Supervised vs unsupervised machine learning approaches

Originally published at https://www.itransition.com/blog/supervised-vs-unsupervised-learning
Published: June 4, 2021 | By Martin Anderson Independent AI expert

When an organization decides to exploit existing data through machine learning solutions, there are often some unpleasant shocks in store as to how unsuitable the data might be for this purpose.

For instance, long-term historical data archives are likely to be large, variegated and unstructured, and lacking in the systematic labels that machine analytics systems will need in order to iterate meaningfully through the data and generate useful insights.

These labels can either be created manually, which can be prohibitively expensive and time-consuming, or systematically, with unsupervised learning systems that can help to explore and discover data streams and relationships that can then be labeled for use in a supervised learning system.

However, this places unsupervised learning in the context of a burdensome 'extra' step for organizations that did not have the foresight to establish data-labeling systems in the first place. Rather, unsupervised learning can be a valuable tool in its own right to gain insights that supervised learning will entirely miss.

In this article, we'll take a look at both approaches, and we'll see that the choice of supervised vs unsupervised machine learning solutions may be decided by more than just the state of the data.

Supervised learning use cases

Supervised learning makes use of datasets that already contain labels or classifications—metadata that gives useful context to the information in the database.

Sometimes data-generating systems create such labels automatically. For instance, digital photos may contain a wide range of embedded metadata about the equipment used, the date and time the photo was taken, the GPS coordinates of the location, and various other useful snippets that can be utilized in the training of a machine learning system.

In most cases, however, the labeling is either achieved manually (where actual humans make judgements on the data and annotate it accordingly) or through algorithmic means (for instance, via optical character recognition for scans of printed documents)—methods which, respectively, may either be time-consuming and expensive, or quicker but less accurate.

Classification tasks in supervised learning

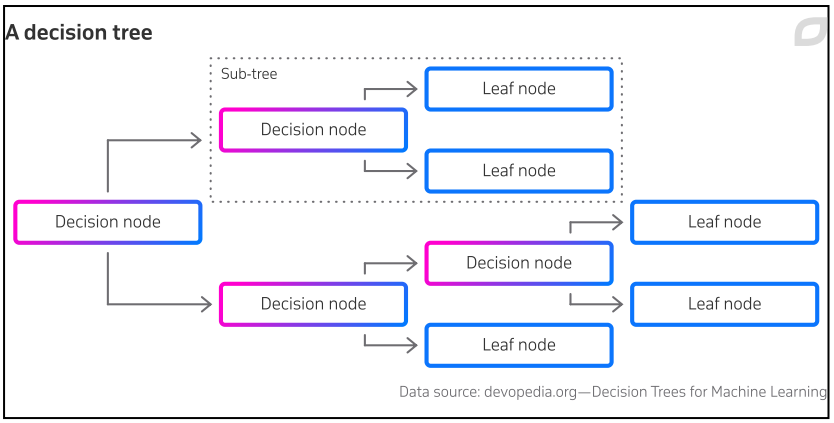
The two most common uses for supervised learning are **classification** and **regression**.

Classification tasks involve distinguishing and labeling different types of content based on historical data. This is a suitable approach for object recognition and facial recognition tasks, as well as industrial monitoring and weather forecasting—all cases where there are relatively few 'surprises' in the data and it is possible to hone a reliable labeling algorithm.

Popular models for classification in supervised learning include:

