

# Machine learning

For the journal, see [Machine Learning \(journal\)](#).

**Machine learning** is a subfield of [computer science](#)<sup>[1]</sup> that evolved from the study of [pattern recognition](#) and [computational learning theory](#) in [artificial intelligence](#).<sup>[1]</sup> In 1959, [Arthur Samuel](#) defined machine learning as a “Field of study that gives computers the ability to learn without being explicitly programmed”.<sup>[2]</sup> Machine learning explores the study and construction of [algorithms](#) that can [learn](#) from and make predictions on [data](#).<sup>[3]</sup> Such algorithms operate by building a [model](#) from example inputs in order to make data-driven predictions or decisions expressed as outputs,<sup>[4]:2</sup> rather than following strictly static program instructions.

Machine learning is closely related to and often overlaps with [computational statistics](#); a discipline which also focuses in prediction-making through the use of computers. It has strong ties to [mathematical optimization](#), which delivers methods, theory and application domains to the field. Machine learning is employed in a range of computing tasks where designing and programming explicit algorithms is unfeasible. Example applications include [spam filtering](#), [optical character recognition \(OCR\)](#),<sup>[5]</sup> [search engines](#) and [computer vision](#). Machine learning is sometimes conflated with [data mining](#),<sup>[6]</sup> where the latter sub-field focuses more on exploratory data analysis and is known as [unsupervised learning](#).<sup>[4]:vii[7]</sup>

Within the field of data analytics, machine learning is a method used to devise complex models and algorithms that lend themselves to prediction. These analytical models allow researchers, [data scientists](#), engineers, and analysts to “produce reliable, repeatable decisions and results” and uncover “hidden insights” through learning from historical relationships and trends in the data.<sup>[8]</sup>

## 1 Overview

[Tom M. Mitchell](#) provided a widely quoted, more formal definition: “A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P* if its performance at tasks in *T*, as measured by *P*, improves with experience *E*”.<sup>[9]</sup> This definition is notable for its defining machine learning in fundamentally [operational](#) rather than cognitive terms, thus following [Alan Turing](#)'s proposal in his paper “[Computing Machinery and Intelligence](#)” that the question “Can machines think?” be replaced with the ques-

tion “Can machines do what we (as thinking entities) can do?”<sup>[10]</sup>

### 1.1 Types of problems and tasks

Machine learning tasks are typically classified into three broad categories, depending on the nature of the learning “signal” or “feedback” available to a learning system. These are<sup>[11]</sup>

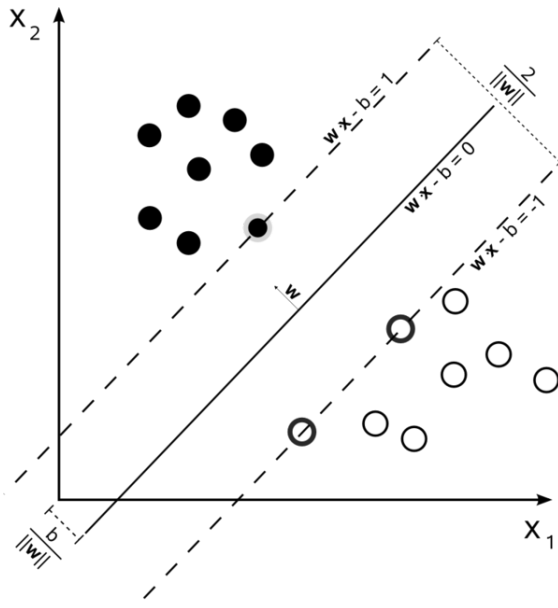
- **Supervised learning:** The computer is presented with example inputs and their desired outputs, given by a “teacher”, and the goal is to learn a general rule that [maps](#) inputs to outputs.
- **Unsupervised learning:** No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end ([feature learning](#)).
- **Reinforcement learning:** A computer program interacts with a dynamic environment in which it must perform a certain goal (such as [driving a vehicle](#)), without a teacher explicitly telling it whether it has come close to its goal. Another example is learning to play a game by playing against an opponent.<sup>[4]:3</sup>

Between supervised and unsupervised learning is [semi-supervised learning](#), where the teacher gives an incomplete training signal: a training set with some (often many) of the target outputs missing. [Transduction](#) is a special case of this principle where the entire set of problem instances is known at learning time, except that part of the targets are missing.

Among other categories of machine learning problems, [learning to learn](#) learns its own [inductive bias](#) based on previous experience. [Developmental learning](#), elaborated for [robot learning](#), generates its own sequences (also called curriculum) of learning situations to cumulatively acquire repertoires of novel skills through autonomous self-exploration and social interaction with human teachers, and using guidance mechanisms such as active learning, maturation, motor synergies, and imitation.

Another categorization of machine learning tasks arises when one considers the desired *output* of a machine-learned system:<sup>[4]:3</sup>

- In [classification](#), inputs are divided into two or more classes, and the learner must produce a model that



A support vector machine is a classifier that divides its input space into two regions, separated by a linear boundary. Here, it has learned to distinguish black and white circles.

assigns unseen inputs to one or more (multi-label classification) of these classes. This is typically tackled in a supervised way. Spam filtering is an example of classification, where the inputs are email (or other) messages and the classes are “spam” and “not spam”.

