Artificial Neural Networks

RESTRICTED BOLTZMANN MACHINES

Topics

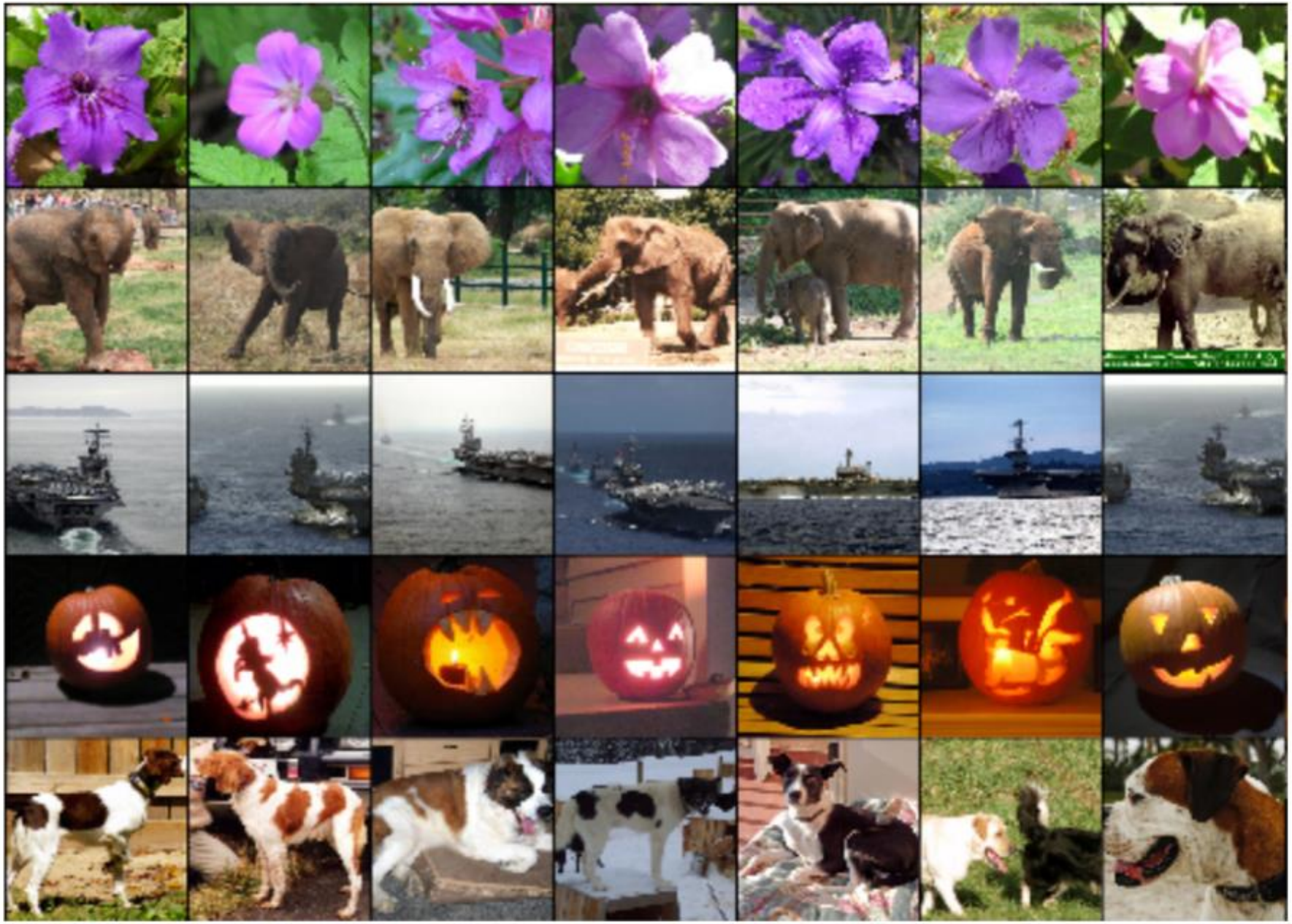
1. Introduction
   a) Motivation
   b) Training networks with unlabeled data
   c) Vanishing gradient problem
2. Restricted Boltzmann Machine
   a) Structure
   b) Training
   c) Combining Boltzmann Machines into a Belief Network

Introduction

Neural networks are learning machines which learn from data in a supervised way

In practice the data can be:

1. Large in size (e.g. Images with several MPixels)
2. Large in number
3. Contain complicated patterns (requiring multilayer networks)
4. Unlabeled (Data coming from sensors, etc.)



