Unsupervised Learning: Generation
Creation

- Generative Models: https://openai.com/blog/generative-models/

What I cannot create, I do not understand.

Richard Feynman

Creation – Image Processing

Now

In the future

Machine draws a cat

http://www.wikihow.com/Draw-a-Cat-Face
Generative Models

- PixelRNN
- Variational Autoencoder (VAE)
- Generative Adversarial Network (GAN)
PixelRNN

• To create an image, generating a pixel each time

E.g. 3 x 3 images

Can be trained just with a large collection of images without any annotation

PixelRNN


Real World
More than images ……..


Practicing Generation Models: Pokémon Creation

• Small images of 792 Pokémon's
  • Can machine learn to create new Pokémon's?

Don't catch them! Create them!

• Source of image:
  http://bulbapedia.bulbagarden.net/wiki/List_of_Pokémon_by_base_stats_(Generation_VI)

Original image is 40 x 40
Making them into 20 x 20
Practicing Generation Models: Pokémon Creation

• Tips (?)

➤ Each pixel is represented by 3 numbers (corresponding to RGB)

➤ Each pixel is represented by a 1-of-N encoding feature

Clustering the similar color 167 colors in total
Practicing Generation Models: Pokémon Creation

• Original image (40 x 40): http://speech.ee.ntu.edu.tw/~tlkagk/courses/ML_2016/Pokemon_creation/image.rar

• Pixels (20 x 20): http://speech.ee.ntu.edu.tw/~tlkagk/courses/ML_2016/Pokemon_creation/pixel_color.txt
  • Each line corresponds to an image, and each number corresponds to a pixel
    • http://speech.ee.ntu.edu.tw/~tlkagk/courses/ML_2016/Pokemon_creation/colormap.txt

• Following experiment: 1-layer LSTM, 512 cells
It is difficult to evaluate generation.

Real Pokémon

Never seen by machine!

Cover 50%

Cover 75%
Pokémon Creation
Drawing from scratch
Need some randomness
Generative Models

PixelRNN

Variational Autoencoder (VAE)

Generative Adversarial Network (GAN)

Auto-encoder

As close as possible

Randomly generate a vector as code

Image ?
Auto-encoder

input \rightarrow \text{NN Encoder} \rightarrow \text{code} \rightarrow \text{NN Decoder} \rightarrow \text{output}

\[ \text{Minimize reconstruction error} \]

From a normal distribution

\[ c_i = \exp(\sigma_i) \times e_i + m_i \]

\[ \sum_{i=1}^{3} (\exp(\sigma_i) - (1 + \sigma_i) + (m_i)^2) \]
Cifar-10

Pokémon Creation

Pick two dim, and fix the rest eight

Training
Writing Poetry

Ref: [http://www.wired.co.uk/article/google-artificial-intelligence-poetry](http://www.wired.co.uk/article/google-artificial-intelligence-poetry)
Why VAE?

Intuitive Reason

decode

code

encode

noise

noise
Why VAE?

**Intuitive Reason**

The variance of noise is automatically learned

What will happen if we only minimize reconstruction error?

\[
\sum_{i=1}^{3} \left( \exp(\sigma_i) - (1 + \sigma_i) + (m_i)^2 \right)
\]
Why VAE?

Intuitive Reason

We want $\sigma_i$ close to 0 (variance close to 1)

What will happen if we only minimize reconstruction error?

Original Code

Code with noisy

Minimize

$$\sum_{i=1}^{3} (\exp(\sigma_i) - (1 + \sigma_i) + (m_i)^2)$$

L2 regularization
Why VAE?

