Title: Multilayer perceptron
URL: https://en.wikipedia.org/wiki/Multilayer_perceptron
PageID: 2266644
Categories: Category:Classification algorithms, Category:Neural network architectures

-----

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 deep learning , a multilayer perceptron (MLP) is a name for a modern feedforward neural network consisting of fully connected neurons with nonlinear activation functions , organized in layers, notable for being able to distinguish data that is not linearly separable . [1]

Modern neural networks are trained using backpropagation [2] [3] [4] [5] [6] and are colloquially referred to as "vanilla" networks. [7] MLPs grew out of an effort to improve single-layer perceptrons , which could only be applied to linearly separable data. A perceptron traditionally used a Heaviside step function as its nonlinear activation function. However, the backpropagation algorithm requires that modern MLPs use continuous activation functions such as sigmoid or ReLU . [8]

Multilayer perceptrons form the basis of deep learning, [9] and are applicable across a vast set of diverse domains. [10]

Timeline

In 1943, Warren McCulloch and Walter Pitts proposed the binary artificial neuron as a logical model of biological neural networks. [11]

In 1958, Frank Rosenblatt proposed the multilayered perceptron model, consisting of an input layer, a hidden layer with randomized weights that did not learn, and an output layer with learnable connections. [12]

In 1962, Rosenblatt published many variants and experiments on perceptrons in his book Principles of Neurodynamics , including up to 2 trainable layers by "back-propagating errors". [13] However, it was not the backpropagation algorithm, and he did not have a general method for training multiple layers.

In 1965, Alexey Grigorevich Ivakhnenko and Valentin Lapa published Group Method of Data Handling . It was one of the first deep learning methods, used to train an eight-layer neural net in 1971. [14] [15] [16]

In 1967, Shun'ichi Amari reported [17] the first multilayered neural network trained by stochastic gradient descent , was able to classify non-linearily separable pattern classes. Amari's student Saito conducted the computer experiments, using a five-layered feedforward network with two learning layers. [16]

Backpropagation was independently developed multiple times in early 1970s. The earliest published instance was Seppo Linnainmaa 's master thesis (1970). [18] [19] [16] Paul Werbos developed it independently in 1971, [20] but had difficulty publishing it until 1982. [21]

In 1986, David E. Rumelhart et al. popularized backpropagation. [22] [23]

In 2003, interest in backpropagation networks returned due to the successes of deep learning being applied to language modelling by Yoshua Bengio with co-authors. [24]

In 2021, a very simple NN architecture combining two deep MLPs with skip connections and layer normalizations was designed and called MLP-Mixer; its realizations featuring 19 to 431 millions of parameters were shown to be comparable to vision transformers of similar size on ImageNet and similar image classification tasks. [25]

Mathematical foundations

Activation function

If a multilayer perceptron has a linear activation function in all neurons, that is, a linear function that maps the weighted inputs to the output of each neuron, then linear algebra shows that any number of layers can be reduced to a two-layer input-output model. In MLPs some neurons use a nonlinear activation function that was developed to model the frequency of action potentials , or firing, of biological neurons.

The two historically common activation functions are both sigmoids , and are described by

The first is a hyperbolic tangent that ranges from –1 to 1, while the other is the logistic function , which is similar in shape but ranges from 0 to 1. Here y i {\displaystyle y_{i}} is the output of the i {\displaystyle i} th node (neuron) and v i {\displaystyle v_{i}} is the weighted sum of the input connections. Alternative activation functions have been proposed, including the rectifier and softplus functions. More specialized activation functions include radial basis functions (used in radial basis networks , another class of supervised neural network models).

In recent developments of deep learning the rectified linear unit (ReLU) is more frequently used as one of the possible ways to overcome the numerical problems related to the sigmoids.

Layers

The MLP consists of three or more layers (an input and an output layer with one or more hidden layers) of nonlinearly-activating nodes. Since MLPs are fully connected, each node in one layer connects with a certain weight w i j {\displaystyle w_{ij}} to every node in the following layer.

Learning

Learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result. This is an example of supervised learning , and is carried out through backpropagation , a generalization of the least mean squares algorithm in the linear perceptron.

We can represent the degree of error in an output node j {\displaystyle j} in the n {\displaystyle n} th data point (training example) by e j (n) = d j (n) − y j (n) {\displaystyle e_{j}(n)=d_{j}(n)-y_{j}(n)} , where d j (n) {\displaystyle d_{j}(n)} is the desired target value for n {\displaystyle n} th data point at node j {\displaystyle j} , and y j (n) {\displaystyle y_{j}(n)} is the value produced by the perceptron at node j {\displaystyle j} when the n {\displaystyle n} th data point is given as an input.

The node weights can then be adjusted based on corrections that minimize the error in the entire output for the n {\displaystyle n} th data point, given by

Using gradient descent , the change in each weight w i j {\displaystyle w_{ij}} is

where y i (n) {\displaystyle y_{i}(n)} is the output of the previous neuron i {\displaystyle i} , and η {\displaystyle \eta } is the learning rate , which is selected to ensure that the weights quickly converge to a response, without oscillations. In the previous expression, ∂ E (n) ∂ v j (n) {\displaystyle {\frac {\partial {\mathcal {E}}(n)}{\partial v_{j}(n)}}} denotes the partial derivate of the error E (n) {\displaystyle {\mathcal {E}}(n)} according to the weighted sum v j (n) {\displaystyle v_{j}(n)} of the input connections of neuron i {\displaystyle i} .

