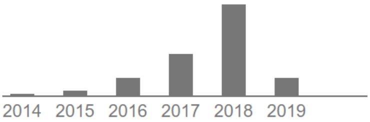
Auto-Encoding Variational Bayes

Milan Ilic

3rd April 2019

Paper

Diederik P Kingma, Max Welling. Universiteit van Amsterdam.
Auto-Encoding Variational Bayes. December, 2013



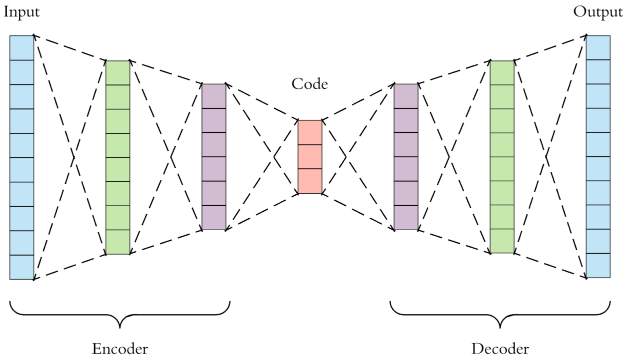
Number of citations: 4364

Table of contents

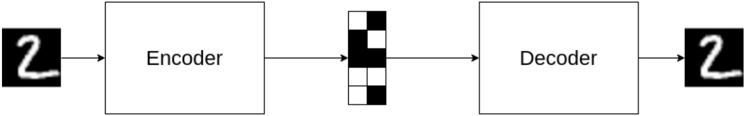
Autoencoders

Autoencoders

Autoencoder Architecture



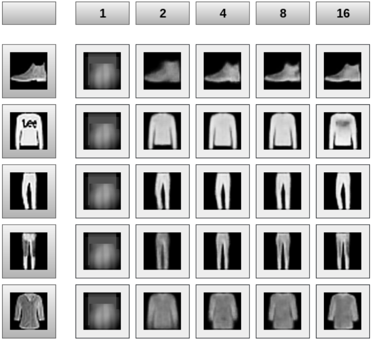
Conventional Autoencoder



- Encoder: $p_{encoder}(\boldsymbol{h} \mid \boldsymbol{x})$
- Decoder: $p_{decoder}(\boldsymbol{x} \mid \boldsymbol{h})$

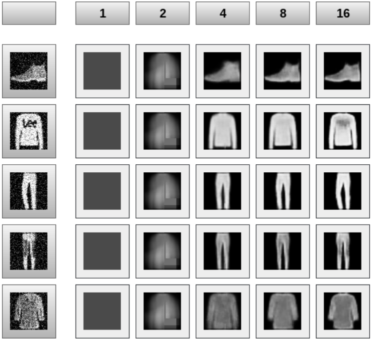
Where $\boldsymbol{h}$ is the *code* and $\boldsymbol{x}$ is the input.

Conventional Autoencoder



- Dimensionality reduction
- Outlier detection

Denoising Autoencoder



- Reducing noise in an image
- Removing some object from an image (e.g. watermark)

Generative models

Generative models

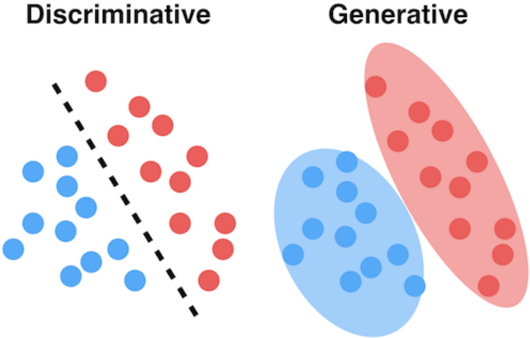
Discriminative & Generative models

Let (x, y) be the inputs and the corresponding labels, in that order

- Discriminative classifiers model the posterior $p(y \mid x)$ directly, or learn a direct map from inputs x to the class labels
- Generative classifiers learn a model of joint probability $p(x, y)$ and make their predictions by using the Bayes rule to calculate $p(y \mid x)$, and then picking the most likely label y

$$p(y \mid x) = \frac{p(x \mid y)p(y)}{p(x)}$$

Discriminative & Generative models



Discriminative models

- Logistic regression
- Linear regression
- Support vector machines
- Random forests
- Traditional neural networks
- etc...

