindig nehéz,} \\ \text{viszont } \geq 0 \text{, dobjuk el,} \\ \text{és akkor van egy alsó becslésünk,} \\ \text{így kapjuk az ELBO-t}}}$$

[1] Diederik P Kingma, Max Welling: "Auto-Encoding Variational Bayes", ICLR 2014 https://arxiv.org/abs/1312.6114

VAE - Reparametrization Trick



Az ELBO különböző alakjai

Az ELBO

$$\log p_\theta(x) = \log \int p_\theta(x,z)dz = \log \int q_\theta(z|x)\frac{p_\theta(x,z)}{q_\theta(z|x)}dz$$

$$= \log \mathop{\mathbb{E}}_{q_\theta(z|x)} \frac{p_\theta(x,z)}{q_\theta(z|x)} \geq \mathop{\mathbb{E}}_{q_\theta(z|x)} \log \frac{p_\theta(x,z)}{q_\theta(z|x)} = ELBO$$

Átlagos negatív energia + $\mathbb{H}(q_\theta(z|x))$ alak

$$ELBO = \mathop{\mathbb{E}}_{q_\theta(z|x)} \log \frac{p_\theta(x,z)}{q_\theta(z|x)} = \mathop{\mathbb{E}}_{q_\theta(z|x)} \log p_\theta(x,z) - \mathop{\mathbb{E}}_{q_\theta(z|x)} \log q_\theta(z|x)$$

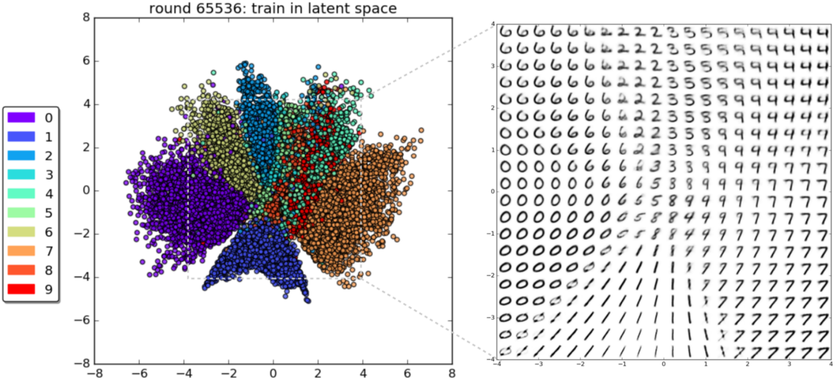
$$= \underbrace{\mathop{\mathbb{E}}_{q_\theta(z|x)} \log(p_\theta(x|z)p_\theta(z))}_{p_\theta(z)\text{ legyen nagy ott, ahol jól rekonstruál}} \underbrace{- \int q_\theta(z|x)\log q_\theta(z|x)dz}_{\mathbb{H}(q_\theta(z|x))}$$

Az ELBO különböző alakjai

Pontonkénti rekonstrukció – $q_\theta(z_i|x_i)$ priorhoz húzása

$$\begin{aligned} ELBO &= \mathop{\mathbb{E}}_{q_\theta(z|x)} \log \frac{p_\theta(x,z)}{q_\theta(z|x)} \\ &= \frac{1}{N}\sum_{i=1}^{N} \mathop{\mathbb{E}}_{q_\theta(z_i|x_i)} [\log p(x_i, z_i)] - \mathrm{KL}(q_\theta(x_i|z_i) \parallel p(z_i)) \end{aligned}$$

Látens tér a Variational Autoencoderben



- Aritmetika a látens térben
- Diagonális prior $p(z)$-n, dekorrálált látens jellemzők
- Van inferencia modellünk: $q_\theta(z|x)$
- Van likelihood modellünk $p_\phi(x|z)$ segítségével
- Előrelépés: komplexebb, jobban illeszkedő priorok, jobb inferencia modell mint a dekorrelált Gaussian

VAE példa program

https://github.com/keras-team/keras/blob/master/examples/variational_autoencoder.py

GAN

Generatív modellek

Feladat: Adott $\{x_1, ..., x_N\}$ minta valamilyen $p(x)$ eloszlásból.
Generáljunk elemeket $p(x)$ eloszlás szerint.