- In **regression**, also a supervised problem, the outputs are continuous rather than discrete.
- In **clustering**, a set of inputs is to be divided into groups. Unlike in classification, the groups are not known beforehand, making this typically an unsupervised task.
- **Density estimation** finds the distribution of inputs in some space.
- **Dimensionality reduction** simplifies inputs by mapping them into a lower-dimensional space. **Topic modeling** is a related problem, where a program is given a list of **human language** documents and is tasked to find out which documents cover similar topics.

## 2 History and relationships to other fields

As a scientific endeavour, machine learning grew out of the quest for artificial intelligence. Already in the early days of AI as an academic discipline, some researchers were interested in having machines learn from data. They attempted to approach the problem with various symbolic methods, as well as what were then termed

“neural networks”; these were mostly **perceptrons** and other models that were later found to be reinventions of the **generalized linear models** of statistics. Probabilistic reasoning was also employed, especially in automated medical diagnosis.<sup>[11]:488</sup>

However, an increasing emphasis on the logical, **knowledge-based approach** caused a rift between AI and machine learning. Probabilistic systems were plagued by theoretical and practical problems of data acquisition and representation.<sup>[11]:488</sup> By 1980, **expert systems** had come to dominate AI, and statistics was out of favor.<sup>[12]</sup> Work on symbolic/knowledge-based learning did continue within AI, leading to **inductive logic programming**, but the more statistical line of research was now outside the field of AI proper, in **pattern recognition** and **information retrieval**.<sup>[11]:708–710; 755</sup> Neural networks research had been abandoned by AI and computer science around the same time. This line, too, was continued outside the AI/CS field, as “**connectionism**”, by researchers from other disciplines including **Hopfield**, **Rumelhart** and **Hinton**. Their main success came in the mid-1980s with the reinvention of **backpropagation**.<sup>[11]:25</sup>

Machine learning, reorganized as a separate field, started to flourish in the 1990s. The field changed its goal from achieving artificial intelligence to tackling solvable problems of a practical nature. It shifted focus away from the symbolic approaches it had inherited from AI, and toward methods and models borrowed from statistics and **probability theory**.<sup>[12]</sup> It also benefited from the increasing availability of digitized information, and the possibility to distribute that via the **Internet**.

Machine learning and data mining often employ the same methods and overlap significantly. They can be roughly distinguished as follows:

- Machine learning focuses on prediction, based on *known* properties learned from the training data.
- **Data mining** focuses on the **discovery** of (previously) *unknown* properties in the data. This is the analysis step of **Knowledge Discovery in Databases**.

The two areas overlap in many ways: data mining uses many machine learning methods, but often with a slightly different goal in mind. On the other hand, machine learning also employs data mining methods as “unsupervised learning” or as a preprocessing step to improve learner accuracy. Much of the confusion between these two research communities (which do often have separate conferences and separate journals, **ECML PKDD** being a major exception) comes from the basic assumptions they work with: in machine learning, performance is usually evaluated with respect to the ability to *reproduce known* knowledge, while in Knowledge Discovery and Data Mining (KDD) the key task is the discovery of previously *unknown* knowledge. Evaluated with respect to known knowledge, an uninformed (unsupervised) method will easily be outperformed by supervised

methods, while in a typical KDD task, supervised methods cannot be used due to the unavailability of training data.

Machine learning also has intimate ties to optimization: many learning problems are formulated as minimization of some **loss function** on a training set of examples. Loss functions express the discrepancy between the predictions of the model being trained and the actual problem instances (for example, in classification, one wants to assign a label to instances, and models are trained to correctly predict the pre-assigned labels of a set examples). The difference between the two fields arises from the goal of generalization: while optimization algorithms can minimize the loss on a training set, machine learning is concerned with minimizing the loss on unseen samples.<sup>[13]</sup>

## 2.1 Relation to statistics

Machine learning and **statistics** are closely related fields. According to **Michael I. Jordan**, the ideas of machine learning, from methodological principles to theoretical tools, have had a long pre-history in statistics.<sup>[14]</sup> He also suggested the term **data science** as a placeholder to call the overall field.<sup>[14]</sup>

**Leo Breiman** distinguished two statistical modelling paradigms: data model and algorithmic model,<sup>[15]</sup> wherein 'algorithmic model' means more or less the machine learning algorithms like **Random forest**.

Some statisticians have adopted methods from machine learning, leading to a combined field that they call *statistical learning*.<sup>[16]</sup>

## 3 Theory

Main article: **Computational learning theory**

A core objective of a learner is to generalize from its experience.<sup>[17][18]</sup> Generalization in this context is the ability of a learning machine to perform accurately on new, unseen examples/tasks after having experienced a learning data set. The training examples come from some generally unknown probability distribution (considered representative of the space of occurrences) and the learner has to build a general model about this space that enables it to produce sufficiently accurate predictions in new cases.