Generative models

- If we are not interested in a supervised problem, we can use generative model to only learn the distribution of our data $p(x)$
- After training we can generate new data similar to x
- *"What I cannot create, I do not understand"*

— Richard Feynman

Generative models

- Boltzmann machine
- PixelRNN
- GAN
- Variational autoencoder

One problem with generative models is that they need very large datasets to work properly.

Variational Autoencoder

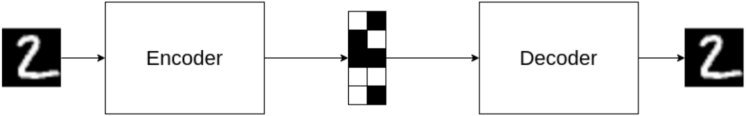
Variational Autoencoder

VAE

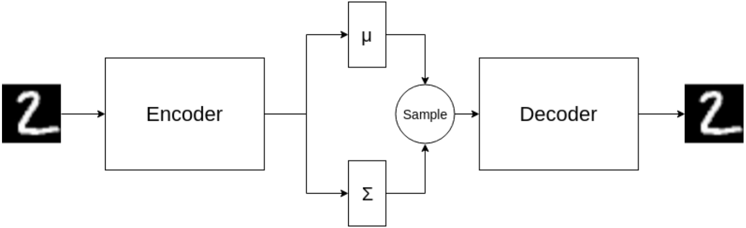
"The variational autoencoder approach is elegant, theoretically pleasing, and simple to implement"

— Ian Goodfellow

Reminder



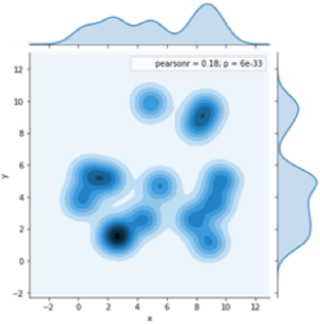
VAE Architecture



What can we now do?

- After training the VAE, we could simply discard the encoder part and use the decoder to generate new data
- To get the decoders input, we just sample from the Gaussian distribution and pass that sample
- Generated data instances should come from points with high probability in datasets distribution space

Vizualization of latent space



On the left the Conventional AE latent space and on the right VAEs latent space

Variational Inference

Variational Inference

Set up

- Assume that $x = x_{1:n}$ are the observations, $z = z_{1:m}$ are hidden variables and α are fixed parameters
- First we generate value z from a prior distribution $p(z)$ and then generate x form conditional distribution $p(x \mid z)$, we can assume what form these two distributions take.
- We want to calculate the *posterior distribution*:

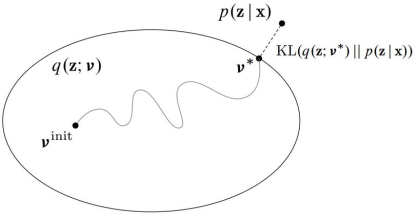
$$p_\alpha(z \mid x) = \frac{p_\alpha(x \mid z) p_\alpha(z)}{\int_z p_\alpha(z, x) dz}$$

- In most cases the posterior is intractable
- Variational inference treats the inference problem as an approximation problem

Approaches

- Deterministic approximation
  - Mean field variational inference
  - Stochastic variational inference
- Stochastic approximation (Markov Chain Monte Carlo)
  - Metropolis–Hastings algorithm
  - Gibbs sampling
- By doing the deterministic approximation we will converge but not find the optimal solution
- The main problem with the stochastic approximation is that it is very slow due to the sampling step

Main Idea



- The main idea behind variational methods is to, first pick a tractable family of distributions over the latent variables with its own variational parameters $q(z_{1:m} \mid \nu)$
- Then to find parameters that make it as close as possible to the true posterior
- Use that q instead of the posterior to make predictions about future data

Kullback-Leibler Divergence

$$D_{KL}(p\|q) = \mathop{\mathbb{E}}_{x \sim p}\left[\log \frac{p(x)}{q(x)}\right]$$

Discrete and continuous form:

- $D_{KL}(p\|q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$
- $D_{KL}(p\|q) = \int_x p(x) \log \frac{p(x)}{q(x)} \, dx$

Used to measure similarity between two probability distributions (*w.r.t. one of them*).

