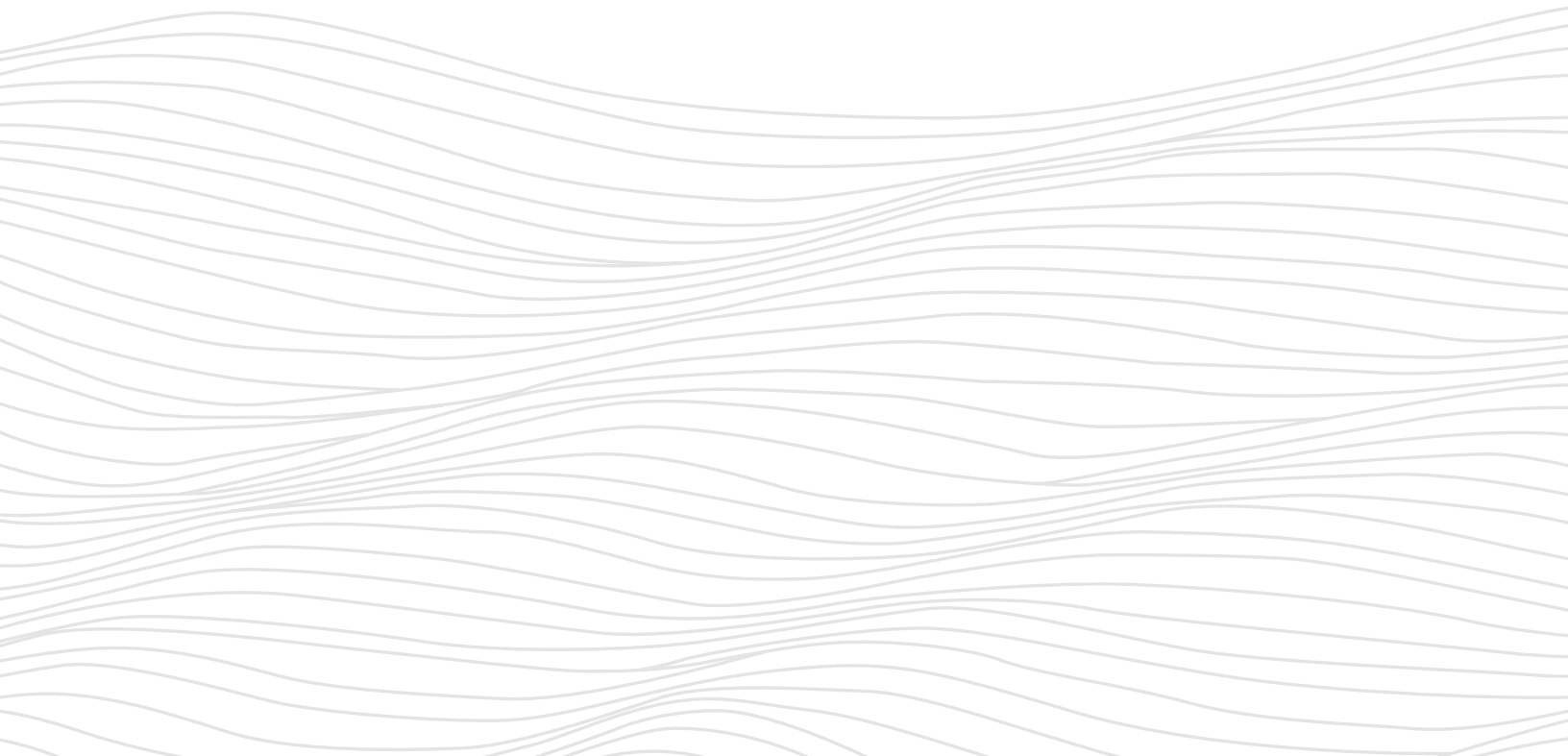




WHITE PAPER



DEEP LEARNING FOR BUSINESS LEADERS



DEEP LEARNING: AN OVERVIEW

Deep Learning is a rapidly growing discipline that models high-level patterns in data as complex multi-layered networks. Because it is the most general way to model a problem, Deep Learning has the potential to solve the most challenging problems in machine learning and artificial intelligence.

In principle, Deep Learning is an approach that is not limited to a specific machine learning technique; however, in actual practice, most Deep Learning applications use artificial neural networks (ANN) with multiple hidden layers, also called deep neural networks (DNN). Of the many different types of ANN, the most pervasive is the feed-forward architecture, where information flows in a single direction from the raw data, or input layer to the prediction, or output layer. The most popular technique for training feed-forward networks is the back-propagation method, which uses errors at the highest level of abstraction to adjust weights at lower levels.