The computational analysis of machine learning algorithms and their performance is a branch of **theoretical computer science** known as **computational learning theory**. Because training sets are finite and the future is uncertain, learning theory usually does not yield guarantees of the performance of algorithms. Instead, probabilistic bounds on the performance are quite common. The

**bias–variance decomposition** is one way to quantify generalization error.

How well a model trained with existing examples predicts the output for unknown instances is called generalization. For best generalization, complexity of the hypothesis should match the complexity of the function underlying the data. If the hypothesis is less complex than the function, we've underfitted. Then, we increase the complexity, the training error decreases. But if our hypothesis is too complex, we've overfitted. After then, we should find the hypothesis that has the minimum training error.<sup>[19]</sup>

In addition to performance bounds, computational learning theorists study the time complexity and feasibility of learning. In computational learning theory, a computation is considered feasible if it can be done in **polynomial time**. There are two kinds of time complexity results. Positive results show that a certain class of functions can be learned in polynomial time. Negative results show that certain classes cannot be learned in polynomial time.

There are many similarities between machine learning theory and **statistical inference**, although they use different terms.

## 4 Approaches

Main article: **List of machine learning algorithms**

### 4.1 Decision tree learning

Main article: **Decision tree learning**

Decision tree learning uses a **decision tree** as a **predictive model**, which maps observations about an item to conclusions about the item's target value.

### 4.2 Association rule learning

Main article: **Association rule learning**

Association rule learning is a method for discovering interesting relations between variables in large databases.

### 4.3 Artificial neural networks

Main article: **Artificial neural network**

An **artificial neural network** (ANN) learning algorithm, usually called "neural network" (NN), is a learning algorithm that is inspired by the structure and functional aspects of **biological neural networks**. Compu-

tations are structured in terms of an interconnected group of **artificial neurons**, processing information using a **connectionist** approach to **computation**. Modern neural networks are **non-linear statistical data modeling** tools. They are usually used to model complex relationships between inputs and outputs, to **find patterns** in data, or to capture the statistical structure in an unknown **joint probability distribution** between observed variables.

## 4.4 Deep Learning

Main article: **Deep learning**

Falling hardware prices and the development of GPUs for personal use in the last few years have contributed to the development of the concept of **Deep learning** which consists of multiple hidden layers in an artificial neural network. This approach tries to model the way the human brain processes light and sound into vision and hearing. Some successful applications of deep learning are computer vision<sup>[20]</sup> and speech recognition.<sup>[21]</sup>

## 4.5 Inductive logic programming

Main article: **Inductive logic programming**

Inductive logic programming (ILP) is an approach to rule learning using **logic programming** as a uniform representation for input examples, background knowledge, and hypotheses. Given an encoding of the known background knowledge and a set of examples represented as a logical database of facts, an ILP system will derive a hypothesized logic program that **entails** all positive and no negative examples. **Inductive programming** is a related field that considers any kind of programming languages for representing hypotheses (and not only logic programming), such as functional programs.

## 4.6 Support vector machines

Main article: **Support vector machines**

Support vector machines (SVMs) are a set of related **supervised learning** methods used for **classification** and **regression**. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that predicts whether a new example falls into one category or the other.

## 4.7 Clustering

Main article: **Cluster analysis**

Cluster analysis is the assignment of a set of observations into subsets (called *clusters*) so that observations within the same cluster are similar according to some pre-designated criterion or criteria, while observations drawn from different clusters are dissimilar. Different clustering techniques make different assumptions on the structure of the data, often defined by some *similarity metric* and evaluated for example by *internal compactness* (similarity between members of the same cluster) and *separation* between different clusters. Other methods are based on *estimated density* and *graph connectivity*. Clustering is a method of **unsupervised learning**, and a common technique for **statistical data analysis**.

## 4.8 Bayesian networks

Main article: **Bayesian network**

A Bayesian network, belief network or directed acyclic graphical model is a **probabilistic graphical model** that represents a set of **random variables** and their **conditional independencies** via a **directed acyclic graph** (DAG). For example, a Bayesian network could represent the probabilistic relationships between diseases and symptoms. Given symptoms, the network can be used to compute the probabilities of the presence of various diseases. Efficient algorithms exist that perform **inference** and learning.

## 4.9 Reinforcement learning

Main article: **Reinforcement learning**

Reinforcement learning is concerned with how an *agent* ought to take *actions* in an *environment* so as to maximize some notion of long-term *reward*. Reinforcement learning algorithms attempt to find a *policy* that maps *states* of the world to the actions the agent ought to take in those states. Reinforcement learning differs from the **supervised learning** problem in that correct input/output pairs are never presented, nor sub-optimal actions explicitly corrected.

## 4.10 Representation learning

Main article: **Representation learning**

Several learning algorithms, mostly **unsupervised learning** algorithms, aim at discovering better representations of the inputs provided during training. Classical examples include **principal components analysis** and **cluster analysis**. Representation learning algorithms often attempt to preserve the information in their input but transform it in a way that makes it useful, often as a pre-processing step before performing classification or pre-

dictionaries, allowing to reconstruct the inputs coming from the unknown data generating distribution, while not being necessarily faithful for configurations that are implausible under that distribution.