Properties:

- $D_{KL}(p\|q) \geq 0, \forall p, q$
- $D_{KL}(p\|q) = 0 \iff p = q$
- $D_{KL}(p\|q) \neq D_{KL}(q\|p)$ in general

The Variational Lower Bound

$$
\begin{aligned}
D_{KL}(q(z \mid x) \| p(z \mid x)) &= \int_z q(z \mid x) \log \frac{q(z \mid x)}{p(z \mid x)} \\
&= -\int_z q(z \mid x) \log \frac{p(z \mid x)}{q(z \mid x)} \\
&= -\left[\int_z q(z \mid x) \log \frac{p(x, z)}{q(z \mid x)} - \int_z q(z \mid x) \log p(x)\right] \\
&= -\int_z q(z \mid x) \log \frac{p(x, z)}{q(z \mid x)} + \log p(x) \int_z q(z \mid x) \\
&= -\mathcal{L} + \log p(x)
\end{aligned} \tag{1}
$$

$$\boldsymbol{\log p(x) = \mathcal{L} + D_{KL}(q(z \mid x) \| p(z \mid x))}$$

Minimizing the KL divergence is equal to maximizing **the variational lower bound**!

The Variational Lower Bound

$$
\begin{aligned}
\mathcal{L} &= \int_z q(z \mid x) \log \frac{p(x, z)}{q(z \mid x)} \\
&= \int_z q(z \mid x) \log \frac{p(x \mid z) p(z)}{q(z \mid x)} \\
&= \int_z q(z \mid x) \log p(x \mid z) + \int_z q(z \mid x) \log \frac{p(z)}{q(z \mid x)}
\end{aligned} \tag{2}
$$

$$
\boldsymbol{\mathcal{L} = \mathbb{E}_{q(z \mid x)} \log p(x \mid z) - D_{KL}(q(z \mid x) \| p(z))}
$$

- The first term is conceptually the negative reconstruction error and the second makes our $q(z \mid x)$ close to the prior $p(z)$

Back to VAE

- Let θ be the generative parameters and ϕ the variational parameters and assume that $x^{(1)}, ..., x^{(N)}$ are i.i.d:

$$\log p_\theta(x^{(1)}, ..., x^{(N)}) = \sum_{i=1:N} \log p_\theta(x^{(i)})$$

- Each term on the right hand side can be written as:

$$\log p_\theta(x^{(i)}) = D_{KL}(q_\phi(z \mid x^{(i)}) \| p_\theta(x^{(i)} \mid z)) + \mathcal{L}(\theta, \phi; x^{(i)})$$

- The variational lower bound is now:

$$\mathcal{L}(\theta, \phi; x^{(i)}) = \mathbb{E}_{q_\phi(z|x^{(i)})} \log p_\theta(x^{(i)} \mid z) - D_{KL}(q_\phi(x^{(i)} \mid z) \| p_\theta(z))$$