The foundation techniques of Deep Learning date back to the 1950s, but interest among managers has surged in the last several years, led by reports in the *New York Times* and *The New Yorker*. Companies like Microsoft and Google use Deep Learning to solve difficult problems in areas such as speech recognition, image recognition, 3-D object recognition, and natural language processing. Deep Learning requires considerable computing power to construct a useful model; until recently, the cost and availability of computing limited its practical application. Moreover, researchers lacked the theory and experience to apply the technique to practical problems; given available time and resources, other methods often performed better.

The advance of Moore's Law has radically reduced computing costs, and massive computing power is widely available. In addition, innovative algorithms provide faster and more efficient ways to train a model. And, with more experience and accumulated knowledge, data scientists have more theory and practical guidance to drive value with Deep Learning.

While news reports tend to focus on futuristic applications in speech and image recognition, data scientists are using Deep Learning to solve highly practical problems with bottom-line impact:

- Payment systems providers use Deep Learning to identify suspicious transactions in real time.
- Organizations with large data centers and computer networks use Deep Learning to mine log files and detect threats.
- A Class One railroad in the United States uses Deep Learning to mine data transmitted from railcars to detect anomalies in behavior that may indicate part failure.
- Vehicle manufacturers and fleet operators use Deep Learning to mine sensor data to predict part and vehicle failure.
- Deep Learning helps companies with large and complex supply chains predict delays and bottlenecks in production.
- Researchers in the health sciences have used Deep Learning to solve a range of problems, including detecting the toxic effects of chemicals in household products.

With increased availability of Deep Learning software and the skills to use it effectively, the list of commercial applications will grow rapidly in the next several years.

PROS AND CONS OF DEEP LEARNING

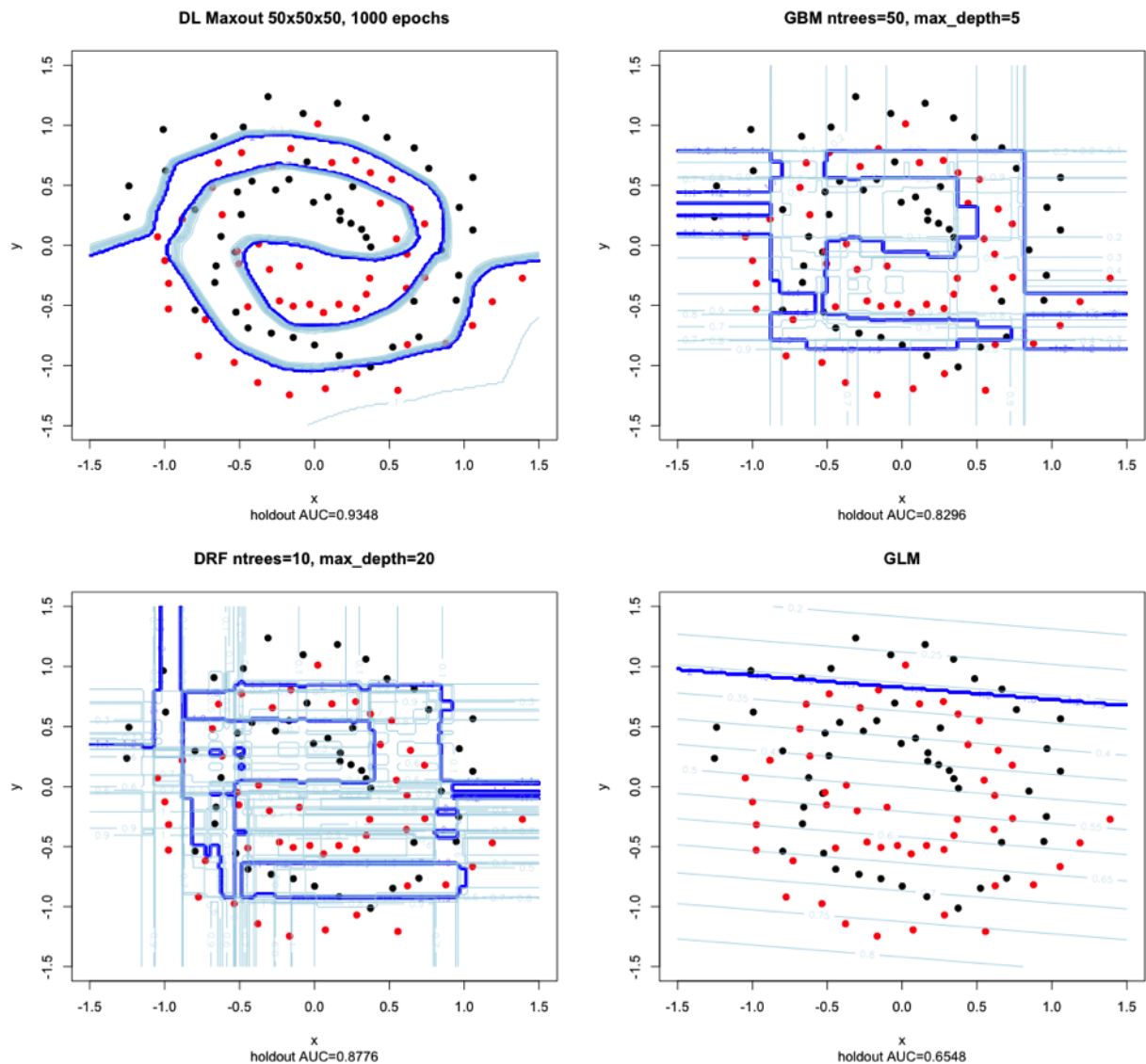
Relative to other machine learning techniques, Deep Learning has four key advantages: its ability to detect complex interactions among features; its ability to learn low-level features from minimally processed raw data; its ability to work with high-cardinality class memberships; and its ability to work with unlabeled data. Taken together, these four strengths mean that Deep Learning can produce useful results where other methods fail; it can build more accurate models than other methods; and it can reduce the time needed to build a useful model.

