Title: Deep belief network

URL: https://en.wikipedia.org/wiki/Deep_belief_network

PageID: 41416740

Categories: Category:Neural network architectures, Category:Probabilistic models

-----

Supervised learning

Unsupervised learning

Semi-supervised learning

Self-supervised learning

Reinforcement learning

Meta-learning

Online learning

Batch learning

Curriculum learning

Rule-based learning

Neuro-symbolic AI

Neuromorphic engineering

Quantum machine learning

Classification

Generative modeling

Regression

Clustering

Dimensionality reduction

Density estimation

Anomaly detection

Data cleaning

AutoML

Association rules

Semantic analysis

Structured prediction

Feature engineering

Feature learning

Learning to rank

Grammar induction

Ontology learning

Multimodal learning

Apprenticeship learning

Decision trees

Ensembles Bagging Boosting Random forest

Bagging

Boosting

Random forest

k -NN

Linear regression

Naive Bayes

Artificial neural networks

Logistic regression

Perceptron

Relevance vector machine (RVM)

Support vector machine (SVM)

BIRCH

CURE

Hierarchical

k -means

Fuzzy

Expectation–maximization (EM)

DBSCAN

OPTICS

Mean shift

Factor analysis

CCA

ICA

LDA

NMF

PCA

PGD

t-SNE

SDL

Graphical models Bayes net Conditional random field Hidden Markov

Bayes net

Conditional random field

Hidden Markov

RANSAC

k -NN

Local outlier factor
Isolation forest
Autoencoder
Deep learning
Feedforward neural network
Recurrent neural network LSTM GRU ESN reservoir computing
LSTM
GRU
ESN
reservoir computing
Boltzmann machine Restricted
Restricted
GAN
Diffusion model
SOM
Convolutional neural network U-Net LeNet AlexNet DeepDream
U-Net
LeNet
AlexNet
DeepDream
Neural field Neural radiance field Physics-informed neural networks
Neural radiance field
Physics-informed neural networks
Transformer Vision
Vision
Mamba
Spiking neural network
Memtransistor
Electrochemical RAM (ECRAM)
Q-learning
Policy gradient
SARSA
Temporal difference (TD)
Multi-agent Self-play
Self-play
Active learning
Crowdsourcing
Human-in-the-loop

Mechanistic interpretability

RLHF

Coefficient of determination

Confusion matrix

Learning curve

ROC curve

Kernel machines

Bias–variance tradeoff

Computational learning theory

Empirical risk minimization

Occam learning

PAC learning

Statistical learning

VC theory

Topological deep learning

AAAI

ECML PKDD

NeurIPS

ICML

ICLR

IJCAI

ML

JMLR

Glossary of artificial intelligence

List of datasets for machine-learning research List of datasets in computer vision and image processing

List of datasets in computer vision and image processing

Outline of machine learning

v

t

e

In machine learning , a deep belief network (DBN) is a generative graphical model , or alternatively a class of deep neural network , composed of multiple layers of latent variables ("hidden units"), with connections between the layers but not between units within each layer. [1]

When trained without supervision on a set of examples , a DBN can learn to probabilistically reconstruct its inputs. The layers then act as feature detectors . [1] After this learning step, a DBN can be further trained with supervision to perform classification . [2]

DBNs can be viewed as a composition of simple, unsupervised networks such as restricted Boltzmann machines (RBMs) [1] or autoencoders , [3] where each sub-network's hidden layer serves as the visible layer for the next. An RBM is an undirected , generative energy-based model with a "visible" input layer and a hidden layer and connections between but not within layers. This composition leads to a fast, layer-by-layer unsupervised training procedure, where contrastive divergence is applied to each sub-network in turn, starting from the "lowest" pair of layers (the lowest visible layer is a training set).

The observation [2] that DBNs can be trained greedily , one layer at a time, led to one of the first effective deep learning algorithms. [4] : 6 Overall, there are many attractive implementations and uses of DBNs in real-life applications and scenarios (e.g., electroencephalography , [5] drug discovery [6] [7] [8]).