Motivation

A main challenge in using the Neural Networks therefore, is developing networks which can be trained to cluster unlabeled data

Vanishing Gradient

A second challenge in using neural networks is the time required for training the network

This situation is further complicated considering that

1. The data dimension is large
2. Number of data items for training is large
3. Patterns are complex (requiring larger number of hidden layers)
4. Gradient-based training can be very slow when the gradient value is small (in plateaus) (Vanishing gradient)

Learning from Unlabeled Data

In learning from unlabeled data, the training of the model cannot rely on the samples taken from each class.

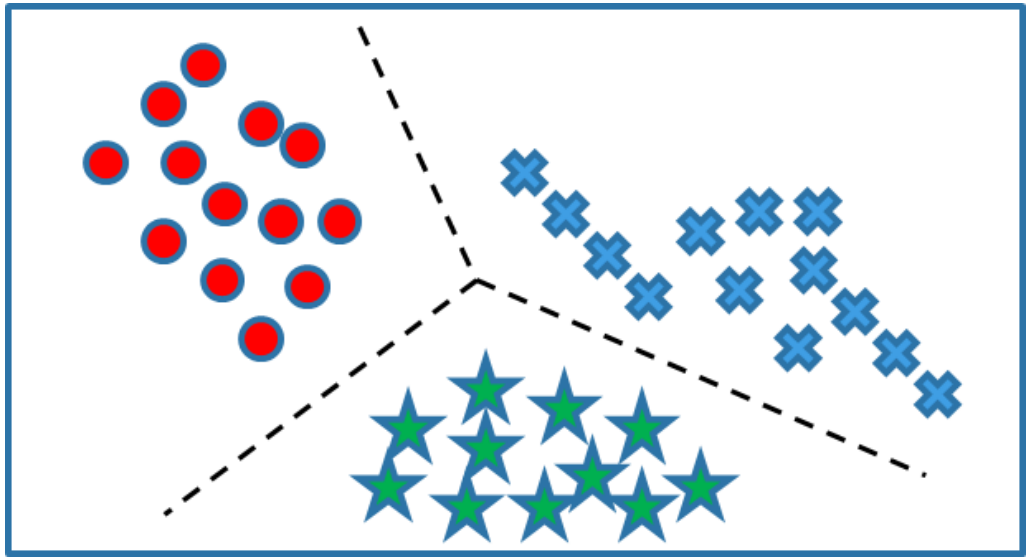
The main criteria here is the proximity of the sample points in the feature space.

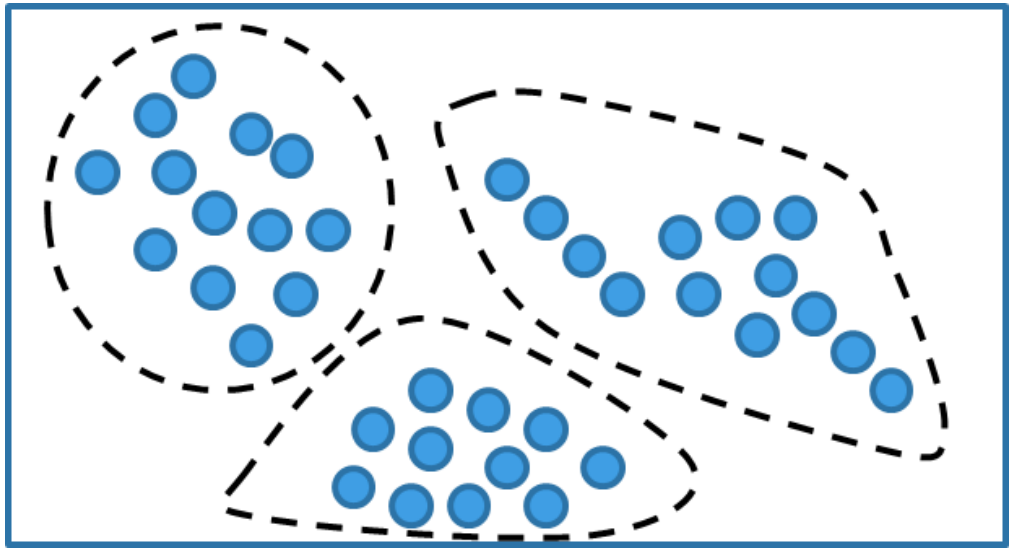
Supervised Learning vs. Unsupervised Learning