**Manifold learning** algorithms attempt to do so under the constraint that the learned representation is low-dimensional. **Sparse coding** algorithms attempt to do so under the constraint that the learned representation is sparse (has many zeros). **Multilinear subspace learning** algorithms aim to learn low-dimensional representations directly from **tensor** representations for multidimensional data, without reshaping them into (high-dimensional) vectors.<sup>[22]</sup> **Deep learning** algorithms discover multiple levels of representation, or a hierarchy of features, with higher-level, more abstract features defined in terms of (or generating) lower-level features. It has been argued that an intelligent machine is one that learns a representation that disentangles the underlying factors of variation that explain the observed data.<sup>[23]</sup>

#### 4.11 Similarity and metric learning

Main article: [Similarity learning](#)

In this problem, the learning machine is given pairs of examples that are considered similar and pairs of less similar objects. It then needs to learn a similarity function (or a distance metric function) that can predict if new objects are similar. It is sometimes used in Recommendation systems.

#### 4.12 Sparse dictionary learning

Main article: [Sparse dictionary learning](#)

In this method, a datum is represented as a linear combination of basis functions, and the coefficients are assumed to be sparse. Let  $x$  be a  $d$ -dimensional datum,  $D$  be a  $d$  by  $n$  matrix, where each column of  $D$  represents a basis function.  $r$  is the coefficient to represent  $x$  using  $D$ . Mathematically, sparse dictionary learning means solving  $x \approx Dr$  where  $r$  is sparse. Generally speaking,  $n$  is assumed to be larger than  $d$  to allow the freedom for a sparse representation.

Learning a dictionary along with sparse representations is **strongly NP-hard** and also difficult to solve approximately.<sup>[24]</sup> A popular heuristic method for sparse dictionary learning is **K-SVD**.

Sparse dictionary learning has been applied in several contexts. In classification, the problem is to determine which classes a previously unseen datum belongs to. Suppose a dictionary for each class has already been built. Then a new datum is associated with the class such that it's best sparsely represented by the corresponding dictionary. Sparse dictionary learning has also been applied

in image de-noising. The key idea is that a clean image patch can be sparsely represented by an image dictionary, but the noise cannot.<sup>[25]</sup>

#### 4.13 Genetic algorithms

Main article: [Genetic algorithm](#)

A genetic algorithm (GA) is a search heuristic that mimics the process of **natural selection**, and uses methods such as **mutation** and **crossover** to generate new **genotype** in the hope of finding good solutions to a given problem. In machine learning, genetic algorithms found some uses in the 1980s and 1990s.<sup>[26][27]</sup> Vice versa, machine learning techniques have been used to improve the performance of genetic and **evolutionary algorithms**.<sup>[28]</sup>

### 5 Applications

Applications for machine learning include:

- Adaptive websites
- Affective computing
- Bioinformatics
- Brain-machine interfaces
- Cheminformatics
- Classifying DNA sequences
- Computational anatomy
- Computer vision, including object recognition
- Detecting credit card fraud
- Game playing<sup>[29]</sup>
- Information retrieval
- Internet fraud detection
- Marketing
- Machine perception
- Medical diagnosis
- Natural language processing<sup>[30]</sup>
- Optimization and metaheuristic
- Online advertising
- Recommender systems
- Robot locomotion
- Search engines



- Sentiment analysis (or opinion mining)
- Sequence mining
- Software engineering
- Speech and handwriting recognition
- Stock market analysis
- Structural health monitoring
- Syntactic pattern recognition
- Economics

In 2006, the online movie company **Netflix** held the first "Netflix Prize" competition to find a program to better predict user preferences and improve the accuracy on its existing Cinematch movie recommendation algorithm by at least 10%. A joint team made up of researchers from **AT&T Labs-Research** in collaboration with the teams **Big Chaos** and **Pragmatic Theory** built an **ensemble model** to win the Grand Prize in 2009 for \$1 million.<sup>[31]</sup> Shortly after the prize was awarded, Netflix realized that viewers' ratings were not the best indicators of their viewing patterns ("everything is a recommendation") and they changed their recommendation engine accordingly.<sup>[32]</sup>

In 2010 The Wall Street Journal wrote about money management firm **Rebellion Research's** use of machine learning to predict economic movements. The article describes **Rebellion Research's** prediction of the financial crisis and economic recovery.<sup>[33]</sup>

In 2014 it has been reported that a machine learning algorithm has been applied in **Art History** to study fine art paintings, and that it may have revealed previously unrecognized influences between artists.<sup>[34]</sup>

## 6 Ethics

Machine Learning poses a host of ethical questions. Systems which are trained on datasets collected with biases may exhibit these biases upon use, thus digitizing cultural prejudices such as institutional racism and classism.<sup>[35]</sup> Responsible collection of data thus is a critical part of machine learning. See **Machine ethics** for additional information.

