DTU Course 02456 Deep learning
5 Un- and semi-supervised learning
2017 Updates

Ole Winther

Dept for Applied Mathematics and Computer Science
Technical University of Denmark (DTU)

September 24, 2017

Objectives of lectures week 5

- Pro and cons of variational auto-encoders (VAE).
- Generative adversarial networks (GAN)
- Some extensions of VAE and GAN.

VAE pro and con

- Pro: variational auto-encoder computes a lower bound on the log likelihood

$$\log p(x) \geq \int q(z|x) \log \frac{p(x|z)p(z)}{q(z|x)}$$

- Quantitative model comparison for test data x_{test}.: $\log p(x_{\text{test}})$.
- Pro: It is a generative model - we can synthesise new data

$$z \sim p(z)$$
$$x \sim p(x|z)$$

VAE pro and con

- Pro: variational auto-encoder computes a lower bound on the log likelihood

$$\log p(x) \geq \int q(z|x) \log \frac{p(x|z)p(z)}{q(z|x)}$$

- Quantitative model comparison for test data x_{test}.: $\log p(x_{\text{test}})$.
- Pro: It is a generative model - we can synthesise new data

$$z \sim p(z)$$
$$x \sim p(x|z)$$

- Con: Too restricted choice of variational distribution $\rightarrow$ $q(z|x)$ is far from

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)}$$

and the generative model we learn will be poor.
- Con: If our likelihood function $p(x|z)$ is a poor fit to reality

Generative adversarial network (GAN)

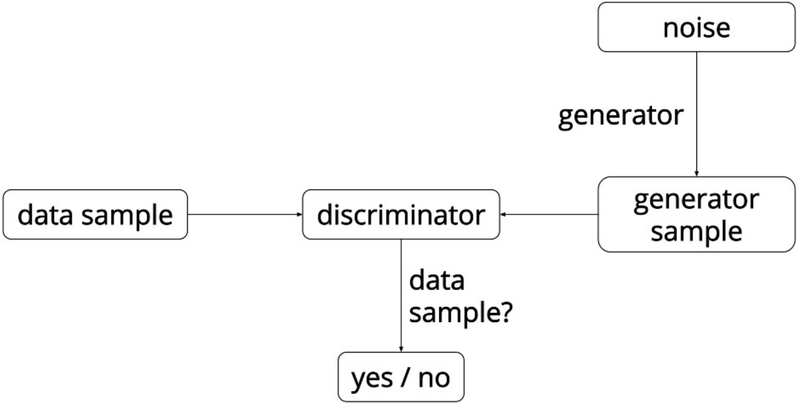
- A latent variable (implicit) generative approach
- without explicit likelihood model $p(x|z)$

Generative adversarial network (GAN)

- A latent variable (implicit) generative approach
- without explicit likelihood model $p(x|z)$
- Pro: Not dependent on (wrong) likelihood function
- Pro: Synthesise new data
- Con: No quantitative model comparison - only indirectly in for example semi-supervised learning.

Generative adversarial network (GAN)

- A latent variable (implicit) generative approach
- without explicit likelihood model $p(x|z)$
- Pro: Not dependent on (wrong) likelihood function
- Pro: Synthesise new data
- Con: No quantitative model comparison - only indirectly in for example semi-supervised learning.



GAN

- GAN has two components:
  - Non-probabilistic generative model $G(z)$:

    $$z \sim p(z)$$
    $$x = G(z)$$

    should be able generate realistic data.
  - Binary true/generated data discriminator $D(x)$ should be able distinguish real and fake data.

GAN

- GAN has two components:
  - Non-probabilistic generative model $G(z)$:

$$z \sim p(z)$$
$$x = G(z)$$

    should be able generate realistic data.
  - Binary true/generated data discriminator $D(x)$ should be able distinguish real and fake data.
- Objective:
  - Generated data - $z \sim p(z)$, $x = G(z)$:

$$\max_D \min_G \log(1 - D(G(z)))$$

  - True data x:

$$\max_D \log D(x)$$

GAN

- GAN has two components:
  - Non-probabilistic generative model $G(z)$:

$$z \sim p(z)$$
$$x = G(z)$$

    should be able generate realistic data.
  - Binary true/generated data discriminator $D(x)$ should be able distinguish real and fake data.
- Objective:
  - Generated data - $z \sim p(z)$, $x = G(z)$:

$$\max_D \min_G \log(1 - D(G(z)))$$

  - True data x:

$$\max_D \log D(x)$$

- In practice hard to solve this min max objective.
- Generates very nice images.