Classification is referred to as supervised learning where the training data is tagged

Clustering only considers the proximity of object in feature space for categorization

Supervised Learning vs. Unsupervised Learning





Classification

Clustering

K-means

Iteratively re-assign points to the nearest cluster center

Update cluster center (mean)

Problem:

- Number of clusters should be given

Boltzmann Machine

Boltzmann machine is a network of symmetrically connected, neuron-like units that make stochastic decisions about whether to be on or off.

Boltzmann machines have a simple learning algorithm that allows them to discover interesting features that represent complex regularities in the training data.

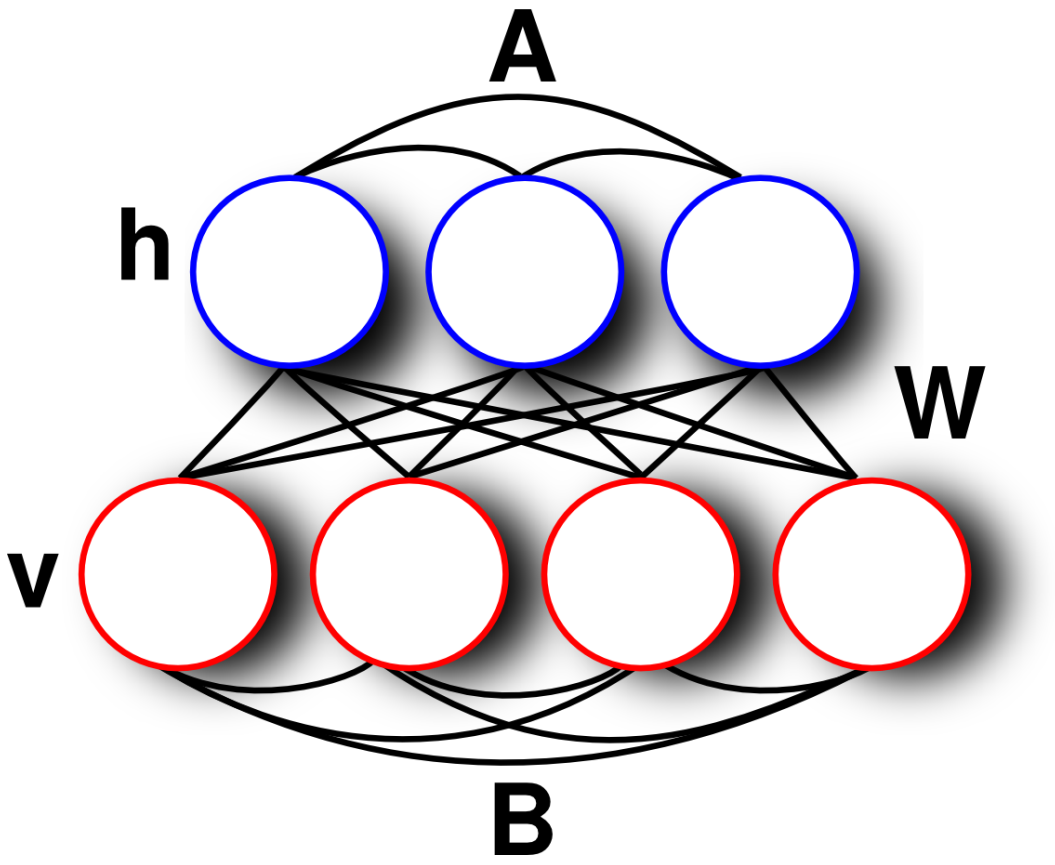
Boltzmann Machine

A Boltzmann machine consists of two layers:

- Hidden layer
- Visible layer

Visible layer is the input layer

Boltzmann Machine

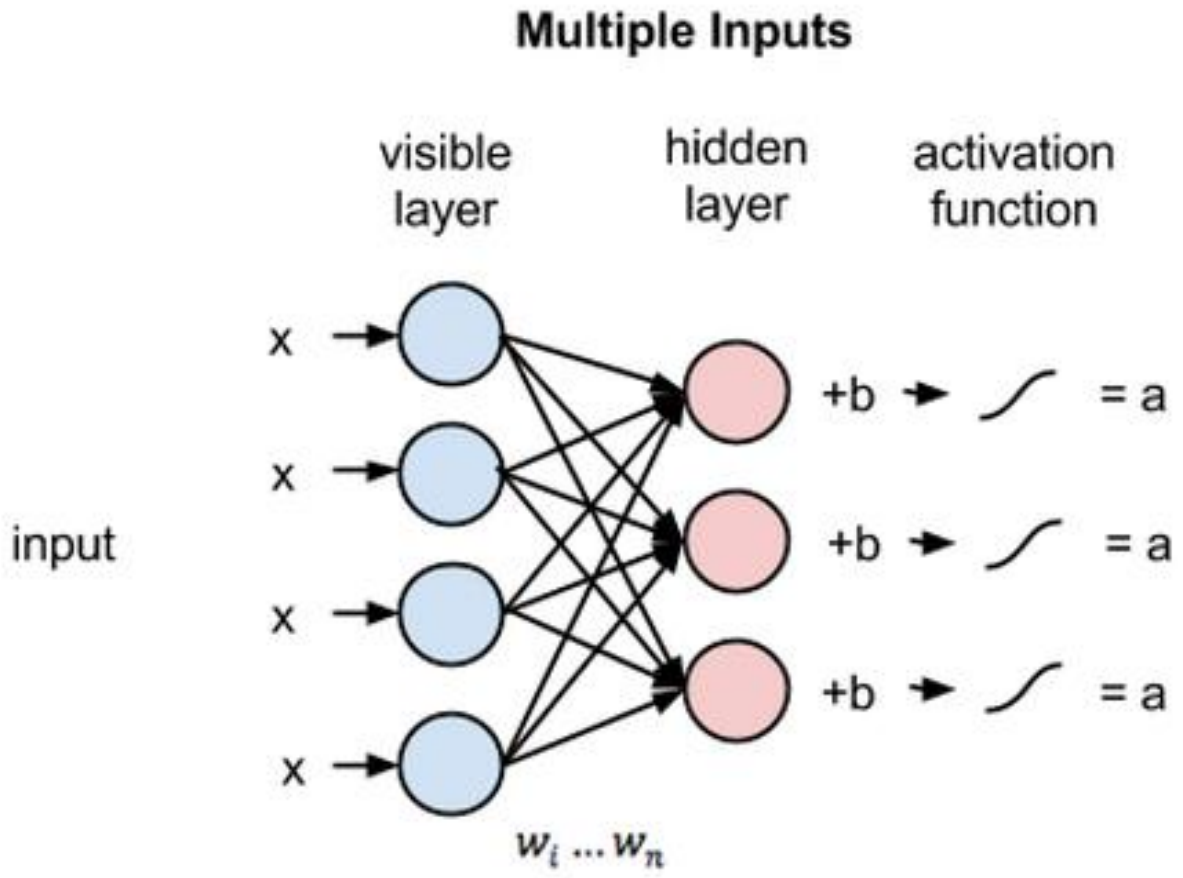


Restricted Boltzmann Machines

The restricted Boltzmann machines do not let the connections between neurons in the same layer (Hidden-hidden, visible-visible)

However, each input neuron is connected to every hidden layer neuron

Restricted Boltzmann Machines



Restricted Boltzmann Machines

The visible layer gets the input and forwards it to the hidden layer through weights (W_{ij}).