- Back to what we want to do

Estimate the probability distribution

Each Pokémon is a point $x$ in the space
Gaussian Mixture Model

\[ P(x) = \sum_m P(m)P(x|m) \]

How to sample?

\[ m \sim P(m) \text{ (multinomial)} \]

\[ x|m \sim N(\mu^m, \Sigma^m) \]

Each x you generate is from a mixture. Distributed representation is better than cluster.

m is an integer
VAE

\[ z \sim N(0, I) \]

\[ x | z \sim N(\mu(z), \sigma(z)) \]

\( z \) is a vector from normal distribution

Each dimension of \( z \) represents an attribute

Even though \( z \) is from \( N(0, I) \), \( P(x) \) can be very complex

\[ P(x) = \int\limits_{\mathbf{z}} P(z) P(x | z) dz \]

Infinite Gaussian
Maximizing Likelihood

\[ P(x) = \int_P(z)P(x|z)dz \]

\[ L = \sum_x \log P(x) \]

P(z) is normal distribution
\[ x|z \sim N(\mu(z), \sigma(z)) \]

\( \mu(z), \sigma(z) \) is going to be estimated

Maximizing the likelihood of the observed \( x \)

Tuning the parameters to maximize likelihood \( L \)

We need another distribution \( q(z|x) \)

\[ z|x \sim N(\mu'(x), \sigma'(x)) \]
Maximizing Likelihood

\[
P(x) = \int P(z)P(x|z)dz
\]

\[
L = \sum_x \log P(x)
\]

\[
\log P(x) = \int q(z|x) \log P(x) dz
\]

\[
= \int q(z|x) \log \left( \frac{P(z, x)}{P(z|x)} \right) dz
\]

\[
= \int q(z|x) \log \left( \frac{P(z, x)}{q(z|x)} \right) dz + \int q(z|x) \log \left( \frac{q(z|x)}{P(z|x)} \right) dz
\]

\[
\geq \int q(z|x) \log \left( \frac{P(x|z)P(z)}{q(z|x)} \right) dz
\]

\[
P(z) \text{ is normal distribution}
\]

\[
x|z \sim N(\mu(z), \sigma(z))
\]

\[
\mu(z), \sigma(z) \text{ is going to be estimated}
\]

\[
q(z|x) \text{ can be any distribution}
\]

\[
\text{Maximizing the likelihood of the observed } x
\]

\[
\text{KL}(q(z|x) || P(z|x)) \geq 0
\]

\[
\text{lower bound } L_b
\]
**Maximizing Likelihood**

\[
\log P(x) = L_b + KL(q(z|x) \| P(z|x))
\]

\[
L_b = \int_z q(z|x) \log \left( \frac{P(x|z)P(z)}{q(z|x)} \right) dz
\]

Find \( P(x|z) \) and \( q(z|x) \) maximizing \( L_b \)

\( q(z|x) \) will be an approximation of \( p(z|x) \) in the end
Maximizing Likelihood

\[ P(x) = \int P(z)P(x|z)dz \]

\[ L = \sum_x \log P(x) \]

\[ L_b = \int q(z|x)\log \left( \frac{P(z,x)}{q(z|x)} \right) dz = \int q(z|x)\log \left( \frac{P(x|z)P(z)}{q(z|x)} \right) dz \]

\[ = \int q(z|x)\log \left( \frac{P(z)}{q(z|x)} \right) dz + \int q(z|x)\log P(x|z)dz \]

\[ -KL(q(z|x)||P(z)) \]

P(z) is normal distribution
\[ x|z \sim N(\mu(z), \sigma(z)) \]

\[ \mu(z), \sigma(z) \text{ is going to be estimated} \]

\[ z|x \sim N(\mu'(x), \sigma'(x)) \]

Maximizing the likelihood of the observed x
Connection with Network

Minimizing \( KL(q(z|x) || P(z)) \)

\[
\begin{align*}
\sum_{i=1}^{3} (exp(\sigma_i) - (1 + \sigma_i) + (m_i)^2)
\end{align*}
\]

(Refer to the Appendix B of the original VAE paper)

Maximizing

\[
\int_{z} q(z|x) \log P(x|z) dz = E_{q(z|x)} [\log P(x|z)]
\]

This is the auto-encoder
Conditional VAE

To learn more ...