## 7 Software

**Software suites** containing a variety of machine learning algorithms include the following:

### 7.1 Open-source software

- Caffe

- dlib
- ELKI
- Encog
- GNU Octave
- H2O
- Mahout
- Mallet (software project)
- mlpy
- MLPACK
- MOA (Massive Online Analysis)
- ND4J with Deeplearning4j
- NuPIC
- OpenCV
- OpenNN
- Orange
- R
- scikit-learn
- scikit-image
- Shogun
- TensorFlow
- Torch (machine learning)
- Spark
- Yooreeka
- Weka

### 7.2 Commercial software with open-source editions

- KNIME
- RapidMiner

### 7.3 Commercial software

- Angoss KnowledgeSTUDIO
- Ayasdi
- Databricks
- Google Prediction API
- IBM SPSS Modeler
- KXEN Modeler
- LIONsolver
- Mathematica
- MATLAB
- Microsoft Azure Machine Learning
- Neural Designer
- NeuroSolutions
- Oracle Data Mining
- RCASE
- SAS Enterprise Miner
- STATISTICA Data Miner

- Cache language model
- Cognitive model
- Cognitive science
- Computational intelligence
- Computational neuroscience
- Data science
- Ethics of artificial intelligence
- Existential risk from advanced artificial intelligence
- Explanation-based learning
- Glossary of artificial intelligence
- Important publications in machine learning
- List of machine learning algorithms
- List of datasets for machine learning research
- Spike-and-slab variable selection
- Bayesian Structural Time Series

## 8 Journals

- *Journal of Machine Learning Research*
- *Machine Learning*
- *Neural Computation*

## 9 Conferences

- Conference on Neural Information Processing Systems
- International Conference on Machine Learning
- International Conference on Learning Representations

## 10 See also

- Adaptive control
- Adversarial machine learning
- Automatic reasoning
- Big data

## 11 References

- [1] <http://www.britannica.com/EBchecked/topic/1116194/machine-learning> This tertiary source reuses information from other sources but does not name them.
- [2] Phil Simon (March 18, 2013). *Too Big to Ignore: The Business Case for Big Data*. Wiley. p. 89. ISBN 978-1-118-63817-0.
- [3] Ron Kohavi; Foster Provost (1998). "Glossary of terms". *Machine Learning* **30**: 271–274.
- [4] Machine learning and pattern recognition "can be viewed as two facets of the same field."
- [5] Wernick, Yang, Brankov, Yourganov and Strother, Machine Learning in Medical Imaging, *IEEE Signal Processing Magazine*, vol. 27, no. 4, July 2010, pp. 25-38
- [6] Mannila, Heikki (1996). *Data mining: machine learning, statistics, and databases*. Int'l Conf. Scientific and Statistical Database Management. IEEE Computer Society.
- [7] Friedman, Jerome H. (1998). "Data Mining and Statistics: What's the connection?". *Computing Science and Statistics* **29** (1): 3–9.
- [8] "Machine Learning: What it is and why it matters". *www.sas.com*. Retrieved 2016-03-29.
- [9] Mitchell, T. (1997). *Machine Learning*, McGraw Hill. ISBN 0-07-042807-7, p.2.

