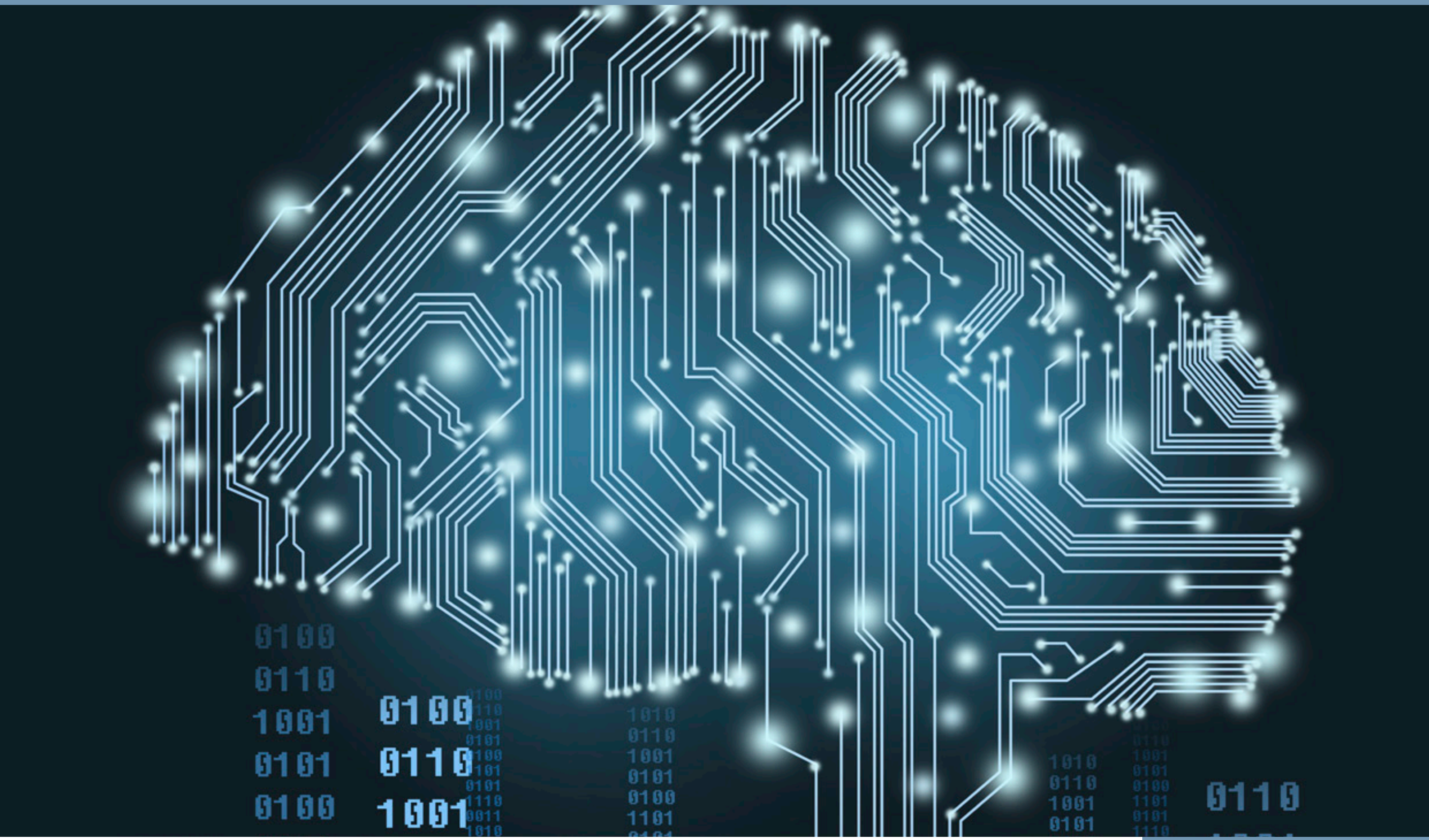
WHITE PAPER

The Machine Learning Landscape

A quick guide to the different types of learning



THE POWER TO KNOW®

Contents

What Is Machine Learning and Its Relationship to Artificial Intelligence (AI)? 1

Styles of Machine Learning 1

- Supervised 1
- Unsupervised 2
- Semi-supervised 2
- Reinforcement 2

What Can Machine Learning Do? 3

Machine Learning Algorithms 3

Machine Learning Methods 4

- Online Learning: "Data in Motion" 4
- Transfer Learning: The Key to Lifelong Learning in Machine Learning 5
- Ensemble Learning: The Wisdom of a Crowd 6
- Deep Learning: Model With Depth 7

Other Machine Learning Considerations 7

- Interpretability 7
- Automated Machine Learning (AML) 8
- Self-taught Learning 8

An End-to-End Machine Learning Environment 9

Conclusion 9

Learn More 10

References and Resources 10

What Is Machine Learning and Its Relationship to Artificial Intelligence (AI)?