Generatív modellek

Feladat: Adott $\{x_1, ..., x_N\}$ minta valamilyen $p(x)$ eloszlásból.
Generáljunk elemeket $p(x)$ eloszlás szerint.

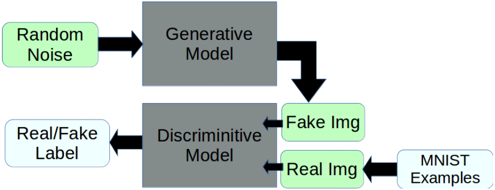
Engedjük el, hogy legyen explicit sűrűség modellünk. Csak generálni akarunk.

Generatív modellek

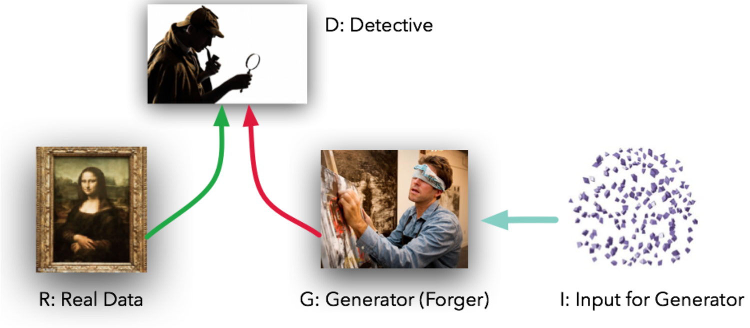
Feladat: Adott $\{x_1, ..., x_N\}$ minta valamilyen $p(x)$ eloszlásból.
Generáljunk elemeket $p(x)$ eloszlás szerint.

Engedjük el, hogy legyen explicit sűrűség modellünk. Csak generálni akarunk.

- Rejtett változós modell $z \mapsto x$
- Játékelméleti megközelítés: két játékos (modell) játszik egymás ellen.
  - Generátor hálózat: próbálja átverni a diszkriminátor hálózatot úgy, hogy minél realiszikusabb képeket generál
  - Diszkriminátor hálózat: próbálja megkülönböztetni a hamis képeket a valós képektől.



GAN modell


D: Detective
R: Real Data
G: Generator (Forger)
I: Input for Generator

GAN optimalizálás

Minimax célfüggvény:

$$\min_{\theta_g} \max_{\theta_d} [\mathop{\mathbb{E}}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathop{\mathbb{E}}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))]$$

Tanítás: alternáljuk a következőket

- Gradiens emelkedés a diszkriminátoron
$$\max_{\theta_d} [\mathop{\mathbb{E}}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathop{\mathbb{E}}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))]$$
- Gradiens eredszkedés a generátoron
$$\min_{\theta_g} \mathop{\mathbb{E}}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))$$

GAN optimalizálás

Minimax célfüggvény:

$$\min_{\theta_g} \max_{\theta_d} [\mathop{\mathbb{E}}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathop{\mathbb{E}}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))]$$

