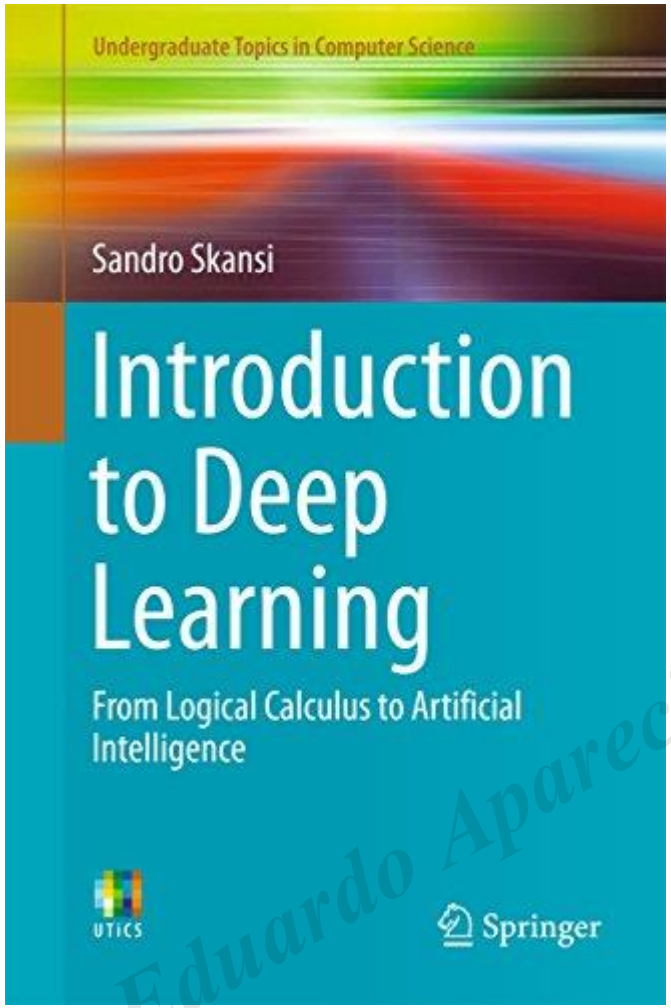
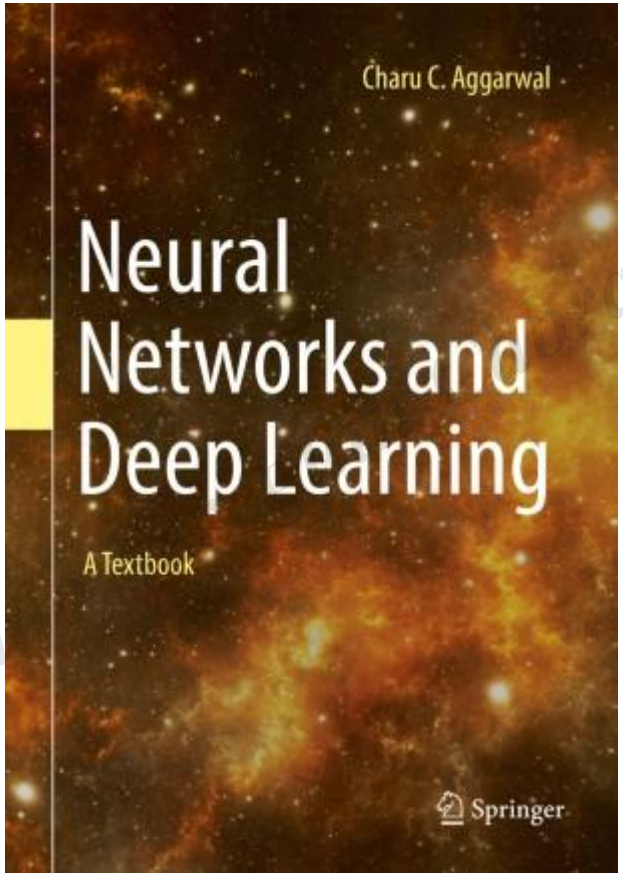
MBA USP ESALQ