Machine learning is a method of data analysis that automates analytical model building. It's considered a branch of artificial intelligence based on the idea that machines can learn and adapt through experience.

In 1959, Arthur Samuel, a pioneer in artificial intelligence (AI) and computer gaming, defined machine learning as a "field of study that gives computers the ability to learn without being explicitly programmed."

AI refers to a broader idea where a machine can execute tasks "smartly," emulating human tasks through learning and automation. It applies machine learning and other techniques to solving problems.

Both of these fields are used in all kinds of organizations and by data scientists who apply these methods to solve problems that range from product recommendations to fraud prevention. This paper is aimed at novice and intermediate data scientists who not only need to understand the basic styles and uses of machine learning, but also the methodologies for exploring data and modeling options.

> Machine learning is a "field of study that gives computers the ability to learn without being explicitly programmed."
>
> Arthur Samuel, 1959

Styles of Machine Learning

Today, there are four widely recognized styles of machine learning.

Supervised

Supervised learning algorithms make predictions based on a set of examples. For example, historical sales can be used to estimate the future prices. With supervised learning, you have data observations consisting of input variables and an output variable representing a label or a quantity of interest. You use an algorithm to analyze the training data to learn a function that maps the inputs to the output. This inferred function maps new, unknown examples by generalizing from the training data to anticipate results in unseen situations. Supervised learning is widely used for classification, regression and forecasting.

- **Classification**: When the data is used to predict a categorical variable, supervised learning is also called classification. This is the case when assigning a label or indicator; for example, "dog" or "cat" to an image. When there are only two labels, this is called binary classification. When there are more than two categories, the problem is called multi-class classification.
- **Regression**: When predicting continuous values, the problem becomes a regression problem.
- **Forecasting**: This is the process of making predictions about the future based on the past and present data. It is most commonly used to analyze trends. A common example might be estimation of the next year sales based on the sales of the current and previous years.

Characteristic: Data has known labels.

Examples: Insurance underwriting and fraud detection.

Unsupervised

With unsupervised learning, there is a focus on finding patterns and gaining insight from data. When performing unsupervised learning, the machine is presented with totally unlabeled data. It is asked to discover the intrinsic patterns that underlie the data, such as a clustering structure, a low-dimensional manifold, or a sparse tree and graph.

- **Clustering**: This involves grouping a set of data examples so that examples in one group (or one cluster) are more similar to each other (according to some criteria) than to those in other groups. This is often used to segment an entire data set into several groups. Analysis can be performed in each group to help users to find intrinsic patterns.
- **Dimension reduction**: This involves reducing the number of variables under consideration. In many applications, the raw data has very high dimensional features and some features are redundant or irrelevant to the task. Reducing the dimensionality helps to find the true, latent relationship.

Characteristic: Labels or output are unknown.

Examples: Customer clustering and market basket analysis.

Semi-supervised

Semi-supervised learning is a blend of supervised and unsupervised learning. The challenge with supervised learning is that labeling data can be expensive and time-consuming. If labeled data is limited, you can use unlabeled examples to enhance the data for supervised learning. Because the learning is not fully supervised in this case, we say the learning is semi-supervised. With semi-supervised learning, you use unlabeled examples with a small amount of labeled data to improve the learning accuracy.

Characteristic: Labels or output are known for a subset of data.

Examples: Medical predictions where tests and expert diagnoses are expensive and only part of the population receives them.

Reinforcement

Reinforcement learning analyzes and optimizes the behavior of an agent based on the feedback from the environment. Machines try different scenarios to discover which actions yield the greatest reward, rather than being told which actions to take. This type of learning has three primary components: the agent (the learner or decision maker), the environment (everything the agent interacts with) and actions (what the agent can do). The objective is for the agent to choose actions that maximize the expected reward over a given amount of time. Trial and error and delayed reward distinguishes reinforcement learning from other techniques.

Characteristic: Focus on making decisions based on previous experience.

Examples: Robotics, gaming, navigation, complex decision problems and reward systems.