• Carl Doersch, Tutorial on Variational Autoencoders


• Cool demo:
  • http://vdumoulin.github.io/morphing_faces/
  • http://fvae.ail.tokyo/
Problems of VAE

• It does not really try to simulate real images

One pixel difference from the target

Realistic  Fake

One pixel difference from the target

VAE may just memorize the existing images, instead of generating new images
Generative Models

PixelRNN

Variational Autoencoder (VAE)

Generative Adversarial Network (GAN)

Yann LeCun’s comment

What are some recent and potentially upcoming breakthroughs in unsupervised learning?

Yann LeCun, Director of AI Research at Facebook and Professor at NYU
Written Jul 29 · Upvoted by Joaquin Quiñonero Candela, Director Applied Machine Learning at Facebook and Huang Xiao

Adversarial training is the coolest thing since sliced bread.

I’ve listed a bunch of relevant papers in a previous answer.

Expect more impressive results with this technique in the coming years.

What’s missing at the moment is a good understanding of it so we can make it work reliably. It’s very finicky. Sort of like ConvNet were in the 1990s, when I had the reputation of being the only person who could make them work (which wasn’t true).

Yann LeCun’s comment

What are some recent and potentially upcoming breakthroughs in deep learning?

Yann LeCun, Director of AI Research at Facebook and Professor at NYU
Written Jul 29 · Upvoted by Joaquín Quiñonero Candela, Director Applied Machine Learning at Facebook and Nikhil Garg, I lead a team of Quora engineers working on ML/NLP problems

The most important one, in my opinion, is adversarial training (also called GAN for Generative Adversarial Networks). This is an idea that was originally proposed by Ian Goodfellow when he was a student with Yoshua Bengio at the University of Montreal (he since moved to Google Brain and recently to OpenAI).

This, and the variations that are now being proposed is the most interesting idea in the last 10 years in ML, in my opinion.

擬態的演化

蝴蝶不是棕色
蝴蝶沒有葉脈
……..

http://peellden.pixnet.net/blog/post/40406899-2013-
The evolution of generation

NN Generator v1 -> Discriminator v1

NN Generator v2 -> Discriminator v2

NN Generator v3 -> Discriminator v3

Real images: 5 0 4 1
GAN - Discriminator

Vectors from a distribution → NN Generator v1 → Decoder in VAE

Real images:

image → Discriminator v1 → 1/0 (real or fake)
GAN - Generator

“Tuning” the parameters of generator

The output be classified as “real” (as close to 1 as possible)

Generator + Discriminator = a network

Using gradient descent to find the parameters of generator

Fix the discriminator
GAN – Toy Example

Real data (black points) → Green distribution → Discriminator → 1/0

NN Generator

Demo: http://cs.stanford.edu/people/karpathy/gan/
Cifar-10

• Which one is machine-generated?

Ref: https://openai.com/blog/generative-models/
Moving on the code space

畫漫畫

・Ref: https://github.com/mattya/chainer-DCGAN
画漫画

Web demo: http://mattya.github.io/chainer-DCGAN/

・Ref: http://qiita.com/mattya/items/e5bfe5e04b9d2f0bbd47

長髪化ベクトル

一番左のキャラクターが元画像で、右に行くほど長髪化ベクトルを強く足している
In practical ......

• GANs are difficult to optimize.
• No explicit signal about how good the generator is
  • In standard NNs, we monitor loss
  • In GANs, we have to keep “well-matched in a contest”
• When discriminator fails, it does not guarantee that generator generates realistic images
  • Just because discriminator is stupid
  • Sometimes generator find a specific example that can fail the discriminator
• Making discriminator more robust may be helpful.
To learn more ...

• “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”
• “Improved Techniques for Training GANs”
• “Autoencoding beyond pixels using a learned similarity metric”
• “Deep Generative Image Models using a Laplacian Pyramid of Adversarial Network”
• “Super Resolution using GANs”
• “Generative Adversarial Text to Image Synthesis”
To learn more ...

• Basic tutorial:
  • http://blog.aylien.com/introduction-generative-adversarial-networks-code-tensorflow/
  • https://bamos.github.io/2016/08/09/deep-completion/
  • http://blog.evjang.com/2016/06/generative-adversarial-nets-in.html
Acknowledgement

• 感謝 Ryan Sun 來信指出投影片上的錯字