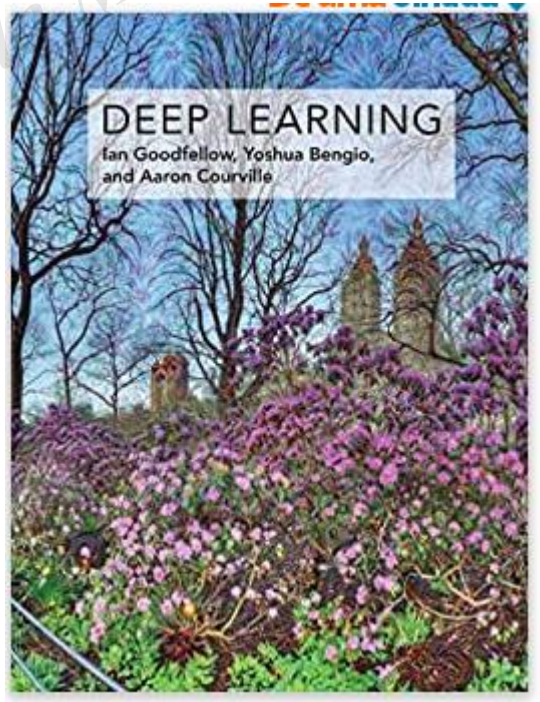
Deep Learning

Prof. Dr. Jeronymo Marcondes

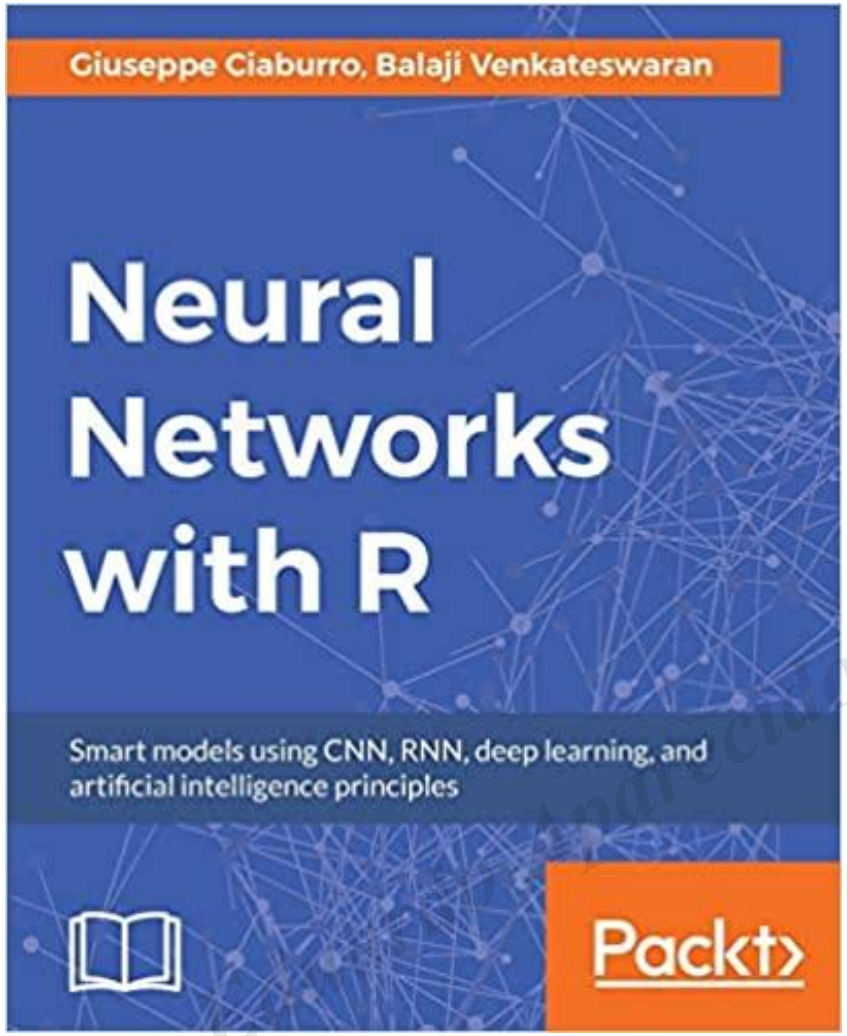
Introduction

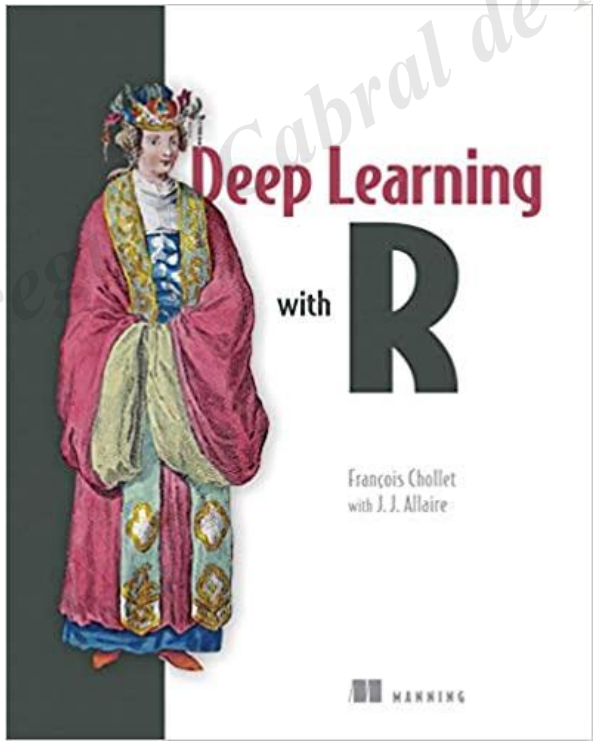






Introduction





Introduction

- Unsupervised Deep Learning
- What is not supervised?
- What is the difference regarding the FNN models?