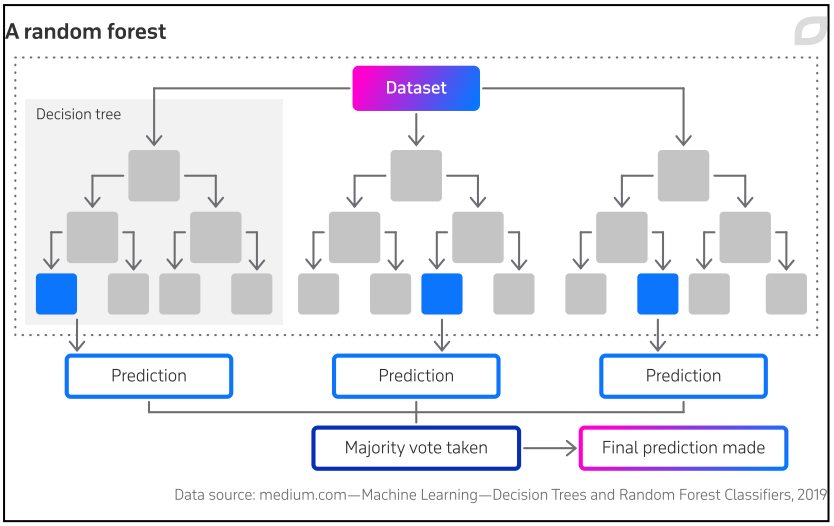
Decision trees

A decision tree is an automated method for mapping the branches of possible outcomes from an initial starting point. The calculations result in a graph that's easy to understand and explain, but which requires a level of human-generated insight and interpretation at each node of the branch.



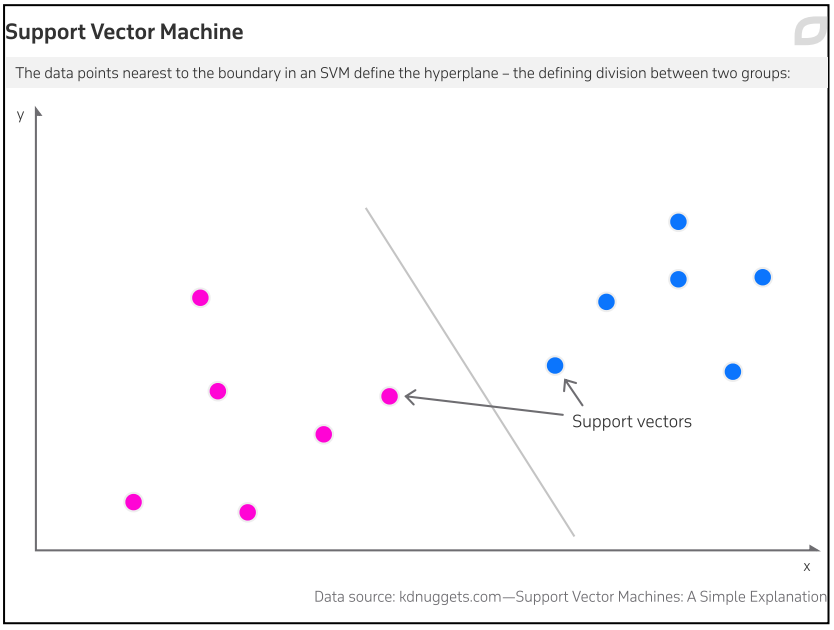
Random forest

A random forest approach throws multiple decision trees at a problem and averages the likelihood of certain types of outcome. Though this method is less easy to visualize than a decision tree, it is better at avoiding the problem of 'overfitting'—cases where the model and the data can become so tailored to each other that the model's workflow will not function well on other datasets.



Support Vector Machine (SVM)

A Support Vector Machine approach defines a boundary between groups of data, known as a *hyperplane*.



Where the number of data instances to classify is just two or three, an SVM produces a two-dimensional graph. If the model's features are more numerous than that, the boundary (and the graph) becomes three-dimensional.

The 'support vectors' themselves are named thus because they are placed so centrally in the graph and so near to the borders of each group, that they actually define these boundaries, and therefore characterize the points of greatest correlation between two different types of identified data. They are also the most difficult data points to identify.

SVMs are useful for high-dimensional data and suited for variegated and diverse datasets in sectors such as handwriting analysis, image classification, bioinformatics, text categorization in natural language processing, and protein folding, among others.

