Deep Learning

AutoEncoders

Edgar F. Roman-Rangel.
edgar.roman@itam.mx

Digital Systems Department.
Instituto Tecnológico Autónomo de México, ITAM.

March 26th, 2021.

Outline

Supervised learning

- Pairs $\{x^{(n)}, y^{(n)}\}_{n=1}^{N}$ of input and output data.
- Goal: learn a model $\hat{y}^{(n)} = f(x^{(n)}; \Omega)$.
- Such that $\mathcal{L}(y^{(n)}, \hat{y}^{(n)}) \approx 0$.
- ANN's are, by design, models for supervised learning.

Unsupervised learning

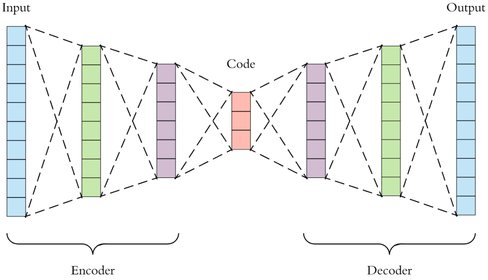
- No labels $y^{(n)}$ for training, i.e., only $\{x^{(n)}\}_{n=1}^{N}$.
- We do not learn a mapping function.
- Rather, we try to make sense of $\{x^{(n)}\}$.
- Examples: clustering, density estimation, anomaly detection, dimensionality reduction.

Self-supervised learning

Strictly speaking, unsupervised learning is not possible with ANN's.

We take a roundabout with self-supervised learning.

- Learn models to approximate a task on the input data.
- Input and output are the same data, i.e., pairs $(x^{(n)}, \hat{x}^{(n)})$.
- Examples: clean noisy data, dimensionality reduction.
- Enter: *AutoEncoders*.



Discriminative vs Generative

Discriminative models

Conditional probability:

$$p(y|x;\Omega).$$

Generative models

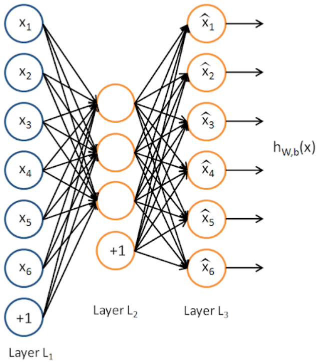
Join probability distribution:

$$p(x, y; \Omega).$$

Outline

Autoencoder

LeCun, 1987. “Modèles connexionistes de l’apprentissage”, Ph.D. Thesis.



Vanilla AE.

ANN that learns to reproduce its own input, while learning interesting data representations.

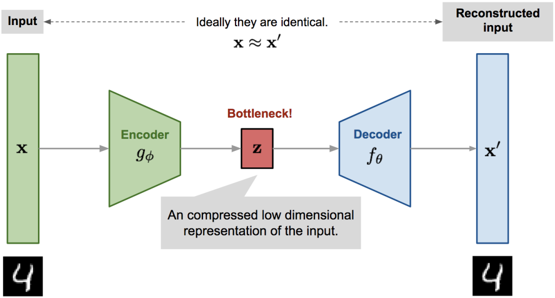
E.g., sparse representations,

$$\mathcal{L}(y, \hat{y}; \Omega) = \|y - \hat{y}\|^2 + \alpha\|\Omega\|_1.$$

Intermediate output is known as *latent representation*, and denotes by z.

Structure

- Encoder.
- Latent representation.
- Decoder.
- Regularizers (optional).



Representation Learning

Representation Learning is a subset of machine learning, which of high relevance in deep learning.

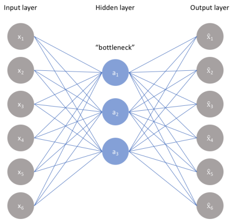
It deals with the study (analysis, design, interpretability, etc.) of the intermediate representations of a deep neural network.

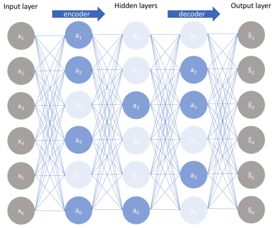
Understand how knowledge is learned and represented.

Understanding and designing *latent representations* can be done by principles of representations learning.

Undercomplete AE

- AE with hidden representation that are shorter than input.
- Produces undercomplete representations.
- Output after the linear transformation equivalent to PCA.