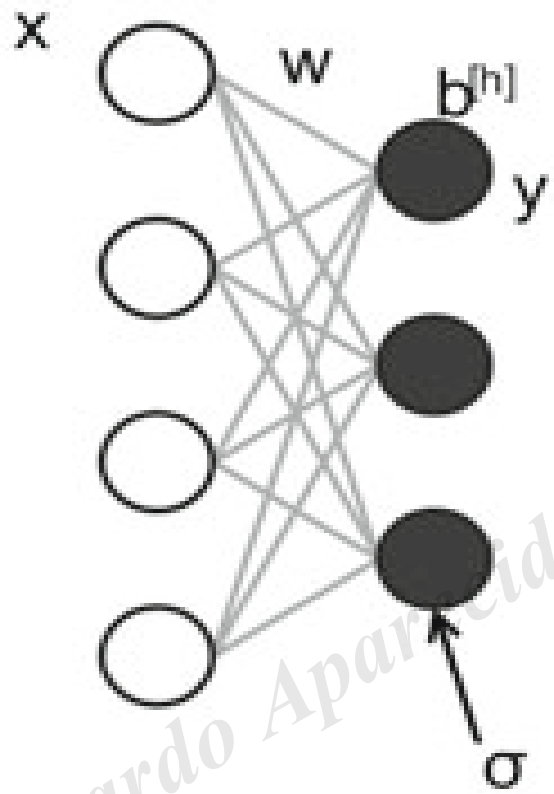
Introduction

- Models based on Energy
- Objective: Reduce energy
- Similar to the problem with our cost function

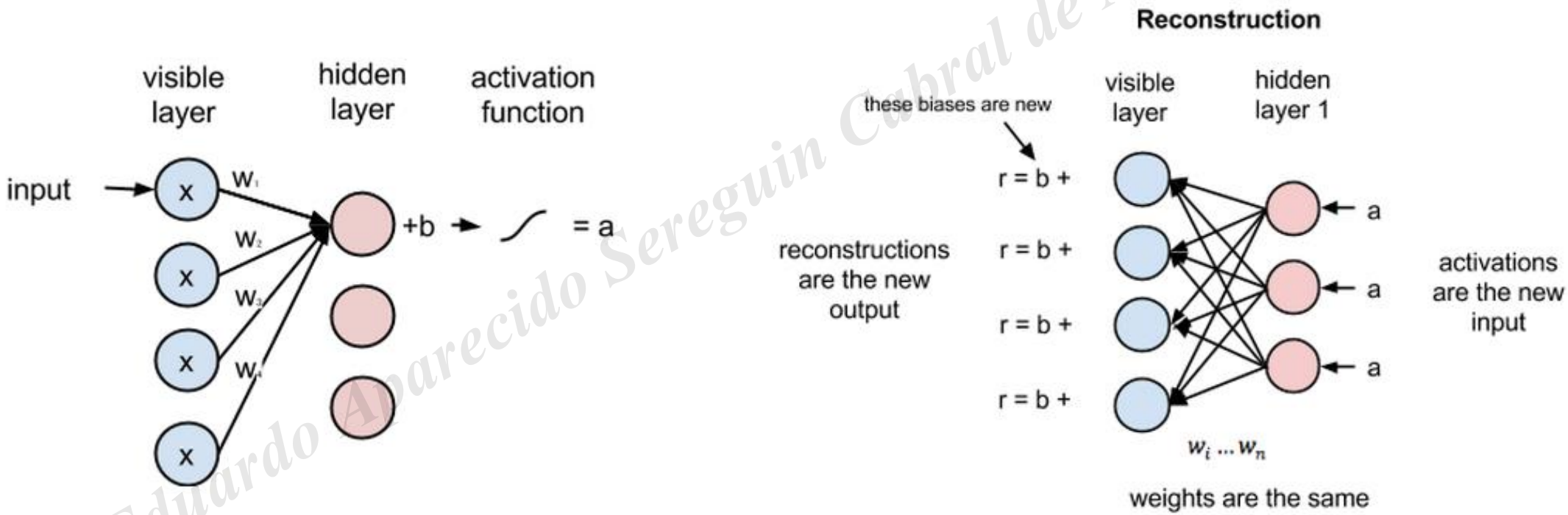
Boltzmann Machines

- The Boltzmann machine is a FNN of a layer
- Our objective: learn the distribution of input probabilities
- Adjust weights to be possible to rebuild inputs
- Restricted Boltzmann Machine

Boltzmann Machines



Boltzmann Machines



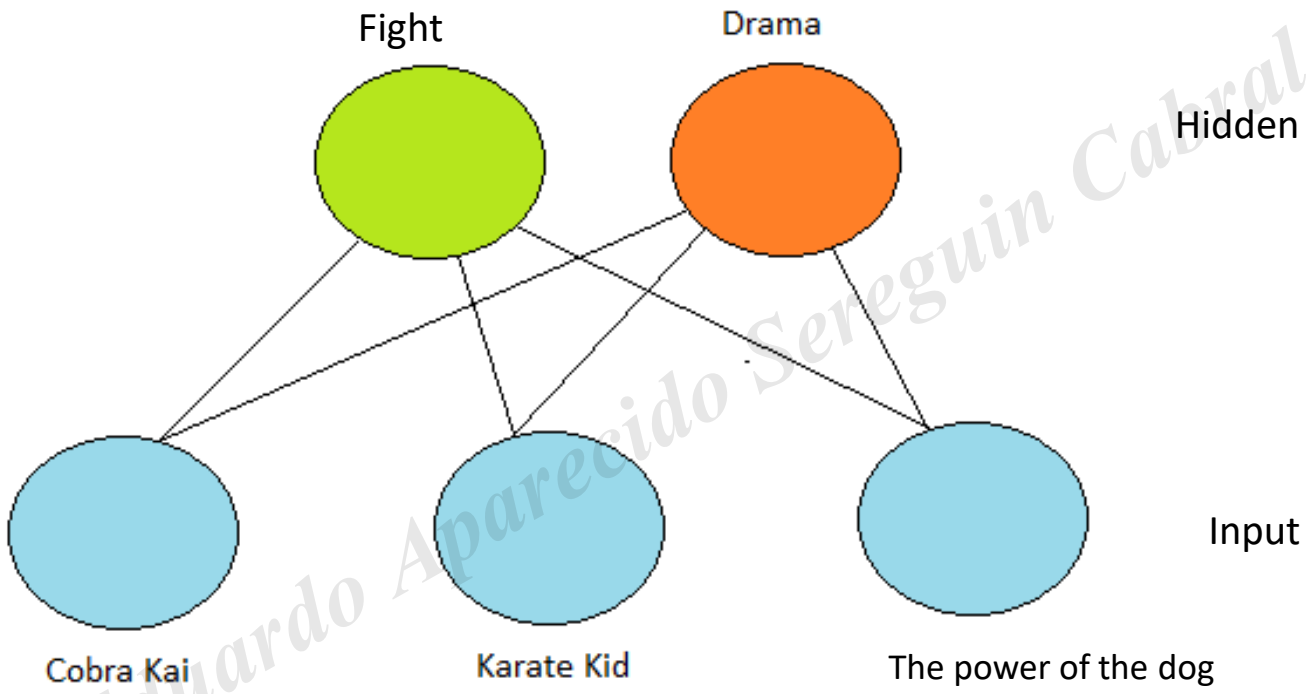
Source: https://wiki.pathmind.com/restricted-boltzmann-machine