Naive Bayes classifiers

A naive Bayes classifier is an efficient and highly scalable routine for classification based on Bayes' theorem, a simple but powerful method of calculating probabilities from historical data. A Bayes classifier is an adroit and economical solution for reliably-labeled datasets in a supervised learning pipeline, and may be the first approach to consider in the development of a supervised architecture.

Neural networks

A neural network can iterate through very high volumes of data in order to discern relationships and classify the data successfully. However, this approach is time-consuming, often costly, and can require a great deal of experimentation to produce workflows that generalize well on different datasets. With those caveats, a neural network is a powerful and full-featured resource for supervised learning.

Regression tasks in supervised learning

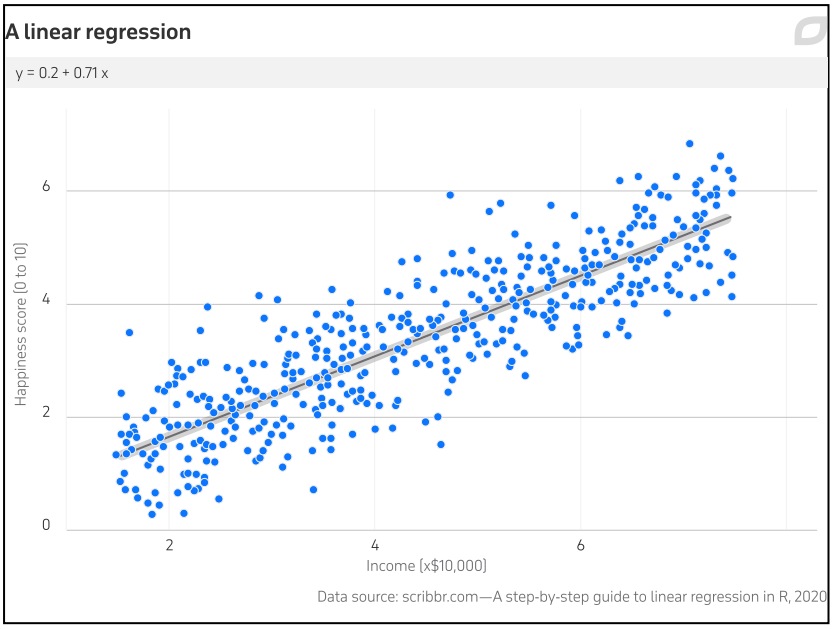
Regression in supervised learning is occupied with predicting continuing outcomes from an ongoing stream of data. It is well-suited to predictive analytics in finance such as stock market predictions, and to business analytics models that operate on similar principles.

There are several commonly used regression models:

Linear regression

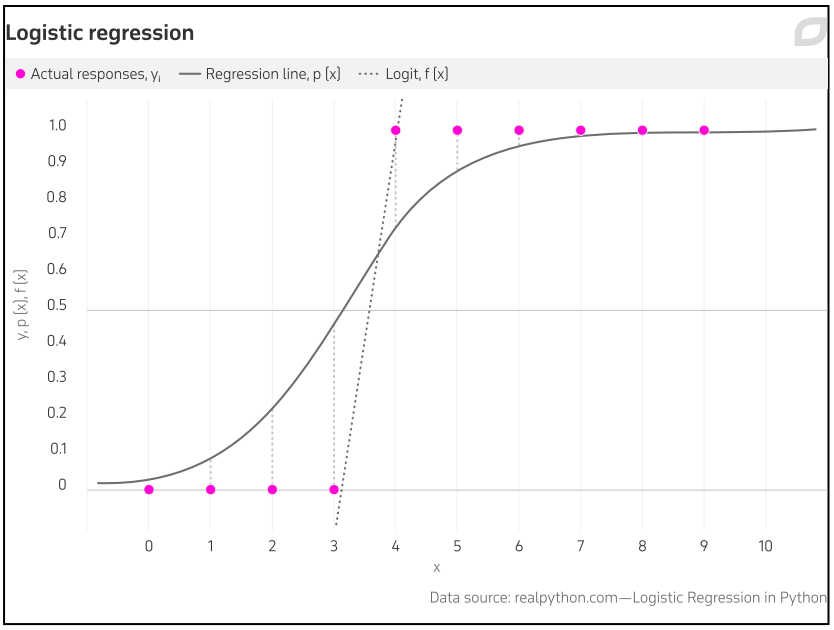
Linear regression maps the relationship between two variables in order to develop an algorithm that defines a relationship between one variable and a desired or predicted outcome.

