Fivetran

Machine learning demystified

Machine learning refers to methods that allow machines to "learn" specific patterns and tasks without explicit programming by combining algorithms with data. It is the key capability behind predictive analytics.

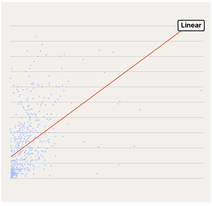
Generative AI is sometimes overkill. Your use case might benefit from these, instead:

Supervised

Supervised learning is used when there is an existing set of known inputs and outputs. The goal is to construct a model that maps between inputs and outputs in order to predict future outputs.

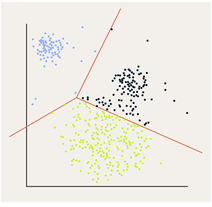
Practical use cases include anything that requires predicting a numerical trend or classifying records (e.g. image recognition). In general, predictive modeling is supervised.

A basic example is linear regression, which we will walk through shortly.



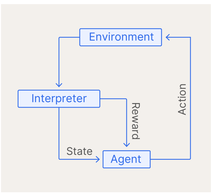
Unsupervised

Unsupervised learning does not depend on known inputs and outputs but instead uncovers and identifies patterns. It is useful when exploring data to group data points together (cluster analysis) or to identify factors that are highly correlated with each other and can be consolidated (principal components analysis).



Reinforcement learning

Reinforcement learning similarly does not depend on known inputs and outputs but instead incentivizes artificial agents to engage in reward-maximizing behavior. This tends to be used for problems that require automatic, on-the-fly decision-making, such as self-driving vehicles, gaming bots, and elevator scheduling.



An example of supervised machine learning at a glance

Consider linear regression, which you may have encountered on an Excel spreadsheet as the "trendline" function. Here we have an example of ordinary least squares. For the sake of simplicity, the example only has one independent (input) and one dependent (output) variable.

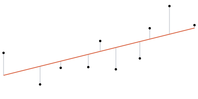
A supervised machine learning model consists of the following parts:

1. Data, consisting of known inputs and outputs to be used as a training set. Let's suppose the dummy data to the right corresponds with the scatterplot on the far right.

2. An algorithm, or problem-solving procedure meant to identify estimators by solving an objective function.

3. A model function that is populated with estimators by executing the algorithm using the data.

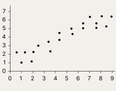
The overall goal is to draw a line through the scatterplot such that the sum of squared* errors (distance between the line and each point) is minimized.



Data

X	Y
X_1	Y_1
X_2	Y_2
X_3	Y_3
X_4	Y_4
X_5	Y_5
X_6	Y_6
X_7	Y_7

Scatterplot



Model function

$y = a + Bx$

Parameters (real, unobserved values): y-intercept a, slope B

Use algorithm to find estimators $\hat{a}$ and $\hat{B}$; these values will be used to populate your model.

Objective function

$$\min_{a,\,B} \sum_{i=1}^{n} (y_i - a - Bx_i)^2$$

*In statistics, square functions are used to describe variance instead of absolute value functions because the functions are continuous and differentiable - a topic beyond the scope of this infographic!

You can read the objective function as "For each data point i from 1 to n (note that the data set has n records), find the values $\hat{a}$ and $\hat{B}$ that minimize the total squared difference between all the observed data points and the value of the line for the corresponding x value." You can then plug these $\hat{a}$ and $\hat{B}$ values back into your model function and plot a line.

A human would use the symbolic manipulation taught in calculus to solve this function, but a computer is more likely to use some form of numerical analysis to approximate the result within specified error bounds. The computations performed by the computer to determine the estimators make up the actual algorithm part of this model. These days, you seldom need to handwrite algorithms; there are many out-of-the box packages available through libraries like scikit-learn for Python that only require a user to provide the data and specify the variables.

This approach can be generalized to multiple linear regression, which is essentially the same thing but with multiple explanatory variables. If you have two explanatory variables, instead of a one-dimensional list of x values, your data set will include an n x 2 matrix consisting of n observations and 2 explanatory variables (see below). The objective function would try to fit a line to a 3D scatter plot.

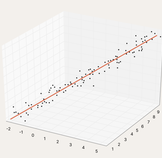
Data

$$X = \begin{bmatrix} X_{1a} & X_{1b} \\ X_{2a} & X_{2b} \\ X_{3a} & X_{3b} \\ X_{4a} & X_{4b} \\ X_{5a} & X_{5b} \\ X_{6a} & X_{6b} \\ X_{7a} & X_{7b} \end{bmatrix} \quad Y = \begin{bmatrix} Y_{1a} \\ Y_{2a} \\ Y_{3a} \\ Y_{4a} \\ Y_{5a} \\ Y_{6a} \\ Y_{7a} \end{bmatrix}$$

Model function

$Y = a + B1X1 + B2X2$

Scatterplot and trendline



You will simply keep adding dimensions for any additional number of explanatory variables! (Obviously, you can't really visualize a scatterplot in more than three dimensions).

Linear regression is the most straightforward of many machine learning techniques that fundamentally rest on statistics and linear algebra. There are many topics we have elided in this infographic, such as the exact symbolic and numeric manipulations required to solve objective functions, splitting data into test and validation sets to improve the quality of a model, the uses of semi-supervised learning, and the various elements of the full data science workflow.

Hopefully, this infographic has taken some of the mystery out of machine learning! Machine learning is a rich topic and older than you think. Much of the math was first discovered many decades (or even centuries) ago, but only recently have there been machines capable of executing it at scale.