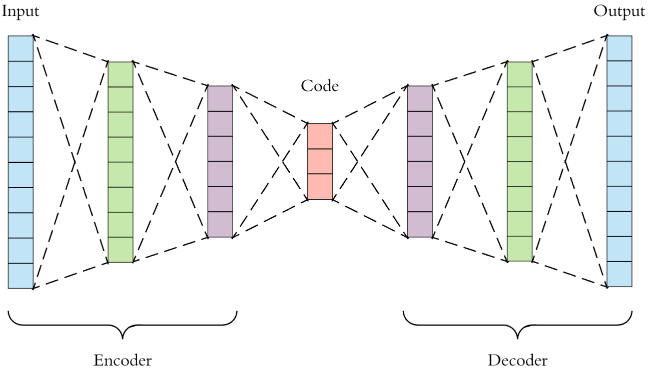




- Overcomplete AE: larger latent representation.
- Sparse AE: sparsity by constrains, e.g., L1 regularization.

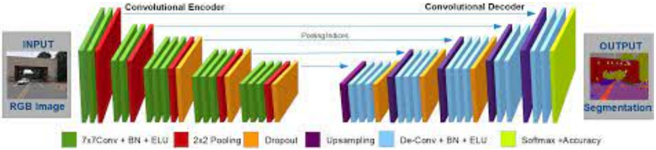
Deep AE

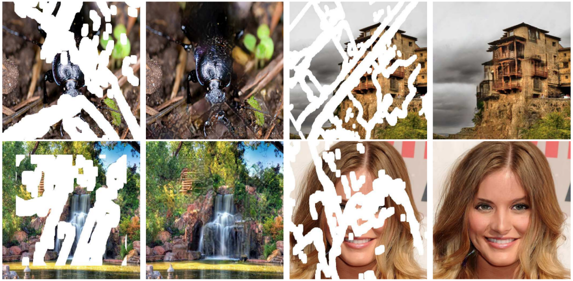
Deeper models enable more non-linear transformations, which might lead to richer latent representations.



ConvAE

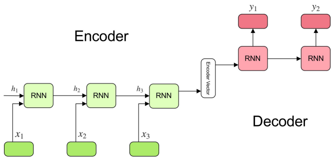
Combine the potential of CNN's with AE architectures.

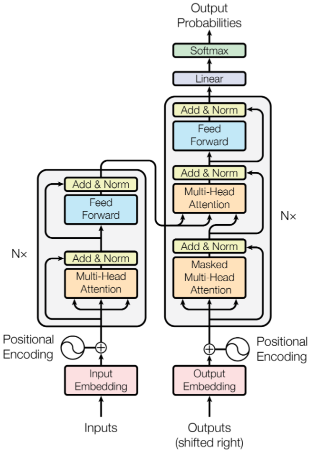




LSTM and transformer

Similarly, we can combine AE with language models.

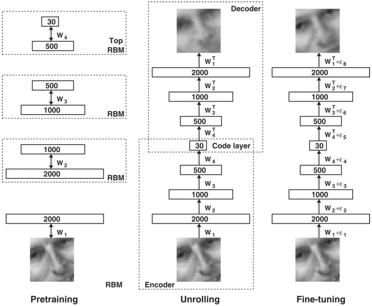




Pretraining with AE's

Often, there is a lot of unlabeled data out there.

1. Pretrain your model with an AE.
2. Transfer weights to a decision model (classifier).
3. Fine tune the transferred weights for specific task.



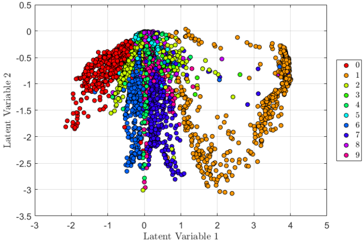
Hinton & Salakhutdinov, 2006. "Reducing the Dimensionality of Data with Neural Networks".

Outline

AE's limitations

Latent space of AE is learned to comply with the task of interest, but with no other particular restriction, specially not on its shape.

- No special structure.
- Latent space might be discontinue.
- Sampling from it produces samples from the unknown space.
- Decoding unknown samples might lead to noise.



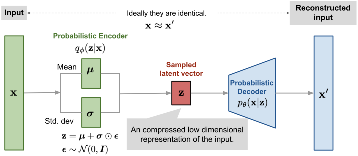
Variational AE (VAE)

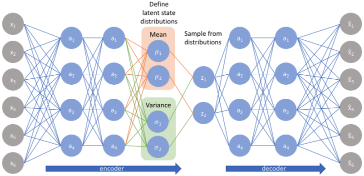
Kingma & Welling, 2013. “Auto-Encoding Variational Bayes”.

Variational inference

Approximate an observable probability distribution $p(z|x)$, with another parametric distribution $q(z)$.

- $q(z)$ is chosen to be a Gaussian.
- VAE’s latent space is, by design, continuous.
- Latent representation: means (μ) and standard deviations (σ).