In practical terms, the first variable might be the weight of a person and the second their height, with the objective being to establish whether there is a governing relationship between these two factors, and whether that relationship can be expressed algorithmically.



Logistic regression

Logistic regression maps a relationship between an independent and a dependent variable. It's useful in cases where the seed variable is binary (i.e. it might be one thing or another), and consequently generates a more complicated type of graph, called a sigmoid.



Logistic regression is well-suited for predictive modeling in healthcare, as well as systems to predict customer churn and recommender systems.

Advantages of supervised learning

Long-term reporting consistency

One major advantage of defining labels, classes and criteria in advance is that you'll be able to rely on the reporting consistency of the model. An ideal use case of this nature is a predictive analytics model with strictly defined features such as 'market forecasts', 'raw material prices', 'time of year' and 'sales'.

While an unsupervised model may unearth additional insights, its reports are more likely to be structurally inconsistent, and therefore unsuitable for year-on-year reporting.

Superior explainability and accountability

Since a supervised learning model runs on a fairly narrow set of pre-defined tracks, it's easier to understand how it arrives at its conclusions than is the case with an unsupervised model. This reduces the chance of the model generating unintentional biases, makes its output more explainable, and lowers a company's legal exposure in terms of the growing regulatory concern around 'black-box' AI.

Decision boundaries can be re-used

In the course of evaluating the data, a supervised training model will produce a number of decision boundaries which can then be applied against subsequent data as a straightforward mathematical formula for classification, improving the efficiency of the data pipeline.

Disadvantages of supervised learning

The need to handle under-represented data

When training data includes an under-represented data facet (such as a product that is only 3% of your inventory lists), it can be difficult for a supervised system to represent it adequately, since its low incidence count could push that data to the wrong side of a decision boundary.

This can be remedied by adjusting the model's weights so that it takes correct account of the under-represented data. However, once the weights have been hand-crafted in this way, it will be necessary to check that later fluctuations in the frequency of that data do not cause other types of imbalance in reporting. Since unsupervised systems have fewer pre-defined expectations, they are more likely to handle such anomalies in a non-destructive way.

Difficulty in discovering new or emerging relationships in data

One disadvantage of presuming that you have discovered all the relevant relationships and features for your model is that there might be some very interesting business data in the 'outliers' that get discarded in a rigid model with pre-assigned variables—and some of these signifiers may not have existed when the model was first designed.

Preprocessing requirements

A training network based on supervised learning will continue to depend on consistent labeling and classification. It will not usually be able to correct anomalies in this respect, and may well simply reject such data, irrespective of its value. This makes the automated or human-centered labeling systems that pre-process the data a mission-critical aspect of production.

