# Machine Learning: An In-Depth Guide

**Subscribe** to the *AI with Alex* YouTube channel, home of *AI News*, to stay current on all things AI!

## Articles in This Series

1. Overview, goals, learning types, and algorithms
2. Data selection, preparation, and modeling
3. Model evaluation, validation, complexity, and improvement
4. Model performance and error analysis
5. Unsupervised learning, related fields, and machine learning in practice

## Introduction

Welcome! This is the first article of a five-part series about machine learning.

Machine learning is a very hot topic for many key reasons, and because it provides the ability to automatically obtain deep insights, recognize unknown patterns, and create high performing predictive models from data, all without requiring explicit programming instructions.

Despite the popularity of the subject, machine learning's true purpose and details are not well understood, except by very technical folks and/or data scientists.

This series is intended to be a comprehensive, in-depth guide to machine learning, and should be useful to everyone from business executives to

machine learning practitioners. It covers virtually all aspects of machine learning (and many related fields) at a high level, and should serve as a sufficient introduction or reference to the terminology, concepts, tools, considerations, and techniques of the field.

This high level understanding is critical if ever involved in a decision-making process surrounding the usage of machine learning, how it can help achieve business and project goals, which machine learning techniques to use, potential pitfalls, and how to interpret the results.

Note that most of the topics discussed in this series are also directly applicable to fields such as predictive analytics, data mining, statistical learning, artificial intelligence, and so on.

## Machine Learning Defined

The oft quoted and widely accepted formal definition of machine learning as stated by field pioneer Tom M. Mitchell is:

> 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

The following is my less formal way to describe machine learning.

Machine learning is a subfield of computer science, but is often also referred to as predictive analytics, or predictive modeling. Its goal and usage is to build new and/or leverage existing algorithms to learn from data, in order to build generalizable models that give accurate predictions, or to find patterns, particularly with new and unseen similar data.

## Machine Learning Process Overview

Imagine a dataset as a table, where the rows are each observation (aka measurement, data point, etc), and the columns for each observation represent the features of that observation and their values.

At the outset of a machine learning project, a dataset is usually split into two or three subsets. The minimum subsets are the training and test datasets, and often an optional third validation dataset is created as well.

Once these data subsets are created from the primary dataset, a predictive model or classifier is trained using the training data, and then the model's predictive accuracy is determined using the test data.

As mentioned, machine learning leverages algorithms to automatically model and find patterns in data, usually with the goal of predicting some target output or response. These algorithms are heavily based on statistics and mathematical optimization.

Optimization is the process of finding the smallest or largest value (minima or maxima) of a function, often referred to as a loss, or cost function in the minimization case. One of the most popular optimization algorithms used in machine learning is called gradient descent, and another is known as the the normal equation.

In a nutshell, machine learning is all about automatically learning a highly accurate predictive or classifier model, or finding unknown patterns in data, by leveraging learning algorithms and optimization techniques.

## Types of Learning

The primary categories of machine learning are supervised, unsupervised, and semi-supervised learning. We will focus on the first two in this article.

In supervised learning, the data contains the response variable (label) being modeled, and with the goal being that you would like to predict the value or class of the unseen data. Unsupervised learning involves learning from a dataset that has no label or response variable, and is therefore more about finding patterns than prediction.

As i'm a huge NFL and Chicago Bears fan, my team will help exemplify these types of learning! Suppose you have a ton of Chicago Bears data and stats dating from when the team became a chartered member of the NFL (1920) until the present (2016).

Imagine that each row of the data is essentially a team snapshot (or observation) of relevant statistics for every game since 1920. The columns in this case, and the data contained in each, represent the features (values) of the data, and may include feature data such as game date, game opponent, season wins, season losses, season ending divisional position, post-season berth (Y/N), post-season stats, and perhaps stats specific to the three phases of the game: offense, defense, and special teams.

In the supervised case, your goal may be to use this data to predict if the Bears will win or lose against a certain team during a given game, and at a given field (home or away). Keep in mind that anything can happen in football in terms of pre and game-time injuries, weather conditions, bad referee calls, and so on, so take this simply as an example of an application of supervised learning with a yes or no response (prediction), as opposed to determining the probability or likelihood of 'Da Bears' getting the win.

Since you have historic data of wins and losses (the response) against certain teams at certain football fields, you can leverage supervised learning to create a model to make that prediction.

Now suppose that your goal is to find patterns in the historic data and learn

something that you don't already know, or group the team in certain ways throughout history. To do so, you run an unsupervised machine learning algorithm that clusters (groups) the data automatically, and then analyze the clustering results.

With a bit of analysis, one may find that these automatically generated clusters seemingly groups the team into the following example categories over time:

- Strong defense, weak running offense, strong passing offense, weak special teams, playoff berth
- Strong defense, strong running offense, weak passing offense, average special teams, playoff berth
- Weak defense, strong all-around offense, strong special teams, missed the playoffs
- and so on

An example of unsupervised cluster analysis would be to find a potential reason why they missed the playoffs in the third cluster above. Perhaps due to the weak defense? Bears have traditionally been a strong defensive team, and some say that defense wins championships. Just saying...

In either case, each of the above classifications may be found to relate to a certain time frame, which one would expect. Perhaps the team was characterized by one of these groupings more than once throughout their history, and for differing periods of time.

To characterize the team in this way without machine learning techniques, one would have to pour through all historic data and stats, manually find the patterns and assign the classifications (clusters) for every year taking all

data into account, and compile the information. That would definitely not be a quick and easy task.