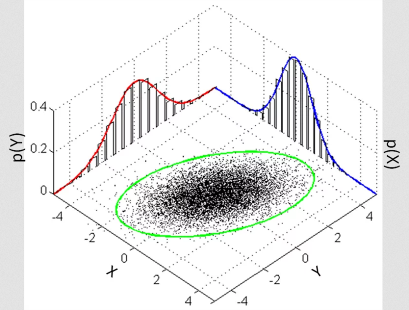


Variational inference

Goal: find a distribution $q(z|x)$ of some latent variable z, from which we could sample $z \sim q(z|x)$, and then we can generate new synthetic samples $\hat{x} \sim p(x|z)$.

- Hard to find the marginal $p(x)$, as we need to integrate its conditional over the whole latent space.

$$p(x) = \int_\Omega p(x,z)dz,$$
$$= \int_\Omega p(x|z)p(z)dz.$$

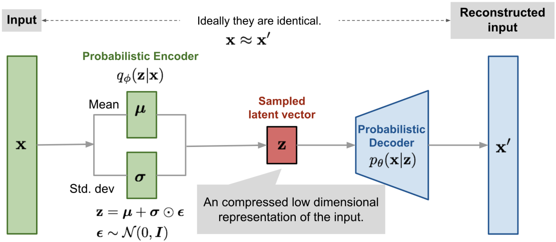


VAE loss function

- Instead, minimize the difference between the observable distribution and the ideal Gaussian.
- Use Kullback-Leibler divergence.

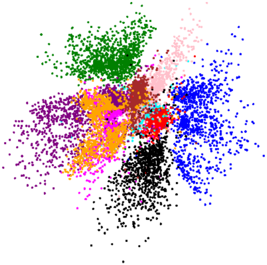
$$\mathcal{L}^T = \mathcal{L}^R(x, \hat{x}) + \gamma D_{KL}(q(z)||p(z)),$$

where, $p(z) \sim \mathcal{N}(\mu, \sigma)$, and γ is a weighting factor.

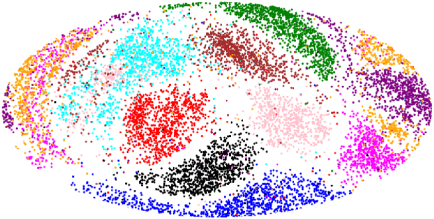


Variational latent space

Example no the MINST dataset.

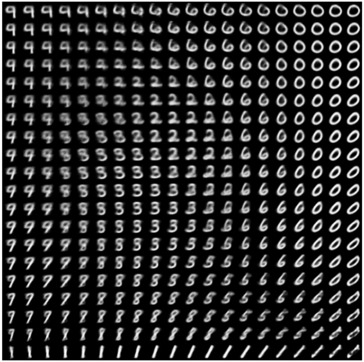


(a) $\mathbb{R}^2$ latent space of the $\mathcal{N}$-VAE.



(b) Hammer projection of $\mathcal{S}^2$ latent space of the $\mathcal{S}$-VAE.

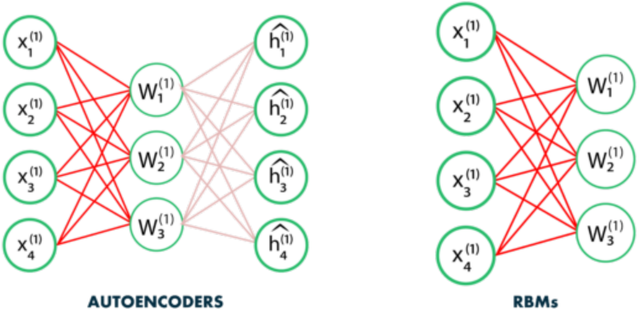
VAE examples



Outline

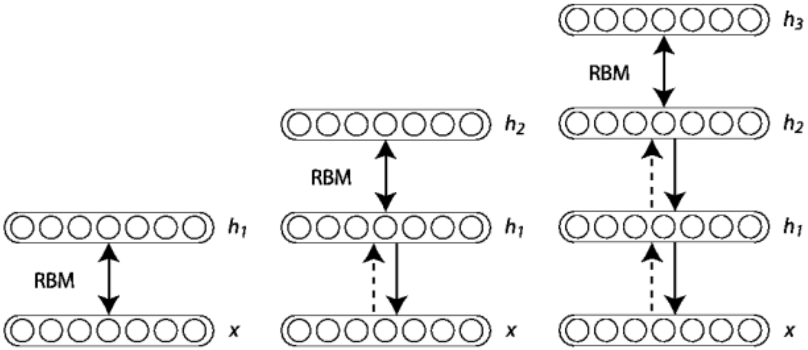
Restricted Boltzman Machines (RBM)

- Two-layers NN: Input and hidden representation.
- No output layer.
- Backprop against feed-backward loss.
- Input reconstruction from hidden representation.



Deep Belief Networks (DBN)

Cascaded array of RBM's.



Q&A

Thank you!

edgar.roman@itam.mx