Unsupervised learning use cases

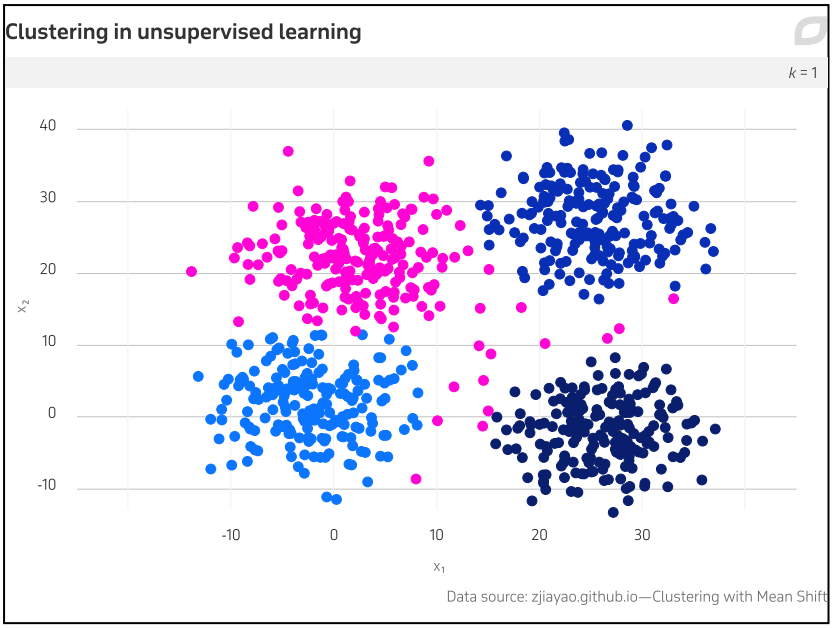
Unsupervised learning analyzes datasets that don't have any labels or any secondary information (such as metadata) that could be used as labels. Instead, an unsupervised machine learning system looks across the breadth of unprocessed data for recurrent patterns and attempts to perform classification and labeling tasks without human supervision.

Broadly speaking, unsupervised learning can aid data discovery and massively cut down on expensive manual pre-processing of data. It can also serve as an exploratory stage to discover data relationships and features that will later be utilized in the more reductive and linear workflow of a supervised system.

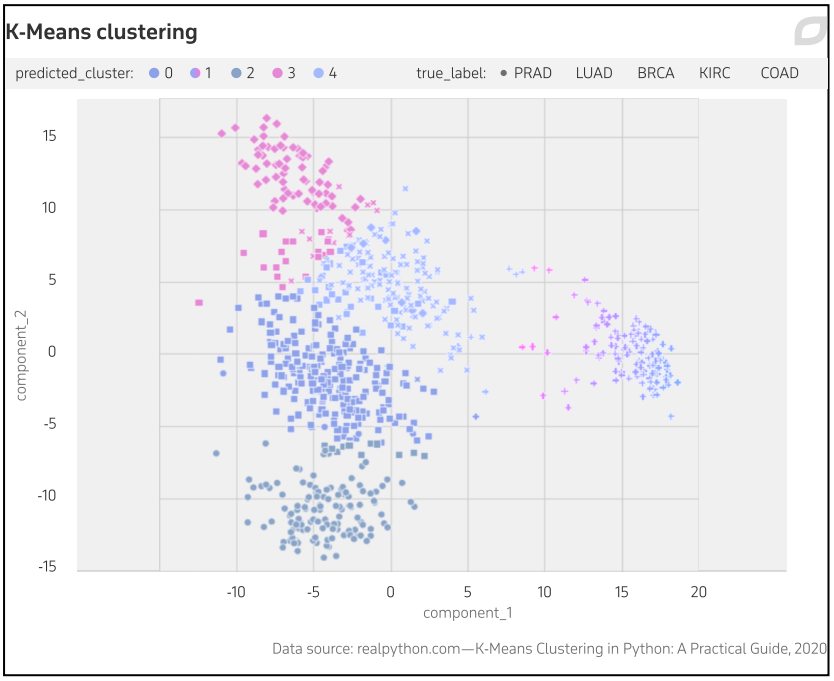
Clustering in unsupervised learning

The three most common applications for unsupervised learning are **clustering**, **dimensionality reduction,** and **association**.

Clustering performs *density estimation*, mapping the way that data is distributed in the dataset.