The derivative to be calculated depends on the induced local field v j {\displaystyle v_{j}} , which itself varies. It is easy to prove that for an output node this derivative can be simplified to

where φ ′ {\displaystyle \phi ^{\prime }} is the derivative of the activation function described above, which itself does not vary. The analysis is more difficult for the change in weights to a hidden node, but it can be shown that the relevant derivative is

This depends on the change in weights of the k {\displaystyle k} th nodes, which represent the output layer. So to change the hidden layer weights, the output layer weights change according to the derivative of the activation function, and so this algorithm represents a backpropagation of the activation function. [26]

References

External links

Weka: Open source data mining software with multilayer perceptron implementation .

Neuroph Studio documentation, implements this algorithm and a few others .

Bias–variance tradeoff
Double descent
Overfitting
Clustering
Gradient descent SGD Quasi-Newton method Conjugate gradient method
SGD
Quasi-Newton method
Conjugate gradient method
Backpropagation
Attention
Convolution
Normalization Batchnorm
Batchnorm
Activation Softmax Sigmoid Rectifier
Softmax
Sigmoid
Rectifier
Gating
Weight initialization
Regularization
Datasets Augmentation
Augmentation
Prompt engineering
Reinforcement learning Q-learning SARSA Imitation Policy gradient
Q-learning
SARSA
Imitation
Policy gradient
Diffusion
Latent diffusion model
Autoregression
Adversary
RAG
Uncanny valley
RLHF
Self-supervised learning
Reflection
Recursive self-improvement

Hallucination

Word embedding

Vibe coding

Machine learning In-context learning

In-context learning

Artificial neural network Deep learning

Deep learning

Language model Large language model NMT

Large language model

NMT

Reasoning language model

Model Context Protocol

Intelligent agent

Artificial human companion

Humanity's Last Exam

Artificial general intelligence (AGI)

AlexNet

WaveNet

Human image synthesis

HWR

OCR

Computer vision

Speech synthesis 15.ai ElevenLabs

15.ai

ElevenLabs

Speech recognition Whisper

Whisper

Facial recognition

AlphaFold

Text-to-image models Aurora DALL-E Firefly Flux Ideogram Imagen Midjourney Recraft Stable Diffusion

Aurora

DALL-E

Firefly

Flux

Ideogram

Imagen

Midjourney

Recraft

Stable Diffusion

Text-to-video models Dream Machine Runway Gen Hailuo AI Kling Sora Veo

Dream Machine

Runway Gen

Hailuo AI

Kling

Sora

Veo

Music generation Riffusion Suno AI Udio

Riffusion

Suno AI

Udio

Word2vec

Seq2seq

GloVe

BERT

T5

Llama

Chinchilla AI

PaLM

GPT 1 2 3 J ChatGPT 4 4o o1 o3 4.5 4.1 o4-mini 5

1

2

3

J

ChatGPT

4

4o

o1

o3

4.5

4.1

o4-mini

5

Claude

Gemini Gemini (language model) Gemma

Gemini (language model)

Gemma
Grok
LaMDA
BLOOM
DBRX
Project Debater
IBM Watson
IBM Watsonx
Granite
PanGu-Σ
DeepSeek
Qwen
AlphaGo
AlphaZero
OpenAI Five
Self-driving car
MuZero
Action selection AutoGPT
AutoGPT
Robot control
Alan Turing
Warren Sturgis McCulloch
Walter Pitts
John von Neumann
Claude Shannon
Shun'ichi Amari
Kunihiko Fukushima
Takeo Kanade
Marvin Minsky
John McCarthy
Nathaniel Rochester
Allen Newell
Cliff Shaw
Herbert A. Simon
Oliver Selfridge
Frank Rosenblatt
Bernard Widrow
Joseph Weizenbaum

Seymour Papert
Seppo Linnainmaa
Paul Werbos
Geoffrey Hinton
John Hopfield
Jürgen Schmidhuber
Yann LeCun
Yoshua Bengio
Lotfi A. Zadeh
Stephen Grossberg
Alex Graves
James Goodnight
Andrew Ng
Fei-Fei Li
Alex Krizhevsky
Ilya Sutskever
Oriol Vinyals
Quoc V. Le
Ian Goodfellow
Demis Hassabis
David Silver
Andrej Karpathy
Ashish Vaswani
Noam Shazeer
Aidan Gomez
John Schulman
Mustafa Suleyman
Jan Leike
Daniel Kokotajlo
François Chollet
Neural Turing machine
Differentiable neural computer
Transformer Vision transformer (ViT)
Vision transformer (ViT)
Recurrent neural network (RNN)
Long short-term memory (LSTM)
Gated recurrent unit (GRU)
Echo state network

Multilayer perceptron (MLP)
Convolutional neural network (CNN)
Residual neural network (RNN)
Highway network
Mamba
Autoencoder
Variational autoencoder (VAE)
Generative adversarial network (GAN)
Graph neural network (GNN)
Category