What Can Machine Learning Do?

Machine learning can be used for many things, including data exploration, prediction, rule learning, decision making and policy making. This table presents a few examples.

What Can Machine Learning Do?

Data Exploration	Prediction	Rule Learning	Decision and Policy Making
• Clustering • Dimension reduction • Density estimation • Anomaly detection • Feature engineering	• Predict label: classification • Predict value: regression	• Indentify relational or contextual rules within data • Association rule mining • Learning classifier system	• Learn through trial and error to identify best action under different scenarios • Game playing • Handle control problems
• Often unsupervised learning • Also supervised and semi-supervised learning	• Supervised learning • Semi-supervised learning	• Supervised learning • Unsupervised learning • Semi-supervised learning	• Reinforcement learning

Figure 1: These are just a few of the different machine learning tasks.

Machine Learning Algorithms

An algorithm is a recipe for using logic and mathematical operations to solve a problem. In machine learning, the goal is to use input data to gain insight or construct a transformation (model) to classify observations or predict output values.

A machine learning algorithm is often referred to as a learner. They range from very simple and understandable, to very complex and obscure. Often, simple algorithms serve as a foundation for devising more effective, complex algorithms.

As machine learning algorithms are provided with more data, they build models in an iterative and adaptive fashion. However, the flexibility of machine learning algorithms can lead them to overfit the training data. Validating a model during the training process (with a holdout set or through cross-validation) is critical for constructing models that will generalize well for scoring (making predictions for) new observations.

Popular machine learning algorithms include:

- Linear/logistic regression (with regularization).
- Linear/kernel support vector machine.
- Decision tree.
- Decision forest.
- Gradient boosting.
- Neural network.
- Deep learning (deep neural networks, CNN, RNN, etc.).
- K-means/k-modes clustering.
- Gaussian mixture model (GMM) clustering.
- DBSCAN clustering.
- Hierarchical clustering.
- t-SNE.
- Principal component analysis (PCA).
- Singular value decomposition (SVD).
- Latent Dirichlet allocation (LDA).

For more details on these algorithms, check out this blog post.

Machine Learning Methods

In this section, we discuss four different methods of machine learning (online, transfer, ensemble and deep). All four can be any of the machine learning styles discussed previously (supervised, unsupervised, semi-supervised or reinforcement).

Online Learning: "Data in Motion"

Online learning uses data in motion. These models are updated as data arrives, and they do not store previous data. Some online learning algorithms are also "adaptive," meaning the models can change over time to follow or track changes in the data. They do so by gradually "forgetting" the outdated data and are well-suited for applications where the environment/model changes with time.

Batch (or offline) learning, which uses data at rest, can serve as a warm-start for online learning. In batch learning, one model is trained on all the data at once. Many online learning algorithms use a batch/offline algorithm (on a small batch of data) to initialize (warm-start) the model. This practice can significantly speed up the convergence.

There are two ways data in motion can be received.

1. Streaming: Data arrives sequentially via automatic streams. In this case, there is a limited observation window. Real-time processing of surveillance video is one example of streaming data.
2. Query: The algorithm interactively queries the user for labels/outcome of a subset of the data, obtains a label and updates the model. An interactive ranking system uses data queries.

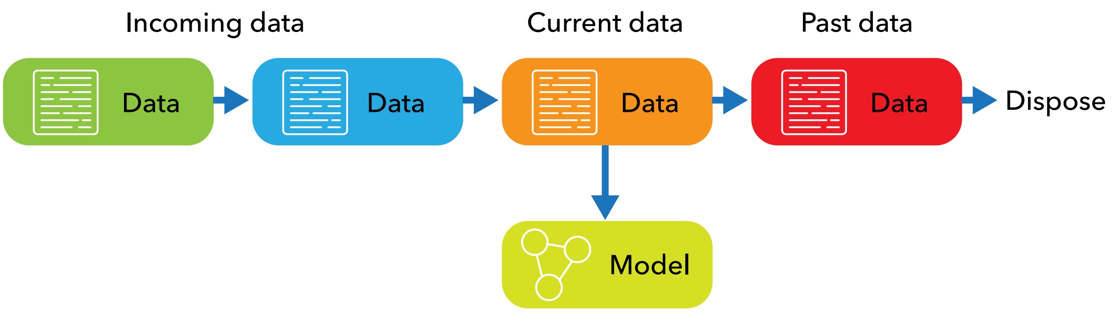


Figure 2: Online learning uses data in motion.