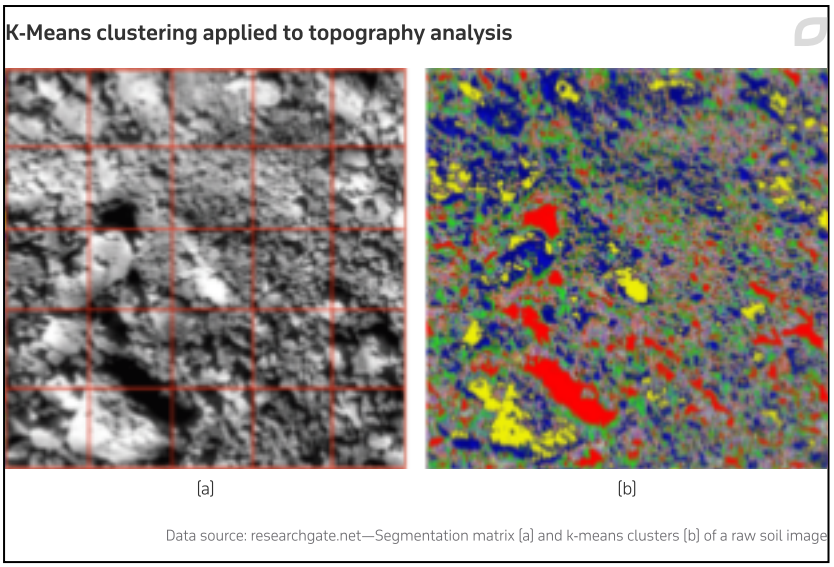


K-Means clustering is a popular implementation of this, and assigns data points to 'K groups'. The K value represents the volume of distinct and identifiable clusters that exist in a dataset, based on their similarity to each other.



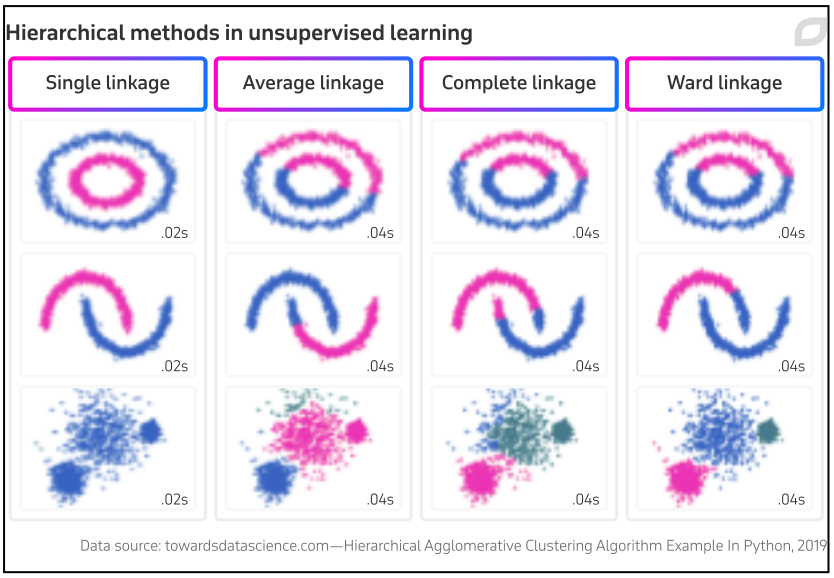
The higher the K value, the more groups there are, and the more diverse the possible outcomes and inferred relationships between the data points. Where the K value is lower, it's easier to determine direct relationships between the different groups.

K-Means is commonly used for image segmentation tasks, compression algorithms, and market segmentation, among other applications.



More complicated clusters, featuring more heterogeneous types of data, will require more sophisticated approaches to clustering in order to determine the distance between data points.

Several popular hierarchical methods (which can use Euclidian or Manhattan distance) are single (minimum) linkage; complete (maximum) linkage; Ward's linkage; and average linkage.



Dimensionality reduction

An unsupervised approach is helpful where the dataset is so large and variegated that manual pre-processing would be prohibitively expensive.

In such cases, it's possible to reduce the *dimensionality* of the data—to strip redundant data points or features down to a more essential and manageable dataset, so that a subsequent system (supervised or unsupervised) can realistically approach and interpret the data.