Boltzmann Machines

Characteristics:

- It does not have an output layer
- Inputs are transferred to the hidden layer
- Why is “restricted” machine?

Boltzmann Machines



Boltzmann Machines

Functioning:

- It initiates internal layer randomly
- It receives x supply.
- It calculates (conditional probability that the neuron is activated):

$$y = \sigma(xw + b_h)$$

Boltzmann Machines

Functioning:

- Y is returned to the internal layer for rebuilding.
- Calculate:

$$r = \sigma(yw + b_v)$$

- Intuition: latent variable

Boltzmann Machines

Learning by Contrastive Divergence

Contrastive divergence is an alternative training technique to approximate the graphic inclination that represents the relationship between weights of a network and its error, called gradient. As the most probabilistic learning algorithms try to optimize the value of the logarithmic probability, this gradient represents the desired direction of change, of learning, for the network parameters.

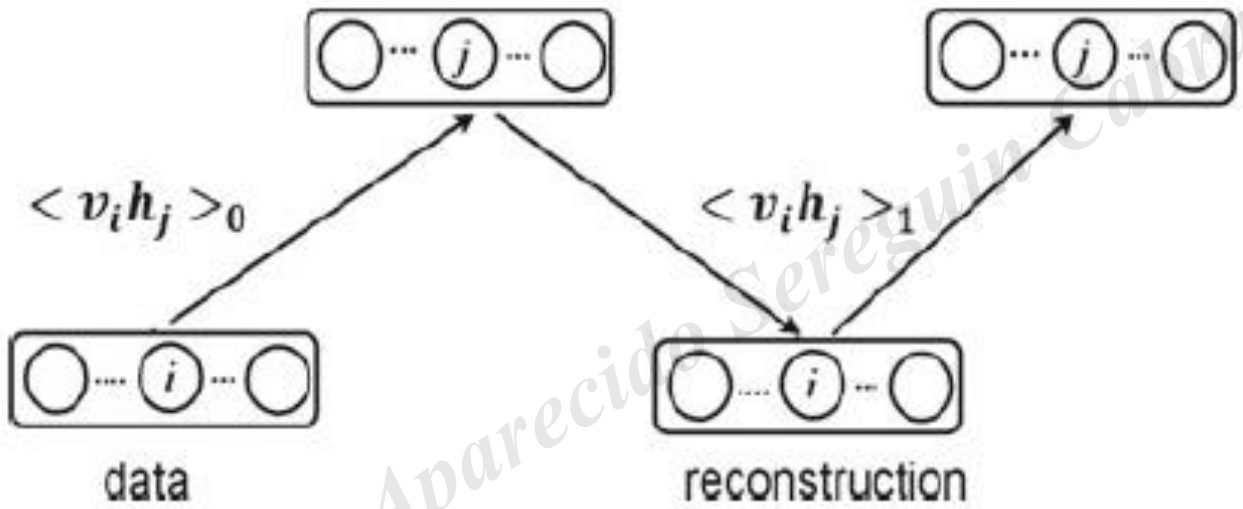
Boltzmann Machines

Functioning:

- The difference between input and r is verified by the Kullback-Leibler divergence.