Transfer Learning: The Key to Lifelong Learning in Machine Learning

Transfer learning applies knowledge from one domain to another. It reuses old data, models and parameters for new problems, which is crucial to lifelong learning of machine learning models. Transfer learning comes naturally to people. For example, when learning a new language, we apply the knowledge (vocabulary, grammar, etc.) from languages we already know. The closer the two languages are, the easier it is to transfer the knowledge. This is why it is usually easier for an English speaker to learn Italian and French than Chinese and Japanese.

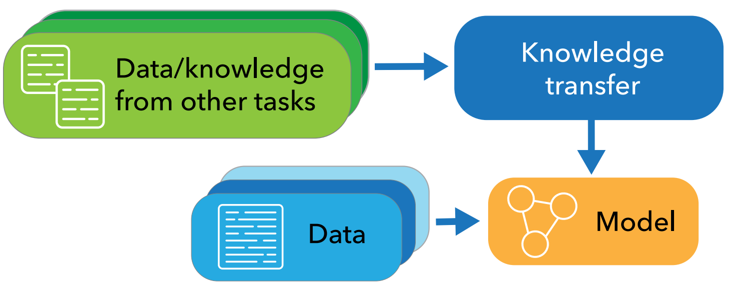


Figure 3: Transfer learning uses many data sources and parameters to learn.

Single-task transfer learning learns the model based on only data related to the current problem. Multi-task transfer learning learns several models simultaneously. Multi-task learning is a special case of transfer learning, where the data and/or model information are shared between several related tasks, while the models for all tasks are learned simultaneously. For example, in medical applications, you may want to build separate models for different subpopulations (based on family/personal history and gene expressions) for better prediction and personalized treatment.

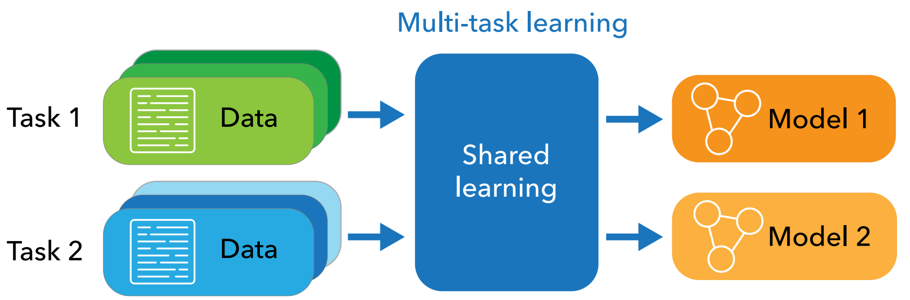


Figure 4: In multi-task learning, data and/or model information are shared between related tasks and the models for all tasks are learning simultaneously.

Ensemble Learning: The Wisdom of a Crowd

While a single learner model uses only one learner (algorithm), ensemble learning uses multiple learner/experts. Common ensemble algorithms include gradient boosting, bootstrap aggregating (bagging), forest (decision forest), stacked ensembles and super learner.

Ensemble learning algorithms can combine weaker learners (in many cases learners with prediction accuracy only slightly better than random guessing) to produce a strong, accurate model. There are many ways to combine these weaker learners, including majority voting, weighted sum, follow-the-leader and stacking. The unique hierarchical structure of ensemble learning allows the learning of complex models by using a collection of simple, off-the-shelf learners.

Common criticisms of ensemble learning include:

- Challenging to deploy due to the need for multiple learners.
- Computationally more expensive than a single learner.
- Less interpretable than a single learner.

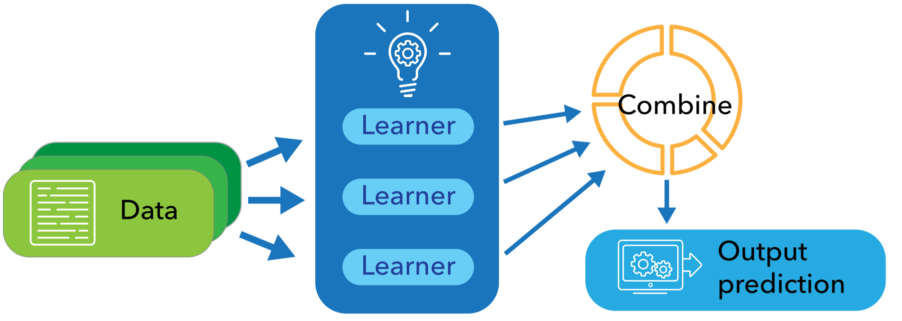


Figure 5: Ensemble learning use multiple learners.