- [10] Harnad, Stevan (2008), “The Annotation Game: On Turing (1950) on Computing, Machinery, and Intelligence”, in Epstein, Robert; Peters, Grace, *The Turing Test Sourcebook: Philosophical and Methodological Issues in the Quest for the Thinking Computer*, Kluwer
- [11] Russell, Stuart; Norvig, Peter (2003) [1995]. *Artificial Intelligence: A Modern Approach* (2nd ed.). Prentice Hall. ISBN 978-0137903955.
- [12] Langley, Pat (2011). “The changing science of machine learning”. *Machine Learning* **82** (3): 275–279. doi:10.1007/s10994-011-5242-y.
- [13] Le Roux, Nicolas; Bengio, Yoshua; Fitzgibbon, Andrew (2012). “Improving First and Second-Order Methods by Modeling Uncertainty”. In Sra, Suvrit; Nowozin, Sebastian; Wright, Stephen J. *Optimization for Machine Learning*. MIT Press. p. 404.
- [14] MI Jordan (2014-09-10). “statistics and machine learning”. reddit. Retrieved 2014-10-01.
- [15] Cornell University Library. “Breiman : Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author)”. Retrieved 8 August 2015.
- [16] Gareth James; Daniela Witten; Trevor Hastie; Robert Tibshirani (2013). *An Introduction to Statistical Learning*. Springer. p. vii.
- [17] Bishop, C. M. (2006), *Pattern Recognition and Machine Learning*, Springer, ISBN 0-387-31073-8
- [18] Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar (2012) *Foundations of Machine Learning*, MIT Press ISBN 978-0-262-01825-8.
- [19] Ethem Alpaydin. “Introduction to Machine Learning” The MIT Press, 2010.
- [20] Viebke, Andre; Pillana, Sabri. “The Potential of the Intel Xeon Phi for Supervised Deep Learning” (PDF). *The 17th IEEE International Conference on High Performance Computing and Communications (HPCC 2015)*, Aug. 24 - 26, 2015, New York, USA. IEEE. Retrieved 1 April 2016.
- [21] Honglak Lee, Roger Grosse, Rajesh Ranganath, Andrew Y. Ng. “Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations” Proceedings of the 26th Annual International Conference on Machine Learning, 2009.
- [22] Lu, Haiping; Plataniotis, K.N.; Venetsanopoulos, A.N. (2011). “A Survey of Multilinear Subspace Learning for Tensor Data” (PDF). *Pattern Recognition* **44** (7): 1540–1551. doi:10.1016/j.patcog.2011.01.004.
- [23] Yoshua Bengio (2009). *Learning Deep Architectures for AI*. Now Publishers Inc. pp. 1–3. ISBN 978-1-60198-294-0.
- [24] A. M. Tillmann, “On the Computational Intractability of Exact and Approximate Dictionary Learning”, IEEE Signal Processing Letters 22(1), 2015: 45–49.
- [25] Aharon, M, M Elad, and A Bruckstein. 2006. “K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation.” *Signal Processing, IEEE Transactions on* 54 (11): 4311–4322
- [26] Goldberg, David E.; Holland, John H. (1988). “Genetic algorithms and machine learning”. *Machine Learning* **3** (2): 95–99. doi:10.1007/bf00113892.
- [27] Michie, D.; Spiegelhalter, D. J.; Taylor, C. C. (1994). *Machine Learning, Neural and Statistical Classification*. Ellis Horwood.
- [28] Zhang, Jun; Zhan, Zhi-hui; Lin, Ying; Chen, Ni; Gong, Yue-jiao; Zhong, Jing-hui; Chung, Henry S.H.; Li, Yun; Shi, Yu-hui (2011). “Evolutionary Computation Meets Machine Learning: A Survey” (PDF). *Computational Intelligence Magazine (IEEE)* **6** (4): 68–75. doi:10.1109/mci.2011.942584.
- [29] Tesauro, Gerald (March 1995). “Temporal Difference Learning and TD-Gammon”. *Communications of the ACM* **38** (3): 58–68. doi:10.1145/203330.203343.
- [30] Daniel Jurafsky and James H. Martin (2009). *Speech and Language Processing*. Pearson Education. pp. 207 ff.
- [31] “BelKor Home Page” research.att.com
- [32] “The Netflix Tech Blog: Netflix Recommendations: Beyond the 5 stars (Part 1)”. Retrieved 8 August 2015.
- [33] Scott Patterson (13 July 2010). “'Artificial Intelligence' Gains Fans Among Investors - WSJ”. *WSJ*. Retrieved 8 August 2015.
- [34] When A Machine Learning Algorithm Studied Fine Art Paintings, It Saw Things Art Historians Had Never Noticed, *The Physics at ArXiv blog*
- [35] Bostrom, Nick (2011). “The Ethics of Artificial Intelligence” (PDF). Retrieved 11 April 2016.

## 12 Further reading

- Pedro Domingos (September 2015), *The Master Algorithm*, Basic Books, ISBN 978-0-465-06570-7
- Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar (2012). *Foundations of Machine Learning*, The MIT Press. ISBN 978-0-262-01825-8.
- Ian H. Witten and Eibe Frank (2011). *Data Mining: Practical machine learning tools and techniques* Morgan Kaufmann, 664pp., ISBN 978-0-12-374856-0.
- Sergios Theodoridis, Konstantinos Koutroumbas (2009) “Pattern Recognition”, 4th Edition, Academic Press, ISBN 978-1-59749-272-0.
- Mierswa, Ingo and Wurst, Michael and Klinkenberg, Ralf and Scholz, Martin and Euler, Timm: *YALE: Rapid Prototyping for Complex Data Mining*



- Tasks*, in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06), 2006.
- Bing Liu (2007), *Web Data Mining: Exploring Hyperlinks, Contents and Usage Data*. Springer, ISBN 3-540-37881-2
  - Toby Segaran (2007), *Programming Collective Intelligence*, O'Reilly, ISBN 0-596-52932-5
  - Huang T.-M., Kecman V., Kopriva I. (2006), *Kernel Based Algorithms for Mining Huge Data Sets, Supervised, Semi-supervised, and Unsupervised Learning*, Springer-Verlag, Berlin, Heidelberg, 260 pp. 96 illus., Hardcover, ISBN 3-540-31681-7.
  - Ethem Alpaydm (2004) *Introduction to Machine Learning (Adaptive Computation and Machine Learning)*, MIT Press, ISBN 0-262-01211-1
  - MacKay, David J. C.. *Information Theory, Inference, and Learning Algorithms* Cambridge: Cambridge University Press, 2003. ISBN 0-521-64298-1
  - KECMAN Vojislav (2001), *Learning and Soft Computing, Support Vector Machines, Neural Networks and Fuzzy Logic Models*, The MIT Press, Cambridge, MA, 608 pp., 268 illus., ISBN 0-262-11255-8.
  - Trevor Hastie, Robert Tibshirani and Jerome Friedman (2001). *The Elements of Statistical Learning*, Springer. ISBN 0-387-95284-5.
  - Richard O. Duda, Peter E. Hart, David G. Stork (2001) *Pattern classification* (2nd edition), Wiley, New York, ISBN 0-471-05669-3.
  - Bishop, C.M. (1995). *Neural Networks for Pattern Recognition*, Oxford University Press. ISBN 0-19-853864-2.
  - Ryszard S. Michalski, George Tecuci (1994), *Machine Learning: A Multistrategy Approach*, Volume IV, Morgan Kaufmann, ISBN 1-55860-251-8.
  - Sholom Weiss and Casimir Kulikowski (1991). *Computer Systems That Learn*, Morgan Kaufmann. ISBN 1-55860-065-5.
  - Yves Kodratoff, Ryszard S. Michalski (1990), *Machine Learning: An Artificial Intelligence Approach, Volume III*, Morgan Kaufmann, ISBN 1-55860-119-8.
  - Ryszard S. Michalski, Jaime G. Carbonell, Tom M. Mitchell (1986), *Machine Learning: An Artificial Intelligence Approach, Volume II*, Morgan Kaufmann, ISBN 0-934613-00-1.
  - Ryszard S. Michalski, Jaime G. Carbonell, Tom M. Mitchell (1983), *Machine Learning: An Artificial Intelligence Approach*, Tioga Publishing Company, ISBN 0-935382-05-4.
  - Vladimir Vapnik (1998). *Statistical Learning Theory*. Wiley-Interscience, ISBN 0-471-03003-1.
  - Ray Solomonoff, *An Inductive Inference Machine*, IRE Convention Record, Section on Information Theory, Part 2, pp., 56-62, 1957.
  - Ray Solomonoff, "An Inductive Inference Machine" A privately circulated report from the 1956 Dartmouth Summer Research Conference on AI.