Reparameterization Trick

Reparameterization Trick

Definition

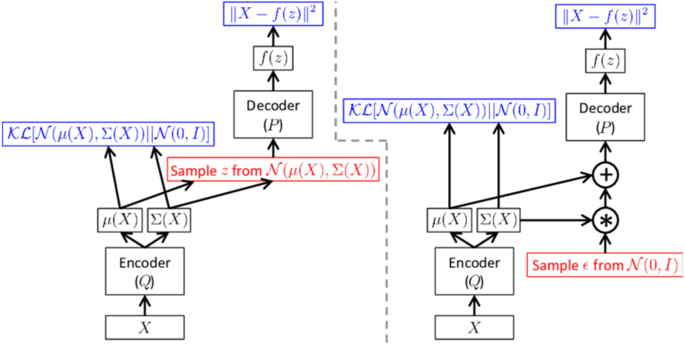
- In the middle of VAEs architecture there should be sampling from $z \sim q_\phi(z \mid x) \sim \mathcal{N}(\mu, \Sigma)$
- We cannot do such thing because backpropagation can't go through a sampling node
- It is often possible to express the random variable z as a deterministic variable $z = g_\phi(\epsilon, x)$, where ϵ is an auxiliary variable with independent marginal $p(\epsilon)$, and $g_\phi(.)$ is some vector-valued function parameterized by ϕ
- In our (Gaussian) case $\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\Sigma} * \boldsymbol{\epsilon}$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

Other distributions

Normal distribution isn't the only one on which we can do this transformation, there are three groups of distributions:

- Tractable inverse CDF:
  - Let $\epsilon \sim \mathcal{U}(\mathbf{0}, \boldsymbol{I})$ and $g_\phi(\epsilon, x)$ be the inverse CDF of $q_\phi(z \mid x)$.
  - Exponential, Cauchy, Logistic...
- Location-scale family:
  - As in the example from the last slide, $z = location + scale * \epsilon$, where ϵ is from the standard distribution.
  - Laplace, Student's t, Uniform, Normal...
- Composition:
  - It is often possible to express random variables as different transformations of auxiliary variables.
  - Log-Normal, Gamma, Beta, Chi-Squared, F...