Training

The training method for RBMs proposed by Geoffrey Hinton for use with training " Product of Experts " models is called contrastive divergence (CD). [9] CD provides an approximation to the maximum likelihood method that would ideally be applied for learning the weights. [10] [11] In training a single RBM, weight updates are performed with gradient descent via the following equation: w i j (t + 1) = w i j (t) + η ∂ log ■ (p (v)) ∂ w i j {\displaystyle w_{ij}(t+1)=w_{ij}(t)+\eta {\frac {\partial \log(p(v))}{\partial w_{ij}}}}

where, p (v) {\displaystyle p(v)} is the probability of a visible vector, which is given by p (v) = 1 Z ∑ h e − E (v , h) {\displaystyle p(v)={\frac {1}{Z}}\sum _{h}e^{-E(v,h)}} . Z {\displaystyle Z} is the partition function (used for normalizing) and E (v , h) {\displaystyle E(v,h)} is the energy function assigned to the state of the network. A lower energy indicates the network is in a more "desirable" configuration. The gradient ∂ log ■ (p (v)) ∂ w i j {\displaystyle {\frac {\partial \log(p(v))}{\partial w_{ij}}}} has the simple form ■ v i h j ■ data − ■ v i h j ■ model {\displaystyle \langle v_{i}h_{j}\rangle _{\text{data}}-\langle v_{i}h_{j}\rangle _{\text{model}}} where ■ ■ ■ p {\displaystyle \langle \cdots \rangle _{p}} represent averages with respect to distribution p {\displaystyle p} . The issue arises in sampling ■ v i h j ■ model {\displaystyle \langle v_{i}h_{j}\rangle _{\text{model}}} because this requires extended alternating Gibbs sampling . CD replaces this step by running alternating Gibbs sampling for n {\displaystyle n} steps (values of n = 1 {\displaystyle n=1} perform well). After n {\displaystyle n} steps, the data are sampled and that sample is used in place of ■ v i h j ■ model {\displaystyle \langle v_{i}h_{j}\rangle _{\text{model}}} . The CD procedure works as follows: [10]

Initialize the visible units to a training vector.

Update the hidden units in parallel given the visible units: p (h j = 1 ■ V) = σ (b j + ∑ i v i w i j) {\displaystyle p(h_{j}=1\mid {\textbf {V}})=\sigma (b_{j}+\sum _{i}v_{i}w_{ij})} . σ {\displaystyle \sigma } is the sigmoid function and b j {\displaystyle b_{j}} is the bias of h j {\displaystyle h_{j}} .

Update the visible units in parallel given the hidden units: p (v i = 1 ■ H) = σ (a i + ∑ j h j w i j) {\displaystyle p(v_{i}=1\mid {\textbf {H}})=\sigma (a_{i}+\sum _{j}h_{j}w_{ij})} . a i {\displaystyle a_{i}} is the bias of v i {\displaystyle v_{i}} . This is called the "reconstruction" step.

Re-update the hidden units in parallel given the reconstructed visible units using the same equation as in step 2.

Perform the weight update: Δ w i j ∝ ■ v i h j ■ data − ■ v i h j ■ reconstruction {\displaystyle \Delta w_{ij}\propto \langle v_{i}h_{j}\rangle _{\text{data}}-\langle v_{i}h_{j}\rangle _{\text{reconstruction}}} .

Once an RBM is trained, another RBM is "stacked" atop it, taking its input from the final trained layer. The new visible layer is initialized to a training vector, and values for the units in the already-trained layers are assigned using the current weights and biases. The new RBM is then trained with the procedure above. This whole process is repeated until the desired stopping criterion is met. [12]

Although the approximation of CD to maximum likelihood is crude (does not follow the gradient of any function), it is empirically effective. [10]

See also

Bayesian network

Convolutional deep belief network

Deep learning

Energy based model

Stacked Restricted Boltzmann Machine

References

External links

Hinton, Geoffrey E. (2009-05-31). "Deep belief networks" . Scholarpedia . 4 (5): 5947. Bibcode : 2009SchpJ...4.5947H . doi : 10.4249/scholarpedia.5947 . ISSN 1941-6016 .

"Deep Belief Networks" . Deep Learning Tutorials.

"Deep Belief Network Example" . Deeplearning4j Tutorials. Archived from the original on 2016-10-03 . Retrieved 2015-02-22 .