## 13 External links

- International Machine Learning Society
- Popular online course by Andrew Ng, at Coursera. It uses GNU Octave. The course is a free version of Stanford University's actual course taught by Ng, whose lectures are also available for free.
- mloss is an academic database of open-source machine learning software.

## 14 Text and image sources, contributors, and licenses

### 14.1 Text

- **Machine learning** *Source:* [https://en.wikipedia.org/wiki/Machine\\_learning?oldid=723304919](https://en.wikipedia.org/wiki/Machine_learning?oldid=723304919) *Contributors:* Arvindn, ChangChienFu, Michael Hardy, Kku, Delirium, Ahoerstemeier, Ronz, BenKovitz, Mxn, Hike395, Silvonen, Furrykef, Buridan, Jmartinezot, Phoebe, Shizhao, Topbanana, Robbot, Plehn, KellyCoinGuy, Ancheta Wis, Fabiform, Centrx, Giftlite, Seabhcán, Levin, Dratman, Jason Quinn, Khalid hassani, Utcursch, APH, Gene s, Paulscrawl, Clemwang, Nowozin, Silence, Bender235, ZeroOne, Superbacana, Aaronbrick, Jojit fb, Nk, Rajah, Tritium6, Haham hanuka, Mdd, HasharBot~enwiki, Vilapi, Arcenciel, Denoir, Wjbean, Stephen Turner, Learn-More, Rrenaud, Leondz, Soultaco, Ruud Koot, BlaiseFEgan, JimmyShelter~enwiki, Essay, Joerg Kurt Wegner, Adiel, BD2412, Qwertyus, Rjwilmsi, Emrysk, VKokielov, Eubot, Celendin, Intgr, Predictor, Kri, BMF81, Chobot, Bobdc, Adoniscik, YurikBot, Misterwindupbird, Trondtr, Nesbit, Grafen, Gareth Jones, Srinivasasha, Crasshopper, DaveWF, Masatran, CWenger, Fram, KnightRider~enwiki, SmackBot, Mneser, InverseHypercube, CommodiCast, Jyoshimi, MclD, KYN, Ohnoitsjamie, Chris the speller, FidesLT, Nbarth, Cfallin, Moorejh, JonHarder, Baguasquirrel, Krexer, Shadow1, Philpraxis~enwiki, Daniel.Cardenas, Sina2, ChaoticLogic, NongBot~enwiki, Rex-Survey, Beetstra, WMod-NS, Julthep, Dsilver~enwiki, Dicklyon, Vsweiner, Optakeover, Ctacmo, MTSbot~enwiki, Ralf Klinkenberg, Dave Runger, Doceddi, Scigrex14, Pgr94, Innohead, Bumbulski, PeterdJones, Dancter, Msnicki, Quintopia, Thijs!bot, Mereda, Perrygogas, Djbwiki, GordonRoss, Kinimod~enwiki, Damienfrancois, Natalie Erin, Seaphoto, AnAj, Ninjakannon, Kimptoc, Penguinbroker, The Transhumanist, Jrennie, Hut 8.5, Kyhui, Magioladitis, Ryszard Michalski, Jwojt, Transcendence, Tedickey, Pebkac, Robotman1974, Jroudh, Businessman332211, Pmbhagat, Caltech, STBot, Keith D, Glrx, Nickvence, Gem-fanat, Salih, AntiSpamBot, Gombang, Chri-blo, Mxwsn, Dana2020, DavidCBryant, Bonadea, WinterSpw, RJASE1, Funandtrvl, James Kidd, LokiClock, Redgecko, Markcsq, Jrlrl, Like.liberation, A4bot, Daniel347x, Joel181, Wikidemon, Lordvolton, Defza, Chrisoneall, Spiral5800, Kesshaka, Cvdwalt, Why Not A Duck, Sebastjanmm, LittleBenW, Gal chechik, Biochaos, Cmbishop, Jbmurray, IradBG, Smsarmad, Scorpion451, Kumioko (renamed), CharlesGillingham, StaticGull, CultureDrone, Anchor Link Bot, ImageRemovalBot, ClueBot, GorillaWarfare, Ahyeek, Sonu mangla, Ggia, Debejyo, D.scain.farenzena, He7d3r, Magdon~enwiki, WilliamSewell, Jim15936, Vanished user uih38riiw4hjlsd, Evansad, Roxy the dog, PseudoOne, André P