The hidden layer adds up the inputs, passes the sum through a sigmoid function, and obtains output ***α***

Reconstruction Phase

In the reconstruction phase, the layers change position.

The output(s) a are given as input to the hidden layer.

The hidden layer multiplies them by weights W_{ij}, and passes to the visible layer

The visible layer adds up the weighted values from all hidden layer neurons and passes it through a sigmoid activation function

Comparing the Results

The original input and the reconstructed values are compared.

The training phase of the restricted Boltzmann machine aims at minimizing the difference between the input and the reconstructed value (error)

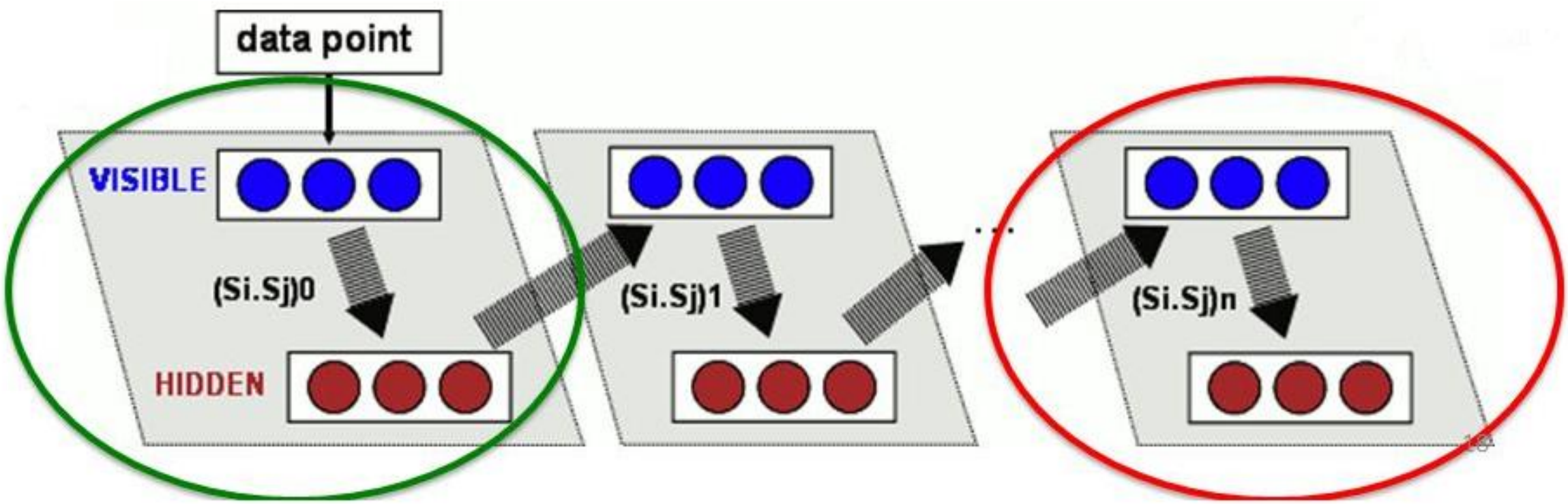
Equilibrium State

The reconstructed value obtained from the reconstruction phase is given as input to the restricted Boltzmann machine.

The output from the hidden layer is fed back as input in a new reconstruction phase.