Deep Learning: Model With Depth

Deep learning contains multiple layers and learns hierarchical or multi-scale representations of the data. This is in contrast to "shallow learning" which is simply the application of standard machine learning modeling algorithms. Shallow learning typically requires feature engineering to ensure appropriate representation of inputs, while deep learning inherently learns the features during training.

Common deep learning architectures include convolutional neural networks (CNNs), recurrent neural networks (RNNs) and stacked auto-encoders.

When should you use deep learning?

Deep learning models are usually complex and nonlinear. Thus, deep learning is most efficient when the underlying model is also complex and non linear. Successful training of a deep model requires a large amount of training data (due to the inherent complexity of the model). When only a small amount of training data is available, you should choose a shallow model and train with available data or adopt a deep model, but use transfer learning to warm-start with related data or model information.

Common criticisms of deep learning include:

- Requires a large amount of training data.
- Computationally expensive and the model training can be very time-consuming.
- Difficult to interpret – often treated as a "black-box" model.

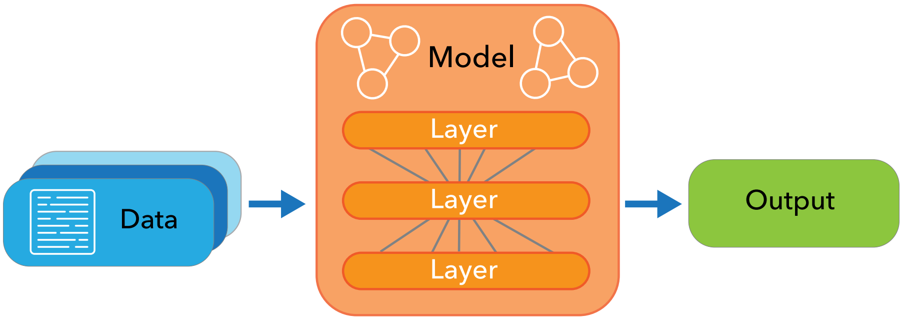


Figure 6: Deep learning models require large amounts of training data.

Other Machine Learning Considerations

Interpretability

The interpretability of a machine learning model is the degree it can be understood and explained by people. For example, if we build a model for disease prediction, we would not only want the prediction to be accurate, but we would also want to understand the reasons behind the prediction (e.g., what factors contribute to the diagnosis and how much they contribute).

Another important aspect of interpretability is the ability to test the model to see its sensitivity to data changes and its robustness to adversarial corruptions. This is critical for applications with legal, financial or security implications.

An interpretable model usually exhibits most or all the following attributes:

- Transparent: Model can explain its reasoning (white box).
- Simple: Simple enough to be tested and examined by a human.
- Informative: Conveys useful information.
- Trustworthy: Trust/confidence that the model will work in the real world.

As AI and machine learning become integral parts of modern life, the study of interpretability becomes an urgent task, not only for providing better understanding of the models but also for gaining public trust in AI and machine learning themselves. This SAS blog series explores topics related to model interpretability.

Automated Machine Learning (AML)

Applying machine learning to real-life problems can be difficult. Each step – data preparation, model selection and model training – requires great expertise from human analysts. Recent research in automated machine learning helps streamline the process and brings more insight to human analysts in the process.

- **Automated feature engineering**. Feature engineering is the process of designing and producing inputs to a machine learning algorithm. It is difficult and time-consuming, yet it's a critical process that greatly impacts the success of a machine learning model. Automated feature engineering aims at selecting and/or embedding features that preserve relevant information while minimizing redundancy.
- **Automated model selection**. With so many machine learning models available (and more being developed), it is often difficult to know which model works best for a data set. Automated model selection can build, analyze and compare multiple models on the data (or even build different models on different subsets of the data), which helps human analysts make informed decisions on the best model(s) to use for a specific problem.
- **Automated hyperparameter tuning**. Many machine learning algorithms are governed by hyperparameters that require careful tuning (e.g., learning step size and penalization weight). Automated hyperparameter tuning techniques test and optimize the hyperparameters without the need to manually experiment.

Self-taught Learning

Self-taught learning was first proposed as a framework of unsupervised feature learning plus supervised classification by Raina et al. from Stanford University in 2007. The basic idea is to first learn a feature extractor from a large set of unlabeled data (unsupervised learning), extract features of a small set of labeled data, and then learn a classifier based on the features from the labeled data (supervised learning).