Generating faces with GAN

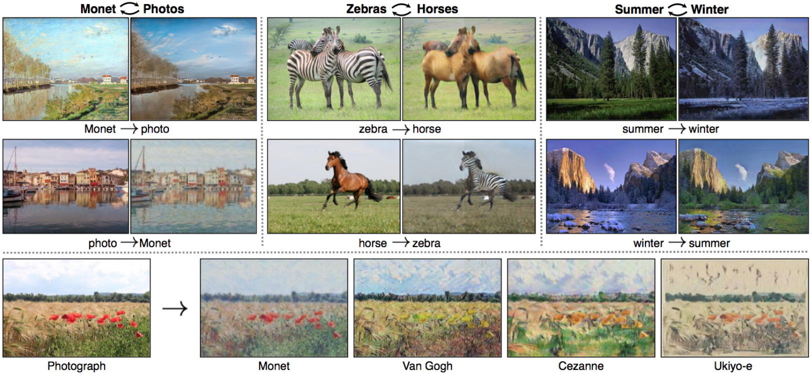
http://torch.ch/blog/2015/11/13/gan.html

Cycle GAN

- GAN very popular - really nice images.
- Many variants - here example on style transfer.
- Make two generators F and G - for example:
  - F: Van Gogh $\rightarrow$ Cezanne style
  - G: Cezanne $\rightarrow$ Van Gogh style

Cycle GAN

- GAN very popular - really nice images.
- Many variants - here example on style transfer.
- Make two generators F and G - for example:
  - F: Van Gogh $\rightarrow$ Cezanne style
  - G: Cezanne $\rightarrow$ Van Gogh style
- Objective: normal GAN +
  - Van Gogh image x: $x \approx G(F(x))$
  - Cezanne image y: $y \approx F(G(y))$.



Grammar VAE

- When data has certain structure then the generative model becomes better if we can build that into the model $\rightarrow$
- It will always generate data obeying the structure.
- Example - context-free grammar:

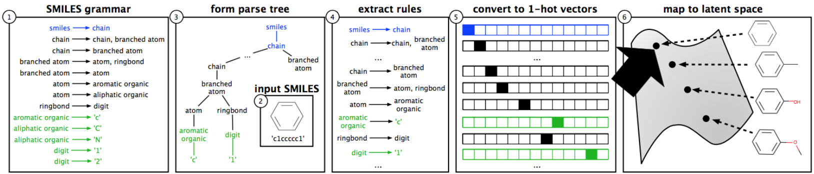
```
S → S '+' T | S '*' T | S '/' T | T
T → '(' S ')' | 'sin(' S ')' | 'exp(' S ')'
T → 'x' | '1' | '2' | '3'
```

Grammar VAE

- When data has certain structure then the generative model becomes better if we can build that into the model $\rightarrow$
- It will always generate data obeying the structure.
- Example - context-free grammar:

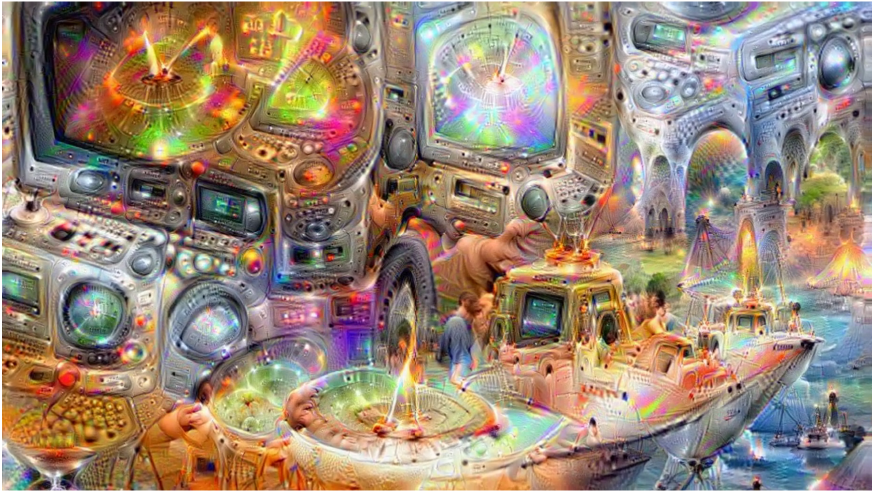
$$
\begin{aligned}
S &\rightarrow S \text{ `+' } T \mid S \text{ `*' } T \mid S \text{ `/' } T \mid T \\
T &\rightarrow \text{`(' } S \text{ `)'} \mid \text{`sin(' } S \text{ `)'} \mid \text{`exp(' } S \text{ `)'} \\
T &\rightarrow \text{`x'} \mid \text{`1'} \mid \text{`2'} \mid \text{`3'}
\end{aligned}
$$

- Encoder model for SMILES representation of molecules:



References

- D P Kingma and M Welling, Auto-Encoding Variational Bayes, 2013 https://arxiv.org/pdf/1312.6114.pdf
- D J Rezende, S Mohamed and D Wierstra, Stochastic Backpropagation and Approximate Inference in Deep Generative Models, 2014 https://arxiv.org/pdf/1401.4082.pdf
- D P Kingma, D J Rezende, S Mohamed and M Welling, Semi-Supervised Learning with Deep Generative Models, 2014 https://arxiv.org/pdf/1406.5298.pdf
- I J Goodfellow et al, Generative Adversarial Networks, 2014 https://arxiv.org/pdf/1406.2661.pdf
- S Mohamed and B Lakshminarayanan. 2016 Learning in Implicit Generative Models https://arxiv.org/pdf/1610.03483.pdf
- T Salimans et al, Improved Techniques for Training GANs, 2016 https://arxiv.org/pdf/1606.03498.pdf
- J-Y Zhu et al, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, 2017 https://arxiv.org/pdf/1703.10593.pdf
- M J Kusner, B Paige, J M Hernández-Lobato, Grammar Variational Autoencoder, 2017 https://arxiv.org/pdf/1703.01925.pdf



Thanks!
Ole Winther