These steps are repeated until the input and the reconstructed values are similar up to a threshold. This state is called ***equilibrium*** state

Equilibrium State



Training a Restricted Boltzmann Machine

Theoretically the learning happens by updating the weights using the difference between the initial state and the equilibrium state as:

$$\Delta W_{ij} = \varepsilon \left(< v_i h_j >^0 - < v_i h_j >^\infty \right.$$

However, this can slow down the learning process. Therefore, instead of the equilibrium state, the difference between the first and the second states are used

Kullback-Leibler Divergence

KL-Divergence is used to compare two distributions

The goal is using less parameters to represent information. Therefore, a metric is needed to measure the amount of information loss when a simpler model is used for data

Kullback-Leibler Divergence

Entropy (H) is used to measure the amount of information and hence, the number of bits required to represent a data item

$$H = -\sum_{k=1}^{n} P(x_i).\log(p(x_i))$$

KL-Divergence measures the amount of data loss

KL-Divergence

Assume the new approximating distribution is given by p’. The data loss is given by:

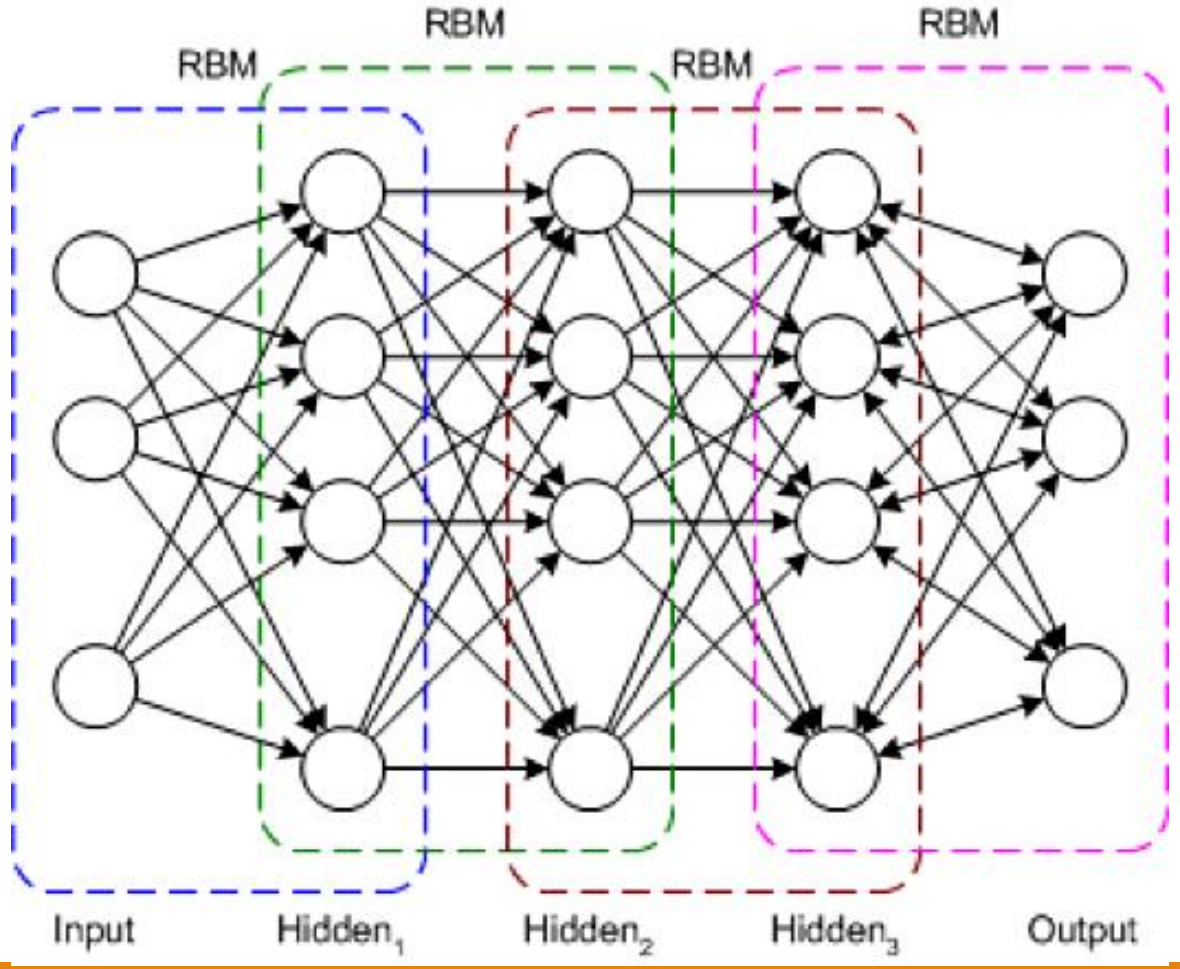
$$D_{KL}(p, p') = \sum_{k=1}^{n} P(x_i).\left(\log\left(p(x_i)\right) - \log\left(p'(x_i)\right)\right)$$

KL-Divergence is NOT a distance measure

Belief Networks

Belief networks are created by stacking the restricted Boltzmann machines where, the hidden layer of an RBM is visible layer of the next RBM.