Generally it can be seen as a special case of semi-supervised learning, though classical semi-supervised learning methods often assume both labeled and unlabeled data come from the same distribution, while self-taught learning does not make such assumptions.

An End-to-End Machine Learning Environment

There will always be debates about whether it's better to build or buy a machine learning environment. SAS provides a unifying platform for machine learning and an end-to-end analytics environment with everything needed to create accurate predictive models and solve difficult business problems. It includes:

- **Data preparation**. Ingest, merge and append from a variety of sources; apply table or column-based transformations; and write custom code to wrangle your data.
- **Exploration and interactive modeling**. Visually examine variable distributions, trends and relationships. Build models interactively with immediate assessment results.
- **Automated machine learning pipelines**. Automatically chain together machine learning tasks for feature engineering, model training/tuning, ensembling and model assessment and comparison; insert custom code (SAS and other languages); collaborate through shareable nodes/pipelines; and deploy models with a single click.
- **Deployment and real-time scoring**. Deploy and manage versions of your models; publish models for scoring directly in database; incorporate business rules; score streaming data; and monitor model health and retrain.
- **Programming in SAS or other popular languages**. Write programs in SAS, Python, R, Java or Lua, or invoke a REST API to use all of the machine learning capabilities in your own applications.

Conclusion

From detecting fraud and terrorist threats to producing driverless cars and reducing manufacturing defects, machine learning is a science of modeling and data whose time has come. Machine learning gives organizations the potential to make more accurate data-driven decisions and to solve problems that have stumped traditional analytical approaches, such as those involving new sources of unstructured data, including graphics, sound, videos, and other high-dimensional, machine-generated data.

It is simple in theory (large amounts of data are run through a model until it finds enough patterns to make an accurate prediction about that data, and the trained model is then used on new data to make new predictions). But the actual application can become quite complex.

This paper provided an introduction to the machine learning landscape of today. It outlined the different types of learning styles, algorithms and methods that can be used, depending on factors involved and the problem at hand. In addition, we covered other considerations such as data and model interpretability, automation and new techniques like self-taught learning.

The SAS® Platform has provided machine learning capabilities for more than four decades, whether it was the k-nearest neighbor algorithm first included with SAS® Enterprise Miner™ or newer algorithms like factorization machines and latent Dirichlet allocation in SAS Visual Data Mining and Machine Learning. SAS programming APIs and graphical user interfaces help you build machine learning models and implement iterative machine learning processes. Our comprehensive selection of machine learning algorithms helps you creatively solve your most challenging problems and is included in many SAS products.

Learn More

SAS Visual Data Mining and Machine Learning is our newest solution for machine learning. Learn more at sas.com/vdmml.

References and Resources

Resources for developers

- Developer.sas.com: documentation, examples and access to free trials
- Github.com/sassoftware: examples of machine learning with SAS

Publications

"Statistical Modeling, The Two Cultures." Leo Breiman.
http://projecteuclid.org/euclid.ss/1009213726

"Fifty Years of Data Science." David Donoho.
http://courses.csail.mit.edu/18.337/2015/docs/50YearsDataScience.pdf.

Pattern Recognition and Machine Learning. Christopher Bishop.
http://www.springer.com/us/book/9780387310732.

"Machine Learning with SAS® Enterprise Miner™." A SAS White Paper.
https://www.sas.com/content/dam/SAS/en_us/doc/whitepaper1/machine-learning-with-sas-enterprise-miner-107521.pdf

Posts and reposts

Machine Learning: What it is and why it matters.
https://www.sas.com/en_us/insights/analytics/machine-learning.html

Deep Learning: What it is and why it matters.
https://www.sas.com/en_us/insights/analytics/deep-learning.html

Which machine learning algorithm should I use?
https://blogs.sas.com/content/subconsciousmusings/2017/04/12/machine-learning-algorithm-use/

A curated list of awesome machine learning frameworks, libraries and software.
https://github.com/josephmisiti/awesome-machine-learning

Quick reference tables for machine learning best practices and algorithm usage.
https://github.com/sassoftware/enlighten-apply/tree/master/ML_tables

Interpretability in Machine Learning. https://blogs.sas.com/content/tag/interpretability

To contact your local SAS office, please visit: sas.com/offices

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies. Copyright © 2018, SAS Institute Inc. All rights reserved. 109909_G80817.0818