- Negative and positive phase

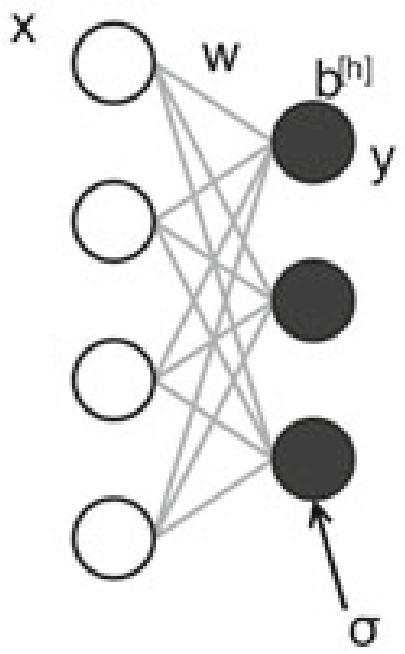
Boltzmann Machines



Boltzmann Machines

Applications:

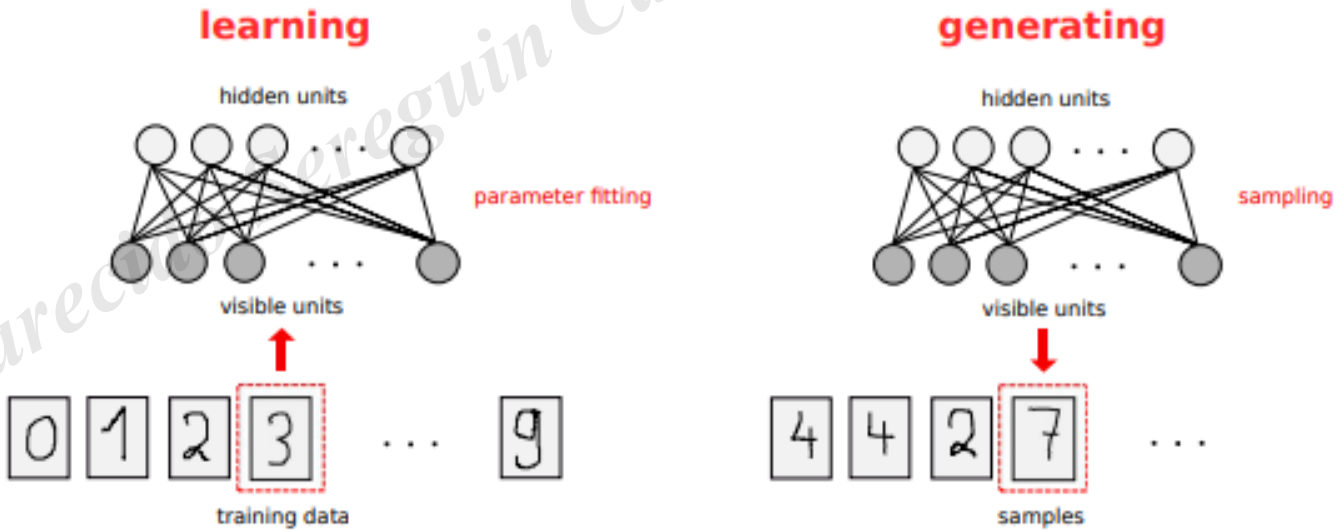
- Dimensionality Reduction
- Example: problem with many variables.



Boltzmann Machines

Applications:

- Recommender systems
- Image Reconstruction



Boltzmann Machines

Example: NETFLIX Case

- Recommendation based on content – choice similar to what you already do

- Recommendation based on collaborative filtering – similar profiles

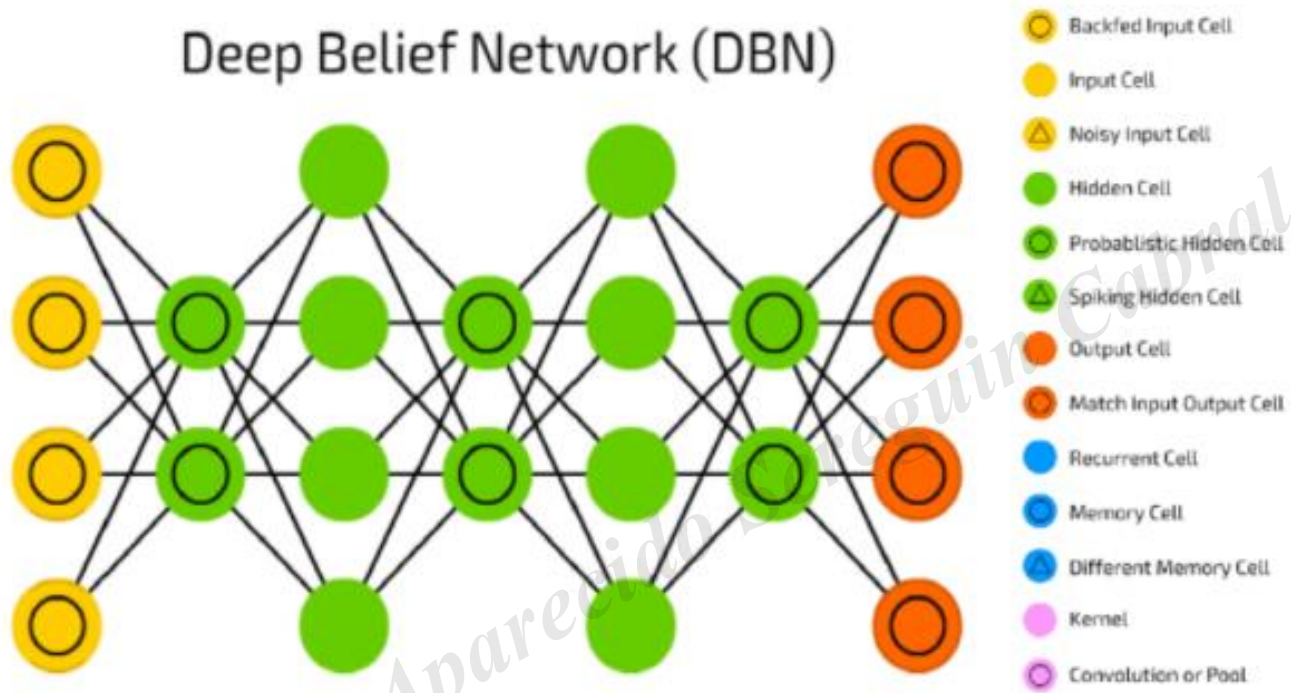
https://tecnetit.com.br/como-a-netflix-usou-a-data-science-para-melhorar-seu-sistema-de-recomendacao/