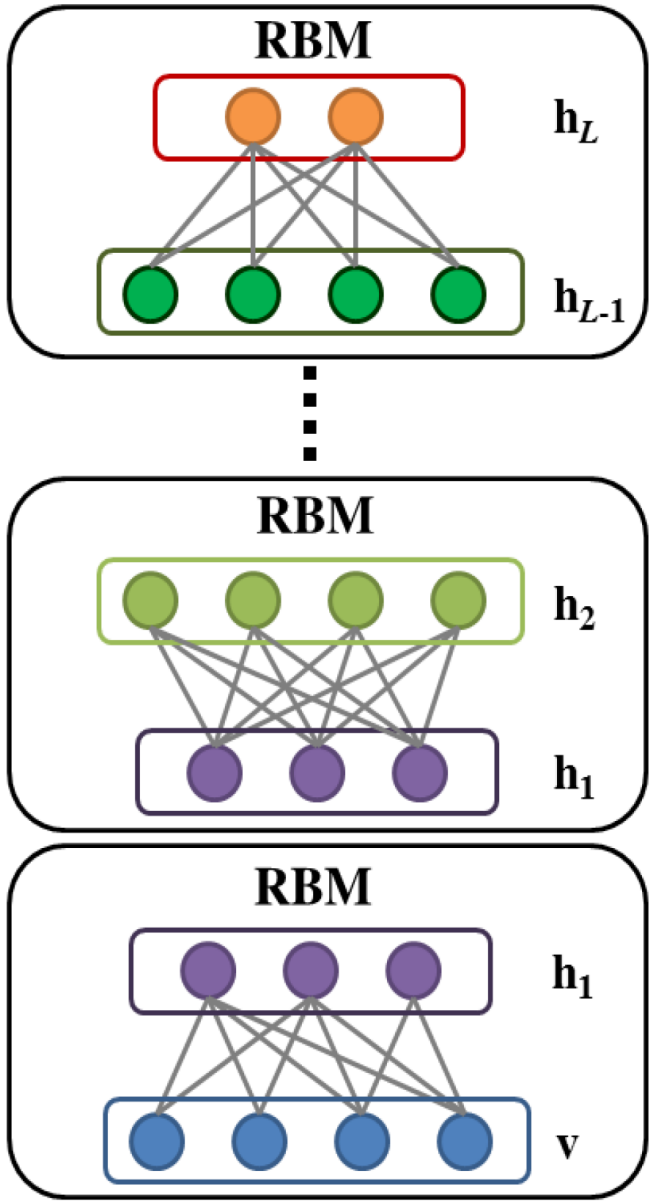
Belief Networks



Belief Networks

The training is performed starting from the first RBM.

The output of the first RBM is used as the visible layer values of the second RBM.



Labeling Clusters

A small set of labeled samples can be used for labeling the clusters.

The labeling can be done after finishing the network training

Belief Networks vis-à-vis Convolutional Neural Networks

In a convolutional neural network, the first part of the network extracts the features (the convolution part) while the second part uses those features for classification.

In belief networks all layers are involved in developing the model for the pattern (gradually as the input propagates through RBMs)

Example

MNIST Data set:

MNIST is a database of handwritten digits, available from http://yann.lecun.com/exdb/mnist/

MNIST has a training set of 60,000 examples, and a test set of 10,000 examples.

It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image.

The numbers have been written by more than 500 people and are given as 28x28 matrices.

MNIST Data Set

MNIST Data Set

The webpage includes a list of methods used for classifying the data and their accuracy rates.

References to the methods are also available.