Ricardo, Agamemnonc, Darnelr, MystBot, Dsimic, YrPolishUncle, MTJM, Addbot, Mortense, Fyrael, Aceituno, MrOllie, LaaknorBot, Jarble, Movado73, Luckas-bot, QuickUkie, Yobot, NotARusski, Genius002, Examtester, AnomieBOT, Piano non troppo, MaterialsScientist, Clicky, Devantheryv, Vivohobson, ArthurBot, Quebec99, Xqbot, Happyrabbit, Gtfjbl, Kithira, J04n, Addingrefs, Webidiap, Shirik, JoeHms22, Aaron Kauppi, Velblod, Prari, FrescoBot, Jdizzle123, Featherard, WhatWasDone, Siculars, Prof-fvikt, Boxplot, SwordsmanKirby, I dream of horses, Wikinacious, Skyerise, Mostafa mahdieh, Lars Washington, TobeBot, OnceAlpha, AXRL, Иъ Лю Ха, BertSeghers, Edouard.darchimbaud, Winnerdy, Zosoin, Helwr, EmausBot, Johncasey, Dzkd, Primefac, MartinThoma, MarsTrombone, Wht43, Chire, GZ-Bot, Jcautilli, Jorjulio, AManWithNoPlan, Pintaio, L Kensington, Ataulf, Zfeinst, Yoshua.Bengio, Casia wyq, Ego White Tray, Blaz.zupan, Shinosin, Marius.andreiana, Lovok Sovok, Graytay, Liuyipei, ClueBot NG, Tillander, Keefaas, Lawrence87, Aiwing, Pranji973, Candace Gillhoolley, Robiminer, Leonardo61, Wrdieter, Arrandale, O.Kosowski, WikiMSL, Helpful Pixie Bot, RobertPollak, BG19bot, Smorsy, Mohamed CJ, Lisasolomonsalford, Anubhab91, Chafe66, Solomon7968, Ishq2011, Autologin, Brooksrichardbrown, DasAllFolks, Billhodak, Debora.riu, Ohandyya, Davidmetcalfe, David.moreno72, Mdann52, JoshuSasori, Ulugen, IjonTichyIjonTichy, Keshav.dhandhan, Dexbot, Mogism, Djfrost711, Bkuhlman80, Frosty, James12345, Shubhi choudhary, Jochen Burghardt, Joeinwiki, BrettMurphy, Phamnhatkhanh, Ppilote, Delafé, InnocuousPilcrow, Kittensareawesome, Statpumpkin, Neo Poz, Dustin V. S., TJLaHer123, Ankit.ufl, Francisbach, Aleks-ger, MarinMersenne, Weiping.thu, LokeshRavindranathan, Tonyszedlak, Proneat123, GrowthRate, Sami Abu-El-Haija, Mpgoldhirsh, Work Shop Corpse, Superploro, Dawolakamp, Waggie, Justincagoon, Jorge Guerra Pires, Hm1235, Velvel2, Vidhul sikka, Erik Itter, Annaelison, Tgriffin9, Chazdywaters, Rmashrmash, Komselvam, Robbyblue-dogs, HelpUsStopSpam, EricVSiegel, KenTancwell, Justinqnabel, Rusky.ai, Datapablo, Aetille, JenniferTheEmpress0, Dsysko, Lr0^k, BNoack, NightOwl15, Latosh Boris, Thejavis86, Muratovst, Pinsi281, ArguMentor, Datakeeper, Doctasarge, Espyromi, Kailey 2001, WunderStahl, Natenatenatenate, Fmadd, Vladiatorr, Ayasdi and Anonymous: 410

### 14.2 Images

- **File:Animation2.gif** *Source:* <https://upload.wikimedia.org/wikipedia/commons/c/c0/Animation2.gif> *License:* CC-BY-SA-3.0 *Contributors:* Own work *Original artist:* MG (talk · contribs)
- **File:Portal-puzzle.svg** *Source:* <https://upload.wikimedia.org/wikipedia/en/f/fd/Portal-puzzle.svg> *License:* Public domain *Contributors:* ? *Original artist:* ?
- **File:Svm\_max\_sep\_hyperplane\_with\_margin.png** *Source:* [https://upload.wikimedia.org/wikipedia/commons/2/2a/Svm\\_max\\_sep\\_hyperplane\\_with\\_margin.png](https://upload.wikimedia.org/wikipedia/commons/2/2a/Svm_max_sep_hyperplane_with_margin.png) *License:* Public domain *Contributors:* Own work *Original artist:* Cyc

### 14.3 Content license

- Creative Commons Attribution-Share Alike 3.0