Deep Learning detects interactions among variables that may be invisible on the surface. Interactions are the effect of two or more variables acting in combination. For example, suppose that a drug causes side effects in young women, but not in older women. A predictive model that incorporates the combined effect of sex and age will perform much better than a model based on sex alone. Conventional predictive modeling methods can measure these effects, but only with a lot of manual hypothesis testing. Deep Learning detects these interactions automatically, and does not depend on the analyst's expertise or prior hypotheses. It also creates non-linear interactions automatically and can approximate any arbitrary function with enough neurons, especially when deep neural networks are used.

With conventional predictive analytics methods, success depends heavily on the data scientist's ability to use feature engineering to prepare the data, a step that requires considerable domain knowledge and skill. Feature engineering is also takes time. Deep Learning works with minimally transformed raw data; it learns the most predictive features automatically, and without making assumptions about the correct distribution of the data.



The figure below illustrates the power of Deep Learning. The four charts demonstrate how different techniques model a complex pattern. The lower right hand chart shows how a Generalized Linear Model (GLM) fits a straight line through the data. Tree-based methods, such as Random Forests (DRF) and Gradient Boosted Machines (GBM), in the lower left and upper right, respectively, perform better than GLM; instead of fitting a single straight line, these methods fit many straight lines through the data, markedly improving model “fit”. Deep Learning, shown in the upper left, fits complex curves to the data, delivering the most accurate model.



Deep Learning works well with what data scientists call high-cardinality class memberships, a type of data that has a very large number of discrete values. Practical examples of this type of problem include speech recognition, where a sound may be one of many possible words; image recognition, where a particular image belongs to a large class of images; or recommendation engines, where the optimal item to offer can be one of many.

Another strength of Deep Learning is its ability to learn from unlabeled data. Unlabeled data lacks a definite “meaning” pertinent to the problem at hand; common examples include untagged images, videos, news articles, tweets, computer logs, and so forth. In fact, most of the data generated in the information economy today is unlabeled. Deep Learning can detect fundamental patterns in such data, grouping similar items together or identifying outliers for investigation.

Deep Learning also has some disadvantages: compared to other machine learning methods, it can be very difficult to interpret a model produced with Deep Learning. Such models may have many layers and thousands of nodes; interpreting each of these individually is impossible. Data scientists evaluate Deep Learning models by measuring how well they predict, treating the architecture itself as a “black box.”

Critics sometimes object to this aspect of Deep Learning, but it’s important to keep in mind the goals of the analysis. For example, if the primary goal of the analysis is to explain variance or to attribute outcomes to treatments, Deep Learning may be the wrong method to choose. It is, however, possible to rank the predictor variables based on their importance, which is often all the data scientists look for. Partial dependency plots offer the data scientist an alternative way to visualize a Deep Learning model.

Deep Learning shares with other machine learning methods a propensity to overlearn the training data. This means that the algorithm “memorizes” characteristics of the training data that may or may not generalize to the production environment where the model will be used. This problem is not unique to Deep Learning, and there are ways to avoid it through independent validation.

Because Deep Learning models are complex, they require a great deal of computing power to build. While the cost of computing has declined radically, computing is not free; for simpler problems with small data sets, Deep Learning may not produce sufficient “lift” over simpler methods to justify the cost and time.

Complexity is also a potential issue for deployment. Netflix never deployed the model that won its million-dollar prize because the engineering costs were too high. A predictive model that performs well with test data but cannot be implemented is useless.



SOFTWARE REQUIREMENTS

Software for Deep Learning is widely available, and organizations seeking to develop a capability in this area have many options. The following are key requirements to keep in mind when evaluating Deep Learning:

OPEN SOURCE: Nobody owns the math; Deep Learning methods and algorithms are all in the public domain, and proprietary software implementations do not perform better than open source software. While it makes sense for an organization to pay for support, training and custom services, there is no reason to pay a vendor for a capability that is freely available in open source software.