Alternatively, you could write an explicitly coded program to pour through the data, and that has to know what team stats to consider, what thresholds to take into account for each stat, and so forth. It would take a substantial amount of time to write the code, and different programs would need to be written for every problem needing an answer.

Or... you can employ a machine learning algorithm to do all of this automatically for you in a few seconds.

## Machine Learning Goals and Outputs

Machine learning algorithms are used primarily for the following types of output:

- Clustering (Unsupervised)
- Two-class and multi-class classification (Supervised)
- Regression: Univariate, Multivariate, etc. (Supervised)
- Anomaly detection (Unsupervised and Supervised)
- Recommendation systems (aka recommendation engine)

Specific algorithms that are used for each output type are discussed in the next section, but first, let's give a general overview of each of the above output, or problem types.

As discussed, clustering is an unsupervised technique for discovering the composition and structure of a given set of data. It is a process of clumping

data into clusters to see what groupings emerge, if any. Each cluster is characterized by a contained set of data points, and a cluster centroid. The cluster centroid is basically the mean (average) of all of the data points that the cluster contains, across all features.

Classification problems involve placing a data point (aka observation) into a pre-defined class or category. Sometimes classification problems simply assign a class to an observation, and in other cases the goal is to estimate the probabilities that an observation belongs to each of the given classes.

A great example of a two-class classification is assigning the class of Spam or Ham to an incoming email, where ham just means 'not spam'. Multi-class classification just means more than two possible classes. So in the spam example, perhaps a third class would be 'Unknown'.

Regression is just a fancy word for saying that a model will assign a continuous value (response) to a data observation, as opposed to a discrete class. A great example of this would be predicting the closing price of the Dow Jones Industrial Average on any given day. This value could be any number, and would therefore be a perfect candidate for regression.

Note that sometimes the word regression is used in the name of an algorithm that is actually used for classification problems, or to predict a discrete categorical response (e.g., spam or ham). A good example is logistic regression, which predicts probabilities of a given discrete value.

Another problem type is anomaly detection. While we'd love to think that data is well behaved and sensible, unfortunately this is often not the case. Sometimes there are erroneous data points due to malfunctions or errors in measurement, or sometimes due to fraud. Other times it could be that anomalous measurements are indicative of a failing piece of hardware or electronics.

Sometimes anomalies are indicative of a real problem and are not easily explained, such as a manufacturing defect, and in this case, detecting anomalies provides a measure of quality control, as well as insight into whether steps taken to reduce defects have worked or not. In either case, there are times where it is beneficial to find these anomalous values, and certain machine learning algorithms can be used to do just that.

The final type of problem is addressed with a recommendation system, or also called recommendation engine. Recommendation systems are a type of information filtering system, and are intended to make recommendations in many applications, including movies, music, books, restaurants, articles, products, and so on. The two most common approaches are content-based and collaborative filtering.

Two great examples of popular recommendation engines are those offered by Netflix and Amazon. Netflix makes recommendations in order to keep viewers engaged and supplied with plenty of content to watch. In other words, to keep people using Netflix. They do this with their “Because you watched ...“, “Top Picks for Alex”, and “Suggestions for you” recommendations.

Amazon does a similar thing in order to increase sales through up-selling, maintain sales through user engagement, and so on. They do this through their “Customers Who Bought This Item Also Bought”, “Recommendations for You, Alex”, “Related to Items You Viewed”, and “More Items to Consider” recommendations.

## Machine Learning Algorithms

We’ve now covered the machine learning problem types and desired outputs. Now we will give a high level overview of relevant machine learning

algorithms.

Here is a list of algorithms, both supervised and unsupervised, that are very popular and worth knowing about at a high level. Note that some of these algorithms will be discussed in greater depth later in this series.

**Supervised Regression**

- Simple and multiple linear regression
- Decision tree or forest regression
- Artificial Neural networks
- Ordinal regression
- Poisson regression
- Nearest neighbor methods (e.g., k-NN or k-Nearest Neighbors)

**Supervised Two-class & Multi-class Classification**

- Logistic regression and multinomial regression
- Artificial Neural networks
- Decision tree, forest, and jungles
- SVM (support vector machine)
- Perceptron methods
- Bayesian classifiers (e.g., Naive Bayes)
- Nearest neighbor methods (e.g., k-NN or k-Nearest Neighbors)

- One versus all multiclass

### Unsupervised

- K-means clustering
- Hierarchical clustering

### Anomaly Detection

- Support vector machine (one class)
- PCA (Principle component analysis)

Note that a technique that's often used to improve model performance is to combine the results of multiple models. This approach leverages what's known as ensemble methods, and random forests are a great example (discussed later).

If nothing else, it's a good idea to at least familiarize yourself with the names of these popular algorithms, and have a basic idea as to the type of machine learning problem and output that they may be well suited for.

## Summary

Machine learning, predictive analytics, and other related topics are very exciting and powerful fields.

While these topics can be very technical, many of the concepts involved are relatively simple to understand at a high level. In many cases, a simple understanding is all that's required to have discussions based on machine learning problems, projects, techniques, and so on.

Part two of this series will provide an introduction to model performance,

cover the machine learning process, and discuss model selection and associated tradeoffs in detail.

Stay tuned!

## About the Author

Alex is the founder of InnoArchiTech and InnoArchiTech Institute, as well as the author of *AI for People and Business* published by O'Reilly Media.