DBN

Deep Belief Networks

- The most generalized case of the Boltzmann Restricted Machine
- Stack Bolztamnn restricted machines

DBN

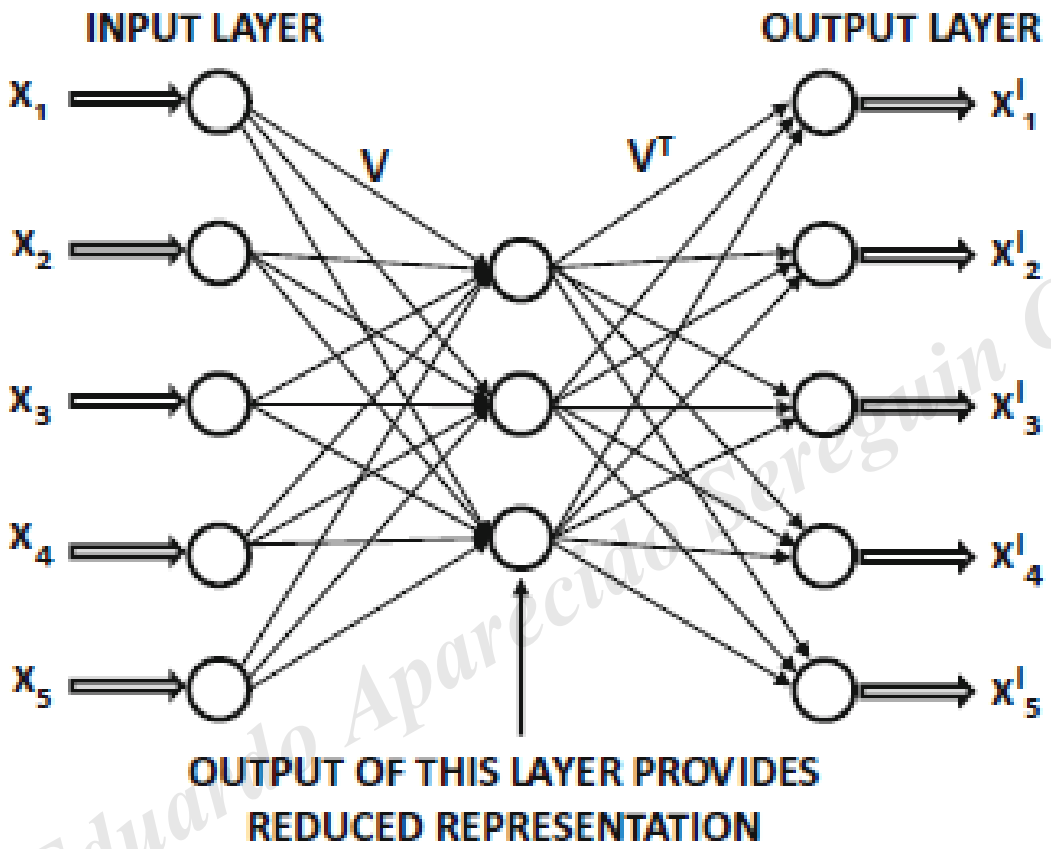


Source: https://www.deeplearningbook.com.br/

Autoencoder

- Objectives very similar to the Boltzmann Machine
- Input Reconstruction
- Dimensionality Reduction

Autoencoder



Autoencoder

- Autoencoder is like a funnel
- The information goes through and it is restricted to a smaller number of layers
- After this, we try to rebuild information

Autoencoder

- "Encode" step - codes the image in a smaller dimensionality:

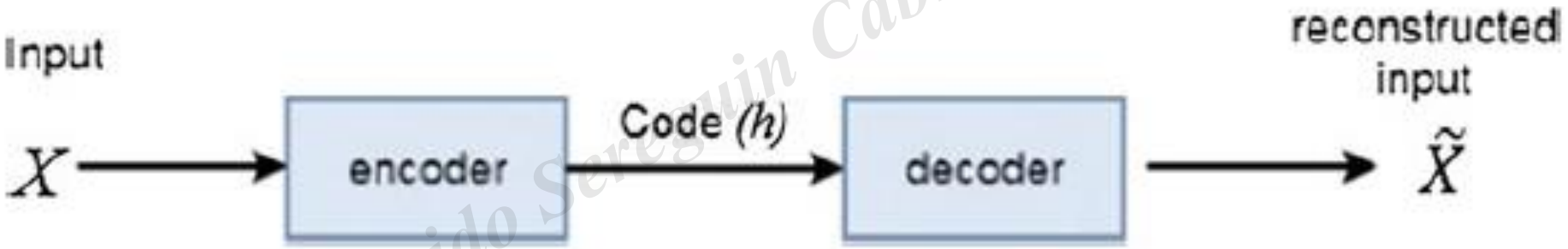
$$h = f(x)$$

- "Decode" step – decodes the image to rebuild the input.