DISTRIBUTED COMPUTING PLATFORM: In the face of growing data volumes and computing workloads, software for machine learning must be able to leverage the power of distributed computing. In practice, this means two things:

- Machine learning software must be able to work with distributed data, including data stored in the Hadoop Distributed File System (HDFS) or the Google File System.
- Machine learning software should be able to distribute its workload over many machines, enabling it to scale without limit.

MAINSTREAM HARDWARE: Some Deep Learning platforms rely on specialized HPC machines based on GPU chips and other exotic hardware. These platforms are more difficult to integrate with your production systems, since they cannot run on the standard hardware used throughout your organization.

HADOOP INTEGRATION: Your data is in Hadoop, and should stay there. Many Deep Learning packages will work with your data only if you extract it to an edge node or an external machine. This physical data movement takes time and, in the case of curated data, may violate IT standards and policies. Your Deep Learning software should integrate seamlessly with popular Hadoop distributions, such as Cloudera, MapR and Hortonworks. In practice this means that you should be able to distribute learning across the Hadoop cluster, run the software under YARN and ingest data from HDFS in a variety of formats.

CLOUD INTEGRATION: Your organization may or may not use the cloud, but if cloud is part of your architecture you will need Deep Learning software that can run in a variety of cloud platforms, such as Amazon Web Services, Microsoft Azure and Google Cloud.

FLEXIBLE INTERFACE: Your data scientists use many different software tools to perform their work, including analytic languages like R, Python and Scala. Your Deep Learning software should integrate with a variety of standard analytic languages, and should provide business users with a means to visualize predictive performance, model characteristics and variable importance.



SMART ALGORITHMS: While nobody owns the math, Deep Learning differs in the depth of features they support. The capabilities listed below save time and manual programming for the data scientist, which leads to better models.

- Automatic standardization of the predictors
- Automatic initialization of model parameters (weights & biases)
- Automatic adaptive learning rates
- Ability to specify manual learning rate with annealing and momentum
- Automatic handling of categorical and missing data
- Regularization techniques to manage complexity
- Automatic parameter optimization with grid search
- Early stopping based on validation datasets
- Automatic cross-validation

RAPID MODEL DEPLOYMENT: Deep Learning produces highly complex models. Programming these models from scratch is a significant effort; that is why Netflix never implemented its prize-winning model and organizations report cycle times of weeks to months for production scoring applications. Your Deep Learning software should generate plain (Java) code that captures the complex math in an application you can deploy throughout the organization.

DEEP LEARNING IN ACTION

FRAUD DETECTION: An online payments company handles more than \$10 billion in money transactions every month. At that volume, small improvements in fraud detection and prevention translate to significant bottom-line impact: each one percent reduction in fraud saves the company \$1 million per month. Using H2O, the company found that Deep Learning works well for this use case.

Working with a dataset of 160 million records and 1,500 features, the company's machine learning team sought to identify the best possible model for fraud detection. Thanks to H2O's scalability, individual model runs ran quickly; this enabled the team to test a wide range of model architectures, parameter settings and feature subsets. The models retain their predictive power over time, which reduces the need for model maintenance and updates. Putting the model into production is a straightforward process, since H2O exports a Plain Old Java Object (POJO) that will run anywhere that Java runs.

RESUME MATCHING: A global staffing, human resources and recruiting company receives thousands of resumes daily from job seekers. Matching these candidate resumes with one of the hundreds of open positions was a major problem. Highly trained placement professionals spent hours poring through resumes, seeking out the right candidates for a position. This manual process necessarily led to duplicated effort and missed opportunities.

The company's machine learning team attacked this problem by first passing millions of resumes through text analytics that extracted keywords reflecting the candidate's skills. Using this data as input for Deep Learning, the team used H2O to predict the best position for the candidate (from up to fifty possibilities).

With H2O's rapid model deployment, the team is able to deploy the Deep Learning model to classify resumes in real time, as they arrive. This enables placement professionals to immediately act on the good matches, "striking while the iron is hot." Overall, the company estimates that this Deep Learning-enabled system contributes \$20-\$40 million per year in incremental profit through higher success rates in placement and time savings for professional staff.

ABOUT H2O.AI

H2O is the #1 open source machine learning platform for smarter applications. H2O.ai is the Silicon Valley software company supporting and developing H2O. Leading insurance, healthcare and financial services companies are using H2O to make smarter predictions about churn, pricing, fraud and more. H2O.ai is fostering a grassroots movement of systems engineers, data scientists, data developers and predictive analysts to move machine learning forward. A rapidly growing community of H2O users is now active in more than 5000 organizations worldwide. H2O.ai is a Gartner Cool Vendor in Data Science for 2015.