Methods used to reduce dimensionality include Principal Component Analysis (PCA); autoencoders in neural networks; and Singular Value Decomposition (SVD).

Association

Association rule learning identifies dependencies between data points, establishing an *antecedent* and a *consequent* data point. Algorithms used include Apriori, Frequent Pattern (FP) Growth, and the Apriori derivative Eclat (Equivalence Class Clustering and bottom-up Lattice Traversal).

Association is the most direct way of establishing correlations between data points, and will arguably be the initial approach in the development of shopping pattern analysis systems and other types of framework that prioritize one-to-one relationships in datasets.

Advantages of unsupervised learning

Data discoverability

Unsupervised approaches will not discard any data relationships that emerge above the thresholds and weights defined by the model architecture. This allows for the serendipitous discovery of unforeseen data patterns and relationships.

Improved anomaly detection

Besides the obvious notion of deriving unexpected business intelligence insights, the 'avid curiosity' of unsupervised approaches is particularly useful for fraud detection frameworks, where 'dumb' heuristic methods may have failed to spot anomalous behavior.

Automated preprocessing

Unsupervised training can help to automate labeling and classification. Where the data is consistent, it's possible to create reliable workflows with very high levels of labeling accuracy. Datasets processed in this way can be shunted on to supervised systems once all the exploitable relationships are discovered, and new data reviewed regularly for novel patterns and trends.

Disadvantages of unsupervised learning

Expensive and time-consuming

Unsupervised systems generally deal with high-volume data and use higher-impact methods (such as encoder/decoder systems in neural networks) to parse the data. Training sessions are longer and consume more power than supervised frameworks, because the opacity of the unlabeled dataset will often require GPU-level resources. By contrast, lighter mathematical formulae can be applied to labeled data in supervised systems.

Downstream errors can be frequent

Reliable error correction is difficult to perform on an unsupervised system, because there are no initial criteria to adhere to (i.e. starting labels and target classes), but merely the general imperative to classify data frequencies and types and to attempt to map any relationships that may exist between the data points.

If a training session takes a 'wrong turn' on the unlabeled data early enough, an unsupervised neural network is likely to build many subsequent wrong assumptions on this error, risking to have wasted time and resources, and necessitating a fatiguing number of re-attempts, with adjusted weights or modification of the data or other parameters.

Interpretation is usually necessary

An unsupervised system may indicate frequencies and new data relationships, but it cannot make 'sense' of them. As with supervised systems, it falls to us to define the value of mined relationships and discovered data, and to create subsequent or additional architectures dedicated to exploiting them.

Semi-supervised learning

Both supervised and unsupervised approaches have much to offer in a system that uses them as complementary rather than opposing technologies.

Unsupervised learning is capable of discovering profitable trends and data streams from otherwise ungovernable data lakes, while supervised learning can act as a refinement processing layer, informed by these insights and honing in on the 'difficult' discoveries unearthed by the unsupervised system.

Though these two approaches can effectively be used sequentially, it's possible to adopt a *semi-supervised* approach, which uses small amounts of human-labeled data as broad examples that the system can adopt to apply its own labels.

Exemplary data in semi-supervised learning

In semi-supervised learning, a model is initially trained on a dataset where only a small subset of the data contains hand-crafted labels. The trained model is then run on a completely unlabeled dataset, which will produce pseudo-labels—not in themselves necessarily accurate, but acting rather as placeholders.

Prior to a third training session, the pseudo-labels are linked to the hand-crafted labels and the data inputs from the labeled and unlabeled data are also linked.

Thus a final training session now has 'guideline' mapping and can produce effective labeling without excessive manual pre-processing.

Semi-supervised learning can be applied to many domains but is particularly well-suited for text classification and other functions of natural language processing. Amazon used this technique to achieve a 40x reduction in the amount of human-labeled data without sacrificing accuracy, according to their 2017 letter to the shareholders. Semi-supervised learning has also found favor in the aviation industry, to improve the quality of machine translation and to achieve improved results in sentiment analysis.