$$\mathrm{r} = f(h)$$

- Difference regarding the boltzmann restricted machine

Autoencoder

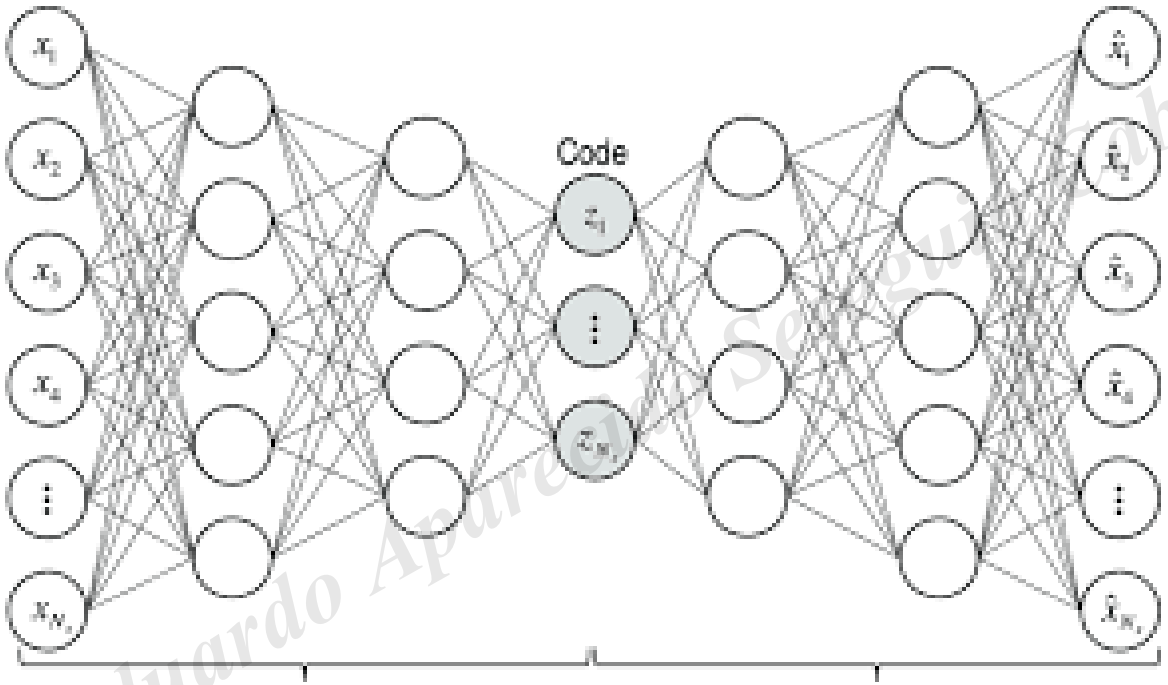


Autoencoder

https://douglasduhaime.com/posts/visualizing-latent-spaces.html

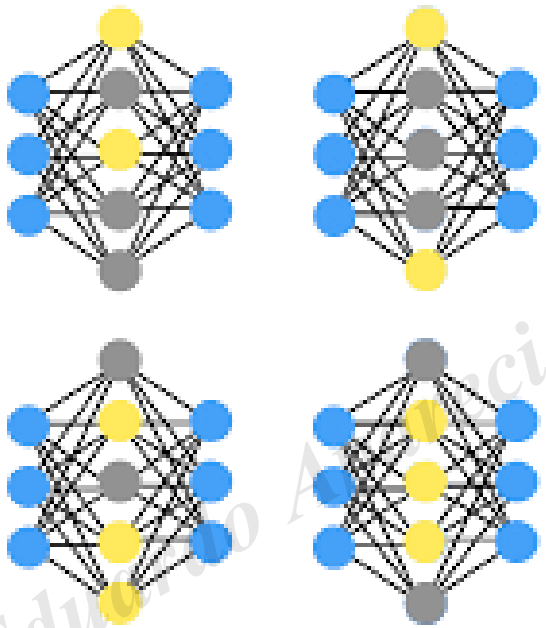
Autoencoder Types

- Deep Autoencoder



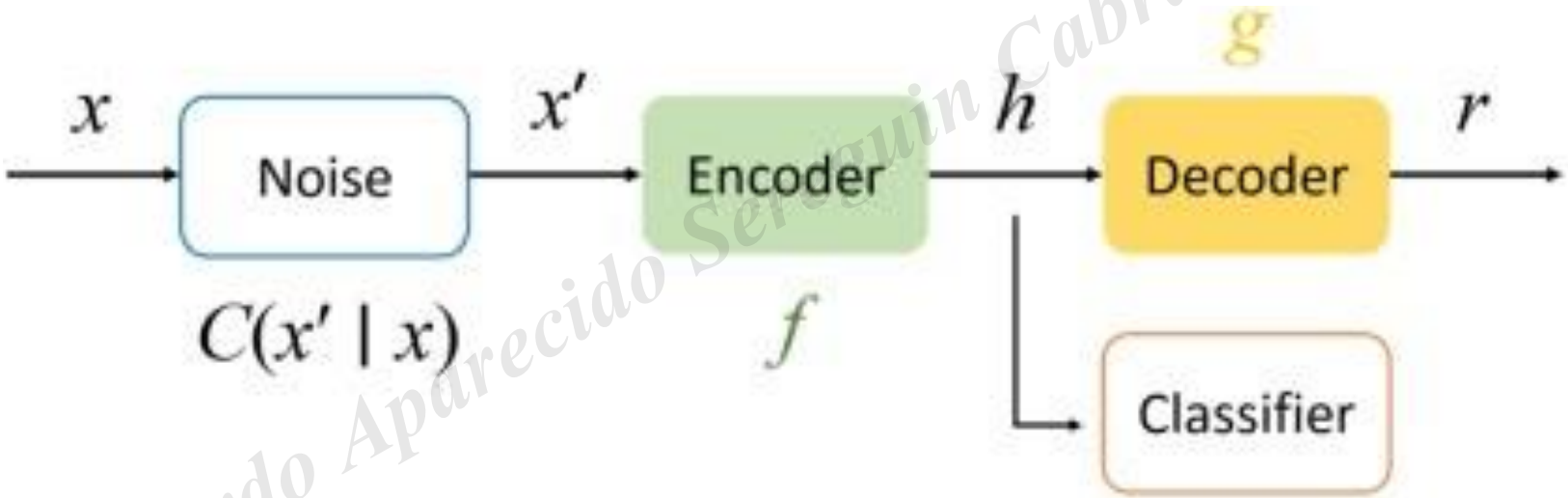
Autoencoder Types

- Sparse Autoencoder
- Regularization term is applied.



Autoencoder Types

- Denoising Autoencoder



Autoencoder Types

- Contractive Autoencoder
- Difference regarding denoising – includes penalty in the cost function
- Denoising includes in training data
- It obtains more interesting results than denoising

GAN

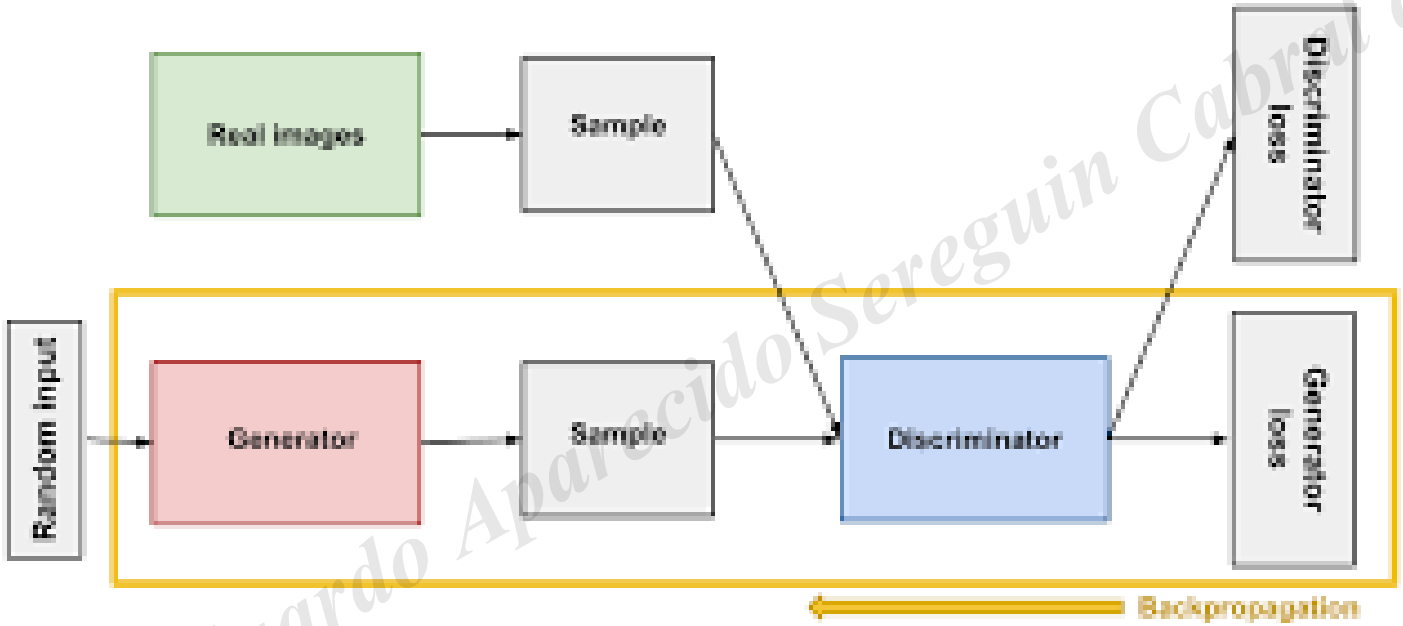
- Generative Adversarial Networks
- "the most interesting idea in the last 10 years in Machine Learning"
- Two networks competing
- Generative Model

GAN

GAN

- How to obtain y when x is given?
- Generative models model the individual classes distribution.
- Discriminative models learn the border between classes.

GAN



GAN

- MNIST example
- Generator will generate “fake” images
- We will pass real images with the generated ones
- The discriminator must recognize the images and make the difference

Practical part

GAN

- We perform backpropagation
- Generator will generate increasingly better images
- The discriminator must be increasingly better when discriminating
- The discriminator receives real and true images and returns probabilities, a number between 0 and 1, with 1 representing an authentic image prediction and 0 representing prediction of false images (generated by the generative network).

Discussion

https://www.linkedin.com/in/jeronymo-marcondes-585a26186