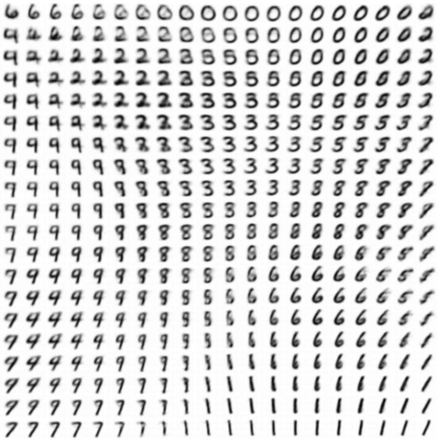
Putting everything together



Results

Results

Healing Imagery

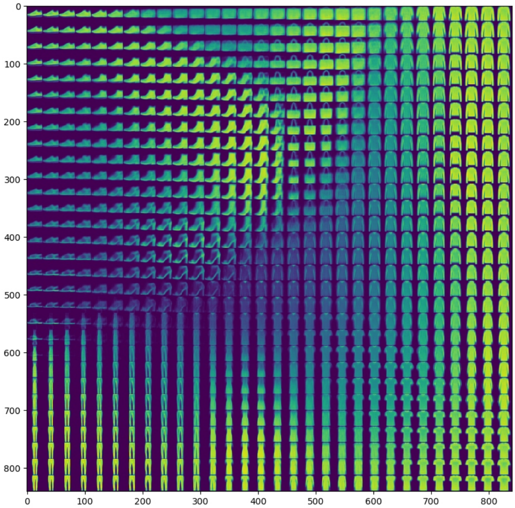


(b) Learned MNIST manifold

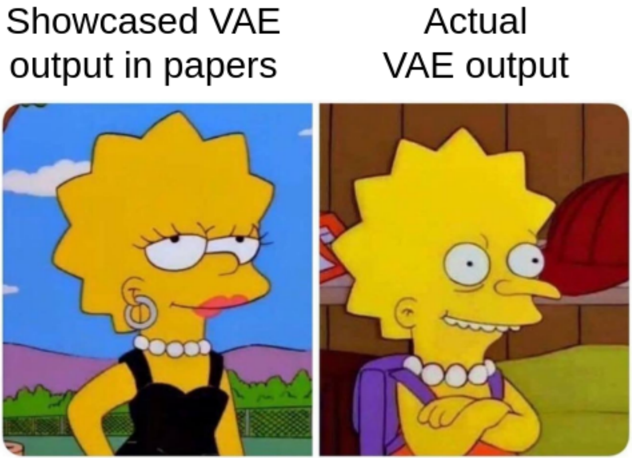
Healing Imagery

(a) Learned Frey Face manifold

Healing Imagery



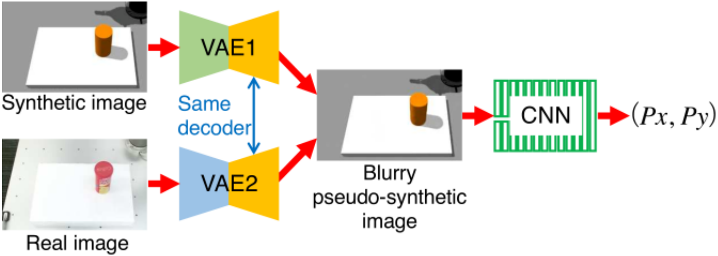
Healing Imagery



Applications

- Data generation (e.g. images, music...)
- Caption generation
- Anomaly detection
- Image segmentation
- Super resolution
- etc...

Applications



Conditional VAE

Conditional VAE

Motivation

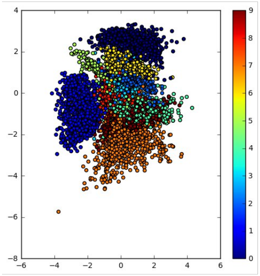
- By using the variational autoencoder, we do not have control over the data generation process.
- E.g. we cannot generate only one specific digit from a model trained on MNIST dataset.
- We want, for example, to input the character 9 to our model and get a generated image of a handwritten digit 9.

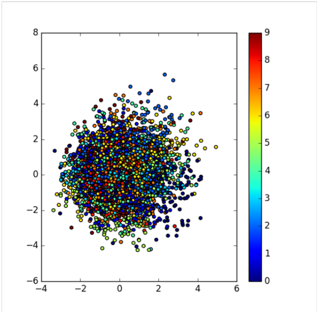
Approach

- We will condition encoder and decoder on other inputs as well as the image, lets call those inputs c.
- Encoder becomes: $q(z \mid x, c)$
- Decoder becomes: $p(x \mid z, c)$
- Now our variational lower bound objective becomes:

$$\mathcal{L} = \mathbb{E}[\log p(x \mid z, c)] - D_{KL}(q(z \mid x, c) \| p(z \mid c))$$

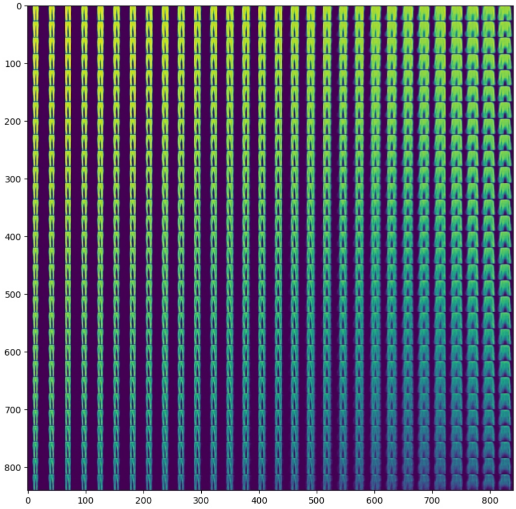
Latent space





On the left the Vanilla VAE latent space and on the right CVAEs latent space.

Results


0
100
200
300
400
500
600
700
800
0
100
200
300
400
500
600
700
800

References

[KW14, GDG+15, GBC16, NJ01, SLY15]

- Ian Goodfellow, Yoshua Bengio, and Aaron Courville, **Deep learning**, MIT Press, 2016, `http://www.deeplearningbook.org`.
- Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra, **Draw: A recurrent neural network for image generation**, cite arxiv:1502.04623.
- Diederik P. Kingma and Max Welling, **Auto-encoding variational bayes**.
- Andrew Y. Ng and Michael I. Jordan, **On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes**, 841–848.
- Kihyuk Sohn, Honglak Lee, and Xinchen Yan, **Learning structured output representation using deep conditional generative models**, 3483–3491.

Thanks

Thanks for listening!