Tanítás: alternáljuk a következőket

- Gradiens emelkedés a diszkriminátoron

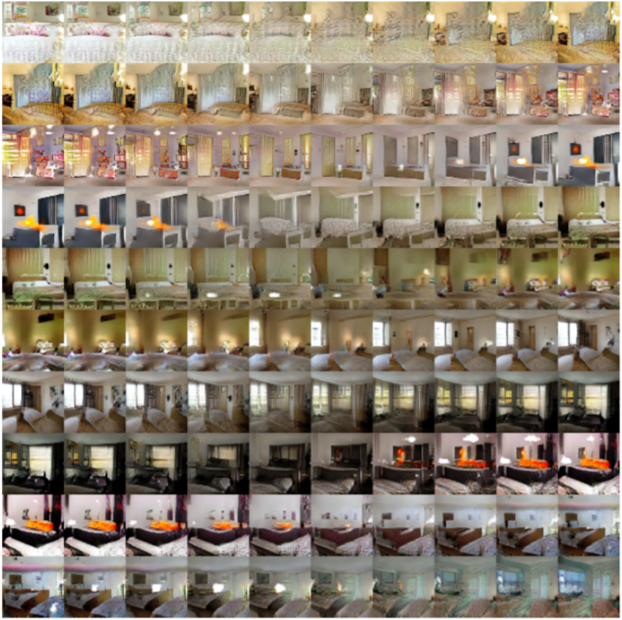
$$\max_{\theta_d} [\mathop{\mathbb{E}}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathop{\mathbb{E}}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))]$$

- Gradiens eredszkedés a generátoron

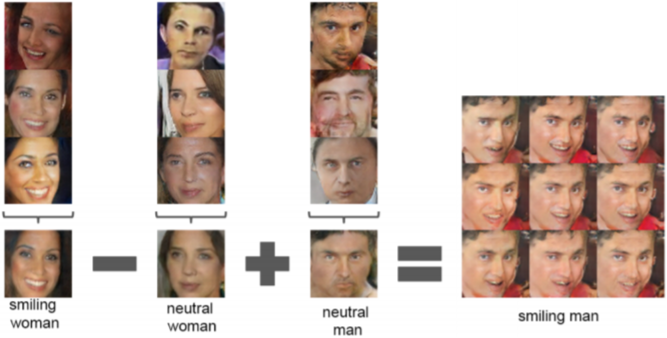
$$\min_{\theta_g} \mathop{\mathbb{E}}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))$$

- Gradiens eredszkedés a generátoron (másik változat)

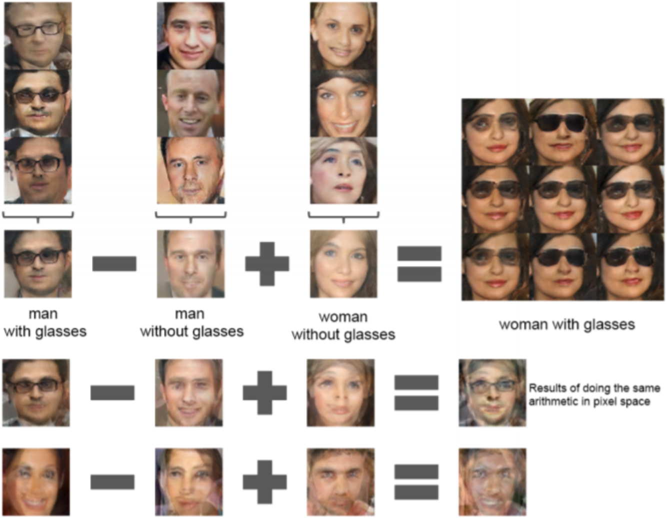
$$\max_{\theta_g} \mathop{\mathbb{E}}_{z \sim p(z)} \log D_{\theta_d}(G_{\theta_g}(z))$$



[1]Radford et al.: Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, ICLR 2016



[1] Radford et al.: Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, ICLR 2016


man
with glasses
man
without glasses
woman
without glasses
woman with glasses
Results of doing the same
arithmetic in pixel space

The GAN Zoo

https://github.com/hindupuravinash/the-gan-zoo

GAN példa program

https://github.com/eriklindernoren/Keras-GAN

Összefoglalás

- VAE (Variational Autoencoder)
  - Van inferencia modell
  - Van likelihood modell
- GAN (Generative Adversarial Networks)
  - Látványos eredmények
  - Trükkösebb, instabil tanulás
  - Nincs inferencia modell
  - Izgalmas konstrukció
  - The GAN Zoo
- Variational optimization
- Adversarial konstrukció úgy általában
- Nemlineáris PCA
- Valószínűségeloszlások közötti távolság fogalmak